

Gwangju Institute of
Science and Technology

School of Information and Communications



WorldLand™, My AI Network™

핵심 기술 연구 결과

2024. 01. 08.

Department : EECS

Professor : Heung-No Lee

Research Team : INFONET, LiberVance

MY AI Network Project Progress Report

LiberVance Co. Ltd.

2024. 1. 9.(Tue)

□ Project Overview

- Project Title: MY AI Network
- Goal: To address the challenges in AI development, which include securing high-quality training data and resolving collaborative learning issues*, this project aims to improve data value and model performance through mutual cooperation while mitigating issues. This project is implemented based on decentralized AI, where individuals own their AI models**.

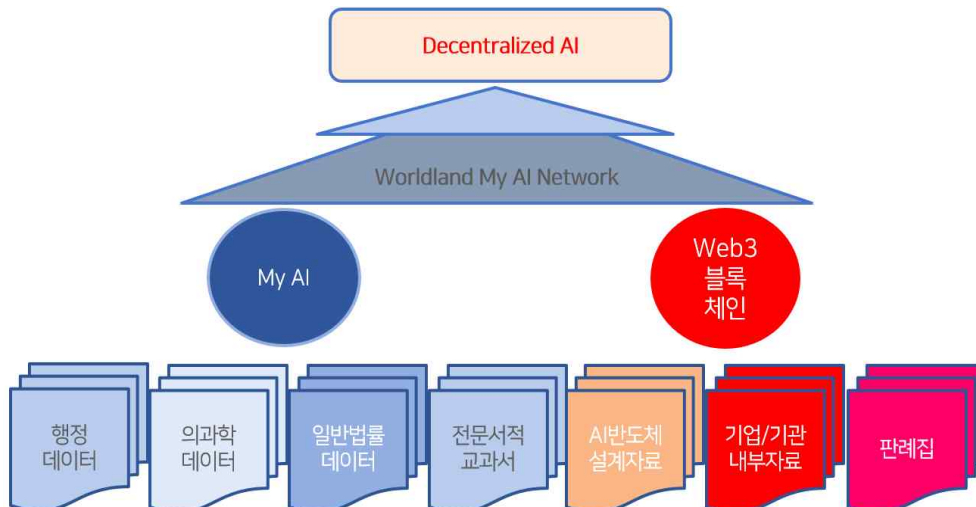
* Chat GPT model, which are representative of centralized AI, can cause several issues coming from their centralized nature, leading to potential social problems.

(i.e. bias, inaccuracy, security, and difficulties in protecting copyright rights)

** Through the application of Web3 blockchain, we can expect enhanced transparency, security, and trustworthiness, as well as the digitization and economic improvement of digital content creation through smart contracts and tokens. Furthermore, optimization of AI utilization governance processes can be expected within DAOs.

□ Service Overview

- Providing Decentralized AI*** services that encourage mutual cooperation for individual AI owners to secure data sovereignty, productivity, and competitiveness.



*** Decentralized AI: Ensuring distributed ownership and control of AI models, individual data protection and safety, autonomous collaboration and interaction, and automated transactions with transparency in governance through smart contracts.

□ **Current Project Progress**

- **(Technological Differentiation)** Secured papers/patents on applications, specialization, and foundational technologies^{****} (such as **generative AI training**, hyperspectral identity authentication, Verifiable Coin Toss, ECCPoW, DeSecure blockchain, zero-knowledge proofs, quantum-resistant cryptography, etc.) to ensure expertise and technological differentiation for service implementation.
- **Launch of Web3 network (WorldLand)** capable of implementing the service.
 - Selected for the Ministry of SMEs and Startups' Technology Innovation Development Program (TIPS) and ongoing technology development ('22.7 - '24.6).
 - Global mainnet launch on August 8, 2023 (current average block creation time 10.4 seconds, 1.23 million generated blocks, 673 connected wallets, 143 daily nodes).
 - Mainnet launch event on August 10, 2023, attended by over 60 relevant experts from academia, industry, and government (including speeches by Congressman Yang Hyang-ja, Chairman of the Korea ICT Convergence Association, DSRV CEO, Blockchain Factory CEO, Blockchain Today CEO, etc.).
- **** Secured 2 domestic patents, with 3 patent technology transfers planned (including knowledge SNARK proof system and method), 300+ domestic and international research papers from INFONET Lab, 9 international patent applications with 7 registrations, 7 domestic patent applications with 22 registrations, and 3 technology transfers.

□ **Future Works**

- **(DAO)** Performing the role of the WorldLand Ecosystem Growth Foundation through WorldLand DAO, which makes important decisions through stake proof.
- **(WorldLand Eco)** Providing decentralized services to various ecosystem players, including DAO foundations, investors, AI network developers, node operators, users, dApp operators, etc., through a decentralized ecosystem.

WorldLand™ My AI Network™ 기술 연구 결과

작성일 : 20240108

작성자 : 강주성

목적:

- 연구실 My AI Network 및 WorldLand 관련 논문 목록 조사 및 수합

핵심 기술 키워드 및 연구 결과 (논문) :

1. WorldLand™

- i. **Heung-No Lee, Young-Sik Kim, Dilbag Singh, and Manjit Kaur**, “Green Bitcoin: Global Sound Money,” *The Journal of Digital Assets*, Vol.1, No. 1, pg. 33-47, October, 2022. (The Inaugural Issue of the Journal of Digital Assets, No Impact Factor) ([PDF](#))
- ii. **Hyongsung Kim, Jehyuk Jang, Sangjun Park, and Heung-No Lee***, “Error-Correction Code Proof-of-Work on Ethereum”, *IEEE Access*, Vol. 9, pp. 135942-135952, Sep 2021. doi: <https://doi.org/10.1109/ACCESS.2021.3113522> (Impact Factor: 3.367, IITP & Do-Yak project) Paper: ([Open Access](#))
- iii. **Sangjun Park, Nam Yul Yu and Heung-No Lee**, “An Information-Theoretic Study for Joint Sparsity Pattern Recovery with Different Sensing Matrices”, *IEEE Transactions on Information Theory*, Vol. 63, no. 9, pp. 5559-5571, May. 2017. (Impact Factor: 2.679, Do-Yak Project) ([PDF](#)) DOI: [10.1109/TIT.2017.2704111](https://doi.org/10.1109/TIT.2017.2704111)
- iv. **Sangjun Park, Haeung Choi and Heung-No Lee**, "Time-Variant Proof-of-Work Using Error-Correction Codes," preprint on arXiv, June 2020, <https://arxiv.org/abs/2006.12306> .
- v. **Jehyuk Jang and Heung-No Lee**, “Profitable Double-Spending Attacks,” *Applied Sciences*, 10, 8477, .Nov. 2020. doi: <https://doi.org/10.3390/app10238477>. (IF: 2.474, Do-Yak and IITP) Paper: ([Open access](#))
- vi. **Hyunjun Jung, Heung-No Lee***, “ECCPoW: Error-Correction Code Based Proof-of-Work for ASIC Resistance”, *Symmetry*, June. 2020, 12(6), 988 (impact Factor: 2.143, Do-Yak project) ([PDF](#))
- vii. **Seungmin Kim, Gyeongdeok Maeng, Heung-No Lee**, "Smart Contract-Based Checkpoint for Initial PoW Network Security," 2023 Fourteenth International Conference on Ubiquitous and Future Networks (ICUFN 2023), July 2023, pp.89-93.
- viii. **맹경덕, 김승민, 이흥노**, "지속 가능한 경제와 암호화폐," 한국핀테크학회 2023 동계학술대회, Feb. 2023.

2. My AI Network™

- i. **Jian Su; Zhenlong Liao, Zhengguo Sheng, Alex X. Liu, Dilbag Singh, and Heung-No Lee**, “Human Activity Recognition Using Self-powered Sensors Based on Multilayer Bi-directional Long Short-Term Memory Networks” *IEEE Sensors Journal*, (Impact factor: 4.325, Category Instruments & Instrumentation, JCR Quartile: Q1: JIF Percentile: 78.91, Do-Yak project). ([Link](#))
- ii. **Kiwon Yang, Jusung Kang, Jehyuk Jang and Heung-No Lee**, “Multimodal Sparse Representation-Based Classification Scheme for RF Fingerprinting,” *IEEE Communications Letters*, Vol. 23, Issue 5, pp. 867-870, May 2019. (Impact Factor: 2.723, Doyak Project) ([PDF](#)) ([Used DB](#)) ([Source Code](#))
- iii. **Manjit Kaur, Dilbag Singh, Vijay Kumar, Heung-No Lee**, “MLNet: Metaheuristics-based Lightweight Deep Learning Network for Cervical Cancer Diagnosis”, *IEEE Journal of Biomedical and Health Informatics*, (Impact factor: 7.021, Category Mathematical & Computational Biology, JCR Quartile: Q1: JIF Percentile: 93.86). ([Link](#))
- iv. **Dilbag Singh, Manjit Kaur, Jazem Mutared Alanazi, Ahmad Ali AlZubi, Heung-No Lee**, “Efficient evolving deep ensemble medical image captioning model”, *IEEE Journal of Biomedical and Health Informatics*, (Impact factor: 7.021, Category Mathematical & Computational Biology, JCR Quartile: Q1: JIF Percentile: 93.86). ([Link](#))
- v. **Dilbag Singh, Manjit Kaur, Vijay Kumar, Mohamed Yaseen Jabarulla, and Heung-No Lee**, “Artificial Intelligence-based Cyber-Physical System for Severity Classification of Chikungunya Disease”, *IEEE Journal of Translational Engineering in Health and Medicine*, (Impact factor: 3.316, Category: Engineering, Biomedical, JCR Quartile: Q2: JCR Percentile: 51.12). ([Open Access](#))
- vi. **Dilbag Singh, Aravinda C.V, Manjit Kaur, Meng Lin, Jyoti Shety, Vikram Raju Reddicherla, and Heung-No Lee**, “DKNet: Deep Kuzushiji Characters Recognition Network” *IEEE Access*, (Impact factor: 3.476, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 62.14, IITP & Do-Yak project). (Open Access)
- vii. **Jin-Taek Seong and Heung-No Lee**, “Predicting the Performance of Cooperative Wireless Networking Schemes with Random Network Coding,” *IEEE Transactions on Communications*, vol. 62, no. 8, pp. 2951-2964, Aug. 2014. (Impact Factor: 1.979, Do-Yak Project) (pdf)
- viii. **Sang-Seon Byun, Ilango Balasingham, Athanasios V. Vasilakos, and Heung-No Lee** “Computation of an Equilibrium in Spectrum Markets for Cognitive Radio Networks,” to appear *IEEE Transactions on Computers*, Published on-line first on Aug. 30th, 2012. (Impact Factor: 1.473, Do-Yak Project, Haek-Sim Project)(pdf)
- ix. **Cheng-Chun Chang, Zhi-Hong Mao, and Heung-No Lee**, “MB iterative decoding algorithm on systematic LDGM codes: Performance evaluation,” *Signal Processing*, vol. 90, issue. 1, pp. 373-377, 2010.(Impact Factor: 2.238) (pdf)
- x. **Rahman S M Wahidur, Ishmam Tashdeed, Manjit Kaur, and Heung-No Lee**, “Enhancing Zero-Shot Crypto Sentiment with Fine-tuned Language Model and Prompt Engineering”, in *IEEE Access*, Early Access, doi:

10.1109/ACCESS.2024.3350638. (URL)

3. Hyper Spectrometer Imaging (HSI) system.

- i. **Cheolsun Kim, Pavel Ni, Kang Ryeol Lee, and Heung-No Lee***, “Mass production-enabled computational spectrometers based on multilayer thin films”, Accepted in Scientific Reports, (Impact Factor: 4.380, Category: Multidisciplinary Sciences, JCI Quartile: Q1: JCI Percentile: 85.55, Do-Yak project).
- ii. **Cheolsun Kim, Dongju Park, and Heung-No Lee***, “Compressive sensing spectroscopy using a residual convolutional neural network”, *MDPI Sensors*, Vol. 20(3), Jan. 2020. (Impact Factor: 3.031, Do-Yak project) ([PDF](#))
- iii. **Cheolsun Kim, Woong-Bi Lee, Soo Kyung Lee, Yong Tak Lee, and Heung-No Lee***, “Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy”, *Optics and Lasers in Engineering*, Vol. 115, pp. 53-58, Apr. 2019. ([PDF](#)) (Impact Factor: 3.388 Doyak project)
- iv. **J. Oliver, WoongBi Lee, and Heung-No Lee**, “Filters with random transmittance for improving resolution in filter-array-based spectrometers,” *Optics Express*, Vol. 21, No. 4, pp. 3969–3989, Feb. 2013. (Impact Factor: 3.587; Do-Yak Project). ([pdf](#))

기술 연구 결과 리스트:

[기술 관련 연구 결과, 핵심 기술 연구 결과]

(AI + Blockchain)

1. **Mohamed Yaseen Jabarulla and Heung-No Lee***, “Blockchain and Artificial Intelligence-based Patient-Centric Healthcare System for Combating COVID-19 Pandemic: Opportunities and Applications,” *Healthcare*, 2021, 9(8), 1019. doi: <https://doi.org/10.3390/healthcare9081019> (Impact Factor: 2.645, Do-Yak project). Paper: ([Journal Page](#)) ([Cover Story](#)) ([Cover Image](#)).
2. **Mohamed Yaseen Jabarulla and Heung-No Lee***, “Blockchain-Based Distributed Patient-Centric Image Management System”, *Applied Sciences*, 11(1), 196, Dec. 2020. doi: <https://doi.org/10.3390/app11010196>. (IF: 2.679, Do-Yak and IITP) Paper: ([Journal Page](#)). PCAC-SC Solidity Code: <https://github.com/infonetGIST/PCAC-SC>

(Green Bitcoin)

1. **Heung-No Lee, Young-Sik Kim, Dilbag Singh, and Manjit Kaur**, “Green Bitcoin: Global Sound Money,” *The Journal of Digital Assets*,” Vol.1, No. 1, pg. 33-47, October, 2022. (The Inaugural Issue of the Journal of Digital Assets, No Impact Factor) ([PDF](#))

(ECCPoW)

1. **Hyungsung Kim, Jehyuk Jang, Sangjun Park, and Heung-No Lee***, “Error-Correction Code Proof-of-Work on Ethereum”, *IEEE Access*, Vol. 9, pp. 135942-135952, Sep 2021. doi: <https://doi.org/10.1109/ACCESS.2021.3113522> (Impact Factor: 3.367, IITP & Do-Yak project) Paper: ([Open Access](#))
2. **Haeung Choi, Sangjun Park and Heung-No Lee**, “Covert Anti-Jamming Communication Based on Gaussian Coded Modulation”, *Applied Sciences*, 11(9), April 2021, doi: <https://doi.org/10.3390/app11093759> (EW33 project), (Impact factor: 2.679, Category: Engineering, Multidisciplinary, JCI Quartile: Q2, JCI Percentile: 67.94) Paper: ([Open Access](#))
3. **Jehyuk Jang and Heung-No Lee**, “Profitable Double-Spending Attacks,” *Applied Sciences*, 10, 8477, .Nov. 2020. doi: <https://doi.org/10.3390/app10238477>. (IF: 2.474, Do-Yak and IITP) Paper: ([Open access](#))
4. **Hyunjun Jung, Heung-No Lee***, “ECCPoW: Error-Correction Code Based Proof-of-Work for ASIC Resistance”, *Symmetry*, June. 2020, 12(6), 988 (impact Factor: 2.143, Do-Yak project) ([PDF](#))
5. **Sangjun Park, Nam Yul Yu, Heung-No Lee**, “An Information-Theoretic Study for Joint Sparsity Pattern Recovery with Different Sensing Matrices”, *IEEE Transactions on Information Theory*, Vol. 63, no. 9, pp. 5559-5571, May. 2017. (Impact Factor: 2.679, Do-Yak Project) ([PDF](#)) DOI: [10.1109/TIT.2017.2704111](https://doi.org/10.1109/TIT.2017.2704111)
6. **J. Zhang and Heung-No Lee**, “Performance Analysis on LDPC-Coded System over Quasi-Static (MIMO) Fading Channels,” *IEEE Transactions on Communications*, vol. 56, no. 12, pp. 2080-2093, Dec. 2008. ([pdf](#))
7. **J. Wu and Heung-No Lee**, “Performance Analysis for LDPC-Coded Modulation in MIMO Multi-Access Systems,” *IEEE Transactions on Communications*, vol. 55, no. 7, pp.1417-1426, Jul, 2007. ([pdf](#))
8. **J. Zhang and Heung-No Lee**, “Performance Analysis of LDPC-Coded Space-Time Modulation over MIMO Fading Channels,” *IEEE Communications Letters*, vol. 11, no. 3, pp. 234 – 236, Mar. 2007. ([pdf](#))
9. **Heung-No Lee and X. Hu**, “Robust Iterative Tree-Pruning Detection and LDPC Decoding,” *IEEE Journal on Selected Areas in Communications*, vol. 23, no.5, pp. 1013-1025, May 2005. ([pdf](#))
10. **Sangjun Park, Haeung Choi and Heung-No Lee**, "Time-Variant Proof-of-Work Using Error-Correction Codes," preprint on arXiv, June 2020, <https://arxiv.org/abs/2006.12306> .
11. **Hyunjun Jung, Heung-No Lee***, “ECCPoW: Error-Correction Code Based Proof-of-Work for ASIC Resistance”, *Symmetry*, June. 2020, 12(6), 988 (impact Factor: 2.143, Do-Yak project) ([PDF](#))
12. **Seungmin Kim, Gyeongdeok Maeng, Heung-No Lee**, "Smart Contract-Based Checkpoint for Initial PoW Network Security," 2023 Fourteenth International Conference on Ubiquitous and Future Networks (ICUFN 2023), July 2023, pp.89-93.

(Smart Contract)

1. **Satpal Singh Kushwaha, Sandeep Joshi, Dilbag Singh, Manjit Kaur, and Heung-No**

Lee, "Ethereum Smart Contract Analysis Tools: A Systematic Review" n *IEEE Access*, (Impact factor: 3.367, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 66.75, IITP & Do-Yak project). ([Open Access](#))

2. **Satpal Singh Kushwaha, Sandeep Joshi, Dilbag Singh, Manjit Kaur, And Heung-No Lee**, "Systematic Review of Security Vulnerabilities in Ethereum Blockchain Smart Contract", *IEEE Access*, Vol. 10, pp. 6605-6621, doi: 10.1109/ACCESS.2021.3140091 (Impact factor: 3.367, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 66.75, IITP & Do-Yak project). Paper: ([Open Access](#))

(Cryptography)

1. **M. Kaur et al.**, "EGCrypto: A Low-Complexity Elliptic Galois Cryptography Model for Secure Data Transmission in IoT," in *IEEE Access*, vol. 11, pp. 90739-90748, 2023, doi: 10.1109/ACCESS.2023.3305271.
2. **M. Kaur, A. A. AlZubi, D. Singh, V. Kumar and Heung-No Lee**, "Lightweight Biomedical Image Encryption Approach," in *IEEE Access*, vol. 11, pp. 74048-74057, 2023, doi: 10.1109/ACCESS.2023.3294570.
3. **Varsha Himthani, Vijaypal Singh Dhaka, Manjit Kaur, Geeta Rani, Meet Oza, and Heung-No Lee**, "Comparative Performance Assessment of Deep Learning based Image Steganography Techniques", *Scientific Reports*, (Impact factor: 4.996, Category: Multidisciplinary Sciences, JCR Quartile: Q2: JIF Percentile: 74.66).
4. **Varsha Himthani, Vijaypal Singh Dhaka, Manjit Kaur, Dilbag Singh and Heung-No Lee**, "Systematic Survey on Visually Meaningful Image Encryption Techniques" *IEEE Access*, (Impact factor: 3.476, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 62.14, IITP & Do-Yak project). ([Open Access](#))
5. **Jusung Kang, Young-Sik Kim, and Heung-No Lee**, "Radio Frequency Public Key Generator for Digital Application", *IEEE Access*, Vol. 11, Dec. 2023, doi: [10.1109/ACCESS.2023.3340305](https://doi.org/10.1109/ACCESS.2023.3340305), (ITRC project), (Impact factor: 3.9, Category: Engineering, Electrical & Electronic, JCI Quartile: Q2, JCI Percentile: 69.12). Paper: ([Link](#)).
6. **Sharanpreet Kaur, Surender Singh, Manjit Kaur, and Heung-No Lee**, "A systematic review of computational image steganography approaches", *Archives of Computational Methods in Engineering*, Springer, (Impact factor: 7.302, Category: Mathematics, Interdisciplinary Applications, JCR Quartile: Q1: JCR Percentile: 99.54, IITP & Do-Yak project). ([PDF](#))
7. **Praveen Bondada, Debabrata Samanta, Manjit Kaur, and Heung-No Lee***, "Data Security based routing in MANETs using key management mechanism", *Applied Sciences*, MDPI, 12(3), 20 Jan. 2022, doi: 10.3390/app12031041, (Impact factor: 2.679, Category: Engineering, Multidisciplinary, JCI Quartile: Q2, JCI Percentile: 67.94, IITP & Do-Yak project). ([Open Access](#))
8. **Dilbag Singh, Vijay Kumar, Manjit Kaur, Mohamed Yaseen Jabarulla, and Heung-No Lee***, "Screening of COVID-19 Suspected Subjects using Multi-Crossover Genetic Algorithm based Dense Convolutional Neural Network", *IEEE Access*, Vol. 9, pp. 142566-142580, Oct 2021. doi: <https://doi.org/10.1109/ACCESS.2021.3120717> (Impact factor: 3.367, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 66.75, IITP & Do-Yak project).

project) Paper: ([Open Access](#))

(Semantic Analysis)

1. **Shashank S. Singh, Vishal Srivastava, Ajay Kumar. Shailendra Tiwari, Dilbag Singh, and Heung-No Lee**, “Social Network Analysis: A Survey on Measure, Structure, Language Information Analysis, Privacy, and Applications” Accepted in ACM Transactions on Asian and Low-Resource Language Information Processing, (Impact factor: 3.316, Category: Computer Science, Artificial Intelligence, JCR Quartile: Q4: JCR Percentile: 18.35, IITP & Do-Yak project).
2. **Apeksha Aggarwal, Akshat Srivastava, Ajay Agarwal, Nidhi Chahal, Dilbag Singh, Abeer Alnuaim, Aseel Alhadlaq and Heung-No Lee***, “Two-Way Feature Extraction for Speech Emotion Recognition Using Deep Learning”, MPDI Sensors (Impact Factor: 3.576, Category: Instruments & Instrumentation, JCI Quartile: Q1: JCI Percentile: 78.91, Do-Yak & IITP project). ([Open Access](#))
3. **Rahman S M Wahidur, Ishmam Tashdeed, Manjit Kaur, and Heung-No Lee**, “Enhancing Zero-Shot Crypto Sentiment with Fine-tuned Language Model and Prompt Engineering”, in IEEE Access, Early Access, doi: 10.1109/ACCESS.2024.3350638. (URL)

(HSI Classification / regression)

1. **Cheolsun Kim, Pavel Ni, Kang Ryeol Lee, and Heung-No Lee***, “Mass production-enabled computational spectrometers based on multilayer thin films”, Accepted in Scientific Reports, (Impact Factor: 4.380, Category: Multidisciplinary Sciences, JCI Quartile: Q1: JCI Percentile: 85.55, Do-Yak project).
2. **Cheolsun Kim, Dongju Park, and Heung-No Lee***, “Compressive sensing spectroscopy using a residual convolutional neural network”, *MDPI Sensors*, Vol. 20(3), Jan. 2020. (Impact Factor: 3.031, Do-Yak project) ([PDF](#))
3. **Cheolsun Kim. Woong-Bi Lee, Soo Kyung Lee, Yong Tak Lee, and Heung-No Lee***, “Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy”, *Optics and Lasers in Engineering*, Vol. 115, pp. 53-58, Apr. 2019. ([PDF](#)) (Impact Factor: 3.388 Doyak project)
4. **J. Oliver, WoongBi Lee, and Heung-No Lee**, “Filters with random transmittance for improving resolution in filter-array-based spectrometers,” *Optics Express*, Vol. 21, No. 4, pp. 3969–3989, Feb. 2013. (Impact Factor: 3.587; Do-Yak Project). ([pdf](#))

(Radio Frequency IoT AI)

1. **Jian Su; Zhenlong Liao, Zhengguo Sheng, Alex X. Liu, Dilbag Singh, and Heung-No Lee**, ” Human Activity Recognition Using Self-powered Sensors Based on Multilayer Bi-directional Long Short-Term Memory Networks” *IEEE Sensors Journal*, (Impact factor: 4.325, Category Instruments & Instrumentation, JCR Quartile: Q1: JIF Percentile: 78.91, Do-Yak project). ([Link](#))
2. **Harpreet Singh, Suchita Sharma, Manju Khurnana, Manjit Kaur, and Heung-No**

- Lee, “Binary drone squadron optimization approaches for feature selection” *IEEE Access*, (Impact factor: 3.476, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 62.14, IITP & Do-Yak project). ([Open Access](#))
3. **Jusung Kang, Younghak Shin, Hyunku Lee, Jintae Park, and Heung-No Lee**, “Radio Frequency Fingerprinting for Frequency Hopping Emitter Identification”, *Applied Sciences*, *11*(22), Nov. 2021, doi: 10.3390/app112210812 (LIG Nex1 project), (Impact factor: 2.679, Category: Engineering, Multidisciplinary, JCI Quartile: Q2, JCI Percentile: 67.94). Paper: ([Open Access](#)) ([Paper](#)) ([Explanatory Material](#))
 4. **Kiwon Yang, Jusung Kang, Jehyuk Jang and Heung-No Lee**, “Multimodal Sparse Representation-Based Classification Scheme for RF Fingerprinting,” *IEEE Communications Letters*, Vol. 23, Issue 5, pp. 867-870, May 2019. (Impact Factor: 2.723, Doyak Project) ([PDF](#)) ([Used DB](#)) ([Source Code](#))
 5. **Jin-Taek Seong and Heung-No Lee**, “Predicting the Performance of Cooperative Wireless Networking Schemes with Random Network Coding,” *IEEE Transactions on Communications*, vol. 62, no. 8, pp. 2951-2964, Aug. 2014. (Impact Factor: 1.979, Do-Yak Project) ([pdf](#))
 6. **Sang-Seon Byun, Ilango Balasingham, Athanasios V. Vasilakos, and Heung-No Lee** “Computation of an Equilibrium in Spectrum Markets for Cognitive Radio Networks,” to appear *IEEE Transactions on Computers*, Published on-line first on Aug. 30th, 2012. (Impact Factor: 1.473, Do-Yak Project, Haek-Sim Project)([pdf](#))
 7. **Cheng-Chun Chang, Zhi-Hong Mao, and Heung-No Lee**, “MB iterative decoding algorithm on systematic LDGM codes: Performance evaluation,” *Signal Processing*, vol. 90, issue. 1, pp. 373-377, 2010.(Impact Factor: 2.238) ([pdf](#))

(Medical AI)

1. **Manjit Kaur, Dilbag Singh, Vijay Kumar, Heung-No Lee**, “MLNet: Metaheuristics-based Lightweight Deep Learning Network for Cervical Cancer Diagnosis”, *IEEE Journal of Biomedical and Health Informatics*, (Impact factor: 7.021, Category Mathematical & Computational Biology, JCR Quartile: Q1: JIF Percentile: 93.86). ([Link](#))
2. **Dilbag Singh, Manjit Kaur, Jazem Mutared Alanazi, Ahmad Ali AlZubi, Heung-No Lee**, “Efficient evolving deep ensemble medical image captioning model”, *IEEE Journal of Biomedical and Health Informatics*, (Impact factor: 7.021, Category Mathematical & Computational Biology, JCR Quartile: Q1: JIF Percentile: 93.86). ([Link](#))
3. **Rohini Raina, Naveen Kumar Gondhi, Chaahat, Dilbag Singh, Manjit Kaur, and Heung-No Lee**, “A Systematic Review on Acute Leukemia Detection Using Deep Learning Techniques” *Archives of Computational Methods in Engineering*, Springer, (Impact factor: 7.302, Category: Mathematics, Interdisciplinary Applications, JCR Quartile: Q1: JCR Percentile: 99.54, IITP & Do-Yak project). ([Link](#))
4. **Dilbag Singh, Manjit Kaur, Vijay Kumar, Mohamed Yaseen Jabarulla, and Heung-No Lee**, “Artificial Intelligence-based Cyber-Physical System for Severity Classification of Chikungunya Disease”, *IEEE Journal of Translational Engineering in Health and Medicine*, (Impact factor: 3.316, Category: Engineering, Biomedical, JCR Quartile: Q2: JCR Percentile: 51.12). ([Open Access](#))
5. **Jayashree Piri, Puspanjali Mohapatra, Debabrata Singh, Debabrata Samanta, Dilbag**

Singh, Manjit Kaur and Heung-No Lee*, “Mining and Interpretation of Critical Aspects of Infant Health Status Using Multi-Objective Evolutionary Feature Selection Approaches”, Accepted in *IEEE Access*, 10.1109/ACCESS.2022.3161154, (Impact factor: 3.367, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 66.75, IITP & Do-Yak project). ([Open Access](#))

6. **Sushama Tanwar, S. Vijayalakshmi, Munish Sabharwal, Manjit Kaur, Ahmad Ali AlZubi, and Heung-No Lee***, “Detection and Classification of Colorectal Polyp using Deep Learning.” *BioMed Research International*, Mar. 2022, (Impact Factor: 3.411, Category: BIOTECHNOLOGY & APPLIED MICROBIOLOGY, JCI Quartile: Q3: JCI Percentile: 43.94, Do-Yak & IITP project). ([Open Access](#))
7. **Priyanka Malhotra, Sheifali Gupta, Deepika Koundal, Atef Zaguia, Manjit Kaur and Heung-No Lee***, “Deep Learning Based Computer Aided Pneumothorax Detection Using Chest X-Ray Images”, *MPDI Sensors* (Impact Factor: 3.576, Category: Instruments & Instrumentation, JCI Quartile: Q1: JCI Percentile: 78.91, Do-Yak & IITP project). ([Open Access](#))
8. **Abhinav Mishra, Ganapathiraju Dharahas, Shilpa Gite, Ketan Kotecha, Deepika Koundal, Atef Zaguia, Manjit Kaur, and Heung-No Lee***, “ECG Data Analysis with Denoising Approach and Customized CNNs”, *Sensors*, *MPDI*, 22(5), 1928, 2022 (Impact Factor: 3.576, Category: Instruments & Instrumentation, JCI Quartile: Q1: JCI Percentile: 78.91, Do-Yak & IITP project). ([Open Access](#))

(Agricultural AI)

1. **Harshit Kaushik, Anvi Khanna, Dilbag Singh, Manjit Kaur, Heung-No Lee,**” TomFusioNet: A tomato crop analysis framework for mobile applications using the multi-objective optimization based late fusion of deep models and background elimination” ([PDF](#))
2. **Dilbag Singh, Yavuz Selim Taspinar, Ramazan Kursun, Ilkay Cinar, Murat Koklu, Ilker Ali Ozkan, and Heung-No Lee***, “Classification and Analysis of Pistachio Species with Pre-Trained Deep Learning Models”, *MPDI Electronics* (Impact Factor: 2.397, Category: Physics and Applied, JCI Quartile: Q2: JCI Percentile: 57.60, Do-Yak & IITP project). ([Open Access](#))

(Character recognition AI)

1. **Dilbag Singh, Aravinda C.V, Manjit Kaur, Meng Lin, Jyoti Shety, Vikram Raju Reddicherla, and Heung-No Lee,** “DKNet: Deep Kuzushiji Characters Recognition Network” *IEEE Access*, (Impact factor: 3.476, Category: Engineering, Electrical & Electronic, JCR Quartile: Q2: JCI Percentile: 62.14, IITP & Do-Yak project). ([Open Access](#))

(Air quality prediction)

1. **Manjit Kaur, Dilbag Singh, Mohamed Yaseen Jabarulla, Vijay Kumar, Jusung Kang, and Heung-No Lee.** 2023. Computational deep air quality prediction techniques: a systematic review. *Artificial Intelligence Review* (2023), 1–46

Gwangju Institute of
Science and Technology

School of Information and Communications



WorldLand™

- WorldLand, My AI Network 핵심 기술 연구 결과-

Green Bitcoin: Global Sound Money

Heung-No Lee, Young-Sik Kim, Dilbag Singh, and Manjit Kaur

Abstract

Modern societies have adopted government-issued fiat currencies many of which exist today mainly in the form of digits in credit and bank accounts. Fiat currencies are controlled by central banks for economic stimulation and stabilization. Boom-and-bust cycles are created. The volatility of the cycle has become increasingly extreme. Social inequality due to the concentration of wealth is prevalent worldwide. As such, restoring sound money, which provides stored value over time, has become a pressing issue. Currently, cryptocurrencies such as Bitcoin are in their infancy and may someday qualify as sound money. Bitcoin today is considered as a digital asset for storing value. But Bitcoin has problems. The first issue of the current Bitcoin network is its high energy consumption consensus mechanism. The second is the cryptographic primitives which are unsafe against post-quantum (PQ) attacks. We aim to propose Green Bitcoin which addresses both issues. To save energy in consensus mechanism, we introduce a post-quantum secure (self-election) verifiable coin-toss function and novel PQ secure proof-of-computation primitives. It is expected to reduce the rate of energy consumption more than 90 percent of the current Bitcoin network. The elliptic curve cryptography will be replaced with PQ-safe versions. The Green Bitcoin protocol will help Bitcoin evolve into a post-quantum secure network. In addition, it improves the properties of Bitcoin's hash PoW while addressing environmental concerns.

Keywords: Bitcoin, energy consumption, error-correction codes, post-quantum security, sound money, verifiable random function

I. INTRODUCTION

WE the global citizens not by our choice live in a world of boom-and-bust cycles created by the Federal Reserve Board (FED) of the United States. In the boom phase of a cycle, the FED supplies debt-monetized US dollars to the world, and supplied USDs are used to purchase real items, goods, and services from developing countries. Such supplies prop up bubble markets, such as the US housing, stock, and derivatives markets. Global elite financial institutions, such as investment banks and hedge funds, benefit the most from making risky investments. When the bust part of the cycle comes, the working class worldwide suffers from

This work was supported in part by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (NRF-2021R1A2B5B03002118) and the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center (ITRC) support program (IITP-2021-0-01835). This work was also supported by the Tech Incubator Program for Startup (TIPS) program (S3306777) awarded in June 2022.

Heung-No Lee is with Gwangju Institute of Science and Technology, Gwangju, 61005, South Korea and affiliated with LiberVance, Co., Ltd., Gwangju, South, Korea.

Dilbag Singh and Manjit Kaur are with Gwangju Institute of Science and Technology, Gwangju, 61005, South Korea.

Young-Sik Kim is with Chosun University, Gwangju, 61452, South Korea, and affiliated with LiberVance, Co., Ltd., Gwangju, 61005, South Korea.

Corresponding author: Heung-No Lee (e-mail: heungno@gist.ac.kr).

Received: September 15, 2022; Revised: October 20, 2022; Published: October 31, 2022.

inflation and market crashes. In the aftermath of a market crash, governments bail out financial institutions to prevent chain bankruptcies. Such centralized planning has disrupted societies globally [21][29]. The boom-and-bust volatility peaked with the Dow to gold ratio has become increasingly extreme [6]. Inequality is a prevalent condition worldwide. Work ethics fade. Growing are speculative markets. Keeping this system in its current form does not help advance humanity to the next level.

Throughout history, money has taken many forms, including gold, silver, copper, salt, and seashells [21]. As modern society developed, government-issued fiat currencies became normal, many of which exist in the form of digits in banks and credit accounts today. The soundness of money is determined by its stability; if it is stable, it can function as a medium of exchange, unit of account, and store of value. Central banks know this and seek to stabilize their currency by controlling money supply as the economic condition changes. Today, however, this flexibility is often misused and overused, especially by a new government that needs to satisfy voters and make way for its political agenda. The boom comes from stimulation, and the bust comes back from monetary tightening. It seems clear that innovative measures are needed to address the current situation, which depends entirely on central bank policy decisions.

Today, central banks are often tightly coupled with political powers. When new presidents come into the office, they are tempted to use the central bank's power in money creation capability to fulfill their political agenda. The stability of money, and thus the market order, is broken in the name of economic growth and stabilization. Owing to the manipulation of money, monetized debts, inflation of asset prices, and growing inequality are rampant [4]. Thus, there is a need to restore sound money whose supply is independent of governments' control, which is left alone to a free market and its self-regulation mechanisms.

Bitcoin was humanity's first success in creating decentralized money [21][41]. It realized a sound money Fredrick Hayek proposed in his book, *The Denationalization of Money* [22]. Its success was possible because of cryptographic technologies, such as SHA, digital signature algorithms, and elliptic curve cryptography. Currently, Bitcoin is one of the soundest currencies, but it still does not fully satisfy itself as sound money. For example, it does not work as a means of payment, as blockchain networks incur transaction costs that are too high for daily spending. A lightning protocol (along with other second-layer solutions) [53] facilitates off-chain transactions at high speeds. Unfortunately, these solutions are not secure against post-quantum (PQ) attacks.

These cryptographic primitives must be upgraded regularly. Otherwise, their use is limited and useless if not upgraded. IBM announced its plan to introduce quantum computers with more than four thousand qubits by 2023 [26]. With such advances, cryptographic algorithms used in Bitcoin are on the verge of breaking.

The main contributions of this paper are as follows. This paper reviews Hayek's sound money and discusses the pressing need to restore it in our society. The advantages of Bitcoin concerning its robust performance, such as simplicity in consensus and time-energy-borne wealth characteristics, are emphasized. Subsequently, a post-quantum (PQ) safe Green Bitcoin protocol is proposed. The proposed protocol can help achieve the majority of sound money properties. Green Bitcoin comprises two major parts: a PQ secure verifiable (self-election) coin-toss (VCT) function and a novel PQ secure proof-of-computation (PoC) primitive. The PoC part is built based on a newly published finding known as the error-correction code anti-ASIC proof-of-work (ECCPoW) [30][31][33][45]. PoC primitives will make PQ safer than the ECCPoW. Environmental concerns can be addressed with the VCT function, aided by the PoolResistantComp protocol. This can be used to control the network's energy consumption efficiency. Critical components of opcodes, such as digital signature algorithm, will be enhanced, and elliptic curve cryptography will be replaced with PQ safe versions. The Green Bitcoin protocol will help Bitcoin evolve into a post-quantum secure network. In addition, it helps achieve good properties of Bitcoin's hash PoW while alleviating environmental concerns.

The remainder of this paper is organized as follows: Section II discusses the significance of cryptocurrency and blockchain. Section III discusses the benefits of Bitcoin, sound money, and problems. Section IV presents the Green Bitcoin proposal. Novel PQ secure primitives are discussed in Section V, and a Green Bitcoin testbed and discussion are presented in Section VI. Section VII concludes this paper.

II. CRYPTOCURRENCY AND BLOCKCHAIN

Cryptocurrencies are digital currencies created through encryption algorithms that can be used as a form of payment. Using encryption technologies, cryptocurrencies can serve as currency and virtual accounting systems. Digital wallets are used for cryptocurrency trades. Blockchain networks were developed to provide decentralized requirements for cryptocurrencies. Consensus, virtual machines, and peer-to-peer (p2p) networking are the three primary components of a blockchain. One of the most pressing demands is to 1) update the consensus mechanism that allows a new PQ secure and decentralized blockchain network and 2) upgrade the cryptographic primitives used in consensus and virtual machines to be PQ safe.

Currently, blockchains are not PQ secure, there are environmental concerns and scalability issues. Several major projects have implemented Proof of Stake (PoS) paired with a Byzantine agreement (BA) algorithm to resolve these concerns. The PoS and BA algorithms are perhaps good for a fast-computing platform, but they are not secure enough for a global monetary-grade blockchain. The ideas are not new; the BA algorithm (developed in the 1980s [12][32]) relies on communication across committee nodes to reach a consensus, rendering it subject to various assaults, including DDoS (distributed denial of service) and network partition attacks. To scale it well in terms of the number of nodes, the committee size cannot but remain small; hence, decentralization is compromised.

Blockchains are a dear but expensive solution. The blocks are stored redundantly within every consensus-participating node, and all nodes perform the same work. Each new block is made with an effort, an enormous amount of time and energy, by the entire network. This is the source of the immutability of the records stored in the blockchain; the network must be decentralized to the maximum possible extent. The greater the number of individual nodes participating, the more secure the network is regarding censorship resistance, thwarting Sybil, and double-spending attacks [27][41]. Consider Bitcoin; each block contains a massive amount of computational energy stored in each block. If the block needs to be forged again, it requires the same amount of energy to be stored. On the one hand, the large redundancy and numerous independently working nodes doing the same work can be viewed as a source of security. On the other hand, it can be viewed as a waste of resources and a waste of energy. The blockchain trilemma [5][7][14]—it is difficult to achieve the three blockchain properties simultaneously such as scalability, decentralization, and security—represents a scalability challenge caused by inefficient resource consumption. The complaint about energy issues leads to environmental concerns.

Blockchain technology requires a simple protocol to withstand attacks and perform robustly for years to come [33]. The consensus mechanism should be able to maximize the resilience of numerous unknown attack vectors. How can we make it simple while accommodating many p2p nodes working together to reach a consensus? The participating nodes must make timely judgments and choose one block from several candidate blocks to be the new block attached to the status quo chain. A timely consensus decision should be distributed among numerous independent working nodes over the internet. Consequently, an agreement must be reached with as few contacts as possible among the p2p nodes.

Frequent network delays and partitions occur on the internet. They might be caused by momentary router failures, traffic congestion, or purposeful antagonistic activities. Therefore, a global blockchain network must be resilient against possible attacks and losses.

A consensus mechanism using hash function-based PoW [41] has been shown to provide the most decentralized and secure operations. Thus, it is challenging to design a consensus mechanism that achieves all these needs: a large number of p2p nodes, making timely decisions while working independently, with minimal inter-node communication.

III. BITCOIN, SOUND MONEY, PROBLEMS

A. Bitcoin

Bitcoin [41] is a newly developed type of money known as “cryptocurrency.” Some in the Bitcoin community believe that Bitcoin already represents sound money [23][24]. Sound money is defined as money that has a purchasing power determined by markets, independent of governments and political parties. There would be a vast difference between a world with Bitcoin and one without Bitcoin. While governments continue to print

cheap money, ordinary people can resort to Bitcoin. Thus, the wealth gap can be significantly reduced. We will come closer to a produce-first-and-spend economy rather than a debt-monetized spending-first economy. Bitcoin encourages savings and autonomous management of money; it does not allow bailouts. Governments are not held accountable for policy failure. Tax-paying individuals and corporations are the entities who are held accountable for the economic setbacks caused by policy failures. The use of the power to mint the currency held by governments (non-responsible entities) should be limited. That way, central banks will no longer be able to punish diligent savers by sprinkling them with debt-based currencies. With a stable store of value, people can confidently plan for the future. Zombie companies and bubble markets that thrive on government budgets and easy monetary policies will find it difficult to survive.

An organism's code of life is written at its conception. Bitcoin's DNA, or genetic code, was carefully crafted by Satoshi as the soundest money ever created. Bitcoin's genetic code can be considered a set of instructions designed to incentivize the coordination and organization of cellular functions.

The genetic code of Bitcoin [41] is as follows:

- Bitcoin had to be ignited to become real, so Satoshi coded a fixed supply (21 million Bitcoins) into its DNA. Sound money comes from this fixed supply. Due to the increase in users, miners and developers, Bitcoin has become more expensive over time. Thus, the feedback loop has become self-reinforcing.
- The mining function, that is, the one utilizing a hash PoW, is the metabolism and defense mechanism of Bitcoin. For instance, Bitcoin consumes a lot of electric energy to make new blocks and create virtual walls to protect the network from hackers. The anti-fragility of Bitcoin is attributed to PoW, which makes it more resistant to attacks as it grows.
- The Bitcoin network generates a new block every 10 min on average. This rate is carefully determined for the robust operation of the nodes. Bitcoin nodes are scattered all over the globe; thus, they are separated over long distances. However, they can still communicate and coordinate effectively and have never stopped making new blocks since their inception in January 2009.

B. What Is Sound Money?

The 1974 Nobel Prize in Economics Laureate Friedrich Hayek wrote the *Denationalization of Money* (1976) [22]. He said privately issued money, which continues to develop through competition, will inevitably be superior to fiat money, which does not evolve because the government has exclusive power to supply it.

“I am more convinced than ever that if we ever again have sound money, it will not come from the government. This is issued by a private enterprise,”
Friedrich Hayek, 1977

Hayek emphasizes sound money in his book. Literally translated, it can be interpreted as honest money. However, one may wonder what sound money is and why he emphasized it. What will happen if money is sound, and what will happen if it is not? As a firm definition is not given in his book, the answers to these questions can be examined via the statements he provides in his book.

Today, we live in the age of fiat currencies. For example, the US dollar is a fiat currency. Fiat means “by government decree.” By law, the U.S. government declares the dollar to be a currency. Because the United States has the most powerful military and the largest economy in the world, it has gained worldwide trust, and the US dollar has become the world reserve currency.

Individuals earn sound money through solid value-creation processes. To produce more gold, one has to mine deeper in the gold mines. Gold has been utilized for thousands of years as a means to store value for future use. No government can print a sound money. No government can devalue the money and weaken the savings. To earn other people's money, one must provide equivalent time and energy for others. Governments should not be exceptions. They must also be prudent and considerate. They must win the hearts of their peoples with reliable, agile and considerate services, and a clear vision. They shall be limited to printing new money.

C. What Is the Problem If Money Is Not Sound?

The US dollar has caused serious social problems, as revealed by several studies [23][25]. The dollar has been overused by the US government and the central bank FED in the name of “stimulating the economy,” “resolving the financial crisis,” and “fighting the pandemic.” Whenever a bust phase occurred, dollars were

printed and supplied to society. Recent examples include quantitative easing measures to address the financial crisis caused by subprime mortgages in 2008. The US government issues treasury bonds and the FED provides dollars to society by buying them. The newly issued dollars spread to the world society. Major Western advanced countries will also have to expand the supply of their currencies to resolve the negative side effects of the large-scale supply of dollars.

This creates a situation where debt increases in major countries worldwide, and an oversupply of money is widespread. The additional money supplied each year causes real estate prices in major cities to skyrocket. The value of financial assets, such as the S&P 500 ETF in the US, continues to grow in a right-upward way by more than 15% annually. There is a huge wealth gap between those who can own these assets and those who cannot. Speculative markets in which unearned income can be obtained as much as the amount of over-supplied fiat currency will grow. However, the value of honest work based on diligence, knowledge, and skills decreases. Knowledge-seeking and corporate activities weaken. What labor activity, what knowledge-seeking activity, or what company's business model could produce growth rates higher than that of the S&P 500?

In a society where honest money reigns, bubbles weaken, and honest work is promoted. Companies invest in new knowledge- and skill-creation projects and develop new products and services. Honest labor and knowledge-seeking activities are vibrantly carried out, the workforce participating in production activities increases, and the economy grows robustly. Because there is no oversupply of money, no speculative bubble markets grow. As the speculative market disappears, more people can find a stable life through honest work.

Gold has been used as honest money for thousands of years [21][22]. Honest money is very important to humans because it performs useful roles such as a means of exchange, a measure of value, and store of value. However, gold is not easily produced. Producing even a single gram of new gold requires someone to work very hard over a long period. The only way to produce gold is to mine it from the ground. All gold that exists on the earth's surface has already been mined. Therefore, to mine gold today, you have to dig deeper. However, the production of gold has become increasingly difficult. This rarity is a key characteristic of gold. Because of these properties, gold was used as sound money.

The United States adopted the gold standard through the Bretton Woods Agreement of 1944. An ounce of gold was pegged at 35 US Dollars. Currencies in the Western world, such as the Franc (France), Pound (United Kingdom), Mark (Germany), and Yen (Japan), were pegged to the US Dollar, and currencies in developed countries were pegged to gold. Therefore, these currencies of major countries around the world were honest. On August 15, 1971, the 37th President of the United States, Nixon, announced [55] that he would temporarily withdraw from the gold standard, shocking the world. Since then, there have been many reports that easy monetary policy has created a polarization of wealth [25].

D. Is Bitcoin Sound Money?

Bitcoin has already acquired the status of an asset (a means of saving and investing) in many countries. El Salvador became the first country in the world to recognize Bitcoin as its legal tender on September 7, 2021. The US Securities and Exchange Commission (SEC) approved the Bitcoin Futures ETF on October 16, 2021 [9].

Is Bitcoin already the sound money Hayek was talking about? This is believed to be so among many Bitcoin enthusiasts [23][24]. According to Hayek's proposition of sound money, however, it should evolve through competition. Therefore, many Bitcoin hard forks should emerge in the future and continue competing to provide better services. As numerous new Bitcoins compete, the quality of monetary services constantly evolves. Fiat currency dominance will not disappear, as both the U.S. government and the FED issuing fiscal and monetary policies will continue to exist. In such a world, ever-evolving sound money protocols will provide a window through which citizens can protect themselves from the side effects of recklessly issued fiat currencies.

E. What Are Good Properties to Inherit from Bitcoin?

(Simplicity in consensus) The consensus among many p2p nodes is made every 10 min over the best-effort service and often-times hostile internet. How can robust operations become possible? It is interesting to focus on the Bitcoin consensus. The nodes in the Bitcoin network are not divided according to their job. Each node performs the necessary work, confirms transactions, groups them into a block, adds the proof of work, and publishes the block as quickly as feasible. Consensus is reached as the result of each node simply performing

its work for its own benefit. No node is forced to work in a time-division schedule. Each node does not need to adapt to the progress of the other nodes (therefore requiring no contact with each other) to obtain a consensus. Every node creates blocks, and every node validates the blocks. The protocol is simple and plain. This simplicity results in a robust performance. Hence, blocks cannot but be kept on produced. Rewards are provided as incentives to nodes. The more effort a node makes the more opportunities it will earn rewards. Node righteousness is not required, and no punishment that exists in a PoS-based consensus algorithm [14] is required.

(BA algorithms are not decentralized.) Jobs are divided under a BA algorithm. A set of nodes under the proposer's name creates blocks. The proposed blocks are validated by another group of nodes, such as *attesters*. They vote for each candidate block. Cast votes are collected and counted. A block with a satisfying number (i.e., supermajority) of votes was selected and connected to the *status quo* chain. To complete this, each node must fit into a tight schedule. Consequently, the time axis must be separated into slots and epochs. The number of nodes engaged in consensus must be limited to a few hundred to operate properly across a hostile environment such as the internet.

(Time-energy-borne wealth: Bitcoin is a stored wealth transformed from time and energy.) Each block header contains information on time and energy consumption. Take any blocks from the past. The block header contains the time stamp when the block was created. The difficulty level of the puzzle is also specified. Using this information, one can calculate the amount of computation (hash cycles) required to solve the hash puzzle. It would have taken more than three years for a single node working alone to solve the hash challenge. But it takes only 10 min for the entire network of operating nodes distributed independently worldwide. See the alone impossible together possible (Al-Im-To-Po) theory [33]. Thus, it indicates that hundreds of millions of computing nodes have to work together at that moment to produce that block. Given the energy efficiency of a mining node, the amount of total energy expended to create a block can be calculated. A predetermined number of new bitcoins are minted on the block. It can be said that the blockchain network has transformed the energy spent on creating the block into the Bitcoins issued from that block. The miners invested time and energy, thus are deserved to get the Bitcoins reward.

In this approach, each block generation process may be considered a new way of storing wealth. It is a transformation of the most basic resources of time and energy. Time and energy are the most valuable resources humans have and are the most fundamental types of wealth. Thus, the coins created in each block are assigned a monetary value from birth. The effort and time are not squandered; they were converted into something valuable: Bitcoin. Others have used their time and energy to create important commodities and services such as food and building houses. One can give Bitcoin to purchase goods and services that others have produced and offered.

F. What Are the Issues That Require Attention?

There are two problems. The first is the security risk in cryptography used in Bitcoin owing to advances in quantum computing technology [1][17]. The second is the huge energy consumption issue of the Bitcoin network.

As mentioned earlier, IBM is close to building a quantum computer that is known to break well-established cryptographic algorithms, such as elliptic curve cryptography, digital signature algorithm, and RSA algorithm. President Biden has signed two executive orders [50]. The first is to address how US leadership can be maintained in quantum computing technology. The second is to outline the U.S. government's strategy for mitigating the risks to vulnerable cryptographic systems due to advances in quantum technology. We address this further in Section V.

Bitcoin's energy consumption is enormous; its annual energy consumption has grown to 138 TWh in early 2022, which is more than the power consumption of a country such as Norway. Its annual carbon dioxide emission reached 114 million tons, which is comparable to that of Belgium [49][54].

Bitcoin advocates claim:

- The energy usage is still a very tiny proportion of global electricity consumption.
- Miners with self-interest move to places where surplus gas, solar panel farms, hydro and wind power plants, low-carbon nuclear plants, and geothermal energy are available.

But many still express concerns [50][54]:

- Chinese ban on Bitcoin mining was a response to a power deficit.
- Kazakhstan has imposed a limit on it owing to energy shortages.
- Sweden has called for the Europe-wide ban, blaming it for slowing climate transition.
- Tesla has withdrawn from its plan to accept Bitcoin as a form of payment, citing environmental concerns.

G. Alternatives miss the merits of Bitcoin

Detractors claim that energy-intensive PoWs are the culprit. Rival currencies that use variants based on proof-of-stake (PoS), such as Ethereum and Solana, are gaining popularity. Ethereum [14] announced its plan to reduce its estimated energy consumption by up to 99% with the completion of the transition to its PoS variant system [5][7][8][14]. Ethereum completed the Merge on September 15, 2022.

However, PoS is known to have many obvious concerns [7][10][35]. The PoS is being introduced to address energy concerns and increase the transactions per second (TPS). However, the penalty may be significant because it is neither decentralized nor secure. The PoS is not a technological advancement; it lacks the advantages of the PoW. It resorts to a sociopolitical solution: Plutocratic politics and the time-energy-born wealth property is lost. Because the PoS does not retain any energy on a block, the blocks may be readily rewritten; hence, it is insecure. Advocates of PoS put forward a policy for bad actors, such as “your stakes will be confiscated if you act badly,” This is a “fixing a barn after losing a cow” approach. The richest few can make confidential agreements off-chain and take control of the blockchain. These off-chain conspiring operations leave no on-chain trace; thus, no one can become aware of them. Hence, it is sensitive to bribery and conspiracy [38]. There is a nothing-at-stake risk [35]. There is a risk of a grinding attack if the random function for selecting a block creation node is unfair or predictable [10].

A PoS-based alternative may serve as a global computing platform but may not be decentralized, nor secure and thus not suitable for a global monetary grade network such as Bitcoin.

IV. GREEN BITCOIN

In this section, we discuss Green Bitcoin.

A. Key Performance Metrics

A set of key performance metrics of Green Bitcoin are:

- M1. PQ secure cryptography and consensus
- M2. Energy consumption efficiency (ECE)
- M3. Byzantine fault tolerance (1/2)
- M4. The mining schedule (21 million Bitcoin)
- M5. The block generation time (10 min average), and
- M6: The block size (1 mb)

Note that M4, M5, and M6 are the same as those of Bitcoin. M1 and M2 are our major focus. We aim to develop a PQ secure cryptography and a new energy-efficient consensus mechanism. For M3, as later shown in this paper, Green Bitcoin supports a BFT of up to 50%.

B. Green Bitcoin Consensus

The Green Bitcoin consensus comprises two new major parts: a verifiable (self-election) coin-toss function (VCT) and a novel proof-of-computation (PoC) primitive. A simple PoolResistantComp algorithm can be used to aid these two parts. Green Bitcoin will base its PoC part on a recently published anti-ASIC technology known as the error-correction code proof of work (ECCPoW) [30][31][45]. A more detailed discussion on ECCPoW is provided in Section V.F. A critical component of the virtual machine will also be enhanced; in particular, elliptic curve cryptography will be replaced by our new PQ safe cryptography.

We aim to address the PQ security and environmental concerns while maintaining decentralization. We will let computational challenge decreased. More nodes will be able to join the network as computational difficulty decreases. As a result, security will be bolstered via enhanced decentralization. The time-energy-borne wealth property of Bitcoin will be left de-emphasized. We aim to achieve the sound money property

without its price relative to fiat soaring.

Such goals will be met through the development of new on-chain means. Each node performs a simple task independently; the system's simplicity allows the system to run steadily even with a large body of participant nodes.

The consensus protocol is simple. Each participating node can easily obey the rules, and each node performs the same simple job. No job and time are divided. They need no inter-node communications to reach a consensus; each node performs its work independently. The same procedure is repeated for each new block; consensus for the current block is completed when some nodes announce the valid next block. The only announcement that each participant needs to keep a vigil for is the announcement of this valid new block. This simplicity lowers the entry barrier and invites more participating nodes.

Thus, Green Bitcoin enables the construction of a decentralized, scalable, and secure consensus solution for an extremely large network of participating nodes. It works well even if the number of participating p2p nodes exceeds one million.

(The base-set of p2p nodes) The base set of p2p nodes is defined as all the nodes participating in transaction validation and block formation. The protocol is set to perform well with base-set sizes greater than one million.

(Green Bitcoin consensus) Like the hash PoW, all nodes in the base set collaborate and contribute to creating each new block. Each node self-selects as a validator, attaches a proof for it, validates transactions, forms a block, and attaches a proof of the solution. Each node repeats this procedure for each new block. The benefit of this is the simplicity of the algorithm. Finality is determined by the longest chain or the amount of energy stored (measurable by each block difficulty level) in the blockchain. If there are two blockchains, each node for its own benefit will select the one with the most energy stored and connect a new block into it.

(Verifiable coin-toss function) Let us consider a coin toss game. Every node has its own unique (secret key) coin. Each node tosses its coin. A coin toss features a single output that may be either a pass or fail. Verifiable coin-toss function (VCT) is a verifiable random function (VRF). It has two inputs. One input is the secret key of the node, and the other is the previous block header. Thus, each node cannot but toss this VCT once and only once for each block. The purpose of VCT is to provide a means to turn off a certain portion of the base-set nodes, thus saving energy while allowing a large number of nodes to participate. If the probability of failure is set to 90%, 90% of network nodes are put to rest, and thus energy savings of 90% is achieved. To work on a new next block, each node tosses coin again.

We attempt to design the VCT function so that the odd of pass can be controlled. The probability of pass is a critical parameter that the network designer can use to change the amount of energy saving, given the size of the base set. For example, when the number of nodes participating in the network is small, it can be set to 100% to maximize security.

For those nodes that have been self-selected to perform the computing work, there are three types of computations: VeriComp, SolComp, and PoolResistantComp, each of which has to be performed.

(VeriComp) validates transactions and compiles them into a new block. Each node sets out to validate the new block upon receiving a new block announcement. If the block is valid, the node sets out to begin extending a new block next to the validated block.

(SolComp) is the computation required to solve the crypto puzzle. Each round presents a completely fresh crypto-puzzle. The puzzle problem is not predictable in advance, but it is determined if the preceding block header is fixed. Each node in the self-elected set starts the race to solve the crypto-puzzle as quickly as possible. A node with a proof of solution inserts the proof into the block header and broadcasts the new block instantly.

(Coin is tossed before any energy is spent on SolComp.) In the Green Bitcoin consensus, we observed that all nodes participate in the same simple routine. Each node performs a random turn to form a new block and attaches the proof of solution. The new VCT function makes it possible for them to take random turns. Each node progresses into the energy hungry SolComp routine if its coin toss result is a pass.

C. What If Each Selected Node Forms a Mining Pool?

It is possible that the nodes selected by tossing coins can promise a reward distribution to the unselected nodes and request collaboration. Selected nodes would enjoy such collaboration because they can use them to stay ahead in competition among the selected nodes and increase the probability of winning the block reward. Unless there is a deterrent to prevent selected nodes from making these choices, the proposition that VCT can

reduce energy consumption may turn out to be untrue.

The key question then is: Can we find an on-chain means to discourage such cooperation? Can we find a way to reduce the incentive for a selected node to form a pool by itself? Such a concern has its roots in pool mining practice in current PoW mining networks such as Bitcoin and Ethereum Classic. In a mining pool, miners subscribe to a mining pool server, perform mining tasks, and share rewards by submitting proper solutions.

To quickly grasp the idea, we can delineate our proposed solution for the pool-mining case. Here, the pool mining server is compared with one of the *selected* nodes in the Green Bitcoin consensus mechanism; the miners subscribing to the pool mining server can be compared with the *unselected* nodes. Each selected node can solicit cooperation from unselected nodes and increase its probability of winning the block reward. Unselected nodes can increase their chance of winning a share of the block reward by helping the selected node. Because both parties may benefit from such cooperation, unselected nodes continue to spend energy doing the SolComp work even if they were unselected in the first place by coin-toss. This would erase the energy reduction effect. In a pool mining protocol [46], five major steps are required for miners to obtain shared rewards:

- 1) register itself as a miner to a Stratum server
- 2) get the block header information
- 3) work until a mining success is announced
- 4) submit the mining results, and
- 5) get shared rewards from the server.

The key idea in breaking apart the cooperation between the selected and the unselected nodes is to create a conflict between them. One such scheme can be designed such that the selected node is asked to have its private key at risk revealed in order to delegate SolComp work to the unselected nodes.

(PoolResistantComp) There is a method proposed in the literature [15] that aims to break the cooperative tie between the pool server (vis-a-vis the selected node in our case) and the miners (the unselected nodes), thus discouraging pool mining. This is known as a two-phase PoW system. The first PoW is the routine PoW part, which is the same as the existing Bitcoin crypto-puzzle. Miners seek to find the hash of the header that is smaller than the published difficulty parameter. The second PoW is PoolResistantComp: it is the second puzzle in which the miner is asked to include a critical piece of information as an input to the hash function. Namely, the miner is asked to have the header signed with the private key of the coin-base transaction, that is, $\text{SHA256}(\text{sign}(\text{header}, \text{privkey}))$, and the second puzzle is resolved if a node successfully presents a hash of that signature that is smaller than a second difficulty parameter.

Note here that the second puzzle cannot be relegated to unselected nodes unless each selected node takes the risk of revealing its private key to the unselected nodes.

In our case, we do not need such a two-phase protocol. The PoolResistantComp routine can be incorporated right into the SolComp stage with minimal effort. The PoolResistantComp part is a routine asking for the inclusion of the private key corresponding to the coin-base transaction. ECCPOW (see Fig. 1 in Ref. [45]) is a routine in which a decoder finds a codeword from the hash output, that is, $\text{OUT} = \text{SHA256}(\text{current block header})$. To embed PoolResistantComp into it, we can replace the hash output routine to include the private key's signature, i.e., $\text{OUT} = \text{SHA256}(\text{sign}(\text{current block header}, \text{privkey}))$. Such a single-line update is sufficient to regulate the nodes and force them to rest if they are not selected. This update does not lose the characteristics of *the simple* and *plain* routine discussed in IV.B. Each node simply does its routine, regardless of the states of the other nodes.

V. NOVEL PQ SECURE PRIMITIVES

We now discuss the novel postquantum-ready and suitable cryptographic primitives. They are the novel key generation, sign, and verification functions; it also has a new Green Bitcoin, VRF, VCT, and VC.

Quantum computers are known to break well-established cryptographic algorithms such as the elliptic curve cryptography, digital signature algorithm, and RSA algorithm. In particular, these methods are based on integer factorization and discrete logarithm problems, which are known not quantum-safe. In contrast,

code-based cryptography issues are known to be quantum-safe.

(A brief history of early code-based cryptography) McEliece first presented code-based cryptosystems using binary Goppa codes in 1978 [37]. In 1986, Niederreiter proposed a knapsack-type public-key cryptosystem based on error-correction codes using GRS codes [42]. Subsequently, the Niederreiter method was demonstrated to be as secure as the McEliece cryptosystem. Sidel'nikov and Shestakov demonstrated in 1992 that Niederreiter's plan to employ GRS codes was insecure [48]. Various methods have been proposed to minimize the public key size by employing different codes, such as the Gabidulin code [18][19], algebraic geometry code [20][28], and Reed-Muller code [47]. However, all of these approaches ultimately proved to be unstable [34][44].

A. Recent PQ Secure Signature Primitives

This section covers recent advances in PQ cryptography and selects a set of suitable PQ secure algorithms. They can be used to meet our goal of developing a PQ secure signature and a PQ secure VRF for Green Bitcoin.

A digital signature (DS) algorithm comprises three parts. The first is the KeyGen component, which produces a public and private key pair. The second is the Sign part. Given the message and private key, it generates a signature. The third is the Verify component, which generates a binary pass or fail output based on a message and signature.

A VRF is similar to the DS algorithm in that it comprises three functions: a KeyGen function that generates a private and public key pair, a VRF function that outputs a signature (proof), and a random number, given the input of a private key and a message, and a Verify function that generates an output of pass or fail, given the input of the public key, message, random number, and proof.

We conducted preliminary research and discovered that Dilithium [36], Falcon [52], and Durandal [2] are good candidates for PQ-safe signature algorithms suitable for Green Bitcoin: the key metrics we used to select them are the size of the keys, the size of the signatures, the time it takes to complete a sign, and the time it takes to verify.

Let us now compare them. The time unit is msec. Dilithium requires 1.4, 6.2, and 1.5 for KeyGen, Sign, and Verify, respectively. Falcon, however, was 197.8, 38.1, and 0.5. Durandal received four, four, and five points, respectively. Durandal, therefore, has the quickest signature time compared with rival methods. The signature time corresponds to the VRF generation time. Thus, Durandal could serve as the first candidate for building a fast PQ-safe VRF. It offers a code-based DS algorithm for ranking metrics. While rendering a quantum-safe signature, it is sufficiently concise. The signature is 4 kb (kilo byte) long, whereas the public key is approximately 20 kb. The signature and verification processes took only 4 ms and 5 ms, respectively. It is powerful, quick, and concise enough to be considered a candidate for a worldwide public monetary-grade blockchain, such as Green Bitcoin.

We proceed to carefully study these candidate PQ secure algorithms and select the best one that satisfies all the key performance metrics of Green Bitcoin. The selected signature algorithm will be implemented using C++, and we will replace elliptic curve DS cryptography.

B. Novel PQ Secure VRF, and VCT Functions

We aimed to discuss how to create a novel PQ secure VCT function. To this end, we first need to create a good PQ secure VRF function. We then use it to create the VCT function.

A VRF is a function that generates a unique random number with a unique signature (proof) attached to it, given a private key and message. It is distinguished from an ordinary random number generator because it also offers a verification procedure. Thus, any verifier can check whether the random number is properly calculated. The quality of the random number generated must be high; given the size of the keys, the entropy of the random number is maximized.

A VRF is similar to a DS scheme, as mentioned in Section V.A. However, a significant difference was observed. It is the signature. The signature for a DS method needs to be non-unique stochastic by design. Stochastic signatures can increase security. However, for a VRF, the signature must be unique to each fixed input. We recall our goal of using VRF to save energy in the SolComp stage. Hence, the VRF should be designed to execute just once and only once every block; otherwise, the node would abuse it by running the VRF as many times as possible until the node produces a suitable output; no energy savings are realized as a result. Such enforcement is achieved if VRF generates a unique signature for a given fixed input message.

The input can be designed to be a piece of public information that existed before the VRF was performed, such as the block header of previous block. The same applies to the public keys. The public key should have been already posted somewhere in the blockchain before running the VRF. Therefore, the private key associated to the fixed public key is fixated. As the result, each node cannot but run it once and only once per each block.

C. Our Approach

Creating a new routine within a cryptographic algorithm is generally difficult because it changes the method's security output. However, it appears that there are a few options for this scenario. Any modification to make a unique proof reduces the degree of freedom (DoF) in the proof part. The system can remain secure if the same amount of DoF is increased elsewhere. To illustrate, suppose that the DoF in the proof is reduced so that the proof is made unique for a given fixed input. We now aim to discuss two possible directions. First, we add the same degree of DoF to the random value; the VCT function may be made to accommodate this modification, which slightly increases the size of the random value. Second, we may take the approach of increasing the same degree of DoF in the private key and this will slightly increase the size of the private key. A right balance shall be found to ensure that such a change does not compromise security.

D. Cryptanalysis for PQ secure VRF

The security of the proposed VRF can be evaluated using routine methods [3][4] such as (a) uniqueness analysis, (b) collision resistance analysis, and (c) pseudo-randomness analysis. Our proposed VRF candidate's performance can be measured by analyzing the time required for making the proposed signature, followed by the hashing time and the overall time, which includes key generation, sign, proof, verification, and building a block [13][51]. Other factors, such as the size of private and public keys, length of signature and hash, computational complexity, and energy consumption, can all be included to evaluate efficiency [16].

E. How to Make a VCT function from the VRF Output?

Given a new good VRF function, we aim to utilize it to create the VTC function. The VCT function accepts the result of the VRF function as input and produces a binary output, pass or fail. Namely, the output space of VRF can be partitioned into two. VRF output is a number. To create a VCT, all we need is a threshold. If VRF output is a number larger than or equal to the threshold, then VCT is set to give a pass output. The probability of pass can then be controlled by raising or lowering the threshold. The probability of pass can be set for any particular energy efficiency goal.

In addition, the probability of pass can be weighted based on the option the network designer can choose. For example, the designer can choose to enable a PoS option. A node's probability of passing can be determined by the stake it has made on chain.

F. PQ Safe Error-Correction Codes PoW

For the VC part, we aim to use ECCPoW for its ASIC resistance, simplistic, time-varying, PQ ready properties. ECCPoW is a new VC method the first author has developed and published [30][31][45]. It is based on an error-correction code called low-density parity check (LDPC) code. It works like an error-correction code (ECC) based cryptosystem. The resistance characteristic of ECC cryptosystems to quantum Fourier sampling attacks has been demonstrated [11]. Faster and more secure ECC cryptosystems are still under study [1]. Durandal, for example, is a lightweight and secure rank-metric code-based cryptosystem [2].

In summary, we aim to extend ECCPoW in two ways. The first is to make it a Green Bitcoin suitable (see M1–M6). The second is to make it PQ safer using medium-density codes. We also aim to ensure that these extensions are safe from well-known security threats.

(Difficulty Control Algorithm) The tradeoff relationship between the amount of verifiable computation and energy expenditure can be precisely determined. This tradeoff relationship can be used to devise a difficulty control (DC) algorithm. The DC algorithm seeks to create blocks in a defined regular (average) interval while reacting to variation in the total number of participating p2p nodes over time.

Here, we discuss a way to make ECCPoW PQ safer. In ECCPoW, LDPC codes are used to generate time-varying crypto puzzles. We aim to replace it with moderate-density parity check (MDPC) codes and make ECCPoW PQ safer. LDPC codes do not possess any algebraic structure but only a simple combinatorial

property, i.e., sparsity in the parity-check matrix; this makes them postquantum secure. There have been various suggestions to make a McEliece scheme using LDPC codes [3][4][39][40]. The low-weight parity-check rows in the parity-check matrix correspond to low-weight codewords in dual codes. As such, sparsity can be utilized to draw cryptographic attacks. Consequently, MDPC codes have been proposed in which the density is increased about ten times. Furthermore, a quasi-cyclic structure was devised for shorter public and private keys [43]. One famous example of this cryptosystem is BIKE, one of the third-round algorithms in NIST Post-Quantum cryptography (PQC) standardization [1].

G. BFT and Energy Consumption Efficiency of Green Bitcoin

Each node runs the VCT, sees the outcome either as pass or fail, and advances itself to the verifiable computation stage if it is passed. Green Bitcoin, therefore, supports a BFT of 1/2. Suppose a base set of a certain size for Green Bitcoin nodes. To launch a 51% double-spending attack, the attacker must hold 51% of the seats on the SolComp Committee. Using VCT only decreases the overall size of the committee but does not affect its proportion. This necessitates the adversary to have 51% presence in the base set to launch a 51% attack. This remains valid regardless of VCT's pass probability.

Green Bitcoin can choose a certain pass probability (PP) to determine its network's energy consumption efficiency (ECE). An ECE of 90% can be achieved when PP is set to 10%.

VI. TESTBED AND DISCUSSION

A. Testbed

The Green Bitcoin protocol suite will be developed on an existing open-source Bitcoin suite. The opcode table will be upgraded with the developed Green Bitcoin cryptography, replacing elliptic curve encryption for sign and verification. We will present emulation results on a proof-of-concept (PoC) network with a larger number of p2p computers. The Amazon Web Services will be used. Nodes will be scattered around the globe. We aim to employ more than a thousand nodes. Quantum attacks [18] will be used to assess the security of the PoC network.

Similar to our ECCPoW implementation [30][45], we will use the C++ as the developer language. A new genesis block will be created. To build a client, Green Bitcoin will be selected as the consensus algorithm. After defining the chain ID and exporting the genesis, the test Green Bitcoin network will be launched.

B. Safeguarding against Profitable Double-Spending Attacks

The most secure PoW protocols are still vulnerable. Double-spending attacks are still possible to occur [27][33]. This problem is exacerbated when the network's computational power is small. By borrowing computational resources from a mining rig lending site, an attacker can launch a DS assault. If there is a profit-taking opportunity, an attack is possible. Such an opportunity opens up as long as profits overwhelm the costs. The key new finding in [28] is that profitable double-spending (PDS) attacks can occur even if honest nodes have more than 50% of the computational resources of the network. The attacker can attempt to double-spend a transaction whose stake exceeds the cost of leasing mining machines from a lending service. Green Bitcoin aims to safeguard networks against PDS attacks. Such assaults cannot be completely forbidden, but they can be discouraged by lowering the profit the attacker can make and increasing the cost the attacker must bear. Micropayments are not affected by this, but large transactions require attention. We will develop new APIs based on [28] and use them to secure large transactions. These will be made available to the global research community.

C. Discussion

Bitcoin improves self-sovereignty of individuals. One can move along with one's stored wealth in Bitcoin anywhere in the world. No powerful entity can confiscate the portable wealth stored in Bitcoins. One can memorize the pass phrase, move to a new country, and restore one's wallet. One does not need to worry about bandits on the travel route or a government's confiscation at the entry point of a port. One can make international payments anywhere in the world using Bitcoin. Currently, Bitcoin faces energy concerns. Green Bitcoin resolves this with its VCT and PoolResistantComp mechanism. Bitcoin faces quantum computer risk. Green Bitcoin provides PQ secure computations. Green Bitcoin is a new protocol designed to retain the merits of Bitcoin and addresses the two pressing concerns of Bitcoin.

VII. CONCLUSION

Imagine a society where sound money is restored and prevails; suppose there is no cheap money. In such a society, when a man has money in his savings account, it means that he has done something beneficial to others in the past. Because this was the only way for him to earn sound money. He earned it. He was paid by the payer. It means he satisfied the payer. To satisfy the payer, he must have worked hard to produce useful goods and services. He reached his silver age. He no longer can work and produce. But he has kept sound money in his wallet. It has retained good value because it is a sound money. He can use it to purchase the valuable goods and services he needs. He does not need a government for that. He could purchase a house in a nice neighborhood and put food on the table for his family. Unearned income disappears in a world dominated by sound money, and honest work prevails.

We made two technical proposals to enhance Bitcoin. The new bitcoin (Green Bitcoin) addresses two issues: reduction of energy consumption and post-quantum computer risk. In the future, we plan to complete the Green Bitcoin protocol and bring forth a Green Bitcoin network to life. Green Bitcoin is a proposal to upgrade Bitcoin, the best sound cryptocurrency, into its quantum-safe and energy-efficient version.

REFERENCES

- [1] G. Alagic, J. Alperin-Sheriff, D. Apon, D. Cooper, Q. Dang, J. Kelsey, Y. K. Liu, C. Miller, D. Moody, R. Peralta, and R. Perlner, Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process, National Institute of Standards and Technology Interagency or Internal Report 8309 (NISTIR 8309), 2020.
- [2] N. Aragon, O. Blazy, P. Gaborit, A. Hauteville, and G. Zémor, “Durandal: A rank metric based signature scheme,” *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 728-758, Springer, 2019.
- [3] M. Baldi, M. Bodrato, and F. Chiaraluce, “A new analysis of the McEliece cryptosystem based on QC-LDPC codes,” *Proceedings of the 6th International Conference on Security and Cryptography for Networks*, Springer, pp. 246–262, 2008.
- [4] M. Baldi and F. Chiaraluce, “Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes,” *Proceedings of the IEEE Int. Symposium on Information Theory*, pp. 2591 –2595, 2007.
- [5] C. Beekhuizen, “Ethereum’s energy usage will soon decrease by ~99.95%,” *Ethereum Foundation Blog*, May 18, 2021. <https://blog.ethereum.org/2021/05/18/country-power-no-more/>.
- [6] T. Carlisle, “Chart of the DJIA priced in gold: Buffett and gold redux,” November 18, 2009. <https://greenbackd.com/2009/11/18/chart-of-the-djia-priced-in-gold/>.
- [7] A. Castor, “Why Ethereum is switching to proof of stake and how it will work,” March 4, 2022. <https://www.technologyreview.com/2022/03/04/1046636/ethereum-blockchain-proof-of-stake/>.
- [8] T. Dar, “Lighthouse confirms prototype merge between Ethereum and ETH2.0 with 99.98% less energy consumption,” March 26, 2021. <https://crypto8.com/2021/03/26/lighthouse-confirms-prototype-merge-between-ethereum-and-eth-2-0-with-99-98-less-energy-consumption/>.
- [9] N. De and D. Nelson, “SEC approves Bitcoin futures ETF, Opening crypto to wider investor base,” *CoinDesk*, October 16, 2021.
- [10] E. Deirmentzoglou, G. Papakyriakopoulos, and C. Patsakis, “A survey on long-range attacks for proof-of-stake protocols,” *IEEE Access*, vol. 7, pp. 28712-28725, 2019.
- [11] H. Dinh, C. Moore, and A. Russell, “McEliece and Niederreiter cryptosystems that resist quantum Fourier sampling attacks,” *Proceedings of the 31st Annual Conference on Advances in Cryptology*, Springer-Verlag, Berlin, Heidelberg, pp. 761–779, 2011.
- [12] D. Dolev, and H. R. Strong, “Authenticated algorithms for Byzantine agreement,” *SIAM Journal on Computing*, vol. 12, no. 4, pp. 656-666, 1983.
- [13] M. F. Esgin, V. Kuchta, A. Sakzad, R. Steinfeld, Z. Zhang, S. Sun, and S. Chu, “Practical post-quantum few-time verifiable random function with applications to Algorand,” *Proceedings of the International Conference on Financial Cryptography and Data Security*, pp. 560-578. Springer, Berlin, Heidelberg, 2021.
- [14] Ethereum Foundation, “The Merge,” July 28, 2022. <https://ethereum.org/en/upgrades/merge/>.

- [15] I. Eyal and E. G. Sirer, “How to disincentivize large Bitcoin mining pools,” June 18, 2014. <https://hackingdistributed.com/2014/06/18/how-to-disincentivize-large-bitcoin-mining-pools/>.
- [16] M. S. Ferdous, M. J. M. Chowdhury, M. A. Hoque, “A survey of consensus algorithms in public blockchain systems for crypto-currencies,” *Journal of Network and Computer Applications*, vol. 182, 2021.
- [17] T. M. Fernández-Caramès and P. Fraga-Lamas, “Towards post-quantum blockchain: A review on blockchain cryptography resistant to quantum computing attacks,” *IEEE Access*, vol. 8, pp. 21091–21116, 2020.
- [18] E. Gabidulin, A. Ourivski, B. Honary, and B. Ammar, “Reducible rank codes and their applications to cryptography,” *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3289–3293, 2003.
- [19] E. Gabidulin, A. Paramonov, and O. Tretjakov, “Ideals over a non-commutative ring and their applications to cryptography,” *Lecture Notes in Computer Science*, vol. 547, Springer Verlag, 1991.
- [20] P. Gaborit, “Shorter keys for code based cryptography,” *Proceedings of the 2005 International Workshop on Coding and Cryptography*, pp. 81–90, 2005.
- [21] G. Gilder, *Scandal of Money: Why Wall Street Recovers but the Economy Never Does*, Regnery Publishing, 2016.
- [22] F. A. Hayek, “Denationalization of Money, The Institute of Economic Affairs,” 1976. <https://iea.org.uk/wp-content/uploads/2016/07/Denationalisation%20of%20Money.pdf>.
- [23] D. Held, “Planting Bitcoin—species: Sound money (sanum pecuniam),” October 26, 2018. <https://danheldl.medium.com/planting-bitcoin-sound-money-72e80e40ff62>.
- [24] D. Held, “Bitcoin, the purchasing power preserver,” *Bitcoin Magazine*, July 11, 2022. <https://Bitcoinmagazine.com/markets/Bitcoin-purchasing-power-preserver>.
- [25] M. E. Herradi and A. Leroy, “Monetary Policy and the Top 1%: Evidence from a Century of Modern Economic History,” *International Journal of Central Banking*, vol. 18, no. 5, pp. 237–277, 2021.
- [26] IBM Newsroom, “IBM Unveils New Roadmap to Practical Quantum Computing Era; Plans to Deliver 4,000+ Qubit System,” May 10, 2022. <https://newsroom.ibm.com/2022-05-10-IBM-Unveils-New-Roadmap-to-Practical-Quantum-Computing-Era-Plans-to-Deliver-4,000-Qubit-System>.
- [27] J. Jang and H.-N. Lee, “Profitable double-spending attacks,” *Applied Sciences*, vol. 10, p. 8477, 2020.
- [28] H. Janwa and O. Moreno, “McEliece public key cryptosystems using algebraic geometric codes,” *Designs, Codes and Cryptography*, vol. 8, pp. 293–307, 1996.
- [29] T. J. Jordan, “What constitutes sound money?,” *Economic Conference, Progress Foundation*, Swiss National Bank, Zurich, October 8, 2020, https://www.snb.ch/en/mmr/speeches/id/ref_20201008_tjn.
- [30] H. Jung, and H.-N. Lee, “ECCPoW: Error-correction code based proof-of-work for ASIC resistance,” *Symmetry*, vol. 12, no. 6, pp. 988, 2020.
- [31] H. Kim, J. Jang, S. Park, and H.-N. Lee, “Error-correction code proof-of-work on Ethereum,” *IEEE Access*, vol. 9, pp. 135942–135952, 2021.
- [32] L. Lamport, R. Shostak, and M. Pease, “The Byzantine generals problem,” *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 387–389, 1982.
- [33] H.-N. Lee, “Blockchain and future society,” *4th Lecture Module: PoW Success Probability and Alone-Impossible-Together-Possible Theory*, October 31, 2019. Massive Open Online Course Lecture Video <https://gist.edwith.org/Bitcoin-tech> and lecture note https://infonet.gist.ac.kr/?page_id=8954.
- [34] P. Lee and E. Brickell, “An observation on the security of McEliece’s public key cryptosystem,” *Lecture Notes in Computer Science*, vol. 330, LNCS, pp. 275–280, Springer Verlag, 1989.
- [35] W. Li, S. Andreina, J.-M. Bohli, and G. Karame, “Securing proof-of-stake blockchain protocols,” *Data Privacy Management, Cryptocurrencies and Blockchain Technology*, pp. 297–315. Springer, 2017.
- [36] V. Lyubashevsky, L. Ducas, E. Kiltz, T. Lepoint, P. Schwabe, G. Seiler, D. Stehlé, and S. Bai, “Crystals-dilithium,” *NIST Post-Quantum Cryptography Standardization*, 2017.
- [37] R. McEliece, “A public key cryptosystem based on algebraic coding theory,” *DSN Progress Report*, pp. 42–44:114–116, 1978.
- [38] B. McElrath, “What’s wrong with proof-of-stake?” April 13, 2022. <https://medium.com/@BobMcElrath/whats-wrong-with-proof-of-stake-77d4f370be15>.
- [39] R. Misoczki, J.-P. Tillich, N. Sendrier, and P. L. S. M. Barreto, “MDPC-McEliece: New McEliece variants from moderate density parity-check codes,” *Proceedings of the IEEE International Symposium on Information Theory*, pp. 2069–2073, Istanbul, Turkey, 2013.

- [40] C. Monico, J. Rosenthal, and A. Shokrollahi, "Using low density parity check codes in the McEliece cryptosystem," *Proceedings of the IEEE International Symposium on Information Theory*, p. 215, 2000.
- [41] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008. <https://bitcoin.org/bitcoin.pdf>.
- [42] H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," *Problems of Control and Information Theory*, pp. 15:19–34, 1986.
- [43] A. Otmani, J. Tillich, and L. Dallot, "Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes," *Mathematics in Computer Science*, vol. 3, no. 2, pp. 129–140, 2010.
- [44] R. Overbeck, "A new structural attack for GPT and variants," *Lecture Notes in Computer Science*, vol. 3715, pp. 50–63. Springer Verlag, 2005.
- [45] S. Park, H. Choi, H.-N. Lee, "Time-variant proof-of-work using error-correction codes," June 22, 2020. <https://arxiv.org/abs/2006.12306>.
- [46] M. Rosenfeld, "Analysis of bitcoin pooled mining reward systems," 2011. <https://arxiv.org/abs/1112.4980>.
- [47] V. Sidelnikov, "A public-key cryptosystem based on binary Reed-Muller codes," *Discrete Mathematics and Applications*, vol. 4, no. 3, pp. 191–207, 1994.
- [48] V. Sidelnikov, Michilovich, and S. O. Shestakov, "On insecurity of cryptosystems based on generalized Reed-Solomon codes," pp. 439–444, 1992.
- [49] J. Starn and J. Saul, "Why Bitcoin's Environmental Problems Are So Hard to Fix," *Washington Post*, March 29, 2022. https://www.washingtonpost.com/business/energy/why-bitcoins-environmental-problems-are-so-hard-to-fix/2022/03/16/b71e1d22-a4df-11ec-8628-3da4fa8f8714_story.html.
- [50] The White House, "National Security Memorandum on Promoting United States Leadership in Quantum Computing While Mitigating Risks to Vulnerable Cryptographic Systems," May 4, 2022. <https://www.whitehouse.gov/briefing-room/statements-releases/2022/05/04/national-security-memorandum-on-promoting-united-states-leadership-in-quantum-computing-while-mitigating-risks-to-vulnerable-cryptographic-systems/>.
- [51] L. Zengpeng, T. G. Tan, P. Szalachowski, V. Sharma, and J. Zhou, "Post-quantum VRF and its applications in future-proof blockchain system," 2021. <https://arxiv.org/abs/2109.02012>.
- [52] X. Zhang, W. Wu, S. Yang, and X. Wang, "Falcon: a blockchain-based edge service migration framework in MEC," *Mobile Information Systems*, 2020. <https://doi.org/10.1155/2020/8820507>.
- [53] "Lightning network," July 1, 2022. <https://lightning.network/>.
- [54] "Mining: Bitcoin takes more energy than mining gold," *Nature*, November 6, 2018. <https://doi-org.lib-proxy.gist.ac.kr/10.1038/d41586-018-07283-3>.
- [55] "Nixon Shock", August 12, 2022. https://en.wikipedia.org/wiki/Nixon_shock.
- [56] "Why sharding is great: demystifying the technical properties," July 28, 2022. <https://vitalik.ca/general/2021/04/07/sharding.html>.

Heung-No Lee received the B.S., M.S., and Ph.D. degrees from the University of California, Los Angeles, CA, USA, in 1993, 1994, and 1999, respectively, all in electrical engineering. He worked at HRL Laboratories, LLC, Malibu, CA, USA, from 1999 to 2002 as a Research Staff Member. From 2002 to 2008, he was Assistant Professor with the University of Pittsburgh, PA, USA. Since 2009, he has been with the School of Electrical Engineering and Computer Science, GIST, South Korea, where he is a full professor. His technical works are in information theory, signal processing theory, communications/networking theory, and their applications to Wireless Communications, Networking, Medical Imaging, Brain-Computer Interfaces, Spectroscopy, Cryptocurrencies and Decentralized Finance.

Young-Sik Kim received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from Seoul National University, in 2001, 2003, and 2007, respectively. He joined the Semiconductor Division, Samsung Electronics, where he worked in the research and development of security hardware IPs for various embedded systems, including modular exponentiation hardware accelerator (called Tornado 2MX2) for RSA and elliptic-curve cryptography in smartcard products and mobile application processors, until 2010. He is currently a professor with Chosun University, Gwangju, South Korea. He is also a submitter for two candidate algorithms (McNie and pqsigRM) in the first round for the NIST Post Quantum Cryptography Standardization and a submitter for three candidate algorithms for Korean Post-Quantum Cryptography Competition (KpqC).

Dilbag Singh received a Ph.D. degree in computer science and engineering from the Thapar Institute of Engineering and Technology, Patiala, India, in 2019. He was an Assistant Professor at Chandigarh University, Mohali, India; Manipal University Jaipur, Jaipur, India; and Bennett University, Greater Noida, India. In 2021, he joined the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, South Korea, where he is currently affiliated. His research interests include image processing, computer vision, deep learning, metaheuristic techniques, and information security. He was in the top 2% list issues by “World Ranking of Top 2% Scientists” in 2021.

Manjit Kaur received the M.E. degree in information technology from Punjab University, Chandigarh, India, in 2011, and the Ph.D. degree from the Thapar Institute of Engineering and Technology, Patiala, India, in 2019. She was an Assistant Professor at Chandigarh University, Mohali, India; Manipal University Jaipur, Jaipur, India; and Bennett University, Greater Noida, India. In 2021, she joined the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, South Korea, where she is currently affiliated. Her research interests include wireless sensor networks, digital image processing, and metaheuristic techniques. She was in the top 2% list issues by “World Ranking of Top 2% Scientists” in 2021.

Error-Correction Code Proof-of-Work on Ethereum

HYOUNGSUNG KIM¹, JEHYUK JANG¹, SANGJUN PARK²,
AND HEUNG-NO LEE¹, (Senior Member, IEEE)

¹School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea

²Electronics and Telecommunications Research Institute (ETRI), Gwangju 500-712, South Korea

Corresponding author: Heung-No Lee (heungno@gist.ac.kr)

This research was supported in part by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2021-0-01835) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation), in the part by the IITP Grant through Korean Government MSIT under Grant 2020-0-00958, and in part by the National Research Foundation of Korea (NRF) Grant through Korean Government MSIP under Grant NRF-2021R1A2B5B03002118.

ABSTRACT The error-correction code proof-of-work (ECCPoW) algorithm is based on a low-density parity-check (LDPC) code. ECCPoW can impede the advent of mining application-specific integrated circuits (ASICs) with its time-varying puzzle generation capability. Previous research studies on ECCPoW algorithm have presented its theory and implementation on Bitcoin. In this study, we have not only designed ECCPoW for Ethereum, called ETH-ECC, but have also implemented, simulated, and validated it. In the implementation, we have explained how ECCPoW algorithm has been integrated into Ethereum 1.0 as a new consensus algorithm. Furthermore, we have devised and implemented a new method for controlling the difficulty level in ETH-ECC. In the simulation, we have tested the performance of ETH-ECC using a large number of node tests and demonstrated that the ECCPoW Ethereum works well with automatic difficulty-level change capability in real-world experimental settings. In addition, we discuss how stable the block generation time (BGT) of ETH-ECC is. Specifically, one key issue we intend to investigate is the finiteness of the mean of ETH-ECC BGT. Owing to a time-varying cryptographic puzzle generation system in ECCPoW algorithm, BGT in the algorithm may lead to a long-tailed distribution. Thus, simulation tests have been performed to determine whether BGT distribution is not heavy-tailed and has a finite mean. If the distribution is heavy-tailed, stable transaction confirmation cannot be guaranteed. In the validation, we have presented statistical analysis results based on the two-sample Anderson–Darling test and discussed how the BGT distribution follows an exponential distribution which has a finite mean. Our implementation is available for download at <https://github.com/cryptoecc/ETH-ECC>.

INDEX TERMS Anderson–Darling test, ASIC-resistant, blockchain, error-correction codes, Ethereum, hypothesis test, LDPC, proof-of-work, simulation, statistical analysis.

I. INTRODUCTION

Blockchain is a peer-to-peer (P2P) network that consists of trustless nodes. In a reliable P2P network, no peers (nodes) would intentionally send wrong information to others. In contrast, in an unreliable P2P network (e.g., a group of trustless nodes), the possibility that some peers may send false information to others should be considered. For example, a node may spread wrong or fake information to others. To address these issues in an unreliable P2P network, Nakamoto proposed using blocks and chaining these blocks with a novel consensus algorithm [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Giambattista Grusso¹.

In a blockchain, a peer sends a new block containing transactions to other peers. These peers validate the received block and link it to the previous block when there is no problem in the received block, i.e., when the authenticity of the block has been verified. A consensus algorithm is used to accomplish this verification task. If a peer has sent false information to others, such information is detected by the consensus algorithm as there is no collusion among the peers. A generated block contains information about previous blocks, i.e., all blocks are chained; thus, if someone wants to change one block in a chain, all previous blocks of the block to be changed must also be changed. Therefore, unless the network is centralized within a particular group, sending fake information about previous blocks to new peers is impossible.

Therefore, to prevent collusion, an unreliable network should avoid centralization.

Nakamoto proposed a proof-of-work (PoW) system for a consensus algorithm. In the PoW system, peers repeat a type of work to solve a cryptographic puzzle using a hash function (e.g., SHA256 [1] and Keccak [2]). When a peer successfully solves a cryptographic puzzle, the peer generates a block. In addition, the peer gets an incentive as a reward for the work done. In an ideal PoW system, new nodes can join to work and receive as much reward as they complete work. However, with an increase in the price of reward, attempts have been made to centralize the network to monopolize incentives.

Centralization is a phenomenon that occurs in PoW-based blockchain networks. In blockchains using PoW as a consensus algorithm, an oligarchy of miners with a disproportionate share of computation resources can monopolize block generation. Such centralization negatively impacts the credibility of a blockchain. For example, in a centralized network, a group of dominant nodes can selectively filter out some transactions belonging to others for their benefit. New nodes will find it difficult to earn trust and join the network in the fear of possible unfair treatment [3], [4].

The emergence of application-specific integrated circuits (ASICs) has accelerated the centralization of PoW. As more nodes use ASICs in generating blocks, the computation complexity in block generation increases. Thus, it has become difficult to generate blocks using general-purpose units, such as a central processing unit (CPU) and a graphics processing unit (GPU). As a result, a few groups equipped with powerful ASICs have surfaced and centralized the blockchain networks. To avoid centralization, researchers have proposed the use of ASIC-resistant PoW (e.g., Ethash of [2], X11 of [12], and Random X of [24]) and alternative consensus algorithms (e.g., proof-of-stake, delegated proof-of-stake, and Byzantium fault tolerance [25]). Networks using alternative algorithms have presented lesser decentralization effects than those have using ASIC-resistant PoW [25]. Specifically, in networks using alternative algorithms, only limited participants can generate blocks, but ASIC-resistant PoW has no limit on the number of participants. Thus, ASIC-resistant PoW presents a more decentralized network than do alternative algorithms.

For an ASIC-resistant PoW, an error-correction code proof-of-work (ECCPoW) algorithm was proposed [6], [7]. In ECCPoW algorithms, a hash value of a previous block generates a varying parity-check matrix (PCM) for error correction. This varying PCM works as a cryptographic puzzle in ECCPoW. These time-varying cryptographic puzzles make ECCPoW ASIC resistant. It is possible to use an ASIC for a specific fixed cryptographic puzzle. In ECCPoW, every newly created puzzle differs from all the previously created puzzles. As a result, if there was an ASIC for ECCPoW, such an ASIC must cover a wide range of cryptographic puzzle generation systems. Such a system, however, would incur huge chip space and cost [10], [11].

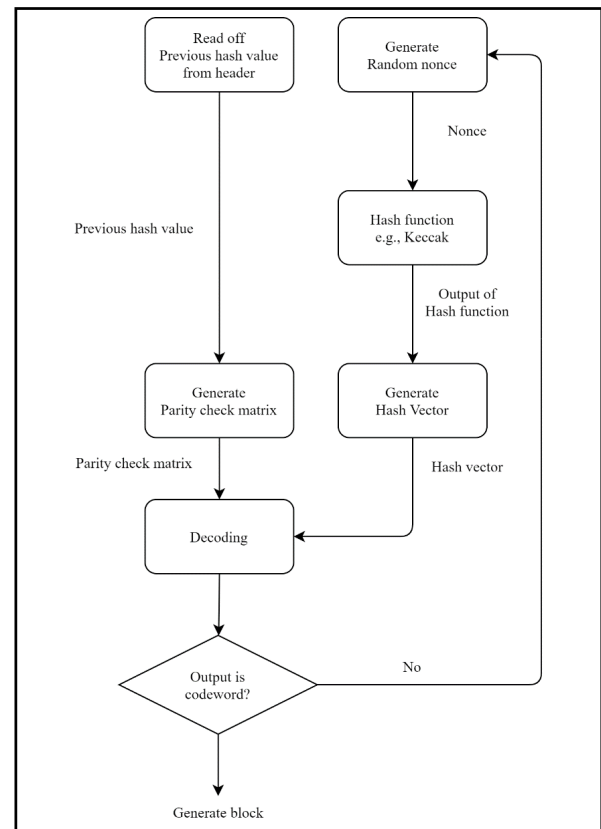


FIGURE 1. Flowchart of ECCPoW Ethereum. Every miner who generates blocks can construct a parity check matrix using a previous hash value. A generated nonce becomes an input of a hash function. A hash vector used for decoding can be generated using the output of a hash function. If decoding is successful, the block is generated; otherwise, a miner generates a new nonce to make a new hash vector for decoding.

In [7], the authors have reported that the time-varying puzzle system may generate large block generation time (BGT), i.e., outliers, for ECCPoW implemented on Bitcoin. If outliers occur frequently, it is of our interest to see whether or not the distribution of BGT is heavy-tailed with an infinite mean [15], [26]. As a result, the proposition made in [6] that BGT has a finite mean needs to be challenged. Previous works on ECCPoW [6], [7] did not include sufficient real-world experiments to conclude that BGT has a finite mean. If BGT does not have a finite mean, ECCPoW cannot be used as a consensus algorithm. In this paper, we aim to study the distribution of BGT of ECCPoW implemented on Ethereum (ETH-ECC). Our experimental results show that the BGT distribution is not heavy-tailed and has a finite mean.

The contributions of our work are as follows:

- We show how ECCPoW is implemented on Ethereum.
- We present a method for controlling the difficulty level in ETH-ECC and report the results of automatic difficulty level change with real-world experiments of ETH-ECC.
- We present a goodness-of-fit result using the Anderson–Darling (AD) test for distribution validation and discuss

the BGT distribution follows the exponential distribution which has a finite mean.

The remainder of this paper is organized as follows. Section II provides a background of the requirements of an ASIC-resistant PoW. Section III demonstrates the implementation of ETH-ECC. Section IV discusses the formulation of the problem. Section V provides the experimental result of the implemented ETH-ECC. Finally, Section VI summarizes our work and concludes the paper.

II. BACKGROUND

We introduce three approaches that can be used to avoid centralization problems in PoW. The first is an intentional bottleneck between an arithmetic logic unit (ALU) and memory, which is used by Ethash of Ethereum [2], [5]. It is also termed a memory-hard technique. The second is the *high complexity of ASIC design* used by Dash [12], Raven [13], and our method, ECCPoW. The third is *hybrid methods* of two methods; Random X of Monero uses *hybrid methods* [24].

A. INTENTIONAL BOTTLENECK

The most known PoW of the intentional bottleneck is Ethash of Ethereum [2], [5]. This method uses the difference between the throughput of ALU and the bandwidth of the memory. If there is a bottleneck between the ALU and memory, it is impossible to use the entire throughput of ALU. Specifically, if a miner needs to obtain data from memory to generate a block, the number of block generation attempts is determined by memory bandwidth. Ethash uses a directed acyclic graph (DAG), which is a set of randomly generated data for the bottleneck. The DAG is a huge dataset that cannot be stored in a cache memory; therefore, the DAG is stored in memory. To generate a block using Ethash, a miner must mix a part of the DAG that is stored in the memory. Owing to this procedure, the miner cannot avoid the bottleneck because of limited memory bandwidth. This method has been ASIC resistant for a long time; however, Bitmain released ASIC for Ethash in 2018.

B. HIGH COMPLEXITY OF ASIC DESIGN

Because of the high complexity of ASIC design, ASICs are less efficient. For example, if ASICs are less efficient than a general-purpose unit such as CPU or GPU, there is no reason to design ASIC. X11 of Dash [12] and X16R of Raven [13] use this method. Unlike PoW of Bitcoin, which uses only one hash function (SHA-256), X11 uses 11 hash functions consecutively: BLAKE, BMW, Grosetl, JH, Keccak, Skein, Luffa, Cubehash, SHAvite-3, SIMD, and ECHO. The BLAKE, which is the first hash function of X11, uses a block header with nonce as inputs; its output becomes the input of the next hash function. Similarly, the next hash function uses the output of the previous hash function. This procedure is repeated until a result is obtained for the last hash function. Miners determine whether they have found a valid nonce using the output of the final hash function.

Designing an ASIC for X11 was expensive; therefore, X11 was ASIC resistant. However, Bitmain released an ASIC for X11 in 2016. There are a few PoW algorithms that extend X11 (e.g., X13, X14, and X15); however, the ASICs for these have been released. X16R of Raven is an extended version of X11 of Dash. In X16R, unlike the previous extension of X11, the sequence of 16 hash functions is randomly changed. Therefore, it is costly to design an ASIC for X16R. However, T. Black, who designed X16R, mentioned that there is some evidence that ASICs for X16R exist [23]. Our ECCPoW employs a *time-varying puzzle generation system* to make ASIC design difficult. ECCPoW can make ASIC powerless as the puzzle generation system changes from block to block. We explain this further in Section III.

C. HYBRID METHODS

Random X of Monero combines the above two methods. Random X uses memory-hard techniques for the bottleneck with random code execution; Random X is optimized for CPU mining [24]. In [24], they mentioned that mining can be performed using a field-programmable gate array; however, it will be much less efficient than CPU mining. It implies that efficient mining hardware can be developed when the cost of developing chipsets is low in comparison to the mining reward. With the proposed ECCPoW, attempts in developing efficient mining hardware can be made when the reward-to-cost ratio increases. However, such attempts can be easily evaded since the parameters of ECCPoW can be easily changed, such as increasing the length of code and the code rate. The next section illustrates further the ASIC-resistance characteristic of ECCPoW.

III. ECCPoW IMPLEMENTED ON ETHEREUM

In this section, we briefly introduce ECCPoW and present how ECCPoW has been implemented on Ethereum using Fig. 1. Furthermore, we present how the difficulty level of ETH-ECC is automatically controlled.

A. OVERVIEW OF ECCPoW

In a blockchain employing the PoW consensus algorithm, a node solves cryptographic puzzles to publish a block. For a given puzzle, the node who solves the puzzle first obtains the authority to publish a block. For example, in the PoW of Bitcoin, the first node that finds a specific output of the secure hash algorithm (SHA) obtains the authority to publish a block. The PoW of Ethereum uses Keccak instead of SHA. The ECCPoW algorithm proposed in [6] is a PoW consensus algorithm that uses error-correction code, which comprises the low-density parity-check (LDPC) code [8], as a cryptographic puzzle. The ECCPoW algorithm consists of a pseudo-random puzzle generator (PRPG) and an ECC puzzle solver. Fig. 1 presents the flowchart of the ECCPoW algorithm. For every block, the PRPG generates a new pseudo-random LDPC matrix. A new LDPC matrix is distinct from the other previously generated matrices. Such a pseudo-random LDPC matrix takes the role of issuing an

independently announced cryptographic puzzle. The ECC puzzle solver uses the LDPC decoder to solve the given announced puzzle. Specifically, to publish a block, a node is required to run through an input header until the LDPC decoder hits a satisfying result; for instance, the output of the decoder is an LDPC codeword (with a certain Hamming weight). In the next subsection, we will discuss ECCPoW implementation on Ethereum with the flowchart presented in Fig. 1.

B. COMPARISON OF ETHASH AND ECCPoW

Ethereum uses Ethash for ASIC resistance, and ETH-ECC uses ECCPoW for ASIC resistance. In this subsection, we present how Ethash and ETH-ECC apply ASIC-resistance property to PoW with pseudo-codes.

Ethash uses a DAG for ASIC resistance. The DAG is a large size of data and is typically stored in a random access memory (RAM), not in cache memory. It implies that a miner must access the RAM to get the DAG data. Although the miner could be equipped with a high-throughput ALU, the bandwidth access from the RAM to the ALU is limited. That is, the bottleneck is the limited bandwidth of reading DAG information from the RAM; thus, any fast ALU, e.g., an ASIC implementation of keccak512, exceeding this bottleneck is of no use. This makes Ethash ASIC resistant.

When a miner reads DAG data from the RAM, the location where the data are read varies. The location of data reading is selected by the “mix”; the mix is a 128-byte hash value generated by the block header and a nonce. The mix is updated using the Fowler-Noll-Vo (FNV) hash function. The miner repeats this process 64 times. After updating the mix, the miner compresses the mix; for compression, the FNV hash is used again. The miner returns a hash value of the result of concatenating the compressed mix and the seed.

If this hash value is less than the desired target, the nonce is validated, and a new block is linked to the previous block. **Algorithm 1** denotes the pseudo-code of Ethash.

Algorithm 1 Ethash

Require: block header (BH), nonce, DAG

- 1: Initialize seed: $\text{seed} = \text{keccak512}(\text{BH}, \text{nonce})$
- 2: Initialize 128 bytes mix:
 $\text{mix} = \text{concatenate}(\text{seed}, \text{seed})$
- 3: **for** $i = 0, 1, 2, \dots, 63$:
- 4: Get data from DAG using mix:
 $\text{data} = \text{DAG_lookup}(\text{DAG}, \text{mix}, i)$
- 5: update mix: $\text{mix} = \text{FNV_hash}(\text{mix}, \text{data})$
- 6: **end for**
- 7: **for** $i = 0, 4, 8, \dots, \text{length}(\text{mix})$:
- 8: Compress mix: $\text{cmix} = \text{compress_mix}(\text{mix}, i)$
- 9: **end for**
- 10: **return** $\text{keccak256}(\text{concatenate}(\text{seed}, \text{cmix}))$

Ethash uses *the intentional bottleneck* for ASIC resistance, but ETH-ECC aims to use *a time-varying puzzle*

generation system for ASIC resistance. In ETH-ECC, two factors make the design of ASICs very difficult. One is flexible code lengths and randomly generated PCMs. The `ECC_puzzle_solver` generates a hash vector of length- n (subsection C) using a nonce; this n determines the code length. The development of an ASIC for a PCM with length n cannot be realized, as the ETH-ECC network changes n and the PCM from one block to another block. The `PRPG` creates a PCM \mathbf{H} . A PCM uses a BH as a seed; thus, it is randomly generated. All miners that work to extend the same previous block use the same PCM to solve the ECCPoW puzzle. Thus, it is highly expensive, if not impossible, to implement an ASIC that can handle a time-varying PCM [10], [11]. After generating a hash vector and a PCM, a miner works out how to generate an output word. If this output word satisfies a specific condition, the miner is successful at completing ECCPoW; e.g., the output word can be a codeword, and then, a new block is linked to the previous block. **Algorithm 2** denotes the pseudo-code of ETH-ECC. In our implementation, we have replaced Ethash and all its relevant peripheral systems with ECCPoW; thus, it has the same requirement as Ethash except for the DAG. We present more details about ETH-ECC in the following subsections.

Algorithm 2 ETH-ECC

Require: block header (BH), nonce

- 1: Generate hash vector:
 $\text{hash_vector} = \text{ECC_puzzle_solver}(\text{nonce})$
 - 2: Generate parity check matrix: $\text{PCM} = \text{PRPG}(\text{header})$
 - 3: $\text{output_word} = \text{decoder}(\text{PCM}, \text{hash_vector})$
 - 4: **return** output_word
-

C. ECCPoW ON ETHEREUM

In this subsection, we present how the error-correction process is applied to ETH-ECC using Fig. 1.

$$\mathbf{C} := \{\mathbf{c} | \mathbf{H}\mathbf{c} = \mathbf{0} \cap \mathbf{c} \in \{0, 1\}^{n \times 1}\} \quad (1)$$

when a PCM \mathbf{H} is given, a code \mathbf{c} , satisfying (1), is referred to as an LDPC code. The goal of the ECCPoW algorithm is to find an LDPC code \mathbf{c} using the PCM \mathbf{H} , which is derived by PRPG, and a hash vector \mathbf{r} , which is obtained using the ECC puzzle solver. For the PRPG, we employ the previous hash value; the previous hash value, known as the parent hash in the Ethereum block header, randomly generates a PCM. Specifically, we use Gallagher’s method to create random PCM [9]; we use the previous hash value as a seed of randomness. Thus, PCMs are changed for every block; because every node has the same seed, they use the same PCM until a block is generated [6].

1) ECC PUZZLE SOLVER ON ECCPoW ETHEREUM

Here, we introduce the ECC puzzle solver process in ETH-ECC. Our definitions are based on [6]. The equations below follow the right-hand side of Fig. 1.

Definition 1 (ECC Puzzle Solver): Hash vector \mathbf{r} in which the size of n can be obtained as follows:

$$s_1 := Keccak(nonce) \in \{0, 1\}^{256} \quad (2)$$

where *Keccak* denotes the hash function used in Ethash of Ethereum [5]. We generate a *nonce* in the same way that Ethereum does. Furthermore, for a longer length of a hash vector, we use $s_u := Keccak(s_1) \in \{0, 1\}^{256}$ with $u = 2, 3, \dots, l + 1$. We slice or concatenate the result of *Keccak* to generate a flexible length hash vector \mathbf{r} :

$$\mathbf{r} := \begin{cases} s_1[1 : n] & \text{if } n \leq 256 \\ [s_1 \cdots s_l s_{l+1}[1 : j]] & \text{if } n > 256 \end{cases} \quad (3)$$

where $l = \lfloor n/256 \rfloor$ and $j = n - 256 \times l$. For example, when n is less than 256, \mathbf{r} obtains the same length as n , whereas when n is not less than 256, \mathbf{r} concatenates the results of *Keccak*. This flexible length hash vector is used for ASIC resistance.

2) PoW OF THE LDPC DECODER

The goal of the LDPC decoder is to find a hash vector \mathbf{c} that satisfies $\mathbf{H}\mathbf{c} = \mathbf{0}$. The definition below explains the decoding presented in Fig. 1.

Definition 2 (Decoder): Given a PCM \mathbf{H} , which is the size of $m \times n$, and hash vector \mathbf{r} , which is the size of n , are given, the LDPC decoder uses \mathbf{H} and \mathbf{r} as inputs and obtains output \mathbf{c} using the message-passing algorithm [6], [14]. When \mathbf{c} satisfies (1), \mathbf{c} becomes an LDPC code, and a miner completes LDPC decoding.

$$D_{np} : \{\mathbf{r}, \mathbf{H}\} \mapsto \mathbf{c} \in \{0, 1\}^{n \times 1} \quad (4)$$

A PCM \mathbf{H} is randomly generated; however, all miners use the same previous hash value, which is derived from the previous block. Therefore, predicting the next PCM to mine a block in advance is impossible. In the PoW of Ethereum, miners change a nonce when they obtain a wrong output. We follow the same procedure as Ethereum to obtain a hash value from *Keccak* with a *nonce*, but ETH-ECC uses one more step (3) to generate a hash vector for decoding. When the code derived by (4) does not satisfy (1), the miner generates a new *nonce* and repeats all steps.

Our method is based on the *high complexity of ASIC design* in Section II for an ASIC-resistant PoW. However, unlike the mentioned method in Section II, ECCPoW generates varying cryptographic puzzles of *high complexity*. Specifically, ECCPoW uses two factors to achieve *high complexity*: flexible length LDPC code \mathbf{c} and randomly generated PCM \mathbf{H} . ASICs can be released for the n length of code. However, extending the length of code (e.g., $n + 1$) makes ASICs powerless. Furthermore, in [10], [11], it has been proven that implementing an ASIC that can handle variable PCMs is expensive and occupies a lot of space. If developing an ASIC costs more than buying a CPU or GPU, there is no incentive to develop an ASIC. In other words, the ECCPoW algorithm is ASIC resistant as implementing an ASIC that can handle various lengths of changing codes and randomly generated PCMs is inefficient.

D. DIFFICULTY-LEVEL CONTROL OF ETH-ECC

In this subsection, we demonstrate the implementation of ETH-ECC's difficulty-level control. Bitcoin [1] and Ethereum [2] have different difficulty-level control methods. Furthermore, we present one way to add fine difficulty control.

In Bitcoin, the Bitcoin network changes the difficulty level every 2016 block; the desired BGT is 10 min for a block. If miners generate a block every 10 min, generating 2016 blocks takes precisely 2 weeks. Thus, if generating 2016 blocks takes more than 2 weeks, the difficulty level decreases; otherwise, the difficulty level increases. Unlike Bitcoin, the Ethereum network changes the difficulty level every block. Ethereum network allows for a block to be generated between 9 and 18 s. If a block is generated within 9 s, then the difficulty level increases. If it exceeds 18 s, then the difficulty level decreases. Because of this difference between Bitcoin and Ethereum, ECCPoW-based Bitcoin (BIT-ECC) and ETH-ECC also have different difficulty-level control methods. Thus, ETH-ECC cannot use BIT-ECC's method. Because of the need for a new method, we demonstrate the implementation of ETH-ECC's difficulty level control with a difference from Ethereum's method.

Ethereum uses the number of attempts to generate a block per second, termed hash rate, and a probability of block generation. Similarly, ETH-ECC uses the hash rate but considers a probability of decoding success. In [5], the difficulty of Ethereum is defined by the probability of block generation. The difficulty is as follows:

$$n \leq \frac{2^{256}}{Diff} \quad (5)$$

It indicates that

$$Diff \leq \frac{2^{256}}{n} \quad (6)$$

where n denotes the result of PoW and *Diff* denotes the difficulty of Ethereum. Thus, (6) means that when the difficulty level increases, the number of n that satisfies (6) decreases. Furthermore, we can consider that the reciprocal of difficulty is a probability of block generation. Ethereum uses this probability and hash rate to control BGT. For example, without replacement, when the probability of block generation is 1/150 and hash rate is 10 hash per second, brute force takes 15 s. If the hash rate increases, such as 20 hash per second, Ethereum's method adjusts the probability of BGT to 1/300. Thus, brute force takes 15 s even though the hash rate increases.

For ECCPoW, if we can calculate a probability of decoding success, it is possible to control the difficulty level similar to the process in Ethereum. Thus, it is important to know the probability of a successful LDPC decoding according to the LDPC parameter. We use the pseudo-probability of a successful LDPC decoding according to the parameters to test the difficulty level change using the BGT [7]. That is, ETH-ECC uses the probability of decoding success and

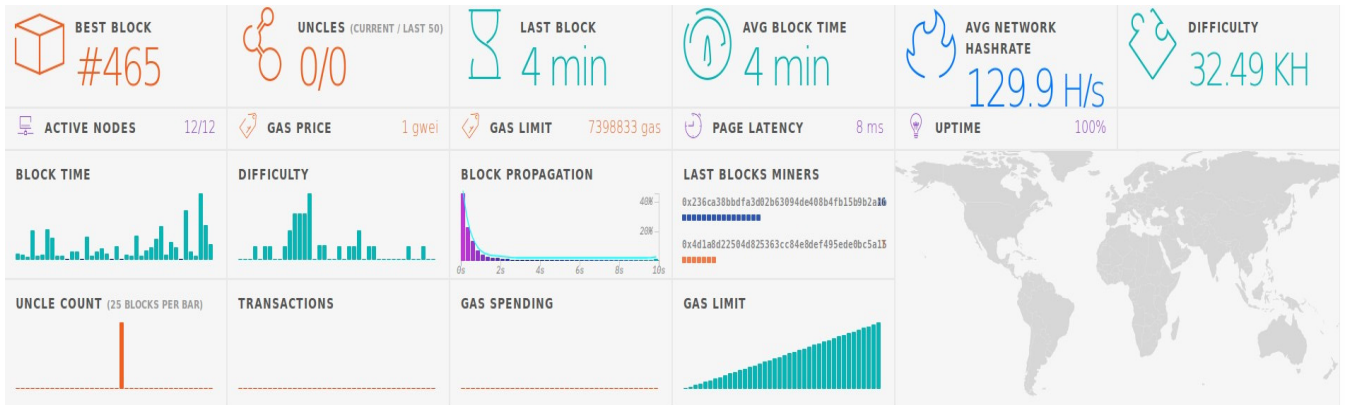


FIGURE 2. This figure shows the simulation results of ECCPoW Ethereum on Amazon Web Services AWS). Twelve nodes are used in the simulation. The tw nodes are bootnodes that help connect th nodes, and the other 10 nodes are sealnodes that participate i block generation. We use the m5.xlarge of AWS EC2 for the simulation. In the charts, BLOCK TIME shows the block generation times for the last 40 blocks, and DIFFICULTY shows the difficulty levels of the last 40 blocks. BLOCK PROPAGATION shows th percentage of the block propagation time corresponding to time.

hash rate to control the difficulty level. For example, without replacement, when the probability of decoding success is 1/150 and the hash rate is 10 hash per second, it takes 15 s, as in the above example of Ethereum’s method. However, unlike Ethereum, when the hash rate increase, ETH-ECC tunes parameters of LDPC to adjust the probability of decoding success. By tuning parameters, ECCPoW achieves both difficulty-level control and ASIC resistance. These parameters can be found at https://github.com/cryptoecc/ETH-ECC/blob/master/consensus/eccpow/LDPCDifficulty_utils.go#L65. In Fig. 2, the difficulty of ETH-ECC is 32.49 KH, indicating that the probability of block generation is 1 of 32,490 hash.

One Way to Add Fine Difficulty Control. ECCPoW controls difficulty using integer and discrete variable n . Thus, it may look inappropriate to manage difficulty precisely. However, as the number of blocks increase, block generation time (BGT) converges to the ideal BGT time, which is suitable for a network. For example, when there exist two difficulties: n and $n+1$, we can define average BGT of each difficulty as t_n and t_{n+1} . Thus, we can define the average BGT:

$$averageBGT = \frac{\alpha t_n + \beta t_{n+1}}{k} \tag{7}$$

where α denotes the number of generated blocks with difficulty t_n , β denotes the number of generated blocks with difficulty t_{n+1} , and k denotes the total number of generated blocks (TNGB). Thus, α can be replaced as $\alpha = k - \beta$. As a result, equation (7) is:

$$averageBGT = \frac{(k - \beta)t_n + \beta t_{n+1}}{k} \tag{8}$$

When TNGB k is kept constant, the average BGT is determined by the number of generated blocks β in Eq. (8). Thus, the ideal average BGT, which is suitable for the number of nodes in a network, depends on β . In other words, when TNGB k is low, the average BGT cannot meet the ideal

average BGT because there are not enough blocks of each difficulty. However, as TNGB k increases, the number of blocks corresponding to the difficulty, such as β , getting closer to the proportion that fits the probability of block generation. As a result, average BGT converges to the ideal average BGT; this convergence confirms our proposition that the network can control difficulty precisely.

IV. PROBLEM FORMULATION

In PoW, there is a case that nodes generate blocks at the same time. Bitcoin allows only one block to be generated at a time; Ethereum allows three blocks to generate at the same time. However, in Ethereum, only one block can be canonical; the other blocks cannot. Blocks that cannot be canonical are called uncle blocks. In Ethereum, nodes rollback transactions of uncle blocks [5]. Therefore, the transaction’s participants must wait for block confirmation to prevent a rollback. That is, in the blockchain using PoW, the BGT must have a finite mean for the block confirmation time. For example, if the BGT has an infinite mean, the waiting time for the confirmation of transactions cannot be determined. Therefore, to apply the ECCPoW algorithm in a real network, the BGT must have a finite mean.

In [6], the authors presented the definition of the block generation of the ECCPoW algorithm using a hash rate with a geometric distribution. That is, they assumed that nodes generate a block with specific block generation attempts. However, if the BGT has an infinite mean, there is no guarantee that nodes generate a block with specific attempts. In [7], the authors presented a practical experiment using the ECCPoW algorithm. However, they only mentioned that the BGT of ECCPoW is “unstable.” That is, they mentioned that the BGT of ECCPoW has outliers; however, they did not present a discussion on the BGT. Thus, in this study, we present a discussion on the BGT. Specifically, our experimental result presents evidence that the exponential distribution describes the distribution of the BGT of ECCPoW.

V. EXPERIMENT ON ETH-ECC

In this section, we conduct experiments using ETH-ECC. First, we simulate the difficulty level change using multinode networks. Second, we conduct a goodness-of-fit experiment using the AD test [16]–[18] to discuss the distribution of the BGT with a fixed difficulty level.

A. SIMULATION OF THE DIFFICULTY CHANGE

We simulate the difficulty-level change employing Amazon Web Services (AWS) using 12 nodes. Two nodes are *bootnodes* that help connect the nodes, and the other 10 nodes are *sealnodes* that participate in block generation. In the charts presented in Fig. 2, *BLOCK TIME* presents the BGT of the last 40 blocks, and *DIFFICULTY* shows the difficulty level of the last 40 generated blocks. *BLOCK TIME* and *DIFFICULTY* show that because of the large standard deviation, a block is gradually generated despite the low difficulty level, as mentioned in [7]; in the next subsection, we discuss the BGT. In the charts presented in Fig. 2, *LAST BLOCK* shows the BGT of the previous block, and *AVG BLOCK TIME* shows the average of the BGT. In addition, *AVG NETWORK HASHRATE* shows the average hash rate of all miners. *BLOCK PROPAGATION* shows the block propagation time from a miner who generated a block to other miners. We used two different regions: Seoul and US East for *sealnodes*. Specifically, 3 of the 10 *sealnodes* are in the US East region, whereas the rest are in the Seoul region. *BLOCK PROPAGATION* also shows the percentage of blocks that are propagated at corresponding times. *BLOCK PROPAGATION* indicates that the propagation of approximately all blocks between Seoul and US East regions takes less than 2 s. The block propagation is the same method as that of Ethereum.

B. STABILITY OF THE BLOCK GENERATION TIME

Fig. 2 demonstrates the importance of determining whether varying puzzles may result in outliers. That is, in *BLOCK TIME* and *DIFFICULTY* of Fig. 2, slow block generations are observed despite the low difficulty level. In other words, the observation of BGT shows outliers. If the outliers are uncontrollable, the BGT distribution has an infinite mean similar to the heavy-tailed distribution. An infinite mean cannot guarantee transaction confirmation. Thus, to achieve a stable BGT that can guarantee transaction confirmation, the BGT must have a finite mean.

We obtain the BGT of ECCPoW Ethereum with a fixed difficulty level to observe the type of distribution with a finite mean the BGT follows. Specifically, if BGT follows an exponential distribution, it has a finite mean. However, if the BGT follows a heavy-tailed distribution, it has an infinite mean [15]. Thus, through the goodness-of-fit experiment, we aimed to discuss what type of distribution the BGT follows. For the goodness-of-fit experiment, we set a null hypothesis H_0 and an alternative hypothesis H_A :

- H_0 : BGT has the exponential distribution
- H_A : BGT does not have the exponential distribution

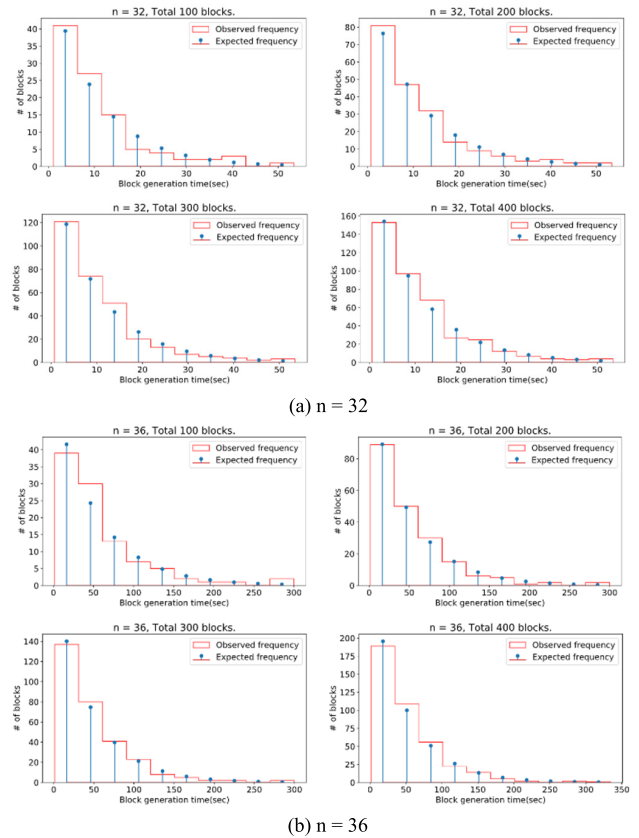


FIGURE 3. We did experiments for 100, 200, 300, and 400 blocks to observe the distribution over the number of blocks. As the number of blocks increases, the standard error decrease. That is, when the number of blocks increases, sample distribution reflects an actual distribution of sample distribution. In these figures, experiment results show the tendency; the distributions of observed frequency, known as sample distribution, follow the distribution of expected frequency.

For the goodness-of-fit experiment, we use the AD test [16]–[18]. Other available tests can be used in the goodness-of-fit experiment, such as the chi-square test [19], Kolmogorov–Smirnov test [20], and AD test [16]. The chi-square test has a restrictive assumption that all expected frequencies should be greater or equal to 5 [21]. However, there is no guarantee that our samples will achieve this assumption. If we collect more samples, the chi-square test can be used. However, the p -values used to validate the hypotheses are affected by the number of samples. When the number of samples increased in the chi-square test, the p -values tend to decrease. Therefore, the assumption of the chi-square test is inappropriate for verifying our distributions. The Kolmogorov–Smirnov test is unaffected by sample sizes; however, it is more sensitive to the center of the distribution rather than the tail [22]. We must consider verifying the tail of the distribution to cover all possibilities. Therefore, we have chosen to use the AD test [16], which gives more weight to the tail than does the Kolmogorov–Smirnov test.

C. AD TESTS

In this subsection, we discuss the AD test and verify its usage using test examples. The AD test is used to verify if a sample

TABLE 1. Example of the Anderson–Darling test results.

The number of samples	Standardized A_{MN}^2	p -value
10	-0.59	$p \geq 0.25$
20	0.44	$p = 0.21$
30	0.69	$p = 0.17$

(a) $F \sim \text{Exp}(1)$, $G \sim \text{Normal}(1, 1)$

10	1.20	$p = 0.11$
20	3.57	$p = 0.02$
30	4.67	$p = 0.01$

(b) $F \sim \text{Exp}(1)$, $G \sim \text{Exp}(2)$

10	1.11	$p = 0.12$
20	-0.41	$p \geq 0.25$
30	-0.08	$p \geq 0.25$

(c) $F \sim \text{Exp}(1)$, $G \sim \text{Exp}(1)$

follows a specific distribution. We discuss one-sample and two-sample AD tests. In our work, we use the two-sample AD test; however, to clearly present our contribution, we briefly introduce the one-sample AD test first.

1) ONE-SAMPLE AD TEST

The one-sample AD test is suitable to verify a hypothesis that a sample set comes from a population. The one-sample AD test is as follows. When the cumulative distribution function (CDF) of the population distribution is $F(x)$, and the CDF of the empirical distribution is $F_M(x)$, the one-sample AD test [18] is used as follows:

$$A_M^2 = M \int_{-\infty}^{\infty} (F_M(x) - F(x))^2 w(x) dF(x) \quad (9)$$

and

$$w(x) = [F(x)(1 - F(x))]^{-1} \quad (10)$$

where M denotes the number of samples and A_M^2 denotes the results of the one-sample AD test. Intuitively, in (9), if $F_M(x) - F(x)$ is 0 for all x , A_M^2 is 0. This means that when A_M^2 is small, the empirical distribution $F_M(x)$ is close to the population distribution $F(x)$. As we have noted, we focus on the tail of the distribution; it can be accomplished using (10). The one-sample AD test result A_M^2 can be used to verify if a given sample comes from a population with a specific distribution.

2) TWO-SAMPLE AD TEST

In our work, we want to verify that two-sample sets come from the same unknown population. The two-sample AD test is appropriate for such verification. The two-sample AD test [17], [18] is as follows. There are two-sample empirical distributions $F_M(x)$ and $G_N(x)$. The $F_M(x)$ is an empirical distribution derived from the set \mathcal{F} with a cardinality of the sample set $M = |\mathcal{F}|$. The $G_N(x)$ is also an empirical distribution derived from the set \mathcal{G} with a cardinality of the sample set $N = |\mathcal{G}|$. $F_M(x)$ and $G_N(x)$ are the respective sample sets

TABLE 2. The observed frequency is calculated using the histogram in Fig. 4, and the expected frequency is calculated using the CDF of the exponential distribution derived from the mean in Fig. 4.

Interval(%)	Observed frequency	Expected frequency
[0, 10)	107	118.70
[10, 20)	82	71.73
[20, 30)	56	43.35
[30, 40)	20	26.19
[40, 50)	14	15.83
[50, 60)	7	9.56
[60, 70)	5	5.78
[70, 80)	4	3.49
[80, 90)	2	2.11
[90, 100]	3	1.27

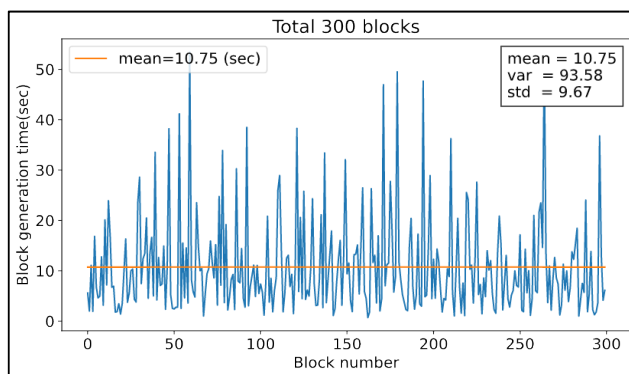


FIGURE 4. This figure presents block generation time of 300 blocks when n is 32. The mean block generation time of 300 blocks is 10.75 s, and it is presented as a horizontal line. Such a result is converted to a histogram. The observed frequency of Table 2 denotes the histogram of Fig. 4. The legend at the top right shows the mean, variance, and standard deviation of the BGT.

independently obtained from two different testing locations. The two-sample AD test can be used to determine whether both sample distributions come from the same distribution. In [17], [18], the two-sample version is defined as follows:

$$A_{MN}^2 = \frac{MN}{K} \int_{-\infty}^{\infty} \frac{(F_M(x) - G_N(x))^2}{H_K(x)(1 - H_K(x))} dH_K(x) \quad (11)$$

where $H_K(x) = (MF_M(x) + NG_N(x))/K$ with $K = M + N$. A_{MN}^2 is standardized to remove the dependencies derived by the number of samples. This standardized form is used to calculate the p -value [17], [18]. The p -value evidences the hypothesis test.

The two-sample AD test is suitable to verify a hypothesis that two-sample sets come from the same population. As a null hypothesis H_0 for the two-sample AD test, we set $F_M(x)$ to have the same population as $G_N(x)$. In addition, we set $G_N(x)$ as an exponential distribution. Thus, if $F_M(x)$ and $G_N(x)$ comes from same the population, that is, H_0 is true, we may consider that $F_M(x)$ is the exponential distribution. If the p -value of the AD test is sufficiently large, it proves that H_0 is true.

The p -value is the false positive probability under the assumption that the null hypothesis is true. A low p -value indicates that a test result provides evidence against the null hypothesis; a large p -value does not. That is, a large p -value denotes the probability of a true negative is low. The p -value is determined from the observation of the sample data. Thus, before observing the data, we first set the threshold significance level (TSL), $TSL \in [0, 1]$. The TSL can be used to determine the critical value. Given a TSL and the number of samples that are used in the AD test, the TSL table in [18] is used to read off a value corresponding to the TSL and the number of samples. This read-off value is called the critical value. If the standardized A_{MN}^2 is smaller than the critical value, this result indicates that the p -value is larger than the predefined TSL . In the TSL table of [18], the maximum TSL is 0.25. Thus, when standardized A_{MN}^2 is less than the critical value corresponding to the 0.25 TSL , the p -value is capped at 0.25.

3) VERIFICATION OF THE AD TEST

In this subsection, we verify the two-sample AD test method. Verification is performed under the assumption that the input distributions are *a priori* known. This will clearly illustrate how we will use the AD test and interpret its test results.

In Table 1, we present three examples to give an insight into the p -value of the AD test; in this example, we use true distributions for $F_M(x)$ and $G_N(x)$. In Table 1, $\text{Exp}(\theta)$ indicates the exponential distribution with mean θ and $\text{Normal}(\mu, \sigma)$ indicates the normal distribution with mean μ and standard deviation σ . That is, $\mathcal{F} \sim \text{Exp}(\theta)$ denotes the sample set \mathcal{F} of $F_M(x)$; samples are derived from the exponential distribution with mean θ . In Table 1 (a), we use the exponential distribution for $F_M(x)$ and the normal distribution for $G_N(x)$; these distributions have the same mean. This example shows that as the number of samples increases, the p -value tends to decrease if samples are drawn from different distributions. In Table 1 (b), we set both $F_M(x)$ and $G_N(x)$ as the exponential distribution, but each with different mean values. This example shows that as the number of samples increases even though samples are drawn from the same exponential distribution, the p -value tends to decrease if the means of distributions are different. In Table 1 (c), we set both $F_M(x)$ and $G_N(x)$ to be exactly the same exponential distribution. That is, the two-sample sets $\mathcal{F} \sim F_M(x)$ and $\mathcal{G} \sim G_N(x)$ come from the same population. This example shows that, as the number of samples increases, the p -value tends to increase when two-sample sets are drawn from the same population. From these examples in Table 1, we note that the closer the two distributions $F_M(x)$ and $G_N(x)$ are to each other, the larger p -value is obtained.

We determine whether the AD test result of our experiments indicates that $F_M(x)$ is sufficiently close to $G_N(x)$. That is, given there are two-sample sets, one of $F_M(x)$ and the other of the exponential $G_N(x)$, we want to determine whether we can make a quality statement about how close the two-sample sets are to each other according to the AD test.

TABLE 3. Anderson-Darling test result. The test result presents a large p -value. It means that if we reject the null hypothesis, the probability of a true negative is low.

n	# of blocks	Observed mean(sec)	std	Standardized A_{MN}^2	p -value
32	100	10.86	9.84	-1.12	$p \geq 0.25$
32	200	11.24	10.16	-1.20	$p \geq 0.25$
32	300	10.74	9.67	-1.18	$p \geq 0.25$
32	400	11.08	9.84	-1.09	$p \geq 0.25$
32	500	10.91	9.62	-1.11	$p \geq 0.25$
32	600	10.87	9.48	-0.80	$p \geq 0.25$
32	700	10.84	9.41	-0.36	$p \geq 0.25$
32	800	10.76	9.40	-0.36	$p \geq 0.25$
36	100	56.00	55.20	-1.11	$p \geq 0.25$
36	200	51.04	49.71	-1.19	$p \geq 0.25$
36	300	47.84	45.49	-1.12	$p \geq 0.25$
36	400	49.97	47.80	-1.19	$p \geq 0.25$
36	500	49.24	46.95	-1.11	$p \geq 0.25$
36	600	48.23	46.96	-1.18	$p \geq 0.25$
36	700	48.36	47.68	-1.18	$p \geq 0.25$
36	800	48.03	46.89	-1.18	$p \geq 0.25$

The AD test result presents a significant p -value, i.e., $p \geq 0.25$; it is a necessary condition but not a sufficient one for the case that the two distributions are the same. In other words, if a decision is made to reject the null hypothesis, that is, the distribution $F_M(x)$ is not close to the exponential distribution $G_N(x)$, such a decision will result in an error with a probability greater than 0.25.

D. APPLICATION OF AD TEST TO BGT DISTRIBUTION

In this subsection, we use the AD test to determine the distribution of the BGT of ETH-ECC. For this experiment, 90 threads were used to generate a block. We experimented using a fixed code length to observe the BGT without changing the difficulty level. In the test, two kinds of code length n are used: 32 and 36. These are the two lowest types of code length n in our pseudo-difficulty table used in the simulation. We divided the BGT into 10 intervals between the minimum BGT and maximum BGT for a histogram. For example, when the minimum BGT is 10 and the maximum BGT is 20, there are 10 intervals, i.e., [10,11], [11,12], ..., [19,20]. Using these intervals, we count the observed frequency of the BGT data. We set $F_M(x)$ using the observed frequency and set $G_N(x)$ using the mean of the BGT data. The mean in Fig. 4 is used for the expected frequency of $G_N(x)$ in Table 2. That is, the mean in Fig. 4 is used as $1/\lambda$ for the CDF of the exponential distribution $G_N(x)$:

$$G_N(x) = 1 - e^{-\lambda x} \quad (12)$$

The expected frequency of Table 1 is calculated using the integral of $G_N(x)$ corresponding to the interval time. Because $G_N(x)$ is the exponential distribution, if $F_M(x)$ is close to $G_N(x)$ we may consider $F_M(x)$ is an exponential distribution.

E. DISCUSSION ON AD TEST RESULTS

Fig. 4 shows the example result of the BGT over different blocks. Each block denotes the trial to obtain the BGT. We converted the test results, such as those in Fig. 4, to a distribution over time to analyze the BGT. These converted distributions are presented in Fig. 3. Fig. 3 presents the plots of the distribution of the observed and expected frequencies. These frequencies are calculated using the method described in Section V-D.

When we obtain a distribution using a sample set, there is a standard error; the standard error is high when the number of samples in the set is small. The standard error is expressed as

$$\frac{\sigma}{\sqrt{N}}$$

where σ denotes the standard deviation of a population and N denotes the cardinality of the sample set. The standard error decreases as the number of samples increases. Thus, the sample distribution becomes closer to the actual distribution of the observed samples. If the sample distribution, which reflects the actual distribution, differs from the expected distribution, we can observe that the sample distribution differs from the expected distribution. To observe the tendency of distribution over some blocks, we experimented with 100, 200, 300, and 400 blocks. Fig. 3 shows that the distribution of the observed frequency tends to follow the distribution of the expected frequency. In addition, Table 3 shows that the observed mean and standard deviation tend to converge as the number of blocks increases.

Furthermore, for the quantitative analysis, we use the AD test. Table 3 presents the AD test results to discuss hypotheses H_0 and H_A . These results show a similar result in Table 1 (c). In Table 1 (c), we drew samples from the same true distribution; the results present the largest possible p -value. All p -values in Table 3 are larger than or equal to 0.25, regardless of the number of blocks. In other words, if the null hypothesis is rejected, this decision will cause an error with a probability greater than 0.25. That is, the decision that the BGT distribution $F_M(x)$ does not follow the exponential distribution could be made with a high decision error.

VI. DISCUSSION

The purpose of ECCPoW is not to replace the current PoW of Ethereum. We propose our algorithm to present as one of the options for the Ethereum network. Ethereum can be utilized, for example, not only in a large-scale network but also in local-scale networks. To support a local-scale network, Ethereum provides PoW and PoA (Proof-of-Authority) as consensus algorithms. These algorithms have limitations for the local-scale network. For instance, PoW based local network has a risk of a double-spending attack by ASIC miners; PoA based network has a limitation of a participant because the time complexity of a PoA increases exponentially when the number of participants increases. Our algorithm, ECCPoW, can be utilized in such cases for the benefit of

offering a novel PoW that allows numerous participants with deterrence to ASIC-borne attacks. In addition, our novel ECCPoW may open up for an expected use and thus unraveled future to Ethereum.

Extensive Simulation Set up at AWS. We have recruited twelve instances on Amazon Web Service (AWS) EC2; each instance of EC2 instances works as a node in a blockchain network. The cost of using AWS EC2 increases rapidly because PoW utilizes all the resources of instances. We were able to confirm that this scale of the experiment was good enough to achieve our main goal, which is aimed at verifying the stability of the block generation time of ECCPoW Ethereum. AWS simulation was done to obtain the trace data of block generation times. The twelve nodes employed in our simulation were divided into two different kinds of nodes. One kind is *bootnodes* which help the nodes connected. Nodes that want to join a network are connected to *bootnodes* first. After connection, *bootnodes* relay nodes to other nodes. In Ethereum, *bootnodes* addresses are hardcoded on source codes, but it is possible to set *bootnodes* addresses manually for private networks. We have chosen two *bootnodes*. The other kind of nodes are *sealnodes* that participate in block generation as a miner in the PoW network. We have chosen the number of *sealnodes* to be 10. We use the m5.xlarge of AWS EC2, which has conventional node specification: four virtual CPUs and 16 GB memory for the real-world simulation. All nodes are deployed by Docker according to the guidance of Ethereum. Thus, all of our simulation results, which are shown in Fig. 2, are reproducible.

VII. CONCLUSION

In this work, we present the implementation, simulation, and validation of ETH-ECC. In the implementation, we showed how Ethereum can be updated with ECCPoW as its new consensus algorithm. In the simulation, we conducted a multi-node experiment using AWS EC2. The results showed that ETH-ECC with its adaptive difficulty-level controllability is successfully implemented. In the validation, we showed statistical results in which the necessary condition for a finite mean BGT is satisfied such that the distribution of the ECCPoW block generation time is exponential.

REFERENCES

- [1] S. Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. Accessed: Nov. 3, 2021. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>
- [2] V. Buterin. *A Next-Generation Smart Contract and Decentralized Application Platform*. Accessed: Nov. 3, 2021. [Online]. Available: <https://ethereum.org/en/whitepaper/>
- [3] M. Rosenfeld, "Analysis of hashrate-based double spending," 2014, *arXiv:1402.2009*. [Online]. Available: <http://arxiv.org/abs/1402.2009>
- [4] J. Jang and H.-N. Lee, "Profitable double-spending attacks," 2019, *arXiv:1903.01711*. [Online]. Available: <http://arxiv.org/abs/1903.01711>
- [5] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, pp. 1–32, Apr. 2014.
- [6] S. Park, H. Choi, and H. Lee, "Time-variant proof-of-work using error correction codes," Jun. 2020, *arXiv:2006.12306*. [Online]. Available: <https://arxiv.org/abs/2006.12306>
- [7] H. Jung and H. Lee, "Error-correction code based proof-of-work for ASIC resistance," *Symmetry*, vol. 12, no. 6, p. 988, Jun. 2020.

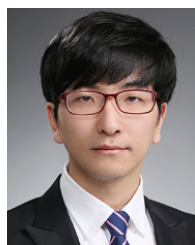
- [8] W. Ryan and S. Lin, "Low-density parity-check codes," in *Channel Codes: Classical and Modern*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
- [9] R. G. Gallager, *Low Density Parity Check Codes*. Cambridge, MA, USA: MIT Press, 1963.
- [10] S. Shao, P. Hailes, T.-Y. Wang, J.-Y. Wu, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "Survey of turbo, LDPC, and polar decoder ASIC implementations," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2309–2333, 3rd Quart., 2019.
- [11] Y.-L. Ueng, B.-J. Yang, C.-J. Yang, H.-C. Lee, and J.-D. Yang, "An efficient multi-standard LDPC decoder design using hardware-friendly shuffled decoding," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 3, pp. 743–756, Mar. 2013.
- [12] E. Duffield and D. Diaz, *Dash: A Payments-Focused Cryptocurrency*. Accessed: Nov. 3, 2021. [Online]. Available: <https://github.com/dashpay/dash/wiki/Whitepaper>
- [13] T. Black and J. Weight, *X16R ASIC Resistant by Design*. Accessed: Nov. 3, 2021. [Online]. Available: <https://ravencoin.org/assets/documents/X16R-Whitepaper.pdf>
- [14] W. E. Ryan and S. Lin, *Channel Codes—Classical and Modern*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
- [15] S. Foss, D. Korshunov, and S. Zachary, *An Introduction to Heavy-Tailed and Subexponential Distributions*. New York, NY, USA: Springer, 2011.
- [16] T. W. Anderson and D. A. Darling, "Asymptotic theory of certain 'goodness of fit' criteria based on stochastic processes," *Ann. Math. Statist.*, vol. 23, no. 2, pp. 193–212, Jun. 1952.
- [17] A. N. Pettitt, "A two-sample anderson-darling rank statistic," *Biometrika*, vol. 63, no. 1, pp. 161–168, 1976.
- [18] F. Scholz and M. Stephens, "K-sample anderson-darling tests," *J. Amer. Stat. Assoc.*, vol. 82, no. 399, pp. 918–924, Sep. 1987.
- [19] W. G. Cochran, "The X^2 test of goodness of fit," *Ann. Math. Statist.*, vol. 23, no. 3, pp. 315–345, Sep. 1952.
- [20] J. L. Hodges, "The significance probability of the Smirnov two-sample test," *Arkiv Matematik*, vol. 3, no. 5, pp. 469–486, Jan. 1958.
- [21] W. G. Cochran, "Some methods for strengthening the common tests," *Biometrics*, vol. 10, pp. 417–451, Dec. 1954.
- [22] J. J. Filliben, 1.3.5.16. *Kolmogorov–Smirnov Goodness-of-Fit Test*. NIST/SEMATECH e-Handbook of Statistical Methods. Accessed: Nov. 3, 2021. [Online]. Available: <http://www.itl.nist.gov/div898/handbook/>
- [23] T. Black, *Ravencoin–ASIC Thoughts—Round Two*. Accessed: Nov. 3, 2021. [Online]. Available: <https://medium.com/@tronblack/ravencoin-asic-thoughts-round-two-f4f743942656>
- [24] Tevador, *Random X*. Accessed: Nov. 3, 2021. [Online]. Available: <https://github.com/tevador/RandomX>
- [25] M. Belotti, N. Bozic, G. Pujolle, and S. Secci, "A vademecum on blockchain technologies: When, which, and how," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3796–3838, 4th Quart., 2019.
- [26] S. Asmussen, "Steady-state properties of $G/G/1$," in *Applied Probability and Queues*. New York, NY, USA: Springer, 2003, pp. 266–301, doi: 10.1007/0-387-21525-5_10.



HYONGSUNG KIM received the B.S. degree in electronics and computer engineering from Chonnam National University, Gwangju, South Korea, in 2019, and the M.S. degree from the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju. Since 2021, he has working with Korea Electronics Technology Institute. His research interests include blockchain, distributed systems, and statistical analysis.



JEHYUK JANG received the B.S. degree in electronic engineering from the Kumoh National Institute of Technology, Gumi, South Korea, in 2014, and the M.S. degree in information and communication engineering from GIST, South Korea, where he is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Computer Science. His research interests include sub-Nyquist sampling and compressed sensing.



SANGJUN PARK received the B.S. degree in computer engineering from Chungnam National University, Daejeon, South Korea, in 2009, and the Ph.D. degree in electrical engineering and computer science from Gwangju Institute of Science and Technology, Gwangju, South Korea, in 2019. Since 2020, he has been working with the Electronics and Telecommunications Research Institute. His research interests include information theory, numerical optimization, compressed sensing, blockchain, deep-neural networks, and finite state machine.



HEUNG-NO LEE (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees from the University of California, Los Angeles, CA, USA, in 1993, 1994, and 1999, respectively, all in electrical engineering. From 1999 to 2002, he was a Research Staff Member with HRL Laboratories, LLC, Malibu, CA, USA. From 2002 to 2008, he was an Assistant Professor with the University of Pittsburgh, PA, USA. In 2009, he was with the School of Electrical Engineering and Computer Science, GIST, South Korea, where he is currently affiliated. His research interests include information theory, signal processing theory, blockchain communications/networking theory, and their application to wireless communications and networking, compressive sensing, future internet, and brain–computer interface. He was a recipient of several prestigious national awards, including the Top 100 National Research and Development Award, in 2012; the Top 50 Achievements of Fundamental Researches Award, in 2013; and the Science/Engineer of the Month, in January 2014.

...

An Information-Theoretic Study for Joint Sparsity Pattern Recovery With Different Sensing Matrices

Sangjun Park, Nam Yul Yu, *Member, IEEE*, and Heung-No Lee, *Senior Member, IEEE*

Abstract—In this paper, we study a support set reconstruction problem for multiple measurement vectors (MMV) with different sensing matrices, where the signals of interest are assumed to be jointly sparse and each signal is sampled by its own sensing matrix in the presence of noise. Using mathematical tools, we develop upper and lower bounds of the failure probability of the support set reconstruction in terms of the sparsity, the ambient dimension, the minimum signal-to-noise ratio, the number of measurement vectors, and the number of measurements. These bounds can be used to provide guidelines for determining the system parameters for various compressed sensing applications with noisy MMV with different sensing matrices. Based on the bounds, we develop necessary and sufficient conditions for reliable support set reconstruction. We interpret these conditions to provide theoretical explanations regarding the benefits of taking more measurement vectors. We then compare our sufficient condition with the existing results for noisy MMV with the same sensing matrix. As a result, we show that noisy MMV with different sensing matrices may require fewer measurements for reliable support set reconstruction, under a sublinear sparsity regime in a low noise-level scenario.

Index Terms—Compressed sensing, support set reconstruction, joint sparsity structure, multiple measurement vectors model.

I. INTRODUCTION

CONVENTIONALLY, signals sensed from sensors such as microphones and imaging devices are sampled following the Shannon and Nyquist sampling theory [1] at a rate higher than twice the maximum frequency for signal reconstruction. As the number of samples decided by this theory is often large, the samples go through a compression stage before being stored. Therefore, taking numerous samples, where most of them will be discarded in this stage, is inefficient. Because compressed sensing (CS) [2]–[7] removes the inefficiency, CS has been applied in various areas such as wireless communications [8]–[11], spectrometers [12], multiple input multiple output (MIMO) radars [13], magnetic resonance imaging [14], and imaging/signal processing [15]–[17].

The CS theory states that signals that are sparsely representable in a certain basis are compressively sampled and reconstructed from what we thought is incomplete

information. Let $\mathbf{x} \in \mathbb{R}^N$ be a K -sparse vector with a support set $\mathcal{I} := \{i | x(i) \neq 0\}$ whose indices indicate the positions of the nonzero coefficients of \mathbf{x} . It is compressively sampled by a model called *single measurement vector (SMV)* as follows:

$$\mathbf{y} = \mathbf{F}\mathbf{x} + \mathbf{n} \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^M$ is a (noisy) measurement vector, $\mathbf{F} \in \mathbb{R}^{M \times N}$ is a sensing matrix, and $\mathbf{n} \in \mathbb{R}^M$ is a noise vector, whose elements are independent and identically distributed (i.i.d) Gaussian with a zero mean and a σ^2 variance. Once the support set is correctly reconstructed, then (1) can be well-posed, which allows us to obtain an accurate estimate of \mathbf{x} using the least square approach. We thus aim to focus on the support set reconstruction problem.

A. Information-Theoretic Works for CS With SMV

Works [18]–[23] have studied the support set reconstruction problem from an information-theoretic perspective. For reliable support set reconstruction, sufficient and necessary conditions were established in the linear and sublinear sparsity regimes.

For support set reconstruction, Wainwright [18] used the union bound to establish a sufficient condition on the number of measurements M for a maximum likelihood (ML) decoder and used Fano's inequality [24] to obtain a necessary condition on M . This ML decoder was analyzed by Fletcher *et al.* [19] to establish a necessary condition on M . Aeron *et al.* [20] used Fano's inequality to form necessary conditions on both M and σ^2 . Then, they used the union bound to obtain sufficient conditions on both M and σ^2 for their sub-optimal decoder. Akcakaya and Tarokh [21] used the union and the large deviation bounds based on empirical entropies to get sufficient conditions on M for their joint typical decoder. They used the converse of the channel coding theorem to get necessary conditions on M . Scarlett *et al.* [22] extended this decoder [21] with the assumption that the distribution of the support set is provided. For a uniform distribution case, their necessary and sufficient conditions are equivalent to those of [21]. However, they are better for a non-uniform distribution case. Scarlett and Cevher [23] linked the support set reconstruction with the problem of coding over a mixed channel, where information spectrum methods were used to obtain necessary and sufficient conditions on M .

B. Information-Theoretic Works for CS With MMV

CS has many applications in wireless sensor networks (WSNs) [8]–[11] and MIMO radars [13]. In these

Manuscript received April 1, 2016; revised November 14, 2016; accepted January 17, 2017. Date of publication May 15, 2017; date of current version August 16, 2017. This work was supported by the National Research Foundation of Korea through the South Korean Government under Grant NRF-2015R1A2A1A05001826. This paper was presented at the 2012 International Symposium on Information Theory.

The authors are with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea (e-mail: sjpark1@gist.ac.kr; nyu@gist.ac.kr; heungno@gist.ac.kr).

Communicated by H. Rauhut, Associate Editor for Signal Processing.

Digital Object Identifier 10.1109/TIT.2017.2704111

0018-9448 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

applications, the signals of interest $\mathbf{x}^s \in \mathbb{R}^N$, $s = 1, 2, \dots, S$ are often modeled as *jointly K -sparse vectors*, implying that $\mathcal{I} = \mathcal{I}^1 = \mathcal{I}^2 \dots = \mathcal{I}^S$, where \mathcal{I}^s is the support set of \mathbf{x}^s and $|\mathcal{I}| = K$, which is referred to as a *joint sparsity structure*.

There are two models for sampling jointly K -sparse vectors. The first model is called *multiple measurement vectors (MMV) with the same sensing matrix* [25], in which they are sampled by the same sensing matrix. The second model is named as MMV with *different sensing matrices* [8], [9], in which each one is sampled by its own sensing matrix.

The authors of [26]–[28] have conducted information-theoretic research to obtain conditions under which the support set of both the models was reconstructed with a high probability. In noisy MMV with the same sensing matrix, Tang and Nehorai [26] used the hypothesis theory to obtain necessary and sufficient conditions on both the number of measurements M and the number of measurement vectors S , and proved that the success probability of the support set reconstruction increases with S , if $M = \Omega(K \log \frac{N}{K})$. Jin and Rao [27] exploited the communication theory to establish necessary and sufficient conditions on M and demonstrated the benefits of the joint sparsity structure based on their conditions. A detailed comparison between the results of our paper and those of [27] will be presented in Section IV. Finally, Duarte *et al.* [28] studied noiseless MMV with different sensing matrices, and formed necessary and sufficient conditions on M . However, it is difficult to apply the conditions to noisy MMV with different sensing matrices.

Meanwhile, works [8], [29], [30] have presented conditions of practical algorithms for a reliable support set reconstruction. In noiseless MMV with the same sensing matrix, Blanchard and Davies [30] obtained conditions for a reliable reconstruction from rank aware orthogonal matching pursuit (OMP). In noisy MMV with the same sensing matrix, Kim *et al.* [29] created compressive MUSIC, and presented its sufficient condition. In noiseless MMV with different sensing matrices, Baron *et al.* [8] produced trivial pursuit (TP) and distributed compressed sensing-simultaneous OMP (DCS-SOMP). By analyzing TP with the assumption that each sensing matrix contains i.i.d. Gaussian elements and that the nonzero values of each sparse vector are i.i.d. Gaussian variables, they demonstrated that with $M \geq 1$, TP reconstructs the support set as S is sufficiently large. They conjectured that $M \geq K + 1$ suffice for DCS-SOMP to reconstruct the support set as S is sufficiently large, based on its empirical results.

To the best of our knowledge, no information-theoretic study has been published to get necessary and sufficient conditions for reliable support set reconstruction in noisy MMV with different sensing matrices. Besides, these conditions have not been provided from the practical recovery algorithms for CS with noisy MMV with different sensing matrices.

C. Motivations of This Paper

CS with noisy MMV with different sensing matrices has been applied in many applications and the benefits facilitated by the joint sparsity structure have been empirically reported in [10] and [14]. In WSNs, Caione *et al.* [10] used the joint sparsity structure to reduce the number of

transmitted bits per sensor and reported that each sensor can reduce its transmission cost. In magnetic resonance imaging (MRI), Wu *et al.* [14] modeled multiple diffusion tensor images (DTIs) as jointly sparse vectors. They exploited the joint sparsity structure to reduce the number of samples per DTI, while retaining the reconstruction quality. Using the joint sparsity structure, they also empirically reported that the reconstruction quality of each DTI can be improved for a fixed number of samples per DTI.

To theoretically explain the above empirical benefits facilitated by the joint sparsity structure, theoretical tools are required to measure the performance of CS with noisy MMV with different sensing matrices. Such tools can be useful as guidelines for determining the system parameters in various CS applications with noisy MMV with different sensing matrices. For example, if the number of samples per DTI is fixed in the MRI [14], the theoretical tools may enable us to determine the number of DTIs required for achieving a given reconstruction quality. Thus, the first motivation of this paper is to provide the theoretical tools by establishing sufficient and necessary conditions for reliable support set reconstruction.

Next, for noiseless MMV with the same sensing matrix, let $\mathbf{Y}_A = \mathbf{F} \times [\mathbf{x}^1 \ \mathbf{x}^2 \ \dots \ \mathbf{x}^S] \in \mathbb{R}^{M \times S}$. Also, for noiseless MMV with different sensing matrices, let $\mathbf{Y}_B = [\mathbf{F}^1 \mathbf{x}^1 \ \mathbf{F}^2 \mathbf{x}^2 \ \dots \ \mathbf{F}^S \mathbf{x}^S] \in \mathbb{R}^{M \times S}$. Then, all the elements of \mathbf{Y}_B are uncorrelated because all the sensing matrices are independent. In contrast, those of \mathbf{Y}_A are correlated because they are taken from the same sensing matrix. Now, we consider a case where we set $S > K$ and $M > K$. Then, it is clear that $\text{rank}(\mathbf{Y}_B) = \min(S, M)$ with a high probability and $\text{rank}(\mathbf{Y}_A) \leq K$. Therefore, for this case, we conclude that $\text{rank}(\mathbf{Y}_B) > \text{rank}(\mathbf{Y}_A)$. This implies that a more reliable support set reconstruction can be expected in noiseless MMV with different sensing matrices for this case. Thus, the second motivation is to verify this perception in the presence of noise, by comparing our results with the existing ones in noisy MMV with the same sensing matrix [27].

D. Contributions of This Paper

The contributions of this paper are as follows: First, we derive upper and lower bounds of a failure probability of the support set reconstruction from Lemmas 1 and 2, by exploiting Fano's inequality [24] and the Chernoff bound [31]. These bounds are used for measuring the performance of CS with noisy MMV with different sensing matrices.

Next, we develop necessary and sufficient conditions for reliable support set reconstruction. Theorem 1 states that

$$M > K \left(1 + \frac{1}{Sf(\text{SNR}_{\min})} \right)$$

suffices to achieve reliable support set reconstruction in the *linear sparsity* regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = \beta \in (0, 1/2)$, and it also states that

$$M > K \left(1 + \frac{1}{Sf(\text{SNR}_{\min})} \log \frac{N}{K} \right)$$

suffices to achieve reliable support set reconstruction in the *sublinear sparsity* regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = 0$, where

$f(\text{SNR}_{\min})$ is an increasing function with respect to the minimum signal-to-noise ratio SNR_{\min} defined in (4). Next, for a finite S , N , K , and SNR_{\min} , Theorem 3 states that

$$M < \frac{2K \log \frac{N}{K} - 2 \log 2}{S \log(1 + K \times \text{SNR}_{\min})}$$

is necessary for reliable support set reconstruction. The necessary and sufficient conditions can be useful as guidelines to determine the system parameters of CS applications with noisy MMV with different sensing matrices. Corollaries 1 and 2 indicate that reliable support set reconstruction is possible as sufficiently many measurement vectors S for a fixed M are taken at a low SNR_{\min} . For a fixed N and K , Theorem 2 shows that $M \geq K + 1$ measurements suffice for reconstructing the support set, as S is sufficiently large. Then, for a fixed N , K , and $M = K + 1$, Corollary 3 provides a sufficient condition on S for reliable support set reconstruction. We provide theoretical explanations of the benefits of the joint sparsity structure, which conform with the empirical results of CS applications with noisy MMV with different sensing matrices [10], [14]. Finally, we compare the sufficient condition (11) with the known one (26) for noisy MMV with the same sensing matrix [27]. Therefore, we demonstrate that if $S \geq K$, noisy MMV with different sensing matrices may require fewer measurements M for reliable support set reconstruction than noisy MMV with the same sensing matrix under a low noise-level scenario. It confirms the superiority of MMV with different sensing matrices.

II. NOTATIONS, SYSTEM MODEL & PROBLEM FORMULATION

A. Notations

The following notations will be used in the whole paper.

1. \mathbb{P} , \mathbb{E} and \mathbb{V} denote the probability, expectation and (co)variance, respectively.
2. A small (capital) bold letter \mathbf{f} (\mathbf{F}) is a vector (matrix).
3. A sub-vector (sub-matrix) formed by the elements (columns) of \mathbf{f} (\mathbf{F}) indexed by a set \mathcal{I} is denoted by $\mathbf{f}_{\mathcal{I}}$ ($\mathbf{F}_{\mathcal{I}}$).
4. For a given matrix \mathbf{F} , its inversion, transpose, trace and the i th eigenvalue are denoted by \mathbf{F}^{-1} , \mathbf{F}^T , $\text{tr}[\mathbf{F}]$ and $\lambda_i(\mathbf{F})$, respectively. Also, its orthogonal projection matrix is defined by

$$\mathbf{Q}(\mathbf{F}) := \mathbf{I}_M - \mathbf{F}(\mathbf{F}^T \mathbf{F})^{-1} \mathbf{F}^T \quad (2)$$

where $\mathbf{Q}(\mathbf{F})$ maps an arbitrary vector to the space orthogonal onto the space spanned by the columns of \mathbf{F} .

5. For given sets \mathcal{I} and \mathcal{J} , the relative complements of \mathcal{J} in \mathcal{I} is denoted as $\mathcal{J} \setminus \mathcal{I}$. The cardinality of a set \mathcal{I} is denoted by $|\mathcal{I}|$.
6. For a given function $f(x)$, its n th derivation with respect to x is denoted by $f^n(x)$.
7. The *linear sparsity regime* is defined by $\lim_{N \rightarrow \infty} \frac{K}{N} = \beta \in (0, 1/2)$.
8. The *sublinear sparsity regime* is defined by $\lim_{N \rightarrow \infty} \frac{K}{N} = 0$.

9. The expression $f(x) = \Omega(g(x))$ denotes $|f(x)| \geq c|g(x)|$ as $x \rightarrow \infty$ for a constant $c > 0$.

B. System Model

Let $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^S$ be jointly K -sparse vectors with a support set \mathcal{I} that belongs to

$$\mathcal{S} := \{\mathcal{H} | \mathcal{H} \subset \{1, 2, \dots, N\}, |\mathcal{H}| = K\}.$$

Thus, the number of nonzero coefficients of each sparse vector is K , the indices of the nonzero coefficients of all the sparse vectors are the same and the indices belong to the support set.

In noisy MMV with different sensing matrices, each sparse vector is sampled by its own sensing matrix, i.e.,

$$\mathbf{y}^s = \mathbf{F}^s \mathbf{x}^s + \mathbf{n}^s \quad s = 1, 2, \dots, S \quad (3)$$

where all the sensing matrices have i.i.d. Gaussian elements with a zero mean and a unit variance, and all the noise vectors have i.i.d. Gaussian elements with a zero mean and a σ^2 variance. We assume that all the noise vectors and all the sensing matrices are mutually independent. Then, we let x_{\min} be the smallest nonzero magnitude of all the sparse vectors and SNR_{\min} be the minimum signal-to-noise ratio given by

$$\text{SNR}_{\min} := x_{\min}^2 / \sigma^2. \quad (4)$$

C. Problem Formulation

We extend Akcakaya and Tarokh [21]'s decoder for noisy MMV with different sensing matrices. It takes all the measurement vectors as its input and yields a support set decision as its output

$$d : \{\forall_s (\mathbf{y}^s, \mathbf{F}^s)\} \mapsto \hat{\mathcal{I}} \in \mathcal{S}, \quad s = 1, 2, \dots, S.$$

Its decision rules are given in Definition 1.

Definition 1: All the measurement vectors $\{\mathbf{y}^1, \mathbf{y}^2, \dots, \mathbf{y}^S\}$ and a set $\mathcal{J} \in \mathcal{S}$ are δ jointly typical if the rank of $\mathbf{F}_{\mathcal{J}}^s$, $s = 1, \dots, S$, is K and

$$\left| \left(\sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{J}}^s) \mathbf{y}^s\|_2^2 \right) - S(M - K)\sigma^2 \right| < SM\delta. \quad (5)$$

As each sensing matrix contains i.i.d. Gaussian elements, the rank of each $\mathbf{F}_{\mathcal{J}}^s$, $s = 1, \dots, S$, is K with a high probability. The decision rule is to find sets that satisfy (5) for all the given measurement vectors and $\delta > 0$. In the entire paper, the support set is denoted by \mathcal{I} and any incorrect support set is denoted by \mathcal{J} , where their cardinalities are K , i.e., $|\mathcal{I}| = |\mathcal{J}| = K$.

We define the failure events, wherein the joint typical decoder fails to reconstruct the correct support set. First,

$$\mathcal{E}_{\mathcal{I}}^c := \left\{ \left| \left(\sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{I}}^s) \mathbf{y}^s\|_2^2 \right) - S(M - K)\sigma^2 \right| \geq SM\delta \right\} \quad (6)$$

implies that the correct support set is not δ jointly typical with all the measurement vectors. Next, for any $\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}$,

$$\mathcal{E}_{\mathcal{J}} := \left\{ \left| \left(\sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{J}}^s) \mathbf{y}^s\|_2^2 \right) - S(M - K)\sigma^2 \right| < SM\delta \right\} \quad (7)$$

implies that an incorrect support set is δ jointly typical with all the measurement vectors. Based on these failure events, we define a failure probability and give its upper bound as follows:

$$\begin{aligned} p_{err} &:= \mathbb{P} \left\{ \hat{\mathcal{I}} \neq \mathcal{I} \mid \mathbf{x}^1, \dots, \mathbf{x}^S \right\} \\ &= \mathbb{P} \left\{ \mathcal{E}_{\mathcal{I}}^c \cup \bigcup_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \mathcal{E}_{\mathcal{J}} \right\} \\ &\leq \mathbb{P} \left\{ \mathcal{E}_{\mathcal{I}}^c \right\} + \sum_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \mathbb{P} \left\{ \mathcal{E}_{\mathcal{J}} \right\} \end{aligned} \quad (8)$$

where $\mathbb{P} \left\{ \mathcal{E}_{\mathcal{I}}^c \right\}$ is taken with respect to all the noise vectors and $\mathbb{P} \left\{ \mathcal{E}_{\mathcal{J}} \right\}$ is taken with respect to all the noise vectors and all the sensing matrices. We establish Lemmas 1 and 2 given in Appendix A to give upper bounds of the probabilities of the failure events. Combining these lemmas with (8) yields

$$\begin{aligned} p_{err} &\leq \mathbb{P} \left\{ \mathcal{E}_{\mathcal{I}}^c \right\} + \sum_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \mathbb{P} \left\{ \mathcal{E}_{\mathcal{J}} \right\} \\ &\leq 2p(d_1) + \binom{N}{K} p(d_{2,\alpha^*} - 1) \end{aligned}$$

where p is defined in (31), $d_1 = \frac{M\delta}{(M-K)\sigma^2}$, $d_{2,\alpha^*} = \frac{(M-K)\sigma^2 + M\delta}{(M-K)\alpha^*}$, and $\alpha^* = \sigma^2 + x_{\min}^2$.

It is of interest to examine why $\mathbb{P} \left\{ \mathcal{E}_{\mathcal{I}}^c \right\}$ depends only on the noise vectors. As shown in Lemma 3, the random variable to define the event $\mathcal{E}_{\mathcal{I}}^c$ in (6) is $\sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{I}}^s) \mathbf{y}^s\|_2^2 / \sigma^2$, where the measurement vector in (3) consists of the two parts: the noise part \mathbf{n}^s and the signal part $\mathbf{F}_{\mathcal{I}}^s \mathbf{x}_{\mathcal{I}}^s$. The signal part belongs to the space spanned by the columns of $\mathbf{F}_{\mathcal{I}}^s$. Then, as specified in (2), the orthogonal projection matrix $\mathbf{Q}(\mathbf{F}_{\mathcal{I}}^s)$ maps the measurement vector to the space orthogonal onto the space spanned by the columns of $\mathbf{F}_{\mathcal{I}}^s$. Thus, the random variable is a function of the noise vectors only.

III. MAIN RESULTS

As the main contribution of this paper, this section presents sufficient and necessary conditions on M for reliable support set reconstruction, i.e., p_{err} converges to zero, in noisy MMV with different sensing matrices. We then interpret the conditions to demonstrate the benefits facilitated by the joint sparsity structure.

A. Sufficient Conditions on M

In [18] and [21], the authors have shown that fewer measurements M for a reliable support set reconstruction are required for noisy SMV in the linear sparsity regime, compared to the sublinear sparsity regime. Based on the results of [18] and [21], we are motivated to examine if the same result can be observed in noisy MMV with different sensing matrices.

Theorem 1: For any $\rho > 1$, we let $\delta = \rho^{-1} (1 - K/M) x_{\min}^2$. If the number of measurements satisfies

$$M > K + v_1 \frac{K}{S} \quad (9)$$

then the failure probability p_{err} defined in (8) converges to zero in the linear sparsity regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = \beta \in (0, 1/2)$, where

$$v_1 = - \frac{2(1 - \log \beta)}{\log \left(1 - \frac{1 - \rho^{-1}}{1 + \text{SNR}_{\min}^{-1}} \right) + \frac{1 - \rho^{-1}}{1 + \text{SNR}_{\min}^{-1}}} > 0. \quad (10)$$

Also, under the same conditions on ρ and δ , if the number of measurements satisfies

$$M > K + v_2 \frac{K}{S} \log \frac{N}{K} \quad (11)$$

then the failure probability p_{err} defined in (8) converges to zero in the sublinear sparsity regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = 0$, where

$$v_2 = - \frac{2}{\log \left(1 - \frac{1 - \rho^{-1}}{1 + \text{SNR}_{\min}^{-1}} \right) + \frac{1 - \rho^{-1}}{1 + \text{SNR}_{\min}^{-1}}} > 0. \quad (12)$$

Proof: The proof is given in Appendix C.

In terms of N , K , and S , the asymptotic order of the sufficient condition on M for the linear sparsity regime is $\Omega \left(K + \frac{K}{S} \right)$, whereas the order for the sublinear sparsity regime is $\Omega \left(\frac{K}{S} \log \frac{N}{K} \right)$. It confirms that fewer measurements are required in the linear sparsity regime, compared to the sublinear sparsity regime. Next, from the sufficient conditions, we observe an inverse relationship between M and S , owing to the joint sparsity structure. This relationship implies that taking more measurement vectors S reduces the number of required measurements M for reliable support set reconstruction. Then, the relationship can be used for explaining the empirical results of Caione *et al.* [10] and Wu *et al.* [14]. In [10], the authors have reported that the number of transmitted bits per sensor could be inversely reduced by the number of sensors, which implies that the transmission cost of each sensor could be saved. The result can be confirmed by our inverse relationship by considering S and M as the number of sensors and the number of transmitted bits per sensor, respectively. In [14], S and M are considered as the number of DTIs and the number of samples of each DTI, respectively. Again, it has been observed from [14] that the joint sparsity structure enabled the number of samples of each DTI to be inversely reduced by the number of DTIs, reducing the acquisition time for each DTI. These results can be confirmed by our inverse relationship.

Theorem 2: For any $\rho > 1$, we let $\delta = \rho^{-1} (1 - K/M) x_{\min}^2$, N and K be fixed. If the number of measurements satisfies $M \geq K + 1$, the failure probability p_{err} defined in (8) converges to zero as the number of measurement vectors is increased to the infinity.

Proof: The proof is given in Appendix C.

Theorem 2 suggests that with $M \geq K + 1$, reliable support set reconstruction for noisy MMV with different sensing matrices is possible when a large number of measurement vectors is available. The sufficient conditions in Theorem 1, i.e., (9) and (11) have SNR_{\min} values as shown in (10) and (12). They disappear in the sufficient condition of Theorem 2, i.e., $M \geq K + 1$. The support set reconstruction problem becomes

robust against noise when the number of measurement vectors is large.

B. Discussions on the Sufficient Conditions

We now examine the effect of SNR_{\min} on the sufficient conditions of Theorem 1. The aim is to determine the relationship among S , M and SNR_{\min} for reliable support set reconstruction.

Corollary 1: For any $\rho > 1$, we let $\delta = \rho^{-1}(1 - K/M)x_{\min}^2$. The sufficient conditions of Theorem 1 are rewritten as

$$M > K + \left(\frac{\sqrt[3]{S} + (\sqrt{S} \times \text{SNR}_{\min})^{-1}}{1 - \rho^{-1}} \right)^2 4K \log \frac{N}{K} \quad (13)$$

in the sublinear sparsity regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = 0$, and

$$M > K + \left(\frac{\sqrt[3]{S} + (\sqrt{S} \times \text{SNR}_{\min})^{-1}}{1 - \rho^{-1}} \right)^2 4K(1 - \log \beta) \quad (14)$$

in the linear sparsity regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = \beta \in (0, 1/2)$.

Proof: The proof is given in Appendix D.

Corollary 1 suggests that for a fixed M , reliable support set reconstruction is possible as the number of measurement vectors S is increased to infinity, although SNR_{\min} is low. Namely, we observe a noise reduction effect, which shows that using the joint sparsity structure leads to an increase in SNR_{\min} or a decrease in σ^2 by the square root of S . This effect can explain the improvement in the reconstruction quality of the DTIs, as empirically reported in [14].

We then improve our noise reduction effect by considering that SNR_{\min} is larger than a certain value.

Corollary 2: For any $\rho > 3$, we let $\delta = \rho^{-1}(1 - K/M)x_{\min}^2$ and $\alpha = 2/3$. If

$$\text{SNR}_{\min} \geq \frac{\alpha}{1 - \rho^{-1} - \alpha} = \frac{2\rho}{\rho - 3}, \quad (15)$$

the sufficient conditions of Theorem 1 are rewritten as

$$M > K + \frac{S^{-1} + (S \times \text{SNR}_{\min})^{-1}}{1 - \rho^{-1}} 4K \log \frac{N}{K} \quad (16)$$

in the sublinear sparsity regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = 0$, and

$$M > K + \frac{S^{-1} + (S \times \text{SNR}_{\min})^{-1}}{1 - \rho^{-1}} 4K(1 - \log \beta) \quad (17)$$

in the linear sparsity regime, i.e., $\lim_{N \rightarrow \infty} \frac{K}{N} = \beta \in (0, 1/2)$.

Proof: The proof is given in Appendix D.

First of all, Corollary 2 requires $\rho > 3$ to ensure that the lower bound in (15) is positive. A simple computation

shows that Corollary 2 requires fewer measurements in both the regimes compared to Corollary 1 because

$$\begin{aligned} \left(\frac{\sqrt[3]{S} + (\sqrt{S} \times \text{SNR}_{\min})^{-1}}{1 - \rho^{-1}} \right)^2 &= S^{-1} \left(\frac{1 + \text{SNR}_{\min}^{-1}}{1 - \rho^{-1}} \right)^2 \\ &\geq S^{-1} \left(\frac{1 + \text{SNR}_{\min}^{-1}}{1 - \rho^{-1}} \right) \\ &= \frac{S^{-1} + (S \times \text{SNR}_{\min})^{-1}}{1 - \rho^{-1}} \end{aligned}$$

where the second inequality is owing to $\frac{1 + \text{SNR}_{\min}^{-1}}{1 - \rho^{-1}} = \frac{1}{t} > 1$ for any $\rho > 3$ and t defined in (61). Besides, Corollary 2 improves the noise reduction effect observed in Corollary 1 by showing that SNR_{\min} is increased by S for the region of SNR_{\min} in (15).

Theorem 2 suggests, it is to be noted, that $M = K + 1$ is sufficient for reliable support set reconstruction if S is sufficiently large with a fixed N and K . Then, it would be interesting to determine how large S should be required for achieving the minimum number of measurements at each sensor, i.e., $M = K + 1$. In wireless sensor networks [34], energy sources used in sensors are very limited due to limitation of sensor sizes. Thus, minimizing the energy used for transmission of data at each sensor which often leads to extending the lifetime of the sensor battery is a value of importance. This point is noted in Caione *et al.* [10] as an advantage of using distributed compressed sensing on joint sparse model-2 signal ensembles (see Section V there). Corollary 3 which aims to provide a sufficient condition on S for achieving $M = K + 1$ thus is motivated.

Corollary 3: Let N and K be fixed and finite. For any $\rho > 1$, we let $\delta = \rho^{-1}(K + 1)^{-1}x_{\min}^2$ and $M = K + 1$. If the number of measurement vectors satisfies

$$S > \underbrace{\left(\log \left(\binom{N}{K} + 2 \right) - \log \varepsilon \right)}_{:=S^*} \times \max \left[\left| \frac{1}{\log \mu_{\mathcal{I}}} \right|, \left| \frac{1}{\log \mu_{\mathcal{J}}} \right| \right] \quad (18)$$

reliable support set reconstruction is possible, i.e., $p_{err} < \varepsilon$ for sufficiently small $\varepsilon \in (0, 1)$, where $\log \mu_{\mathcal{I}}$ and $\log \mu_{\mathcal{J}}$ are defined in (63) and (65), respectively. The sufficient condition on S is decreasing with respect to SNR_{\min} .

Proof: The proof is given in Appendix D.

To the best of our knowledge, the sufficient conditions on S for a reliable support set reconstruction have not yet been developed. A similar result has been reported by Tang and Nehorai [26], in which they reported that $M = \Omega(K \log \frac{N}{K})$ and $S = \frac{\log N}{\log \log N}$ suffice for a reliable support set reconstruction in noisy MMV with the same sensing matrix, as N is sufficiently large.

It is of interest to examine whether the sufficient condition S^* in (18) is good. For this, we implement the joint typical decoder in (5) and conduct experiments for different values of SNR_{\min} and K , for a fixed $N = 50$. We count the number of failure occurrences, wherein the joint typical

decoder fails to reconstruct the support set. We obtain the smallest S^{emp} such that the ratio of the failure occurrences is smaller than $\varepsilon = 0.01$. By comparing S^{emp} with S^* in (18), we see that S^* approaches S^{emp} , as SNR_{\min} is sufficiently large. For example, we see that $S^{emp} = 8$ and $S^* = 12$ at $\text{SNR}_{\min} = 20$ [dB], $K = 2$, and $S^{emp} = 5$ and $S^* = 6$ at $\text{SNR}_{\min} = 30$ [dB], $K = 2$. A similar trend is observed with a bigger K , i.e., at $K = 5$. For example, we see that $S^{emp} = 12$ and $S^* = 19$ at $\text{SNR}_{\min} = 20$ [dB], and $S^{emp} = 7$ and $S^* = 10$ at $\text{SNR}_{\min} = 30$ [dB].

Fletcher *et al.* [19] have reported that the ML decoder requires $M = K + 1$ measurements for a reliable support set reconstruction in noisy SMV, when the signal-to-noise ratio is sufficiently large. This result can be observed from Corollary 3. Specifically, we assume that SNR_{\min} is sufficiently large for a fixed N and K . Then, from (63) and (65), it is easy to see that

$$\begin{aligned} \lim_{\text{SNR}_{\min} \rightarrow \infty} \log \mu_{\mathcal{I}} &= -\infty, \\ \lim_{\text{SNR}_{\min} \rightarrow \infty} \log \mu_{\mathcal{J}} &= 2^{-1} (1 - \rho^{-1} - \log \rho). \end{aligned}$$

Hence, (18) is simplified to

$$S > \left(\log \left(\binom{N}{K} + 2 \right) - \log \varepsilon \right) \times \left| 2 \left(1 - \rho^{-1} - \log \rho \right)^{-1} \right|. \quad (19)$$

Note that N , K , and ε are fixed. Thus, for a large ρ , we have

$$\left| 1 - \rho^{-1} - \log \rho \right| \gg 2 \left(\log \left(\binom{N}{K} + 2 \right) - \log \varepsilon \right), \quad (20)$$

which leads to $S \geq 1$. This result suggests that the joint typical decoder requires $M = K + 1$ measurements for reliable support set reconstruction in noisy SMV, whenever SNR_{\min} is sufficiently large and ρ satisfies (20).

C. Necessary Condition on M

We specify a necessary condition that must be satisfied by a decoder for reliable support set reconstruction in noisy MMV with different sensing matrices. Unlike the sufficient conditions of Theorem 1, the necessary condition is presented for a finite N and K .

We begin by transforming (3) into

$$\underbrace{\begin{bmatrix} \mathbf{y}^1 \\ \vdots \\ \mathbf{y}^S \end{bmatrix}}_{=: \mathbf{y} \in \mathbb{R}^{SM}} = \underbrace{\begin{bmatrix} \mathbf{F}^1 & & \\ & \ddots & \\ & & \mathbf{F}^S \end{bmatrix}}_{=: \tilde{\mathbf{F}} \in \mathbb{R}^{SM \times SN}} \underbrace{\begin{bmatrix} \mathbf{x}^1 \\ \vdots \\ \mathbf{x}^S \end{bmatrix}}_{=: \mathbf{x} \in \mathbb{R}^{SN}} + \underbrace{\begin{bmatrix} \mathbf{n}^1 \\ \vdots \\ \mathbf{n}^S \end{bmatrix}}_{=: \mathbf{n} \in \mathbb{R}^{SM}} \quad (21)$$

where \mathbf{x} is an SK -sparse vector belonging to an infinite set

$$\mathcal{X}_{x_{\min}} := \left\{ \mathbf{x} \in \mathbb{R}^{SN} \mid |x(i)| \geq x_{\min}, \forall i \in \mathcal{I}, |\mathcal{I}| = SK \right\}$$

where $x(i)$ is the i th element of \mathbf{x} and \mathcal{I} is the support set of \mathbf{x} . Owing to the joint sparsity structure, the number of possible support sets is $\binom{N}{K}$. Then, we define a failure probability as:

$$p_{err} := \mathbb{E}_{\tilde{\mathbf{F}}} \sup_{\mathbf{x} \in \mathcal{X}_{x_{\min}}} \mathbb{P} \left\{ \hat{\mathcal{I}} \neq \mathcal{I} \mid \mathbf{x}, \tilde{\mathbf{F}} \right\} \quad (22)$$

where $\hat{\mathcal{I}}$ is an estimate of the support set based on \mathbf{y} and $\tilde{\mathbf{F}}$ in (21). Then, Lemma III-3 of [20] yields

$$\sup_{\mathbf{x} \in \mathcal{X}_{x_{\min}}} \mathbb{P} \left\{ \hat{\mathcal{I}} \neq \mathcal{I} \mid \mathbf{x}, \tilde{\mathbf{F}} \right\} \geq \min_{\hat{\mathbf{x}} \in \mathcal{X}_{\{x_{\min}\}}} \max_{\mathbf{x} \in \mathcal{X}_{\{x_{\min}\}}} \mathbb{P} \left\{ \hat{\mathbf{x}} \neq \mathbf{x} \mid \mathbf{x}, \tilde{\mathbf{F}} \right\} \quad (23)$$

where $\hat{\mathbf{x}}$ is an estimate for \mathbf{x} based on \mathbf{y} and $\tilde{\mathbf{F}}$ in (21) and

$$\mathcal{X}_{\{x_{\min}\}} := \left\{ \mathbf{x} \in \mathbb{R}^{SN} \mid x(i) = x_{\min}, \forall i \in \mathcal{I}, |\mathcal{I}| = SK \right\}$$

which is a finite set. Assume that \mathbf{x} is uniformly distributed over this finite set. Applying Fano's inequality [24] to (23) yields

$$\begin{aligned} \max_{\mathbf{x} \in \mathcal{X}_{\{x_{\min}\}}} \mathbb{P} \left\{ \hat{\mathbf{x}} \neq \mathbf{x} \mid \mathbf{x}, \tilde{\mathbf{F}} \right\} &\geq \mathbb{P} \left\{ \hat{\mathbf{x}} \neq \mathbf{x} \mid \tilde{\mathbf{F}} \right\} \\ &\geq 1 - \frac{\mathbb{I}(\mathbf{x}; \mathbf{y} \mid \tilde{\mathbf{F}}) + \log 2}{\log (|\mathcal{X}_{\{x_{\min}\}}| - 1)} \end{aligned} \quad (24)$$

where \mathbf{x} and $\hat{\mathbf{x}}$ belong to the finite set $\mathcal{X}_{\{x_{\min}\}}$ and $\mathbb{I}(\mathbf{x}; \mathbf{y})$ is the mutual information between \mathbf{x} and \mathbf{y} . We get a necessary condition on M to ensure that the lower bound in (24) is bounded away from zero, as follows:

Theorem 3: Let N and K are fixed and finite. In (21), if the number of measurements satisfies

$$M < \frac{2K \log \frac{N}{K} - 2 \log 2}{S \log (1 + K \times \text{SNR}_{\min})} \quad (25)$$

then the failure probability p_{err} defined in (22) is bounded away from zero.

Proof: The proof is given in Appendix C.

IV. RELATIONS TO THE EXISTING INFORMATION-THEORETIC RESULTS

A. Relations to Noisy MMV With the Same Sensing Matrix [27]

Jin and Rao [27] have exploited the Chernoff bound to obtain a tight sufficient condition on M for a reliable support set reconstruction for noisy MMV with the same sensing matrix in the sublinear sparsity regime. Owing to the complicated form of their sufficient condition, they could not clearly show the benefits facilitated by the joint sparsity structure. Thus, they simplified their condition under scenarios such as: *i)* a low noise-level scenario and *ii)* a scenario with S identical sparse vectors. In Table I, we summarize our sufficient conditions on M , and compare them to that of [27] under the low noise-level scenario in the sublinear sparsity regime.

First, in a low noise-level scenario, as shown in Table I, the sufficient condition [27] for noisy MMV with the same sensing matrix is

$$M = \Omega \left(\frac{K \log N}{\min(K, S)} \right). \quad (26)$$

If $S < K$, the sufficient conditions (11) and (26) have the same order, implying that there is no significant performance gap in the support set reconstruction between the models. However, if $S > K$, (26) is $M = \Omega(\log N)$, whereas (11) is

TABLE I
SUFFICIENT CONDITIONS ON M FOR SUPPORT SET RECONSTRUCTION

	This paper	Yuzhe and Rao [27]
Linear sparsity regime $\lim_{N \rightarrow \infty} \frac{K}{N} = \beta \in (0, 1/2)$	$M = \Omega(K + \frac{K}{S})$	Not presented
Sublinear sparsity regime $\lim_{N \rightarrow \infty} \frac{K}{N} = 0$	$M = \Omega(\frac{K}{S} \log \frac{N}{K})$	$M = \Omega(\frac{K \log N}{\min(K, S)})$
N and K are finite ($\text{SNR}_{\min} \rightarrow \infty$ or $S \rightarrow \infty$)	$M \geq K + 1$	Not presented

$M = \Omega(\frac{K}{S} \log N)$. It implies that noisy MMV with different sensing matrices is superior to noisy MMV with the same sensing matrix or $S > K$, with respect to M for reliable support set reconstruction. The result of this comparison supports the perception presented in Section I-C, wherein a more reliable support set reconstruction could be expected in a noiseless MMV with different sensing matrices owing to the linear independency of the measurement vectors. Moreover, it validates the perception, even in the presence of noise.

Second, we consider a scenario with S identical sparse vectors. Then, the sufficient condition of [27] is

$$M = \Omega\left(\frac{K \log N}{\log(1 + S \|\mathbf{x}\|_2^2 / \sigma^2)}\right). \quad (27)$$

From (27), we observe that σ^2 is reduced by a factor of S . However, the noise reduction effect for noisy MMV with the same sensing matrix requires a restriction, where all the sparse vectors should be identical, which can be hardly achieved in practice. In contrast, the noise reduction effect for noisy MMV with different sensing matrices does not require this restriction, as shown in Corollaries 1 and 2.

B. Relations to Noisy SMV [21]

Akçakaya and Tarokh [21] have used the joint typical decoder to establish the sufficient conditions on M for a reliable support set reconstruction in noisy SMV. They exploited the exponential inequalities [32] to obtain the upper bounds on the sum of the weighted chi-square random variables. In this subsection, we demonstrate that the approaches developed in this paper are superior to the use of the exponential inequalities. Thus, we use the exponential inequalities to generalize their bounds for noisy MMV with different sensing matrices. We give Propositions 1 and 2 to prove that the generalized bounds are worse than the bounds of Lemmas 1 and 2.

Proposition 1: For any positive δ , we have

$$\mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} \leq 2p(d_1) \leq 2p_{1,\text{exp}}$$

where both $p(d_1)$ and d_1 are given in Lemma 1, and

$$p_{1,\text{exp}} := \exp\left(-\frac{S\delta^2}{4\sigma^4} \frac{M^2}{M - K + 2\delta M / \sigma^2}\right). \quad (28)$$

Proof: The proof is given in Appendix E.

Proposition 2: For any $\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}$ and any $\delta > 0$ such that

$$0 < \delta < (1 - K/M) x_{\min, \mathcal{J}}^2, \quad (29)$$

we have

$$\mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} \leq p(d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1) \leq p_{2, \mathcal{J}, \text{exp}}$$

where both $p(d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1)$ and $d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})}$ are given in Lemma 2 and

$$p_{2, \mathcal{J}, \text{exp}} := \exp\left(-\frac{S^2(M - K)}{4 \sum_{s=1}^S \alpha_{\mathcal{J}, s}^2} \left(x_{\min, \mathcal{J}}^2 - \frac{M\delta}{M - K}\right)^2\right) \quad (30)$$

and $\alpha_{\mathcal{J}, s}$ is defined in (39) and $x_{\min, \mathcal{J}}^2$ is defined in (43).

Proof: The proof is given in Appendix E.

If $S = 1$, we can see that $p_{1,\text{exp}}$ and $p_{2, \mathcal{J}, \text{exp}}$ are equivalent to the bounds of Akçakaya and Tarokh [21]. Propositions 1 and 2 state that the bounds on the failure probability of Lemmas 1 and 2 are tighter than the bounds of [21] for noisy SMV.

V. CONCLUSIONS

We have studied a support set reconstruction problem for CS with noisy MMV with different sensing matrices. The union and Chernoff bounds have been used to obtain the upper bound of the failure probability of the support set reconstruction, and Fano's inequality has been used to obtain the lower bound of this failure probability. As we have obtained the upper bound by analyzing an exhaustive search decoder, the bound is used to measure the performance of CS with noisy MMV with different sensing matrices. We have then developed the necessary and sufficient conditions in terms of the sparsity K , the ambient dimension N , the number of measurements M , the number of measurement vectors S , and the minimum signal-to-noise ratio SNR_{\min} . They can be useful as guidelines to determining the system parameters in various CS applications with noisy MMV with different sensing matrices.

The conditions are interpreted to provide theoretical explanations for the benefits facilitated by the joint sparsity structure in noisy MMV with different sensing matrices:

- i. From the sufficient conditions of Theorem 1, we have observed an inverse relationship between M and S . Due to the inverse relation, we take fewer measurements M per each measurement vector for reliable support set reconstruction by taking more measurement vectors S .
- ii. From the sufficient conditions of Corollaries 1 and 2, we have observed a noise reduction effect, which shows that the usage of the joint sparsity structure results in an increase in SNR_{\min} or a decrease in σ^2 by a factor of S . Therefore, the support set reconstruction can be robust against noise as the number of measurement vectors is increased to infinity.
- iii. From Theorem 2, we have shown that $M = K + 1$ is achieved for a fixed N and K , as S is sufficiently large. From Corollary 3, we have provided the sufficient condition on S to reconstruct the support set for a fixed N , K , and $M = K + 1$.

The theoretical explanations confirm the benefits of the joint sparsity structure, as empirically shown in CS applications with noisy MMV with different sensing matrices [10], [14].

We have compared our sufficient conditions for noisy MMV with different sensing matrices with the other existing results [27] for noisy MMV with the same sensing matrix. For a low-level noise scenario with $S \geq K$, we have shown that the number of measurements for reliable support set reconstruction for noisy MMV with different sensing matrices is lesser than that for noisy MMV with the same sensing matrix. Also, [27] has shown the noise reduction effect. This was achieved under a rather restricted condition though, i.e., all sparse vectors are the same. While such a restricted condition is not required in the noisy MMV problem with *different* sensing matrices studied in this paper, the noise reduction effect has also been observed, which is a significant improvement.

APPENDIX A LEMMAS 1 AND 2

This section presents Lemmas 1 and 2, which give upper bounds of the probabilities of the failure events defined in (6) and (7), respectively. Also, for simplicity, we define

$$p(x) = \exp\left(-\frac{S(M-K)}{2}x\right)(1+x)^{\frac{S(M-K)}{2}}. \quad (31)$$

Lemma 1: For any positive δ , we have

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} &\leq 2 \exp\left(-\frac{S(M-K)}{2}d_1\right)(1+d_1)^{\frac{S(M-K)}{2}} \\ &= 2p(d_1) \end{aligned} \quad (32)$$

where the function p is defined in (31), and

$$d_1 := \frac{M\delta}{(M-K)\sigma^2} > 0. \quad (33)$$

Proof: From (6), we have

$$\mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} = \mathbb{P}\{Z_{\mathcal{I}} \leq W_1\} + \mathbb{P}\{Z_{\mathcal{I}} \geq W_2\} \quad (34)$$

where $Z_{\mathcal{I}}$ is defined in Lemma 3, and

$$W_i = S(M-K) + (-1)^i SM\delta/\sigma^2, \quad i = 1, 2.$$

Applying the Chernoff bound [31] to (34) yields

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} &\leq \sum_{i=1}^2 \exp(-t_i W_i) \mathbb{E}[\exp(t_i Z_{\mathcal{I}})] \\ &= \sum_{i=1}^2 \underbrace{\exp(-t_i W_i) (1 - 2t_i)^{-S(M-K)/2}}_{=: f(t_i; W_i)} \end{aligned} \quad (35)$$

where the equality is from Lemma 3, $t_1 < 0$ and $t_2 \in (0, \frac{1}{2})$. As each $f(t_i; W_i)$ is convex, $t_i = t_i^*$ at $f^{(1)}(t_i; W_i) = 0$ yields the minimizer of $f(t_i; W_i)$, where

$$t_i^* = 2^{-1} \left(1 - W_i^{-1} S(M-K)\right), \quad i = 1, 2.$$

Thus, $f(t_i; W_i) \geq f(t_i^*; W_i)$ for each i . If $W_1 \leq 0$, it is clear that $\mathbb{P}\{Z_{\mathcal{I}} \leq W_1\} = 0$ because $Z_{\mathcal{I}}$ is quadratic. Thus,

$$\mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} = \mathbb{P}\{Z_{\mathcal{I}} \geq W_2\} \leq f(t_2^*; W_2) = p(d_1) \quad (36)$$

where $p(d_1)$ and d_1 are defined in (32) and (33), respectively. If $W_1 > 0$ then $f(t_1^*; W_1) \leq f(t_2^*; W_2)$ because

$$\begin{aligned} \log f(t_1^*; W_1) - \log f(t_2^*; W_2) \\ = S(M-K) [d_1 + 2 \log(1-d_1) - 2 \log(1+d_1)] < 0. \end{aligned}$$

Thus,

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} &= f(t_1^*; W_1) + f(t_2^*; W_2) \leq 2f(t_2^*; W_2) \\ &= 2 \exp\left(-\frac{S(M-K)}{2}d_1\right)(1+d_1)^{\frac{S(M-K)}{2}}. \end{aligned} \quad (37)$$

Finally, combining (36) and (37) leads to (32). \blacksquare

Lemma 2: Let $\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}$ and a matrix $\mathbf{R}_{\mathcal{J}}$ be

$$\mathbf{R}_{\mathcal{J}} = \begin{bmatrix} \alpha_{\mathcal{J},1} \mathbf{I}_{M-K} & & \\ & \ddots & \\ & & \alpha_{\mathcal{J},S} \mathbf{I}_{M-K} \end{bmatrix} \quad (38)$$

where

$$\alpha_{\mathcal{J},s} := \sigma^2 + \|\mathbf{x}_{\mathcal{I} \setminus \mathcal{J}}^s\|_2^2 > 0. \quad (39)$$

Consider any positive δ such that

$$0 < \delta < (1 - K/M) (\lambda_{\min}(\mathbf{R}_{\mathcal{J}}) - \sigma^2)$$

where $\lambda_{\min}(\mathbf{R}_{\mathcal{J}})$ is the smallest eigenvalue of $\mathbf{R}_{\mathcal{J}}$. Then,

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} &\leq \exp\left(-\frac{S(M-K)}{2} (d_{2,\lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1)\right) d_{2,\lambda_{\min}(\mathbf{R}_{\mathcal{J}})}^{\frac{S(M-K)}{2}} \\ &= p(d_{2,\lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1) \\ &\leq p(d_{2,\alpha^*} - 1) \end{aligned} \quad (40)$$

where the function p is defined in (31),

$$d_{2,\lambda_{\min}(\mathbf{R}_{\mathcal{J}})} := \frac{(M-K)\sigma^2 + M\delta}{(M-K)\lambda_{\min}(\mathbf{R}_{\mathcal{J}})} \in (0, 1), \quad (41)$$

$$\alpha^* := \sigma^2 + x_{\min}^2, \quad (42)$$

and

$$x_{\min}^2 = \min_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \min_{s \in \{1, 2, \dots, S\}} \underbrace{\|\mathbf{x}_{\mathcal{I} \setminus \mathcal{J}}^s\|_2^2}_{=: x_{\min, \mathcal{J}}^2}. \quad (43)$$

Proof: From (7), we have

$$\mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} = \mathbb{P}\{Z_{\mathcal{J}} < W_1\} - \mathbb{P}\{Z_{\mathcal{J}} < W_2\} \leq \mathbb{P}\{Z_{\mathcal{J}} < W_1\} \quad (44)$$

where $Z_{\mathcal{J}}$ is defined in Lemma 4, and

$$W_i = S(M-K)\sigma^2 - (-1)^i SM\delta, \quad i = 1, 2. \quad (45)$$

Applying the Chernoff bound [31] to (44) yields for $t < 0$,

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} &\leq \exp(-t W_1) \mathbb{E}[\exp(t Z_{\mathcal{J}})] \\ &= \exp(-t W_1) \prod_{i=1}^{S(M-K)} (1 - 2t \lambda_i(\mathbf{R}_{\mathcal{J}}))^{-1/2} \\ &\leq \exp(-t W_1) (1 - 2t \lambda_{\min}(\mathbf{R}_{\mathcal{J}}))^{-S(M-K)/2} \\ &=: f(t; W_1) \end{aligned} \quad (46)$$

where the equality is from Lemma 4 and the second inequality is due to that all the eigenvalues are positive. We then define a function $h(t) := \log f(t; W_1)$. Then,

$$h^{(2)}(t) = 2S(M-K) \lambda_{\min}^2(\mathbf{R}_{\mathcal{J}}) (1 - 2t \lambda_{\min}(\mathbf{R}_{\mathcal{J}}))^{-2} > 0$$

which implies that h is convex with respect to t . It leads to that f in (46) is logarithmically convex. Thus $t = t^*$ at $f^{(1)}(t; W_1) = 0$ yields the minimizer of $f(t; W_1)$ where

$$t^* = 2^{-1} \left(\lambda_{\min}^{-1}(\mathbf{R}_{\mathcal{J}}) - W_1^{-1} S(M - K) \right) < 0.$$

Substituting t^* in (46) yields

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} &\leq f(t^*; W_1) \\ &= \exp\left(-\frac{S(M-K)}{2} \left(d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1\right)\right) d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})}^{\frac{S(M-K)}{2}} \\ &= p(d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1) \end{aligned} \quad (47)$$

where $d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})}$ is defined in (41) and p is defined in (31).

Next, let $\beta = 2^{-1} S(M - K)$ and $x = d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})}$ in the upper bound (47). Then, we have $p(x - 1) = x^\beta \exp(-\beta(x - 1))$, where

$$\frac{\partial p(x - 1)}{\partial x} = \beta x^{\beta-1} \exp(-\beta(x - 1)) (x^{-1} - 1) > 0 \quad (48)$$

and

$$\frac{\partial x}{\partial \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} = -x < 0. \quad (49)$$

Due to (48) and (49),

$$\begin{aligned} \frac{\partial p(x - 1)}{\partial \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} &= \frac{\partial p(x - 1)}{\partial x} \frac{\partial x}{\partial \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} \\ &= -\beta x^{\beta-1} \exp(-\beta(x - 1)) (x^{-1} - 1) < 0 \end{aligned}$$

which shows that the upper bound in (47) is decreasing with respect to $\lambda_{\min}(\mathbf{R}_{\mathcal{J}})$. Then, remind that the matrix in (38) is the covariance matrix of a multivariate Gaussian vector \mathbf{b} in (58). Then for any incorrect support set, its smallest eigenvalue can be easily computed and lower bounded by

$$\lambda_{\min}(\mathbf{R}_{\mathcal{J}}) = \min_{s \in \{1, 2, \dots, S\}} \alpha_{\mathcal{J}, s} = \sigma^2 + x_{\min, \mathcal{J}}^2 \geq \alpha^* \quad (50)$$

where $x_{\min, \mathcal{J}}^2$ is defined in (43) and α^* is defined in (42). Thus, for any incorrect support set $\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}$, we conclude that

$$\mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} \leq p(d_{2, \lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1) \leq p(d_{2, \alpha^*} - 1)$$

which completes the proof. \blacksquare

APPENDIX B LEMMAS 3 AND 4

First of all, we give the Scharf's theorem [33] to compute the moment generating function of a quadratic random variable. We then make Lemmas 3 and 4 to give the moment generating functions of the random variables of $\mathcal{E}_{\mathcal{I}}^c$ and $\mathcal{E}_{\mathcal{J}}$ that were used in the proofs of Lemmas 1 and 2, respectively.

Scharf's Theorem [33, p. 64]: Let $\mathbf{b} \in \mathbb{R}^N$ be a multivariate Gaussian vector with a mean \mathbf{m} and a covariance \mathbf{R} . Then a random variable $Q \triangleq (\mathbf{b} - \mathbf{m})^T (\mathbf{b} - \mathbf{m})$ is quadratic with $\mathbb{E}[Q] = \text{tr}[\mathbf{R}]$, $\mathbb{V}[Q] = 2\text{tr}[\mathbf{R}^T \mathbf{R}]$ and for any t

$$\mathbb{E}[\exp(tQ)] = \prod_{i=1}^N (1 - 2t\lambda_i(\mathbf{R}))^{-1/2}.$$

Lemma 3: In (6), define a quadratic random variable

$$Z_{\mathcal{I}} := \sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{I}}^s) \mathbf{y}^s\|_2^2 / \sigma^2. \quad (51)$$

Then, $\mathbb{E}[Z_{\mathcal{I}}] = S(M - K)$, $\mathbb{V}[Z_{\mathcal{I}}] = 2S(M - K)$ and for any $0 < t < 0.5$,

$$\mathbb{E}[\exp(tZ_{\mathcal{I}})] = (1 - 2t)^{-S(M-K)/2}. \quad (52)$$

Proof: The orthogonal projection matrix is decomposed as

$$\mathbf{Q}(\mathbf{F}_{\mathcal{I}}^s) = \mathbf{U}_{\mathcal{I}}^s \mathbf{D}^s (\mathbf{U}_{\mathcal{I}}^s)^T$$

where \mathbf{D}^s is a diagonal matrix, whose first $M - K$ diagonals are ones and the remains are zeros, and $\mathbf{U}_{\mathcal{I}}^s$ is a unitary matrix. Then,

$$\begin{aligned} Z_{\mathcal{I}} &= \sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{I}}^s) \mathbf{y}^s\|_2^2 / \sigma^2 = \sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{I}}^s) \mathbf{n}^s\|_2^2 / \sigma^2 \\ &= \sum_{s=1}^S \left\| \mathbf{D}^s \underbrace{(\mathbf{U}_{\mathcal{I}}^s)^T \mathbf{n}^s / \sigma^2}_{=: \mathbf{w}^s} \right\|_2^2 = \sum_{s=1}^S \|\mathbf{D}^s \mathbf{w}^s\|_2^2 \end{aligned} \quad (53)$$

where \mathbf{w}^s is a multivariate Gaussian vector with mean $\mathbf{0}_M$ and covariance \mathbf{I}_M . Since the first $M - K$ diagonal elements of each diagonal matrix are ones, we have

$$\begin{aligned} Z_{\mathcal{I}} &= \sum_{s=1}^S \|\mathbf{D}^s \mathbf{w}^s\|_2^2 = \sum_{s=1}^S \sum_{i=1}^{M-K} |w^s(i)|^2 \\ &= \sum_{s=1}^S (\mathbf{w}_{\mathcal{P}}^s)^T \mathbf{w}_{\mathcal{P}}^s = \mathbf{w}^T \mathbf{w} \end{aligned} \quad (54)$$

which is quadratic, where

$$\mathbf{w}_{\mathcal{P}}^s = [w^s(1) \quad w^s(2) \quad \dots \quad w^s(M - K)]^T$$

and

$$\mathbf{w} = [(\mathbf{w}_{\mathcal{P}}^1)^T \quad (\mathbf{w}_{\mathcal{P}}^2)^T \quad \dots \quad (\mathbf{w}_{\mathcal{P}}^S)^T]^T. \quad (55)$$

In (53), \mathbf{w}^s is determined by $\mathbf{U}_{\mathcal{I}}^s$ and \mathbf{n}^s . Since the elements of $\mathbf{U}_{\mathcal{I}}^s$ and \mathbf{n}^s are independent, \mathbf{w}^i and \mathbf{w}^j are mutually independent for any $1 \leq i \neq j \leq S$. The covariance matrix of \mathbf{w} is an identity matrix. Thus, applying the Scharf's theorem to $Z_{\mathcal{I}}$ completes the proof. \blacksquare

Lemma 4: In (7), for any $\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}$, define a quadratic random variable

$$Z_{\mathcal{J}} := \sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{J}}^s) \mathbf{y}^s\|_2^2. \quad (56)$$

Then, $\mathbb{E}[Z_{\mathcal{J}}] = \text{tr}[\mathbf{R}_{\mathcal{J}}]$, $\mathbb{V}[Z_{\mathcal{J}}] = 2\text{tr}[\mathbf{R}_{\mathcal{J}}^T \mathbf{R}_{\mathcal{J}}]$ and for any t ,

$$\mathbb{E}[\exp(tZ_{\mathcal{J}})] = \prod_{i=1}^{S(M-K)} (1 - 2t\lambda_i(\mathbf{R}_{\mathcal{J}}))^{-1/2},$$

where $\mathbf{R}_{\mathcal{J}}$ is given in (38).

Proof: Similar to the proof of Lemma 3,

$$\mathbf{Q}(\mathbf{F}_{\mathcal{J}}^s) = \mathbf{U}_{\mathcal{J}}^s \mathbf{D}^s (\mathbf{U}_{\mathcal{J}}^s)^T$$

where \mathbf{D}^s is a diagonal matrix, whose first $M - K$ diagonals are ones and the remains are zeros, and $\mathbf{U}_{\mathcal{J}}^s$ is a unitary matrix. Then,

$$\begin{aligned} Z_{\mathcal{J}} &= \sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{J}}^s) \mathbf{y}^s\|_2^2 = \sum_{s=1}^S \|\mathbf{Q}(\mathbf{F}_{\mathcal{J}}^s) \mathbf{c}^s\|_2^2 \\ &= \sum_{s=1}^S \left\| \mathbf{D}^s \underbrace{(\mathbf{U}_{\mathcal{J}}^s)^T \mathbf{c}^s}_{=\mathbf{b}^s} \right\|_2^2 = \sum_{s=1}^S \|\mathbf{D}^s \mathbf{b}^s\|_2^2 \end{aligned} \quad (57)$$

where \mathbf{b}^s is a multivariate Gaussian vector with mean $\mathbf{0}_M$ and

$$\mathbb{V}[\mathbf{b}^s] = \left(\sigma^2 + \|\mathbf{x}_{\mathcal{I} \setminus \mathcal{J}}^s\|_2^2 \right) \mathbf{I}_M$$

and $\mathbf{c}^s = \mathbf{n}^s + \sum_{u \in \mathcal{I} \setminus \mathcal{J}} \mathbf{f}_u^s x^s(u)$. Since the first $M - K$ diagonal elements of each diagonal matrix are ones, we have

$$\begin{aligned} Z_{\mathcal{J}} &= \sum_{s=1}^S \|\mathbf{D}^s \mathbf{b}^s\|_2^2 = \sum_{s=1}^S \sum_{i=1}^{M-K} |b^s(i)|^2 \\ &= \sum_{s=1}^S (\mathbf{b}_{\mathcal{P}}^s)^T \mathbf{b}_{\mathcal{P}}^s = \mathbf{b}^T \mathbf{b} \end{aligned} \quad (58)$$

which is quadratic, where

$$\mathbf{b}_{\mathcal{P}}^s = [b^s(1) \quad b^s(2) \quad \dots \quad b^s(M-K)]^T$$

and

$$\mathbf{b} = [(\mathbf{b}_{\mathcal{P}}^1)^T \quad (\mathbf{b}_{\mathcal{P}}^2)^T \quad \dots \quad (\mathbf{b}_{\mathcal{P}}^S)^T]^T.$$

In (57), \mathbf{b}^s is determined by $\mathbf{U}_{\mathcal{J}}^s$, \mathbf{n}^s and $\{\mathbf{f}_u^s : u \in \mathcal{I} \setminus \mathcal{J}\}$. Since the elements of $\mathbf{U}_{\mathcal{J}}^s$, \mathbf{n}^s and $\{\mathbf{f}_u^s : u \in \mathcal{I} \setminus \mathcal{J}\}$ are independent, \mathbf{b}^i and \mathbf{b}^j are mutually independent for any $1 \leq i \neq j \leq S$. The covariance matrix of \mathbf{b} is diagonal as shown in (38). Thus, applying the Scharf's theorem to $Z_{\mathcal{J}}$ completes the proof. ■

APPENDIX C

PROOFS OF THEOREMS 1, 2 AND 3

A. Proof of Theorem 1

It is clear that K goes to infinity as N goes to infinity in the linear sparsity regime. Then, let $M = cK$ where $c > 1$. From (32),

$$\log \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} \leq 2^{-1}SK(c-1) \underbrace{(\log(1+d_1) - d_1)}_{=:A} + \log 2$$

where $A < 0$ due to (33). Thus,

$$\lim_{N \rightarrow \infty} \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} \leq \lim_{K \rightarrow \infty} \exp\left(2^{-1}SK(c-1)A + \log 2\right) = 0$$

implying that the probability that the correct support set is not δ jointly typical with all the measurement vectors vanishes.

Next, from (40),

$$\begin{aligned} \log \sum_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} &\leq \log\left(\binom{N}{K} p(d_{2,a^*} - 1)\right) \\ &= \log\binom{N}{K} + 2^{-1}SK(c-1) \underbrace{(\log(1-t) + t)}_{=: \gamma} \\ &\leq K \underbrace{\left(1 + \log \frac{N}{K} + 2^{-1}S(c_1 - 1)\gamma\right)}_{=: \eta} \end{aligned} \quad (59)$$

where the last inequality is due to

$$\binom{N}{K} \leq \exp\left(K \log \frac{Ne}{K}\right). \quad (60)$$

In (59), $\gamma < 0$ for any t where

$$t = \frac{1 - \rho^{-1}}{1 + \text{SNR}_{\min}^{-1}} \in (0, 1). \quad (61)$$

If $c > 1 + S^{-1}v_1$, then $\eta < 0$, which yields

$$\lim_{N \rightarrow \infty} \sum_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} \leq \lim_{K \rightarrow \infty} \exp(K\eta) = 0$$

implying that the probability that all incorrect support sets are δ jointly typical with all the measurement vectors vanishes. Thus the failure probability p_{err} defined in (8) converges to zero if M satisfies (9).

Next, the remain is to derive (11) in the sublinear sparsity regime. Similarly, let $M = K + cK \log \frac{N}{K}$ where $c > 1$. From (32),

$$\log \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} \leq 2^{-1}ScK \log \frac{N}{K} \underbrace{(\log(1+d_1) - d_1)}_{=:A} + \log 2$$

where $A < 0$ due to (33). Thus,

$$\lim_{N \rightarrow \infty} \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} \leq \lim_{N \rightarrow \infty} \exp\left(2^{-1}ScKA \log \frac{N}{K} + \log 2\right) = 0$$

implying that the probability that the correct support set is not δ jointly typical with all the measurement vectors vanishes.

Then, from (40),

$$\begin{aligned} \log \sum_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} &\leq \log\left(\binom{N}{K} p(d_{2,a^*} - 1)\right) \\ &= \log\binom{N}{K} + 2^{-1}ScK \underbrace{(\log(1-t) + t)}_{=: \gamma} \log \frac{N}{K} \\ &\leq K \underbrace{\left(1 + 2^{-1}Sc\gamma\right)}_{=: \eta} \log \frac{N}{K} + K \end{aligned}$$

where the last inequality is due to the bound in (60) and $\gamma < 0$ for any t in (61). If $c > S^{-1}v_2$, then $\eta < 0$, which yields

$$\lim_{N \rightarrow \infty} \sum_{\mathcal{J} \in \mathcal{S} \setminus \mathcal{I}} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} \leq \lim_{N \rightarrow \infty} \exp\left(K\eta \log \frac{N}{K} + K\right) = 0$$

implying that the probability that all incorrect support sets are δ jointly typical with all the measurement vectors vanishes. Thus, the failure probability p_{err} defined in (8) converges to zero if M satisfies (11), which completes the proof. ■

B. Proof of Theorem 2

From Lemma 1,

$$\mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} \leq 2 \underbrace{\left(\exp\left(-\frac{M-K}{2}d_1\right) (1+d_1)^{\frac{M-K}{2}}\right)^S}_{=: \mu_{\mathcal{I}}}. \quad (62)$$

If $M \geq K + 1$, we have

$$\log \mu_{\mathcal{I}} = 2^{-1} (M - K) (\log(1 + d_1) - d_1) < 0 \quad (63)$$

due to (33), which implies $\mu_{\mathcal{I}} < 1$. From Lemma 2,

$$\mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} \leq \left(\underbrace{\exp\left(-\frac{M-K}{2} (d_{2,a^*} - 1)\right)}_{=:\mu_{\mathcal{J}}} d_{2,a^*}^{\frac{M-K}{2}} \right)^S. \quad (64)$$

Similarly, if $M \geq K + 1$, we have

$$\log \mu_{\mathcal{J}} = 2^{-1} (M - K) (\log(1 - t) + t) < 0 \quad (65)$$

due to (61), which implies $\mu_{\mathcal{J}} < 1$. Thus, we conclude

$$\lim_{S \rightarrow \infty} p_{err} \leq 2 \lim_{S \rightarrow \infty} \mu_{\mathcal{I}}^S + \binom{N}{K} \lim_{S \rightarrow \infty} \mu_{\mathcal{J}}^S = 0$$

for $M \geq K + 1$ which completes the proof. ■

C. Proof of Theorem 3

The mutual information in (24) is bounded by

$$\begin{aligned} \mathbb{I}(\mathbf{x}; \mathbf{y} | \tilde{\mathbf{F}}) &= h(\mathbf{y} | \tilde{\mathbf{F}}) - h(\mathbf{y} | \mathbf{x}, \tilde{\mathbf{F}}) \leq h(\mathbf{y}) - h(\mathbf{n}) \\ &\leq \sum_{i=1}^{SM} h(y_i) - h(\mathbf{n}) \\ &\leq 2^{-1} SM (\log(2\pi e (Kx_{\min}^2 + \sigma^2))) - \log(2\pi e \sigma^2) \\ &= 2^{-1} SM \log(1 + K \times \text{SNR}_{\min}) \end{aligned}$$

where $h(\mathbf{x})$ is the differential entropy of \mathbf{x} , and $h(\mathbf{x} | \mathbf{y})$ is the conditional entropy of \mathbf{x} given \mathbf{y} . The last inequality is due to that the Gaussian distribution maximizes the differential entropy. The denominator in (24) is bounded by

$$\log(|\mathcal{X}_{\{x_{\min}\}}| - 1) = \log\left(\binom{N}{K} - 1\right) > K \log \frac{N}{K}$$

for sufficiently large N . Then,

$$\begin{aligned} p_{err} &= \mathbb{E}_{\tilde{\mathbf{F}}} \sup_{\mathbf{x} \in \mathcal{X}_{x_{\min}}} \mathbb{P}\{\hat{\mathcal{I}} \neq \mathcal{I} | \mathbf{x}, \tilde{\mathbf{F}}\} \\ &\geq \mathbb{E}_{\tilde{\mathbf{F}}} \min_{\hat{\mathbf{x}} \in \mathcal{X}_{\{x_{\min}\}}} \max_{\mathbf{x} \in \mathcal{X}_{\{x_{\min}\}}} \mathbb{P}\{\hat{\mathbf{x}} \neq \mathbf{x} | \mathbf{x}, \tilde{\mathbf{F}}\} \\ &> 1 - \frac{2^{-1} SM \log(1 + K \times \text{SNR}_{\min}) + \log 2}{K \log \frac{N}{K}}. \quad (66) \end{aligned}$$

From (66), the failure probability is bounded away from zero by zero if (25) is satisfied, which completes the proof. ■

APPENDIX D

PROOFS OF COROLLARIES 1, 2 AND 3

A. Proof of Corollary 1

From the inequality $\log(1 + x) \leq \frac{2x}{2+x}$ for $x \in (-1, 0]$,

$$v_2 = -\frac{2}{\log(1-t) + t} < \frac{4-2t}{t^2} < \frac{4}{t^2} \quad (67)$$

where t is defined in (61). Then,

$$\frac{v_2}{S} < \frac{4}{St^2}. \quad (68)$$

From (61),

$$\sqrt{S}t = \frac{1 - \rho^{-1}}{-\sqrt{S} + (\sqrt{S} \times \text{SNR}_{\min})^{-1}}. \quad (69)$$

Combining (11), (68) and (69) leads to (13). This approach is used to get (14) using the following equality

$$v_1 = v_2 (1 - \log \beta) \quad (70)$$

where $\lim_{N \rightarrow \infty} \frac{K}{N} = \beta \in (0, 1/2)$, which completes the proof. ■

B. Proof of Corollary 2

Substituting $\alpha = \frac{2}{3}$ in (15), and rearranging the result with respect to t can yield $\frac{2}{3} \leq t < 1$, where t is defined in (61). Then from (67), a simple computation yields that

$$v_2 < \frac{4-2t}{t^2} \leq \frac{4}{t}$$

which immediately yields that

$$\frac{v_2}{S} < \frac{4}{St}. \quad (71)$$

where

$$St = \frac{1 - \rho^{-1}}{S^{-1} + (S \times \text{SNR}_{\min})^{-1}}. \quad (72)$$

Combining (11), (71) and (72) leads to (16). This approach is used to get (17) using (70), which completes the proof. ■

C. Proof of Corollary 3

We assume that $\mu_{\mathcal{I}} \geq \mu_{\mathcal{J}}$ and

$$p_{err} \leq \mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} + \sum_{\mathcal{J} \in S \setminus \mathcal{I}} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} \leq \left(\binom{N}{K} + 2\right) \mu_{\mathcal{I}}^S < \varepsilon < 1. \quad (73)$$

Then, if the number of measurement vectors satisfies

$$S > \frac{\log \varepsilon - \log\left(\binom{N}{K} + 2\right)}{\log \mu_{\mathcal{I}}} > 0, \quad (74)$$

(73) is achieved for small ε , and hence, reliable support set reconstruction is possible. If $\mu_{\mathcal{I}} < \mu_{\mathcal{J}}$, we obtain inequalities similar to (73) and (74) by replacing $\mu_{\mathcal{I}}$ by $\mu_{\mathcal{J}}$, where

$$S > \frac{\log \varepsilon - \log\left(\binom{N}{K} + 2\right)}{\log \mu_{\mathcal{J}}} > 0. \quad (75)$$

Combining (74) and (75) yields (18).

Next, a simple computation yields that for any d_1 in (33),

$$\frac{\partial \log \mu_{\mathcal{I}}}{\partial d_1} = -\frac{d_1}{2(1+d_1)} < 0$$

where $\log \mu_{\mathcal{I}}$ is given in (63). From (33), we see $d_1 \propto \text{SNR}_{\min}$ that leads to $\log \mu_{\mathcal{I}} \propto \text{SNR}_{\min}^{-1}$. Also, for any t in (61),

$$\frac{\partial \log \mu_{\mathcal{J}}}{\partial t} = -\frac{t}{2(1-t)} < 0$$

where $\log \mu_{\mathcal{J}}$ is given in (65). From (61), we see $t \propto \text{SNR}_{\min}$ that leads to $\log \mu_{\mathcal{J}} \propto \text{SNR}_{\min}^{-1}$. Hence, the sufficient condition on S in (18) turns out to be a decreasing function with respect to SNR_{\min} , which completes the proof. ■

APPENDIX E
PROOFS OF PROPOSITIONS 1 AND 2

First of all, we introduce the exponential inequalities [32], and use them in the proofs of Propositions 1 and 2.

A. The Exponential Inequalities [32]

Let Y_i , $i = 1, 2, \dots, D$ be i.i.d. Gaussian variables with a zero mean and a unit variance. Then, let α_i , $i = 1, 2, \dots, D$ be non-negative. We set

$$|\alpha|_\infty = \sup |\alpha_i|, \quad |\alpha|_2^2 = \sum_{i=1}^D \alpha_i^2$$

and let

$$Y = \sum_{i=1}^D \alpha_i (Y_i^2 - 1). \quad (76)$$

Then, the following inequalities hold for any positive x

$$\mathbb{P}\{Y \geq 2|\alpha|_2 \sqrt{x} + 2|\alpha|_\infty x\} \leq \exp(-x) \quad (77)$$

$$\mathbb{P}\{Y \leq -2|\alpha|_2 \sqrt{x}\} \leq \exp(-x). \quad (78)$$

B. Proof of Proposition 1

In the proof of Lemma 3, $Z_{\mathcal{I}}$ is represented by

$$Z_{\mathcal{I}} = \sum_{s=1}^S \sum_{i=1}^{M-K} w^s(i)^2$$

where $w^s(i)$ is Gaussian with a zero mean and a unit variance. Define a random variable Y as

$$Y = Z_{\mathcal{I}} - S(M-K) = \sum_{s=1}^S \sum_{i=1}^{M-K} (w^s(i)^2 - 1)$$

which is of the form of (76). Then,

$$\mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} = \underbrace{\mathbb{P}\{Y \leq -SM\delta/\sigma^2\}}_{=:A} + \underbrace{\mathbb{P}\{Y \geq SM\delta/\sigma^2\}}_{=:B}.$$

Combining A with (78) gives

$$\begin{aligned} \mathbb{P}\{Y \leq -SM\delta/\sigma^2\} &= \mathbb{P}\{Y \leq -2\sqrt{S(M-K)}x\} \\ &\leq \underbrace{\exp\left(-\frac{SM^2\delta^2}{4(M-K)\sigma^4}\right)}_{=:C} \end{aligned}$$

and combining B with (77) gives

$$\begin{aligned} \mathbb{P}\{Y \geq SM\delta/\sigma^2\} &= \mathbb{P}\{Y \geq 2\sqrt{S(M-K)}x + 2x\} \\ &\leq p_{1,\text{exp}} \end{aligned}$$

where $p_{1,\text{exp}}$ is defined in (28). It is readily seen that $p_{1,\text{exp}} \geq C$, which leads to $\mathbb{P}\{\mathcal{E}_{\mathcal{I}}^c\} \leq 2p_{1,\text{exp}}$.

Next, from (32) and (28),

$$\log p(d_1) = 2^{-1}S(M-K)(\log(1+d_1) - d_1)$$

and

$$\log p_{1,\text{exp}} = -2^{-1}S(M-K)d_1^2(2+4d_1)^{-1}$$

where $d_1 > 0$ is defined in (33). Then, we have

$$\log \frac{p(d_1)}{p_{1,\text{exp}}} = \frac{S(M-K)}{2} \underbrace{\left(\log(1+d_1) - d_1 + d_1^2(2+4d_1)^{-1}\right)}_{=:g(d_1)}.$$

For any $d_1 > 0$, $\frac{\partial g(d_1)}{\partial d_1} = -d_1^2(2+3d_1)(1+d_1)^{-1}(1+2d_1)^{-2} < 0$ and $\max_{d_1>0} g(d_1) = 0$. Thus, we conclude $\log \frac{p(d_1)}{p_{1,\text{exp}}} \leq 0$, which completes the proof. ■

C. Proof of Proposition 2

In the proof of Lemma 4, $Z_{\mathcal{J}}$ is represented by

$$\begin{aligned} Z_{\mathcal{J}} &= \sum_{s=1}^S \sum_{i=1}^{M-K} b^s(i)^2 \\ &= \sum_{s=1}^S \sum_{i=1}^{M-K} \alpha_{\mathcal{J},s} g^s(i)^2 \end{aligned}$$

where $\alpha_{\mathcal{J},s}$ is defined in (39) and $g^s(i)$ is Gaussian with a zero mean and a unit variance. Define a new random variable Y as

$$\begin{aligned} Y &= Z_{\mathcal{J}} - S(M-K) \\ &= \sum_{s=1}^S \sum_{i=1}^{M-K} \alpha_{\mathcal{J},s} (g^s(i)^2 - 1) \end{aligned}$$

which is of the form of (76). Then, from (44)

$$\begin{aligned} \mathbb{P}\{\mathcal{E}_{\mathcal{J}}\} &\leq \mathbb{P}\left\{Y < SM\delta - (M-K) \sum_{s=1}^S \|\mathbf{x}_{\mathcal{I} \setminus \mathcal{J}}^s\|_2^2\right\} \\ &\leq \mathbb{P}\left\{Y < \underbrace{SM\delta - S(M-K)x_{\min,\mathcal{J}}^2}_{=:A}\right\} \\ &\leq p_{2,\mathcal{J},\text{exp}} \end{aligned} \quad (79)$$

where $p_{2,\mathcal{J},\text{exp}}$ is defined in (30), the last inequality is due to (78). Due to (29), A is negative. Thus the exponential inequality of (78) gives the upper bound $p_{2,\mathcal{J},\text{exp}}$.

Next, from (40) and (30),

$$\log p(d_{2,\lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1) = 2^{-1}S(M-K)(t + \log(1-t))$$

and

$$\begin{aligned} \log p_{2,\mathcal{J},\text{exp}} &\geq -\frac{S(M-K)}{4} \left(\frac{x_{\min,\mathcal{J}}^2 - \frac{M\delta}{M-K}}{x_{\min,\mathcal{J}}^2 + \sigma^2}\right)^2 \\ &= -4^{-1}S(M-K)t^2 \end{aligned}$$

where $t \in (0, 1)$, is defined in (61) and the inequality is due to (50). Then,

$$\log \frac{p(d_{2,\lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1)}{p_{2,\mathcal{J},\text{exp}}} \leq \frac{S(M-K)}{4} \underbrace{\left(t^2 + 2t + 2\log(1-t)\right)}_{=:g(t)}.$$

For any $t \in (0, 1)$, $\frac{\partial g(t)}{\partial t} = -2t^2(1-t)^{-1} < 0$ and

$\max g(t) = 0$. We conclude $\log \frac{p(d_{2,\lambda_{\min}(\mathbf{R}_{\mathcal{J}})} - 1)}{p_{2,\mathcal{J},\text{exp}}} \leq 0$. It completes the proof. ■

REFERENCES

- [1] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul. 1948.
- [2] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [3] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

- [4] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.
- [5] E. J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.
- [6] E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.
- [7] E. J. Candès and M. B. Wakin, "An introduction to compressive sensing," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
- [8] D. Baron, M. B. Wakin, M. F. Duarte, S. Sarvotham, and R. G. Baraniuk. (2009). "Distributed compressed sensing." [Online]. Available: <https://arxiv.org/pdf/0901.3403.pdf>
- [9] M. F. Duarte, S. Sarvotham, D. Baron, M. B. Wakin, and R. G. Baraniuk, "Distributed compressed sensing of jointly sparse signals," in *Proc. Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Nov. 2005, pp. 1537–1541.
- [10] C. Caione, D. Brunelli, and L. Benini, "Compressive sensing optimization for signal ensembles in WSNs," *IEEE Trans. Ind. Informat.*, vol. 10, no. 1, pp. 382–392, Feb. 2014.
- [11] W. Chen, R. D. Rodrigues, and I. J. Wassell, "Distributed compressive sensing reconstruction via common support discovery," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kyoto, Japan, Jun. 2011, pp. 1–5.
- [12] J. Oliver, W.-B. Lee, and H.-N. Lee, "Filters with random transmittance for improving resolution in filter-array-based spectrometers," *Opt. Exp.*, vol. 21, no. 4, pp. 3969–3989, Feb. 2013.
- [13] S. Gogineni and A. Nehorai, "Target estimation using sparse modeling for distributed MIMO radar," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5315–5325, Nov. 2011.
- [14] Y. Wu *et al.*, "Accelerated MR diffusion tensor imaging using distributed compressed sensing," *Magn. Reson. Med.*, vol. 71, no. 2, pp. 763–772, Feb. 2014.
- [15] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer, 2010.
- [16] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, May 2012.
- [17] Y. C. Eldar, *Sampling Theory: Beyond Bandlimited Systems*. Cambridge, U.K.: Cambridge Univ. Press, Apr. 2015.
- [18] M. J. Wainwright, "Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting," *IEEE Trans. Inf. Theory*, vol. 55, no. 12, pp. 5728–5741, Dec. 2009.
- [19] A. K. Fletcher, S. Rangan, and V. K. Goyal, "Necessary and sufficient conditions for sparsity pattern recovery," *IEEE Trans. Inf. Theory*, vol. 55, no. 12, pp. 5758–5772, Dec. 2009.
- [20] S. Aeron, V. Saligrama, and M. Zhao, "Information theoretic bounds for compressed sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5111–5130, Oct. 2010.
- [21] M. Akçakaya and V. Tarokh, "Shannon-theoretic limits on noisy compressive sampling," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 492–504, Jan. 2010.
- [22] J. Scarlett, J. S. Evans, and S. Dey, "Compressed sensing with prior information: Information-theoretic limits and practical decoders," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 427–439, Jan. 2013.
- [23] J. Scarlett and V. Cevher, "Limits on support recovery with probabilistic models: An information-theoretic framework," *IEEE Trans. Inf. Theory*, vol. 63, no. 1, pp. 593–620, Jan. 2017.
- [24] T. Cover and J. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 2006.
- [25] J. Chen and X. Huo, "Theoretical results on sparse representations of multiple-measurement vectors," *IEEE Trans. Signal Process.*, vol. 56, no. 12, pp. 4634–4643, Dec. 2006.
- [26] G. Tang and A. Nehorai, "Performance analysis for sparse support recovery," *IEEE Trans. Inf. Theory*, vol. 56, no. 3, pp. 1383–1399, Mar. 2010.
- [27] Y. Jin and B. D. Rao, "Support recovery of sparse signals in the presence of multiple measurement Vectors," *IEEE Trans. Inf. Theory*, vol. 59, no. 5, pp. 3139–3157, May 2013.
- [28] M. F. Duarte, M. B. Wakin, D. Braon, S. Sarvotham, and R. G. Baraniuk, "Measurement bounds for sparse signal ensembles via graphical models," *IEEE Trans. Inf. Theory*, vol. 59, no. 7, pp. 4280–4289, Jul. 2013.
- [29] J. Kim, O. Lee, and J. Ye, "Compressive MUSIC: Revisiting the link between compressive sensing and array signal processing," *IEEE Trans. Inf. Theory*, vol. 58, no. 1, pp. 278–301, Jan. 2012.
- [30] J. D. Blanchard and M. E. Davies, "Recovery guarantees for rank aware pursuits," *IEEE Signal Process. Lett.*, vol. 19, no. 7, pp. 427–430, Jul. 2012.
- [31] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part I*. New York, NY, USA: Wiley, 2001.
- [32] B. Laurent and P. Massart, "Adaptive estimation of a quadratic functional by model selection," *Ann. Statist.*, vol. 28, no. 5, pp. 1303–1338, Oct. 2000.
- [33] L. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Reading, MA, USA: Addison-Wesley, 1991.
- [34] R. Rajagopalan and P. K. Varsheny, "Data-aggregation techniques in sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 8, no. 4, pp. 48–63, 4th Quart., 2006.

Sangjun Park received the B.S. degree in computer engineering from the Chungnam National University, Daejeon, Korea, in 2009. He is pursuing a Ph.D degree at the School of Electrical Engineering and Computer Science in the Gwangju Institute of Science and Technology (GIST), Gwangju, Korea. His research interests include information theory, numerical optimization and compressed sensing.

Nam Yul Yu (M'07) received the B.S. degree in electronics engineering from the Seoul National University, Seoul, Korea, in 1995 and the M.S. degree in electronic and electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, Korea, in 2000. He received the Ph. D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, Ontario, Canada, in 2007. From 2000 to 2003, he was with Telecommunication Research and Development Center, Samsung Electronics, Korea, where he worked on channel coding schemes for wireless communication systems. In 2007, he was a senior research engineer in LG Electronics, Korea, working on the standardization of the 3GPP-LTE. From 2008 to 2014, he has been working as an Assistant/Associate Professor in the Department of Electrical Engineering at the Lakehead University, Thunder Bay, Ontario, Canada. In 2014, he joined the Gwangju Institute of Science and Technology (GIST), Gwangju, Korea, and is currently working as an Associate Professor in the School of Electrical Engineering and Computer Science. From 2009 to 2011, he served as an Associate Editor for Sequences in the IEEE Transactions on Information Theory. His research interests include sequence design, compressed sensing, and security for wireless communications.

Heung-No Lee (SM'13) received the B.S., M.S., and Ph.D. degrees from the University of California, Los Angeles, CA, USA, in 1993, 1994, and 1999, respectively, all in electrical engineering. He then worked at HRL Laboratories, LLC, Malibu, CA, USA, as a Research Staff Member from 1999 to 2002. From 2002 to 2008, he worked as an Assistant Professor at the University of Pittsburgh, PA, USA. In 2009, he then moved to the School of Electrical Engineering and Computer Science, GIST, Korea, where he is currently affiliated. His areas of research include information theory, signal processing theory, communications/networking theory, and their application to wireless communications and networking, compressive sensing, future internet, and brain-computer interface. He has received several prestigious national awards, including the Top 100 National Research and Development Award in 2012, the Top 50 Achievements of Fundamental Researches Award in 2013, and the Science/Engineer of the Month (January 2014). He has written more than fifty international journal publications and a hundred international conferences and workshop papers. He was the Director of Electrical Engineering and Computer Science track within GIST College in 2014. In March 2015, he was appointed as the Dean of Research at GIST.

Time-Variant Proof-of-Work Using Error-Correction Codes

Sangjun Park, Haeung Choi, and *Heung-No Lee, *Senior Member, IEEE*

Abstract— The protocol for cryptocurrencies can be divided into three parts, namely consensus, wallet, and networking overlay. The aim of the consensus part is to bring trustless rational peer-to-peer nodes to an agreement to the current status of the blockchain. The status must be updated through valid transactions. A proof-of-work (PoW) based consensus mechanism has been proven to be secure and robust owing to its simple rule and has served as a firm foundation for cryptocurrencies such as Bitcoin and Ethereum. Specialized mining devices have emerged, as rational miners aim to maximize profit, and caused two problems: *i*) the re-centralization of a mining market and *ii*) the huge energy spending in mining. In this paper, we aim to propose a new PoW called Error-Correction Codes PoW (ECCPoW) where the error-correction codes and their decoder can be utilized for PoW. In ECCPoW, puzzles can be intentionally generated to vary from block to block, leading to a time-variant puzzle generation mechanism. This mechanism is useful in repressing the emergence of the specialized mining devices. It can serve as a solution to the two problems of recentralization and energy spending.

Index Terms— Consensus, Cryptocurrency, Blockchain, Proof-of-Work, Error-Correction Codes, Hash Functions

I. INTRODUCTION

In cryptocurrencies, the consensus part plays a role in leading an agreement among trustless nodes without any communications. This part is the most innovative because it can prevent the double spending attack [1] in a peer-to-peer network in the absence of trusted parties. In *Bitcoin* [2], as an example, more than ten thousand of nodes randomly scattered across the world aim to reach a consensus in each block time. The Internet is the only way to connect them; communication packets are delayed and sometimes dropped though the Internet that is designed to provide the best effort service. Cyberattacks frequently happen, making transactions over the Internet insecure. Nevertheless, Bitcoin has shown secure peer-to-peer transactions over the past 10 years. With the help of proof-of-work (*PoW*) which is fundamental to the consensus part, this can be possible.

In Bitcoin, each node does competitive work, called *mining*, to forge a block. The node which wins this competition has the right to mint a specified number of coins as this mining reward.

S. Park is with the Electronics and Telecommunications Research Institute (ETRI), South Korea (e-mail: sjpark86@etri.re.kr)

H. Choi and H.-N. Lee are with the School of Electrical Engineering and Computer Science (EECS), Gwangju Institute of Science and Technology (GIST), South Korea (e-mail: haeung@gist.ac.kr, heungno@gist.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean government (MSIP) (NRF-2018R1A2A1A19018665). The corresponding authors: Heung-No Lee (heungno@gist.ac.kr)

If a node was re-forging all the blocks alone, it could spend the total amount of works done to all the mined blocks.

Without PoW, anybody with a computer can alter the content of the blockchain, implying unauthorized changes in any mined blocks can be possible. If PoW is attached to each mined block, attackers cannot make any unauthorized modifications without redoing all the works. No node can alone alter any mined block, meaning an immutability property.

In Bitcoin, miners make rational decisions to maximize their profits by following a two stage process in which *i*) the miners select a blockchain whose length is the longest and *ii*) they extend this longest one by adding a newly mined block. Suppose there are two blockchains where one is longer than the other one in terms of the length. Since the longer chain has the more accumulated works, altering it is more difficult. This longer chain shall be treated the more trustable and preferable by the miners. Thus, they select the longer chain. Making such a selection is rational for the sake of keeping the mining rewards. The mining reward is a delayed conditional payment, i.e., if a miner mines a block at a given time point t_1 , the reward is delayed until the future moment t_2 of time. This time from t_1 to t_2 is measured in terms of the number of blocks, say 100 blocks. If this mined block was not a part of the longest chain at the future time point t_2 , the reward vanishes. Thus, rational miners select the longest chain.

In Bitcoin, miners spend computational resources to forge a block by solving a puzzle carved in a bitcoin program as an on-chain policy. This puzzle is made using the secure hash algorithm 256 (SHA256) [3]. To solve the puzzle, the miners have to repeatedly execute SHA256 by varying an input to SHA256 until a good hash is given. This input is the header of the block, i.e., block header, including six fields such as version, previous hash, difficulty, timestamp and nonce, Merkle tree value. The version is fixed. Given a block at a certain height, i.e., the l^{th} block, the previous hash and the difficulty are obtained from its previous block, i.e., $(l-1)^{\text{th}}$ block. They are constant. The rest varies until SHA256 returns a good hash. A good hash can be spotted since it shall possess a certain number of leading zero bits reflecting the difficulty. The block header of a mined block can serve as the proof that a given puzzle is solved without any falsehood.

Satoshi [2] intended for miners to execute SHA256 using a central processing unit (CPU). But, faster computing machines based on application-specific integrated circuit (ASIC) became available. As a result, the miners have chosen to exploit them to maximize their profits. To date, the miners equipped with ASIC

mining devices have dominated the mining business [4], leading to two problems:

- M1. The mining markets have become re-centralized [5].
- M2. The electrical energy spent to mine blocks is huge [6].

First, the miners have a large portion of the total hash power, implying that the plight of the blockchain is left to a handful of these influential miners. It can be possible to modify any mined blocks on their own rights, leading to shattered trust. Namely, they can break the immutability property. Second, new models of ASIC mining devices can surpass old models with respect to the hash power, which is measured as a hash rate. Each miner is forced to buy newer models to win the mining competition. As the new models are widely used, the total hash power inevitably grows. The difficulty level to solve the puzzle must increase to keep a certain predefined range of block generation time that is expectedly consumed to mine a single block. Today, this level has gotten huge, i.e., $O(10^{20})$ hash rate per second. As a result, using CPUs in mining has gotten no longer profitable. Besides, as the total hash power increases, the miners spend more and more electrical energy to mine blocks.

If we prevent the usage of ASIC mining, we can alleviate the problems M1 and M2. To this end, we use the error-correction codes and their decoder. In general, the aim of using the codes in modern communication systems is to combat errors occurring over noisy channels in which the errors introduced over a noisy channel can be corrected by running a decoding algorithm. The codes have a rich history where there are numerous classes of good codes available. They have been used to define both a good public-key crypto system and a good hash function.

The first result can be traced back to the late 1970. McEliece [8] used Goppa codes to make a McEliece cryptosystem where a message is encoded using a public key $\mathbf{A} := \mathbf{SGP}$ where \mathbf{G} is the generator matrix of a binary Goppa code, \mathbf{S} is a nonsingular random matrix and \mathbf{P} is a permutation matrix. The hash, i.e., the encoded result, of a provided message is made as follows: *a*) a word is made by multiplying the message with the public key and *b*) adding a binary random word whose number of ones is at most t to this word¹ is to get the hash. Peters *et al.* [9] extended this system using non-binary Goppa codes to reduce the size of its public key. Even the size reduces, this system can achieve still the same security level as much as that of [8] could. Other codes such as low density generator matrix codes [10], low density parity check (LDPC) codes [11], [12], Reed-Solomon codes [13] and Reed-Muller codes [14] have been used to replace the Goppa codes. The aim of using them is to reduce the size of the public key.

Aside from the applications of the error-correction codes into the McEliece cryptosystem, the codes are used to construct new hash functions. Preneel [15] proposed a method to make new hash functions and proved that their hash functions can provide strong collision resistant properties. The codes in [15] are either the maximum distance separable codes or the Hamming codes.

¹ For any Goppa code, there is a construction method to guarantee that the minimum distance d of that code is greater than a given positive integer. Thus, the value of t can be known in advance using Theorem 1 [32].

Selman *et al.* [16] used LDPC codes to make a hash function and proved this function as an average universal hash function defined in [17]. These results can motivate us to exploit error-correction codes in designing a new PoW framework.

The contributions of this paper are three folds. First, we propose a new PoW framework which we name as ECCPoW. As the name implies, we add the error-correction code part into PoW. To the best of our knowledge, this is the first work in which the error-correction codes are applied to blockchains. We then explain how we make puzzles, which we call ECC puzzles, and give routines to solve them.

Second, we conduct a probabilistic study to examine a random variable called First Success Hash Cycle (FSHC) representing the number of hash cycles required in solving a given ECC puzzle. Based on this study, we get the following results:

- FSHC follows a geometric distribution with a parameter in (21) that depends on the number of miners M and the code length n .
- The expected value of FSHC is a decreasing function with respect to the number of miners M .
- The expected value of FSHC is an increasing function with respect to the code length n .

Third, we define five properties for a good PoW and explain how ECCPoW satisfies these properties. We shall note that the most innovative property is the time-variant property, making ECCPoW suitable to resolve the problems of recentralization and energy spending.

We organize the rest of this paper as follows. Section II gives literature surveys regarding SHAs and PoWs. Section III elucidates LDPC codes and a decoder. Section IV addresses how ECCPoW works and gives its pseudo codes Section V presents theoretical results of ECCPoW. Section VI discusses properties of ECCPoW. Section VII presents the conclusions of this paper.

II. LITERATURE ON BLOCKCHAIN CRYPTOGRAPHIES

A. Secure hash standard and functions

The secure hash standard was formulated by NIST [3]. The purpose of this standard is to offer the specifications of SHAs that yield a hash of a given message. Even the message changes slightly, the hash of the changed message comes out completely different from that of the original message. Thus, a hash can be used to detect whether an original message was altered or not. SHAs with such a property can be used for the generation and the verification of digital signatures as well as for the message authentication.

A secure hash function takes an arbitrarily sized message and produces a fixed-size hash. Let a function h be

$$h: \mathcal{X} \rightarrow \mathcal{Y}$$

which is said to be a cryptographically secure hash function if it satisfies the three requirements defined in [17] below:

(One-way function) Given any hash y to which a corresponding message is not known, it is computationally infeasible to find a message x such that $h(x) = y$.

TABLE I. The routine of bitcoin

Inputs: \mathcal{S}_0 (block header except for nonce) and L	
Step 1:	for nonce = 0, 1, 2, ... $2^{32} - 1$
Step 2:	$e = \text{SHA256}(\mathcal{S}_0 \cup \{\text{nonce}\})$
Step 3:	If e begins with L zero bits, then go to Step 5.
Step 4:	end
Step 5:	Block generation & broadcast

(Weak collision resistance) Given an arbitrary message x , it is computationally infeasible to find any message x' which results in the same hash, i.e., $h(x) = h(x')$.

(Strong collision resistance) It is computationally infeasible to find any two different messages x and x' which make the same hashes $h(x) = h(x')$.

NIST [3] has proposed a family of SHAs including SHA1, SHA224, SHA256, SHA384, and SHA512. A message of any size less than 2^{64} bits can be given as an input for SHA1, SHA224, and SHA256, while that of less than 2^{128} bits for SHA384 and SHA512. The size of a hash ranges from 160 to 512 bits, depending on the algorithm.

B. PoW of Bitcoin

In Table I, we define a puzzle in Bitcoin and give routines to solve this puzzle. In Step 2, a miner puts a given block header with a selected nonce to SHA256 and obtain a hash of 256-bits. In Step 3, this puzzle is declared to be solved if a hash is smaller than a specified target value, or in a simpler argument it begins with L zero bits, where the target value or the value of L is given as the difficulty level of the PoW puzzle. The miner repeats the routines from Step 2 to Step 3 by varying the nonce. However, there exists a chance in which the miner can fail to hit a good hash even though the whole range of nonces, i.e., $0 \sim 2^{32} - 1$, is used. In such a case, the miner updates the block header, and repeats the routines from Step 2 and Step 3. There are two methods to update the block header. The first one is to update the timestamp field. The second one is to update the Merkle tree value by modifying the list of transactions being included in a block which the miner aims to construct.

C. PoW of Ethereum

Ethash [18] was created for the purpose of preventing the advent of ASIC mining devices in Ethereum. In Ethash, there is a memory structure called directed acyclic graph (DAG) where its data are randomly re-generated every 30,000 block.

Table II shows routines of Ethash. As we have shown in Step 2, the current block header with a nonce is taken by SHA3 to get a hash. This hash is taken by a predefined function to yield mix0 that is random. In Step 4, mix0 is used to determine which data from DAG are fetched. No one predicts which data shall be fetched from DAG because mix0 is random. The mixer takes both the fetched data and mix0 to get a random value in Step 5. In Step 6, mix0 is updated using the random value. The routines from Step 2 to Step 6 are repeated 63 times. Last, the decision is made using this final mix0 , as we have shown in Step 8.

The ASIC resistant property in Ethash is originated from the fact that the operation time for the mixer is shorter than that of

TABLE II. The routine of Ethash

Inputs: \mathcal{S}_0 (block header except for nonce), L and DAG	
Step 1:	for nonce = 0, 1, 2, ... $2^{32} - 1$
Step 2:	$\text{mix0} = f(\text{SHA3}(\mathcal{S}_0 \cup \{\text{nonce}\}))$
Step 3:	for $i = 1, 2, \dots, 63$
Step 4:	$\text{data1} = \text{Fetch}(\text{DAG}, \text{mix0})$
Step 5:	$\text{tmp} = \text{Mixing}(\text{mix0}, \text{data1})$
Step 6:	$\text{mix0} = f(\text{tmp})$
Step 7:	end
Step 8:	If mix0 begins with L zero bits, then go to Step 10.
Step 9:	end
Step 10:	Block generation & broadcast

where f is a predefined function. Details on f is given in [18].

the fetch operation. To be specific, let A_i be the time duration (TD) to conduct the mixing operation in which the subscript i denotes a chip to run this mixing operation. It is clear that the mixing operation TD based on an ASIC chip is significantly less than that based on any CPU chip because the clock speed of an ASIC chip is much faster than that of any CPU chip, i.e.,

$$A_{\text{ASIC}} \ll A_{\text{CPU}}.$$

Next, let B_i be the TD to conduct the fetch operation. Unlike to the mixing operation TD, this TD depends on the communications bandwidth between the memory and the cache of the CPU in which the fetched data from DAG are passed through. In other words, the fetch operation time is connected mainly to the bandwidth but not to the clock speed. The purpose to use ASIC chips is to increase the processing speed; is not for obtaining a higher communications bandwidth. The fetch operation TD for CPU and ASIC are thus similar, i.e.,

$$B_{\text{ASIC}} \approx B_{\text{CPU}}.$$

We consider the inner routines of Ethash, i.e., Step 4 and Step 5. The mixing operation is conducted after the fetching operation is done. This operation TD at Step 4 can be significantly reduced using ASIC chips; but the fetching operation TD at Step 5 is not reduced even if ASIC chips are used. As a result, there is a bottleneck between Step 4 and Step 5. This bottleneck has the ASIC resistant property enabled.

Recently, a programmatic PoW (ProgPoW), which is planning to be used to replace Ethash, has been proposed to further improve the ASIC resistant property. This improvement is done by changing parameters related to DAG randomly from block to block. Such modifications can make the fetch operation time increased. But, the development of ProgPoW is not completed and ProgPoW is not proven to be secure at the time of writing this manuscript.

D. PoW of Dash

X11 was proposed in 2014 by Duffield [19]. In Table III, we give routines of X11 which consists of hash functions below:

Blake, Bmw, Groestl, Jh, Keccak, Skein, Luffa, Cubehash, Shavite, Simd and Echo.

TABLE III. The routine of X11

Inputs: \mathcal{S}_0 (block header except for nonce) and L	
Step 1:	for nonce = 0, 1, 2, ... $2^{32} - 1$
Step 2:	$\mathbf{e} = \text{Blake}(\mathcal{S}_0 \cup \{\text{nonce}\})$
Step 3:	$\mathbf{e} = \text{Bmw}(\mathbf{e})$

Step 12:	$\mathbf{e} = \text{Echo}(\mathbf{e})$
Step 13:	If \mathbf{e} begins with L zero bits, then go to Step 15.
Step 14:	end
Step 15:	Block generation & broadcast

Blake first takes a given set of the current the block header with a selected nonce to get its hash. Next, Bmw takes this hash as its input to yield a hash. The same procedures are repeated until Echo, the last hash function, yields its hash. The decision is made using this last hash, as we have shown in Step 13.

However, the order of using the 11 hash functions is always fixed. This fixed order makes the development of ASIC mining devices possible and in fact an easy task. The development of ASIC mining solution can be done when these hash functions are implemented in a single device. Logic gates to sequentially connect the hash functions can be implemented. The first ASIC mining device targeting X11 was developed in 2016.

The idea behind X11 has been extended to other PoWs such as X13, X15, and X17. As the names suggest, they consists of 13, 15, and 17 hash functions, respectively. To date, there is a set of ASIC mining devices for both X13 and X15 while there is no ASIC mining device yet for X17.

E. PoW of Raven

In 2018, a new extension of X11 was proposed in [20]. This is called X16r. It uses multiple hash functions given in Table IV to get the last hash like to other extensions of X11 that we have discussed in Section II D. But, unlike the others, the sequence of the hash functions in X16r can be made to vary from block to block. This variation seems to be a role for preventing the advent of ASIC mining devices for X16r.

We provide an example to address how X16r operates. The sequence X16r is determined upon the last 16 bytes of a previous hash. Let this previous hash be

0x0000...04def2c3eff6da11542ffcdabce.

The last 16 bytes are 6da11542ffcdabce. Then, the sequence is decided on the basis of Table IV below:

Luffa → Shabal → Echo → Bmw → Bmw → Skein → Keccak → Groestl → Sha512 → Sha512 → Fugue → Shabal → Echo → Hamsi → Fugue → Whirlpool.

A miner puts a given set of the block header with a selected nonce through Luffa to yield a hash. Shabal takes this hash as its input to yield its hash. This routine is repeated until the last hash is yielded. In the above example, the last hash is obtained through Whirlpool.

At the time of writing this manuscript, it seems, no one has officially succeeded in implementing ASIC mining devices for X16r, i.e., there is no announced commercial product. However,

TABLE IV. The map for X16r

Value	Hash	Value	Hash
0	Blake	8	Shavite
1	Bmw	9	Simd
2	Groestl	a	Echo
3	Jh	b	Hamsi
4	Keccak	c	Fugue
5	Skein	d	Shabal
6	Luffa	e	Whirlpool
7	Cubehash	f	Sha512

Black and Weight, the developers of X16r, in [20] stated that reordering the sequence cannot make the development of ASIC mining devices impossible. Recently, in [33] at Nov., 2019, Whitefire990 reported a simulation result which indicates the probability of k time-repetition, such that the same hash function is used at least 5 times consecutively when k is 5, exponentially decreases in k . Insisted further is that k greater than 5 can be ignored in designing of ASIC mining devices. As such, what claimed there is that the ASIC-resistant property of Raven can be broken by the said ASIC designing method. It shall be noted, however, that all these claims provided in [33] have not yet been carefully verified through a peer-review system.

F. Short summary from C to E

From the subsection II.C to II.E, we have reviewed the existing ASIC-resistant PoWs categorized as follows:

- The usage of intentional memory access.
- The usage of multiple hash functions.

Ethash and ProgPoW can belong to the first class while X11 and its variants such as X13, X15, X17 and X16r can belong to the second class. The basic idea of the first class is to use the bottleneck intentionally caused by randomly fetching data from a memory. The basic idea of the second one is to use the multiple hash functions, which can make the development costs of ASIC mining devices expensive.

At the time of writing this manuscript, ASIC mining devices for Ethash, X11, X13 and X15 are available. The development of ProgPoW is not yet available. X17 and X16r can be resistant to ASIC mining devices. But, as the ASIC-resistant property of the PoWs such as X11, X13 and X15 are broken, that of X17 can be cracked in the near future when the hardware development technology is improved. As mentioned in the subsection II.E, there is a claim that the anti-ASIC property of X16r could be broken; but to date no commercial ASIC mining device has been announced.

III. Literature Surveys on LDPC

An LDPC decoder plays an important role in ECCPoW. That is, the decoder is utilized to generate an unpredictable random output that can be later on used to give a proof whether a puzzle is solved or not. We prepare this section to give a quick summary regarding the LDPC codes and their decoders.

In 1963, LDPC codes were proposed by Gallager in his thesis [7]. But, the codes received no attention back then because computers were not sufficiently fast enough to check the per-

formance of a decoder. Mackay and Neal [22] reported in 1997 that the codes could achieve the Shannon limit [21] closely with a message passing decoder that uses a kind of belief propagation algorithms. Since then, numerous studies on the codes have been made. They are categorized as follows: *i*) constructing the codes to approach the Shannon limit [21] and *ii*) implementing the fast decoders based on either ASIC [23]–[26] or field programmable gate array (FPGA) [27]–[28] to support real-time decoding purpose.

A. LDPC codes

LDPC codes can be generalized to a non-binary alphabet for improving its error-correction capability. But, for the purpose of using these codes in ECCPoW to provide a new time-varying anti-ASIC PoW system, it hence is suffice to consider the binary alphabet version only.

An (n, k) LDPC code is a linear code constructed by supplementing each message \mathbf{m} of size k with parity bits to get a codeword of size n . This code is often defined with respect to a parity check matrix \mathbf{H} of size $m \times n$ such that each element is binary either zero or one and the number of ones is very small, where m is the number of parity bits, i.e., $m = n - k$.

For a given parity check matrix \mathbf{H} , its corresponding LDPC code can be either *regular* or *irregular*. If \mathbf{H} contains a constant number w_c , called the column degree, of ones in each column and a constant number w_r , called the row degree, of 1s in each row, the code is called *regular*. For a given *regular* LDPC code, the parameters such as n , k , w_c , and w_r satisfy the following:

$$nw_c = (n - k)w_r = mw_r. \quad (1)$$

If \mathbf{H} contains a different number of 1s in both each column and each row, the code is called *irregular*. In the perspective of the error-correction capability, irregular codes are better than regular codes. To serve our purpose of anti-ASIC PoW mechanism, we aim to consider the *regular* LDPC codes because

- i.* it is much easier to implement a decoder of regular LDPC codes and
- ii.* the aim of using this decoder is not to correct errors but to yield an unpredictable random output.

A bipartite graph is often used to represent an LDPC code, as we have shown in Fig. 1. The lower and upper nodes are called the variable nodes and the check nodes, respectively. Each edge shows the adjacency of the i^{th} variable node and the j^{th} check node and corresponds to a nonzero $(i, j)^{\text{th}}$ element in \mathbf{H} .

For a given LDPC code, its error-correction capability relies on the minimum (Hamming) distance d . This distance is given by solving an optimization problem in which we consider any pair of $2^k - 1$ different codewords below:

$$d = \min_{\mathbf{u} \in \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_{2^k-1}\} \setminus \{\mathbf{0}_n\}} \|\mathbf{u}\|_h \quad (2)$$

which is NP-complete, where $\|\mathbf{x}\|_h$ is the number of 1s in \mathbf{x} .

Thus far, studies on the computation of a good approximation to minimum distance for a given fixed \mathbf{H} with reasonable size have been reported futile and as such it has remained as an

open problem. Keha and Duman [29] proposed a branch and cut algorithm to obtain the minimum distance of LDPC codes. But, this algorithm requires a large amount of time and memory; it is thus only useful if n is small. Then, Hashemi and Banihashemi [30] proposed a method to find lower and upper bounds of the minimum distance of LDPC codes and obtained both of the bounds even when $n > 64,000$.

For regular LDPC codes with a particular pair of w_c and w_r , upper and lower bounds for a relative minimum distance which is the ratio of the minimum distance d to the code length n , are given in [31] and [7], respectively. We use them for our purpose in this paper in Section IV. Once the minimum distance is given, the number of correctable errors can be computed as follows:

Theorem 1 [32]: Let a linear code be defined as a given parity check matrix \mathbf{H} which has the minimum distance d . Then, the number of correctable errors is

$$t = \lfloor (d-1)/2 \rfloor \quad (3)$$

where $\lfloor x \rfloor$ denotes the integer part of x .

We explain how to encode a message \mathbf{m} for a given \mathbf{H} of size $m \times n$. To this end, we build a generator matrix \mathbf{G} of size $n \times k$ whose column space is orthogonal to the row space of \mathbf{H} below:

Step 1: Conduct the Gaussian elimination to rewrite \mathbf{H} as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A}^T & \mathbf{I}_{n-k} \end{bmatrix}$$

where \mathbf{I}_{n-k} is the identity matrix of size $(n - k) \times (n - k)$.

Step 2: Form \mathbf{G} of size $n \times k$ as follows:

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_k \\ \mathbf{A} \end{bmatrix}.$$

It is noted that $\mathbf{H}\mathbf{G} = \mathbf{0}_{n-k,k}$ where $\mathbf{0}_{n-k,k}$ is the zero matrix of size $(n - k) \times k$. Note again $m = n - k$. The message \mathbf{m} is encoded to produce a codeword \mathbf{c} of size $n \times 1$ via $\mathbf{c} = \mathbf{G}\mathbf{m}$. Then, because of the definition of \mathbf{G} , it is always seen that the result of multiplying \mathbf{H} with \mathbf{c} is the zero vector of size m , i.e.,

$$\mathbf{H}\mathbf{c} = \mathbf{H}\mathbf{G}\mathbf{m} = \mathbf{0}_m.$$

A decoder takes both the parity check matrix \mathbf{H} and the corrupted word \mathbf{r} , which is $\mathbf{r} = \mathbf{c} + \mathbf{e}$, where \mathbf{e} is an error pattern. The decoder runs a message-passing algorithm [32] shown to be the standard decoding algorithm to remove \mathbf{e} .

The principle behind the message-passing algorithm is to iteratively propagate probabilistic information among the variable and check nodes. The iterations are terminated if either the number of iterations exceeds a given number or an output is a codeword. Detailed explanations on how this algorithm operates are provided in [32], i.e., Algorithm 5.1 on page 220. The algorithm takes parameters such as \mathbf{H} , \mathbf{r} , maxIter , and ε , where maxIter is the number of maximum iterations, and ε is the cross error probability that is used to determine the initial value of the algorithm.

The error-correction performance of the algorithm depends on both maxIter and the crossover error probability. If maxIter is small, the algorithm fails to obtain a converged solution. If it is large, the algorithm may take a considerably long computational time to obtain its solution. In the literature, maxIter is set from 10 to 20 in general. Next, the crossover error probability is set if the transition probability of a binary symmetric channel is either given or estimated. If this is improperly set, the decoding performance is degraded, leading to the poor error-correction capability. For the purpose in correcting errors, this parameter must be carefully considered.

The aim of using the decoder in ECCPoW, please make note of the fact that, is not to correct errors. Thus, there is no need to set maxIter and ε strictly. One condition that we shall aim to satisfy is that all the miners in ECCPoW system have to use the same values for these parameters. This condition can be easily satisfied by letting them to be published fixed constants in the proposed implemented program. As the miners verify a newly published block before accepting it, there is no benefit not to follow and use different ones for these published parameters. That is, any proof obtained from arbitrary parameters other than the published ones must be rejected.

We will give details on how to construct the other parameters, such as \mathbf{H} and \mathbf{r} , in Section IV.

B. FPGA and ASIC Implementation

LDPC decoders based on ASIC, a.k.a. ASIC-LDPC decoders are implemented to achieve low power consumption and fast processing. In the decoders, the check and variable nodes have to be physically connected using logical gates for a given parity check matrix. Fixed connections are to imply limited flexibility on the designs on the ASIC-LDPC decoders, making them only to support either a set of pre-defined parity check matrices or structured parity check matrices. We give our surveys related to existing ASIC-LDPC decoders as follows.

First, the ASIC-LDPC decoder in [23] supports quasi-cyclic parity check matrices decomposed into cyclic-shifted identity and zero matrices. These matrices have the same structure that is used in implementing the decoder. Second, the ASIC-LDPC decoder in [24] supports the parity check matrices included in the IEEE 802.16e system. These matrices are fixed; not change. Third, in [25], Hanzo *et al.*, reviewed the state-of-the-art of ASIC-LDPC decoders and stated that these decoders must take a bank of hardware to support many random parity check matrices. Namely, additional components such as memories, controllers and switchable interconnections are required, resulting in that these components occupy the most chip area in the decoders. They supported their statement by providing an example of [26] in which the ASIC-LDPC decoder supports about 100 parity check matrices, but its additional components occupy 75% of the total area of the decoder. This shows that there are no practical implementations on ASIC-LDPC decoders to support an infinite number of random parity check matrices.

There are FPGA-LDPC decoders that are the LDPC decoders implemented on FPGA chips. FPGA-LDPC decoders consume more power rather than ASIC-LDPC decoders do. But, it

is much easier to reprogram the FPGA-LDPC decoders, which implies that they can achieve the more flexibility on the designs compared to ASIC-LDPC decoders. The FPGA-LDPC decoder in [27] supports parity check matrices up to $n = 65,000$. But, it is required to load a parity check matrix onto this decoder when it has to be changed, requiring additional time. In [28], Hanzo *et al.*, stated that FPGA-LDPC decoders require additional routing and processing devices to support many parity check matrices. But, as they pointed, the use of these additional devices can lead to complex designs, increasing the cost of the decoders.

IV. ERROR-CORRECTION CODES PROOF OF WORK

In this section, we give details on ECCPoW. For simplicity, we organize this section into four subsections. In the first subsection, we list fields belonging to a block header of ECCPoW and provide their simple explanations. In the second subsection, we illustrate an overall structure of ECCPoW and its explanations. In the third subsection, we explain how we construct two inputs that appear as we use a decoder of error-correction codes. In the last subsection, we give the definition of an ECC puzzle generation function and present how to define an ECC puzzle using this ECC function. We end the last subsection by giving routines for solving this ECC puzzle.

A. Block header in ECCPoW

The block header of ECCPoW is defined to be a data structure that has eight fields such as timestamp, Merkle tree value, previous hash value, nonce, version, code length, row degree and column degree.

We use the fields such as version, timestamp, previous hash value, Merkle tree value and nonce to achieve in the purpose of guaranteeing the immutability property similar to Bitcoin.

We use the remains such as the code length, the row degree and the column degree to assign the size of hash vector and parity check matrix (PCM), which appears due to the usage of a decoder of a family of LDPC codes. As we will show in Section V, we change the difficulty level of a puzzle by varying the code length if the column degree and the row degree are fixed.

B. Overall structure of ECCPoW

Fig. 1 is prepared to present an overall structure of ECCPoW. This structure consists of three parts such as *i*) the hash vector generation (HVG) part, *ii*) the LDPC decoder part and *iii*) the decision part. We explain each part as follows.

In the HVG part, we randomly generate a hash vector of size n using a series of SHA256s taking the CBH with a given nonce generated by the nonce generator. The details how to generate this hash vector from the knowledge of the CBH will be given in the subsection IV.C.

In the LDPC decoder part, there is a decoder from a family of LDPC codes. This decoder takes the above hash vector and runs the message-passing algorithm [32] to yield a binary word \mathbf{c} . It is noted that this decoder takes a parity check matrix (PCM) \mathbf{H} determining the relation between an input and its corresponding output. The details how to construct this PCM will be given in the subsection IV.C.

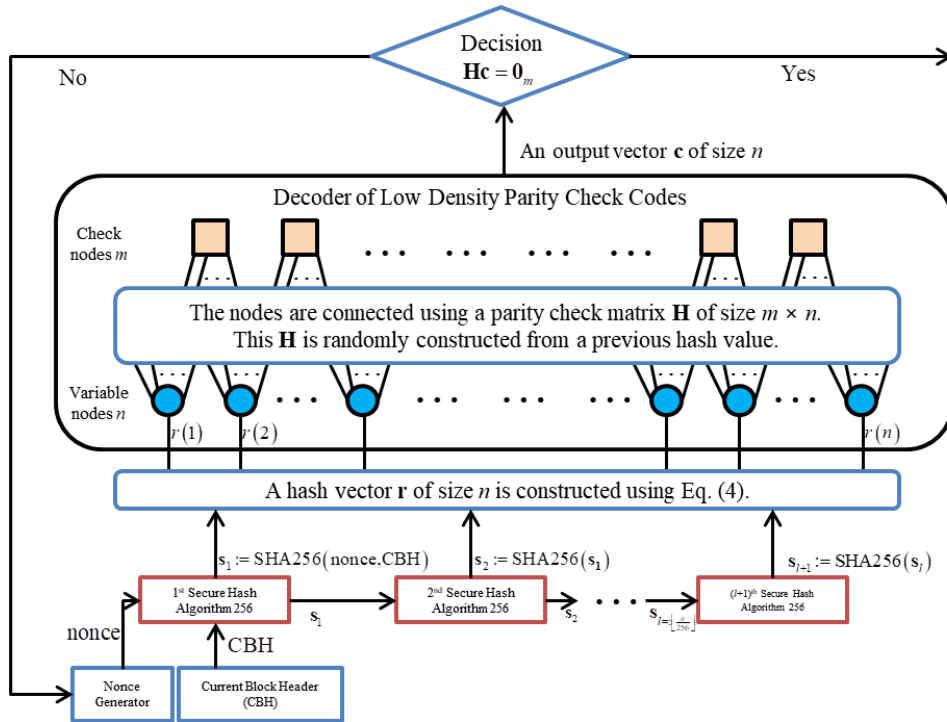


Fig. 1. An overall scheme of ECCPoW.

Last, in the decision part, the decision is made on the basis of the output provided by the decoder, as we have shown in Fig. 1.

C. Construction of hash vector and parity check matrix in ECCPoW

First, we provide the definition of hash vector \mathbf{r} and give how we construct this hash vector using the current block header.

Definition 1 – Hash Vector: A hash vector \mathbf{r} , which is a vector of concatenating outputs of SHA256s, of size n is defined to be:

$$\mathbf{r} := \begin{cases} \mathbf{s}_1[1:n] & \text{if } n \leq 256 \\ [\mathbf{s}_1 \ \cdots \ \mathbf{s}_l \ \mathbf{s}_{l+1}[1:j]] & \text{if } n > 256 \end{cases} \quad (4)$$

where $l = \lfloor n/256 \rfloor$, $j = n - 256 \times l$,

$$\mathbf{s}_1 := \text{SHA256}(\text{CBH}) \in \{0,1\}^{256} \quad (5)$$

and

$$\mathbf{s}_u := \text{SHA256}(\mathbf{s}_1) \in \{0,1\}^{256} \quad (6)$$

where $u = 2, 3, \dots, l+1$ and CBH is the current block header.

The current block header represented to be CBH in (5) is the on-chain information that is stored in the Internet. Anyone thus can access this on-chain information, leading to that anyone can make the same hash vector that a miner made during his mining competition work.

Second, we give a construction method of PCM for a given previous hash. This construction method has to be designed to satisfy two conditions:

C1. Any verifier can reconstruct the PCM using on-chain

information that the miner has used.

C2. Formation of a PCM can vary from block to block.

First, suppose that C1 is not met. One choice a miner can opt out is to include a constructed PCM in a block for making any verifier to check the validity of this block. This may result in, under the condition that the block size is fixed, reducing the number of transactions stored in the block. Second, suppose C2 is not met. We then remind the literature survey given in Section III. This survey is to indicate that developing ASIC-LDPC decoders for supporting a single PCM can be possible. Thus, if C2 is not satisfied, there is a possibility to develop ASIC mining devices. Thus, C1 and C2 must be satisfied simultaneously.

Now, we begin to explain the proposed construction method that can satisfy C1 and C2. First, we let you know that the PCM parameters such as the code length n , the row degree w_r and the column degree w_c are essentially required to construct a PCM \mathbf{H} . More concretely, if they are provided, we can assign the value of m , the number of rows of \mathbf{H} in (1). These parameters are included in the block header, as we have stated in the second paragraph in this section.

The proposed method is based on the method of Gallager [7]. This aims to construct \mathbf{H} that can be decomposed into a set of sub-matrices of size $w_c \times n$ as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} \\ \pi_1(\mathbf{A}) \\ \vdots \\ \pi_{w_c-1}(\mathbf{A}) \end{bmatrix} \in \{0,1\}_{\frac{mw_c}{w_r} \times n} \quad (7)$$

where $\pi_i(\mathbf{A})$ is the i^{th} submatrix constructed by random per-

mutation of the columns of \mathbf{A} , π_i is the i^{th} permutation order, and

$$\mathbf{A} := \begin{bmatrix} \mathbf{1}_{w_r} & & \\ & \ddots & \\ & & \mathbf{1}_{w_r} \end{bmatrix} \in \{0,1\}^{w_c \times n}$$

whose the i^{th} row has w_c 1s in a row from $(i-1) \times w_r$ to $i \times w_r$ and

$$\mathbf{1}_{w_r} := [1 \ 1 \ \cdots \ 1] \in \mathbf{1}^{1 \times w_r}.$$

Let w_c , w_r and n are fixed. Different PCM can be constructed by varying the permutation orders, leading to the serial creation of different PCMs. To come up with different random permutation from block to block, we use a previous hash. Namely, we let PBHV be assumed to be an array of 32 bytes to represent the given previous hash. We generate an initial seed value S below

$$S := \text{PBHV}[0] + \text{PBHV}[1] + \cdots + \text{PBHV}[31] \quad (8)$$

where $\text{PBHV}[i]$ is the i^{th} element of PBHV. The first permutation order is generated using a seed value S . The i^{th} permutation order is then generated using $S - i + 1$. The pseudo code of this construction method of PCM is given in Table VI.

We explain how the pseudo code given in Table VI can satisfy the conditions C1 and C2 mentioned earlier. In Step 4, the i^{th} permutation order is constructed using $S - i + 1$. Any verifier can get the same value S without any communications because PBHVs are on-chain information and thus available within the chain. Thus, the verifiers can easily reconstruct what the miner has constructed in the past, which confirms that the proposed method satisfies C1. Second, PBHVs are hashes and thus possess the characteristics of random numbers; i.e., the initial seed value is a random number. All of the permutation orders are provided using the initial seed value. Thus, the orders vary from block to block, which confirms that the proposed method satisfies C2 as well.

D. Construction of ECC puzzle generation functions and ECC puzzles

Let the current block header (CBH) except nonce be given. We define an ECC puzzle generation function by concatenating the HVG part and the decoder part.

Definition 3 – *ECC puzzle generation function*: Let the current block header (CBH) except nonce be given. For this given CBH, an ECC puzzle generation function (ECCPGF) is defined to be a composite function as follows:

$$h_{\text{CBH}} : \{\text{nonce}\} \mapsto \mathbf{c} \in \{0,1\}^{n \times 1} \quad (9)$$

where \mathbf{c} is the output of a decoder defined in Definition 4.

Definition 4 – *Decoder*: A decoder takes both the hash vector \mathbf{r} in (4) and the PCM \mathbf{H} in (7) as its inputs and runs the message passing algorithm given in [32] to yield a vector \mathbf{c} of size n :

$$\mathcal{D}_{\text{MP}} : \{\mathbf{r}, \mathbf{H}\} \mapsto \mathbf{c} \in \{0,1\}^{n \times 1}. \quad (10)$$

TABLE V. The pseudo codes for ECCPoW

Inputs: CBH and PCM \mathbf{H}	
Step 1:	A nonce is uniformly chosen from $[0, 2^{32} - 1]$
Step 2:	Construct a HV \mathbf{r} using (4) with a chosen nonce and the given CBH.
Step 3:	Obtain a vector \mathbf{c} using (10) with the given PCM \mathbf{H} .
Step 4:	If the constraint in (11) is satisfied, then go to Step 5.
Step 5:	Block generation & broadcast

TABLE VI. The pseudo codes to construct PCM

Inputs: n , w_c , w_r and BHV	
Output: \mathbf{H}	
Step 1:	Construct S using (8).
Step 2:	Construct \mathbf{A} by following the statements below (7) and $\mathbf{H} = \mathbf{A}$.
Step 3:	for $i = 2$ to $w_c - 1$
Step 4:	Construct π_i with the seed value $S - i + 1$.
Step 5:	$\mathbf{H} = [\mathbf{H} \ \pi_i(\mathbf{A})^T]$.
Step 6:	end
Step 7:	$\mathbf{H} = \mathbf{H}^T$.

When the CBH with a selected nonce is given, we can make a hash vector \mathbf{r} according to (4). Using the previous hash value included in the CBH, we can make a PCM \mathbf{H} as well. The decoder then takes both of them to yield the binary word, as we have shown in (10). This binary word is the output of ECCPGF, as we have shown in (9)

A mapping rule in the LDPC decoder depends on the form of \mathbf{H} . As we have discussed in the subsection IV.C, we can choose to vary \mathbf{H} from block to block. As blocks are mined endlessly, infinitely many PCMs can be made. Thus, as we have mentioned in the subsection III.B, this can deter the development of an ASIC-LDPC decoder. Making ASIC chips to function as an ECCPGF becomes extremely difficult, if not impossible. Thus, we use this ECCPGF to define an ECC puzzle as follows.

Definition 5 – *ECC Puzzle*: An ECC puzzle constructed using a given ECCPGF defined in (9) is defined to be

$$\text{find } \text{nonce} \text{ subject to } \mathbf{H}h_{\text{CBH}}(\text{nonce}) = \mathbf{0}_m. \quad (11)$$

Namely, this puzzle is a problem where we aim to find a nonce satisfying the constraint given in (11).

We shall note that this puzzle can be time-variant from block to block and resistant to ASIC chips. We will provide details in Section VI.

Last, we provide codes to solve a puzzle in Table V. In Step 1, a nonce is selected from 0 to $2^{32} - 1$. We construct a hash vector \mathbf{r} using (4), as we have shown in Step 2. In Step 3, we execute the decoder to give an output by taking both \mathbf{r} and \mathbf{H} . The decision is done using this output in Step 4. If the output is not a codeword, we repeat the routines from Step 1 to Step 4. Similar to Bitcoin, there is a case in which we cannot find the solution even we consider the whole nonces. In such a case, we modify the fields such as timestamp and Merkle tree value of the current block header. This modification can lead to the variation of the hash vector (4). We then repeat the whole routines with this modified block header.

V. ANALYSIS ON ERROR-CORRECTION CODES PROOF OF WORK

To solve an ECC puzzle, we repeat the routines from Step 1 to Step 4 many times until finding a nonce that satisfies the constraint in (11). A simple question such that what is the number of trials for finding this nonce naturally arises. To this end, we conduct a probabilistic study for giving answers to the following three questions:

- Q1. What is the number of hash cycles needed to solve a given puzzle?
- Q2. Which are the parameters which affect the number of hash cycles needed?
- Q3. How does the number of miners affect the number of hash cycles needed?

We define the hash cycle, the success event, and the mining game, respectively.

Definition 6 – Hash Cycle: The single execution of the whole routines from Step 1 to Step 4 given in Table V is defined to be single hash cycle.

Definition 7 – Success Event: A success event occurs if a nonce such that the decoder defined in (10) can return a codeword is found, i.e., the constraint in (11) is satisfied.

Definition 8 – Mining Game: Let both a PCM \mathbf{H} of size $m \times n$ and a CBH except for nonce be given. There are M miners, and each use a single computer with the same computing capacity. A mining game (MG)

$$\text{MG}\{\mathbf{H}, \text{CBH}, M, p\} \quad (12)$$

is defined that the miners compete with each other in a race to hit the success event first. We let p be the decoding success (DS) probability of the decoder for this given PCM \mathbf{H} , i.e.,

$$p := \Pr\{\mathbf{r} : \mathbf{H}\mathbf{c} = \mathbf{0}_m\} \quad (13)$$

where \mathbf{c} is the output of the decoder defined in (10).

For a given PCM \mathbf{H} of size $m \times n$, there are 2^k codewords:

$$\{\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \dots, \mathbf{c}_{2^k}\}.$$

Then, we define a sphere set for the given i^{th} codeword

$$\mathcal{A}(\mathbf{c}_i, l) \triangleq \{\mathbf{r} : \|\mathbf{r} - \mathbf{c}_i\|_h \leq l \cap \mathbf{r} \in \{0, 1\}^n\}$$

whose cardinality is

$$|\mathcal{A}(\mathbf{c}_i, l)| = \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{l} = \sum_{l=0}^s \binom{n}{l}$$

where s is a positive integer and $\|\mathbf{x}\|_h$ is the number of 1s in \mathbf{x} .

We assume that the decoder defined in (10) is optimal, implying that it can correct up to t errors where t is obtained by (3). The decoder always yields the i^{th} codeword when it takes an input belonging to the i^{th} sphere, i.e.,

$$\mathcal{D}_{MP} : \{\mathbf{r}, \mathbf{H}\} \mapsto \mathbf{c}_i$$

where $\mathbf{r} \in \mathcal{A}(\mathbf{c}_i, t)$. Then, we have

$$\begin{aligned} p &= \sum_{i=1}^{2^k} \Pr\{\mathbf{c} = \mathbf{c}_i\} = \sum_{i=1}^{2^k} \sum_{l=0}^t \Pr\{\|\mathbf{r} - \mathbf{c}_i\|_h = l\} \\ &= \sum_{i=1}^{2^k} \Pr\{\mathbf{r} \in \mathcal{A}(\mathbf{c}_i, t)\} \end{aligned} \quad (14)$$

where \mathbf{c} is an output of the decoder which takes a hash vector \mathbf{r} . Since the number of inputs that can be mapped into one of the codewords is

$$\sum_{i=1}^{2^k} |\mathcal{A}(\mathbf{c}_i, t)|,$$

the DS probability p can be expressed as follows:

$$p = 2^{-n} \sum_{i=1}^{2^k} |\mathcal{A}(\mathbf{c}_i, t)| = 2^{k-n} |\mathcal{A}(\mathbf{c}_i, t)| = 2^{k-n} \sum_{l=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{l} \quad (15)$$

where the third equality comes from (3).

Intuitively, the number of trials increases as p decreases. It is natural to find which parameters make effects on p . To this end, we establish Proposition 1 to provide the behavior of p in terms of the code length n under the assumption that the row degree and the column degree are fixed.

Proposition 1 – Let $w_c \geq 3$, $w_r > w_c$ be fixed constants and their ratio be a fixed constant as well

$$w_c / w_r =: \alpha \in (0, 1). \quad (16)$$

Let the size of a given PCM \mathbf{H} be $m \times n$. For any $0 < \delta < 1/2$, we have

$$2^{-n\alpha} \leq p \leq 2^{-n(\alpha - H(\delta/2))} \quad (17)$$

where $H(x)$ is the binary entropy function defined as follows:

$$H(x) = -x \log_2 x - (1-x) \log_2 (1-x).$$

Indeed, let the ratio further satisfy the following:

$$\alpha \in (H(0.25), 1). \quad (18)$$

Then, the DS probability p vanishes with increase in n .

Proof: From (1) with (16), we infer that

$$k - n = -n\alpha. \quad (19)$$

The results [7] state that the minimum distance of any regular LDPC code with constant w_r and $w_c \geq 3$ can linearly increase with increase in n . This statement indicates that the distance can be expressed as for any $0 < \delta < 1/2$,

$$d = \lfloor \delta n \rfloor.$$

Substituting (19) into (15) leads to an upper bound to p

$$p \leq 2^{k-n} \sum_{l=0}^{\lfloor \frac{\delta n}{2} \rfloor} \binom{n}{l} \leq 2^{k-n} 2^{nH(\frac{\lfloor \delta n \rfloor}{2n})} \leq 2^{-n(\alpha - H(\delta/2))} \quad (20)$$

where the second inequality comes from the fact that any integers $n \geq k \geq 1$ with $k/n \leq 0.5$

$$\sum_{l=0}^k \binom{n}{l} \leq 2^{nH(k/n)}.$$

By assuming $t = 0$ in (14), the lower bound is obtained below:

$$p = 2^{k-n} |\mathcal{A}(\mathbf{c}_i, t)| \geq 2^{k-n} |\mathcal{A}(\mathbf{c}_i, 0)| = 2^{k-n} = 2^{-n\alpha}.$$

Last, let the ratio satisfy (18). Then, the term of the exponent in the upper bound in (17) is negative because for any $0 < \delta < 1/2$,

$$H(\delta/2) < H(0.25).$$

Thus, increasing n makes the upper bound on p reduced. This leads to that p vanishes as n goes to infinity. ■

We provide quick discussions on Proposition 1. First, for the fixed constant α , decreasing the code length n makes the lower bound on the DS probability p grow. Second, if we select any pairs of w_c and w_r which can satisfy (18), then p vanishes as n increases. This means that no one solves an ECC puzzle using these pairs for sufficiently large n . We thus have to avoid selecting such pairs for preventing this critical problem.

We define a random variable to represent the number of hash cycles required to end a MG and provide its statistical properties, respectively.

Definition 8 – Random Variable: For a given MG, X_M is defined as a random variable that represents the number of hash cycles to end this given MG, where the subscript M denotes the number of miners forging simultaneously and independently. We call this random variable First Success Hash Cycle (FSHC).

Theorem 2 – Let an MG $\{\mathbf{H}, \text{CBH}, M, p\}$ be given. Then the distribution that FSHC occurs at the l^{th} hash cycle is

$$\Pr\{X_M = l\} = p_{\text{f.a.}}^{l-1} (1 - p_{\text{f.a.}})$$

which is a geometric distribution with a parameter

$$(1 - p_{\text{f.a.}}) \quad (21)$$

where $p_{\text{f.a.}} := (1 - p)^M$ is the probability that all of the miners fail to succeed in solving a given puzzle. Then, we have

$$\mathbb{E}[X_M] = (1 - p_{\text{f.a.}})^{-1} \quad (22)$$

and

$$\mathbb{V}[X_M] = p_{\text{f.a.}} (1 - p_{\text{f.a.}})^{-2}.$$

Proof: The proof is clear, as X_M follows the geometric distribution with (21). We thus omit it. ■

For the constant α defined in (16), Proposition 1 shows that the DS probability p grows with decrease in the code length n . When p grows, the parameter (21) reduces and converges to a real positive number. Thus, the expected value gets reduced and

TABLE VII. The lower bound given in (23)

$w_c = 4$ and $w_r = 5$	$M = 1$	$M = 5$	$M = 20$
Lower bounds, i.e., $\delta_1 = 0.3238$ in (23)			
$n = 80, k = 12$	1.58×10^4	0.31×10^4	0.08×10^4
$n = 120, k = 24$	6.03×10^7	1.20×10^7	0.30×10^7
$n = 160, k = 32$	2.46×10^9	0.49×10^9	0.12×10^9

converges to a number, implying that an ECC puzzle becomes easier as n gets smaller.

Next, we consider a case in which n grows. In this case, the upper bound given in (17) is too loose to be considered unless α satisfies (18). Thus, we require another upper bound on p which is tighter than the previous upper bound. To this end, we invoke a table of [31]. For a certain pair of the column degree w_c and the row degree w_r , this table was obtained and given in the form of an upper bound δ_1 to the relative minimum distance δ_0 , the ratio of the minimum distance d to the code length n . For $w_c = 4$ and $w_r = 5$, for example, the upper bound is 0.3238. For $w_c = 4$ and $w_r = 8$, it is 0.1765. The result was obtained from an asymptotic analysis; i.e., by letting n go to infinity.

We use the upper bound δ_1 to obtain a tighter bound on the DS probability p as follows:

$$p \leq g(n, k, \delta_1) \quad (22)$$

where δ_1 is given in the table of [31] for a certain pair of w_c and w_r and

$$g(n, k, \delta) := 2^{k-n} \sum_{l=0}^{\lfloor \frac{n\delta-1}{2} \rfloor} \binom{n}{l}.$$

The bound in (22) is obtained by simply replacing the minimum distance with the upper bound, respectively:

$$d := n\delta_0 \leq \lfloor n\delta_1 \rfloor.$$

Once the upper bound on p has been obtained, we can use it to find a lower bound to the expected value of FSHC below:

$$\frac{1}{1 - (1 - g(n, k, \delta_1))^M} \leq \mathbb{E}[X_M] \quad (23)$$

We can examine the behavior of (22) with respect to the code length n and the number of miners M . In Table VII, we provide the lower bounds to the expected value by varying either n or M . They are obtained for a case in which the column degree w_c and the row degree w_r are 4 and 5, respectively. We can see that the lower bound increases with increase in n for the fixed M . That is, it increases from 1.58×10^4 to 2.46×10^4 when n is increased from 80 to 160. This result implies that increasing n makes an ECC puzzle more difficult to solve. For the other pairs of w_c and w_r given in [31], the same result is observed.

Now, for the fixed code length n , we consider the behavior of the expected value of FSHC by growing the number of miners M . Intuitively, as more miners are involved in solving a given puzzle, this puzzle has to end early. In addition, if an infinite number of miners work, any MG has to end at the 1st hash cycle. These intuitions can be confirmed by Corollary 1 given below.

Corollary 1: Let a MG $\{\mathbf{H}, \text{CBH}, M, p\}$ be given. The expected value of a FSHC given in (22) decreases with increase in the number of miners M . In particular, this value can converge to 1 as M goes to infinity.

Proof: It is immediately seen that

$$\frac{d\mathbb{E}[X_M]}{dM} = -\frac{\log p_{f.a}}{p_{f.a}} \leq 0$$

implying that the expected value given in (22) is a decreasing function of M . As M goes to infinity, the parameter defined in (21) goes to one. Thus, the expected value converges to one. ■

The decoding process, i.e., Step 3 in Table V, can occupy the most computational time in a single hash cycle. In this decoding process, matrix-vector products are required, implying that the computational cost to run a single hash cycle can be modeled as $O(mn)$. Each miner uses the same single computer in a given MG. Thus, we can assume that each miner only runs τ operations per second. This assumption makes us define an expected value of a block generation time as follows.

Definition 10 – Block Generation Time: A MG $\{\mathbf{H}, \text{CBH}, M, p\}$ is given. Each miner is assumed to run τ operations per second. Then, the block generation time T can be defined as

$$T := \tau^{-1} \mathbb{E}[X_M] O(mn). \quad (24)$$

Both Proposition 1 and Theorem 2 can indicate that the expected value in (24) is an increasing function of the code length n . We thus immediately conclude that the block generation time T is an increase function of n .

VI. DISCUSSIONS ON ECCPoW

We prepare this section to provide plentiful discussions on ECCPoW. To this end, we begin to define general properties of PoWs and explain how ECCPoW can have them. We introduce a new property that ECCPoW can only have. This new property makes ECCPoW become a solution to the problems such as M1 and M2 which we have stated in Section I.

We now begin to define properties below.

- P1. A puzzle has to be time-consuming, but it is easy to check whether a given solution is correct or not.
- P2. Any previous solution cannot be used to find a current solution.
- P3. A puzzle can be solved with overwhelming probability 1 if and only if miners follow the routines of PoW.
- P4. The difficulty of a puzzle can change.
- P5. A puzzle can be time-variant from block to block.

The existing PoWs have the properties from P1 to P4. Let us begin to consider how bitcoin satisfies them. First, each puzzle is expected to be solved per 10 minutes. In contrast, validating a given solution can be instantly done. Second, SHA256 takes the block header to get its hash. The block header in Bitcoin has the timestamp field. Due to this timestamp field, the contexts of the current block header can be different to those of any previously mined blocks. Thus, any solutions given in these mined blocks

are useless in finding a current solution. Thus, P2 holds. Third, the number of possible hashes is 2^{256} while the number of solutions is about $2^{(256-L)}$ where L is a pre-defined value according to the difficulty of the provided block header. Thus, a possibility that a provided nonce is a solution is 2^L . In the 567,657th block of bitcoin, L is 72. This is to imply that for this block, the probability that a randomly given nonce can be a solution is 2^{-72} . Thus, all the miners have to follow the routines in Table I to solve a puzzle, meaning P3. Next, whenever 2016 blocks are mined, the difficulty periodically changes, implying P4. Last, existing PoW does not hold P5. The reasons for this issue will be given after explaining how ECCPoW satisfies P5.

Now, we begin to explain how ECCPoW can satisfy all the properties using the results established in the previous section.

Corollary 2: ECCPoW can satisfy the first property P1.

Proof: For a given nonce, we complete the verification whether this nonce is a solution or not by conducting the steps from Step 1 to Step 4 in Table V. This verification requires a single construction of a hash vector and a single execution of the decoder (10). In contrast, to solve an ECC puzzle, one has to do work by repeating them many times, as we have stated in Section V. ■

Corollary 3: ECCPoW can satisfy the second property P2.

Proof: As we have stated in Definition 5, we construct an ECC puzzle using an ECCPGF in Definition 3. This ECCPGF takes a hash vector as its input and produces a binary word as its output. The decision is made based on this binary word. To make this hash vector, we put a given set of the block header with a nonce through SHA256. Similar to Bitcoin block header, our block header includes the timestamp field. Thereby, one cannot effortlessly create a particular hash vector even if one has the full knowledge of all the previous solutions, i.e., all the collection of the nonces each of which was a solution to mine a block in the past. This makes ECCPoW satisfy P2. ■

Let us assume that there exists a knowledgeable miner who can solve ECC puzzles by referring to an input-output mapping table of ECCPoW without actually carrying out the decoding work. This miner can then mine blocks much faster than other honest miners can. To show the non-existence of such a miner in ECCPoW, we prepare Corollary 4. This corollary is to imply that all miners cannot but have to run the decoder to solve ECC puzzles.

Corollary 4: ECCPoW satisfies the third property P3.

Proof: Such a malicious miner can appear under an assumption that this miner has a mapping table that maps a given input to its corresponding output without actually carrying out the decoding work (10). That is, he puts each hash vector into the decoder to find what this decoder returns. He then uses this known information to construct the mapping table. However, the number of hash vectors is 2^n . For sufficiently large n , i.e. $2^{256} \sim 10^{77}$ for $n = 256$, we can easily see that this construction is impossible. Besides, this table depends on a PCM \mathbf{H} , meaning that the mapping table is updated newly whenever \mathbf{H} varies. As we have stated in the subsection IV.C, \mathbf{H} can be set to vary from block to block. Thereby, whenever he aims to solve a new ECC puzzle, he has to construct a new mapping table. The above

assumption there exists the mapping table, we thus conclude, is invalid. Thereby, any miners in ECCPoW cannot but have to obey the routines in Table V to solve puzzles; this leads to the conclusion that ECCPoW satisfies P3. ■

Now, we remind the discussions regarding Theorem 2. In the discussions, we have proven that an ECC puzzle can become easier as the code length n gets smaller. We then have empirically shown that the increase in n makes this puzzle more difficult to solve. They are to indicate that ECCPoW holds P4.

Corollary 5: ECCPoW can satisfy the fourth property P4.

Proof: It is clear; we thus omit it. ■

A PCM \mathbf{H} defines a mapping function of the decoder in (10). This decoder is used to define an ECCPGF in (9) which is used to construct an ECC puzzle. We conclude that this ECC puzzle can be a function of \mathbf{H} . As we have mentioned in the subsection IV.C, we construct \mathbf{H} using a previous hash, i.e., the i^{th} PCM is made using a hash of the $(i-1)^{\text{th}}$ block. Such a construction can make \mathbf{H} time-variant from block to block, leading to that the ECC puzzle can also be time-variant from block to block. This discussion leads to a conclusion that ECCPoW holds P5 below.

Corollary 6: ECCPoW can satisfy the fifth property P5.

Proof: It is clear; we thus omit it. ■

We now prepare to provide discussions how ECCPoW can be a solution to the problems such as *i*) M1 the re-centralization of the mining markets and *ii*) M2 the huge energy consumption for mining. As miners mine new blocks continuously, an infinite number of PCMs are constructed. Hence, we cannot count how many the number of PCMs is required in advance. Also, we cannot expect what PCMs will be generated. We remind the example given in [25], showing that there is no ASIC-LDPC decoder to support the infinite number of PCMs. By combining this example with the fact that there is an infinite number of PCMs are required, we thus conclude that the LDPC decoder in ECCPoW is operated by either graphical processing units or CPUs. This is the fundamental reason that ECCPoW becomes the solution remediating the aforementioned problems M1 and M2 which we have stated in Section I. This is the most remarkable contribution of ECCPoW.

We end this section by providing a reason that SHA based PoW does not hold P5. We consider SHA256 as a puzzle generation function (PGF) in Bitcoin because it is used to generate a puzzle. This PGF is fixed regardless of the height of a block that miners aim to mine. The puzzle made using a fixed PGF is not time-variant at all, leading to that such a puzzle cannot hold P5.

VII. CONCLUSIONS

PoW is fundamental to public blockchains, as it can be used to prohibit an unauthorized modification of mined blocks. For existing PoWs, ASIC mining devices have been introduced and used to mine blocks. The usage of such devices can cause the problems such as M1 and M2 that we have stated in Section I.

In this paper, as a solution to these problems, we proposed a new proof-of-work using error-correction codes which we call

ECCPoW. To the best of our knowledge, this is the first study in which a decoder of LDPC codes is applied to the consensus part of blockchain. Specifically, we combined this decoder with SHA256 to construct a composite function named as ECCPGF (9). We used ECCPGF to define a corresponding ECC puzzle (11) and provided the routines to solve a given ECC puzzle.

We also studied the behavior of the expected value of the number of hash cycles for solving a given ECC puzzle. We showed that this value can be either increased or decreased as varying the code length, the size of a hash vector taken by the decoder, and the number of miners. Indeed, we discussed how ECCPoW can satisfy the five properties defined in Section VI, which shows the value of ECCPoW as a general PoW.

As we have reviewed in Section III, there is no ASIC decoder to support an infinite number of LDPC codes. By motivated this survey, we intended to vary the codes from block to block. As a result, we made ECCPGF time-variant, meaning that its mapping function can vary from block to block. This leads to the time-variant property P5 defined in Section VI. This is the most innovative part of ECCPoW in repressing the advent of ASICs, implying that the problems caused from the usage of ASICs can be solved using our ECCPoW.

We have implemented ECCPoW and forked two blockchains such as Bitcoin and Ethereum by replacing their consensus with the implemented ECCPoW. We name these forked versions as BTC-ECC and ETH-ECC, respectively. All of the source codes including ECCPoW, BTC-ECC and ETH-ECC can be available in a GitHub site [34]. We have also provided manuals that shows how to install them, how to compile them and how to run. We believe that anyone can easily operate their own BTC-ECC or ETH-ECC by following these manuals. We believe that this site can be the first repository that makes people in the error-correction codes community get involved in the blockchain community.

REFERENCES

- [1] U. W. Chohan, "The double spending problem and cryptocurrencies", SSRN Electronic Journal, Jan. 2019
- [2] Nakamoto, S. *Bitcoin: A Peer-to-Peer Electronic Cash System*. Retrieved from: <https://bitcoin.org/bitcoin.pdf>
- [3] Quynh H. Dang, "Secure Hash Standard FIPS PUB 180-4," U.S. Dept. of Commerce and NIST, Aug. 4th 2015. <https://www.nist.gov/publications/secure-hash-standard>, Accessed Sept. 25th, 2018.
- [4] Jordan Tuwiner, "Bitcoin mining hardware ASICs," <https://www.buybitcoinworldwide.com/mining/hardware/>, June 30th, 2018. Accessed Sept. 26th, 2018.
- [5] A. Gervais, G. Karame, V. Capkun and S. Capkun, "Is bitcoin a decentralized currency?," *IEEE Security and Privacy Magazine*, Vol. 12, No. 3, pp. 54 – 60, May, 2014
- [6] Digiconomist. (2018). *Bitcoin Energy Consumption Index*. Retrieved April 3, 2018 from: <https://digiconomist.net/bitcoin-energy-consumption>
- [7] R. G. Gallager, *Low Density Parity Check Codes*, Monograph, M.I.T. Press, 1963
- [8] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory", DSN Progress Report, pp. 114-116, Jan.-Feb. 1978
- [9] D. J. Bernstein, T. Lange and C. Peters, "Wild McEliece", in *Proc. Sel. Areas Cryptography*, vol. 6544, pp. 143 – 158, 2011
- [10] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal and D. Schipani, "Using LDGM codes and sparse syndromes to achieve digital signatures", in *Post-Quantum Cryptography 2013, ser. Lecture Notes in Comput. Sci.*, vol. 7931, Springer, pp. 1 – 15, 2013.

- [11] Marco Baldi, “LDPC codes in the McEliece cryptosystem: attacks and countermeasures”, NATO Science for Peace and Security – D: Information and Communication Security, vol. 23, pp. 160 – 174,
- [12] M. Baldi, M. Bodrato and F. Chiaraluca, “A new analysis of the McEliece cryptosystem based on QC-LDPC codes”, in 6th Int. Conf. on Sec. and Cryptography for Networks. Springer, pp. 246 – 262, 2008
- [13] T. P. Berger, P.-L. Cayrel, P. Gaborit and A. Otmani, “Reducing key length of the McEliece cryptosystem”, in Progress in Cryptology, Africacrypt’2009, ser. LNCS, vol. 5580. Springer, pp. 77 – 97, 2009
- [14] V. M. Sidelnikov, “A public-key cryptosystem based on binary Reed–Muller codes,” Discrete Mathematics and Applications, vol. 4 no. 3, 1994.
- [15] L. R. Knudsen and B. Preneel, “Construction of secure and fast hash functions using nonbinary error correcting codes”, IEEE Transactions on Information Theory, vol. 48, no. 9, pp. 2254 – 2539, Sept., 2002.
- [16] S. Ermon, C. P. Gomes, A. Sabharwal and B. Selman, “Low-Density Parity Constraints for Hasing-Based Discrete Integration”, Proceedings of the 31st International Conference on Machine Learning, PMLR 32(1):271-279, 2014.
- [17] Vadhan, S., *Pseudorandomness*. Foundations and Trends in Theoretical Computer Science, 2011.
- [18] Vitalik Buterin. Dagger: *A memory-hard to compute, memory-easy to verify scrypt alternative*, Tech Report, hashcash.org website, 2013.
- [19] <https://github.com/dashpay/dash/wiki/Whitepaper>
- [20] T. Black and J. Weight, X16R ASIC resistant by design, 2018. <https://ravencoin.org/assets/documents/X16R-Whitepaper.pdf>
- [21] C. E. Shannon, “A mathematical theory of communication”, The Bell System Technical Journal, Vol. 27, 99. 379 – 423, 623 – 656, Jul., Oct., 1948
- [22] D. J. C. Mackay and R. M. Neal, “Near Shannon limit performance of low density parity check codes”, Electronics Letters, Vol. 33, No. 6, pp. 457 – 458, 1997
- [23] M. Awais and C. Condo, “Flexible LPDC decoder architectures”, VLSI Design, Vol. 2012, Doi: 10.1155/2012/730835
- [24] C-H. Liu, S-W. Yen, C-L. Chen, H-C. Chang, C-Y. Lee and S-J. Jou, “An LDPC decoder chip based on self-routing network for IEEE 802.16e Applications” IEEE Journal of Solid-State Circuits, Vol. 43, No. 3, pp. 684 – 694, Mar. 2008
- [25] S. Shao, P. Hailes, T-Y. Wang, J-Y. Wu, R. G. Maunder, B. M Al-Hashimi and L. Hanzo, “Survey of Turbo, LDPC and Polar Decoder ASIC Implementation”, IEEE Communications Surveys & Tutorials 2019
- [26] Y. L. Ueng, B. J. Yang, C. J. Yang, H. C. Lee, and J. D. Yang, “An Efficient Multi-Standard LDPC Decoder Design Using Hardware-Friendly Shuffled Decoding,” IEEE Trans. Circuits Syst. I, vol. 60, no. 3, pp. 743–756, March 2013.
- [27] C. Beuschel and H-J. Pfleiderer, “FPGA implantation of a flexible decoder for long LDPC codes”, Proc. Int. Conf. Field Program. Logic Appl., Sep. 2008, pp. 185–190
- [28] P. Hailes, R. G. Maunder and L. Hanzo, “A survey of FPGA-based LDPC decoders”, IEEE Communications Surveys & Tutorials, Dec. 2015.
- [29] A. B. Keha and T. M. Duman, “Minimum distance computation of LDPC codes using a branch and cut algorithm”, IEEE Transactions on Communication, Vol. 58, No. 4, pp. 1072 – 1079, Apr., 2010
- [30] Y. Hashemi and A. H. Banihashemi, “Tight Lower and Upper Bounds on the Minimum Distance of LDPC Codes”, IEEE Communications Letters, Vol. 22, No. 1, pp. 33 – 36, Jan., 2018
- [31] Y. Ben-Haim and S. Litsyn, “Upper bounds on the rate of LDPC codes as a function of minimum distance”, IEEE Transactions on Information Theory, vol. 52, no. 5, pp. 2092 – 2100, May., 2006
- [32] W. E. Ryan and S. Lin, *Channel Codes Classical and modern*, Cambridge, 2009
- [33] Whitefire990, https://drive.google.com/file/d/14ZYwpI-t6rnQ3I_SEZHvDwe2yQUnTJ5/view, Nov., 2019
- [34] DeSecure Blockchain, <https://github.com/cryptoecc/ETH-ECC>, 2020

Article

Profitable Double-Spending Attacks

Jehyuk Jang  and Heung-No Lee * 

School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology (GIST), 123 Cheomdangwagi-ro, Buk-gu, Gwangju 61005, Korea; jjh2014@gist.ac.kr

* Correspondence: heungno@gist.ac.kr

Received: 9 November 2020; Accepted: 25 November 2020; Published: 27 November 2020;
Corrected: 15 September 2022



Abstract: Our aim in this paper is to investigate the profitability of double-spending (DS) attacks that manipulate an a priori mined transaction in a blockchain. It was well understood that a successful DS attack is established when the proportion of computing power an attacker possesses is higher than that of the honest network. What is not yet well understood is how threatening a DS attack with less than 50% computing power used can be. Namely, DS attacks at any proportion can be a threat as long as the chance to make a good profit exists. Profit is obtained when the revenue from making a successful DS attack is greater than the cost of carrying out one. We have developed a novel probability theory for calculating a *finite time* attack probability. This can be used to size up attack resources needed to obtain the profit. The results enable us to derive a sufficient and necessary condition on the value of a transaction targeted by a DS attack. Our result is quite surprising: we theoretically show how a DS attack at any proportion of computing power can be made profitable. Given one's transaction value, the results can also be used to assess the risk of a DS attack. An example of profitable DS attack against BitcoinCash is provided.

Keywords: blockchain; double-spending attack; Fraud risk analysis; profitability; time-finite analysis; probability distribution; combinatorics

1. Introduction

A blockchain is a distributed ledger which has originated from the desire to find a novel alternative to centralized ledgers such as transactions through third parties [1]. Besides the role as a ledger, blockchains have been applied to many areas, e.g., managing the access authority to shared data in the cloud network [2] and averting collusion in e-Auction [3]. In a blockchain network based on the proof-of-work (PoW) mechanism, each miner verifies transactions and tries to put them into a block and mold the block to an existing chain by solving a cryptographic puzzle. This series of processes is called mining. However, the success of mining a block is given to only a single miner who solves the cryptographic puzzle for the first time. The reward of minting a certain amount of coins to the winner motivates more miners to join and remain in the network. As a result, blockchains have been designed so that the validity of transactions is confirmed by a lot of decentralized miners in the network.

A consensus mechanism is programmed for decentralized peers in a network to share a common chain. If a full-node succeeds in generating a new block, it has the latest version of the chain. All of the nodes in the network continuously communicate with each other to share the latest chain. A node may run into a situation in which it encounters mutually different chains more than one. In such a case, it utilizes a consensus rule with which it selects a single chain. Satoshi Nakamoto suggested the longest chain consensus for Bitcoin protocol in which the node selects the longest chain among all competing chains [1]. There are also other consensus rules [4,5], but a common goal of consensus rules is to select the single chain by which the most computation resources have been consumed based on the belief that it may have been verified by the largest number of miners.

A double-spending (DS) attack aims to double-spend a cryptocurrency for the worth of which a corresponding delivery of goods or services has already been completed. The records of payment are written in transactions and shared in a network via the status-quo chain. Thus, to double spend, attackers need to replace the status-quo chain in the network with their new one, after taking the goods or services. For example, under the longest chain consensus, this attack will be possible if an attacker builds a longer chain than the status-quo. Nakamoto [1] and Rosenfeld [6] have shown that the higher computing power is employed, the higher probability to make a DS attack successful is. In addition, if an attacker invests more computing power than that invested by a network, a success of DS attack is guaranteed. Such attacks are called the 51% attack.

In the last few years, unfortunately, blockchain networks have been recentralized [7,8], which make them vulnerable to DS attacks. To increase the chance of mining blocks, some nodes may form a pool of computing chips. The problem arises when a limited number of pools occupy a major proportion of the computing power in the network. For example, the pie chart (date accessed from [BTC.com](https://www.btc.com) on November 24, 2020) shown in Figure 1 illustrates the proportion of computing power in the Bitcoin network as of January 2020. In the chart, five pools such as F2Pool, [BTC.com](https://www.btc.com), Poolin, and Huobi.pool occupy more than 50% of the total computing power of Bitcoin. In a recentralized network, since most computing resources are concentrated on a small number of pools, it could be not difficult for them to conspire to alter the block content for their own benefits, if aiming to double-spend. Indeed, there have been a number of reports in 2018 and 2019 in which cryptocurrencies such as Verge, BitcoinGold, Ethereum Classic, Feathercoin, and Vertcoin suffered from DS attacks and millions of US dollars have been lost [9].

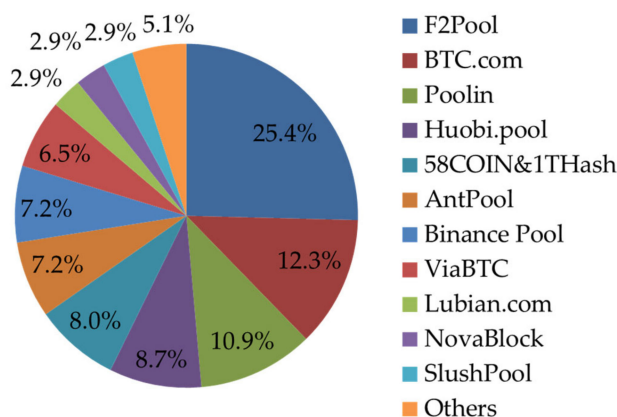


Figure 1. Computation power distribution among the largest mining pools.

In addition to the recentralization, the advent of rental services which lend the computing resources can be a concern as well [10]. Rental services such as nicehash.com which provide a brokerage service between the suppliers and the consumers have indeed become available. The rental service can be misused for making DS attacks easier. The presence of such computing resource rental services significantly reduce the cost of making a profit from double spending. This is because renting a required computing power for a few hours is much cheaper than building such a computing network. Indeed, nicehash.com attracts DS attackers to use their service by posting one-hour fees for renting 51% of the total computing power against dozens of blockchain networks on their website crypto51.app (accessed on 26 November 2020).

Success by making DS attacks is possible but is believed to be difficult for a public blockchain with a large pool of mining network support. By the results in [1,6], 51% attack has been considered as the requirement for a successful DS attack [11]. This conclusion however shall be reconsidered given our result in the sequel that there are significant chances of making a good profit from DS attacks

regardless of the proportion of computing power. The problem to consider, therefore, is to analyze the profitability of such attacks.

The analysis of attack profitability requires the ability to predict the time an attack will take, since the profit would be a function of time. Studies in [12–20] provided DS attack profitability analyses, but their time predictions were not accurate. Specifically, to make the time prediction easier, they either added impractical assumptions to the DS attack model defined by Nakamoto [1] and Rosenfeld [6] or oversimplified the time prediction formula (see Section 6 for details). Whereas, we follow the definition of DS attack in [1,6], and therefore we need to develop a new set of mathematical tools for precise analysis of attack profitability that we aim to report in this paper.

1.1. Contributions

We study the profitability of DS attacks. The concept of *cut-time* is introduced. Cut-time is defined to be the duration of time, from the start time to the end time of an attack. For each DS attempt, the attacker needs to pay for the cost to run his mining rig. A rational attacker would not, therefore, continue an attack indefinitely especially when operating within the regime of less than 50% computing power. To reduce the cost, the attacker needs to figure out how his attack success probability rolls out to be as the time progresses. We define that a DS attack is profitable if and only if the expected profit, the difference between revenue and cost (see Equation (33)), is positive. Our contributions are summarized into two folds:

First, we theoretically show that DS attacks can be profitable not only in the regime of 51% attack but also in the sub-50% regime where the computing power invested by the attacker is smaller than that invested by the target network. Specifically, a sufficient and necessary condition is derived for profitable DS attacks on the minimum value of target transaction. In the sub-50% regime, we also show that profitable DS attacks necessitate setting a finite cut-time.

Second, we derive novel mathematical results that are useful for an analysis of the attack success time. Specifically, the probability distribution function and the first moment expectation of the attack success time have been derived. They enable us to estimate the expected profit of a DS attack for a given cut-time. All mathematical results are numerically-calculable. All numerical examples of the theoretical results given in this paper are reproducible in our web-site (<https://codeocean.com/capsule/2308305/tree>).

1.2. Organization of the Paper

In Section 2, we define DS attack scenario and sufficient and necessary conditions required for successful DS attacks. Also, we define random variables that are useful in analyzing the attack profits. Section 3 comprises the analytic results of stochastics of the time-finite attack success. In Section 4, we define the profit function of DS attacks, followed by new theoretical results about the conditions for making them profitable. In Section 5, an example analysis of DS attack profitability in sub-50% regime against BitcoinCash network is given. Section 6 compares our results with related works. Finally, Section 7 concludes the paper with a summary.

2. The Attack Model

We define DS attack that we consider throughout this paper. We also define DS attack achieving (DSA) time, which is the least time spent for an occurrence of double-spending. The DSA time is a random variable derived from a random walk of Poisson counting processes (PCP).

2.1. Attack Scenario

We extend a DS attack scenario which has been considered by Nakamoto [1] and Rosenfeld [6]. Specifically, we add a time-finite attack scenario. There are two groups of miners, the normal group of honest miners and a single attacker. The normal group tends the honest chain.

When the attacker decides to launch a DS attack, he/she makes a target transaction for the payment of goods or services. In the target transaction, the transfer of cryptocurrency ownership from the attacker to a victim is written. We denote $t = 0$ as the time at which the last block of the honest chain has been generated. At time $t = 0$, the attacker announces the target transaction to normal group so that normal group starts to put it into the honest chain. At the same time $t = 0$, the attacker makes a fork of the honest chain which stems from the last block and builds it in secret. We refer to this secret fork as fraudulent chain. In the fraudulent chain, a fraudulent transaction is contained which alters the target transaction in a way that deceives the victim and benefits the attacker.

Before shipping goods or providing services to the attacker, the victim will obviously choose to wait for a few more blocks on the honest chain in addition to the block on which the his/her transaction has been entered, i.e., so-called block confirmation. Karame et al. [21] showed the importance of block confirmation: attackers are able to double-spend against zero block-confirmation even without mining a single block on the fraudulent chain at all. The number of blocks the victim chooses to wait for is referred to as the block confirmation number $N_{BC} \in \mathbb{N}$, which includes the block on which the target transaction is entered.

The attacker chooses to make the fraudulent chain public if his/her attack was successful. An attack is successful if the fraudulent chain is longer than the honest chain after the moment the block confirmation is satisfied. We define two necessary conditions $\mathcal{G}^{(1)}$, $\mathcal{G}^{(2)}$, for a success of DS attack:

Definition 1. A DS attack succeeds only if there exists a DS attack achieving (DSA) time $T_{DSA} \in (0, \infty)$ such that

1. $\mathcal{G}^{(1)}$: (block confirmation) the length of the honest chain for the duration of time T_{DSA} has grown greater than or equal to N_{BC} , and
2. $\mathcal{G}^{(2)}$: (success in PoW competition) the length of the fraudulent chain for the duration of time T_{DSA} has grown longer than that of the honest chain.

Rational attackers will not wait for his success indefinitely since growing the attacker's chain incurs the expense per time spent for operating the computing power. The attack thus shall put a limit to the end time to cut loss. We refer to this end time as the cut-time $t_{cut} \in \mathbb{R}^+$. A sufficient condition for the success of DS attack can be defined with applying the cut-time t_{cut} :

Definition 2. For a given cut-time $t_{cut} \in \mathbb{R}^+$, the success of DS attack is declared if, and only if, there exists a DSA time $T_{DSA} \in (0, t_{cut})$ at which $\mathcal{G}^{(1)}$ and $\mathcal{G}^{(2)}$ in Definition 1 have been achieved.

2.2. Stochastic Model

We model the conditions in Definition 2 with a stochastic model. We fit the block generation process using the PCP [22] with a given block generation rate λ (blocks per second). Including Nakamoto [1] and Rosenfeld [6], it has been most conventional to analyze the block generation process of a blockchain using PCP. A rationale why the block generation process is modeled as PCP is given in Bowden et al. [23], where experiments show the fitness of PCP model to real data samples from a live network.

We denote the lengths of the honest chain and the fraudulent chain over time $t \in (0, \infty]$ by two independent PCPs, $H(t) \in \mathbb{N}_0$ with the block generation rate λ_H (blocks per second) and $A(t) \in \mathbb{N}_0$ with the block generation rate λ_A , respectively. Both processes start at the time origin $t = 0$ (at which the DS attack is launched) at which the both chains are at the zero states, i.e., $H(0) = A(0) = 0$. Each chain independently increases at most by 1 at a time point. An increment of 1 in the counting process occurs when the pertinent network adds a new block to its chain.

We represent the difference between $A(t)$ and $H(t)$ in a discrete-time domain as a random walk $S_i \in \mathbb{Z}$ for $i \in \mathbb{N}$. For this purpose, we first define two continuous stochastic processes $M(t)$ and $S(t)$, which are respectively defined as

$$M(t) := H(t) + A(t), \tag{1}$$

and

$$S(t) := H(t) - A(t). \tag{2}$$

The first process $M(t)$ is also a PCP [22] with the rate

$$\lambda_T := \lambda_A + \lambda_H. \tag{3}$$

The second process $S(t)$ is the continuous-time analog of the random walk $S_i \in \mathbb{Z}$ for $i \in \mathbb{N}$ such that

$$S_i := S(T_i), \tag{4}$$

where T_i is the state progression time defined by

$$T_i := \inf\{t \in \mathbb{R}^+ : M(t) = i\}, \tag{5}$$

which increases as i increases. Random walk S_i is a stationary Markov chain starting from $S_0 = 0$. The state transition probabilities [22] are given by

$$p_A := \Pr(S_i = n - 1 | S_{i-1} = n) = \frac{\lambda_A}{\lambda_T}, \tag{6}$$

and

$$p_H := \Pr(S_i = n + 1 | S_{i-1} = n) = \frac{\lambda_H}{\lambda_T}, \tag{7}$$

for all $i \in \mathbb{N}$ and $n \in \mathbb{Z}$. The state transition probabilities p_H and p_A are the proportions of computing power occupied by the normal miners and that by the attacker, respectively.

We define independent and identically distributed (i.i.d.) state transition random variables $\Delta_i \in \{\pm 1\} \sim \text{Bernoulli}(p_H)$ as

$$\Delta_i := S_i - S_{i-1}, \tag{8}$$

for $i \in \mathbb{N}$. Note that $S_i = \sum_{k=0}^i \Delta_k$.

Definition 3. A DS attack $\text{DS}(p_A, t_{cut}; N_{BC})$ is a random experiment that picks a sample $\omega \in \Omega$. Each element ω is an infinite-length sequence of pairs of T_i and Δ_i in Equations (5) and (8) for all $i \in \mathbb{N}$, i.e.,

$$\omega := ((T_1, \Delta_1), (T_2, \Delta_2), \dots, (T_\infty, \Delta_\infty)). \tag{9}$$

The set Ω is the universal set of all possible ω , i.e.,

$$\Omega := \{\omega \in \{\mathbb{R}^+ \times \{\pm 1\}\}^\infty\}. \tag{10}$$

For given a DS sample $\omega \in \Omega$ and a state index $i \in \mathbb{N}$, we denote projections

$$\pi_{T_i}(\omega) := T_i \tag{11}$$

and

$$\pi_{\Delta_i}(\omega) := \Delta_i \tag{12}$$

that retrieve the progression time T_i and the transition Δ_i of the i -th state, respectively.

2.3. DS Attack Achieving Time

Definition 4. For a DS sample ω of $DS(p_A, t_{cut}; N_{BC})$, we define the DSA time T_{DSA} which measures the least one among the state progression times $\pi_{T_i}(\omega)$ of state indices i at which ω achieves the necessary conditions $\mathcal{G}^{(1)}$ and $\mathcal{G}^{(2)}$ in Definition 1.

To express T_{DSA} as a random variable, we construct event sets $\mathcal{D}_j^{(1)} \subset \Omega$ and $\mathcal{D}_{i,j}^{(2)} \subset \Omega$. The sets $\mathcal{D}_j^{(1)}$ for $j \in \{N_{BC}, N_{BC} + 1, \dots, \infty\}$ consist of DS samples ω which achieves the block confirmation $\mathcal{G}^{(1)}$ at state j for the first time. The sets $\mathcal{D}_{i,j}^{(2)}$ for $i \in \{j, j + 1, \dots, \infty\}$ and $j \in \{N_{BC}, N_{BC} + 1, \dots, \infty\}$ consists of ω which achieves the success in the PoW competition $\mathcal{G}^{(2)}$ at state i for the first time, given that $\mathcal{G}^{(1)}$ has been already achieved at state j . Subsequently, we aim for the samples $\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$ to achieve the two conditions in Definition 1 at a state pair (i, j) for the first time.

Formally, we first construct a set $\mathcal{D}_j^{(1)}$ focusing only on the first j transitions Δ_k for $k = 1, \dots, j$ of DS samples $\omega \in \Omega$ with two requirements; one is that they must have N_{BC} number of +1's and $j - N_{BC}$ number of -1's; and the other is that the j -th transition Δ_j must be +1 to guarantee that they have never been achieved in any states prior to the state j . The former requirement implies that all $\omega \in \mathcal{D}_j^{(1)}$ hold $S_j = \sum_{k=1}^j \pi_{\Delta_k}(\omega) = 2N_{BC} - j$. For example, when $N_{BC} = 2$ and $j = 5$, a sequence $(+1, -1, -1, -1, +1, \dots)$ of state transitions satisfies the first requirement, and also satisfies $S_j = 2N_{BC} - j$.

We next construct a set $\mathcal{D}_{i,j}^{(2)} \subset \Omega$ which does not care about the first j transitions Δ_k for $k = 1, \dots, j$, but only focuses on the interim transitions Δ_m for $m = j + 1, \dots, i$. By the definition, all sequences $\omega \in \mathcal{D}_{i,j}^{(2)}$ must achieve $\mathcal{G}^{(1)}$ before the j -th state, which implies that they must hold $S_j = 2N_{BC} - j$. The rest requirement for each $\omega \in \mathcal{D}_{i,j}^{(2)}$ is that the state changes from starting state $S_j = 2N_{BC} - j$ to state $S_i = -1$, while any interim states S_k remain non-negative; i.e., $S_k \geq 0$ for each $k = j + 1, \dots, i - 1$.

The sets $\mathcal{D}_j^{(1)}$ for all j are mutually exclusive as each of them represents the first satisfaction of the block confirmation condition exactly at the j -th state. For example, if $\omega \in \mathcal{D}_5^{(1)}$ then $\omega \notin \mathcal{D}_6^{(1)}$ since ω already has achieved the block confirmation at the 5-th state for the first time before reaching the 6-th state. The sets $\mathcal{D}_{i,j}^{(2)}$ for all (i, j) are also mutually exclusive for the same reason. Thus, their intersections $\mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$ for all (i, j) are also mutually exclusive.

By Definition 4, the attack achieving time T_{DSA} can be measured if there exist index pairs (i, j) such that $\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$. By the mutual exclusivity of $\mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$, if there exists such a pair (i, j) , it must be unique. In addition, if $\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$, T_{DSA} equals $\pi_{T_i}(\omega)$, since the state progression time T_k is non-decreasing as k increases. As the result, T_{DSA} can be rewritten as follows,

$$T_{DSA} = \begin{cases} \pi_{T_i}(\omega), & \text{if } \exists (i, j) \in \mathbb{N}^2: \omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}, \\ \infty, & \text{otherwise.} \end{cases} \quad (13)$$

3. The Attack Probabilities

We aim to calculate the probability distribution function (PDF) of the DSA time T_{DSA} . Using this, the success probability of DS attack with a given cut-time t_{cut} can be figured out as the probability that $T_{DSA} < t_{cut}$. Also, the expectation of attack success time can be calculated. The expected attack success time will be used in Section 4 to estimate the attack profits.

From Equation (13), the PDF of T_{DSA} requires the probabilities of two random events: one is the state progression time T_i in Equation (5); and the other is the event that a given state index i satisfies $\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$. It has been well known that T_i follows Erlang distribution [22] given as

$$f_{T_i}(t) = \frac{\lambda_T(\lambda_T t)^{i-1} e^{-\lambda_T t}}{(i-1)!}. \tag{14}$$

We provide the probability for the latter event, i.e., $p_{DSA,i} = \Pr(\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)})$ in the following Lemma 1:

Lemma 1. For a sample ω of random experiment $DS(p_A, t_{cut}; N_{BC})$, the probability $p_{DSA,i} = \Pr(\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)})$ can be computed as

$$p_{DSA,i} = \sum_{j=N_{BC}}^{j=2N_{BC}} \binom{j-1}{N_{BC}-1} C_{\frac{i-1}{2}-N_{BC}, 2N_{BC}-j} p_A^{\frac{i+1}{2}} p_H^{\frac{i-1}{2}} + \binom{i-1}{N_{BC}-1} p_H^{N_{BC}} p_A^{i-N_{BC}} \tag{15}$$

for odd $i > 2N_{BC}$, where $C_{n,m}$ is the ballot number [24] given by

$$C_{n,m} := \begin{cases} \frac{m+1}{n+m+1} \binom{2n+m}{n}, & n, m \in \mathbb{Z}^+ \cup \{0\}, \\ 0, & \text{otherwise,} \end{cases} \tag{16}$$

and for $i \leq 2N_{BC}$ and for all even-numbered i , $p_{DSA,i} = 0$.

Proof. See Appendix A. \square

By taking infinite summations of $p_{DSA,i}$ in Lemma 1 for all indices $i \in \mathbb{N}$, we can compute the probability \mathbb{P}_{DSA} that a DS attack will ever achieve the necessary conditions in Definition 1.

Corollary 1. For a sample ω of random experiment $DS(p_A, t_{cut}; N_{BC})$ with $t_{cut} = \infty$, the probability \mathbb{P}_{DSA} has an algebraic expression

$$\mathbb{P}_{DSA} = \begin{cases} 1, & p_H \leq p_A, \\ 1 - p_A^{N_{BC}+1} p_H^{N_{BC}} \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} A_j, & p_H > p_A, \end{cases} \tag{17}$$

where

$$A_j := p_A^{j-2N_{BC}-1} - p_H^{j-2N_{BC}-1}. \tag{18}$$

Proof. See Appendix B. \square

From Equation (13), the PDF of T_{DSA} follows the PDF of T_i at a given state index i , if at which it holds that $\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$, with the probability of $p_{DSA,i}$. If there does not exist such an index i , with the probability of $1 - \mathbb{P}_{DSA}$, then $T_{DSA} = \infty$. Thus, we can write the PDF $f_{T_{DSA}}$ of T_{DSA} as follows,

$$f_{T_{DSA}}(t) = \sum_{i=2N_{BC}+1}^{\infty} p_{DSA,i} f_{T_i}(t) + (1 - \mathbb{P}_{DSA}) \delta(t - \infty), \tag{19}$$

where $\delta(t)$ is the Dirac delta function.

Proposition 1. The PDF $f_{T_{DSA}}$ has an analytic expression:

$$f_{T_{DSA}}(t) = \frac{p_A \lambda_T e^{-\lambda_T t} (p_A p_H (\lambda_T t)^2)^{N_{BC}}}{(2N_{BC})!} \cdot \sum_{j=N_{BC}}^{j=2N_{BC}} \binom{j-1}{N_{BC}-1} {}_2F_3(\mathbf{a}; \mathbf{b}; p_A p_H (\lambda_T t)^2) + \frac{e^{-\lambda_T t} (p_H \lambda_T t)^{N_{BC}}}{(N_{BC}-1)!} \left(e^{p_A \lambda_T t} - \sum_{i=0}^{N_{BC}} \frac{(p_A \lambda_T t)^i}{i!} \right) + (1 - \mathbb{P}_{DSA}) \delta(t - \infty), \tag{20}$$

where ${}_pF_q(\mathbf{a}; \mathbf{b}; x)$ is the generalized hypergeometric function (See Appendix E for definition) with the parameter vectors

$$\mathbf{a} = \begin{bmatrix} N_{BC} + 1 - j/2 \\ N_{BC} + 1/2 - j/2 \end{bmatrix} \tag{21}$$

and

$$\mathbf{b} = \begin{bmatrix} 2N_{BC} + 2 - j \\ N_{BC} + 1 \\ N_{BC} + 1/2 \end{bmatrix}. \tag{22}$$

Proof. See Appendix C. \square

By Definition 2, the probability \mathbb{P}_{AS} that a DS attack $DS(p_A, t_{cut}; N_{BC})$ succeeds equals

$$\mathbb{P}_{AS}(t_{cut}) = \Pr(T_{DSA} < t_{cut}) \tag{23}$$

Note that for a special case of $t_{cut} = \infty$, $\mathbb{P}_{AS}(t_{cut}) = \mathbb{P}_{DSA}$, which coincides with the result in Rosenfeld [6].

It will be shown to be convenient to define the attack success time T_{AS} of a DS attack as

$$T_{AS} := \begin{cases} T_{DSA}, & \text{if } T_{DSA} < t_{cut}, \\ \text{not defined}, & \text{otherwise.} \end{cases} \tag{24}$$

A random variable for $T_{DSA} > t_{cut}$ does not need to be defined since it is not useful. The PDF $f_{T_{AS}}$ of T_{AS} is just a scaled version of $f_{T_{DSA}}(t)$ for $0 < t < t_{cut}$, which is given in Equation (20), with a scaling factor of \mathbb{P}_{AS}^{-1} . Formally, the PDF $f_{T_{AS}}(t)$ equals

$$f_{T_{AS}}(t) = \begin{cases} \frac{f_{T_{DSA}}(t)}{\mathbb{P}_{AS}}, & \text{for } 0 \leq t < t_{cut}, \\ 0, & \text{for } t \geq t_{cut}. \end{cases} \tag{25}$$

The expectation of attack success time is computed as

$$\mathbb{E}_{T_{AS}}(t_{cut}) = \frac{\int_0^{t_{cut}} t f_{T_{DSA}}(t) dt}{\mathbb{P}_{AS}(t_{cut})}. \tag{26}$$

The following Proposition 2 gives an explicit formula of $\mathbb{E}_{T_{AS}}$ for the special case when $t_{cut} = \infty$.

Proposition 2. Let $p_M := \max(p_A, p_H)$, $p_m := \min(p_A, p_H)$. If $t_{cut} = \infty$, the expectation $\mathbb{E}_{T_{AS}}(t_{cut})$ has a closed-form expression:

$$\lim_{t_{cut} \rightarrow \infty} \mathbb{E}_{T_{AS}}(t_{cut}) = \frac{\lambda_T^{-1} \left(\sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} Z_j + \frac{N_{BC}}{p_H} \right)}{\mathbb{P}_{DSA}}, \tag{27}$$

where

$$Z_j := p_A p_m^{N_{BC}} p_M^{-(N_{BC}-j+1)} \left(\frac{2N_{BC} - 2j p_m + 1}{p_M - p_m} \right) - j p_A^{-(N_{BC}-j)} p_H^{N_{BC}}. \quad (28)$$

Proof. See Appendix B. \square

4. Profitable DS Attacks

The previous probabilistic analyses in [1,6] have shown that the success of DS attacks is not guaranteed when $p_A < 0.5$. However, DS attacks with $p_A < 0.5$ can be vigorously pursued as long as they bring profit.

We analyze the profitability of DS attacks and to this end, we define a profit function P of a DS attack $DS(C, p_A, t_{cut}; N_{BC})$, where C is the value of a fraudulent transaction, in terms of revenue and operating expense (OPEX) of the computing power.

The OPEX X (e.g., the rental fee for the computing power) and the block mining reward R tend to increase with respect to λ_A and the time t consumed during the attack. Thus, X and R are expressed as functions of λ_A and t , and they can be any increasing function; e.g., linear, exponential, or logarithm. We define X and R , respectively, as follows:

$$X(\lambda_A, t) := \gamma \lambda_A t (\log_{x_1} x_2)^{\lambda_A} (\log_{x_3} x_4)^t \quad (29)$$

for real constants $\gamma > 0$, $x_1, x_2 > 1$, and $x_3, x_4 > 1$, and

$$R(\lambda_A, t) := \beta \lambda_A t (\log_{r_1} r_2)^{\lambda_A} (\log_{r_3} r_4)^t \quad (30)$$

for real constants $\beta > 0$, $r_1, r_2 > 1$, and $r_3, r_4 > 1$. We denote the ratio of γ and β by

$$\mu := \beta \gamma^{-1}. \quad (31)$$

With regards to P , if an attack succeeds, the revenue comes from C , as it is double-spent, added to R for the number of blocks mined during the time duration T_{AS} , i.e., $R(\lambda_A, T_{AS})$. In this case, the cost is the OPEX for the time duration T_{AS} , i.e., $X(\lambda_A, T_{AS})$. If the attack fails, the cost is the OPEX $X(\lambda_A, t_{cut})$ for the time duration t_{cut} , and there is no revenue. Hence, for a DS attack $DS(C, p_A, t_{cut}; N_{BC})$, we define P as follows,

$$P := \begin{cases} C + R(\lambda_A, T_{AS}) - X(\lambda_A, T_{AS}), & \text{if } T_{DSA} < t_{cut}, \\ -X(\lambda_A, t_{cut}), & \text{otherwise.} \end{cases} \quad (32)$$

Subsequently, the expected profit function is

$$\begin{aligned} \mathbb{E}_P &= \mathbb{P}_{AS}(t_{cut}) \cdot (C + \mathbb{E}[R(\lambda_A, T_{AS})] - \mathbb{E}[X(\lambda_A, T_{AS})]) - (1 - \mathbb{P}_{AS}(t_{cut}))X(\lambda_A, t_{cut}) \\ &= \mathbb{P}_{AS}(t_{cut}) \cdot (C + \mathbb{E}[R(\lambda_A, T_{AS})]) - \mathbb{E}_X, \end{aligned} \quad (33)$$

where \mathbb{E}_X is the expected OPEX defined as

$$\mathbb{E}_X := \mathbb{P}_{AS}(t_{cut})\mathbb{E}[X(\lambda_A, T_{AS})] + (1 - \mathbb{P}_{AS}(t_{cut}))X(\lambda_A, t_{cut}). \quad (34)$$

Definition 5. A DS attack $DS(C, p_A, t_{cut}; N_{BC})$ is said to be profitable if and only if the expected profit $\mathbb{E}_P > 0$, where \mathbb{E}_P is defined in Equation (33).

The key factor in determining the profitability of DS attacks is the value C of the fraudulent transaction. Thus, attackers would be interested in the minimum value required for profitable DS

attacks [25]. Definition 5 implies that a DS attack $DS(C, p_A, t_{cut}; N_{BC})$ is profitable if and only if $C > C_{Req.}$, where the required value of target transaction $C_{Req.}$ is

$$C_{Req.} = \frac{\mathbb{E}_X}{\mathbb{P}_{AS}} - \mathbb{E}[R(\lambda_A, T_{AS})]. \tag{35}$$

The following results in Theorem 1 and Theorem 2 focus on the case where both $X(\lambda_A, t)$ and $R(\lambda_A, t)$ are linearly increasing functions of λ_A and t .

Theorem 1. Suppose $x_1 = x_2$ and $x_3 = x_4$ in Equation (29), and $r_1 = r_2$ and $r_3 = r_4$ in Equation (30). Then, a DS attack $DS(C, p_A, t_{cut}; N_{BC})$ for any $p_A \in (0, 1)$ and for any $t_{cut} \in (0, \infty]$ is profitable if and only if $C > C_{Req.}$, where

$$C_{Req.} = \frac{(1 - \mathbb{P}_{AS}(t_{cut}))}{\mathbb{P}_{AS}(t_{cut})} \gamma \lambda_A t_{cut} - (\mu - 1) \gamma \lambda_A \mathbb{E}_{T_{AS}}(t_{cut}). \tag{36}$$

Proof. Substituting $x_1 = x_2, x_3 = x_4, r_1 = r_2,$ and $r_3 = r_4$ into Equation (35) results in Equation (36). \square

Theorem 1 shows that not only superior attackers with $p_A \in (0.5, 1)$ but also inferior attackers with $p_A \in (0, 0.5)$ are able to expect profitable DS attacks once a high enough value C greater than $C_{Req.}$ of the target transaction is designed. The condition $C_{Req.}$ in Equation (36) can be pre-computed before carrying out an attack, as it stochastically estimates the future expected cost, for a given position $p_A \in (0, 1)$ and a cut-time t_{cut} of an attacker, and a given set of network environment parameters γ and β .

Tables 1 and 2 list the resources including $C_{Req.}, \mathbb{E}_X,$ and $\mathbb{E}_{T_{AS}}$ required for profitable DS attacks respectively using $p_A = 0.35$ and $p_A = 0.4,$ when $t_{cut} = cN_{BC}\lambda_H^{-1}$ with $c = 4.$ Note that the expectation of the time spent for the block confirmation equals $N_{BC}\lambda_H^{-1},$ and we let t_{cut} linear to it. In other words, as normal traders wait for $N_{BC}\lambda_H^{-1}$ seconds on the average, attackers shall be tolerable as well and wait for the same scale of time duration. Note that the \mathbb{P}_{AS} for $N_{BC} = 1$ is smaller than that for $N_{BC} = 3$ due to not long enough $t_{cut}.$ We scaled the results by parameters λ_H and $\gamma,$ which we will explain how to obtain from the internet in the next subsection.

Table 1. Numerical computations of required resources for profitable double-spending (DS) attacks with $p_A = 0.35$ when $t_{cut} = cN_{BC}\lambda_H^{-1}$ with $c = 4.$

Block Confirmation Number (N_{BC})	1	3	5	7	9
Attack success probability (\mathbb{P}_{AS})	0.315	0.279	0.218	0.170	0.132
Expected attack success time ($\mathbb{E}_{T_{AS}}$) (Scaled by λ_H^{-1})	2.004	5.518	8.681	11.694	14.607
Expected OPEX (\mathbb{E}_X) (Scaled by γ)	1.815	5.487	9.440	13.588	17.859
Required value of target transaction ($C_{Suf.}$) (Scaled by γ)	1.079 $\cdot(1 - \mu)$ + 4.680	2.971 $\cdot(1 - \mu)$ + 16.68	4.675 $\cdot(1 - \mu)$ + 38.62	6.297 $\cdot(1 - \mu)$ + 73.84	7.866 $\cdot(1 - \mu)$ + 127.00

Table 2. Numerical computations of required resources for profitable DS attacks with $p_A = 0.4$ when $t_{cut} = cN_{BC}\lambda_H^{-1}$ with $c = 4$.

Block Confirmation Number (N_{BC})	1	3	5	7	9
Attack success probability (\mathbb{P}_{AS})	0.411	0.419	0.376	0.334	0.297
Expected attack success time ($\mathbb{E}_{T_{AS}}$)(Scaled by λ_H^{-1})	1.953	5.338	8.434	11.418	14.325
Expected OPEX (\mathbb{E}_X)(Scaled by γ)	2.106	6.139	10.436	14.977	19.716
Required value of target transaction ($C_{Suf.}$)(Scaled by γ)	1.302 ·(1 - μ) + 3.819	3.559 ·(1 - μ) + 11.10	5.622 ·(1 - μ) + 22.15	7.612 ·(1 - μ) + 37.25	9.550 ·(1 - μ) + 56.96

The following Theorem 2 is for the inferior attackers with $p_A \in (0, 0.5)$ and shows the importance of setting a finite t_{cut} .

Theorem 2. Suppose $x_1 = x_2$ and $x_3 = x_4$ in Equation (29), and $r_1 = r_2$ and $r_3 = r_4$ in Equation (30). Then, a DS attack $DS(C, p_A, t_{cut}; N_{BC})$ with $p_A \in (0, 0.5)$ is profitable only if $t_{cut} < \infty$.

Proof. For any $p_A \in (0, 0.5)$, it always holds that $\mathbb{P}_{AS} < 1$. In this case, if $t_{cut} \rightarrow \infty$ then $C_{Req.} \rightarrow \infty$ from Equation (36); i.e., infinite value C of fraudulent transaction is required for a DS attack $DS(C, p_A, t_{cut}; N_{BC})$ to be profitable. Thus, for a DS attack with $p_A \in (0, 0.5)$ to be profitable, a finite cut-time $t_{cut} < \infty$ must be set. □

Theorem 2 shows that for $p_A \in (0, 0.5)$, setting $t_{cut} = \infty$ is expected to incur infinite deficit. On the contrary, for $p_A \in (0.5, 1)$, what we have numerically checked out but omitted due to space limitation is the result that \mathbb{E}_P is an increasing function of t_{cut} ; i.e., setting $t_{cut} = \infty$ is the optimal choice in the superior attack regime. Applying $p_A \in (0.5, 1)$ and $t_{cut} = \infty$ into Equation (36) leads to $\mathbb{P}_{AS} = 1$, and thus $C_{Req.}$ turns into

$$C_{Req.} = -(\mu - 1)\gamma\lambda_A\mathbb{E}_{T_{AS}}, \tag{37}$$

where a closed-form expression of $\mathbb{E}_{T_{AS}}$ is given in Proposition 2. In this case, if $\beta > \gamma$; i.e., $\mu > 1$, DS attacks are always profitable regardless of C . According to nicehash.com, most networks maintain $\beta > \gamma$ by the economic equilibrium. As the result, in addition to the results in [1] and [6] that DS attacks with $p_A \in (0.5, 1)$ guarantee probabilistic success, we show that such attacks guarantee economic gain as well.

5. Practical Example of Profitable DS Attacks against BitcoinCash

We analyze resources required for profitable DS attacks against BitcoinCash network. The resources include the computing power proportion p_A , expected OPEX \mathbb{E}_X , expected attack success time $\mathbb{E}_{T_{AS}}$, and the required value of fraudulent transaction $C_{Req.}$.

To this end, we first recall the parameters involved in block mining reward R and the OPEX X . The parameters used in Equation (29) and Equation (30) are assumed to $x_1 = x_2$, $x_3 = x_4$, $r_1 = r_2$, and $r_3 = r_4$ which lead to linear functions $X(\lambda_A, t)$ and $R(\lambda_A, t)$ with respect to λ_A and t . There are three more parameters: γ , β , and λ_H^{-1} . From Equation (29) and Equation (30), the parameter γ is the expected cost spent per generating a block; and the parameter β is the reward per generating a block. Parameter λ_H^{-1} is the average block generation time of the honest chain. All the parameters are different for each blockchain network.

In BitcoinCash, the reward β per block mining was 12.5 BCH (without transaction fees), which is around $\beta = 0.44$ BTC per block mining (as of 26 February 2020). The average block generation time was fixed at $\lambda_H^{-1} = 600$ seconds.

The parameter γ is obtainable from nicehash.com. BitcoinCash uses the SHA-256 cryptographic puzzle for which the unit of computation is hash. As of 26th Feb. 2020, the rental fee for 1-peta (P) hashes per second for a day was around 0.017 BTC, which was around 1.97×10^{-7} BTC per second. In other words, the rental fee was approximately 1.97×10^{-22} BTC per the computing of a hash. Referring to BTC.com, the network’s computing speed is 3.57-exa (E) hashes per second; i.e., $3.57E \cdot 600 = 2142E$ hashes are needed to generate one block on the average. As the result, the parameter γ is obtained as

$$\begin{aligned} \gamma &= 1.97 \times 10^{-22} \text{ [BTC/hash]} \times 2142E \text{ [hashes/block mining]} \\ &\approx 0.422 \text{ [BTC/block mining]}. \end{aligned} \tag{38}$$

Note that it holds $\beta > \gamma$. From Equation (37), this relationship makes DS attack $DS(C, p_A, t_{cut}; N_{BC})$ with $p_A > 0.5$ and $t_{cut} = \infty$ always profitable regardless of the value C of target transaction.

In case of DS attacks with $p_A < 0.5$, the cut-time t_{cut} must be determined as a finite value for profitable DS attacks by Theorem 2. We set $t_{cut} = cN_{BC}\lambda_H^{-1} = 12000$ seconds with $c = 4$ and $p_A = 0.35$. We compute the resources required for profitable DS attacks against BitcoinCash when $N_{BC} = 5$. Results are obtainable from the values in Tables 1 and 2 by multiplying the scaling parameters $\gamma = 0.422$ and $\lambda_H^{-1} = 600$ and by substituting $\mu = \beta\gamma^{-1} = 1.04$ and $c = 4$.

As the results, we obtain $\mathbb{P}_{AS} \approx 0.218$, $\mathbb{E}_{T_{AS}} \approx 5200$ seconds, $\mathbb{E}_X \approx 3.98$ BTC, and $C_{Req.} \approx 16.22$ BTC. One can compute expected running time; i.e., the expected time spent for a single DS attack attempt as $\mathbb{P}_{AS}\mathbb{E}_{T_{AS}} + (1 - \mathbb{P}_{AS})t_{cut}$, which is around 2 h and 55 min. That is to say, attackers can repeatedly perform n number of attacks every 2 h and 55 min on the average. With the value C of target transaction, by the strong law of large numbers, the multiple attack attempts will return net profit $n\mathbb{P}_{AS}(t_{cut}) \cdot (C - C_{Req.})$ as $n \rightarrow \infty$ with probability 1.

6. Related Works

By Nakamoto [1] and Rosenfeld [6], the probabilities have been studied that a DS attack will ever succeed when there is no time limit, i.e., the cut-time is set to $t_{cut} = \infty$. Both of them applied PCPs to model the growth of chains $H(t)$ and $A(t)$. On one hand, the main difference between them was in probability calculations of the block confirmation process in Definition 1. Rosenfeld applied the PCPs to both $H(t)$ and $A(t)$, whereas Nakamoto assumed the time spent for $H(t) \geq N_{BC}$ deterministic to simplify the calculation. On the other hand, they both used the gambler’s ruin approach to obtain the asymptotical behavior of S_i as $i \rightarrow \infty$ by manipulating the recurrence relationship between two adjacent states. Namely, their results are based on an assumption that an indefinite number of attack chances are given [12].

On the contrary, we introduce the cut-time t_{cut} which generalizes analytical framework to the more interesting finite attack time and inferior attacker regime. By setting t_{cut} infinite, the same result \mathbb{P}_{DSA} was obtained in [6] as well. By setting a finite t_{cut} , our results shall be useful when attack chances are limited due to limited amount of resources such as time and cost. In addition, we show in Theorem 2 that DS attacks with $p_A < 0.5$ must set a finite t_{cut} in order to expect a non-negative profit. It shall be noted that there has been no intermediate result like $p_{DSA,i}$ in Lemma 1. We use Lemma 1 to derive the novel results.

Rosenfeld [6] and Bissias et al. [13] have analyzed the profitability of DS attacks. However, they put additional assumptions on the attack scenario to simplify the calculation of the attack time. Specifically, Rosenfeld assumed the attack time to be a constant. Bissias et al. assumed that the attack stops if either the normal peers or the attacker achieves the block confirmation first. On the contrary, in our model, an attack can be continued for a random attack time as long as it brings profit, even if the normal peers achieve the block confirmation before the attacker does.

In Zaghloul et al. [14], the profit of DS attack has been analyzed. Interestingly, they have discussed the need of cut-time for DS attacks with $p_A < 0.5$, which is theoretically proven in this paper in

Theorem 2. They also calculated the profit of DS attacks with a finite time-limit (see Section IV-C in [14]), but their calculation was not as precise as ours in three points:

First, the probability of attack success within a finite time-limit, i.e., $\mathbb{P}_{AS}(t_{cut})$ in Equation (23) was never considered, which requires the distribution of the DS achieving time, i.e., T_{DSA} given in Proposition 1. Instead, their calculation used \mathbb{P}_{DSA} referring to the result in Rosenfeld [6]. This contradicts their time-limited attack scenario, since \mathbb{P}_{DSA} in [6] was resulted from the assumption of infinite time-limit.

Second, they approximated costs and revenues of DS attack spent within a time-limit. Estimation of the costs and revenues requires estimations of the numbers of blocks respectively mined by honest nodes and attackers within a time-limit, but those were assumed to be constant. This was due to the absence of the time analysis we provide in Proposition 1.

Third, they assumed the average block generation rates λ_H, λ_A respectively by honest miners and by attackers are always the same. Since, the proportions p_H, p_A of computing power occupied by the two groups can be quite different in general, such a result is not very useful. We agree to their assumption that most blockchains control the difficulty of block mining puzzle to keep the average speed of block generation constant, and thus λ_H can be considered as a constant. However, λ_A should be left as a varying quantity by p_A . The fact is that the computing power invested by the attacker cannot be monitored by the honest network and thus it cannot be reflected in the difficulty control routine.

Budish [15] conducted simulations on the profitability of DS attacks only in the cases of $p_A > 0.5$. Under the cases, a condition on the value of the target transaction that makes DS attacks not profitable has been given based on the simulations. We give theoretical and numerically-calculable results for any $p_A \in (0, 1)$, which do not require massive simulations.

Gervais et al. [16] and Sompolinsky et al. [12] have used a Markov decision process (MDP) to analyze profits from DS attacks. These works differ from our contributions in the following regards:

First, they did not follow the DS attacks scenario considered by Nakamoto [1] and Rosenfeld [6]. Instead, the scenario in [12] was a special case of the pre-mining strategy which was introduced in [17,18]. We show that the success of DS attack under this scenario is even more difficult to occur than the success of the DS attack under the scenario of Nakamoto and Rosenfeld (see Appendix D for details). Also, the attack scenario in [16] went even further by modifying the condition for block confirmation in Definition 1. Specifically, under our definition, it is required for the honest chain to have added N_{BC} blocks, while under their condition it was fraudulent for the chain to do so (see Section 3 of [16]). Thus, it was not ensured that the potential victim has shipped the goods or service, and an attack success did not guarantee for the attacker to obtain the benefit of attacking.

Second, one important new advance in this paper is the derivation of the time analysis $f_{T_{AS}}$ given in Proposition 1. When one uses the MDP framework, one can obtain similar information such as the estimations for the attack success time $E_{T_{AS}}$, the future profit P that an attacker will earn in the end, and the minimum value of target transaction $C_{Req.}$. However, using MDP to make such estimations would have required extensive Monte Carlo simulations. Using our mathematical results, such estimations can be obtained without Monte Carlo simulations.

In addition, we believe that our mathematical results can be utilized in the MDP frameworks to improve the reliability of analyses. Conventionally, a rational user of an MDP will make a decision at every state whether to stop or to continue the process by comparing the rewards that will be incurred in the future by his/her decision. The rewards for stop actions are clear because such actions are either an attack success or a give-up. The reward for the continue action is complex because it needs to consider all the actions in all future possible states as well. In [12,16], the rewards for the continue action were over-simplified as they were evaluated only for the very next state and did not include the estimation of the profits in further future actions. To improve the reliability, the PDF $f_{T_{AS}}$ in Proposition 1 can be used at any intermediate Markov state to estimate the future profits. Specifically, the conditional expectation of the time left for an attack success T_{AS} given $T_{AS} > \tau$ can be calculated using $f_{T_{AS}}$, where τ is the observable time elapsed for reaching the current state. Once the time left is estimated, the estimation

of future profits can be updated by substituting it into Equation (33). That is to say, at each state, the estimation of profits can be updated and used as the rewards resulting from the continue action.

Goffard [19] and Karame et al. [20] have derived the PDFs of attack success time, but none of their DS attack scenarios matched with ours in Definition 1. In [19], Goffard derived the PDF of catch-up time spent for the fraudulent chain to catch up with the honest chain given that the length of honest chain is initially ahead by several blocks. The author used counting processes such as order statistic point process and renewal process which are more general than PCP, but there was no analytic result similar to what is given in Proposition 1. In [20], Karame et al. derived the PDF of the first attack success time under a fast-payment model which fixed $N_{BC} = 0$. To sum up, the attack success time in neither analysis included the time spent for achieving the first condition: the block confirmation should be realized.

7. Discussion and Conclusions

We showed that DS attacks using 50% or a lower proportion of computing power can be profitable and thus quite threatening. We provided how much quantitative resources are required to make a profitable DS attack. We derive the PDF of attack success time which enables us to figure out the operating time and the expense of mining rigs. We provided MATLAB codes on the website (<https://codeocean.com/capsule/2308305/tree>) for numerical evaluation of the expected profit function in Equation (33). We also listed an example of the minimum resources required for a profitable DS attack, which is applicable to any blockchain networks by substituting the network parameters γ , β , and λ_H . We also showed a more specific example of the required resources against BitcoinCash network.

Our results quantitatively guide how to set a block confirmation number for a safe transaction. The lower the block confirmation number is, the lower the minimum resource is required for a profitable attack. A solution can be utilized by the network developers to discourage such an attack. On the one hand, given a block confirmation number, we can have the value of any transaction to be set below the required value of making a profitable attack in a given network. On the other hand, given the value of transaction, the network can provide a service to inform the payee with the lowest block confirmation number that leads to negative DS attack profit.

Author Contributions: Conceptualization, J.J. and H.-N.L.; methodology, J.J.; software, J.J.; validation, J.J.; formal analysis, J.J.; investigation, J.J.; resources, J.J.; data curation, J.J.; writing—original draft preparation, J.J.; writing—review and editing, J.J. and H.-N.L.; visualization, J.J. and H.-N.L.; supervision, H.-N.L.; project administration, H.-N.L.; funding acquisition, H.-N.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by Institute of Information & Communications Technology Planning & Evaluation, grant number 2020-0-00958. This work was partially supported by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (NRF-2018R1A2A1A19018665).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Appendix A

Proof of Lemma 1. For a given sample ω and a given index i , the event $\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$ is equivalent to the event that there exists an intermediate state index j such that $\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$. By the mutual exclusiveness of $\mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}$ for integers j , such a state j is unique if it exists. Thus, we can write the probability $p_{DSA,i}$ as follows,

$$\begin{aligned}
 p_{DSA,i} &= \Pr\left(\exists j \in \mathbb{N}: \omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}\right) \\
 &= \sum_{j=N_{BC}}^{\infty} \Pr\left(\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}\right).
 \end{aligned}
 \tag{A1}$$

Note that $\mathcal{D}_j^{(1)} \cap (\mathcal{D}_{i,j}^{(2)}) = \phi$ for $i \leq 2N_{BC}$, since the minimum number of states for an attack success is $2N_{BC} + 1$: N_{BC} number of +1 's state transitions for the block confirmation; and $N_{BC} + 1$ number of -1 's state transitions for the success of PoW competition. Thus, $p_{DSA,i} = 0$ for $i \leq 2N_{BC}$.

We further explore $\mathcal{D}_j^{(1)}$ and $\mathcal{D}_{i,j}^{(2)}$. We divide the domain of state index j in Equation (A1) into two exclusive domains; one is $j \leq 2N_{BC}$; and the other is $j > 2N_{BC}$. First, for $j \leq 2N_{BC}$, two sets $\mathcal{D}_j^{(1)}$ and $\mathcal{D}_{i,j}^{(2)}$ are independent, since their requirements on the state transitions are focusing on disjoint indices of state by their definitions. Formally, $\Pr(\omega \in \mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)}) = \Pr(\omega \in \mathcal{D}_j^{(1)})\Pr(\omega \in \mathcal{D}_{i,j}^{(2)})$. Second, we explore the domain $j > 2N_{BC}$. By the definition of $\mathcal{D}_j^{(1)}$, all $\omega \in \mathcal{D}_j^{(1)}$ satisfy $S_j = \sum_{k=1}^j \pi_{\Delta_k}(\omega) = 2N_{BC} - j$. Thus, for every $j > 2N_{BC}$, S_j is already negative, which implies all $\omega \in \mathcal{D}_j^{(1)}$ satisfy both and at state j . The set $\mathcal{D}_{i,j}^{(2)} = \phi$ for $j > 2N_{BC}$ and $j < i$, since the state $S_j = 2N_{BC} - j$ contradicts one requirement of $\mathcal{D}_{i,j}^{(2)}$: the interim transitions between the states j and i should be non-negative. For $j > 2N_{BC}$ and $j = i$, let us set $\mathcal{D}_{i,j}^{(2)} = \Omega$, since there is no interim state to apply the requirement to. To sum up, $\mathcal{D}_j^{(1)} \cap \mathcal{D}_{i,j}^{(2)} = \mathcal{D}_i^{(1)}$ for $j > 2N_{BC}$ and $i = j$, and $\mathcal{D}_j^{(1)} \cap (\mathcal{D}_{i,j}^{(2)}) = \phi$ for $j > 2N_{BC}$ and $i > j$. Subsequently, Equation (A1) is computed as

$$p_{DSA,i} = \sum_{j=N_{BC}}^{2N_{BC}} \Pr(\omega \in \mathcal{D}_j^{(1)})\Pr(\omega \in \mathcal{D}_{i,j}^{(2)}) + \Pr(\omega \in \mathcal{D}_i^{(1)}). \tag{A2}$$

We now compute the ingredient probabilities $\Pr(\omega \in \mathcal{D}_j^{(1)})$ and $\Pr(\omega \in \mathcal{D}_{i,j}^{(2)})$ in Equation (A2). First, by the definition, all samples in $\mathcal{D}_j^{(1)}$ must have $N_{BC} - 1$ number of +1 's state transitions among the first $j - 1$ transitions. And the rest of the $j - 1$ transitions must be valued by -1. In addition, the j -th transition must be valued by +1 so that the block confirmation is achieved exactly at the j -th state index. As the result, the probability $\Pr(\omega \in \mathcal{D}_j^{(1)})$ equals the point mass function of a negative binomial distribution:

$$\Pr(\omega \in \mathcal{D}_j^{(1)}) = \binom{j-1}{N_{BC}-1} p_H^{N_{BC}} p_A^{j-N_{BC}}. \tag{A3}$$

Second, computing the probability $\Pr(\omega \in \mathcal{D}_{i,j}^{(2)})$ starts from counting the number of combinations of state transitions satisfying the requirements of set $\mathcal{D}_{i,j}^{(2)}$. Recall the requirements on every element of $\mathcal{D}_{i,j}^{(2)}$, for $j = N_{BC}, \dots, 2N_{BC}$, are that the state starts from the state $S_j = 2N_{BC} - j$ and ends at the state $S_i = -1$ while all the $i - j - 1$ number of interim states remain nonnegative. The i -th transition should be $\Delta_i = -1$ so that the success of PoW competition is achieved exactly at the state index i . The number of combinations of such state transitions can be counted using the ballot number $C_{n,m}$ [24], which is the number of random walks that consist of $2n + m$ steps and never become negative, starting from the origin and ending at the point m . In our problem, the number of random walk steps is $2n + m = i - j - 1$ with $m = 2N_{BC} - j$. As a result, by multiplying the probabilities p_A and p_H for state transitions, the probability $\Pr(\omega \in \mathcal{D}_{i,j}^{(2)})$ is computed as

$$\Pr(\omega \in \mathcal{D}_{i,j}^{(2)}) = C_{n,m} p_A^{(n+m+1)} p_H^n, \tag{A4}$$

where $2n + m = i - j - 1$ and $m = 2N_{BC} - j$.

Finally, substituting Equations (A3) and (A4) into Equation (A2) results in Equation (15). \square

Appendix B

Proof of Corollary 1. Taking infinite summations of $p_{DSA,i}$ for all indices i results in \mathbb{P}_{DSA} :

$$\mathbb{P}_{DSA} = \sum_{i=2N_{BC}+1}^{\infty} p_{DSA,i} \tag{A5}$$

By substituting $p_{DSA,i}$ in Lemma 1 into Equation (A5), the probability \mathbb{P}_{DSA} becomes

$$\mathbb{P}_{DSA} = \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} p_A \sum_{i=2N_{BC}+1}^{\infty} C_{\frac{i-1}{2}-N_{BC}, 2N_{BC}-j} (p_A p_H)^{\frac{i-1}{2}} + \left(\frac{p_H}{p_A}\right)^{N_{BC}} \sum_{i=2N_{BC}+1}^{\infty} \binom{i-1}{N_{BC}-1} p_A^i. \tag{A6}$$

By rearranging the indices i in the summations, we can obtain

$$\begin{aligned} \mathbb{P}_{DSA} &= \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} p_A \sum_{i=0}^{\infty} C_{i, 2N_{BC}-j} (p_A p_H)^{i+N_{BC}} \\ &+ \left(\frac{p_H}{p_A}\right)^{N_{BC}} \left(\sum_{i=N_{BC}}^{\infty} \binom{i-1}{N_{BC}-1} p_A^i - \sum_{i=N_{BC}}^{2N_{BC}} \binom{i-1}{N_{BC}-1} p_A^i \right). \end{aligned} \tag{A7}$$

We define two generating functions as

$$M_k(x) := \sum_{i=0}^{\infty} C_{i,k} x^i, \tag{A8}$$

and

$$G_k(x) := \sum_{i=k}^{\infty} \binom{i}{k} x^i. \tag{A9}$$

By substituting M_k and G_k into Equation (A7), we can write

$$\begin{aligned} \mathbb{P}_{DSA} &= \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} p_A (p_A p_H)^{N_{BC}} M_{2N_{BC}-j}(p_A p_H) \\ &+ \left(\frac{p_H}{p_A}\right)^{N_{BC}} \left(p_A G_{N_{BC}-1}(p_A) - \sum_{i=N_{BC}}^{2N_{BC}} \binom{i-1}{N_{BC}-1} p_A^i \right) \end{aligned} \tag{A10}$$

The function $M_k(x)$ is a generating function of the ballot numbers $C_{i,k}$, for which the algebraic expression given in [26] is

$$M_k(x) = \left(\frac{2}{1 + \sqrt{1-4x}} \right)^{k+1}. \tag{A11}$$

Putting $x = p_A p_H$ into $M_k(x)$ results in

$$\begin{aligned} M_k(p_A p_H) &= \left(\frac{2}{1 + \sqrt{1-4p_A p_H}} \right)^{k+1} \\ &= \begin{cases} \left(\frac{2}{1 + \sqrt{1-4p_A(1-p_A)}} \right)^{k+1}, & \text{if } p_A < p_H, \\ \left(\frac{2}{1 + \sqrt{1-4(1-p_H)p_H}} \right)^{k+1}, & \text{if } p_A \geq p_H \\ \left(\frac{1}{p_M} \right)^{k+1}, & \end{cases} \end{aligned} \tag{A12}$$

where $p_M := \max(p_H, p_A)$. The function $G_k(x)$ is a generating function of binomial coefficients, and the algebraic expression for it is given in [27]:

$$G_k(x) = \frac{x^k}{(1-x)^{k+1}}. \tag{A13}$$

Putting $x = p_A$ into $G_k(x)$ results in

$$G_k(p_A) = p_H^{-1} \left(\frac{p_A}{p_H} \right)^k. \tag{A14}$$

Substituting Equation (A12) and Equation (A14) into Equation (A10) provides

$$\mathbb{P}_{DSA} = \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} p_A (p_A p_H)^{N_{BC}} p_M^{-(2N_{BC}-j+1)} + 1 - \left(\frac{p_H}{p_A} \right)^{N_{BC}} \sum_{i=N_{BC}}^{2N_{BC}} \binom{i-1}{N_{BC}-1} p_A^i. \tag{A15}$$

We define $p_m := \min(p_A, p_H)$, then the relationship $p_A p_H = p_m p_M$ holds. By rearranging the order of operands, we can obtain

$$\mathbb{P}_{DSA} = 1 - \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} \left(\left(\frac{p_H}{p_A} \right)^{N_{BC}} p_A^j - \frac{p_A}{p_M} \left(\frac{p_m}{p_M} \right)^{N_{BC}} p_M^j \right), \tag{A16}$$

which is equal to Equation (17). □

Proof of Proposition 2. From Equations (19) and (26), when $t_{cut} = \infty$, we obtain

$$\begin{aligned} \mathbb{E}_{T_{AS}} &= \frac{\lim_{t_{cut} \rightarrow \infty} \int_0^{t_{cut}} t f_{T_{DSA}}(t) dt}{\mathbb{P}_{AS}(t_{cut})} = \frac{\sum_{i=2N_{BC}+1}^{\infty} \mathbb{E}[T_i] p_{DSA,i}}{\mathbb{P}_{DSA}} \\ &= \frac{\sum_{i=2N_{BC}+1}^{\infty} \frac{i}{\lambda_T} p_{DSA,i}}{\mathbb{P}_{DSA}}, \end{aligned} \tag{A17}$$

where $E[T_i] = i \lambda_T^{-1}$ [22]. By substituting $P_{D_{SA},i}$ in Equation (15) into Equation (A17) and rearranging the order of operands, we obtain

$$\begin{aligned} \lambda_T \mathbb{P}_{DSA} \mathbb{E}_{T_{AS}} &= \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} \sum_{i=2N_{BC}}^{\infty} (i+1) C_{\frac{i}{2}-N_{BC}, 2N_{BC}-j} p_A^{\frac{i+2}{2}} p_H^{\frac{i}{2}} \\ &+ \sum_{i=N_{BC}-1}^{\infty} (i+1) \binom{i}{N_{BC}-1} p_A^{i+1-N_{BC}} p_H^{N_{BC}} - \sum_{i=N_{BC}-1}^{2N_{BC}-1} (i+1) \binom{i}{N_{BC}-1} p_A^{i+1-N_{BC}} p_H^{N_{BC}}. \end{aligned} \tag{A18}$$

By rearranging the indices of summations, we arrive at

$$\begin{aligned} \lambda_T \mathbb{P}_{DSA} \mathbb{E}_{T_{AS}} &= \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} p_A^{N_{BC}+1} p_H^{N_{BC}} \cdot \sum_{i=0}^{\infty} (2i+2N_{BC}+1) C_{i, 2N_{BC}-j} (p_A p_H)^i \\ &+ p_A \left(\frac{p_H}{p_A} \right)^{N_{BC}} \sum_{i=N_{BC}-1}^{\infty} (i+1) \binom{i}{N_{BC}-1} p_A^i - \sum_{i=N_{BC}}^{2N_{BC}} i \binom{i-1}{N_{BC}-1} p_A^{i-N_{BC}} p_H^{N_{BC}}. \end{aligned} \tag{A19}$$

By substituting the generating functions $M_k(x)$ and $G_k(x)$ defined respectively in Equation (A8) and Equation (A9), Equation (A19) becomes

$$\begin{aligned} \lambda_T \mathbb{P}_{DSA} \mathbb{E}_{T_{AS}} &= \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} p_A^{N_{BC}+1} p_H^{N_{BC}} \cdot \left(2 \sum_{i=0}^{\infty} i C_{i,2N_{BC}-j} (p_A p_H)^i + (2N_{BC} + 1) M_{2N_{BC}-j}(p_A p_H) \right) \\ &+ p_A \left(\frac{p_H}{p_A} \right)^{N_{BC}} \left(\sum_{i=N_{BC}-1}^{\infty} i \binom{i}{N_{BC}-1} p_A^i + G_{N_{BC}-1}(p_A) \right) - \sum_{i=N_{BC}}^{2N_{BC}} i \binom{i-1}{N_{BC}-1} p_A^{i-N_{BC}} p_H^{N_{BC}}. \end{aligned} \tag{A20}$$

We use the following relationships,

$$\sum_{i=0}^{\infty} i C_{i,k} x^i = x M'_k(x) \tag{A21}$$

and

$$\sum_{i=k}^{\infty} i \binom{i}{k} x^i = x G'_k(x), \tag{A22}$$

and their derivatives are given by

$$\begin{aligned} M'_k(x) &:= \frac{d}{dx} M_k(x) = \sum_{i=0}^{\infty} i C_{i,k} x^{i-1} \\ &= \frac{(k+1)}{\sqrt{1-4x}} \left(\frac{2}{1+\sqrt{1-4x}} \right)^{k+2} \end{aligned} \tag{A23}$$

and

$$\begin{aligned} G'_k(x) &:= \frac{d}{dx} G_k(x) \\ &= \sum_{i=k}^{\infty} i \binom{i}{k} x^{i-1} \\ &= \frac{(kx^{k-1} + x^k)}{(1-x)^{k+2}}. \end{aligned} \tag{A24}$$

By substituting Equation (A21) and Equation (A22) into Equation (A20), we obtain

$$\begin{aligned} \lambda_T \mathbb{P}_{DSA} \mathbb{E}_{T_{AS}} &= \sum_{j=N_{BC}}^{2N_{BC}} \binom{j-1}{N_{BC}-1} p_A^{N_{BC}+1} p_H^{N_{BC}} \cdot \left(2 p_A p_H M'_{2N_{BC}-j}(p_A p_H) + (2N_{BC} + 1) M_{2N_{BC}-j}(p_A p_H) \right) \\ &+ p_A \left(\frac{p_H}{p_A} \right)^{N_{BC}} \left(p_A G'_{N_{BC}-1}(p_A) + G_{N_{BC}-1}(p_A) \right) - \sum_{i=N_{BC}}^{2N_{BC}} i \binom{i-1}{N_{BC}-1} p_A^{i-N_{BC}} p_H^{N_{BC}} \end{aligned} \tag{A25}$$

Putting $x = p_A p_H$ into $M'_k(x)$ in Equation (A23) results in

$$M'_k(p_A p_H) = M'_k(p_m p_M) = \frac{(k+1)}{1-2p_m} \left(\frac{1}{p_M} \right)^{k+2}. \tag{A26}$$

Putting $x = p_A$ into $G'_k(x)$ in Equation (A24) gives

$$G'_k(p_A) = \frac{(k p_A^{k-1} + p_A^k)}{p_H^{k+2}}. \tag{A27}$$

By substituting Equation (A12), Equation (A14), Equation (A26), and Equation (A27) into Equation (A25), we finally obtain Equation (27). □

Appendix C

Proof of Proposition 1. We use a generating function and generalized hypergeometric functions to compute the infinite summations in Equation (19).

By substituting $P_{D_{SA},i}$ in Equation (15) and $f_{T_i}(t)$ in Equation (14) into Equation (19), we arrive at

$$f_{T_{D_{SA}}}(t) - (1 - \mathbb{P}_{D_{SA}})\delta(t - \infty) = \sum_{j=N_{BC}}^{j=2N_{BC}} \binom{j-1}{N_{BC}-1} \sum_{i=2N_{BC}+1}^{\infty} C_{i-\frac{1}{2}, -N_{BC}, 2N_{BC}-j} p_A^{\frac{i+1}{2}} p_H^{\frac{i-1}{2}} \frac{\lambda_T^{i+1} e^{-\lambda_T t}}{(i-1)!} + \sum_{i=2N_{BC}+1}^{\infty} \binom{i-1}{N_{BC}-1} p_H^{N_{BC}} p_A^{i-N_{BC}} \frac{\lambda_T^{i+1} e^{-\lambda_T t}}{(i-1)!}. \tag{A28}$$

By rearranging the indices of summations and the order of operands, we obtain

$$f_{T_{D_{SA}}}(t) - (1 - \mathbb{P}_{D_{SA}})\delta(t - \infty) = \sum_{j=N_{BC}}^{j=2N_{BC}} \binom{j-1}{N_{BC}-1} \sum_{i=0}^{\infty} (C_{i, 2N_{BC}-j} p_A^{N_{BC}+i+1} p_H^{N_{BC}+i} \frac{\lambda_T^{2N_{BC}+2i+1} e^{-\lambda_T t}}{(2N_{BC}+2i)!}) + (\frac{p_H}{p_A})^{N_{BC}} e^{-\lambda_T t} \left(\sum_{i=N_{BC}}^{\infty} \binom{i-1}{N_{BC}-1} p_A^i \frac{\lambda_T^{i+1} e^{-\lambda_T t}}{(i-1)!} - \sum_{i=N_{BC}}^{2N_{BC}} \binom{i-1}{N_{BC}-1} p_A^i \frac{\lambda_T^{i+1} e^{-\lambda_T t}}{(i-1)!} \right). \tag{A29}$$

We can define two generating functions as

$$B(x) := \sum_{i=0}^{\infty} C_{i, 2N_{BC}-j} \frac{x^i}{(2N_{BC} + 2i)!} = (2N_{BC} - j + 1) \sum_{i=0}^{\infty} \frac{(2i + 2N_{BC} - j)!}{i!(i + 2N_{BC} - j + 1)! (2N_{BC} + 2i)!} x^i \tag{A30}$$

and

$$H(x) := \sum_{i=N_{BC}}^{\infty} \binom{i-1}{N_{BC}-1} \frac{x^{i-1}}{(i-1)!} = \sum_{i=N_{BC}-1}^{\infty} \binom{i}{N_{BC}-1} \frac{x^i}{i!}. \tag{A31}$$

By substituting $B(x)$ and $H(x)$ into Equation (A29), we obtain

$$f_{T_{D_{SA}}}(t) - (1 - \mathbb{P}_{D_{SA}})\delta(t - \infty) = \sum_{j=N_{BC}}^{j=2N_{BC}} \binom{j-1}{N_{BC}-1} p_A \lambda_T e^{-\lambda_T t} (p_A p_H (\lambda_T t)^2)^{N_{BC}} B(p_A p_H (\lambda_T t)^2) + (\frac{p_H}{p_A})^{N_{BC}} e^{-\lambda_T t} \left(p_A \lambda_T H(p_A \lambda_T t) - \sum_{i=N_{BC}}^{2N_{BC}} \binom{i-1}{N_{BC}-1} p_A^i \frac{\lambda_T^{i+1} e^{-\lambda_T t}}{(i-1)!} \right). \tag{A32}$$

We replace function $B(x)$ in Equation (A30) with the generalized hypergeometric functions (See Appendix E for definition). For this purpose, we first denote the sequences in $B(x)$ by

$$\beta_i := \frac{(2i + 2N_{BC} - j)!}{i!(i + 2N_{BC} - j + 1)! (2N_{BC} + 2i)!} \tag{A33}$$

and

$$\beta_0 := \frac{1}{(2N_{BC} - j + 1)(2N_{BC})!}. \tag{A34}$$

Next, the function $B(x)$ can be rewritten as

$$B(x) = (2N_{BC} - j + 1) \sum_{i=0}^{\infty} \beta_i x^i = (2N_{BC} - j + 1) \beta_0 \left(x^0 + \frac{\beta_1}{\beta_0} x^1 + \frac{\beta_2 \beta_1}{\beta_1 \beta_0} x^2 + \dots \right). \tag{A35}$$

The reformulated sequence in Equation (A35) is computed by

$$\frac{\beta_{i+1}}{\beta_i} = \frac{(i + 1 + N_{BC} - j/2)(i + 1/2 + N_{BC} - j/2)}{(i + 2 + 2N_{BC} - j)(i + 1 + N_{BC})(i + 1/2 + N_{BC})(i + 1)}, \tag{A36}$$

which has 2 polynomials in i on the numerator and 3 polynomials in i except for $(i + 1)$ on the denominator. $B(x)$ can be expressed in terms of a generalized hypergeometric function ${}_2F_3$ [28] as follows,

$$B(x) = (2N_{BC} - j + 1) \beta_0 {}_2F_3(\mathbf{a}_j; \mathbf{b}_j; x) = \frac{1}{(2N_{BC})!} {}_2F_3(\mathbf{a}_j; \mathbf{b}_j; x), \tag{A37}$$

where vectors \mathbf{a}_j and \mathbf{b}_j respectively defined in Equations (21) and (22) are the constants in the polynomials in i of the numerator and denominator in Equation (A31), respectively.

We use a closed-form expression of generating function $H(x)$ in Equation (A31) given by

$$H(x) = \sum_{i=N_{BC}-1}^{\infty} \binom{i}{N_{BC}-1} \frac{x^i}{i!} = \frac{1}{(N_{BC}-1)!} \sum_{i=N_{BC}-1}^{\infty} \frac{x^i}{(i-N_{BC}+1)!} \tag{A38}$$

$$= \frac{x^{N_{BC}-1}}{(N_{BC}-1)!} e^x,$$

where the following relationship is used [29]:

$$\sum_{i=0}^{\infty} \frac{x^i}{i!} = e^x. \tag{A39}$$

By substituting Equation (A37) and Equation (A38) into Equation (A32), we arrive at

$$\begin{aligned} & f_{T_{DSA}}(t) - (1 - \mathbb{P}_{DSA})\delta(t - \infty) \\ &= \frac{p_A \lambda_T e^{-\lambda_T t} (p_A p_H (\lambda_T t)^2)^{N_{BC}}}{(2N_{BC})!} \cdot \sum_{j=N_{BC}}^{j=2N_{BC}} \binom{j-1}{N_{BC}-1} {}_2F_3(\mathbf{a}_j; \mathbf{b}_j; p_A p_H (\lambda_T t)^2) \\ &+ \left(\frac{p_H}{p_A}\right)^{N_{BC}} e^{-\lambda_T t} \left(p_A \lambda_T \frac{(p_A \lambda_T t)^{N_{BC}-1}}{(N_{BC}-1)!} e^{p_A \lambda_T t} - \sum_{i=N_{BC}}^{2N_{BC}} \binom{i-1}{N_{BC}-1} p_A^i \frac{\lambda_T^i t^{i-1}}{(i-1)!} \right) \\ &= \frac{p_A \lambda_T e^{-\lambda_T t} (p_A p_H (\lambda_T t)^2)^{N_{BC}}}{(2N_{BC})!} \cdot \sum_{j=N_{BC}}^{j=2N_{BC}} \binom{j-1}{N_{BC}-1} {}_2F_3(\mathbf{a}_j; \mathbf{b}_j; p_A p_H (\lambda_T t)^2) \\ &+ \left(\frac{p_H}{p_A}\right)^{N_{BC}} e^{-\lambda_T t} \left(p_A \lambda_T \frac{(p_A \lambda_T t)^{N_{BC}-1}}{(N_{BC}-1)!} e^{p_A \lambda_T t} - \frac{1}{(N_{BC}-1)!} \sum_{i=N_{BC}}^{2N_{BC}} p_A^i \frac{\lambda_T^i t^{i-1}}{(i-N_{BC})!} \right). \end{aligned} \tag{A40}$$

We obtain Equation (20) by rearranging the indices of the summations and the order of operands. \square

Appendix D

Comparison of Attack Success Probabilities of DS Attack and Pre-Mining Attack

In [12], a special case of pre-mining strategy has been considered, where the condition for a DS attack success was different from Definition 1. Specifically, the only condition was to have the fraudulent chain to grow longer than the honest chain by N_{BC} , i.e., $A(t) > H(t) + N_{BC}$ (see Section 7 of [12]). We refer to $\mathbb{P}_{\text{pre-mine}}$ as the probability of satisfying this condition. The literature has shown that satisfying this condition suffices a success of DS attack [12]. What they have not shown, however, is that this condition is not a necessary one. Thus, we here aim to show that their condition is indeed not a necessary condition, by showing that $\mathbb{P}_{DSA} > \mathbb{P}_{\text{pre-mine}}$ for all $p_A \in (0, 0.5)$. First, it has been given that $\mathbb{P}_{\text{pre-mine}} = (p_A/p_H)^{N_{BC}+1}$. Under the condition of [12], it is required that the fraudulent chain catches up with the honest chain with additional N_{BC} blocks. The catch-up probability has been derived by Nakamoto in [1] using the gambler’s ruin approach as $(p_A/p_H)^k$, where k is the number of blocks that the honest chain leads by at the initial status. Next, we refer to an intermediate step in the derivation of \mathbb{P}_{DSA} by Rosenfeld [6]:

$$\mathbb{P}_{DSA} = \sum_{k=0}^{N_{BC}+1} \binom{N_{BC}+k-1}{k} p_H^{N_{BC}} p_A^k \left(\frac{p_A}{p_H}\right)^{N_{BC}-k+1} + \sum_{k=N_{BC}+2}^{\infty} \binom{N_{BC}+k-1}{k} p_H^{N_{BC}} p_A^k. \tag{A41}$$

Finally, clear inequalities can be used to show $\mathbb{P}_{DSA} > \mathbb{P}_{\text{pre-mine}}$:

$$\begin{aligned} \mathbb{P}_{DSA} &> \sum_{k=0}^{N_{BC}+1} \binom{N_{BC}+k-1}{k} p_H^{N_{BC}} p_A^k \left(\frac{p_A}{p_H}\right)^{N_{BC}-k+1} + \sum_{k=N_{BC}+2}^{\infty} \binom{N_{BC}+k-1}{k} p_H^{N_{BC}} p_A^k \left(\frac{p_A}{p_H}\right)^{N_{BC}+1} \\ &> \left(\frac{p_A}{p_H}\right)^{N_{BC}+1} \sum_{k=0}^{\infty} \binom{N_{BC}+k-1}{k} p_H^{N_{BC}} p_A^k \\ &= \left(\frac{p_A}{p_H}\right)^{N_{BC}+1} = \mathbb{P}_{\text{pre-mine}}. \end{aligned} \tag{A42}$$

For numerical example, when $p_A = 0.35$ and $N_{BC} = 5$ the probabilities can be computed as $\mathbb{P}_{DSA} = 0.2287$ and $\mathbb{P}_{\text{pre-mine}} = 0.0244$. As the gap is significant, it is shown that the DS attack success condition defined in [12] was indeed only a sufficient condition, set to be too strict.

Appendix E

Generalized Hypergeometric Function

We define generalized hypergeometric series and generalized hypergeometric functions [28].

For a variable z and a given set of coefficients $\beta_0, \dots, \beta_\infty$, if the ratio of coefficients b_n can be expressed in terms of two polynomials $A(n)$ and $B(n)$ in n as follows,

$$\frac{\beta_{n+1}}{\beta_n} = \frac{A(n)}{B(n)(n+1)} \tag{A43}$$

for all integer $n \geq 0$, a power series $\sum_{n \geq 0} \beta_n z^n$ is a generalized hypergeometric series, where the polynomials are in the forms of

$$A(n) = c(a_1 + n) \cdots (a_p + n) \tag{A44}$$

and

$$B(n) = d(b_1 + n) \cdots (b_q + n), \tag{A45}$$

for real numbers c and d and complex numbers a_1, \dots, a_p and b_1, \dots, b_q . The generalized hypergeometric series is denoted by

$${}_pF_q(\mathbf{a}; \mathbf{b}; z) := \sum_{n \geq 0} \beta_n z^n, \tag{A46}$$

where \mathbf{a} and \mathbf{b} are the vectors of a_1, \dots, a_p and b_1, \dots, b_q , respectively.

A generalized hypergeometric series can be a generalized hypergeometric function, if it converges. If $p < q + 1$, the ratio Equation (A43) goes to zero as $n \rightarrow \infty$. This implies the series Equation (A46) converges for any finite value z and thus can be defined as a function.

References

1. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: <https://bitcoin.org/bitcoin.pdf> (accessed on 26 November 2020).
2. Ritzdorf, H.; Soriente, C.; Karame, G.O.; Marinovic, S.; Gruber, D.; Capkun, S. Toward Shared Ownership in the Cloud. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 3019–3034. [CrossRef]
3. Wu, S.; Chen, Y.; Wang, Q.; Li, M.; Wang, C.; Luo, X. CReam: A Smart Contract Enabled Collusion-Resistant e-Auction. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 1687–1701. [CrossRef]
4. Nguyen, G.-T.; Kim, K. A Survey about Consensus Algorithms Used in Blockchain. *J. Inf. Process. Syst.* **2018**, *14*, 101–128. [CrossRef]
5. Sompolinsky, Y.; Zohar, A. *Secure High-Rate Transaction Processing in Bitcoin*; Böhme, R., Okamoto, T., Eds.; Springer: Berlin/Heidelberg, Germany, 2015; pp. 507–527.
6. Rosenfeld, M. Analysis of Hashrate-Based Double Spending. *arXiv* **2014**, arXiv:1402.2009 [cs].
7. Beikverdi, A.; Song, J. Trend of centralization in Bitcoin’s distributed network. In Proceedings of the 2015 IEEE/ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Takamatsu, Japan, 1–3 June 2015; pp. 1–6.

8. Gervais, A.; Karame, G.O.; Capkun, V.; Capkun, S. Is Bitcoin a Decentralized Currency? *IEEE Secur. Priv.* **2014**, *12*, 54–60. [[CrossRef](#)]
9. Attah, E. Five most prolific 51% attacks in crypto: Verge, Ethereum Classic, Bitcoin Gold, Feathercoin, Vertcoin. CryptoSlate. Available online: <https://cryptoslate.com/prolific-51-attacks-crypto-verge-ethereum-classic-bitcoin-gold-feathercoin-vertcoin/> (accessed on 26 November 2020).
10. Bonneau, J. *Why Buy When You Can Rent? Bribery Attacks on Bitcoin Consensus*; Springer: Berlin, Germany, 2016.
11. Sayeed, S.; Marco-Gisbert, H. Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack. *Appl. Sci.* **2019**, *9*, 1788. [[CrossRef](#)]
12. Sompolinsky, Y.; Zohar, A. Bitcoin's Security Model Revisited. *arXiv* **2016**, arXiv:1605.09193 [cs].
13. Bissias, G.; Levine, B.N.; Ozisik, A.P.; Andresen, G. An Analysis of Attacks on Blockchain Consensus. *arXiv* **2016**, arXiv:1610.07985 [cs].
14. Zaghoul, E.; Li, T.; Mutka, M.W.; Ren, J. Bitcoin and Blockchain: Security and Privacy. *IEEE Internet Things J.* **2020**, *7*, 10288–10313. [[CrossRef](#)]
15. Budish, E.B. The Economic Limits of Bitcoin and the Blockchain. *SSRN J.* **2018**. [[CrossRef](#)]
16. Gervais, A.; Karame, G.O.; Wüst, K.; Glykantzis, V.; Ritzdorf, H.; Capkun, S. On the Security and Performance of Proof of Work Blockchains. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security—CCS'16, Vienna, Austria, 24–28 October 2016; pp. 3–16.
17. Ramezan, G.; Leung, C.; Jane Wang, Z. A Strong Adaptive, Strategic Double-Spending Attack on Blockchains. In Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, 30 July–3 August 2018; pp. 1219–1227.
18. Pinzón, C.; Rocha, C. Double-spend Attack Models with Time Advantage for Bitcoin. *Electron. Notes Theor. Comput. Sci.* **2016**, *329*, 79–103. [[CrossRef](#)]
19. Goffard, P.-O. Fraud risk assessment within blockchain transactions. *Adv. Appl. Probab.* **2019**, *51*, 443–467. [[CrossRef](#)]
20. Karame, G.O.; Androulaki, E.; Roeschlin, M.; Gervais, A.; Čapkun, S. Misbehavior in Bitcoin: A Study of Double-Spending and Accountability. *ACM Trans. Inf. Syst. Secur.* **2015**, *18*, 2:1–2:32. [[CrossRef](#)]
21. Karame, G.O.; Androulaki, E.; Capkun, S. Double-spending fast payments in bitcoin. In Proceedings of the 2012 ACM Conference on Computer and Communications Security—CCS '12, Raleigh, NC, USA, 16–18 October 2012; p. 906.
22. Papoulis, A.; Pillai, S.U. Random walks and other applications. In *Probability, Random Variables and Stochastic Processes*; McGraw-Hill Europe: Boston, MA, USA, 2002; ISBN 978-0-07-122661-5.
23. Bowden, R.; Keeler, H.P.; Krzesinski, A.E.; Taylor, P.G. Block arrivals in the Bitcoin blockchain. *arXiv* **2018**, arXiv:1801.07447 [cs].
24. Flajolet, P.; Sedgewick, R. Combinatorial structures and ordinary generating functions. In *Analytic Combinatorics*; Cambridge University Press: Cambridge, UK, 2009; ISBN 978-1-139-47716-1.
25. Conti, M.; Sandeep Kumar, E.; Lal, C.; Ruj, S. A Survey on Security and Privacy Issues of Bitcoin. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 3416–3452. [[CrossRef](#)]
26. Wilf, H.S. Analytic and asymptotic methods. In *Generatingfunctionology*, 3rd ed.; A K Peters/CRC Press: Wellesley, MA, USA, 2005; ISBN 978-1-56881-279-3.
27. Wilf, H.S. Introductory ideas and examples. In *Generatingfunctionology*, 3rd ed.; A K Peters/CRC Press: Wellesley, MA, USA, 2005; ISBN 978-1-56881-279-3.
28. Gasper, G.; Rahman, M. Basic Hypergeometric series. In *Basic Hypergeometric Series*; Encyclopedia of Mathematics and Its Applications; Cambridge University Press: Cambridge, UK, 2004; Volume 96, ISBN 978-0-521-83357-8.
29. Flajolet, P.; Sedgewick, R. Labelled structures and exponential generating functions. In *Analytic Combinatorics*; Cambridge University Press: Cambridge, UK, 2009; ISBN 978-1-139-47716-1.


Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

ECCPoW: Error-Correction Code based Proof-of-Work for ASIC Resistance

Hyunjun Jung ¹  and Heung-No Lee ^{2,*}

¹ Blockchain Internet Economy Research Center, Gwangju Institute of Science and Technology, Gwangju 61005, Korea; junghj85@gist.ac.kr

² School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, Korea

* Correspondence: heungno@gist.ac.kr; Tel.: +82-62-715-2237

Received: 20 May 2020; Accepted: 5 June 2020; Published: 9 June 2020



Abstract: Bitcoin is the first cryptocurrency to participate in a network and receive compensation for online remittance and mining without any intervention from a third party, such as financial institutions. Bitcoin mining is done through proof of work (PoW). Given its characteristics, the higher hash rate results in a higher probability of mining, leading to the emergence of a mining pool, called a mining organization. Unlike central processing units or graphics processing units, high-cost application-specific integrated circuit miners have emerged with performance efficiency. The problem is that the obtained hash rate exposes Bitcoin's mining monopoly and causes the risk of a double-payment attack. To solve this problem, we propose the error-correction code PoW (ECCPoW), combining the low-density parity-check decoder and hash function. The ECCPoW contributes to the phenomenon of symmetry in the proof of work (PoW) blockchain. This paper proposes the implementation of ECCPoW, replacing the PoW in Bitcoin. Finally, we compare the mining centralization, security, and scalability of ECCPoW and Bitcoin.

Keywords: error-correction codes proof-of-work (ECCPoW); proof-of-work (PoW); ECCPoW implementation; ASIC resistance

1. Introduction

We use digital signatures from third-party trust agencies to promote trust in Internet commerce. We warrant proof of data forgery using middlemen. Satoshi Nakamoto proposed an electronic money system without a middleman in a peer-to-peer (P2P) network through a Bitcoin white paper [1]. Bitcoin applies blockchain technology to electronic monetary systems to guarantee transactions without intermediaries (e.g., banks). Blockchain is a ledger management technology based on a distributed computing technology that cannot be arbitrarily modified by storing the transaction content in a chain-based distributed data storage environment in the form of a block [2,3].

The blockchain stores the same ledger on a global network and is designed to pay certain rewards to maintain the block. This is called mining, and mining creates blocks and obtains cryptocurrency by executing a hash function. Miners belong to mining pools because of the probability and convenience of being rewarded for mining cryptocurrency [4,5].

The hash rate is the number of hash values calculated per second as a measure of computational processing power for mining cryptocurrency. The hash rate of cryptocurrency is determined by the total number of the participating nodes. Most miners belong to a hash pool and occupy a high percentage of the hash rate. We should be concerned about the risk of a double-payment attack if the mining pool in the

blockchain accounts for a high percentage of the total hash rate of cryptocurrency [6]. A double-payment attack occurs when the mining pool seizes at least 51% of the total hash rate to determine the branch of the blockchain to the desired side. Recent studies have shown that a double-payment attack can be made to benefit from a low share of the hash rate [7].

Bitcoin miners receive blockchain information (version, previous block hash, Merkle root, bits, and others) and execute hash functions (e.g., the secure hash algorithm (SHA 256)). Miners create a block if the output value of the hash function is less than the target level of difficulty. Currently, Bitcoin requires an increasingly high operation to create blocks. Miners purchase application-specific integrated circuits (ASICs) with high per-second computing power to mine Bitcoin and join the mining pool. Low-power miners using central or graphics processing units (CPUs or GPUs) have difficulty mining.

Bitcoin uses the number of zeros in the front digits of the output from the SHA 256 function to generate blocks. The mining difficulty increases with the number of zeros. Miners buy ASICs to compute SHA functions quickly. The mining machine Antminer S9 (13,000 MH/s) is approximately 8800 times faster than GTX1060 (1478 MH/s). The recently released S19 has a speed of 95 TH/s, and S19Pro exhibits performance of 110 TH/s [8]. Miners using CPUs or GPUs in Bitcoin, and miners using ASIC chips do not have equal chances for success because ASIC miners have an 8800-fold greater chance of success in mining.

Blockchain was proposed to allow nodes to participate as miners freely and to share mining rewards fairly. Blockchain is not free to participate in as a miner. The participating miners compete against equity. Several methods have been proposed to curb the development of ASIC miners, but in the end, these methods have not prevented ASIC development. As a new mining function to prevent the development of ASIC miners, we proposed the error-correction code proof of work (ECCPoW) concept, which combines the low-density parity-check (LDPC) decoder and hash function [9]. In addition, we analyzed the hash cycle of ECCPoW and demonstrated that it could be used for blockchain mining.

This paper contributes to the phenomenon of symmetry in the proof of work (PoW) blockchain. The PoW blockchain tends to increase the hash rate along with the total size of the blockchain. The size of the hash rate is related to the computing power required by block generation. Some people take issue with the large amount of wasted power used to create the PoW blockchain. However, the PoW blockchain guarantees high stability with the power required to create the block. The transition to other PoWs due to problems in the PoW blockchain causes dangerous problems. Thus, ECCPoW mitigates the power problems in the PoW blockchain and helps reduce the symmetry of the hash rate, which increases in proportion to the size of the blockchain in the PoW blockchain.

The contribution of this paper is two-fold. It introduces the proposed ECCPoW and proposes an implementation method. The second contribution is to introduce the process of experimenting in Bitcoin by replacing the SHA 256 function with the ECCPoW function. This paper proposes the creation of a cryptographic puzzle that changes every block and shows how to apply the crypto puzzle decoder to the solution. We present the implementation of the proposed method by replacing ECCPoW in Bitcoin. We also measure the block generation time of the ECCPoW. Finally, we compare ECCPoW and Bitcoin by implementing them in the same environment.

This study contributes to the previous literature on ASIC resistance in PoW blockchain. Previous studies have used forced memory access, leading to the inefficient behavior of ASICs. Another method is to make ASICs challenging to produce using various hash functions. The last method is to change the existing hash function to another hash function when ASICs are released. Therefore, ECCPoW combines the LDPC decoder with a hash function to create the effect of releasing different hash functions in each block. Implementing ASICs is difficult for the LDPC decoder due to cost issues. Furthermore, the PoW blockchain without ASICs reduces the mining centralization. In addition, ECCPoW can reduce power consumption by maintaining the advantages of the PoW.

This paper is structured as follows. Section 2 introduces the relevant research on the development of the prevention of ASIC miners. Section 3 presents ECCPoW and development methods. Section 4 reveals the results of the experiment by loading the ECCPoW into Bitcoin. Section 5 evaluates the mining centralization, security, and scalability of the ECCPoW. Finally, Section 6 presents the conclusion of the paper.

2. Related Work

2.1. Ethereum

Ethereum is a distributed computing platform developed to implement smart contract functions based on blockchain technology. In July 2015, Vitalik Buterin developed Ethereum in the C++ and Go languages. Ethereum uses the Ethash algorithm. Ethereum plans to change from a PoW to a proof of stake in the future [10]. Ether Solarium is against the use of the nonlinear directed acyclic graph (DAG) in ASICs. The initial size of the DAG was about 1 GB, and it was designed to increase in size linearly over time. In October 2019, the size of the DAG was 3.99 GB, which was maintained until December 20, 2020 [11,12].

Ethereum approved the application of programmatic proof of work (ProgPoW) [13] to respond to the centralization of mining in ASIC in 2019. First, ProgPoW regularly changes mining problems to a problem in which GPUs can adapt quickly. Second, ProgPoW makes the most of all the components of the graphics card for mining. Moreover, ProgPoW uses randomly generated problems based on block numbers and is designed for the efficient operation of GPUs. This reduces performance differences compared to ASICs.

2.2. The X-11 Series

The X-11 series is a cryptocurrency mining algorithm that uses as many hash functions as the number indicated behind the X [14]. The X-11 series used hash functions to add depth and complexity to curb ASICs. The X-11 series connects several hash functions and uses the output value of the hash as the input value of the next hash. Typically, the algorithm is used in Dash. The concept of the X-11 series uses multiple hashes to increase security and prevent ASIC mining. Currently, however, the X-11 series has been upgraded to increase the number of hash functions, such as X-13, X-14, X-15, X-16R, and X-17 because ASIC mining is still possible.

2.3. CryptoNote

CryptoNote was designed to be more inefficiently executed with a GPU than a CPU [15] to prevent ASIC mining. The performance of CryptoNote is susceptible to memory latency because memory creation and subsequent read operations occur repeatedly. This is similar to Ethereum's Ethash function. CryptoNote creates blocks by determining the hash function to be used after memory-intensive tasks.

Despite such attempts, Bitmain released ASICs optimized for CryptoNote algorithms in March 2018. Monero uses CryptoNote to change its mining algorithm twice a year to prevent ASICs. Monero's algorithm change has prevented the use of ASICs. However, the frequent hard fork execution (radical changes) caused participants to break away from mining. In the end, there was a risk of the centralization of mining. To prevent frequent hard forks, RandomX proposed a key block concept that periodically changes mining methods [16].

3. The Proposed Method

This chapter provides an overview of and the implementation method of the proposed ECCPoW. We proposed the concept of ECCPoW to increase the resistance to ASIC, using a hash function that makes ASIC development difficult. Moreover, ASIC resistivity research reviews methods of inducing the loading

of memory; for example, Ethereum uses several hash algorithms, such as X-11. However, ASICs for Ethereum and the X-11 series have been developed. Thus, ECCPoW is a method for miners to release different hash functions for each block to curb the emergence of ASICs.

The ECCPoW can help ease the centralization of mining. The conversion to ASICs in mining is made by ordinary people to avoid being excluded from mining and by a small group of people with capital power to monopolize mining. Mining centralization of a small group is likely to lead to mining blocks for malicious purposes and forging and tampering with the mined blocks. The implementation of ASICs in the LDPC decoder lacks flexibility due to structural cost issues [17]. Moreover, ECCPoW proposes a method of combining the LDPC decoder with a hash function. We help reduce the risk in the blockchain by mitigating mining centralization.

3.1. ECCPoW Overview

The ECCPoW is a POW that uses the ECC decoder that is used in communication, which can be implemented using ASICs. As a simple example, cell phones use ASICs to implement an ECC decoder quickly and at low power. The parity-check Matrix H determines the design of the ECC decoder based on ASICs. In other words, ASIC equipment can produce a decoder using parity-check matrices. For mobile phones, ASICs have standardized parity-check matrices that allow an ECC decoder design. Building ASICs to match the decoder supporting countless parity-check matrices is difficult due to cost problems and decoder size problems.

Randomly, ECCPoW changes each block parity-check matrix (i.e., ECCPoW uses an infinite number of parity-check matrices). As a result, ECCPoW inhibits the development of ASICs for ECC decoders. The ECC decoding algorithm only runs on a CPU or GPU. For example, even if the SHA function used in the PoW is executed quickly, a bottleneck occurs in the execution of ECC decoding algorithms.

3.2. Create a Cryptographic Puzzle That Changes Every Block

The ECCPoW aims to create a cryptographic puzzle that changes every block. We changed the composite function used to create the cryptographic puzzle using the generation method by Gallager [18] and the previous hash value. In other words, ECCPoW randomly generates an LDPC matrix used by the decoder of the composite function. The Gallager method requires variables. Table 1 displays the definition of the variables used in the LDPC.

Table 1. Variables in the low-density parity-check Matrix H.

Variables	Definition
n	Number of columns in H
m	Number of rows in H
w_c	Number of 1s in each column
w_r	Number of 1s in each row

The LDPC matrix satisfies the equation $nw_c = (n - k)w_r$, where k is $n - m$ and 2^k is the total number of symbols that can be generated. When the variables are given, the LDPC Matrix H of size A is generated by the following method:

Step 1: Create a partial matrix of size $\frac{m}{w_c} \times n$

$$A_1 := \begin{bmatrix} \underbrace{1\ 1\ \dots\ 1}_{w_r} & & & \\ & \underbrace{1\ 1\ \dots\ 1}_{w_r} & & \\ & & \ddots & \\ & & & \underbrace{1\ 1\ \dots\ 1}_{w_r} \end{bmatrix} \in \{0, 1\}^{\frac{m}{w_c} \times n}$$

Step 2: Generate $w_c - 1$ submatrices by randomly permutating the following matrices:

$$A_i := \prod_i i(A_i) \in \{0, 1\}^{\frac{m}{w_c} \times n},$$

where $\prod_i i$ is the i th sequence, and $i = 2, 3, \dots, w_c$.

Step 3: Construct the final LDPC matrix using all submatrices above:

$$H := \begin{bmatrix} A_1^T & A_2^T & \dots & A_{w_c}^T \end{bmatrix} \in \{0, 1\}^{m \times n}.$$

Moreover, ECCPoW changes the sequence of permutations through the previous hash values and uses the previous hash value as the seed value to determine the sequence of permutations. The sequence of permutations is random because the hash values are random. The code is implemented in Reference [19]. Table 2 compares Matrix H produced using different hash values. The lower part of the generated Matrix H in Table 2 is different.

Table 2. Form of the resulting Matrix H using different hash values.

Generated Matrix H $n = 24\ m = 16$ $w_c = 3\ w_r = 4$	<pre>[1 1 1 1 0] [0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1] [0 1 1 0 0] [0 1 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1] [0 0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1] [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 0 0] [1 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0] [0 0 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1] [0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0] [0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0] [0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0] [1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0]</pre>
Previous hash value	0x00000000000000000000000000000001
Generated Matrix H $n = 24\ m = 16$ $w_c = 3\ w_r = 4$	<pre>[1 1 1 1 0] [0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0] [1 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0] [0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0] [0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0 0 0 0 0] [0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1] [0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0 0 0 0 0] [0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1] [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0] [1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0] [0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0]</pre>
Previous hash value	0x00000000000000000000000000000002

3.3. Crypto Puzzle Decoder That Changes Every Block

The LDPC decoder of ECCPoW was developed using a message-passing algorithm. The decoder receives an $m \times n$ LDPC matrix of hash values $r \in \{0, 1\}^n$ of length n as input values. The decoder outputs a

value $c \in \{0, 1\}^n$ of length n . The decoder can produce two types of answers depending on the input hash value r . The decoder outputs a sign $D_{MP} : \{r, H\} \mapsto c_i$ if the entered hash value r satisfies $\|r - c_i\|_h \leq t$ for any sign c_i , where t is the value determined by the LDPC matrix. If not satisfied, the decoder outputs a random vector $c \in \{0, 1\}^n$. The code is implemented in Reference [19].

The two conditions for determining whether to solve a cryptographic puzzle are listed in Table 3. Condition 1 determines that the cryptographic puzzle has been solved if the decoder’s output value c satisfies the conditions. In Condition 1, the output value is the code. Condition 1 is possible because the output value is less likely to code when the value is any input to the decoder. For example, there is a low probability of finding an answer to any input in SHA 256. In Condition 2, the Hamming weight of the output value is an element of the given set S. Set S is the range value of the output value. Condition 2 occurs because the Hamming weights of the possible codes may differ when Matrix H is given.

Table 3. Conditions for the determination of crypto puzzle resolution.

Condition 1	(Original method) If the result of the decoder is code and has a specific Hamming weight, the problem is solved.
Condition 2	(Existing proof of work) If the result of rehashing the result of the decoder is less than a specific value, the problem is solved.

The satisfactory probability of Condition 1 requires a minimum Hamming distance value of H . To calculate this value, all distinct codes in 2^k must be considered, which is possible when the number of codes is small, but it is impossible when the number is large. Litsyn [20] reported the upper and lower bounds of the minimum Hamming distance value of H at specific w_c and w_r values. Table 4 reveals the probability of finding the codes according to the variables in the LDPC matrix. In this way, the upper bound of the probability is small. This makes it less likely for the decoder to meet Condition 1 when any value is entered.

Table 4. Probability of finding a code according to the variables of the low-density parity-check matrix.

$w_c = 4, w_r = 5$	p_1 Upper Bounds	p_1 Lower Bounds
$n = 80, k = 12$	6.32×10^{-5}	2.12×10^{-8}
$n = 120, k = 24$	1.65×10^{-8}	1.49×10^{-13}
$n = 160, k = 32$	4.06×10^{-10}	1.34×10^{-17}

Condition 2 is used to increase the difficulty of cryptographic puzzles when variables $n, m, w_c,$ and w_r are fixed. Table 5 displays the probability that Condition 2 is satisfied when given a set S and part of the distribution of Hamming weights of the codes that can be generated when $n = 256, m = 192,$ and $w_c, w_r = 5$ are given.

Table 5. The probability that Condition 2 is satisfied.

Hamming Weight	Probability	Element of the Set S	Probability that Condition 2 is Satisfied
98	$\approx 5 \times 10^{-5}$	98	$\approx 5 \times 10^{-5}$
...
128	$\approx 9.7 \times 10^{-2}$	98, 100, ..., 126	$\approx 4 \times 10^{-1}$
128	$\approx 1 \times 10^{-1}$	98, 100, ..., 126, 128	$\approx 5 \times 10^{-1}$

The probability of satisfying both Conditions 1 and 2 is as follows:

$$p := Pr\{c|Hc = 0\} \times Pr\{\|c\|_h \in S\}.$$

We produced the difficulty table, as shown in Table 6, by calculating the probability of meeting Conditions 1 and 2 at the same time when the variables n , w_c , and w_r and the set S are given. The probability value p in Table 6 is the difficulty level of the cryptographic puzzle. The closer the probability value is to zero, the higher the difficulty of the cryptographic puzzle.

Table 6. Difficulty table for ECCPoW.

Lv.	n	w_c	w_r	Set S	p
1	32	3	4	{10, 12, ..., 20, 22}	$\approx 3.07 \times 10^{-5}$
2	32	3	4	{10, 12, ..., 14, 16}	$\approx 2.02 \times 10^{-5}$
...					
379	128	3	4	{34, 94}	$\approx 5.12 \times 10^{-23}$
380	128	3	4	{34}	$\approx 2.60 \times 10^{-23}$

Condition 2 determines the resolution of the cryptographic puzzle by comparing the output value of the decoder with the result value and comparing the nonce with the target. Figure 1 illustrates Condition 2 for determining whether ECCPoW cryptographic puzzles are resolved. If the composite function and the hash algorithm are recognized as one hash function, Figure 1 has the same structure as Bitcoin, so the difficulty control function of Bitcoin can be used.

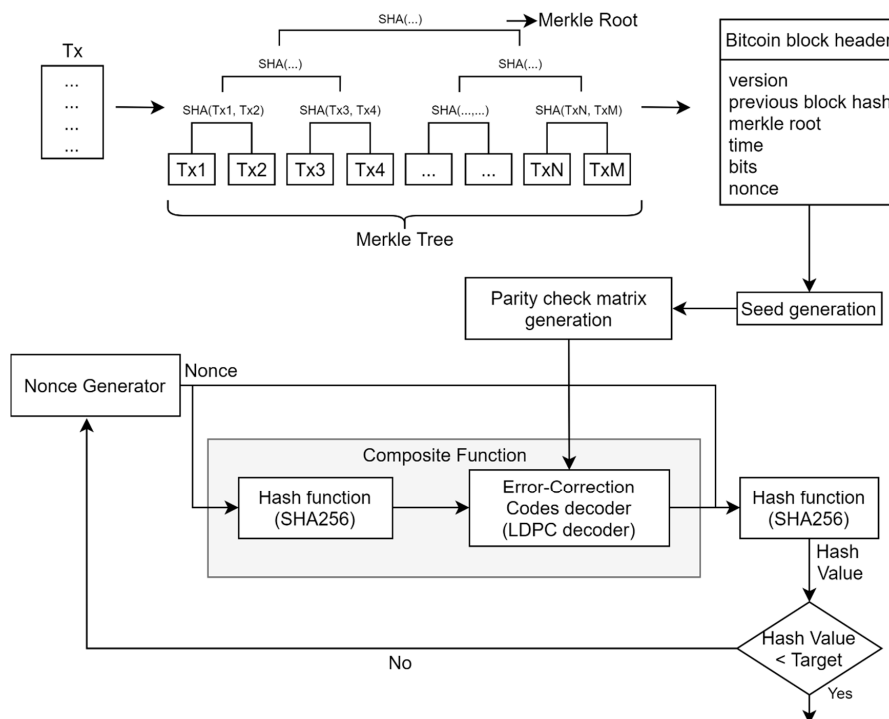


Figure 1. Condition 2 for determining the error-correction codes proof-of-work (ECCPoW) crypto puzzle resolution.

4. Experiment

This chapter presents experiments to verify ECCPoW. In the single-node experiment, the Bitcoin consensus algorithm was replaced by ECCPoW to verify the block generation function. In the multi-node experiment, we experimented with checking whether block generation, block synchronization, and transaction creation and transmission were performed correctly in a multi-node environment. In addition, the block generation time was tested in Bitcoin and Ethereum.

4.1. ECCPoW Operation Single/Multiple Node Experiment

We replaced Bitcoin's consensus algorithm with ECCPoW. The single-node experiment is a block generation experiment of the ECCPoW blockchain. Bitcoin uses the "generatetoaddress" command for block generation and receives the current address of the blockchain as a parameter. Then, a new block address is created using the "getnewaddress" command, and 10 blocks are created using "generatetoaddress." Figure 2 illustrates the change in 10 of the "blocks" after block generation.

Multiple node experiments use three random nodes (Nodes 1, 2, and 3) to experiment with block synchronization, to transmit transactions, and to verify them. This experiment demonstrates that each node mines different numbers of blocks, synchronizing to a long blockchain. Figure 3 indicates the results of mining and checking the blocks in Node 1.

Figure 4 reveals the results of mining and checking blocks on Node 2, and Figure 5 demonstrates the results of mining and checking blocks on Node 3.

Node 3 connects Node 1 (192.168.232.128) and Node 2 (192.168.232.129) using the "addnode" command. Then, Node 3 uses the "getadnednodeinfo" command to check the information on the connected nodes. Figure 6 illustrates the connection between Node 1 (192.168.232.128) and Node 2 (192.168.232.129) using the "addnode" command. Next, Figure 6 indicates the connected node information using the "getadnednodeinfo" command. Bitcoin's nodes synchronize to nodes that hold many blocks, and Bitcoin determines that nodes with many blocks are highly reliable. In the current experiment, Node 3 had the largest number of blocks. Nodes 1 and 2 synchronize to the blocks of Node 3, and these results are displayed in Figure 7.

```

getnewaddress
3KNswAYG9vsMt3t4pCWiEFf2uHAVfDGxr7

generatetoaddress 10 3KNswAYG9vsMt3t4pCWiEFf2uHAVfDGxr7
[
  "96353bf229725fa57cef00a4f26fe4d509349d24bb94ba30370b957b99cb75a0",
  "f6089b0eacc1f88f970596f890b0e4135b0b4a67d41107a79c20bd06f90efde",
  "ef387d88b3bcb7fcd86c437f8a99be75e7e7f74e178ee5b5461dc39d92823f5db",
  "8f0081fa5abdb1ca4565d6897a2211f56d88e42ba774d88d2c85e3d3b93906f",
  "66c4a012d78fa1739c1f0b317e52015837351a0d40e934855c819e50f265e4",
  "20614bde4880b4d57736ecc6ca562d4e84107c9cc1a6588f26fd81fae91b62f48",
  "f3c092a44d247d74ce956bbcb49ad69ff77fe671272bcffaf8719850c59e44c69",
  "b46ce4217afe044a4ae039e3f95731749ff366a73056f9097534181826511fc",
  "d459d9c3a8f0be125b92027d01b3f889ca6344e66a23e53c958f43221d42cf05",
  "6c904baee33423900ba1ebb81056b7f7fe45db8586744aa522f64cac51de42c6b"
]

getblockchaininfo
{
  "chain": "main",
  "blocks": 10,
  "headers": 10,
  "bestblockhash":
"6c904baee33423900ba1ebb81056b7f7fe45db8586744aa522f64cac51de42c6b",
  "difficulty": 4.523059468369196e+73,
  "mediantime": 1569819043,
  "verificationprogress": 1,
  "initialblockdownload": false,
  "chainwork":
"00000000000000000000000000000000000000000000000000000009fa612".

```

Figure 2. Results of block generation.

```

getnewaddress
3FUFZ8ShBQpCGqqkZoFErG2aGo28QwaUDT
generatetoaddress 10 3FUFZ8ShBQpCGqqkZoFErG2aGo28QwaUDT
[
  "2b2d7485a399d93169ff8532de059db263e0ba42db73847489d4620a7b7510bf",
  "099640a372ec928e5af6aa1b2032014b24a04225724c5553c1a80bf53dd0d8c4",
  "1c44c7078afcd449eb9ab8fd4cbc64419cca96afda91c571a51de163b0ac9ce",
  "630b1b3127c74b8892ff7fcbae89f2733aeaf57f225354a05f60583200ac8a2",
  "74449f452fd35732dbfab5e3ada8f531d7df4ee1a7df23ef41ab91ca1b4c3ca3",
  "282768f094a38a6f7113e4ffc45f5cb93ae7fb01f9e164bd8482f59df49712fff",
  "4a16965fc7efab971b99cb8d807eb78b17852fe4be55a620cf00ca865bbafc",
  "5ea006ef89893ab591ba39d995c2b09bcb5a5c21813a3c8ff9d7752765c820e7",
  "c43c6bb01c954f23b998f88ead512a7b7cbc1350e37225e6659f02f923ea8f17",
  "7a39ec86ae15a01d8d6930f2fcc5300d8830726fab16b4e733c59db1719f8e43"
]

getblockchaininfo
{
  "chain": "main",
  "blocks": 10,
  "headers": 10,
  "bestblockhash":
  "7a39ec86ae15a01d8d6930f2fcc5300d8830726fab16b4e733c59db1719f8e43",
  "difficulty": 4.523059468369196e+73,
  "mediantime": 1569822743,
}

```

Figure 3. Block mining and verification in Node 1.

```

getnewaddress
34rqCn3Xf6PDTLoHLZ8wPp1wsRV6Gb9KfY
generatetoaddress 20 34rqCn3Xf6PDTLoHLZ8wPp1wsRV6Gb9KfY
[
  "404a6788ab0dc5bd356b03b400c005fa8f49380ab978d4a7b3537dfc1f6ba23",
  "ca0404483d852a32abb8609073bbbab0d06c55c6c5b9e141de211d570f5576",
  "129b0e34f22c2f745efcd708df8b653ac295ac7b08b9bf36de518d880cfaa9a",
  "807f58aa7d221d15c6acc9e3b0ea89bc96de2205c797547d72d10ea1d463a588",
  "d9265ae393201c60e988cbeada6e82038e90997deadd64e5057d9c523d0abc3",
  "b518305ba918e8619c36c4d449e161d8e8ef9517bc2d2fd2cfaea92af4171008",
  "8d13e15542fda1e5950268f6d87b87a5ac737e4377231d06b79f683255d73869",
  "22c3b9e63d3b7f6fbc8959805fef052aeaf6b136124fe5799097d61c6b29b4c4",
  "5658489c42ca5aecf600e324a8a07e4c1a38b5f0ec8c6367f4c6941f031fd496",
  "43cd9632b9879fa290c3f6c2e63818189d84e75279123ced6bc58e29977450e",
  "2c65177825619bbebc75cacc0afcb2e58fa69ee178cb472eff29c2eda6ca9",
  "67e39e0482caafc683b8e57c6961fd7a4a3f376117c4394acd5ba1a84574235a",
]

getblockchaininfo
{
  "chain": "main",
  "blocks": 20,
  "headers": 20,
  "bestblockhash":
  "9e99fb700b43d5a59fa97a6d1550e536c3933c0899158d9300ea74ac50928279",
  "difficulty": 4.523059468369196e+73,
  "mediantime": 1569822820,
}

```

Figure 4. Block mining and verification in Node 2.

```

getnewaddress
3765hBQFFMn8FProyKFuF4vHKayafka4d
generatetoaddress 30 3765hBQFFMn8FProyKFuF4vHKayafka4d
[
  "20bbd5b26fd5b9530614279cf06a4e036056aa1b996d05d699f73d71b77311c9",
  "11902f08b5c19f9ce6551730311f076453ef620014f703643c9225d1279bf5ea",
  "0610136713ae049268c72dec6d32cd13346f12b9bcc20b1ae19d8d9492eeb955",
  "6848477bf878c2e38ff13d3aedcc047a8f68659da8454e1800ac963933c688c2",
  "1d20dd47c8400a5833fc84b4fe9d91d4fd6dc474fbaf939f45745121669d740c",
  "8f084bc0ea45ca54b81f0bc0d931591572209cf69210ccea9daf16edfc18ae56",
  "f8a52172a10c14283e9749c62aa9d4362d677f32d596b71e0ffced4dd4f4111f",
  "a0b47bde1bf2a11a35b0672e2a0be022eb94cd4dbb063bd35de44bc038cf0c43",
  "56221e5d9bc496a2cf2a61bc13efef967b87e74273e7dc6fbc3da4f7486a532",
  "a93231e200e63364d56b5669144f2353ac22b018c1835220ada57ea471d68f58",
  "0f4913a21d6481cf7f476952dd6e5b7fc8c188c420569fd4365f512a766ab3c",
  "06d84fa3aa0b97b32194a5ff9533847250261d567e33c7b2b199cee0781f3cda",
]

getblockchaininfo
{
  "chain": "main",
  "blocks": 30,
  "headers": 30,
  "bestblockhash":
  "0aaa7eb4a5bbd818e5a76383fd1f014df645ededd0ff9b568ee262b5080d7b5e",
  "difficulty": 4.523059468369196e+73,
  "mediantime": 1569822888,
}

```

Figure 5. Block mining and verification in Node 3.


```

addnode 192.168.232.128 add
null
addnode 192.168.232.129 add
null

getaddednodeinfo
[
  {
    "addednode": "192.168.232.128",
    "connected": true,
    "addresses": [
      {
        "address": "192.168.232.128:9777",
        "connected": "outbound"
      }
    ]
  },
  {
    "addednode": "192.168.232.129",
    "connected": true,
    "addresses": [
      {
        "address": "192.168.232.129:9777",
        "connected": "outbound"
      }
    ]
  }
]

```

Figure 6. Node connections in Node 3.

```

getblockchaininfo
{
  "chain": "main",
  "blocks": 30,
  "headers": 30,
  "bestblockhash":
  "0aaa7eb4a5bbd818e5a76383fd1f014df645ededd0ffb9568ee262b5080d7b5e",
  "difficulty": 4.523059468369196e+73,
  "mediantime": 1569822888,

```

Figure 7. Blockchain synchronization of Nodes 1 and 2.

We checked the behavior of the blockchain by checking the synchronization. We sent the transactions between linked Nodes 2 and 3. Figure 8 indicates the balances held by Nodes 2 and 3. Currently, Node 2 has zero, and Node 3 has 1000.

```

getbalance
0.00000000

getbalance
1000.00000000

```

Figure 8. Balances for Node 2 (top) and Node 3 (bottom).

Figure 9 displays the address of Node 2 (in the Pay-To field) and the amount of coin to send (in the Amount field) for Node 3 to transmit 500 coins to Node 2. A transaction fee was set to the minimum cost of 0.00001. Figure 10 reveals the transaction transfer by identifying the amount of coin in Node 2 and displaying the recent transaction records. Figure 11 illustrates that the number of coins in Node 3 and the recent transaction record decreased.

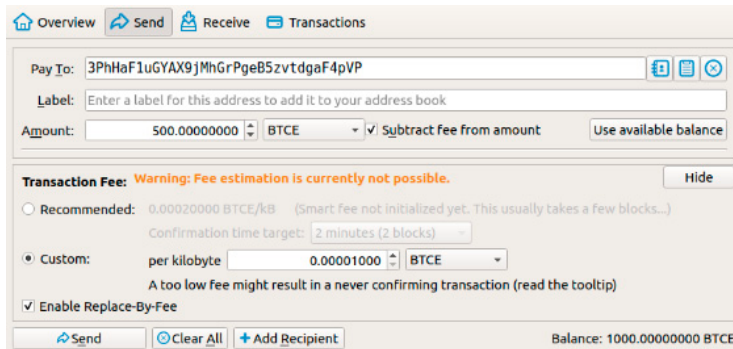


Figure 9. The input of the transaction transfer.

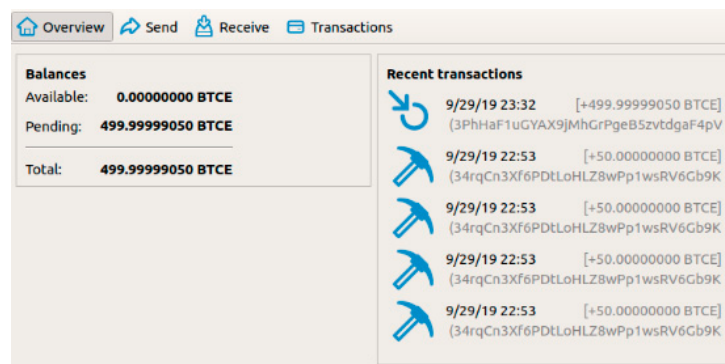


Figure 10. Balances and transaction logs for Node 2.

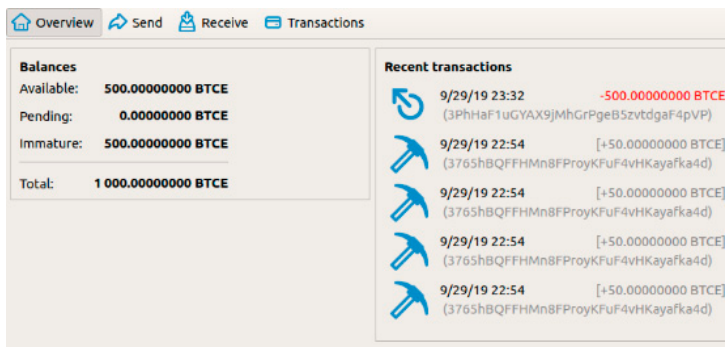


Figure 11. Balances and transaction logs for Node 3.

4.2. Block Generation Time

Figure 12 reveals the time-by-time block generation of Bitcoin with ECCPoW, using the difficulty table proposed in Section 3. Block generation time should be able to meet the target generation time for a stable blockchain. The blockchain adjusts the target block generation time by adjusting the difficulty level over time. The experimental environment used block generation with one node. As shown in Figure 12, the block generation time is unstable in the experiment. Sufficient nodes can confirm the stability of the block generation time. The difficulty table must be finely adjusted.

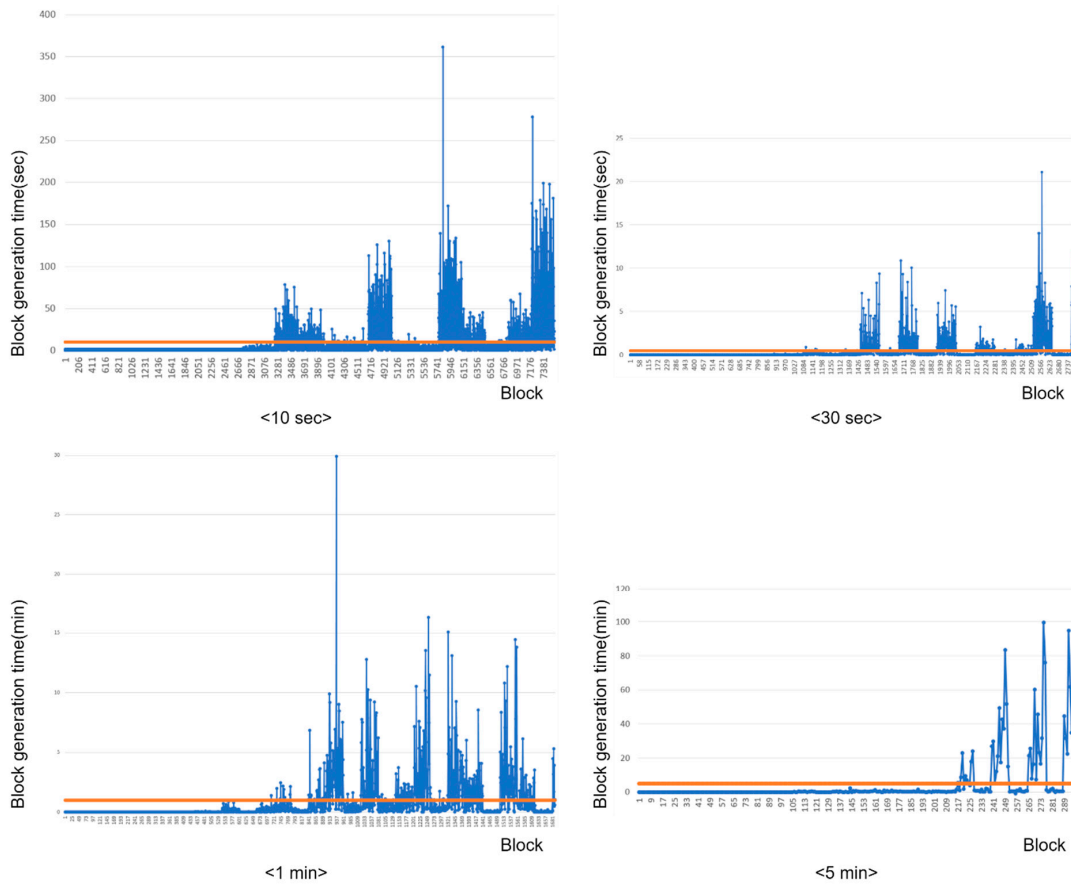


Figure 12. Block generation time of Bitcoin with ECCPoW.

5. Evaluation

We evaluated the mining centralization, security, and scalability of ECCPoW using Amazon Web Services. Table 7 presents the evaluation environment. A monitoring computer checked the network test results. The seed instance made the blockchain network connections between nodes. The mining instance was responsible for block mining. The implementation environment for each node was a Ubuntu Server 18.04 LTS, m5.large (vCPU processor 2 core with 8 GB of RAM, and a 20 GB solid-state drive).

Table 7. Implementation environment of the evaluation.

No	Role	CPU	Memory (GB)	HDD/SSD (GB)	Volume
1	Monitoring PC	Intel i7-8700 3.20 GHz	16	SSD 256	1
2	Seed Instance	AWS m5.xlarge vCPU 2	8	HDD 20	6
3	Mining Instance	AWS m5.xlarge vCPU 2	8	HDD 20	40

The blockchain trilemma problem is that trade-off relationships occur in the evaluation characteristics of the blockchain. The trilemma elements of a blockchain are decentralization, scalability, and security. Thus, we compared Bitcoin in terms of these elements (mining centralization instead of decentralization). Table 8 displays the evaluation standards and goals for evaluation.

Table 8. Description of evaluation features.

Evaluation Features	Unit of Measurement	Evaluation Standards	Evaluation Goal
Mining centralization	Distribution of mining success rate	40% (Estimate) (Bitcoin, Oct.–Dec. 2018)	40%
Security	Security of Bitcoin contrast	100% (Bitcoin)	100%
Scalability	Scalability of Bitcoin contrast	100% (Bitcoin)	100%

Bitcoin has a total hash rate of approximately 90 TH/s (March 2020). It is impossible to compare the current version of Bitcoin with ECCPoW. Moreover, ECCPoW (ver. 0.1.2) replaced the Bitcoin 0.17 version [21] of SHA 256 [22]. We compared two completely initialized blockchains (block count zero).

We compared the Bitcoin and ECCPoW initialized to set the evaluation environment (difficulty change cycle, target block generation time, 23 instances, among others). The blockchain typically sets a difficulty change cycle of 60 min and a target block creation time of 3 min for experimentation.

5.1. Mining the Centralization Evaluation

Mining centralization is when a network is no longer centralized and operates autonomously within a blockchain. We define mining centralization as when the nodes are fairly mined with the same performance. The indicators used in the mining centralization evaluation include the distribution of the mining success rates, which are defined by the formula below. According to the formula, the lower the dispersion of the probability of mining success, the higher the distribution of mining success. In other words, the distribution of mining success rates is higher for each participating node to have a uniform distribution of mining success rates, which means better dispersion. Bitcoin is estimated to have a 40% distribution of mining success rates (October to December 2018). Moreover, ECCPoW targets 40% dispersion:

$$A = \frac{C}{\sqrt{B + C^2}} \times 100A = \text{The distribution of mining success rate (\%)},$$

B = The dispersion of mining success rate,

C = Average of the mining success rate.

We created three seed nodes for the ECCPoW blockchain and composed 20 mining nodes. Table 9 lists the number of mining successes of each mining node at the time of 100 blocks being mined. Table 10 indicates the distribution of mining success.

The dispersion of ECCPoW was measured by the distribution of the mining success rate and measured at 92.22%, which was 32% higher than the assessment target of 60% (Table 10). The results of the experiment revealed that the ECCPoW miner is more likely to be rewarded than a Bitcoin miner in an ideal environment where participating nodes exhibit the same performance. In other words, ECCPoW is stronger in mining centralization than Bitcoin.

Table 9. The number of mining success and number of nodes.

Number of Mining Nodes	Number of Mining Success	Number of Mining Nodes	Number of Mining Success
1	4	11	7
2	9	12	4
3	3	13	12
4	4	14	5
5	6	15	11
6	4	16	2
7	6	17	7
8	6	18	6
9	5	19	10
10	5	20	8

Table 10. Evaluation results of the mining centralization evaluation features.

Total Number of Mining Successes	Average Mining Successes	Square of the Average Number	Dispersion of the Average Number	Distribution of Mining Successes
124	6.2	38.44	6.76	92.21944

5.2. Security Evaluation

Security is associated with the difficulty of the blockchain being altered by a miner's attack. A typical attack on the blockchain is the double-payment issue. One of the causes of double-payment problems occurs in the branch of the blockchain. An orphan chain is a chain that has a branch other than the main chain, and an orphan block is a block belonging to an orphan chain. We assess the security as the ratio of orphaned blocks to the total number of blocks in the blockchain. The security assessment calculates the orphan block ratio of Bitcoin and ECCPoW using the expression below. We evaluated the security of ECCPoW based on the security of Bitcoin at 100%.

$$\text{Orphan block ratio} = \frac{\text{number of orphan blocks}}{\text{Total Height of Blockchain}} \times 100$$

We configured the blockchain test environment with three seed nodes and 20 mining nodes. In addition, we mined 100 blocks in the blockchain and checked the orphan blocks. Figure 13 depicts the orphan block identification for the security assessment. After mining 40 blocks, the evaluation identified blocks for stable synchronization of the blockchain. In the experiment, orphan blocks of a height of one frequently occurred in both environments. In our experiment, we classified the height of subchains of two or higher as orphan blocks. In Figure 13, orphan blocks correspond to Blocks 13, 14, 15, 16, and 18.

Figure 14 demonstrates the blockchain status based on the results of the security assessment in the Bitcoin environment.

Figure 15 reveals the blockchain status according to the results of the security assessment in the ECCPoW environment.

All blockchains were measured at 0% because of the assessment. The security of ECCPoW was the same on a Bitcoin and orphan block basis. Blockchain measurements inhibited the generation of orphan blocks.

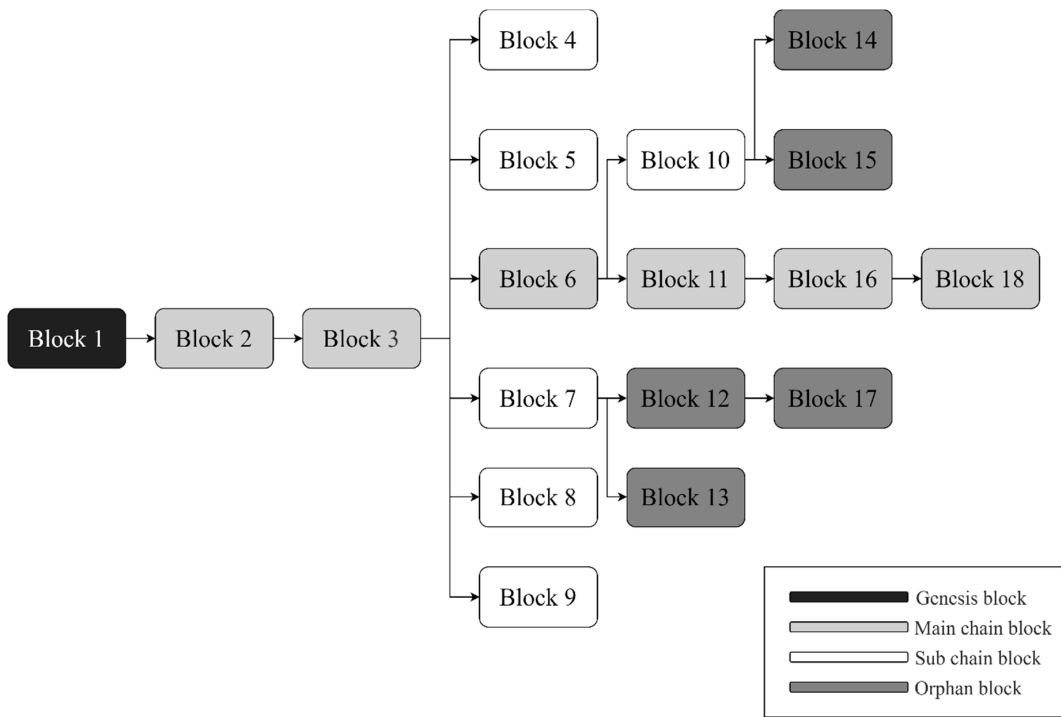


Figure 13. Diagram description of security evaluation.

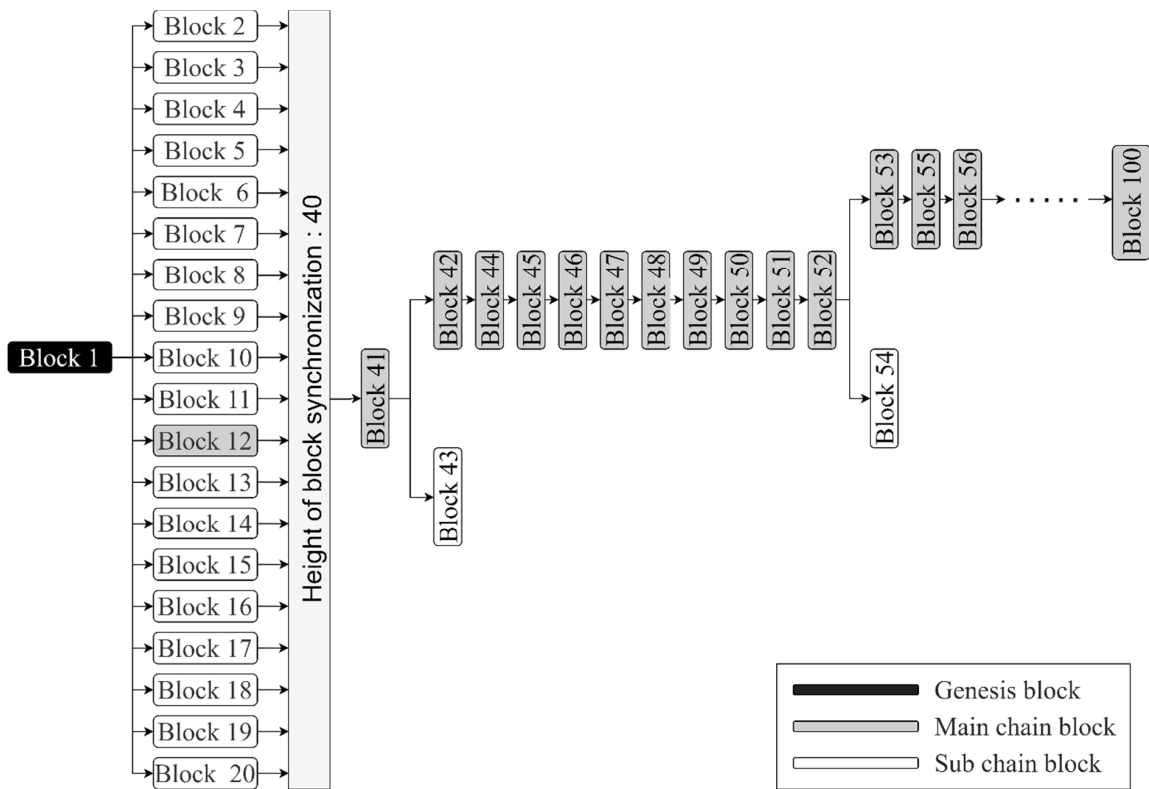


Figure 14. Evaluation diagram of Bitcoin security.

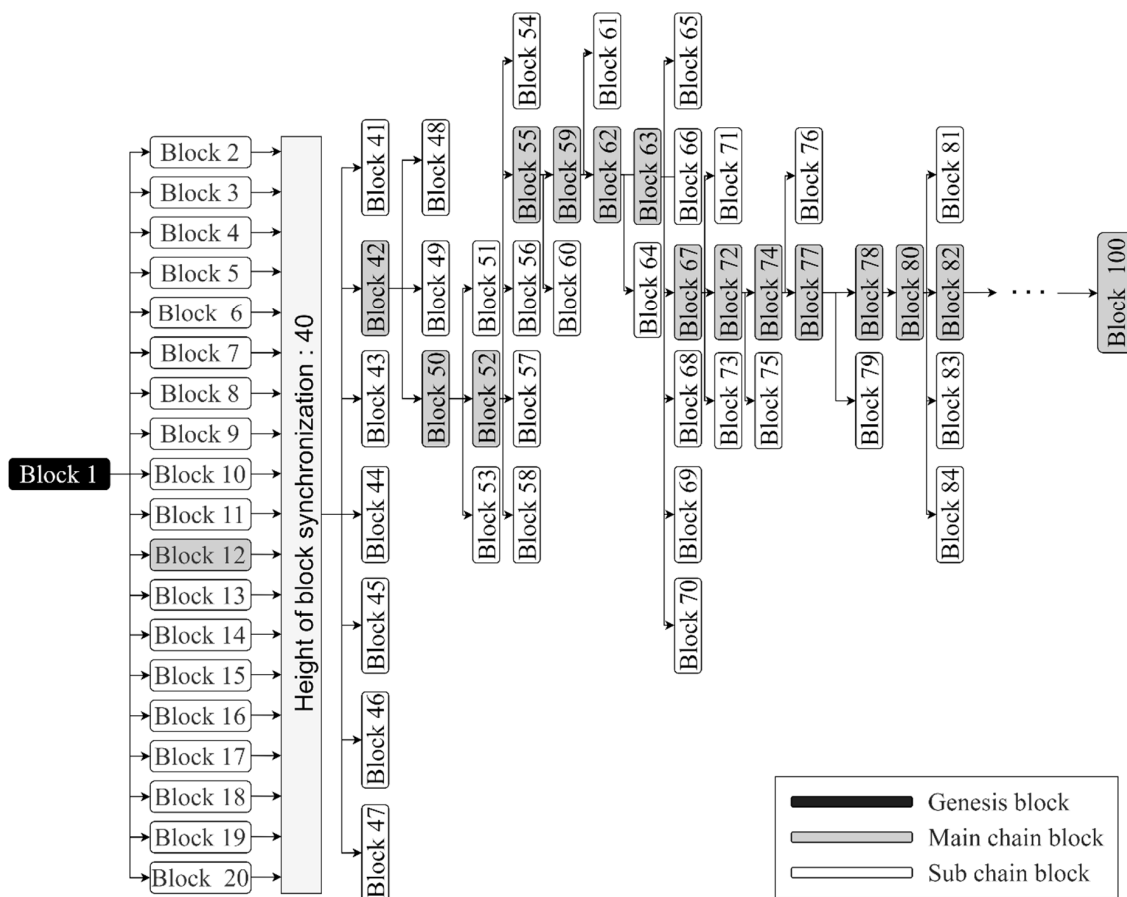


Figure 15. Evaluation diagram of ECCPoW security.

5.3. Scalability Evaluation

Scalability is the extent to which a system can flexibly respond to an increase in the number of users. Blockchain scalability is related to the transaction processing speed of blockchain. Transactions per second (TPS) is the transaction processing speed of the blockchain. Thus, we evaluated the scalability of the blockchain using TPS. The following equation defines how to obtain TPS using transactions in blocks.

$$TPS = \frac{\text{Total number of transactions in block}}{\text{Generation time of total block}}$$

We configured the blockchain test environment with three seed nodes and 20 mining nodes. Transaction generation occurs continuously from height 91 to 100—the blockchain mines 100 blocks. We checked the number of transactions in the height blocks of the blockchain from 91 to 100. We calculated the TPS of the blockchain. Table 11 lists the generation time and number of transactions of blocks of height 91 to 100 in Bitcoin.

Table 12 displays the generation time and the number of transactions of blocks of height 91 to 100, in ECCPoW. The evaluation results revealed that Bitcoin’s TPS is 0.95, and ECCPoW’s average TPS is 0.94. Moreover, ECCPoW’s scalability was measured at 98.95%, which was down 1.02% from Bitcoin’s scalability. In addition, ECCPoW added a process to Bitcoin to resist ASICs. However, the scalability values of ECCPoW and Bitcoin were assessed at a similar level.

Table 11. The evaluation result of Bitcoin scalability.

Number	Height	Block Generation Time (Seconds)	Number of Transactions
1	91	719	672
2	92	19	22
3	93	94	88
4	94	144	141
5	95	275	263
6	96	40	40
7	97	313	296
8	98	6	11
9	99	574	534
10	100	146	137
Total		2330	2204
TPS		0.945922747 TPS	

Table 12. Evaluation results of ECCPoW scalability.

Number	Height	Block Generation Time (Seconds)	Number of Transactions
1	91	151	140
2	92	192	182
3	93	369	351
4	94	361	331
5	95	156	151
6	96	214	205
7	97	124	120
8	98	267	250
9	99	585	548
10	100	514	480
Total		2933	2758
TPS		0.940334129 TPS	

5.4. Comparison to Related Work

This section compares the features with the existing studies on ASIC resistance. Table 13 compares the proposed method and existing relevant research on ASIC resistance. The ASIC resistance study for the PoW blockchain has three approaches. The first method causes bottlenecks using memory access in a hash function—for example, Ethereum (Ethash) and Bytecoin. The second method prevents the generation of ASICs using a hash function overlay, for example, the X-11 series. Finally, the third method uses periodic hash function replacement for ASICs, for example, in Monero (2019.9, RandomX algorithm) and Ethereum (2020.7 applying scheduled, ProgPoW).

Table 13. Comparison results of the related work.

Comparison Features	Proposed Method	Memory Approach	Hash Function Overlay	Periodic Hash Function Replacement
Applied cryptocurrency	-	Ethereum, Bytecoin	X-11 series	Monero (2019.9, RandomX), Ethereum (2020.7 applying scheduled, ProgPoW)
Characteristic	Change hash function every block	Force memory access	Overlapping multiple hash functions	Hard fork manually or automatically with a hash function
Algorithm	ECCPoW	Ethash, ProgPoW, CryptoNight	Blake, Bmw, Groetl, etc.	RandomX, ProgPoW
ASIC appearance	-	Yes	Yes	Unknown
ASIC resistance induction method	Use of ASIC resistance by connecting the LDPC decoder and hash function	Induce cache miss when creating blocks (using a DAG, etc.)	Overlapping the difficulty of known hash functions	Hard fork every six months (formerly Monero). Convert the hash function for a period using the key-block concept (currently Monero)

The resistance induction of each ASIC in the existing studies is as follows. The memory approach induces a cache miss (using DAG, among others) when creating blocks. This method degrades ASIC performance as memory access increases. The hash function overlay method uses the overlap of the difficulty of known hash functions. This method relies on the security of the applied hash function. The periodic hash function replacement method periodically changes the hash function manually or automatically.

6. Conclusions

In this paper, we explained ECCPoW and applied the proposed method to Bitcoin. This paper addressed the problem of centralization of mining due to the emergence of ASICs. We proposed a PoW concept based on error-correction codes to solve this problem. The core of the ECCPoW is the connection of the hash function with the LDPC decoder. Blockchain applied with the ECCPoW determines the completion of the POW using the output value of the decoder. In addition, ASIC suppression is possible because ASICs use the LDPC decoder.

This paper contributes to the phenomenon of symmetry in the PoW blockchain. The total block size of the PoW blockchain symmetrically influences the hash rate. The PoW blockchain increases stability as the hash rate increases in size. However, a high hash rate causes the waste of computing power in block generation. Thus, we mitigate the causes of a high hash rate using ASIC resistance.

The ECCPoW offers a method of solving different puzzles in each block to avoid ASICs. This maximizes the benefits of how existing studies use a limited number of hash functions to solve each block of different hash functions. The proposed method offers more effective connections and the use of multiple hash functions. We presented the difficulty control, parity-check matrix generation method, hash vector generation, and code determination methods for implementing ECCPoW. Furthermore, ECCPoW was applied to Bitcoin to verify the proposed method. We assessed the mining centralization, security, and scalability of ECCPoW and Bitcoin. We found that ECCPoW maintained security and scalability, showing 32% higher mining centralization than Bitcoin. Using the proposed method, ECCPoW does not require high hash rates, and miners can compete more fairly.

This study contributes to the previous literature on ASIC resistance in the PoW blockchain. The ASIC resistance study of the PoW blockchain was conducted in response to hardware development by ASIC manufacturers. One of the studies on ASIC resistance induced forced access to memory in the hash function, thereby lowering the performance of ASICs. Other studies have suggested periodically altering the hash function to render ASICs useless. The ECCPoW provides a method of releasing different hash functions for each block, rather than changing the periodic hash function.

The ECCPoW revealed the limit of block generation time in the experiment. For the stable operation of the blockchain, the block generation time of the ECCPoW must be stable and controllable. The results of the block generation time test were unstable in the experiment using one node. Thus, ECCPoW requires further research regarding the difficulty of block generation for stable operation. We also need to increase the number of nodes in the ECCPoW to carry out mining.

Author Contributions: H.J. and H.-N.L. contributed equally to this paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean government (MSIP) (NRF-2018R1A2A1A19018665).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nakamoto, S. *Bitcoin: A Peer-to-Peer Electronic Cash System*; 2008; Available online: <https://bitcoin.org/bitcoin.pdf> (accessed on 9 June 2020).
2. Nofer, M.; Gomber, P.; Hinz, O.; Schiereck, D. Blockchain. *Bus. Inf. Syst. Eng.* **2017**, *59*, 183–187. [[CrossRef](#)]
3. Saberi, S.; Kouhizadeh, M.; Sarkis, J.; Shen, L. Blockchain Technology and Its Relationships to Sustainable Supply Chain Management. *Int. J. Prod. Res.* **2019**, *57*, 2117–2135. [[CrossRef](#)]
4. Qin, R.; Yuan, Y.; Wang, F.-Y. Research on the Selection Strategies of Blockchain Mining Pools. *IEEE Trans. Comput. Soc. Syst.* **2018**, *5*, 748–757. [[CrossRef](#)]
5. Liu, X.; Wang, W.; Niyato, D.; Zhao, N.; Wang, P. Evolutionary Game for Mining Pool Selection in Blockchain Networks. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 760–763. [[CrossRef](#)]
6. Karame, G.O.; Androulaki, E.; Capkun, S. Double-Spending Fast Payments in Bitcoin. In Proceedings of the 2012 ACM Conference on Computer and Communications Security, Raleigh, NC, USA, 16–18 October 2012; pp. 906–917. [[CrossRef](#)]
7. Jang, J.; Lee, H.-N. Profitable Double-Spending Attacks. *arXiv* **2019**. Available online: <https://arxiv.org/abs/1903.01711> (accessed on 9 June 2020).
8. Antminer S19 Pro. Available online: <https://support.bitmain.com/hc/en-us/articles/900000261726-S19-Pro-Specifications> (accessed on 9 June 2020).
9. Park, S.; Kim, H.; Lee, H.N. Introduction to Error-Correction Codes Proof-of-Work. *Mag. IEIE* **2019**, *5*, 26–32.
10. Buterin, V. A Next-Generation Smart Contract and Decentralized Application Platform. *White Pap.* Available online: <https://ethereum.org/whitepaper/> (accessed on 9 June 2020).
11. Wood, G. Ethereum: A Secure Decentralised Generalised Transaction Byzantium Version. *Yellow Pap.* **2014**. Available online: <https://ethereum.github.io/yellowpaper/paper.pdf> (accessed on 9 June 2020).
12. Zhou, Q.; Huang, H.; Zheng, Z.; Bian, J. Solutions to Scalability of Blockchain: A Survey. *IEEE Access* **2020**, *8*, 16440–16455. [[CrossRef](#)]
13. Greg, C.; Andrea, L.; Michael, C.; IfDefElse. ProgPoW, a Programmatic Proof-of-Work. Ethereum—EIPs, No. 1057. 2018. Available online: <https://eips.ethereum.org/EIPS/eip-1057> (accessed on 9 June 2020).
14. Duffield, E.; Diaz, D. Dash: A PrivacyCentric Crypto-Currency. *White Pap.* **2014**. Available online: <https://docs.dash.org/en/stable/introduction/about.html#whitepaper> (accessed on 9 June 2020).
15. Saberhagen, N.V. Cryptonote v2.0. *White Pap.* **2013**. Available online: <https://cryptonote.org/whitepaper.pdf> (accessed on 9 June 2020).
16. RandomX. Available online: <https://github.com/tevador/RandomX> (accessed on 9 June 2020).
17. Shao, S.; Hailes, P.; Wang, T.-Y.; Wu, J.-Y.; Maunder, R.G.; Al-Hashimi, B.M.; Hanzo, L. Survey of Turbo, LDPC, and Polar Decoder ASIC Implementations. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2309–2333. [[CrossRef](#)]
18. Gallager, R. Low-Density Parity-Check Codes. *IRE Trans. Inf. Theory* **1962**, *8*, 21–28. [[CrossRef](#)]
19. Bitcoin ECC LDPC. Available online: https://github.com/cryptoecc/bitcoin_ECC/blob/ecc-0.1/src/ldpc/LDPC.cpp (accessed on 9 June 2020).

20. Ben-Haim, Y.; Litsyn, S. Upper Bounds on the Rate of LDPC Codes as a Function of Minimum Distance. *IEEE Trans. Inf. Theory* **2006**, *52*, 2092–2100. [[CrossRef](#)]
21. Bitcoin Core 0.17 Version. Available online: <https://github.com/bitcoin/bitcoin/tree/0.17> (accessed on 9 June 2020).
22. Bitcoin ECC 0.1.2 Version. Available online: https://github.com/cryptoecc/bitcoin_ECC/tree/ecc-0.1.2 (accessed on 9 June 2020).



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Smart Contract-Based Checkpoint for Initial PoW Network Security

Seungmin Kim, Gyeongdeok Maeng and Heung-No Lee*

Dept. of Electrical Engineering and Computer Sciences

Gwangju Institute of Science and Technology (GIST)

Gwangju, Korea

heungno@gist.ac.kr

Abstract—Proof of Work (PoW) is the most widely adopted consensus mechanism on public blockchains. The PoW blockchain network achieves consensus by solving computational problems. If a network participant owns more than 50% of the total computational power in the network, they can forge the blockchain. Hence, initial PoW blockchain networks with low computing power are vulnerable to block forgery attacks. We propose a smart contract-based checkpoint method to improve security vulnerabilities in the initial blockchain networks. In our method, participants periodically record block headers in an Ethereum smart contract. The recorded checkpoint block header validates the blockchain. Participants reject blocks with blocks that differ from the recorded checkpoints. Our method ensures the integrity of blocks until the height of the most recently created checkpoint, reducing the risk of double-spending. We optimize checkpoint costs by overlapping multiple checkpoint processes in a single transaction. The interval of our checkpoint method grows with the growth of the network, making the network less dependent on checkpoints. We analyze the performance of checkpoints in mitigating attacks and demonstrate that they significantly decrease the success probability of attacks in the network.

Index Terms—Checkpoint, PoW blockchain, smart contract, double spending attack

I. INTRODUCTION

Blockchain is a decentralized ledger that maintains the integrity of data through consensus. Blockchain uses a consensus mechanism to select decision-makers in each block. [1]. The decision-makers in the blockchain record the validated transactions in block and broadcast block to the network. The consensus mechanism is critical to blockchain decision-making and therefore has a major impact on the security and scalability of a blockchain [2]. Innumerable consensus mechanisms have been studied to increase the security and scalability of blockchains [3]–[6]. Proof of Work (PoW), introduced in Bitcoin, is the most popular consensus mechanism [7]. PoW is a consensus mechanism that involves solving mathematical problems to select decision-makers. The participant who solves the problem first gets the authority to create a block. The computational works involved in decision-making make it difficult for attackers to engage in malicious behavior. Proof of Work is still the most typical consensus mechanism, although Ethereum transitioned to Proof of Stake (PoS). [8].

In a PoW network, participants with more than 50% of the total computing power have majority decision-making power.

This truth means they can monopolize the network's decision-making process and manipulate the data on the blockchain [9]. It is problematic that one participant possesses more than 50% of the network's computing power in a mature PoW network. One participant possessing more than 50% of the network's computing power in a blockchain network is complicated and becomes increasingly difficult as the number of participants increases. However, in early blockchain networks, where the network size is small, possessing more than 50% is relatively easy.

Checkpoints are state values recorded at regular intervals in a blockchain network. Blockchain networks regularly store state values as checkpoints. Network participants recognize blockchains that contain checkpoints as legitimate chains. This new regulation increases the network's security by making it harder for attackers to forge blockchains. Checkpoints also serve as reference points for new network participants. The new participant connects to the blockchain network and attempts to synchronize blocks. There are many forked chains in the network, which can cause new participants to synchronize the wrong chains. New participants can use checkpoints to validate forked chains and prevent incorrect synchronization.

Related Work. There have been various implementations that have applied checkpoints and numerous research studies proposing checkpoint mechanisms. Bitcoin Core, a client software for Bitcoin, utilizes hard-coded checkpoints internally to protect the initial network from potential attack vulnerabilities [10]. Hard-coded checkpoints are inflexible and centralized to the developers. Polygon [11] executes the Proof of Stake (PoS) consensus process in Ethereum smart contracts and stores the outcomes as checkpoints. Polygon does not have its inherent proof-of-work mechanism and relies on Ethereum for network security. In [12], Wang and Kim propose an additionally distributed ledger for external validators and a method for recording checkpoints in the ledger. Their recording checkpoints to a distributed ledger incur a high network cost. [13] proposes to create a minority committee for each checkpoint, which decides and records the checkpoint. Their work assumes a synchronous network, unlike asynchronous blockchain networks.

Contributions. We propose a smart contract-based checkpoint method to improve the security of initial proof-of-work networks. The core idea of our proposed method is

to record checkpoints in the smart contracts of the Ethereum network. The Ethereum blockchain network's enormous size guarantees our checkpoints' security against tampering efforts. We design the checkpoint generation process to consider the asynchronous nature of blockchain networks. The checkpoint creation process consists of three timelines for each checkpoint generation interval. We overlap multiple checkpoint processes in a single transaction to reduce the cost of checkpoint generation. The checkpoint interval gradually increases with the total computing power of the network. As the interval between checkpoints recorded on Ethereum increases, the network becomes independent of Ethereum. We assume a selfish mining attack scenario and analyze the attack probabilities for checkpoints and typical networks. According to our analysis, our proposed method effectively reduces the attack probability compared to regular PoW networks.

Organization. The rest of this paper is structured as follows. In section II, we describe an overview of the research and background concepts related to checkpoints. Section III explains the checkpoint protocol and smart contracts. In section IV, we consider a scenario involving selfish mining attacks and compare the performance of checkpoints with a typical network. In section V, we conclude and discuss future work.

II. BACKGROUND AND OVERVIEW

A. PoW blockchain network attack

Attacks on proof-of-work (PoW) blockchain networks largely stem from the probabilistic nature of the PoW mechanism. PoW network follows the Longest Chain rule, which adopts the longest chain, i.e., the chain with the most accumulated work, as the legitimate chain of the network [14]. Whenever the PoW network finds a longer chain, it replaces the existing chain with the new one.

Double-spending attacks exploit the Longest Chain rule [15]. In a double-spend attack, the attacker pays through a legitimate transaction and receives a reward for the transaction. The attacker uses their computational power to create the longest blockchain that contains the forged transaction. The network recognizes the longest block, the forged blockchain, as legitimate and invalidates the legitimate transaction. The attacker obtains the forged transaction reward and reuses the currency used in the transaction. Selfish mining attack is another form of attack that exploits the Longest Chain rule [16]. In a selfish mining attack, the attacker intentionally withholds generated blocks instead of immediately broadcasting them. The attacker continues to increase the height of their private blockchain without revealing the blocks. The attacker announces the blocks after the attacker's blockchain surpasses the network's height. The transaction and block rewards on the network's chain become insignificant since the attacker's chain becomes the longest.

The attacker must catch up to the network's longest chain to execute such an attack. The attack has a probability of success of 1 if the attacker's computational power is over half of the network's total computing power since the attacker can catch up to the blockchain. However, even though the

attacker's computational power does not exceed 50%, there is a possibility that they can succeed in executing the attack probabilistically [17].

B. Confirmation distance

Confirmation distance is a mechanism introduced in blockchain networks to mitigate double-spending attacks [18]. Confirmation distance refers to the number of blocks a user has to wait for a transaction to be sent and confirmed. A participant sends a transaction to the blockchain but does not immediately supply the transaction reward and waits for the confirmation distance. The participant sends the transaction reward after all the confirmation distance blocks have been generated. A double-spending attacker must forge blocks after the confirmation distance to invalidate transactions. The confirmation distance is effective against double-spending attacks employing less than 50% of the total computational power. The probability of the attacker catching up with the blocks decreases as block height increases, assuming the attacker has less than 50% computing power. To receive a transaction reward, an attacker must attempt a double-spend attack after the confirmation distance. Confirmation distance increases the difficulty of an attack by requiring the attacker to catch up to the block after the confirmation distance.

The confirmation distance in a blockchain network can vary depending on the network conditions. The confirmation distance increases with a higher rate of total computational power for an attacker and decreases with a lower rate. A longer confirmation distance means a longer transaction processing time, which can cause inconvenience for participants who require fast transaction confirmations. Furthermore, attacks with more than 50% computational power always generate longer blocks, ignoring confirmation distance. As a result, additional security protocols are necessary when the attacker can launch an attack with substantial computational power.

C. Smart contract

A smart contract is a self-executing contract where the terms and conditions are directly written into code, enabling automatic execution without the need for intermediaries [19]. These contracts operate based on predefined conditions and rules built on a blockchain platform. The smart contract code automatically executes the contract conditions, ensuring transparency and accuracy throughout the contractual process. Smart contracts are recorded and verified on the blockchain, providing high security and resistance to tampering. They reduce the need for intermediaries, streamline processes, and automate transactions, potentially saving time and cost. The self-executing nature of smart contracts is widely utilized and researched in many fields that rely on data-driven transactions [20]. Increasingly, developers and researchers are actively involved in the development and exploration of smart contracts [21]–[23].

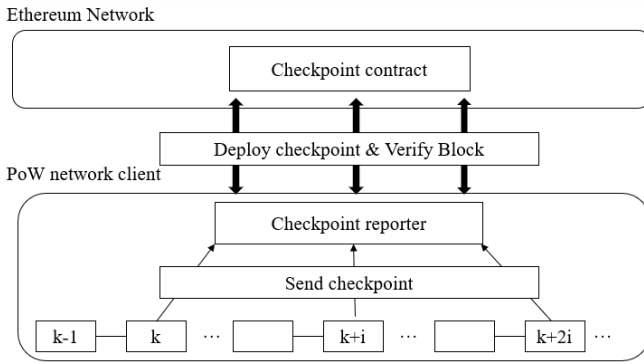


Fig. 1. The architecture of our method

III. PROPOSED METHOD

In this section, we first present an architecture of our proposed checkpoint method, then introduce the details of the method.

A. Architecture

Fig. 1 depicts the architecture of our proposed method. The proposed method consists of two layers: the Ethereum network layer, which includes the checkpoint contract, and the PoW network layer, which the checkpoint aims to protect. Our method combines the Ethereum network layer with the PoW network layer to ensure the security of the PoW network. The checkpoint contract runs on the Ethereum network and serves as the administrator overseeing the management and execution of the checkpoint. The checkpoint contract defines the rules and conditions necessary to verify and record checkpoints on the Ethereum blockchain securely. The PoW network works similarly to a typical blockchain network but includes functionalities for generating, transmitting, and requesting checkpoints.

The PoW network validator generates a checkpoint when it reaches the checkpoint generation interval, which serves as a reference for the network's state. Checkpoint is nearly identical to the block header of the same block height, but they include additional data that marks them as a checkpoint. Checkpoints are generated every interval and sent to the Ethereum checkpoint contract. Participants in the network request the contract for a checkpoint when a block modification occurs such that the new block can be validated. Since checkpoint requests do not affect the state of the contract, they can be provided directly without submitting a transaction. The network client communicates with the checkpoint contract through the checkpoint reporter. The checkpoint reporter consists of an Ethereum light client that can communicate with Ethereum and is compatible with the same private key used by the PoW client.

Ethereum smart contract-based architecture provides the following advantages for the checkpoint. First, it mitigates the risk of checkpoint tampering. Ethereum is a network with many participants, and the likelihood of tampering is

extremely low. Second, checkpoints operate transparently and according to predetermined rules, ensuring that checkpoint development remains decentralized and not centralized in the hands of checkpoint developers.

B. Checkpoint generation process

One of the things to consider when creating checkpoints is that Proof of Work networks operate as asynchronous distributed networks, which means that block propagation can be delayed or uncertain. We consider the asynchronous characteristics of the network during the checkpoint generation process. Despite a checkpoint created by one participant, other participants may need to be made aware of it and may still attempt to generate a checkpoint. We separate the checkpoint generation process into three timelines to reflect the network's asynchronous characteristics. Fig. 2 shows the checkpoint process timelines.

In Fig. 2, The first timeline occurs when the block height is at checkpoint time, defined as k . The decision maker creates block k but waits without creating a checkpoint. It is risky to trust a block that has just been created on an asynchronous network. Other blocks can easily replace blocks. Participants must wait until the network has transmitted block k to all participants.

The second timeline occurs at block height $k + i$, where i represents the checkpoint interval. We assume that i blocks have been generated from height k , so the propagation of block k is almost complete. The participant generates a checkpoint for block k and transmits it to the contract when the height $k + i$ has been achieved. The contract does not immediately complete the checkpoint operation for height k but instead stores the received checkpoints in a checkpoint pending pool and waits. The participant also performs the first timeline process at checkpoint $k + i$.

The third stage similarly occurs, following the completion of the second stage at height $k + i$. The second and third timelines i could differ according to the checkpoint interval adjustments described in the checkpoint interval section. However, we represent it as $k + 2i$ for ease of expression. The contract stores the checkpoints for height k in the checkpoint pending pool and waits. The first timeline for height $k + 2i$ and the second timeline for height $k + i$ execute when the network reaches height $k + 2i$. During this process, participants transmit the checkpoints to contract for height $k + i$. The contract uses the checkpoint for height $k + i$ to recognize the block height, as it cannot identify the network's current block height. The contract finishes the vote for checkpoint k and decides the final checkpoint, k , after receiving the first checkpoint for height $k + i$. The final checkpoint is decided based on receiving the most votes. The checkpoint's recording guarantees the integrity of the blocks up to height k .

The actual network propagation time will probably be shorter than the checkpoint interval, but we await the propagation of the network until the next checkpoint interval. This wait causes multiple checkpoint processes to work simultaneously in an interval. Participants can merge transactions from

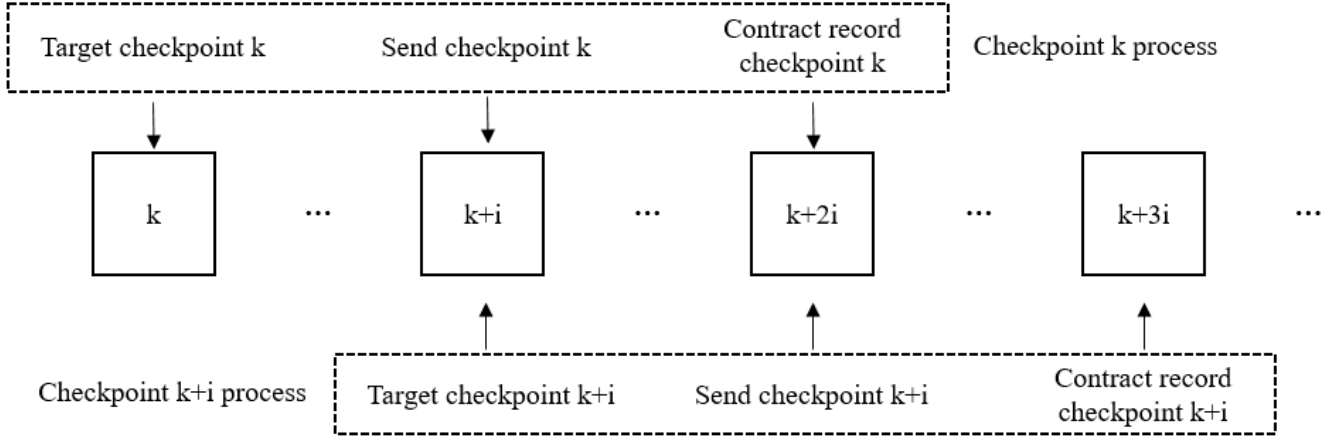


Fig. 2. Checkpoint process timeline

concurrently running checkpoints and send them in a single transaction. Checkpoint transaction merging reduces the cost of concurrently counting up to three processes by up to a third.

C. Checkpoint Smart contract

We establish a validator set that can generate checkpoints and manage the list in the contract. This list of validators forms a decentralized autonomous organization (DAO) that operates autonomously according to predefined rules within the contract. All validators have equal authority and participate in decision-making through voting. The contract records the checkpoint that receives the most votes from validators as the final checkpoint. Validator DAO can make decisions and execute processes without relying on centralized authority. The equal authority of the DAO ensures that decisions are made fairly and equitably.

The signatures of the validators determine the authenticity of a checkpoint transaction. We ensure the validators use the same private key for the PoW network and Ethereum. The validator signs the checkpoint with their private key and sends it to the contract. The contract verifies that the checkpoint's signature is included in the validators. Checkpoints with valid signatures are stored in the checkpoint pending pool. When a contract determines a checkpoint within a pending pool, the contract saves the checkpoint. The contract only stores the most recent checkpoint. The block generation process in PoW includes data from the previous block. The most recent checkpoint includes the previous one, which means the contract does not need to store the previous one. The contract calculates the attacker's computational power as the block difficulty for the spacing adjustment described in the following subsection. Algorithm 1 is a pseudo-code showing the process executed by the checkpoint contract when a checkpoint is transmitted.

D. Checkpoint Interval

The smart contract calculates the checkpoint interval at the time of checkpoint creation. The checkpoint includes the block

Algorithm 1 Checkpoint contract process

```

1: function SUBMITCHECKPOINT(header , sig)
2:   proposer, blocknumber, roothash ← header
3:   if proposer not in validators then
4:     return ValError
5:   if EcVerify(header, sig, proposer) is false then
6:     return SignError
7:   if CheckpointNumber < blocknumber then
8:     checkpoint = PendingHeader.decideCheck()
9:     Interval ← CalNextInterval(checkpoint)
10:  PendingHeader.append(header)
11:  return (checkpoint, Interval)

```

difficulty at the checkpoint's point in time, allowing us to assess the overall scale of the network through the block difficulty. Given the difficulty of block height k as $diff_k$ and block time as T_k , the total computational power of the network c_k is calculated as follows:

$$c_k = \frac{diff_k * 2^{32}}{t_k - t_{k-1}} \quad (1)$$

We find the attacker's computational power c_a to get the attacker's total computational power ratio. The c_a can be roughly gauged from the availability of computing resource rental platforms. For example, the Ethash computational power available for rent on nicehash.com in May 2023 is approximately 3 TH/s [24]. The attacker's total computational power ratio t can be expressed as follows:

$$t = \frac{c_a}{c_k} \quad (2)$$

Our checkpoints are confirmed through the three timelines and finalized two intervals later. A checkpoint interval of i will provide finality for a block up to $2i$ earlier. The confirmation distance z must be greater than $2i$ to be secure against double-spending attacks. The interval of checkpoints

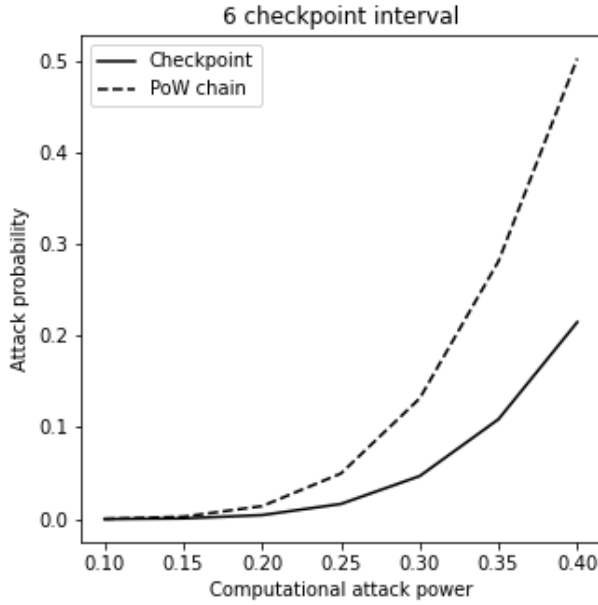


Fig. 3. probability of selfish mining attacks based on the attack power.

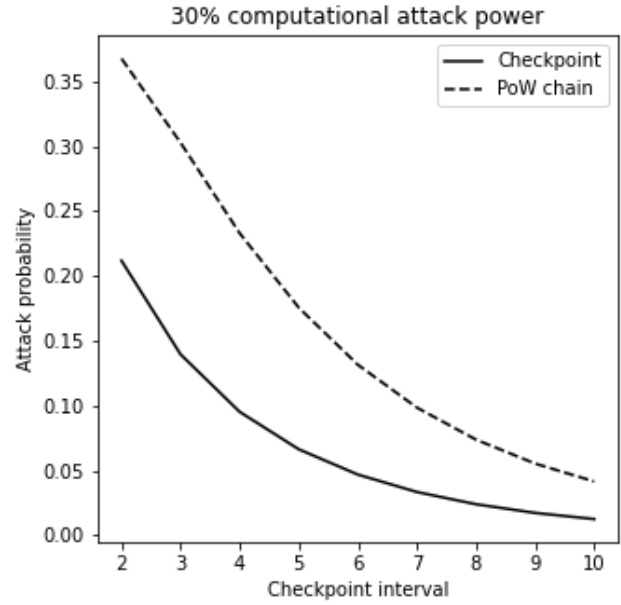


Fig. 4. probability of selfish mining attacks based on the checkpoint interval

significantly affects network performance. It provides significant completeness and security against short-term attacks but increases transaction costs.

Considering the cost of the attack, the network is sufficiently secure if the probability of a double-spend attack is less than 0.001. It means that the network is secure enough without checkpoints. We designed the interval of checkpoints to increase as the network expands. For every halving of the attack probability, the checkpoint interval doubles. This process repeats, gradually reducing the dependency on checkpoints. The attacker's probability p can be calculated from t obtained in Eq.(2) [15]. The network reaches a sufficiently large size to be secure against attacks. Given that the confirmation distance is z , and the attacker's attack probability is p , the interval i of the checkpoint can be obtained as follow:

$$i = \begin{cases} \frac{z}{2} & \text{if } p > 0.001 \\ 2^{\lceil \log_2(\frac{0.001}{p}) \rceil} \cdot z & \text{if } else \end{cases} \quad (3)$$

IV. PERFORMANCE ANALYSIS

This section analyzes our proposed method's Selfish Mining Attack security performance. Our proposed method is resistant to double-spending attacks but only partially immune to selfish mining attacks. It is because Selfish Mining Attacks can occur even within confirmation distance. A typical PoW network has no checkpoints, so there is an infinite amount of time for a selfish mining attack to be attempted. In contrast to typical PoW networks, our method limits selfish mining attackers to only attacking until a checkpoint is generated.

The selfish mining attack scenario in our method is as follows. The attacker confirms that a checkpoint has been

created and then prepares for a selfish mining attack before the next checkpoint. The attacker creates a block but does not broadcast it. The attacker creates and broadcasts the longest block before the checkpoint is created. Blocks and transactions created by other participants become invalid and monopolize the rewards of those blocks. The contract records the blocks generated by the attacker as checkpoints. This attack can harm network performance, but transactions are secure because the attacker's reward is limited to the mining revenue.

The probability of the selfish mining attack scenario is calculated similarly to the probability of a double-spend attack. We express the probability of a selfish attack scenario using a Poisson distribution. The attacker must create the longest chain and forge blocks before recording the checkpoint. We compute the probability distribution of blocks an attacker can generate during a checkpoint interval. We then calculate the total probability of the blocks generated by the attacker being more significant than the checkpoint interval. When the checkpoint interval is denoted as i , and the parameter λ is equal to $i * \frac{q}{1-q}$, the probability of the attacker's selfish mining attack can be computed as follows:

$$\text{attack}_i = \sum_{k=0}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \cdot \begin{cases} 0 & \text{if } k \leq i \\ 1 & \text{if } k > i \end{cases} \quad (4)$$

We compare the performance of our method in the selfish mining attack scenario with a typical PoW network using Eq.(4). In our analysis, and we consider various attackers' computational power and the checkpoint interval. We conducted two experiments to evaluate the efficacy of checkpoints. In the first experiment, we set the checkpoint interval to 6 and analyzed the impact of varying the attacker's computational

power on the success probability of attacks. The results of this experiment are presented in Fig. 3. It shows that our method exhibits a decrease in attack probability for all levels of attack power, with the reduction rate gradually increasing. It showcases a maximum reduction of approximately 60% in the attack probability. In the second experiment, we maintained the attacker's computational power at a fixed rate of 30% and observed the changes in the attack success probability by adjusting the checkpoint interval. Fig. 4 depicts the outcomes of this experiment. Similarly, Fig. 4 shows that our method reduces the attack success probability across all checkpoint intervals.

As a result, the two experiments show that checkpoint serves as an effective solution to prevent selfish mining attacks. The consistent decrease in attack probability for all situations is convincing evidence of the performance of our method.

V. CONCLUSION

In this paper, we present a smart contract-based checkpoint mechanism for Proof of Work (PoW) networks, addressing the challenges associated with network security and vulnerability to attacks. We record checkpoints in Ethereum smart contracts to reduce the probability of checkpoints being forged, as Ethereum is exceedingly difficult to forge. Since blockchain networks are asynchronous, we proposed a checkpoint generation process composed of three timelines that execute each checkpoint interval. To reduce checkpoint costs, we merge concurrently running processes and submit them as a single transaction. The checkpoint interval increases gradually as the PoW network expands. When the checkpoint interval is large enough, the network moves away from its dependence on checkpoints and becomes independent of Ethereum. We performed an attack probability analysis on a selfish mining attack to show the effectiveness of our method. The proposed checkpoint significantly lowers the probabilities of attacks, especially in the initial stages of PoW networks, compared to typical PoW networks. In our future work, we will apply our checkpoint method to existing blockchain networks and research methods to reduce the cost associated with checkpoints.

ACKNOWLEDGMENT

This work was supported by the MSIT, Korea, under the ITRC (Information Technology Research Center) support Program (IITP-2023-2021-0-01835) supervised by the IITP (Institute for Information & Communications Technology Planning Evaluation.)

REFERENCES

- [1] M. Platt, J. Sedlmeir, D. Platt, J. Xu, P. Tascia, N. Vadgama, and J. I. Ibañez, "The energy footprint of blockchain consensus mechanisms beyond proof-of-work," in *2021 IEEE 21st International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 2021, pp. 1135–1144.
- [2] S. Velliangiri and P. Karthikeyan, "Blockchain technology: Challenges and security issues in consensus algorithm," in *2020 International Conference on Computer Communication and Informatics (ICCCI)*, 2020, pp. 1–8.

- [3] S. Chen, H. Mi, J. Ping, Z. Yan, Z. Shen, X. Liu, N. Zhang, Q. Xia, and C. Kang, "A blockchain consensus mechanism that uses proof of solution to optimize energy dispatch and trading," *Nature Energy*, vol. 7, no. 6, pp. 495–502, 2022.
- [4] Y. Wang, H. Peng, Z. Su, T. H. Luan, A. Benslimane, and Y. Wu, "A platform-free proof of federated learning consensus mechanism for sustainable blockchains," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 12, pp. 3305–3324, 2022.
- [5] "Proof-of-contribution consensus mechanism for blockchain and its application in intellectual property protection," *Information Processing Management*, vol. 58, no. 3, p. 102507, 2021.
- [6] P. Fernando, K. Dadallage, T. Gamage, C. Seneviratne, A. Madanayake, and M. Liyanage, "Proof of sense: A novel consensus mechanism for spectrum misuse detection," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 12, pp. 9206–9216, 2022.
- [7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized business review*, p. 21260, 2008.
- [8] E. Kapengut and B. Mizrach, "An event study of the ethereum transition to proof-of-stake," *Commodities*, vol. 2, no. 2, pp. 96–110, 2023. [Online]. Available: <https://www.mdpi.com/2813-2432/2/2/6>
- [9] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 3–16. [Online]. Available: <https://doi.org/10.1145/2976749.2978341>
- [10] "Bitcoin core 0.11.0," <https://bitcoin.org/en/release/v0.11.0>, 2015.
- [11] "Matic whitepaper," <https://github.com/maticnetwork/whitepaper/>, 2020.
- [12] K. Wang and H. S. Kim, "Reducing confirmation reversal probability of pow blockchains using checkpoints," in *2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2022, pp. 1–9.
- [13] D. Karakostas and A. Kiayias, "Securing proof-of-work ledgers via checkpointing," in *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2021, pp. 1–5.
- [14] S. Sayeed and H. Marco-Gisbert, "Assessing blockchain consensus and security mechanisms against the 512019." [Online]. Available: <https://www.mdpi.com/2076-3417/9/9/1788>
- [15] M. Rosenfeld, "Analysis of hashrate-based double spending," *arXiv preprint arXiv:1402.2009*, 2014.
- [16] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang, and Q. Kong, "A deep dive into blockchain selfish mining," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.
- [17] J. Jang and H.-N. Lee, "Profitable double-spending attacks," *Applied Sciences*, vol. 10, no. 23, 2020. [Online]. Available: <https://www.mdpi.com/2076-3417/10/23/8477>
- [18] Y. Kawase and S. Kasahara, "Transaction-confirmation time for bitcoin: A queueing analytical approach to blockchain mechanism," in *Queueing Theory and Network Applications*, W. Yue, Q.-L. Li, S. Jin, and Z. Ma, Eds. Cham: Springer International Publishing, 2017, pp. 75–88.
- [19] N. Szabo, "Formalizing and securing relationships on public networks," *First monday*, 1997.
- [20] W. Zou, D. Lo, P. S. Kochhar, X.-B. D. Le, X. Xia, Y. Feng, Z. Chen, and B. Xu, "Smart contract development: Challenges and opportunities," *IEEE Transactions on Software Engineering*, vol. 47, no. 10, pp. 2084–2106, 2021.
- [21] S. Ahmadiheykhsarmast and R. Sonmez, "A smart contract system for security of payment of construction contracts," *Automation in Construction*, vol. 120, p. 103401, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S092658052030981X>
- [22] A. Khatoun, "A blockchain-based smart contract system for healthcare management," *Electronics*, vol. 9, no. 1, 2020. [Online]. Available: <https://www.mdpi.com/2079-9292/9/1/94>
- [23] H. Hasan, E. AlHadhrami, A. AlDhaheri, K. Salah, and R. Jayaraman, "Smart contract-based approach for efficient shipment management," *Computers Industrial Engineering*, vol. 136, pp. 149–159, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0360835219304140>
- [24] "Nicehash marketplace," <https://www.nicehash.com/marketplace>, 2023.

지속 가능한 경제와 암호화폐

맹경덕^{o*}, 김승민*, 이흥노*

광주과학기술원 전기전자컴퓨터공학부*

maengdeok@gm.gist.ac.kr, seungminkim@gm.gist.ac.kr, heungno@gist.ac.kr

초록

케인스주의, 신자유주의 그리고 실비오 게젤의 이론을 절충하여 새로운 통화 시스템을 제안한다. 현재의 통화 시스템은 중앙집권적이며 과도한 인플레이션과 이윤율 저하와 같은 구조적 한계를 가지고 있다. 통화정책의 중앙집권화와 불확실성을 제거하고 블록체인 기술을 통한 시스템의 보안과 신뢰성을 확보한다. 그리고 DAO와 SmartContract를 활용해 새로운 지식의 생산을 장려함으로써 전통적인 통화 시스템의 구조적 문제를 해소하고 지속 가능한 성장을 목표로 한다. 이 화폐 시스템은 또한 교환 기능을 중심으로 일관되고 투명한 통화팽창을 제공함으로써 안정성과 생산성 향상의 균형을 이룰 수 있음을 제시한다.

지속 가능한 경제와 암호화폐

맹경덕^{○*}, 김승민^{*}, 이흥노^{*}

광주과학기술원 전기전자컴퓨터공학부^{*}

maengdeok@gm.gist.ac.kr, seungmin@gm.gist.ac.kr, heungno@gist.ac.kr

1. 서론

현대 통화 시스템은 케인스주의와 신자유주의 원칙의 혼합이다. 중앙은행들은 경제 성장을 견인하기 위한 케인스식 재정 자극과 함께 물가 안정을 유지하기 위한 감세, 규제 완화, 노동시장 유연화 등 하이에크식 통화 정책을 적절히 활용한다. 그러나 두 가지 접근 방식 모두 경제의 구조적 문제에 대한 일시적 완화만 제공했을 뿐 근본 원인을 해결하지 못하였다.

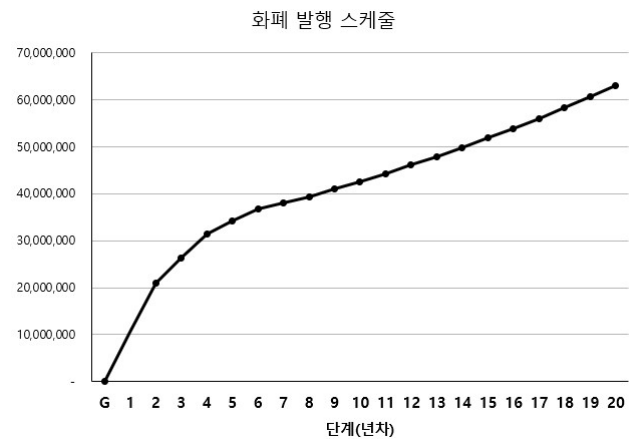
이러한 현대경제의 구조적 문제에 대한 대응으로 발생한 것이 Bitcoin이다. 2008년 경제 위기로 사람들은 중앙은행과 전통적인 금융기관에 대한 신뢰를 잃게 되었고, Satoshi Nakamoto는 어떠한 정부나 금융기관의 통제도 받지 않는 화폐인 Bitcoin을 탄생시켰다[1]. 이는 하이에크의 ‘화폐의 탈국가화’ 와도 맥락을 같이하고 있다. 하이에크는 화폐가 정부 정책에 의해 결정되기보다, 금과 같은 Sound money로 존재해야 한다고 보았다. 이러한 Sound money는 정부나 중앙은행에서 가치를 조작할 수 없고, 수요와 공급에 따라 시장가치를 가지기 때문에 안정적으로 유지될 수 있다고 주장하였다[2].

현재 Bitcoin은 소수의 독점적인 권한 없이 탈중앙적으로 채굴, 유지되고 시장의 수급에 따라 그 가치가 결정되고 있다. 그러나, Bitcoin의 발행량은 제한되어 있고 반감기를 거치면서 생산량이 감소하는 구조로 설계되었기 때문에 장기적으로 그 가치가 계속해서 상승하게 된다. 이러한 구조적 특징으로 인해 Bitcoin을 소비하는 행동은 기회비용이 크고 Bitcoin을 소유한 사람들로 하여금 소비보다 저축에 경제적 유인을 제공한다. 이는 화폐의 유통 속도를 감소시키게 되고 경기순환에 경색을 야기함으로써 경기 침체를 유발할 수 있다. 결론적으로 Bitcoin은 장기적인 측면에서 교환의 목적으로 사용되기 어렵고 자산의 저축에 사용될 가능성이 크다. 따라서 본 연구에서 케인스와 계절의 이론을 바탕으로 교환에 적합하며 건전한 소비와 생산을 장려할 수 있는 화폐 시스템을 제안한다.

2. 제안 모델

궁극적으로 본 연구의 목적은 케인스주의, 신자유주의 그리고 계절 경제학의 핵심 원리와 블록체인 기술을

바탕으로 지속 가능한 화폐 시스템을 제안하는 것이다. 블록체인 기술의 활용은 보안성과 신뢰성의 확립을 가능하게 한다. 그리고 하이에크가 주창한 화폐의 탈국가화를 실현시킴으로써 전통적인 통화 시스템의 중앙 집중적 성격으로 인한 구조적 한계를 해결할 수 있다.



[그림 1] 화폐 발행 스케줄

위의 [그림 1]은 본 연구에서 제안하는 지속 가능한 암호화폐의 발행 스케줄이다. 초기 8년간의 반감기 이후, 매년 4%의 고정된 통화팽창률로 예측 가능한 성장을 지향한다. 이로 인해 화폐의 액면 가치를 감소시키고 유통 속도의 증대를 불러오므로써 케인스와 계절의 관점을 종합한다. 이 통화 시스템은 저축의 기능이 강한 Bitcoin과 다르게 화폐의 교환 기능에 지향점을 두고 활발한 화폐 유통과 투명하게 이루어지는 점진적 통화팽창률로써 생산성 향상과 안정성 간의 균형을 이루는 것을 목표로 한다. 이러한 제안에 대한 구체적인 근거들을 다음 목록에서 제시한다.

3. 화폐의 감가상각

케인스는 경제의 구조적 문제에 대해 유효수요 이론을 바탕으로 그의 해결책을 전개하였다. 그는 수요 부족을 근본적인 문제로 파악하고 지출 증가와 감세를 통한 정부 개입이 경제를 활성화할 수 있다고 주장하였다[3]. 반면 케인스의 이론이 낳게 될 결과를 앞서 예측한

경제학자가 그 이전에 존재했다. 실비오 게젤은 이러한 접근법에 대해 화폐의 축적과 수요 부족은 원인이 아닌 하나의 결과이며, 정부의 화폐 공급정책으로 유효수요 부족 현상을 해소하는 것은 결국 더 많은 화폐의 발행, 과도한 축적 그리고 파괴적인 인플레이션을 야기할 수 있음을 경고하였다. 그는 화폐의 액면 가치가 불변하는 것이 모든 경제 문제의 근본적인 원인으로 보았고, 이러한 성질은 사람들로 하여금 화폐를 교환의 매개체로 사용하는 대신 화폐의 저축으로 유인한다고 주장하였다. 그리고 화폐를 저축의 대상으로 인식하는 경향은 수요 부족과 그에 따른 경기 침체로 이어진다고 지적하였다[4].

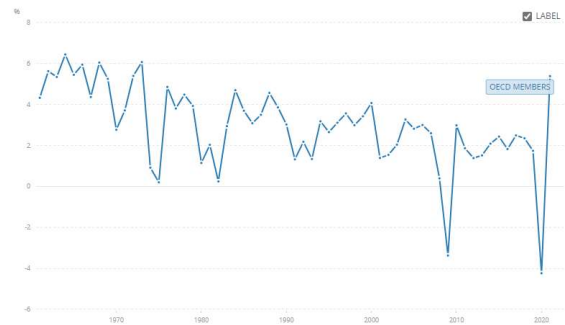
오늘날에 이르러서, 1970년대 스태그플레이션과 COVID-19 시기의 무제한 양적 완화로 인한 결과들은 게젤의 예측이 사실임을 증명하였다. 1970년대 동안 많은 국가들은 케인스의 이론을 바탕으로 경기 침체에 대응하기 위해 정부지출 증가와 통화팽창 정책을 시행하였다. 그러나 통화팽창과 수요 창출 정책은 중동발 오일 쇼크와 이윤율 저하 등의 요인들에 의해 효과가 감소되었고, 인플레이션이 급증하는 동시에 성장 정체가 공존하는 스태그플레이션 현상이 발생하게 되었다. 이어서 2019년 COVID-19 범 유행 기간 동안 시행된 무제한 양적 완화는 단기적으로 경기를 회복시키고 소비시장을 활성화시킬 수 있었지만, 결국 소비자 물가 급등, 자산 시장과 자원의 비생산적 투자 과열 그리고 정부의 부채 증가로 인한 재정 부실 등의 2차 문제를 초래함을 확인하였다.

이러한 사례들을 근거로 본 연구에서는 케인스의 접근법에서 중앙화적인 성질과 통화팽창률의 변동성을 제거하고 투명하고 집진적인 통화팽창정책으로 화폐의 액면 가치를 삭감한다. 이를 통해 화폐의 저축기능을 억제하고, 화폐가 교환의 목적으로 사용되는 것을 유도한다.

4. 예측 가능한 성장

투명한 고정 인플레이션 시스템 뒤에 있는 추론은 고용과 인플레이션의 관계에 뿌리를 두고 있다. 케인스 이론에 따르면 인플레이션은 실질 임금 감소로 인해 고용 증가와 실업 감소로 이어질 수 있다[2]. 그러나 노동자들이 실질 임금 감소에 대응하여 더 높은 임금을 요구함으로써 이러한 효과는 점차 감소하게 된다. 이 점은 로버트 루카스에 의해 증명되었는데, 노동자들은 더 높은 인플레이션을 기대하고 더 높은 임금을 요구하지만, 예상한 만큼 인플레이션율이 높지 않으면 실질 임금의 상승과 고용 감소를 초래할 수 있다[7]. 이러한 불확실성에 비추어 볼 때, 투명한 고정 인플레이션 정책이 변동 인플레이션 정책보다 더 지속 가능한 시스템이라는 것은 분명하다. 이에 본 연구에서는 연 4%의 고정된 통화팽창을 통해 예측 가능한 경제 성장을 설계하

였다.



[그림 2] OECD 국가 연평균 성장률
(<https://data.worldbank.org>)

이러한 수치는 위의 [그림 2]를 바탕으로 산출하였다. [그림 2]는 OECD 국가의 연평균 GDP 성장률 통계치를 나타낸 것이다. OECD의 회원국들은 평균적으로 연간 4%의 성장을 달성하였다. 이러한 통계 데이터에 근거하여 일관되고 투명한 통화팽창을 제공하고 화폐의 액면가치를 감소시킨다. 이로 인해 화폐를 교환의 매개체로 사용함으로써 소비와 생산을 장려하고 자금경색과 금융시장의 버블을 해소할 수 있다.

5. 지속 가능한 성장

지속 가능한 성장은 ‘이윤율 저하를 동반하지 않는 생산성의 증대’에 부합하는 통화 시스템을 필요로 한다. 전통적인 경제 시스템에서 새롭게 발행된 화폐는 자산 저축의 용도로 사용되어 돈의 경색을 발생시키거나 생산에 적절히 투자될 경우에도 문제를 일으키게 된다. 마르크스의 이윤율 저하 법칙에 의하면 사회가 발전함에 따라 실물자본의 양의 증가하면서 잉여가치를 낮추게 되고 결국 실물자본의 이윤율이 감소하게 된다[5]. 이러한 현상에 의해 실물자본의 이윤율은 금융자본 이자율보다 낮아지게 되고 결과적으로 생산에 투자된 돈 또한 금융 시장으로 유입된다. 이러한 일련의 과정들을 거치면서 발생하는 금융 버블은 결국 금융 시장의 불안정뿐만 아니라 실물 자본시장의 침체를 동반하기 때문에 자본의 붕괴를 일으킨다. 본 연구에서는 새로운 지식의 생산에 경제적 유인을 발생시켜 생산성과 더불어 이윤율의 지수함수적 증대에 부합하는 통화시스템을 제안함으로써 그 구조적 한계점을 극복하고자 한다.

지식기반경제에서의 부의 창출은 새로운 지식의 생산을 통해 이루어지는데, 새로운 지식과 생산기술은 재화와 서비스의 비용을 감소시키는 동시에 품질과 이윤율을 지수함수적으로 증대시킴으로써 삶의 질을 향상시킨다. 이러한 성장을 육성하기 위해서 블록체인 상에서 탈중앙적으로 구현되는 DAO와 SmartContract 기술을 응용할 수 있다[6]. DAO의 탈중앙적 의사결정 기능과

SmartContract의 자동화 시스템을 통해 네트워크를 지속적으로 발전시키는 기여자에게 합당한 보상을 제공한다. 기여자는 dApp 개발, 네트워크 하드포크, 신규 유입에 대한 영향력 등의 일을 수행함으로써 이를 제공받을 수 있다.

연 4%의 고정된 통화팽창의 한도 내에서 이러한 지수함수적인 생산을 추구하는 합리적인 부의 분배는 돈이 축적, 경색되는 것을 제한하고 소비와 생산을 장려할 수 있다. 저축되어 소비와 생산에 기여하지 않는 돈은 통화팽창에 의해 점진적으로 그 가치를 잃어가지만, 소비되고 건전하게 투자되는 돈들은 새로운 가치를 창출한다. 그리고 이렇게 창출된 유의미한 가치와 부는 견고하고 지속 가능한 경제 발전을 가능하게 한다.

6. 결론

본 연구에서는 현재 통화시스템이 가지고 있는 한계점들을 명확히 하고 케인스, 하이에크 그리고 케절의 경제학적 이론과 실제 데이터를 근거로 그 한계를 극복하고자 하였다. 이를 달성하기 위해 블록체인과 블록체인 상에서 구현할 수 있는 DAO, SmartContract 기술들을 기반으로 지속 가능한 화폐 체계를 제안하였다. 이 시스템은 블록체인 네트워크에 의해 보안과 탈중앙성 그리고 확장성을 보장받는다. 그리고 투명하고 예측 가능한 통화 공급으로 화폐의 액면가치를 감소시킴으로써 유통 속도를 향상시키고 화폐의 교환 기능을 강화하여 지속적인 생산성 증대를 가능하게 할 것이다. 추후 본 연구의 결과를 적용시켜 발행한 암호화폐가 지속 가능한 경제를 구축하기를 기대한다.

참고문헌

- [1] Nakamoto, S., "Bitcoin white paper." URL: <https://bitcoin.org/bitcoin.pdf>, 2008 Jan.
- [2] Friedrich Hayek, The Denationalization of Money, Institute of Economic Affairs, 1976
- [3] 존 메이너드 케인스, 고용, 이자, 화폐의 일반이론, 필맥, 2010
- [4] 실비오 케절, 자연스러운 경제질서, 클, 2021
- [5] 카를 마르크스, 자본론, 비봉출판사, 2015
- [6] Buterin, V., "Ethereum white paper", GitHub repository. 2013 Jan.
- [7] Lucas Jr, Robert E. "Expectations and the Neutrality of Money." Journal of economic theory, 1972

Gwangju Institute of
Science and Technology

School of Information and Communications



My AI Network™

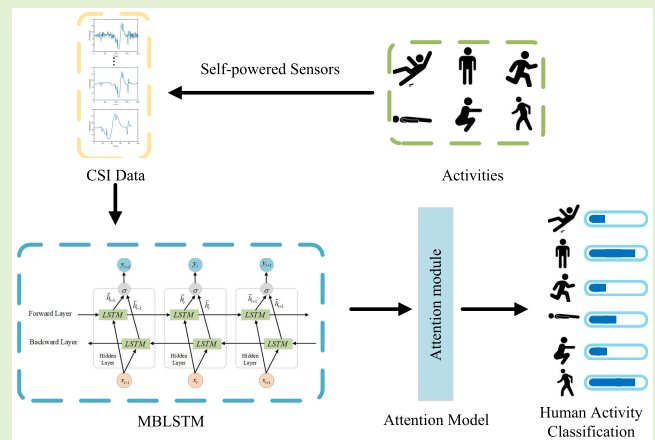
- WorldLand, My AI Network 핵심 기술 연구 결과-

Human Activity Recognition Using Self-Powered Sensors Based on Multilayer Bidirectional Long Short-Term Memory Networks

Jian Su¹, Member, IEEE, Zhenlong Liao¹, Zhengguo Sheng¹, Senior Member, IEEE, Alex X. Liu², Fellow, IEEE, Dilbag Singh, and Heung-No Lee², Senior Member, IEEE

Abstract—Sensor-based human activity recognition (HAR) requires the acquisition of channel state information (CSI) data with time series based on sensors to predict human behavior. Many existing approaches are based on wearable sensors and cameras, which increases the burden and privacy issues for patients. Self-powered sensors are capable of noncontact collection of time series data generated by human activity while ensuring their own stable operation. In this article, we propose a deep-learning-based framework for contactless real-time activity detection of humans using self-powered sensors, which is called multilayer bidirectional long short-term memory (MBLSTM). The collected Wi-Fi CSI data are fed into our proposed network model, which is then used to learn representative features of both sides from the original continuous CSI measurements. The attention model is used to assign different weights to the learned features, and finally, activity recognition is performed. Experimental results show that our proposed method achieves an accuracy of more than 96% for the recognition of six activities in multiple rounds of testing, outperforming other benchmark methods used for comparison.

Index Terms—Bidirectional long short-term memory (BLSTM), channel state information (CSI), deep learning, human activity recognition (HAR), self-powered sensors, Wi-Fi.



NOMENCLATURE

BLSTM Bidirectional long short-term memory.
MBLSTM Multilayer bidirectional long short-term memory.

Manuscript received 2 July 2022; revised 23 July 2022; accepted 23 July 2022. Date of publication 5 August 2022; date of current version 14 September 2023. This work was supported in part by the Natural Science Foundation of China under Grant 61802196, Grant 61872082, and Grant 61472184; in part by the Natural Science Foundation of Jiangsu Province under Grant BK20180791; in part by the Engineering Research Center of Digital Forensics, Ministry of Education; and in part by the National Research Foundation of Korea (NRF) funded by the Government of Korea (MSIP) under Grant NRF-2021R1A2B5B03002118. The associate editor coordinating the review of this article and approving it for publication was Dr. Shahid Mumtaz. (Corresponding author: Jian Su.)

Jian Su and Zhenlong Liao are with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, Jiangsu 210044, China (e-mail: sj890718@gmail.com; liaozhenlong1105@163.com).

Zhengguo Sheng is with the Department of Engineering and Design, University of Sussex, Brighton BN1 9RH, U.K. (e-mail: z.sheng@sussex.ac.uk).

Alex X. Liu is with the Ant Financial Services Group, Hangzhou 310000, China (e-mail: alexliu360@gmail.com).

Dilbag Singh and Heung-No Lee are with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea (e-mail: dilbagsingh@gist.ac.kr; heungno@gist.ac.kr).

Digital Object Identifier 10.1109/JSEN.2022.3195274

CSI Channel state information.
HAR Human activity recognition.
RSS Received signal strength.
APs Access points.
KNN k -nearest neighbor.
STFT Short-time Fourier transform.
GAN Generative adversarial network.
CNN Convolutional neural network.
RF Random forest.
SVM Support vector machine.
LR Logistic regression.
DT Decision tree.
HMM Hidden Markov model.
SAE Sparse autoencoder.

I. INTRODUCTION

IN RECENT years, thanks to the rapid development of Internet-of-Things (IoT) technology, we can get a lot of useful information from different types of sensors in IoT. This information can help IoT technology to be applied in smart cities, smart farms, medical and health services, and so on. The application of IoT sensors in the livestock industry can help practitioners reduce costs and increase efficiency [1].

Use fiber optic sensors for 3D sound source localization [2]. IoT technologies are also gradually entering our daily life and can identify human daily activities. HAR is also receiving increasing attention for research in the field of health detection. For example, we need to understand the health status of elderly people and need to monitor their daily activities [3] over time for fall detection [4] and identification of some diseases that the elderly are prone to, such as Parkinson's [5].

To identify various human activities, many methods have been used in previous work. Cameras [6]–[9], wearable sensors [10], [11], and radio frequency identification (RFID) [12]–[14] have been used for activity recognition. Camera-based systems have the advantage of being able to detect minor human movements. However, these systems face severe problems, such as object blocking and privacy issues. Because of the great recognition accuracy, wearable sensors are also useful in HAR [15]. Wearable sensor-based ones, on the other hand, need the use of additional devices for action recognition, which is both uncomfortable and ineffective. The mobile phone is another popular sensor for recognizing human activity. Smartphones may be considered electricity sensing platforms for HAR since different sensors, such as accelerometers, gyroscopes, and barometers, are incorporated in phones. If the user forgets to carry their smartphone, activity recognition will be turned off. Simultaneously, the operation of the sensors in the phone will be affected by its battery capacity. The usage of Wi-Fi devices for HAR has also been successful currently [16]–[19]. Wi-Fi provides new research directions for universal, nonvisual HAR due to its universality, low cost, and contactless operation. Use self-powered sensors to obtain stable and continuous Wi-Fi signal information.

The basic point of using Wi-Fi to recognize activities is that human motion influences the nearby Wi-Fi signal, and Wi-Fi signals reflected by different activities exhibit different characteristics. RSS, which is most widely practiced in the field of indoor positioning research [20], [21], is the most extensively utilized signal for Wi-Fi. Although it can be used to recognize human activity, it has disadvantages because of noise and unsteady RSS data. Distinguishing different human actions is mainly a matter of analyzing the pattern of the signal, CSI. The most advanced work showed pretty decent recognition accuracy while using a clean Wi-Fi channel in the experiment. However, in the real world, Wi-Fi channels are less than clean. Nowadays, wireless signals abound in indoor places, such as homes, offices, and supermarkets, and there are numerous private APs. Because most systems now utilize stationary Wi-Fi channels for action recognition and CSI acquisition, their performance is extremely vulnerable to cochannel interference, which can significantly decrease the quality of the receiver and distort the extracted recognition features. When classifying activities, traditional classification models utilized in present systems are highly influenced by such distortions. Recently, with the rapid development of deep learning techniques, the method of automatically learning activity features in CSI using deep learning has provided a completely new way of thinking for HAR [22]. There is

also experience in combining machine learning and sensors in previous work, for example, fiber optic tactile sensors combined with machine learning algorithms for surface roughness recognition [23].

The advantage of long short-term memory (LSTM) networks to automatically learn meaningful features and encode data is widely used in deep learning. The traditional LSTM only handles the forward continuous CSI data, which means that the backward CSI data are not used effectively in training. Future CSI data, we believe, will be important for recognizing human activities. Furthermore, typical LSTM sequence properties may contribute differently to the HAR challenge. The learned characteristics, on the other hand, make an equivalent contribution to the final identification of human actions in the classic LSTM technique. We provide a MBLSTM based on Wi-Fi CSI data for HAR in this research paper. Stacking LSTM hidden layers gives more depth to the model and more accurate descriptions obtained as a deep learning technique while increasing the depth of the network, improving the efficiency of training, and obtaining higher accuracy. An MBLSTM network consisting of multiple forward and backward LSTM layers can handle both forward and backward continuous CSI measurements. Furthermore, the attention mechanism can give more weight to more essential characteristics and time steps, resulting in higher generalization for human activity detection. The effectiveness of the proposed personnel activity detection algorithm based on wireless CSI measurement is verified by real experiments. The results are compared to several published benchmark approaches.

In this article, the major contribution of our work is that we establish a framework called MBLSTM to recognize human activities. The following is a detailed description.

- 1) We designed an MBLSTM network to collect Wi-Fi CSI data for autonomous feature extraction and selection using self-powered sensors. Use self-powered sensors to continuously and steadily collect Wi-Fi time series information under different activities, and match this different information with different activities.
- 2) Continuous CSI data in both forward and reverse directions are processed by layering several BLSTM networks. The MBLSTM can simultaneously consider the information of different past and future actions in CSI data, thus bringing richer information reference for feature learning and using it can also speed up the convergence process of the training dataset.
- 3) The MBLSTM network uses an attention model to learn the relevance between activity features and time series. For the final personnel activity recognition, more main features and time series are assigned greater weights, resulting in improved recognition performance.

The rest of this article is organized as follows. Section II reviews some state-of-the-art work on using Wi-Fi signals to identify human activities, and Section III describes the channel sensing model and the MBLSTM network, as well as the proposed approach. Section IV describes the experimental setup and data. Then, this section shows and analyzes the experimental results. Finally, Section V summarizes this work.

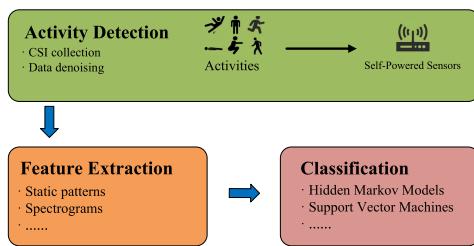


Fig. 1. Basic framework of the Wi-Fi-based activity recognition system.

II. RELATED WORK

As illustrated in Fig. 1, a conventional Wi-Fi-based activity recognition system is composed of three parts.

- 1) *Filtering and Monitoring Channel Status*: Human activity is detected using a self-powered sensor. The human body activity affects the Wi-Fi signal, and this pair of signals can be observed. Therefore, the first step of the activity recognition system is to collect the original signal and denoise it to reveal the changes caused by human activity.
- 2) *Extraction of Features*: The CSI data from this denoising step are still not directly usable, and the next task is to discover and extract features from the existing behavioral data that are initially compatible with the technical requirements. At present, the signal feature extraction method contains the following three kinds: the time-domain analysis method, the frequency-domain analysis method, and the combined method of time–frequency analysis.
- 3) *Training and Recognition*: After getting the feature dataset, the first operation is to distinguish the dataset into the training set and the test set, and choosing the proper division ratio is a key part to ensure the effect of behavior recognition. The next step is to select the appropriate classification algorithm to train and test the data.

Due to the common presence of Wi-Fi in everyday life, many research teams have developed several activity recognition systems using Wi-Fi signals. Sigg *et al.* [24] proposed a wireless HAR system that analyzes the RSS information of the interfered Wi-Fi signal for activity recognition. They extracted several important features from RSS data and used a KNN classifier to recognize four daily activities. Abudulaziz Ali Haseeb and Parasuraman [25] designed an RSS-based gesture recognition system on cell phones. The system uses deep learning networks for gesture recognition and achieves high recognition accuracy. Due to multipath and fading effects, the collection of raw RSS data containing actions can be unstable and noisy, so the performance of using RSS to recognize activities with actions is very limited, even for simple actions. Wi-Fi’s more steady and informative CSI has received a lot of attention recently. Zhang *et al.* [26] investigated the sensitivity of Wi-Fi signals theoretically and proposed a Fresnel zone method to recognize human activity using Wi-Fi CSI data.

Some special features may need to be carefully designed using domain knowledge in order to recognize certain actions using Wi-Fi CSI measurements. When used to recognize other activities, these features may not perform effectively. For example, the traditional KNN method, which has a simple idea, is applicable to multiclassification problems. However, when the sample distribution is unbalanced, the new sample will be classified as the dominant sample, so it cannot better approximate the actual classification result. Furthermore, handcrafted characteristics will gradually lose several of the implicit qualities that are important for recognizing human activity. Deep learning is a useful tool for automatically learning the differentiating features that are used to recognize the human activity.

Deep learning is a type of machine learning method that uses a deep neural network to classify data. In most cases, accurate features need to be identified for input to the training model, and the model classifies and outputs results based on these features. As a result, well-designed features are essential for accurate behavior recognition and have a significant effect on classification accuracy. Some feature extraction, on the other hand, may depend on empirical experience, lowering classification accuracy. Deep learning, unlike machine learning, generally does not require feature extraction stages since a deep neural network is able to automatically identify and extract features from training data. Deep learning allows us a new method to classify data and can deal with enormous amounts of data. In other words, the most significant advantage of deep learning is that it does not require preprocessing of data in order to obtain data features. Meanwhile, deep learning can automatically compute large-scale unknown parameters in neural networks through the training process. Usually, the process of neural network training consumes a lot of practice, but the results achieved are satisfactory. Deep learning algorithms are widely applied in various fields, including picture target identification, natural language processing, video classification, visual arts, and so on [27].

Damodaran *et al.* [28] used a device-free approach (CSI) to identify human activities. Wavelet analysis was used for preprocessing and feature extraction. As a result, they were able to recognize walking, sitting, standing, and running activities. High-bandwidth noise was removed using the principal component analysis by Moshiri *et al.* [29]. The signal was transformed to the frequency domain using STFT, and new data were generated using GANs. The LSTM algorithm was used for classification. The accuracy was 87.2% using 50% of the “real” data plus 50% of the synthetic data and 92.8% using a set of all “real” data.

CNN is a very popular deep learning method that automates feature extraction and can easily handle high-latitude data. However, when the network level is too deep, modifying the parameters using BP propagation will cause the parameters near the input layer to change more slowly, and the pooling layer will lose a lot of valuable information and ignore the local-to-whole correlation.

Since, for different activities, CSI measurements are continuous measurements with temporal information, BLSTM

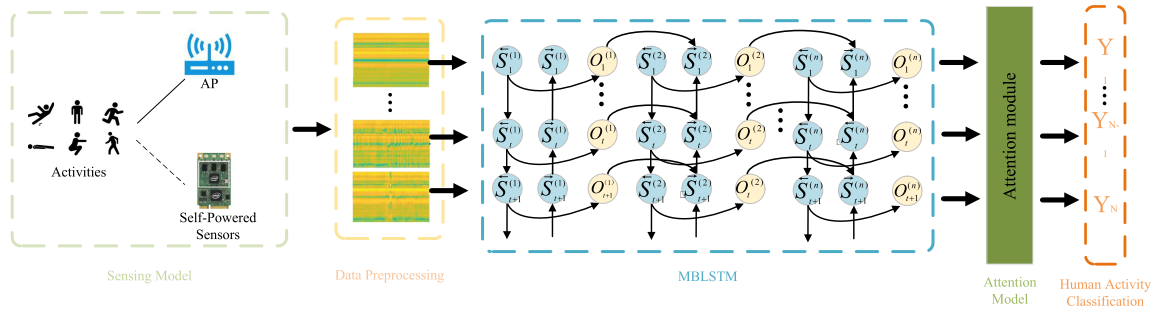


Fig. 2. Proposed MBLSTM framework for CSI-based HAR.

capable of encoding temporal information is a good candidate for automatic feature learning. BLSTM includes both forward and backward processes of feature learning. As a result, when evaluating the current hidden state of the LSTM, BLSTM can take into account both past and future information, resulting in richer information features. We propose stacked multilayer BLSTM networks for human action recognition. Each layer of the BLSTM neural network automatically learns the input action features and passes the learned features to the next layer. At the same time, the feature sequences learned in one temporal instance may contribute differently to the final HAR. Furthermore, the significance of CSI collected at different time stages may differ. Therefore, in order to assign different weights to different action features in the training for the purpose of reducing the training time and improving the accuracy of the model, we add an attention mechanism to the proposed network model.

III. PROPOSED METHOD

A. System Overview

The proposed MBLSTM framework is shown in Fig. 2. First, we use a router and a self-powered sensor to collect CSI signals from Wi-Fi of human actions. Second, we input the processed CSI signals into the MBLSTM framework to automatically learn the forward and backward features. There are 200 hidden nodes in the bidirectional LSTM used for feature learning in this experiment. Since the attention model has no available prior information, it can only use the features learned from BLSTM as input to derive an attention matrix representing the importance of features and time steps. Then, we use element multiplication to merge the learned features with the attention matrix to obtain the modified feature matrix with attention. After that, the feature matrix is flattened into feature vectors for final classification using the flattened layer. Finally, the softmax classification layer is used to identify different activities with the final feature vectors.

B. Channel Sensing Model

Wi-Fi signals are known to fluctuate significantly when objects move within the region of interest. The Fresnel zone model is introduced as a result of this to explore how the Wi-Fi signals on these receiving antennas change as a result of different activities. Furthermore, we infer potential behavioral

information from such activity-induced signal fluctuations. Thus, we use the Intel 5300 NIC, a self-powered sensor, to collect the reflected Wi-Fi information.

In recent years, the Fresnel zone model has been applied to the research of human action recognition based on wireless sensing. It refers to the wireless electromagnetic wave in the transmission process, the formation of the transceiver at both ends of the transceiver device, and the location of the transceiver device as the focus of the ellipse-shaped area; the area is the wireless electromagnetic wave intensity concentration area. One of the most important zones is the first Fresnel zone, where most of the energy of the wireless signal is located. If there is an obstacle in this region, it will affect the wireless signal. The wireless signal will form multiple propagation paths from the receiver (Rx) to the transmitter (Tx), and the direct propagation path that passes through both the transmitter and receiver is called the line-of-sight (LoS) path. When the wireless signal transmission encounters an obstacle, the transmission path produces reflection, scattering and diffraction, called non-line-of-sight (NLoS).

Through the analysis and study of the received signals, the researchers found the characteristics of the changes brought by the human body movements on the signal propagation. Meanwhile, establish the relationship between these features and the mapping of different activities, which built the foundation for Wi-Fi-based HAR.

The phenomenon that different actions have different effects on Wi-Fi signals is a major discovery that the Fresnel zone model can be applied to the field of action recognition. Specifically, different activities cause significant differences in the speed of signal dynamic paths. Furthermore, CSI's amplitude attenuation and phase change can capture these specific patterns. It demonstrates the feasibility and application of using unique CSI variations to effectively and precisely identify and recognize different human activities.

C. MBLSTM Neural Network

In the case of multilayer stacking, each layer of the BLSTM neural network is composed of a forward recurrent network and a backward recurrent network. The combination of the output results of the forward LSTM and the backward LSTM of the previous layer is sent to the next layer of the network. Fig. 3 illustrates the MBLSTM

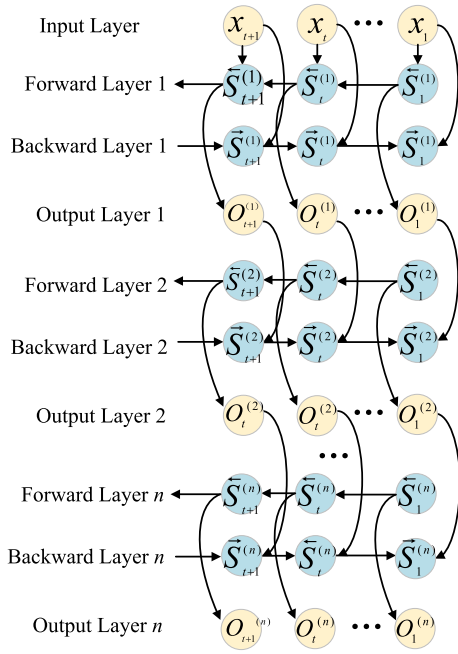


Fig. 3. Structure of the MBLSTM neural network.

network framework structure

$$\begin{aligned}
 S_t^{\prime(1)} &= f\left(U^{\prime(1)}x_t + W^{(1)}S_{t-1}^{\prime}\right) \\
 S_t^{(1)} &= f\left(U^{(1)}x_t + W^{(1)}S_{t-1}\right) \\
 &\dots \\
 S_t^{\prime(i)} &= f\left(U^{\prime(i)}S_t^{(i-1)} + W^{\prime(i)}S_{t+1}^{\prime}\right) \\
 S_t^{(i)} &= f\left(U^{(i)}S_t^{\prime(i-1)} + W^{(i)}S_{t-1}\right) \\
 O_t &= g\left(V^{(i)}S_t^{(i)} + V^{\prime(i)}S_t^{\prime(i)}\right). \quad (1)
 \end{aligned}$$

The output is determined by the sum of each layer's positive and negative computations. Here, $S_{t-1}^{(i)}$ and $S_t^{\prime(i)}$ are the values of the i th hidden layer at times $t-1$ and t , respectively. Forward and backward computations do not share weights; $V^{(i)}$, $U^{(i)}$, and $W^{(i)}$ are the weight matrices of the i th hidden layer to the output layer, the input layer to the hidden layer, and the hidden layer. $V^{\prime(i)}$, $U^{\prime(i)}$, and $W^{\prime(i)}$ are the backward weight matrices used for the computations. i is the number of BLSTM layers, and $i = 0, 1, 2, \dots, \infty$ is the output layer's value.

D. Attention Model

The attentional model is developed to be used for image recognition [30]. The concept was inspired by the human visual system, which says that, during picture recognition, humans always focus on a certain portion of the image and adjust their attention over time. During the recognition work, the attention model allows the computer to attend to the area of interest while blurring other areas. Recently, attention models have been used in language processing, proving that it is clearly effective [31]. For example, in the popular encoder-decoder method for natural language processing, the input sentence is encoded as a fixed vector that is translated

throughout the translation process, meaning that, at every time step, all words in the input sentence contribute equally to the translation. This task of processing sentence translation is inefficient. When the encoder model is utilized with the attention model, translations will focus more on the words that are more relevant to the current translation process at different time steps. Since the MBLSTM network learns high-dimensional sequence features, individual features and time series may contribute differently to the final recognition results. We try to use an attention model to intelligently learn the effects of different actions of features and assign weights according to their importance.

In the recognition system, there is no usable *a priori* information for training. As a result, the attention model, also known as self-attention, will utilize the sequential features learned by MBLSTM as input. This attention model is shown in a simple example here. Given n feature vectors \mathbf{h}_i , $i = 1, 2, \dots, n$ that can be obtained from the feature learning network, we build a score function $\Phi(\cdot)$ to evaluate the significance of each feature vector by computing the score s_i as follows:

$$s_i = \Phi\left(\mathbf{W}^T \mathbf{h}_i + b\right) \quad (2)$$

where \mathbf{W}^T and b are the weight vector and bias, respectively. Any activation function in a neural network, such as tanh, relu, or linear, can be used to build the score function. We can normalize each feature vector's score utilizing the softmax function, which is written as

$$a_i = \text{softmax}(s_i) = \frac{\text{esp}(s_i)}{\sum_i (s_i)}. \quad (3)$$

The final output feature \mathbf{O} of the attention model is the product of the vector and its normalization score as follows:

$$\mathbf{O} = \sum_{i=1}^n a_i * \mathbf{h}_i. \quad (4)$$

E. Training Proposed Method

To identify the model parameters, the proposed MBLSTM framework is trained using CSI data with real labels. First, all parameters are randomly given. The CSI data is then sent into MBLSTM, which uses it to predict the labels. The category cross-entropy errors are measured and backpropagated using a gradient-based optimization approach to adjust the model parameters utilizing the given true labels. We utilize ADAM [32] to calculate the adaptive learning rate for each parameter in the optimization process efficiently.

In learning-based systems, overfitting is a typical problem. To avoid overfitting, we utilize the ADAM optimizer. It provides adaptive learning rates for various parameters. Furthermore, the suggested attention method will only choose a few significant features and time series, decreasing the possibility of overfitting.

IV. EXPERIMENTS AND ANALYSIS

In this section, we first introduce our experimental settings in detail and then present the extensive experimental results that validate the effectiveness of our model.

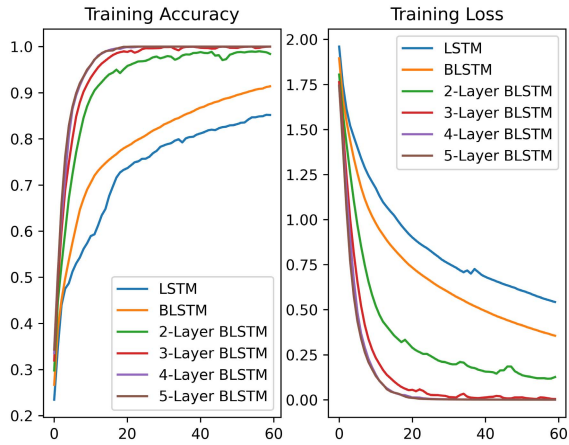


Fig. 4. Trend of accuracy rate of different network trainings.

A. Experiments' Settings

We compared the proposed method to several benchmark CSI-based human activity identification algorithms to evaluate how effective it is. According to Yousefi *et al.* [33], the RF model outperforms SVMs, LR, and DT in Wi-Fi-based HAR. In [34], HMMs have also been found to be useful for recognizing human activity. As a result, we compared our method to these two handmade methods. Manual feature extraction is described in detail in [33]. We also compare it to other deep-learning-based approaches that can learn features automatically, such as SAE [34], [35] and traditional LSTM [33]. The SAE algorithm is an unsupervised algorithm that automatically learns features from unlabeled data and can give a better feature description than the original data. Validation sets from the training examples were used to fine-tune the parameters of all methods. For evaluation, tenfold cross-validation has been used. We divided all of the data tenfold at random. Then, we select onefold of data for testing and the rest for training and, finally, get ten times. The average of all ten runs determines the final recognition accuracy. The dataset used for comparison was taken by Yousefi *et al.* [33] from an office. A router was used as a transmitter, and a laptop with an Intel 5300 NIC was used as a receiver. The sampling frequency was 1 kHz, with three antennas and 30 subcarriers, and the size of the original CSI data was 90. The window size used for data segmentation was a sliding window of 2 s. Transmitters and receivers were separated by three meters under LOS conditions. Each person performed each activity for 20 s during data collection. Note that the person remains stationary at the beginning and end of the activity. Six persons were involved in the data process of collecting, which included six normal daily activities: lie down, fall, run, sit down, stand up and walk. Every volunteer performed 20 rounds of each activity; the resulting dataset was approximately 17 GB in size. All experiments were performed on a workstation in our lab using python to run the code. The workstation is equipped with an eight-core, 16-thread Intel i9-9900 CPU and an NVIDIA GeForce RTX 2080 GPU.

We compare the trend of accuracy and loss of BLSTM networks with the different number of layers in the training

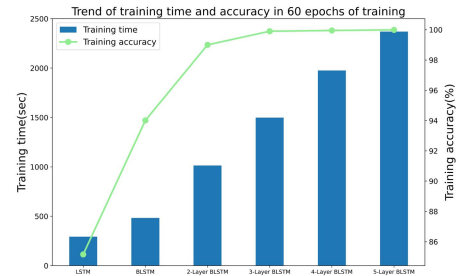


Fig. 5. Trend of training time and accuracy in 60 epochs of training.

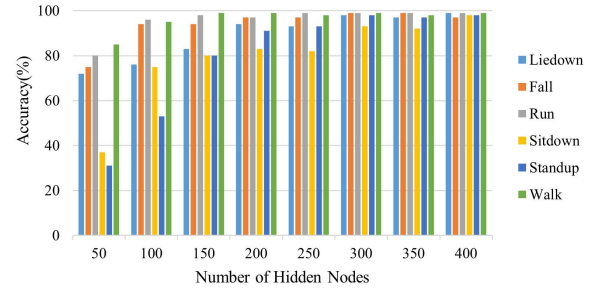


Fig. 6. Recognition accuracy of each activity with different numbers of hidden nodes.

dataset. Fig. 4 shows that the LSTM and BLSTM networks converge more slowly, with accuracy barely reaching 90% at the 60th round of training. The multilayer BLSTM network, on the other hand, converges quickly, with accuracy exceeding 90% at about ten rounds of training, approaching 100% at close to 20 rounds, and preserving stability in accuracy during subsequent training.

Although the training converges faster as the number of BLSTM layers increases, it is not better to have more layers. As the number of layers increases, the network structure becomes increasingly large, which means that more and more computational resources will be used, and more time will be consumed in training. As shown in Fig. 5, we run experiments using 200 hidden nodes. The results show that the more complex the network structure is, the time for training increases significantly. It is obvious that, with the same number of training rounds, the overall training accuracy does not improve much after increasing the BLSTM network to three layers, which are close to 100%, indicating that the limit has been approached. However, the training time spent by each network differs greatly. Considering all factors, we choose the three-layer BLSTM network as the network model for this experiment in order to minimize the computer resources consumed while ensuring high accuracy.

1) *Impact of the Number of Hidden Nodes:* We find that the number of LSTM hidden nodes has a large impact on the experimental results. As a result, we performed a second experiment to see how this parameter affected the accuracy of activity recognition. The results of the experiment are shown in Fig. 6. When using 50 hidden nodes, the recognition accuracy was low for actions, especially for the two activities “sit down” and “stand up,” which, we guess, are too similar. When the number of hidden nodes is raised, the recognition performance

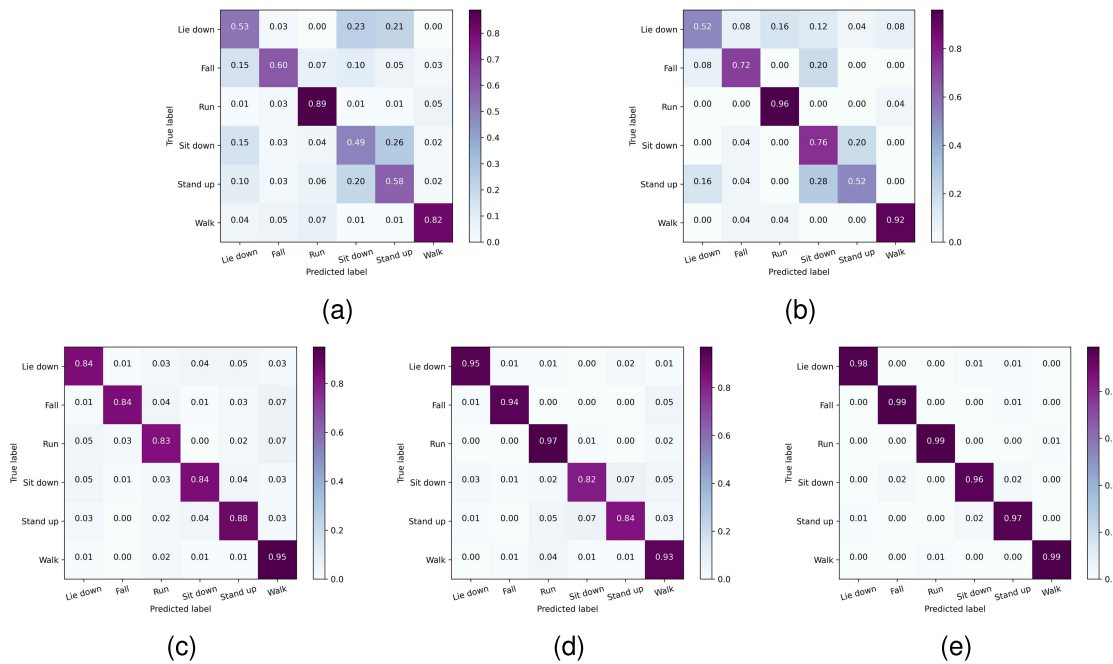


Fig. 7. Confusion matrix of all benchmarks and proposed MBLSTM methods on the dataset. (a) RF. (b) HMM. (c) SAE. (d) LSTM. (e) MBLSTM.

TABLE I
TRAINING AND VALIDATION TIMES FOR DIFFERENT NUMBERS OF HIDDEN NODES

Nodes	50	100	150	200	250	300	350	400
Time								
Training(s)	245.85	349.14	545.11	785.69	880.83	1043.76	1238.80	1444.24
Validation(s)	7.63	7.46	7.04	7.64	7.21	7.24	7.86	8.35

TABLE II
TRAINING AND TESTING TIMES FOR DIFFERENT METHODS

Time	RF	HMM	SAE	LSTM	MBLSTM
Training(s)	5.31	0.024	158.16	493.23	1551.14
Testing(s)	0.008	0.17	0.19	3.54	8.72

of each activity is improved, and after the number reaches 300, the accuracy tends to be stable. As shown in Table I, we use a three-layer BLSTM network, and in the same 30 rounds of training, the more the hidden nodes, the longer the training time, and we choose to use 200 hidden nodes in the MBLSTM.

2) *Time Complexity*: Deep-learning-based approaches’ time complexity is a common issue. We compared the training time and testing time of some methods using the same dataset. Table II shows the training time and testing time for all methods. It can be clearly seen that algorithms using deep learning methods have much longer training times than a typical machine learning algorithm. The proposed MBLSTM consumes the longest training time of all methods using deep learning. All of the approaches have short testing times according to Table II. The proposed MBLSTM, for example, has a testing time of 8.72 s for 420 test samples. This signifies that each sample will be tested for 0.0208 s. The window size for data segmentation is 4 s for each case. We believe that our

proposed MBLSTM approach, which is based on Wi-Fi CSI, may be utilized for real-time personnel activity recognition.

B. Experimental Results

Fig. 7 shows the confusion matrix of all benchmarks and proposed MBLSTM methods on the dataset. Activity recognition algorithms that need manual feature extraction, such as RF and HMM, perform the poorest. The HMM algorithm performs significantly better than the RF algorithm. Unlike RF and HMM manual feature extraction, SAE algorithms using deep learning methods have better performance. This demonstrates the effectiveness of using the SAE method for automatic feature learning. The LSTM network outperforms the SAE method because it incorporates the temporal factors in the CSI sequences into feature learning. Due to the inclusion of the attention model and the structure of the multilayer bidirectional LSTM in our proposed method, our MBLSTM method achieves excellent recognition results in recognizing six daily activities. For all six daily activity recognition, the accuracy is greater than or equal to 96%, which is sufficient for most recognition situations.

The accuracy of recognition varies greatly depending on the activity. Higher physical activities, such as “fall,” “walk,” and “run,” show greater recognition performance. This is due to the fact that these activities have a large impact on the features of the collected CSI data. It is also evident that most methods have relatively low accuracy for recognizing the activity of “sit down.” This might be because this activity has the same effect on CSI features as the “lie down” and “stand up” activities. It is worth noting that the RF method’s recognition accuracy with handmade features is much lower than 50%. The “fall” activity is the most important of these six, especially for the elderly [36]. The proposed MBLSTM approach can

recognize “fall” activities with 99% accuracy, which will be useful in a wide variety of medical applications. The extended training period of the deep-learning-based approach is one of its drawbacks. However, this time-consuming procedure only has to be completed once. It is worth noting that deep-learning-based methods can be tested online quickly enough for most real-time applications.

V. CONCLUSION

In this article, we use self-powered sensors to collect Wi-Fi time series information and propose a multilayer BLSTM network for extracting Wi-Fi signal feature information used for HAR by improving the traditional LSTM model. In both directions, the BLSTM network can learn important sequential features from original Wi-Fi CSI data. The multilayer BLSTM network can enhance accuracy by accelerating convergence during training. We evaluated the method in real environments and compared it to a variety of benchmark methods, such as RF, HMMs, SAEs, and traditional LSTM. The proposed MBLSTM for Wi-Fi CSI-based personnel activity recognition has demonstrated higher performance in experiments. Although our method has a high recognition rate for single-person activities, there is still a big room for improvement in multiperson activities. For future work, we hope to improve the accuracy of multiperson activity recognition and the compatibility of the system with different environments. Glowinski *et al.* [37] used an inertial measurement device to calculate acceleration. This inspired our proposed method helps to recognize the type of body movement. In case of a car accident, it can help to determine the posture of the injured person.

REFERENCES

- [1] S. Neethirajan, “The role of sensors, big data and machine learning in modern animal farming,” *Sens. Bio-Sens. Res.*, vol. 29, Aug. 2020, Art. no. 100367.
- [2] S. E. Hayber and S. Keser, “3D sound source localization with fiber optic sensor array based on genetic algorithm,” *Opt. Fiber Technol.*, vol. 57, Jul. 2020, Art. no. 102229. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1068520020302194>
- [3] T. Huynh-The, C.-H. Hua, N. A. Tu, and D.-S. Kim, “Physical activity recognition with statistical-deep fusion model using multiple sensory data for smart health,” *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1533–1543, Feb. 2021.
- [4] Y. Wang, S. Yang, F. Li, Y. Wu, and Y. Wang, “FallViewer: A fine-grained indoor fall detection system with ubiquitous Wi-Fi devices,” *IEEE Internet Things J.*, vol. 8, no. 15, pp. 12455–12466, Aug. 2021.
- [5] X. Cao *et al.*, “Video based shuffling step detection for parkinsonian patients using 3D convolution,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 641–649, 2021.
- [6] J. K. Aggarwal and M. S. Ryoo, “Human activity analysis: A review,” *ACM Comput. Surv.*, vol. 43, no. 3, pp. 1–43, Apr. 2011, doi: [10.1145/1922649.1922653](https://doi.org/10.1145/1922649.1922653).
- [7] S. Jiang, Y. Qi, H. Zhang, Z. Bai, X. Lu, and P. Wang, “D3D: Dual 3-D convolutional network for real-time action recognition,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 4584–4593, Jul. 2021.
- [8] D. C. Luvizon, D. Picard, and H. Tabia, “Multi-task deep learning for real-time 3D human pose estimation and action recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 8, pp. 2752–2764, Aug. 2021.
- [9] M. Moniruzzaman, Z. Yin, Z. He, R. Qin, and M. C. Leu, “Human action recognition by discriminative feature pooling and video segment attention model,” *IEEE Trans. Multimedia*, vol. 24, pp. 689–701, 2022.
- [10] L. Tong, H. Ma, Q. Lin, J. He, and L. Peng, “A novel deep learning Bi-GRU-I model for real-time human activity recognition using inertial sensors,” *IEEE Sensors J.*, vol. 22, no. 6, pp. 6164–6174, Mar. 2022.
- [11] D. Chen, S. Yongchareon, E. M.-K. Lai, J. Yu, and Q. Z. Sheng, “Hybrid fuzzy C-means CPD-based segmentation for improving sensor-based multiresident activity recognition,” *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11193–11207, Jul. 2021.
- [12] X. Fan, W. Gong, and J. Liu, “TagFree activity identification with RFIDs,” *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 1, pp. 1–23, Mar. 2018, doi: [10.1145/3191739](https://doi.org/10.1145/3191739).
- [13] G. A. Oguntala, Y. F. Hu, A. A. S. Alabdullah, R. A. Abd-Alhameed, M. Ali, and D. K. Luong, “Passive RFID module with LSTM recurrent neural network activity classification algorithm for ambient-assisted living,” *IEEE Internet Things J.*, vol. 8, no. 13, pp. 10953–10962, Jul. 2021.
- [14] F. Wang, J. Liu, and W. Gong, “Multi-adversarial in-car activity recognition using RFIDs,” *IEEE Trans. Mobile Comput.*, vol. 20, no. 6, pp. 2224–2237, Jun. 2021.
- [15] O. D. Lara and M. A. Labrador, “A survey on human activity recognition using wearable sensors,” *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1192–1209, 3rd Quart., 2013.
- [16] J. Zhang *et al.*, “Data augmentation and dense-LSTM for human activity recognition using WiFi signal,” *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4628–4641, Mar. 2021.
- [17] C. Xiao, Y. Lei, Y. Ma, F. Zhou, and Z. Qin, “DeepSeg: Deep-learning-based activity segmentation framework for activity recognition using WiFi,” *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5669–5681, Apr. 2021.
- [18] R. H. Venkatnarayan, S. Mahmood, and M. Shahzad, “WiFi based multi-user gesture recognition,” *IEEE Trans. Mobile Comput.*, vol. 20, no. 3, pp. 1242–1256, Mar. 2021.
- [19] J. Zuo, X. Zhu, Y. Peng, Z. Zhao, X. Wei, and X. Wang, “A new method of posture recognition based on WiFi signal,” *IEEE Commun. Lett.*, vol. 25, no. 8, pp. 2564–2568, Aug. 2021.
- [20] C.-H. Lin *et al.*, “An indoor positioning algorithm based on fingerprint and mobility prediction in RSS fluctuation-prone WLANs,” *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 5, pp. 2926–2936, May 2021.
- [21] D. Yu and C. Li, “An accurate WiFi indoor positioning algorithm for complex pedestrian environments,” *IEEE Sensors J.*, vol. 21, no. 21, pp. 24440–24452, Nov. 2021.
- [22] Z. Wang *et al.*, “A survey on human behavior recognition using channel state information,” *IEEE Access*, vol. 7, pp. 155986–156024, 2019.
- [23] S. Keser and Ş. E. Hayber, “Fiber optic tactile sensor for surface roughness recognition by machine learning algorithms,” *Sens. Actuators A, Phys.*, vol. 332, Dec. 2021, Art. no. 113071.
- [24] S. Sigg, S. Shi, F. Buesching, Y. Ji, and L. Wolf, “Leveraging RF-channel fluctuation for activity recognition: Active and passive systems, continuous and RSSI-based signal features,” in *Proc. Int. Conf. Adv. Mobile Comput. Multimedia (MoMM)*. New York, NY, USA: Association for Computing Machinery, 2013, pp. 43–52, doi: [10.1145/2536853.2536873](https://doi.org/10.1145/2536853.2536873).
- [25] M. Abudulaziz Ali Haseeb and R. Parasuraman, “Wisture: RNN-based learning of wireless signals for gesture recognition in unmodified smartphones,” 2017, *arXiv:1707.08569*.
- [26] D. Zhang, H. Wang, and D. Wu, “Toward centimeter-scale human activity sensing with Wi-Fi signals,” *Computer*, vol. 50, no. 1, pp. 48–57, Jan. 2017.
- [27] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, pp. 436–444, Feb. 2015, doi: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
- [28] N. Damodaran, E. Haruni, M. Kokhkarova, and J. Schäfer, “Device free human activity and fall recognition using WiFi channel state information (CSI),” *CCF Trans. Pervasive Comput. Interact.*, vol. 2, no. 1, pp. 1–17, 2020, doi: [10.1007/s42486-020-00027-1](https://doi.org/10.1007/s42486-020-00027-1).
- [29] P. F. Moshiri, H. Navidan, R. Shahbazian, S. A. Ghorashi, and D. Windridge, “Using GAN to enhance the accuracy of indoor human activity recognition,” Apr. 2020, *arXiv:2004.11228*.
- [30] J. Ji, C. Xu, X. Zhang, B. Wang, and X. Song, “Spatio-temporal memory attention for image captioning,” *IEEE Trans. Image Process.*, vol. 29, pp. 7615–7628, 2020.
- [31] A. Galassi, M. Lippi, and P. Torrioni, “Attention in natural language processing,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 10, pp. 4291–4308, Oct. 2021.
- [32] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–15.
- [33] S. Yousefi, H. Narui, S. Dayal, S. Ermon, and S. Valaei, “A survey on behavior recognition using WiFi channel state information,” *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 98–104, Oct. 2017.
- [34] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, “Device-free human activity recognition using commercial WiFi devices,” *IEEE J. Sel. Area Commun.*, vol. 35, no. 5, pp. 1118–1131, Mar. 2017.

- [35] Q. Gao, J. Wang, X. Ma, X. Feng, and H. Wang, "CSI-based device-free wireless localization and activity recognition using radio image features," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10346–10356, Nov. 2017.
- [36] C. Wang *et al.*, "Low-power fall detector using triaxial accelerometry and barometric pressure sensing," *IEEE Trans. Ind. Informat.*, vol. 12, no. 6, pp. 2302–2311, Dec. 2016.
- [37] S. Glowinski, S. Majdanik, A. Glowinska, and E. Majdanik, "Trauma in a shaken infant? A case study," *Aggression Violent Behav.*, vol. 56, Jan. 2021, Art. no. 101515. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1359178920302196>



Jian Su (Member, IEEE) received the B.S. degree in electronic and information engineering from Hankou University, Wuhan, China, in 2008, the M.S. degree in electronic circuit and system from Central China Normal University, Wuhan, in 2012, and the Ph.D. (Hons.) degree in communication and information systems from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2016.

He has been a Lecturer with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China, since 2017. His current research interests cover the Internet of Things, radio frequency identification (RFID), and wireless sensors' networking.

Dr. Su is also a member of the Association for Computing Machinery (ACM).



Zhenlong Liao received the B.S. degree in Internet-of-Things Engineering from the Binjiang College, Nanjing University of Information Engineering, Nanjing, China, in 2020. He is currently pursuing the M.S. degree in computer science and technology with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing.

His research interests include Wi-Fi-based human activity sensing and the Internet of Things.



Zhengguo Sheng (Senior Member, IEEE) received the B.Sc. degree from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2006, and the M.S. and Ph.D. (Hons.) degrees from Imperial College London, London, U.K., in 2007 and 2011, respectively.

He has been a Senior Lecturer with the Department of Engineering and Design, University of Sussex, Brighton, U.K., since 2015. His current research interests cover the Internet of Things (IoT), connected vehicles, and cloud/edge computing.



Alex X. Liu (Fellow, IEEE) received the Ph.D. degree in computer science from The University of Texas at Austin, Austin, TX, USA, in 2006.

He was a Professor with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI, USA. He is currently an Adjunct Professor with the Shandong Provincial Key Laboratory of Computer Networks, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan, China, and the Chief Scientist with the Ant Group, Hangzhou, China. His research interests focus on networking, security, and privacy.

Dr. Liu is also an IET Fellow and an Association for Computing Machinery (ACM) Distinguished Scientist. He received the IEEE & IFIP William C. Carter Award in 2004, the National Science Foundation CAREER Award in 2009, the Michigan State University Withrow Distinguished Scholar (Junior) Award in 2011, and the Michigan State University Withrow Distinguished Scholar (Senior) Award in 2019. He also received the Best Paper Awards from the IEEE International Conference on Sensing, Communication and Networking (SECON) 2018, the International Conference on Network Protocols (ICNP) 2012, the IEEE Symposium on Reliable Distributed Systems (SRDS) 2012, and the USENIX Large Installation System Administration Conference (LISA) 2010. He has served as the TPC Co-Chair of ICNP 2014 and IFIP Networking 2019. He has served as an Editor for IEEE/ACM TRANSACTIONS ON NETWORKING and an Area Editor for *Computer Communications*. He is also an Associate Editor for IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING and IEEE TRANSACTIONS ON MOBILE COMPUTING.



Dilbag Singh received the Ph.D. degree in computer science and engineering from the Thapar Institute of Engineering and Technology, Patiala, India, in 2019.

He has worked as an Assistant Professor at three well-known universities in India: Chandigarh University, Mohali, India, Manipal University Jaipur, Jaipur, India, and Bennett University, Greater Noida, India. In 2021, he moved to the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, South Korea, where he is currently affiliated. He has published more than 100 research articles in Science Citation Index (SCI)/Science Citation Index Expanded (SCIE)-indexed journals. He has also submitted five patents and has published three books and two book chapters. His H-index is 34. His research interests include image processing, computer vision, deep learning, metaheuristic techniques, and information security.

Dr. Singh has acted as a Lead Guest Editor/an Editorial Board Member of many SCI-/SCIE-indexed journals, such as *Journal of Healthcare Engineering*, *Mathematical Problems in Engineering*, and *Journal of Intelligent and Fuzzy Systems*. He was in the top 2% list issues by World Ranking of Top 2% Scientists in 2021. He was part of the 11 Web of Science-/Scopus-indexed conferences.



Heung-No Lee (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of California at Los Angeles, Los Angeles, CA, USA, in 1993, 1994, and 1999, respectively.

He was a Research Staff Member with HRL Laboratories, LLC, Malibu, CA, USA, from 1999 to 2002. From 2002 to 2008, he was an Assistant Professor with the University of Pittsburgh, Pittsburgh, PA, USA. In 2009, he moved to the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, South Korea, where he is currently affiliated. His research interests include information theory, signal processing theory, blockchain, communications/networking theory, and their application to wireless communications and networking, compressive sensing, future Internet, and brain-computer interface.

Dr. Lee received several prestigious national awards, including the Top 100 National Research and Development Award in 2012, the Top 50 Achievements of Fundamental Researches Award in 2013, and the Science/Engineer of the Month in January 2014.

Multimodal Sparse Representation-Based Classification Scheme for RF Fingerprinting

Kiwon Yang¹, Jusung Kang, Jehyuk Jang¹, and Heung-No Lee¹, *Senior Member, IEEE*

Abstract—In this letter, we propose a multimodal method for improving radio frequency (RF) fingerprinting performance that uses multiple features cultivated from RF signals. Combining multiple features, including a falling transient feature that has not previously been used in RF fingerprinting studies, we aim to demonstrate that the proposed method results in improved accuracy. We show that a sparse representation-based classification (SRC) scheme can be a good platform for combining multiple features. The experimental results on RF signals acquired from eight walkie-talkies show that the RF fingerprinting accuracy of the proposed method improves significantly as the number of features increases.

Index Terms—Classification algorithm, feature extraction, multimodality, RF fingerprinting, radio frequency identification.

I. INTRODUCTION

CLASSIFYING radio frequency (RF) signals is useful in electronic warfare to identify the radio transmission signals of adversaries [1]. For the classification to work well, the availability of good features and a simple but robust technique are essential. A feature is a sample vector cultivated from the transmitted RF signals and bears unique information about the pertinent device. Identification of RF transmitters using such features is called RF fingerprinting. Features are known to arise from many sources, including tiny differences in device fabrication process and electronic components [1].

RF fingerprinting has attracted significant attention [2]–[6]: In [2], RF-DNA features which contain information on variance, skewness, and kurtosis, within a preamble response were used with an ensemble method. In [3], four features—differential constellation trace figure, carrier frequency offset, modulation offset, and I/Q offset where classifications were done by calculating the minimum distance between test data and training data—were used. In [4], the mean of the instantaneous amplitude of the received signal and the modulation symbol were used with an optimal dimension-reduced matrix. In [5], an RF fingerprinting scheme based on low-rank representation of the original data with the robust classifier parameter was investigated. In [6], a convolutional neural network (CNN) for seven commercial Zigbee devices was used. They

collected 1,000 data per class. In [2]–[6], RF fingerprinting schemes with multiple features, which exhibited a high accuracy rate, were proposed for Zigbee devices and satellite terminals.

The contributions and novelties of this letter are as follows:

- We propose a new RF fingerprinting algorithm and a set of three RF features—*rising transient*, *falling transient*, and *sync*—and show the possibility that each feature can provide unique information through a real-life experiment. The falling transient feature has never been used in RF fingerprinting studies. Our results indicate that the performance of RF fingerprinting improves as each feature is additionally employed.
- Even though SRC is a common algorithm in classification [7], no study on RF fingerprinting with a combination of SRC and multiple features has been reported. We show that SRC can be a good platform for RF fingerprinting.

The remainder of this letter is organized as follows. The experimental system is described in Section II. The proposed method is outlined in Section III. Results are presented and analyzed in Section IV. The conclusion is given in Section V.

II. EXPERIMENTAL SYSTEM

A. Walkie-Talkie Signals

Our RF signals follow the digital mobile radio (DMR) standard. The DMR standard follows time-division multiple access (TDMA) and 4-level frequency-shift keying modulation [8]. A signal burst appears for 30ms and disappears for 30ms using the 2-slot TDMA method. This pattern is repeated in transmission.

The signal burst consists of rising transient, falling transient, and steady-state signals. The rising transient signal grows from zero to the designed level of the RF signal. Contrary to the rising transient signal, the falling transient signal decreases from the designed level to zero. The steady-state signal refers to the resting part between the rising and the falling transient signal and is composed of data and a sync signal. The data have 216 bits and the sync signal has 48 bits. The bit rate of the DMR standard is 9,600 bits/s. The sync signal is used to synchronize the transmitter and receiver.

From the pertinent signal part, a feature is obtained. Each feature is the main lobe of a spectrum of the pertinent signal part. We show how to extract each feature from the pertinent signal part in Section III. A. As we mentioned, the features arise from the inherent nonlinear properties of radio transmitters in the manufacturing process [1]. Owing to the presence of such features, RF fingerprinting can be accomplished.

Manuscript received December 24, 2018; revised January 31, 2019; accepted February 28, 2019. Date of publication March 20, 2019; date of current version May 8, 2019. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (NRF-2018R1A2A1A19018665). The associate editor coordinating the review of this letter and approving it for publication was F. Wang. (*Corresponding author: Heung-No Lee.*)

The authors are with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea (e-mail: heungno@gist.ac.kr).

Digital Object Identifier 10.1109/LCOMM.2019.2905205

B. Signal-Acquisition Setup

For our experiment, two walkie-talkie models were used: Motorola *SIIM* and Hytera *BD-358*. Each model follows the DMR standard. Four units of each type, i.e., eight walkie-talkies in total, were used in the experiment.

Signal was transmitted from the transmitter and acquired from an SMA male mini car-mounted antenna, the receiving frequency band of which is 400–470 MHz. We then down-converted 423.1875 MHz to 10 MHz using an XL-11-411 RF mixer and an E4438C ESG vector signal generator. Then, we filtered the signal bandwidth and sampled the signal using an IF recording system with the PX14400 operator functioning as a low-pass filter and analog-to-digital converter. Signals sampled at 96 MHz were saved to a computer and loaded to MATLAB. As we captured 50 signals per walkie-talkie, 400 signals were saved to the computer.

III. PROPOSED FEATURE EXTRACTION AND CLASSIFICATION

A. Signal Burst to Features

To cultivate features, we extracted the three signal parts from a single signal burst. Then, each feature was selected from the pertinent extracted signal parts.

Each signal part is extracted from the first signal burst of the total received signal through time-windowing. To design the time-window for each signal, we used a thresholding method. For a rising transient signal $\mathbf{f}_R \in \mathbb{R}^{500,000 \times 1}$, the starting threshold is the first time point at which the amplitude of the signal burst exceeds 10% of its maximum; the ending threshold is the time point it exceeds 90%. Similarly, for the falling transient signal $\mathbf{f}_F \in \mathbb{R}^{500,000 \times 1}$, the starting threshold and the ending threshold are the latest time points at which the amplitude of the signal burst exceeds 90% and 10% of its maximum, respectively. Since the length of each transient signal fluctuates, we used zero padding method after the ending point of each transient signal to match the length. To design the time window for the sync signal, we referred to the DMR standard [8]. The sync signal $\mathbf{f}_S \in \mathbb{R}^{480,000 \times 1}$ is located at the center of a signal burst. We set the center of the time-window for the sync signal to the central time point between the ending time of the rising transient signal and the starting time of the falling transient signal. The width of the time window was set to 0.005 s, as per the DMR standard [8]. Each extracted signal part was normalized to consider the case of signals with different power levels.

The extracted signal parts were transformed to the spectrum domain by fast Fourier transform (FFT) with the size of the time signal. Then, the operation of taking the absolute value of each element was executed to compare the energy and frequency information of the extracted signal part with those of the others. Since the main lobe occupies most of the energy of each signal part, the main lobe was taken from each spectrum. The main lobes, $F(\mathbf{f}_R)_{ML} \in \mathbb{R}^{2,000 \times 1}$, $F(\mathbf{f}_F)_{ML} \in \mathbb{R}^{2,000 \times 1}$, and $F(\mathbf{f}_S)_{ML} \in \mathbb{R}^{1,920 \times 1}$, are the unique features used in our experiment, where $F(\cdot)$ is the FFT operation function and ML denotes the main lobe of the spectrum.

To extract the main lobe, we used a bandpass filter first. Then, the center frequency of the filtered spectrum was down-converted to zero. Finally, we decimated the signal to reduce the length of the sample sequence. The main lobe was set to occupy most of the energy of each signal part, considering the channel bandwidth.

B. Proposed SRC

SRC is a classification algorithm based on the compressed sensing theory [7] and is used to determine the class from the sparse solution of the representation equation

$$\mathbf{y} = \mathbf{D}\mathbf{s}, \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^{P \times 1}$ is a test data vector with length P , $\mathbf{D} \in \mathbb{R}^{P \times NL}$ is a training data matrix composed of N training data vectors for each class label of transmitters $l \in \{1, \dots, L\}$, and $\mathbf{s} \in \mathbb{R}^{NL \times 1}$ is a vector of sparse representation coefficients. The sparse signal recovery algorithm in [7] was used to solve (1) with $P < NL$. In (1), $\mathbf{D}\mathbf{s}$ can be rewritten as

$$\mathbf{D}\mathbf{s} = [\mathbf{D}^{(1)} \quad \dots \quad \mathbf{D}^{(L)}] \left[(\mathbf{s}^{(1)})^T \quad \dots \quad (\mathbf{s}^{(L)})^T \right]^T, \quad (2)$$

where $\mathbf{D}^{(l)}$ is a submatrix of \mathbf{D} corresponding to the l^{th} class label of transmitters and $\mathbf{s}^{(l)}$ is a subvector of \mathbf{s} corresponding to the l^{th} class label of transmitters, and T denotes the transpose. To identify the class of test data, we solve

$$\text{class} = \arg \min_{l \in \{1, \dots, L\}} \left\| \mathbf{y} - \mathbf{D}^{(l)} \mathbf{s}^{(l)} \right\|_2. \quad (3)$$

If the column vectors in \mathbf{D} are less correlated, the solution \mathbf{s} of (1) is approximated to be sparse since the condition of having a sparse solution depends on the mutual correlation between the columns of \mathbf{D} [9]. Thus, the compressed sensing algorithms in [7] can be used to find a unique solution \mathbf{s} . However, when RF signals are taken directly to form the column vectors of \mathbf{D} , the solution \mathbf{s} cannot be sparse because they may be highly correlated. Then, the performance of SRC may be poor. Thus, RF signals must be processed to remove correlation to obtain high performance in SRC [9].

To remove correlation among RF signals, the proposed method applies principal components analysis (PCA) to the column vectors, each of which combines multiple features. PCA is known to be good at geometrically separating the features in the Euclidean domain by removing the mutual correlation [10]. This section aims to show how three kinds of features are concatenated and how PCA is applied to the features. We first introduce the single modal method and then discuss the proposed multimodal method.

1) *Single Modal RF Fingerprinting*: Consider that one of the rising transient, falling transient, and sync features is used as the sole representative feature of a single transmitter. We first form a feature matrix in which the columns are the sample vectors of the feature of candidate RF transmitters. Mathematically, for L RF transmitters (classes) and N sample vectors of a feature of each RF transmitter, we construct the feature matrix $\mathbf{A} \in \mathbb{R}^{M \times NL}$ as follows:

$$\mathbf{A} = \left[\mathbf{a}_1^{(1)}, \dots, \mathbf{a}_N^{(1)}, \mathbf{a}_1^{(2)}, \dots, \mathbf{a}_N^{(L)} \right], \quad (4)$$

where column vector $\mathbf{a}_n^{(l)} \in \mathbb{R}^{M \times 1}$ is the n^{th} sample vector of a feature of the l^{th} RF transmitter for $n = 1, \dots, N$ and $l = 1, \dots, L$, and M is the length of a feature. We denote a feature of an unknown RF transmitter as $\mathbf{u} \in \mathbb{R}^{M \times 1}$. From the PCA operation, (7) and (9), \mathbf{A} and \mathbf{u} are changed to a training data matrix \mathbf{D} and a test data vector \mathbf{y} , respectively.

2) *Multimodal RF Fingerprinting*: The proposed method is to concatenate the multiple features in the representation equation $\mathbf{u} = \mathbf{A}\mathbf{s}$, as shown in Fig. 1; the feature matrices are concatenated in a row-wise manner. Let us denote the n^{th} sample vector of the k^{th} feature of the l^{th} RF transmitter for $n = 1, \dots, N$, $l = 1, \dots, L$, and $k = 1, \dots, K$ as $\mathbf{a}_{k,n}^{(l)} \in \mathbb{R}^{M \times 1}$, where K is the number of features to be combined. The feature matrices are concatenated as follows:

$$\mathbf{A} = [\mathbf{A}_1^T \quad \mathbf{A}_2^T \quad \dots \quad \mathbf{A}_K^T]^T \in \mathbb{R}^{MK \times NL}, \quad (5)$$

where $\mathbf{a}_{k,n}^{(l)}$ forms the columns of feature matrix \mathbf{A}_k ,

$$\mathbf{A}_k = [\mathbf{a}_{k,1}^{(1)}, \dots, \mathbf{a}_{k,N}^{(1)}, \mathbf{a}_{k,1}^{(2)}, \dots, \mathbf{a}_{k,N}^{(L)}] \in \mathbb{R}^{M \times NL}. \quad (6)$$

Then, PCA is applied to the feature matrix \mathbf{A} . We obtain the training data matrix \mathbf{D} as follows:

$$\mathbf{D} = \mathbf{V}^T(\mathbf{A} - \mathbf{m}\mathbf{1}) \in \mathbb{R}^{P \times NL}, \quad (7)$$

where $\mathbf{m} = \frac{1}{L \times N} \sum_{l=1}^L \sum_{n=1}^N \mathbf{a}_n^{(l)} \in \mathbb{R}^{MK \times 1}$ is the average vector of the columns of \mathbf{A} , $\mathbf{1} := [1 \ 1 \ \dots \ 1]$ is the 1 by NL vector of 1s, $\mathbf{a}_n^l = [(\mathbf{a}_{1,n}^{(1)})^T (\mathbf{a}_{2,n}^{(1)})^T \dots (\mathbf{a}_{K,n}^{(L)})^T]^T \in \mathbb{R}^{MK \times 1}$ is a column vector which combines K features, and $\mathbf{V} \in \mathbb{R}^{MK \times P}$ is a rearranged eigenvector matrix of the covariance matrix $(\mathbf{A} - \mathbf{m}\mathbf{1})(\mathbf{A} - \mathbf{m}\mathbf{1})^T \in \mathbb{R}^{MK \times MK}$. The eigenvectors of $(\mathbf{A} - \mathbf{m}\mathbf{1})(\mathbf{A} - \mathbf{m}\mathbf{1})^T$ are arranged according to the eigenvalues in descending order. Since the eigenvalue of the covariance matrix is proportional to the variance of the columns of \mathbf{A} and the eigenvectors of the covariance matrices are orthonormal, the column vector of \mathbf{V} becomes a basis of the new space on the variance of the columns of \mathbf{A} [10]. The dimension of \mathbf{V} can be selected by the user as $P \in \{1, \dots, MK\}$. To obtain the test data vector \mathbf{y} of SRC, we first concatenate different features of an unknown transmitter $\mathbf{u}_k \in \mathbb{R}^{M \times 1}$ as follows:

$$\mathbf{u} = [\mathbf{u}_1^T \quad \mathbf{u}_2^T \quad \dots \quad \mathbf{u}_K^T]^T \in \mathbb{R}^{MK \times 1}. \quad (8)$$

Finally, PCA is applied to a concatenated feature of unknown transmitter \mathbf{u} . The test data vector \mathbf{y} is obtained by mapping the difference between concatenated feature \mathbf{u} and \mathbf{m} , i.e., $\mathbf{u} - \mathbf{m}$, onto the space with the eigenvector matrix \mathbf{V} ,

$$\mathbf{y} = \mathbf{V}^T(\mathbf{u} - \mathbf{m}) \in \mathbb{R}^{P \times 1}. \quad (9)$$

By using PCA, the equation in Fig. 1 is changed to (1), which has principal components as training and test data. The SRC solution in (1) was determined using the basis pursuit algorithm, which finds the unique sparse solution that has the minimum L1 norm [11].

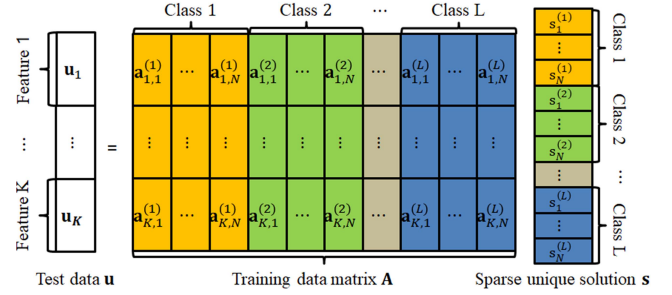


Fig. 1. Feature concatenation in the proposed method.

TABLE I
ACCURACY RATE OF THE PROPOSED METHOD

	4 BD-358	4 SLIM	4 BD-358 4 SLIM
Accuracy rate (Minimum number of PC)			
TR(R)	88% (24)	82% (48)	90.5% (45)
TR(F)	87.5% (45)	90% (12)	92.25% (13)
TR(R + F)	93% (49)	92% (20)	95.5% (63)
Sync	99% (45)	83.5% (22)	93.75% (86)
TR(R + F) + Sync	99% (44)	98.5% (22)	98.75% (21)

R: Rising, F: Falling, PC: Principal components

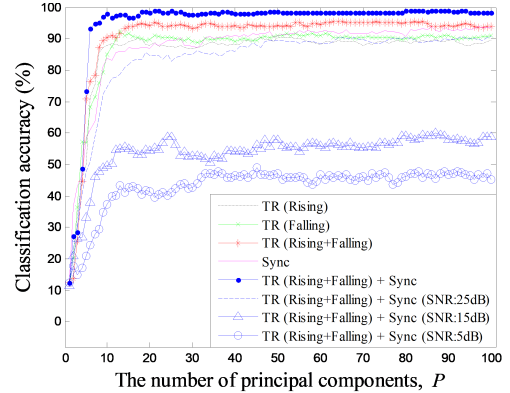


Fig. 2. Classification result of four BD-358 and four SLIM walkie-talkies.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

For our experiment, we set the decimation rate for all feature extractions to 250, considering the bandwidth of the RF signal following the DMR standard [8] and the sampling rate. To evaluate the performance of the proposed classifier, we used a five-fold cross validation technique. Fifty data were used per walkie-talkie, such that each test data was classified on the total of 320 training data. The experiment was performed in a line-of-sight environment and SNR was around 35-40 dB.

Table I shows the accuracy rate of the proposed method. The accuracy rate of the multimodal scheme is much better than that obtained from using only one feature. The minimum number of principal components is the minimum number of column vectors in the eigenvector matrix \mathbf{V} that yields the highest accuracy rate. Fig. 2 shows eight results of classification using 1 to 100 principal components, 1) on the rising transient feature, 2) on the falling transient feature, 3) on both transient features combined, 4) on the sync feature, and

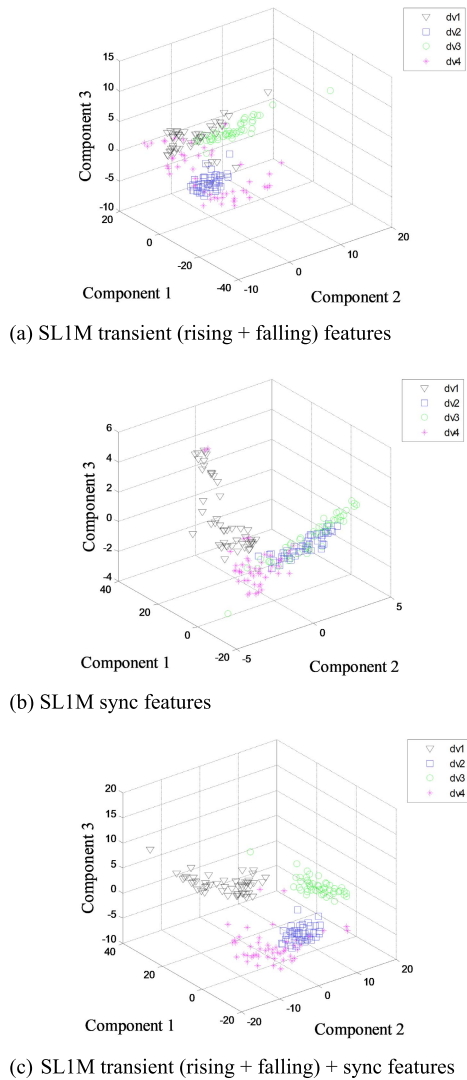


Fig. 3. Feature map of a 3-D principal components space.

5) on all features combined in SNR : 35-40dB, 6) 25dB, 7) 15dB, and 8) 5dB. Improved performance with an increased number of features indicates that each feature could contain unique information. Since the eigenvectors of the 21st and higher eigenvalues of $(\mathbf{A} - \mathbf{m}\mathbf{1})(\mathbf{A} - \mathbf{m}\mathbf{1})^T$ do not have enough information to represent the differences of data, the classification accuracy is not changed as $P > 21$. Fig. 3 shows a feature map in which the principal components of features of Motorola *SLIM* are mapped onto a 3D plot. The label of each axis, such as Component 1 and Component 2, means the projection of $\mathbf{u} - \mathbf{m}$ to the P^{th} column vector of \mathbf{V} . The figure shows distinct cluster formation when a concatenated feature is used.

To compare the proposed method with convolutional neural network (CNN), we referred to the studies [6] and [12]. For additional tests, we used a five-fold cross validation technique. Five training neural networks were constructed

using the same training dataset with the proposed method. We used the concatenated features as input data. The average classification accuracy rate of the CNN was 91.5%.

In our experiment, the size of the training data set is much smaller than that of [6]. Comparing CNN (91.5%) with the proposed method (98.75%), we show that SRC performs better than CNN for this small training data set. Because it is simple to add more training data and different kinds of features in SRC, the performance of the proposed method could be improved with additional training data and other kinds of features.

V. CONCLUSIONS

This letter proposed a multimodal RF fingerprinting scheme based on SRC. We showed that the proposed multimodal scheme, which concatenates multiple features in the row-wise manner and applies PCA to the concatenated dictionary matrix, improves accuracy significantly. We showed the possibility that the three signal features we have cultivated from RF signal samples could provide mutually independent information. The proposed scheme is efficient in the sense that improved RF fingerprinting accuracy is obtained. In addition, it is simple and easy in the proposed scheme to add more data and various kinds of features. The MATLAB source code for this study can be obtained at https://infonet.gist.ac.kr/?page_id=14.

REFERENCES

- [1] Y. Jia, S. Zhu, and L. Gan, "Specific emitter identification based on the natural measure," *Entropy*, vol. 19, no. 3, p. 117, 2017.
- [2] H. J. Patel, M. A. Temple, and R. O. Baldwin, "Improving ZigBee device network authentication using ensemble decision tree classifiers with radio frequency distinct native attribute fingerprinting," *IEEE Trans. Rel.*, vol. 64, no. 1, pp. 221–233, Mar. 2015.
- [3] L. Peng, A. Hu, J. Zhang, Y. Jiang, J. Yu, and Y. Yan, "Design of a hybrid RF fingerprint extraction and device classification scheme," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 349–360, Feb. 2019.
- [4] Y. Jia, J. Ma, and L. Gan, "Combined optimization of feature reduction and classification for radiometric identification," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 584–588, May 2017.
- [5] Y. Jia, J. Ma, and L. Gan, "Radiometric identification based on low-rank representation and minimum prediction error regularization," *IEEE Commun. Lett.*, vol. 21, no. 8, pp. 1847–1850, Aug. 2017.
- [6] K. Merchant, S. Revay, G. Stantchev, and B. Noursain, "Deep learning for RF device fingerprinting in cognitive communication network," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 160–167, Feb. 2018.
- [7] J. Wright, A. Y. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
- [8] *Electromagnetic Compatibility and Radio Spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 1: DMR Air Interface (AI) Protocol*, Standard ETSI TS 102 361-1, European Telecommun. Standards Inst., 2016.
- [9] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, Feb. 2009.
- [10] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdiscipl. Rev. Comput. Statist.*, vol. 2, no. 4, pp. 433–459, 2010.
- [11] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis Pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.
- [12] L. Ding, S. Wang, F. Wang, and W. Zhang, "Specific emitter identification via convolutional neural networks," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2591–2594, Dec. 2018.

MLNet: Metaheuristics-Based Lightweight Deep Learning Network for Cervical Cancer Diagnosis

Manjit Kaur , Senior Member, IEEE, Dilbag Singh , Senior Member, IEEE, Vijay Kumar , and Heung-No Lee , Senior Member, IEEE

Abstract—One of the leading causes of cancer-related deaths among women is cervical cancer. Early diagnosis and treatment can minimize the complications of this cancer. Recently, researchers have designed and implemented many deep learning-based automated cervical cancer diagnosis models. However, the majority of these models suffer from over-fitting, parameter tuning, and gradient vanishing problems. To overcome these problems, in this paper a metaheuristics-based lightweight deep learning network (MLNet) is proposed. Initially, the hyper-parameters tuning problem of convolutional neural network (CNN) is defined as a multi-objective problem. Particle swarm optimization (PSO) is used to optimally define the CNN architecture. Thereafter, Dynamically hybrid niching differential evolution (DHDE) is utilized to optimize the hyper-parameters of CNN layers. Each particle of PSO and solution of DHDE together represent the possible CNN configuration. F-score is used as a fitness function. The proposed MLNet is trained and validated on three benchmark cervical cancer datasets. On the Herlev dataset, MLNet outperforms the existing models in terms of accuracy, f-measure, sensitivity, specificity, and precision by 1.6254%, 1.5178%, 1.5780%, 1.7145%, and 1.4890%, respectively. Also, on the SIPaKMeD dataset, MLNet achieves better performance than the existing models in terms of accuracy, f-measure, sensitivity, specificity, and precision by 2.1250%, 2.2455%, 1.9074%, 1.9258%, and 1.8975%, respectively. Finally, on the Mendeley LBC dataset, MLNet achieves better performance than the competitive models in terms of accuracy, f-measure, sensitivity, specificity, and precision by 1.4680%, 1.5845%, 1.3582%, 1.3926%, and 1.4125%, respectively.

Index Terms—Cervical cancer, deep learning, diagnosis, dynamically hybrid niching differential evolution,

Manuscript received 14 June 2022; revised 29 September 2022; accepted 31 October 2022. Date of publication 18 November 2022; date of current version 5 October 2023. This work was supported in part by the MSIT (Ministry of Science and ICT), Korea, through the ITRC (Information Technology Research Center) Support Program IITP-2021-2021-0-01835, supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation), and in part by the National Research Foundation of Korea (NRF), funded by the Korean government (MSIP) under Grant NRF-2021R1A2B5B03002118. (Corresponding author: Heung-No Lee.)

Manjit Kaur, Dilbag Singh, and Heung-No Lee are with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea (e-mail: manjit.kr@yahoo.com; dggill2@gmail.com; heungno@gist.ac.kr).

Vijay Kumar is with the Computer Science and Engineering Department, NIT Hamirpur, Hamirpur, Himachal Pradesh 177005, India (e-mail: vijaykumarchahar@gmail.com).

Digital Object Identifier 10.1109/JBHI.2022.3223127

lightweight deep learning network, metaheuristics, particle swarm optimization.

I. INTRODUCTION

CERVICAL cancer is found in the lower part of the uterus. The cell of cervix progressively grows in pre-cancerous cells and cultivates cancer. Cervical cancer is the fourth most life-threatening disease in females [1]. It is responsible for a 90% mortality rate in developing countries due to its unawareness of the disease and its impact on life [2]. The early detection of cervical cancer is done by using Pap testing [3]. In this test, the samples are taken from the cervix and examined by the pathologist for any abnormalities present in cervix [4]. Manual testing requires expert pathologists for abnormalities analysis. However, this process is time-consuming and error-prone. To handle these issues, automatic cell classification techniques are required.

The well-known automated analysis systems are BD Focal Point slide profiler [5] and ThinPrep [6]. These automated systems were approved by the Food and Drug Administration (FDA) of the US. These systems not only reduce the analysis time but also improve the classification performance. In recent years, artificial intelligence techniques are widely used for automatic cervical cell classification [7]. Most of the existing techniques utilize the concept of feature extraction to enhance the performance of the classifier and reduce the computational run time. Some of these techniques are support vector machine (SVM), K-nearest neighbor (KNN), and artificial neural networks (ANN). Machine learning techniques provide better results for small datasets. However, the medical dataset is large in size and complex in nature. Sometimes, the accurate classification through machine learning techniques is a little bit difficult due to the variations in shape and size of cells. The different feature extraction techniques provide different results. Hence, there is a need for an automatic feature extraction technique. For this, deep learning techniques are widely used.

Deep learning techniques are usually applied in several areas of medical imaging [8]. Among the existing deep learning techniques, convolutional neural networks (CNNs) are extensively used to classify cervical cells [9]. These techniques eliminate the need for pre-processing steps such as feature extraction. These techniques directly utilize the original image and reduce the

control parameter of the training phase. The main aspects of deep learning techniques are pooling and weight sharing. Some of the well-known techniques are long-short term memory (LSTM), recurrent neural network (RNN), and generative adversarial networks (GAN). The performance of these techniques is still far from the optimal results. It is found that the existing models suffer from over-fitting, parameter tuning, and gradient vanishing problems. Model parameters are generally selected through trial-and-error, resulting in poorer performance. This motivates us to develop a novel technique for cervical cell classification. The main contributions of this paper are as follows:

- 1) A metaheuristics-based lightweight deep learning network (MLNet) is proposed.
- 2) The hyper-parameters tuning problem of convolutional neural network (CNN) is defined as a multi-objective problem.
- 3) Particle swarm optimization (PSO) is used to optimally define CNN architecture. Thereafter, Dynamically hybrid niching differential evolution (DHDE) is utilized to optimize the hyper-parameters of CNN layers. Each particle of PSO and solution of DHDE together represent the possible CNN configuration.

The remaining structure of this paper is as follows. Section II presents the related work done in the field of cervical cell classification. The utilized approaches are discussed in Section III. The proposed model is described in Section IV. Experimental results and discussion are mentioned in Section V followed by the conclusion in Section VI.

II. RELATED WORK

Lin et al. [10] extended CNN-based image identification algorithm. Smooth L1 loss and regression methods were combined to enhance the classification accuracy. Cycle GAN was used due to its ease of implementation. They attained the classification accuracies of 98.1% and 97.6% for binary and seven class problems of the Herlev dataset, respectively. However, the classification accuracy of some samples is not good. Chen et al. [11] developed a lightweight CNN (LCNN) for the classification of cervical cells. The compression method was used to improve the performance of lightweight CNN. VGG, ResNet, and Inception-ResnetV2 were used to design the teacher model and determined labels from the designed model. Thereafter, Xception, MobileNet, MobileNetV2, and DenseNet were used to design the student model. The accuracy obtained from Inception-ResnetV2 was 73.58%. However, it suffers from a hyper-parameter tuning problem.

Sabeena et al. [12] utilized a pyramid scene parsing model (PSPM) for cell recognition from cervical images. The local and global priors were integrated with PSPM. The average classification accuracy obtained from their approach was 99.7%. However, the robustness of this approach has yet to be validated on different datasets. Allehaibi et al. [9] used a Mask regional CNN (Mask R-CNN) for the segmentation of cervical cells. Thereafter, a pretrained VGG model was used for classification. The accuracy obtained from Mask R-CNN was 96% for two-class classification. However, this approach requires high

computational complexity. Zhao et al. [13] utilized dense U-Net (DU-Net) to design the deep learning framework for the segmentation of nuclei from cervical images. The ensemble strategy was used to eliminate the predictive bias during the training phase. Their method achieved precision and recall of 0.946 and 0.984, respectively.

Hussain et al. [14] proposed a fully CNN (FCNN) for the classification of cervical nuclei from the images. It attained an accuracy of 96%. However, it suffers from high computational time. Shi et al. [15] developed a cervical cell classification model using graph CNN (GCNN). They used intrinsic features of an image to capture the correlation among clusters. The average accuracy obtained from this method was 94.23%. This model suffers from annotation costs. Li et al. [16] integrated global information with an attention mechanism to classify the cervical nuclei. The features were extracted through ResNet50. These features were applied to the attention mechanism to find out the cell region. Long-short term memory (LSTM) was used to aggregate the cell features. The average classification accuracy obtained from this model was 98.89%.

Rahaman et al. [17] designed a novel deep learning framework named as DeepCervix for the classification of cervical cells. DeepCervix utilized VGG16, VGG19, XceptionNet, and ResNet50 for feature extraction. Dropout layer and batch normalization were incorporated in DeepCervix. DeepCervix was evaluated on the Herlev dataset and attained the accuracy of 98.91% for two-class and 90.32% for seven classes. However, the poison noise greatly degrades the performance of the model. Dong et al. [18] used InceptionV3 for cervical cell classification. The performance of InceptionV3 was evaluated on the Herlev dataset and attained an accuracy of 98.23%. The artificial features may be incorporated in InceptionV3 for better performance. Chankong et al. [19] designed an automatic cervical image classification. Fuzzy C-means (FCM) were utilized to extract the features. These features were applied on Bayesian, linear discriminant analysis (LDA), K-nearest neighbor (KNN), artificial neural network (ANN), and support vector machine (SVM). For the Herlev dataset, this model obtained the 93.78% and 99.27% accuracies for seven-class and binary-class, respectively. This method was not validated for multiple cell classification.

Wang et al. [20] developed an automatic cervical image classification. Mean-shift clustering was used to determine the regions of interest. The overlapping regions were spitted through morphological operations. The feature set was optimized through a chain-like agent genetic algorithm (CAGA). Their method outperformed the other methods in terms of specificity, sensitivity, and accuracy. The robustness of this method was not tested on different types of datasets. Martin et al. [21] used CNN for cervical cell classification. The average accuracy attained from their model was 60%. Yaman and Tuncer [22] utilized the exemplar pyramid deep feature extraction (EPDFE) technique to detect cervical cancer. Two pre-trained transfer learning models named DarkNet19 and DarkNet53 were used for extracting deep features from cervical images. Neighborhood component analysis was used to select the features obtained from pre-trained models. The selected deep features were applied to SVM for

classification. The average accuracy obtained from this approach was 96%. But this approach did not get better results for the SIPaKMeD dataset.

Manna et al. [23] designed an ensemble CNN model using fuzzy rank (CNN-F) for cervical image classification. The fuzzy rank-based fusion of three pre-trained models such as InceptionV3, Xception, and DenseNet169 was incorporated in this model. It performed better than the other models in terms of accuracy. This model was unable to classify the overlapping and poor contrast images. Chen et al. [24] developed an automatic cervical cell classification. Compact VGG was used for developing a cell classifier. Their method performs better than the other models in terms of classification measures. Tripathi et al. [25] studied four deep learning models namely ResNet50, ResNet152, VGG16, and VGG19 for cell classification. The best accuracy obtained from ResNet152 was 94.89%. However, the parallel implementation of these models is required to improve the classification performance. Basak et al. [26] developed an automatic deep learning framework for cytology cell classification. CNN model was used to extract the deep features. These features were further reduced through principal component analysis (PCA). The reduced features were optimized by a grey wolf optimization algorithm. Thereafter, these optimized features were applied to SVM for classification. The classification accuracies obtained by this method were 98.32% and 97.87% for seven-class of Herlev and five-class of SIPaKMeD, respectively. Kumar and Krishna [27] classified cervical cancer by varying the channels of CNN (c-CNN). c-CNN with channels (32,64) achieved 96.89% accuracy, 94.15% F-score, 93.75% sensitivity, and 93.38% precision.

Martinez-Mas et al. [28] developed a CNN for PAP-smear cell classification. The cell merger approach was used in the developed model. The accuracy obtained from their model was 88.8%. This model can be further improved by merging networks. Pal et al. [29] developed a deep multiple instance learning (DMIL) for cell classification. Deep CNN was used to extract deep features from cervical cell images. The average classification accuracy obtained from this method was 84.55%. Ghoneim et al. [30] developed a cervical cell classification model using CNN and an extreme learning machine (ELM) referred to as CNN-ELM. The deep features were extracted using CNN. These features were applied to ELM for classification. The classification accuracy obtained by their method was 91.2% for the seven-class of Herlev dataset. The architectures of other models can be incorporated for further improvement. Jia et al. [31] developed a novel deep learning framework for cell classification. CNN was used to extract the deep features. Gabor and deep features were combined. These features were reduced through PCA. The reduced features were applied to SVM for classification. This method outperformed the other models in terms of accuracy, sensitivity, and specificity. Alyafeai and Ghouti [32] used YOLO algorithm for the extraction of a region of interest (ROI). Lightweight CNN was used to classify the cervical cells. The accuracy under the curve obtained from this approach was 0.82.

Zhang et al. [33] used the weight transfer learning approach for colposcopy image classification. The pre-trained DenseNet

model was used for classification. The feature map concatenation was used to eliminate the gradient problem. Their approach obtained an accuracy of 73.08% over 600 test images. Xiang et al. [34] developed an automatic cervical cell classification (ACCC). YOLOv3 was used to extract the ROIs. CNN model was used to classify the cervical cell. ACCC performed better than the existing methods in terms of performance measures. Zorkaffi et al. [35] developed a neural network based on hybrid multi-layered perceptron (HMLP) and GA for cervical cell identification. The average accuracy obtained from this model was 74.82%. Bhavani and Govardhan [36] designed a stacked ensemble technique for cervical cell classification. The synthetic minority over-sampling technique (SMOTE) was used for data balancing. Thereafter, KNN, SVM, Random forest, and logistic regression were used to build the model for classification. The developed model was evaluated on the UCI machine learning dataset and attained an accuracy of 91.2%. However, the computational time of the developed model is high.

It is observed that the performance of existing models is still far from the optimal results. The existing techniques suffer from over-fitting, parameter tuning, and gradient vanishing problems. Hence, there is a need to develop a metaheuristics-based lightweight deep learning network that can classify cervical cancer efficiently.

III. PRELIMINARIES

A. Convolutional Neural Network

CNN is a deep learning architecture that learns the key features automatically from images. It can capture temporal and spatial dependencies using filters in an image. It also reduces the images without losing key features to minimize the computational complexity and improve performance. It also requires few parameters in comparison to traditional neural networks. CNN has mainly three layers: convolution, pooling, and fully connected.

In the convolution layer, key features are extracted from images using convolution operation and activation functions. The convolution operation facilitates the layers to identify similar objects in several images with different regions. It contains some parameters such as the size of filter, number of filter, padding, and stride that need to be set before the training process. In the pooling layer, the spatial size of convolved features is reduced through dimensionality reduction. Thus, it minimizes the computational power requirement. Its main benefit is that it extracts dominant features that are positional and rotational invariant. The different types of pooling can be utilized such as max, average, stochastic, spatial pyramid, and def-pooling [37]. But max pooling is mostly used as it provides better robustness and faster convergence. This layer does not contain any learnable parameters but has padding, stride, and filter size parameters. Both convolution and pooling layers together form the i^{th} layer of CNN. Depending on the complexity of the images, the number of these layers can be increased. The output of the last pooling layer, i.e., a feature vector is converted into a one-dimensional vector and fed into a fully connected layer. Fully connected layers are also called densely connected layers because every

neuron in one layer is connected to every neuron of another layer. The final fully connected layer classifies the images.

B. Particle Swarm Optimization

PSO is a metaheuristic technique that is used to solve optimization problems. In PSO [38], each particle can represent as a solution. These particles contain position and velocity vectors and evolved through the problem space. The position of the particles is adjusted through personal- and global-best positions. In the personal best position, each particle keeps the track of its own best position that has been found so far. The global best position keeps the record of the best position which is founded among all particles. Particle i holds velocity vector $Vel_i = (vel_i^1, vel_i^2, \dots, vel_i^D)$ and position vector $Pos_i = (pos_i^1, pos_i^2, \dots, pos_i^D)$, where D is the number of dimensions. PSO finds the best solution by updating Vel_i and Pos_i through each iteration (it) as follows:

$$Vel_i(it+1) = \omega \cdot Vel_i(it) + c_1 r_1 (Best_{p_i}(it) - Pos_i(it)) + c_2 r_2 (Best_g(it) - Pos_i(it)) \quad (1)$$

$$Pos_i(it+1) = Pos_i(it) + Vel_i(it+1) \quad (2)$$

where $Best_{p_i}$ represents the personal best position of the particle i that has been searched so far. $Best_g$ denotes the best position that has been found by a whole swarm so far. r_1 and r_2 are the random numbers that are uniformly distributed in the range of [0,1]. The parameters such as cognitive coefficient (c_2), social coefficient (c_1), and inertia weight (ω) maintain the trade-off between exploitation and exploration. ω impacts the performance of PSO, so it should be calculated carefully. In this paper, sigmoid-like inertia weight, given by [39], is used to achieve a better trade-off and convergence. It is defined as:

$$\omega = \begin{cases} 0.9, & \text{if } it \leq \alpha M_{nt^1} \\ \frac{1}{ite^{(10it-2M_{nt^1})/M_{nt^1}}} + 4, & \text{otherwise} \end{cases} \quad (3)$$

where M_{nt^1} is the maximum number of iterations. α is the predefined constant.

C. Dynamically Hybrid Niching Differential Evolution

Neural network ensembles are a multimodal optimization problem (MMOP) because it has multiple local and/or global optima, rough search space, and a large number of hyper-tunning parameters. To find the optimal solutions and improve the convergence accuracy, a dynamically hybrid niching-based differential evolution (DHDE) [40] is utilized. The reason is that it provides a better trade-off between convergence and diversity. DHDE utilizes crowding and speciation niching techniques dynamically during the execution to achieve the same. In DHDE, diversity is improved using the information of inferior offspring and crowding-based differential evolution (CDE- I_{OA}). Inferior offsprings are those who are weaker than their parents. Instead of discarding these inferior offsprings, DHDE saves them into an archive (i.e., inferior offspring archive (I_{OA})) to explore additional optimal solutions. The convergence is promoted by detecting and removing similar individuals from the population using improved neighborhood speciation-based differential

TABLE I
THE HYPER-PARAMETERS OF CNN ARCHITECTURE AND THEIR RANGES

Hyper-parameter	Range
Number of convolution layers (nCL)	[2,8]
Number of pooling layers (nPL)	[2,8]
Number of fully connected layers (nFL)	[2,8]

TABLE II
THE HYPER-PARAMETERS OF CNN'S LAYER AND THEIR RANGES

Layer	Hyper-parameter	Range
Fully connected	Number of neurons (F_m)	[1,1024]
	Padding pixels (pp_p)	[0,1]
Pooling	Stride size (ss_p)	[1,5]
	Filter size (fs_p)	[1,13]
	Stride size (ss_c)	[1,5]
Convolution	Padding pixels (pp_c)	[0,1]
	Filter size (fs_c)	[1,13]
	Number of filters (nf_c)	[64]

evolution (INSDE). Optimal solution archive (O_{SA}) is used to save the obtained optimal solutions during the execution. The working of the DHDE algorithm is explained in Algorithm 2 (see Section IV). DHDE combines CDE- I_{OA} and INSDE to achieve better diversity and convergence. Firstly, CDE- I_{OA} is called and then INSDE is executed.

IV. PROPOSED MODEL

Although CNN extracts the features automatically from the images, it requires a number of hyper-parameters to be set before starting the training process. The value of these parameters plays a very important role to build an efficient CNN model. These parameters can be set by human expertise or using a trial-error basis. But it is a very lengthy and time-consuming process. Therefore, it is required to select the parameters optimally to build an optimized and efficient CNN model. Table I shows the hyper-parameters of CNN architecture with their ranges. The hyper-parameters of layers with their ranges are shown in Table II. Padding pixels of convolution and pooling layers are either set to 0 or 1. Here, 0 and 1 represent valid and the same paddings, respectively. Odd filter size is used in both convolution and pooling layers. The stride size is always taken smaller than the filter size.

Fig. 1 shows the proposed MLNet for cervical cancer diagnosis. The remaining section describes the optimization of CNN architecture and its hyper-parameters using PSO and DHDE. Inspired from [41], firstly, PSO is used to optimize the hyper-parameters of CNN architecture that are mentioned in Table I. Thereafter, DHDE is utilized to optimize the hyper-parameters of the layers which are mentioned in Table II. Each particle of PSO and solution of DHDE together represent the possible CNN configuration. Finally, the softmax layer that classifies cervical cancer. The fitness of particle and solution is tested using the obtained F-score. The hyper-parameters of CNN are optimized iteratively by PSO and DHDE and find a configuration with the best fitness value. Thereafter, the CNN model is configured using the optimal hyper-parameters and trained with many cervical

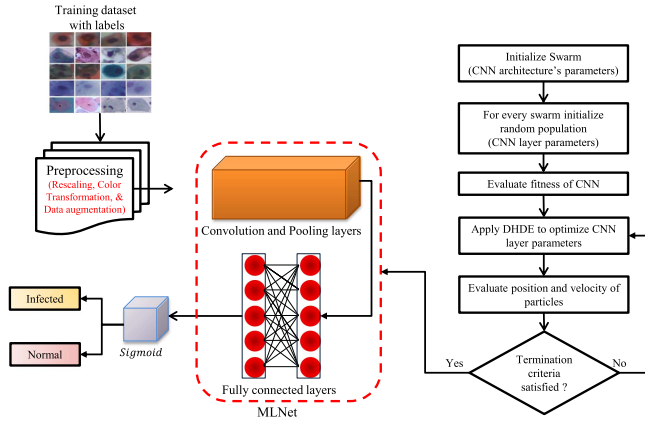


Fig. 1. MLNet: Metaheuristics-based lightweight deep learning network for cervical cancer diagnosis.

images. Finally, optimized CNN is applied for the classification of cervical cancer.

A. Optimization of CNN Architecture

Algorithm 1 describes the optimization of CNN architecture using PSO. A swarm $[SW_{p_1}, SW_{p_2}, \dots, SW_{p_m}]$ of m particles is initialized randomly with specified ranges (line 3). Each particle SW_{p_i} has three-dimensions such as nCL , nPL , and nFL (refer Table I). Initially, $Best_{p_i}$ and $Best_g$ are set to null (line 2). From lines 4 to 6, similar particles are removed if exist in the swarm. From lines 7 to 25, PSO-CNN algorithm iteratively optimizes the hyper-parameters of CNN until M_{nt}^1 is not reached. For every SW_{p_i} , optimal hyper-parameters such as nfc , fs_c , pp_c , ss_c , fs_p , ss_p , pp_p , and F_{nn} (refer Table II) are returned by Algorithm 2 (DHDE algorithm) (line 11). These parameters save in O_{SA} . In line 13, CNN model is configured using current SW_{p_i} and R_{OS} (randomly selected optimal solution from O_{SA}). The fitness of CNN is evaluated in line 14. Based on the fitness value, $Best_{p_i}$ and $Best_g$ are updated (lines 15-20). In line 21, the position and velocity of the particle are updated.

B. Optimization of Hyper-Parameters of CNN's Layers

Table II shows the hyper-parameters of fully connected, pooling, and convolution layers of CNN. These hyper-parameters are optimized using DHDE algorithm. The process of optimization is illustrated in Algorithm 2. This algorithm is called by Algorithm 1 for every particle (SW_{p_i}). Algorithm 2 generates a archive of optimal solution (nfc , fs_c , pp_c , ss_c , fs_p , ss_p , pp_p , and F_{nn}) for SW_{p_i} .

$SW_{p_i}\{nCL, nPL, nFL\}$, Population size S_P , crossover rate CR , scaling factor F , distance threshold d_t , minimal fitness threshold f_t , maximum counter M_C , maximum size of I_{OA} (M_{OA}), and maximum number of fitness evaluations ($M_{N_{fes}}$) are given as inputs to Algorithm 2. Initially, N_{fes} is set 0 and O_{SA} and I_{OA} are set to null. In line 3, a random population PO of S_P is generated. $PO = \{z_1, z_2, \dots, z_{S_P}\}$, where $z_i = \{nfc, fs_c, pp_c, ss_c, fs_p, ss_p, pp_p, F_{nn}\}$. In lines 4-5, CNN

Algorithm 1: PSO-CNN Algorithm.

Input : M_{nt}^1 and search space for hyperparameters

Output: CNN hyperparameters:

$nCL, nPL, nFL, nfc, fs_c, pp_c, ss_c, fs_p, ss_p, pp_p,$
and F_{nn}

1 **begin**

2 $Best_{p_i} = \phi, Best_g = \phi;$

3 Initialize the m particles ($SW_{p_i} = \{nCL, nPL, nFL\}$)
of swarm randomly in specified range;

/* Remove similar particles as:

$\forall i, j \in m \ \& \ i \neq j$ */

4 **if** $SW_{p_i} == SW_{p_j}$ **then**

5 $SW_{p_j} = NULL;$

6 **end**

7 **while** $g \leq M_{nt}^1$ **do**

8 Evaluate ω using Eq. (3);

9 **for** $i = 1$ to m **do**

10 **if** $SW_{p_i} \neq NULL$ **then**

11 $O_{SA} = \text{Algorithm 2}(nCL, nPL, nFL);$

12 Randomly select any optimal solution
from O_{SA} , i.e., $R_{OS} = O_{SA}(rand)$;

13 Setup a CNN model: $\text{CNN}(SW_{p_i}, R_{OS});$
14 Evaluate fitness (f) using CNN;

15 **if** $f(SW_{p_i}) > f(Best_{p_i})$ or $Best_{p_i} == \phi$
then

16 $Best_{p_i} \leftarrow SW_{p_i};$

17 **if** $f(Best_{p_i}) > f(Best_g)$ or

$Best_g == \phi$ **then**

18 $Best_g \leftarrow Best_{p_i};$

19 **end**

20 **end**

21 Update position (Pos_i) and velocity
(Vel_i) of particle using Eq. (2) and Eq.
(1), respectively;

22 **end**

23 **end**

24 **end**

25 **end**

model is set and fitness is calculated. N_{fes} is set equal to S_P . Now, DHDE calls CDE- I_{OA} and INSDE one by one until N_{fes} is less than $M_{N_{fes}}$. In line 8, Algorithm 3 CDE- I_{OA} is called by passing PO, I_{OA}, N_{fes} , and SW_{p_i} parameters.

In Algorithm 3, three individuals are selected from PO randomly to produce an offspring t_i for every target parent z_i using mutation and crossover operations of DE [40] (see lines 2-3). For mutation, "DE/rand/1" [40] is operation is used. The fitness of t_i is evaluated using CNN model (lines 4-5). In line 9, Euclidean distance is used to find the nearest parent q from PO to t_i . In lines 10-13, if fitness of t_i comes better than q , then t_i replaces q . Otherwise, it is added into I_{OA} . In Algorithm 2, I_{OA} is added into PO to explore additional optimal solutions (line 12).

When the size of I_{OA} ($|I_{OA}|$) becomes greater or equal to M_{OA} , Algorithm 4 INSDE is called by using $SW_{p_i}, PO, d_t,$

Algorithm 2: DHDE Algorithm.

Input : $SW_{p_i} \{nCL, nPL, nFL\}$, S_p , CR , F , d_t , M_C , f_t , M_{OA} , $M_{N_{fes}}$

Output: PO and O_{SA}

```

1 begin
2    $N_{fes} = 0$ ,  $O_{SA} = \phi$ ,  $I_{OA} = \phi$ ;
3   Generate random population  $PO$  with  $S_p$ 
   individuals;
4   Setup a CNN model:  $CNN(SW_{p_i}, z_i)$ ;
5   Evaluate  $f(z_i)$  using CNN;
6    $N_{fes} = N_{fes} + S_p$ ;
7   while  $N_{fes} < M_{N_{fes}}$  do
8     Call Algorithm 3 CDE- $I_{OA}$  ( $PO, I_{OA}, N_{fes}$ ,
       $SW_{p_i}$ );
9      $T = 1$ ;
10    while  $|I_{OA}| \geq M_{OA}$  &  $N_{fes} < M_{N_{fes}}$  do
11      if  $T == 1$  then
12         $PO = PO \cup I_{OA}$ ;
13         $T = 0$ ;
14      end
15      Call Algorithm 4 INSDE( $SW_{p_i}, PO, O_{SA}$ ,
       $d_t, f_t, M_C, N_{fes}$ );
16      if  $|PO| < S_p$  then
17         $I_{OA} = \phi$ ;
18        generate  $S_p - |PO|$  individuals
        randomly to fill  $PO$ ;
19      end
20    end
21  end
22 end

```

Algorithm 3: CDE- I_{OA} Algorithm.

Input : $PO, I_{OA}, N_{fes}, SW_{p_i}$

Output: PO and I_{OA}

```

1 for  $i = 1$  to  $S_p$  do
2   Select three individuals from  $PO$  randomly;
3   Apply mutation and crossover of DE to generate
   an offspring  $t_i$ ;
4   Setup a CNN model:  $CNN(SW_{p_i}, t_i)$ ;
5   Evaluate  $f(t_i)$  using CNN;
6    $N_{fes} = N_{fes} + 1$ ;
7 end
8 for  $i = 1$  to  $S_p$  do
9   Find the most similar individual  $q$  in  $PO$  to  $t_i$ ;
10  if  $f(t_i) > f(q)$  then
11     $q \leftarrow t_i$ ;
12  else
13     $I_{OA} = I_{OA} \cup t_i$ ;
14  end
15 end

```

Algorithm 4: INSIDE Algorithm.

Input : $SW_{p_i}, PO, O_{SA}, d_t, f_t, M_C, N_{fes}$

Output: PO and O_{SA}

```

1 Individual  $PO$ 's values are sorted from best to worst as
   $PO_S$ ;
2 for  $i = 1$  to  $\frac{S_p}{S_m}$  do
3   Form  $sub_{pop_i}$  by finding the similar  $S_m$  individuals
   of the first individual from  $PO_S$ ;
4   Remove the used individuals from  $PO_S$ ;
5 end
6 for  $i = 1$  to  $\frac{S_p}{S_m}$  do
7   for  $j = 1$  to  $|sub_{pop_i}|$  do
8     Use mutation and crossover of DE to obtain an
     offspring  $t_{i,j}$  with in  $sub_{pop_i}$ ;
9     Setup a CNN model:  $CNN(SW_{p_i}, t_{i,j})$ ;
10    Evaluate  $f(t_{i,j})$  using CNN;
11     $N_{fes} = N_{fes} + 1$ ;
12  end
13 end
14  $p = 1$ ;
15 for  $i = 1$  to  $\frac{S_p}{S_m}$  do
16   for  $j = 1$  to  $|sub_{pop_i}|$  do
17     if  $f(t_{i,j}) > f(q_{i,j})$  then
18        $C_p = 0$ ;
19        $q_{i,j} \leftarrow t_{i,j}$ ;
20     else
21        $C_p = C_p + 1$ ;
22     end
23      $p = p + 1$ ;
24   end
25 end
26 Merge every  $sub_{pop_i}$  into  $PO$ ;
27 for  $i = 1$  to  $S_p$  do
28   Find Euclidean distances of  $i$ th individual to other
   individuals in  $PO$ , and save them in  $E_d$ ;
29   for  $j = 1$  to  $S_p$  do
30     if  $i \neq j$  &  $E_d(j) < d_t$  &  $f(z_i) > f(z_j)$  &
       $f(z_i) - f(z_j) < f_t$  then
31       Eliminate the  $j$ th individual from  $PO$ ;
32     end
33   end
34 end
35 for  $i = 1$  to  $|PO|$  do
36   if  $C_i > M_C$  then
37     Call Algorithm 4 as Archive_upadte( $z_i, O_{SA}$ ,
      0.01);
38     Eliminate the  $i$ th individual from  $PO$ ;
39   end
40 end

```

f_t , M_C , and N_{fes} parameters (line 15). In Algorithm 4, PO is sorted in descending order based on the fitness value and it becomes PO_S (line 1). PO_S is divided into S_m different sub-populations (sub_{pop}) (lines 2-4). For each sub_{pop_i} , an offspring $t_{i,j}$ is generated using mutation and crossover of DE (lines 6-8). The fitness of $t_{i,j}$ is evaluated using CNN (lines 9-10). The selection operation is performed from 15 to 19. A counter C_p ($p = 1, \dots, S_p$) is assigned to every parent that records the

Algorithm 5: Archive_upadte Algorithm

```

Input :  $z_i, O, e$ 
Output:  $O$ 
1 if  $|O| = 0$  then
2    $O = O \cup z;$ 
3    $|O| = |O| + 1;$ 
4 else
5   Find the nearest solution  $z_n \in O$  to  $z$  in decision
   space;
6   if  $\|z - z_n\| < e$  then
7     if  $f(z) > f(z_n)$  then
8        $z_n \leftarrow z;$ 
9     end
10  else
11    if  $|O| < M_O$  then
12       $O = O \cup z;$ 
13       $|O| = |O| + 1;$ 
14    else
15      if  $f(z) > f(z_n)$  then
16         $z_n \leftarrow z;$ 
17      end
18    end
19  end
20 end

```

times of parent not replaced with offspring. Initially, C_p is 0. In line 26, all sub_{pop_i} are merged into PO . In lines 27-32, Euclidean distance is calculated of every i th individual in PO and store into E_d . Thereafter, similar individuals are identified and eliminated from PO based on E_d , d_t , and f_t . In line 36, if the counter of the individual becomes greater than M_C , then it will be identified as an optimal solution and updated in O_{SA} using Algorithm 5.

In Algorithm 4, $|O|$ represents the size of O_{SA} . In lines 1-3, if $|O|$ is empty, then optimal solution z is added into the O and size is incremented. The nearest solution z_n to z is found in O (line 5). In line 6, crowding distance is calculated among z and z_n , and if it comes to less than $e = 0.01$, then the fitness of both solutions is compared. If the distance is greater than e , the optimal solution is added to the archive O . M_O represents the maximum size of O , i.e., $M_O = 5 \times S_P$. After execution of Algorithm 4, Algorithm 2 checks the size of PO ($|PO|$) (line 16). If $|PO|$ is less than S_p , then I_{OA} is set to empty. Again, Algorithm 3 (CDE- I_{OA}) will be used.

V. PERFORMANCE ANALYSIS

The proposed MLNet is implemented on 11th Gen Intel(R) Core(TM) i5 – 11600 @ 2.80 GHz processor with 64 GB RAM. MATLAB 2021a software is used. The proposed model is trained and tested on three benchmark datasets such as Herlev [42], SIPaKMeD [43], and Mendeley LBC [44] datasets. For all datasets, 65% cell images are used for training, 15% and 20% cell images are used for model validation and testing,

TABLE III
PARAMETER SETTINGS OF PSO

Parameter	Value
Maximum number of iterations (M_{m1})	between 5 and 8
Cognitive coefficient (c_2)	2
Social coefficient (c_1)	2
Swarm size ($nPL \leq nCL, nFL \leq nCL$)	5

TABLE IV
PARAMETER SETTINGS OF DHDE

Parameter	Value
Population size (S_P)	80
Maximum size of I_{OA} (M_{OA})	$2S_P$
Minimal fitness threshold (f_t)	$1.0E-5$
Distance threshold (d_t)	0.1
Neighborhood size (S_m)	5
Crossover rate (CR)	0.3
Scaling Factor (F)	0.9
Maximum number of fitness evaluations ($M_{N_{fes}}$)	50,000

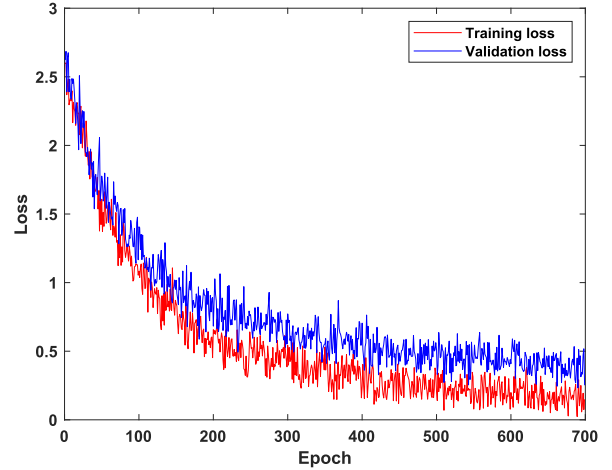


Fig. 2. Training and validation loss analyses of MLNet on Herlev dataset.

respectively. To build more generalized models, Generative adversarial network (GAN)-based data augmentation (refer [45]) is also used on the training dataset. PSO and DHDE parameter settings are shown in Tables III and IV, respectively. Parameters for PSO and DHDE are set based on the information provided in [41] and [40], respectively.

A. Comparative Analysis Based on Herlev Dataset

Fig. 2 shows the training and validation loss analyses of MLNet on the Herlev dataset. It is found that MLNet convergences at a good speed. Lesser difference between training and validation lines indicates that MLNet does not suffer from the overfitting problem. Overall the model achieves best training and validation accuracy as 99.92% and 99.67%, respectively.

A confusion matrix analysis of MLNet on the Herlev dataset is shown in Fig. 3. MLNet performs remarkably better on both normal and infected samples. Based on a better balancing of infected versus normal cases, MLNet yields better sensitivity

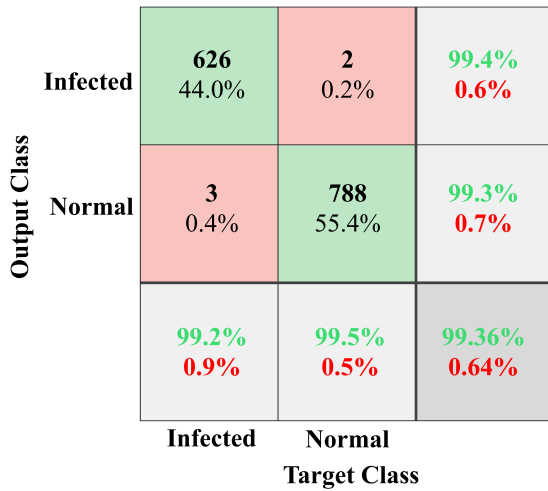


Fig. 3. Training and validation loss analyses of MLNet on Herlev dataset.

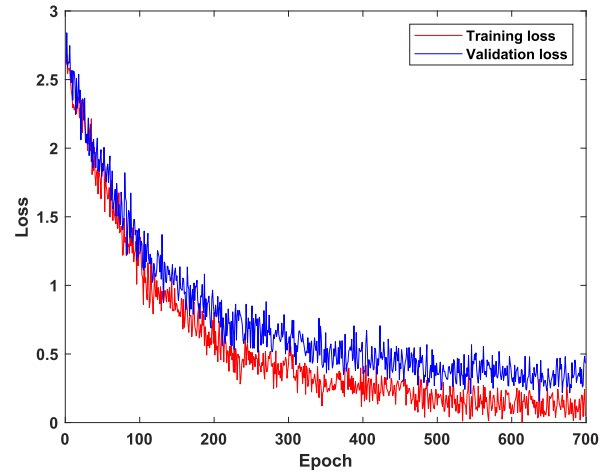


Fig. 4. Training and validation loss analyses of MLNet on SIPaKMeD dataset.

TABLE V
COMPARATIVE ANALYSIS BASED ON HERLEV DATASET

Models	Acc	Spec.	Sens./Rec	Pre	F1
R-CNN [9]	98.1	98.6	96.7	-	96.5
GAN [10]	98.1	98.6	96.7	-	-
LCNN [11]	73.58	-	-	-	-
PSPM [12]	99.7	99.3	99.8	-	-
DU-Net [13]	-	-	98.4	94.6	-
FCN [14]	98.8	-	95.8	96.1	-
DeepCervix [17]	98.91	-	98.0	99.25	98.5
InceptionV3 [18]	98.2	96.7	99.4	-	-
FCM [19]	99.27	96.53	99.85	-	-
VGG [24]	94.81	92.76	95.52	97.42	96.46
CNN [26]	98.32	-	97.65	98.66	98.12
CNN-ELM [30]	99.5	-	-	-	-
CNN-SVM [31]	99.3	99.4	98.9	-	-
MLNet	99.36	99.53	99.36	99.35	99.28

and specificity on the Herlev dataset. In total, MLNet achieves 99.36% accuracy.

Table V shows the comparative analysis among MLNet and the existing cervical cancer classification techniques on the Herlev dataset. It is found that each approach achieves remarkable results. PSPM [12] achieved remarkable accuracy (Acc) value as 99.7%. FCM [19] obtained better sensitivity (Sens./recall (Rec) value as 99.85%. It is found that the proposed MLNet achieves remarkable performance in terms of accuracy, Specificity (Spec.), Sens./Rec, precision (Pre) and F1-score (F1) as 99.36%, 99.36%, 99.53%, 99.32%, and 99.28%, respectively. Compared to PSPM [12] and FCM [19], MLNet achieves better specificity, precision and F1-score. Bold values indicate the higher performance.

B. Comparative Analysis Based on SIPaKMeD

Fig. 4 demonstrates the training and validation loss analyses of MLNet on the SIPaKMeD dataset. MLNet converges at a better speed. MLNet does not suffer from the overfitting problem since there are lesser differences between training and validation lines.

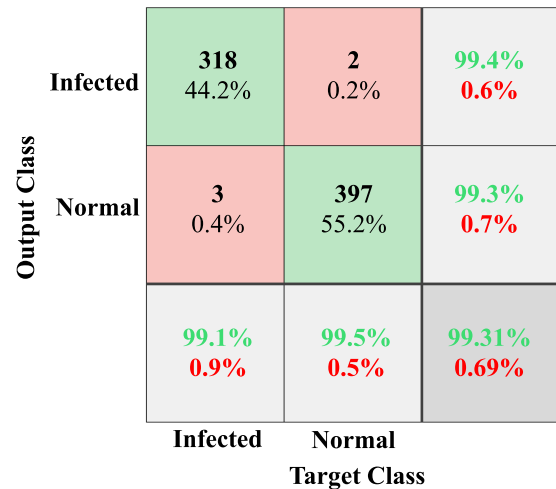


Fig. 5. Training and validation loss analyses of MLNet on SIPaKMeD dataset.

The model achieves an overall accuracy of 99.98% and 99.76% on the training and validation dataset, respectively.

In Fig. 5, the confusion matrix of MLNet on the SIPaKMeD dataset is shown. MLNet obtains remarkable performance on normal and infected samples. As a result of better balancing of infected versus normal cases, MLNet shows higher sensitivity and specificity on the SIPaKMeD dataset. MLNet achieves an accuracy of 99.31% in total.

Table VI shows the comparative analysis among MLNet and the existing cervical cancer classification techniques on SIPaKMeD dataset. It is found that each technique achieved remarkable results. VGG [24] achieved remarkable specificity and F1-score values as 99.50% and 99.26%, respectively. GCNN [15] obtained better F1-score value as 99.3%. In terms of accuracy, specificity, sensitivity/recall, precision, and F1-score, the proposed MLNet performs remarkably better with values of 99.31%, 99.37%, 99.26%, 99.29%, and 99.24%. MLNet

TABLE VI
COMPARATIVE ANALYSIS BASED ON SIPAKMED DATASET

Models	Acc	Spec.	Sens./Rec	Pre	F1
GCNN [15]	98.3	98.6	98.8	-	99.3
ResNet50 [16]	98.8	98.7	99.1	-	98.7
EPDFE [22]	98.26	-	98.28	98.27	98.28
CNN-F [23]	98.55	-	98.52	98.57	98.54
VGG [24]	97.80	99.50	98.10	99.26	98.43
ResNet152 [25]	94.89	-	96.0	96.0	96.0
CNN [26]	97.87	-	99.12	98.56	98.89
MLNet	99.31	99.37	99.26	99.29	99.24

TABLE VII
COMPARATIVE ANALYSIS BASED ON MENDELEY LBC

Models	Acc	Spec.	Sens./ Rec	Pre	F1
EPDFE [22]	99.27	-	98.21	99.26	98.73
CNN-F [23]	99.23	-	99.23	99.13	99.18
CNN [26]	99.27	-	99.27	99.14	99.20
c-CNN [27]	96.89	-	93.75	93.38	94.15
MLNet	99.36	99.47	99.32	99.35	99.32

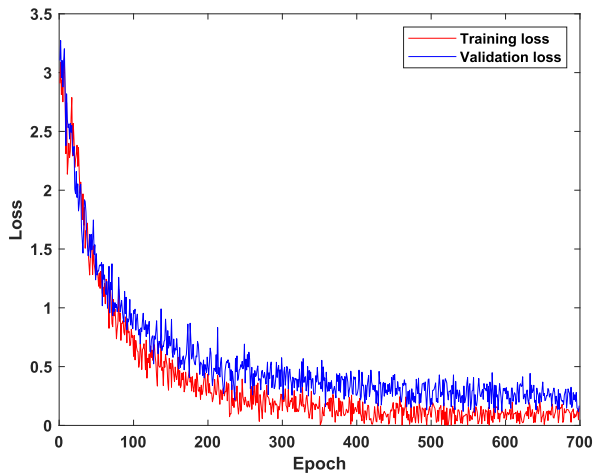


Fig. 6. Training and validation loss analyses of MLNet on Mendelely LBC dataset.

achieves higher accuracy, specificity, and precision values than GCNN [15] and ResNet50 [16].

C. Comparative Analysis Based on Mendelely LBC

Fig. 6 illustrates the training and validation losses of MLNet on the Mendelely LBC dataset. The convergence rate of MLNet is faster. Since there are fewer differences between training and validation lines, MLNet does not suffer from the overfitting problem. It achieves an accuracy of 100.00% and 99.76%, respectively on training and validation datasets.

Fig. 7 represents the confusion matrix of MLNet on Mendelely's LBC dataset. MLNet performs well on both infected and normal samples. MLNet demonstrates a high level of sensitivity and specificity on the Mendelely LBC dataset since it has a better balance between infected and normal cases. A total of 99.36% accuracy is achieved by MLNet.

Output Class	Target Class		
	Infected	Normal	
Infected	516 54.5%	2 0.2%	99.6% 0.4%
Normal	4 0.4%	426 44.9%	99.1% 0.9%
	99.2% 0.8%	99.5% 0.5%	99.36% 0.64%

Fig. 7. Training and validation loss analyses of MLNet on Mendelely LBC dataset.

Comparative analysis of MLNet and existing cervical models on Mendelely LBC dataset is shown in **Table V**. It demonstrates that MLNet provides remarkable performances in terms of accuracy, specificity, sensitivity/recall, precision, and F1-score, at 99.76%, 99.87%, 99.57%, 99.68%, and 99.62%, respectively.

D. Discussion

MLNet successfully overcomes the over-fitting, parameter tuning, and gradient vanishing problems. Since it solves the hyper-parameter tuning problem of CNN by defining it as a multi-objective problem. PSO is used to optimally define CNN architecture and DHDE is utilized to optimize the hyper-parameters of CNN layers. Thus, MLNet achieves better performance than the competitive models. Since the majority of competitive models used a trial and error approach to define the architecture and respective hyper-parameters. The proposed model is compared with some well-known existing cervical cancer classification models by considering three different datasets.

On the Herlev dataset, PSPM [12] achieved better results among the competitive models in terms of accuracy, specificity, and sensitivity by 99.7%, 99.3%, and 99.8%, respectively. These competitive models includes R-CNN [9], GAN [10], LCNN [11], DU-Net [13], FCN [14], DeepCervix [17], InceptionV3 [18], FCM [19], VGG [24], CNN [26], CNN-ELM [30], and CNN-SVM [31]. Also, FCM [19] achieved significantly better sensitivity than the competitive models including PSPM [12]. But PSPM [12] and FCM [19] only limited to specific classes only. Whereas the proposed MLNet achieves better overall performance in terms of accuracy, f-measure, sensitivity, specificity, and precision by 1.6254%, 1.5178%, 1.5780%, 1.7145%, and 1.4890%, respectively.

For comparative analysis on SIPaKMeD dataset, various competitive models are considered such as GCNN [15], ResNet50 [16], EPDFE [22], CNN-F [23], VGG [24], ResNet152 [25], and CNN [26]. Among these models, VGG [24] achieved better performance in terms of accuracy, f-measure,

TABLE VIII

COMPUTATIONAL SPEED ANALYSIS OF MLNET BY VARYING THE IMAGE SIZE

Image size	Time (in milliseconds)
256 × 256	0.58 ± 0.17
512 × 512	0.81 ± 0.21
1024 × 1024	1.28 ± 0.32
2048 × 2048	1.51 ± 0.48
4096 × 4096	3.25 ± 0.59
8192 × 8192	5.16 ± 1.29

sensitivity, specificity, and precision by 97.80%, 99.50%, 98.10%, 99.26%, and 98.43, respectively. Whereas MLNet outperforms every model in terms of performance metrics except VGG [24] in terms of specificity. But overall in terms of other performance metrics, the proposed model is significantly superior to VGG [24]. Overall, MLNet obtains better performance on the SIPaKMeD dataset than the existing models in terms of accuracy, f-measure, sensitivity, specificity, and precision by 2.1250%, 2.2455%, 1.9074%, 1.9258%, and 1.8975%, respectively.

On Mendeley LBC, various models have been considered such as EPDFE [22], CNN-F [23], CNN [26], and c-CNN [27]. Overall, MLNet achieves better performance on the Mendeley LBC dataset in terms of accuracy, f-measure, sensitivity, specificity, and precision by 1.4680%, 1.5845%, 1.3582%, 1.3926%, and 1.4125%, respectively.

By scaling the image sizes, the computational speed analysis of MLNet is achieved (see Table VIII). The images have been scaled to different sizes, including 256×256 , 512×512 , 1024×1024 , 2048×2048 , 4096×4096 , and 8192×8192 . It is observed that MLNet can provide diagnosis results at a better computational speed.

VI. CONCLUSION

We proposed a MLNet to overcome overfitting, parameter tuning, and gradient vanishing problems with existing deep learning models. A multi-objective problem was defined to optimize the hyper-parameters of CNN. The architecture of CNN was optimized by PSO. DHDE was then applied to optimize the hyper-parameters of CNN layers. By concatenating the PSO's particle with the DHDE's solution, optimal CNN configuration was achieved. F-score was used to measure the performance of CNN. MLNet was trained and validated on three benchmark datasets. Performance analyses revealed that MLNet achieved better performance on Herlev, SIPaKMeD, and Mendeley LBC datasets in terms of accuracy, f-measure, sensitivity, specificity, and precision. The proposed MLNet is not limited only to cervical cancer diagnosis. In near future, the proposed MLNet can be utilized for the classification of various computer vision and biomedical problems. Besides one can use MLNet for other kind of datasets such as audio, text, etc. by changing the CNN layers including activation function.

REFERENCES

[1] W. Small Jr. et al., "Cervical cancer: A global health crisis," *Cancer*, vol. 123, no. 13, pp. 2404–2412, 2017.

[2] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal, "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *Cancer J. Clinicians*, vol. 68, no. 6, pp. 394–424, 2018.

[3] T. A. Kessler, "Cervical cancer: Prevention and early detection," *Seminars Oncol. Nurs.*, vol. 33, pp. 172–183, 2017.

[4] M. E., "Pap-smear classification," M.S. thesis, Dept. Automat., 2003.

[5] D. C. Wilbur et al., "The becton Dickinson FocalPoint GS imaging system: Clinical trials demonstrate significantly improved sensitivity for the detection of important cervical lesions," *Amer. J. Clin. Pathol.*, vol. 132, no. 5, pp. 767–775, 2009.

[6] C. V. Biscotti et al., "Assisted primary screening using the automated Thin-Prep imaging system," *Amer. J. Clin. Pathol.*, vol. 123, no. 2, pp. 281–287, 2005.

[7] K. Bora, M. Chowdhury, L. B. Mahanta, M. K. Kundu, and A. K. Das, "Automated classification of Pap smear images to detect cervical dysplasia," *Comput. Methods Programs Biomed.*, vol. 138, pp. 31–47, 2017.

[8] L. Zhang, L. Lu, I. Nogue, R. M. Summers, S. Liu, and J. Yao, "DeepPap: Deep convolutional networks for cervical cell classification," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 6, pp. 1633–1643, Nov. 2017.

[9] K. H. S. Allehaibi et al., "Segmentation and classification of cervical cells using deep learning," *IEEE Access*, vol. 7, pp. 116925–116941, 2019.

[10] Z. Lin, Z. Gao, H. Ji, R. Zhai, X. Shen, and T. Mei, "Classification of cervical cells leveraging simultaneous super-resolution and ordinal regression," *Appl. Soft Comput.*, vol. 115, 2022, Art. no. 108208.

[11] W. Chen, L. Gao, X. Li, and W. Shen, "Lightweight convolutional neural network with knowledge distillation for cervical cells classification," *Biomed. Signal Process. Control*, vol. 71, 2022, Art. no. 103177.

[12] K. Sabeena and C. Gopakumar, "A hybrid model for efficient cervical cell classification," *Biomed. Signal Process. Control*, vol. 72, 2022, Art. no. 103288.

[13] J. Zhao, Q. Li, X. Li, H. Li, and L. Zhang, "Automated segmentation of cervical nuclei in Pap smear images using deformable multi-path ensemble model," in *Proc. IEEE 16th Int. Symp. Biomed. Imag.*, 2019, pp. 1514–1518.

[14] E. Hussain, L. B. Mahanta, C. R. Das, M. Choudhury, and M. Chowdhury, "A shape context fully convolutional neural network for segmentation and classification of cervical nuclei in Pap smear images," *Artif. Intell. Med.*, vol. 107, 2020, Art. no. 101897.

[15] J. Shi, R. Wang, Y. Zheng, Z. Jiang, H. Zhang, and L. Yu, "Cervical cell classification with graph convolutional network," *Comput. Methods Programs Biomed.*, vol. 198, 2021, Art. no. 105807.

[16] J. Li et al., "Cervical cell multi-classification algorithm using global context information and attention mechanism," *Tissue Cell*, vol. 74, 2022, Art. no. 101677.

[17] M. M. Rahaman et al., "DeepCervix: A deep learning-based framework for the classification of cervical cells using hybrid deep feature fusion techniques," *Comput. Biol. Med.*, vol. 136, 2021, Art. no. 104649.

[18] N. Dong, L. Zhao, C.-H. Wu, and J.-F. Chang, "Inception V3 based cervical cell classification combined with artificially extracted features," *Appl. Soft Comput.*, vol. 93, 2020, Art. no. 106311.

[19] T. Chankong, N. Theera-Umporn, and S. Auephanwiriyakul, "Automatic cervical cell segmentation and classification in Pap smears," *Comput. Methods Programs Biomed.*, vol. 113, no. 2, pp. 539–556, 2014.

[20] P. Wang, L. Wang, Y. Li, Q. Song, S. Lv, and X. Hu, "Automatic cell nuclei segmentation and classification of cervical Pap smear images," *Biomed. Signal Process. Control*, vol. 48, pp. 93–103, 2019.

[21] V. Martin, T. H. Kim, M. Kwon, M. Kuko, M. Pourhomayoun, and S. Martin, "A more comprehensive cervical cell classification using convolutional neural network," *J. Amer. Soc. Cytopathol.*, vol. 7, no. 5, 2018, Art. no. S66.

[22] O. Yaman and T. Tuncer, "Exemplar pyramid deep feature extraction based cervical cancer image classification model using Pap-smear images," *Biomed. Signal Process. Control*, vol. 73, 2022, Art. no. 103428.

[23] A. Manna, R. Kundu, D. Kaplun, A. Sinitica, and R. Sarkar, "A fuzzy rank-based ensemble of CNN models for classification of cervical cytology," *Sci. Rep.*, vol. 11, no. 1, pp. 1–18, 2021.





[24] H. Chen et al., "CytoBrain: Cervical cancer screening system based on deep learning technology," *J. Comput. Sci. Technol.*, vol. 36, no. 2, pp. 347–360, 2021.

[25] A. Tripathi, A. Arora, and A. Bhan, "Classification of cervical cancer using deep learning algorithm," in *Proc. 5th IEEE Int. Conf. Intell. Comput. Control Syst.*, 2021, pp. 1210–1218.

[26] H. Basak, R. Kundu, S. Chakraborty, and N. Das, "Cervical cytology classification using PCA and GWO enhanced deep features selection," *SN Comput. Sci.*, vol. 2, 2021, Art. no. 369.

- [27] N. K. Chauhan and K. Singh, "Impact of variation in number of channels in CNN classification model for cervical cancer detection," in *Proc. 9th Int. Conf. Rel. Infocom Technol. Optim.*, 2021, pp. 1–6.
- [28] J. Martínez-Más et al., "Classifying papanicolaou cervical smears through a cell merger approach by deep learning technique," *Expert Syst. Appl.*, vol. 160, 2020, Art. no. 113707.
- [29] A. Pal et al., "Deep multiple-instance learning for abnormal cell detection in cervical histopathology images," *Comput. Biol. Med.*, vol. 138, 2021, Art. no. 104890.
- [30] A. Ghoneim, G. Muhammad, and M. S. Hossain, "Cervical cancer classification using convolutional neural networks and extreme learning machines," *Future Gener. Comput. Syst.*, vol. 102, pp. 643–649, 2020.
- [31] A. D. Jia, B. Z. Li, and C. C. Zhang, "Detection of cervical cancer cells based on strong feature CNN-SVM network," *Neurocomputing*, vol. 411, pp. 112–127, 2020.
- [32] Z. Alyafeai and L. Ghouti, "A fully-automated deep learning pipeline for cervical cancer classification," *Expert Syst. Appl.*, vol. 141, 2020, Art. no. 112951.
- [33] T. Zhang et al., "Cervical precancerous lesions classification using pre-trained densely connected convolutional networks with colposcopy images," *Biomed. Signal Process. Control*, vol. 55, 2020, Art. no. 101566.
- [34] Y. Xiang, W. Sun, C. Pan, M. Yan, Z. Yin, and Y. Liang, "A novel automation-assisted cervical cancer reading method based on convolutional neural network," *Biocybernetics Biomed. Eng.*, vol. 40, no. 2, pp. 611–623, 2020.
- [35] M. F. Zorkafli, M. K. Osman, I. S. Isa, F. Ahmad, and S. N. Sulaiman, "Classification of cervical cancer using hybrid multi-layered perceptron network trained by genetic algorithm," *Procedia Comput. Sci.*, vol. 163, pp. 494–501, 2019.
- [36] C. Bhavani and A. Govardhan, "Cervical cancer prediction using stacked ensemble algorithm with SMOTE and RFERF," *Mater. Today Proc.*, 2021.
- [37] N. Akhtar and U. Ragavendran, "Interpretation of intelligence in CNN-pooling processes: A methodological survey," *Neural Comput. Appl.*, vol. 32, no. 3, pp. 879–898, 2020.
- [38] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw.*, 1995, pp. 1942–1948.
- [39] D. Tian and Z. Shi, "MPSO: Modified particle swarm optimization and its applications," *Swarm Evol. Computation*, vol. 41, pp. 49–68, 2018.
- [40] K. Wang, W. Gong, L. Deng, and L. Wang, "Multimodal optimization via dynamically hybrid niching differential evolution," *Knowl.-Based Syst.*, vol. 238, 2022, Art. no. 107972.
- [41] P. Singh, S. Chaudhury, and B. K. Panigrahi, "Hybrid MPSO-CNN: Multi-level particle swarm optimized hyperparameters of convolutional neural network," *Swarm Evol. Computation*, vol. 63, 2021, Art. no. 100863.
- [42] J. Jantzen, "Pap-smear (DTU/HERLEV) databases." 2008. Accessed: Dec. 17, 2021. [Online]. Available: mde-lab.aegean.gr/downloads
- [43] M. E. Plissiti, P. Dimitrakopoulos, G. Sfikas, C. Nikou, O. Krikoni, and A. Charchanti, "Sipakmed: A new dataset for feature and image based classification of normal and pathological cervical cells in Pap smear images," in *Proc. 25th IEEE Int. Conf. Image Process.*, 2018, pp. 3144–3148.
- [44] E. Hussain, "Liquid based cytology Pap smear images for multi-class diagnosis of cervical cancer," Mendeley Data, V4. 2019. Accessed: Dec. 28, 2021, doi: [10.17632/zddtpgzv63.4](https://doi.org/10.17632/zddtpgzv63.4). [Online]. Available: data.mendeley.com/datasets/zddtpgzv63/4
- [45] S. Yu et al., "Generative adversarial network based data augmentation to improve cervical cell classification model," *Math. Biosci. Eng.*, vol. 18, pp. 1740–1752, 2021.

Efficient Evolving Deep Ensemble Medical Image Captioning Network

Dilbag Singh , Senior Member, IEEE, Manjit Kaur , Senior Member, IEEE, Jazem Mutared Alanazi, Ahmad Ali AlZubi , and Heung-No Lee , Senior Member, IEEE

Abstract—With the advancement in artificial intelligence (AI) based E-healthcare applications, the role of automated diagnosis of various diseases has increased at a rapid rate. However, most of the existing diagnosis models provide results in a binary fashion such as whether the patient is infected with a specific disease or not. But there are many cases where it is required to provide suitable explanatory information such as the patient being infected from a particular disease along with the infection rate. Therefore, in this paper, to provide explanatory information to the doctors and patients, an efficient deep ensemble medical image captioning network (DCNet) is proposed. DCNet ensembles three well-known pre-trained models such as VGG16, ResNet152V2, and DenseNet201. Ensembling of these models achieves better results by preventing an overfitting problem. However, DCNet is sensitive to its control parameters. Thus, to tune the control parameters, an evolving DCNet (EDC-Net) was proposed. Evolution process is achieved using the self-adaptive parameter control-based differential evolution (SAPCDE). Experimental results show that EDC-Net can efficiently extract the potential features of biomedical images. Comparative analysis shows that on the Open-i dataset, EDC-Net outperforms the existing models in terms of BLUE-1, BLUE-2, BLUE-3, BLUE-4, and kappa statistics (KS) by 1.258%, 1.185%, 1.289%, 1.098%, and 1.548%, respectively.

Index Terms—Medical, diagnosis, pre-trained models, explanatory information.

I. INTRODUCTION

WITH the advancement in artificial intelligence (AI) based E-healthcare applications, the role of automated

Manuscript received 4 February 2022; revised 22 August 2022; accepted 30 September 2022. Date of publication 18 November 2022; date of current version 6 February 2023. This work was supported in part by the Ministry of Science and ICT, Korea, through the Information Technology Research Center Support Program under Grant IITP-2021-0-01835 supervised by the Institute for Information and Communications Technology Planning and Evaluation, in part by the National Research Foundation of Korea funded by the Korean Government, MSIP, under Grant NRF-2021R1A2B5B03002118, and in part by the Researchers Supporting Project, King Saud University, Riyadh, Saudi Arabia, under Grant RSP-2021/395. (Corresponding author: Heung-No Lee.)

Dilbag Singh, Manjit Kaur, and Heung-No Lee are with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea (e-mail: dg-gill2@gmail.com; manjitkaur@gist.ac.kr; heungno@gist.ac.kr).

Jazem Mutared Alanazi and Ahmad Ali AlZubi are with the Computer Science Department, Community College, King Saud University, Riyadh 11451, Saudi Arabia (e-mail: ajazem@ksu.edu.sa; aalzubi@ksu.edu.sa).

Digital Object Identifier 10.1109/JBHI.2022.3223181

diagnosis of various diseases has increased at a rapid rate. Deep learning models have recently been used by many researchers to classify patients suffering from a particular disease. But these models generally provide results in a binary fashion such as whether the patient is infected from a specific disease or not. However, there are many cases where it is required to provide suitable information to the patients such as the patient being infected with some disease with this much infection rate. Therefore, these days many researchers have started utilizing image captioning techniques to provide explanatory information to doctors and patients.

Recently, semantic concepts were used to detect the captions from the image. Image-caption pairs were used to train the concept detector. However, it suffers from vocabulary discrepancy and deficiency of required information [1]. To increase the accuracy of pathological information in diagnostic reports, Semantic fusion networks (SFNet) were utilized [2]. Attention-based models were also used in image captioning. The attention masks were queried using hidden states of LSTM from image features. It provided better image information for training deep sequential networks. However, these models did not ensure that layers significantly focused on regions of interest due to indirectly supervised learning [3]. These problems were overcome by utilizing differentiable neural networks to obtain the captions from the images [4]. To describe the disease information in ultrasound images, LSTM was used to decode the encoding vectors [5].

Most image captioning techniques are based on visual information and rarely rely on semantic content. Therefore, it is also needed to express the emotions of text descriptions for better captioning. Yang et al. [6] made sentences using both emotional and visual information by combining latent codes. The quality of AI-generated diagnostic reports is equally important as the development of models. However, [6] performs poorly for images with poor visibility.

Babar et al. [7] proposed a new measure to evaluate the quality of generated diagnostic reports. Convolutional neural network (CNN)-based image captioning models can retain only a few features of the original image. The image captioning DCNet based on a recurrent neural network (RNN) suffers from a gradient vanishing problem. The multimodal fusion method solved these issues by using CNN and RNN into the same DCNet for image captioning [8]. A semi-supervised deep generative DCNet generated the description more naturally from images [9]. An adaptive multimodal attention network was also designed to produce better quality captioned images [10].

Kavur et al. [11] utilized various ensemble models such as majority voting, average combiner, product combiner, and min/max combiner. To prevent overfitting problems with deep learning models, U-Net, deepmedic, V-Net, and dense V-networks were ensemble to enhance the liver segmentation from CT images.

However, it is found that the hyper-parameters selection of the existing models was achieved using the trial-and-error basis. Additionally, the designed model suffers from the over-fitting and gradient vanishing problem. To overcome these problems, an efficient EDC-Net model is designed.

The main contributions of this paper are as follows:

- 1) To provide explanatory information to the doctors and patients, an efficient deep ensemble medical image captioning network (DCNet) is proposed.
- 2) The proposed DCNet ensembles three well-known pre-trained models such as VGG16, ResNet152V2, and DenseNet201.
- 3) To tune the control parameters, evolving DCNet (EDC-Net) was proposed. The evolution process is achieved using the self-adaptive parameter control-based differential evolution (SAPCDE)
- 4) By considering benchmark datasets, extensive experiments are conducted.

The remainder of the paper is organized as follows: Section II describes related work. Section III provides a mathematical formulation of EDC-Net. Section IV presents the comparative analysis. Section V concludes the paper.

II. RELATED WORK

Yu et al. [12] used order embedding to caption the topic-oriented image. A convolutional neural network (CNN)-based classifier is used to select the topics for images from candidates. Wang et al. [13] captioned the images using a recurrent memory network (RMN). In training, topic words were recorded from a topic repository. After that, the image was tested using the retrieval method for the generation of a topic word. Finally, the sentence was generated through a recurrent memory network by incorporating the retrieved topic words. Zhang et al. [1] proposed an image captioning DCNet using missing concepts mining and online positive recall. The suitable captions were automatically generated using the element-wise selection method. Zhang et al. [3] captioned the images using the visual aligning attention DCNet (VAA). The visual aligning loss was designed to optimize the attention layer in the training stage. The non-visual words were filtered out using visual vocab to minimize their effect on the attention layer. Zhou et al. [14] enhanced single-phase image captioning using a saliency-enhanced re-captioning model. In this, two-phase learning was applied to get better results. In the first phase, semantic and visual saliency cues were extracted from the model. In the second phase, cues were fused for the self-boosting of the model. Zhao et al. [15] captioned the images using DCNet adaption and cross-modal retrieval in cross-domain. Firstly, the source domain was used to pre-train the cross-modal and then utilized in the target domain to extract

the initial image-sentence pairs. These pairs were further reefered using a retrieval model. Yang et al. [16] proposed cross-domain image captioning using Multitask learning (ML). Textual explanations of the images were generated using CNN-LSTM. Images were synthesized using a Conditional generative adversarial network (C-GAN) based on generated text descriptions. Hoxha et al. [17] captioned the images using CNN and Recurrent neural network (RNN). Firstly, visual features were extracted and translated into a textual explanation. Secondly, embedding techniques were used to convert the textual explanation into feature vectors. Finally, similar images were retrieved by calculating the similarity between vectors of textual explanation and archive images. Huang et al. [18] utilized a Multimodal attribute detector (MAD) and Subsequent attribute predictor (SAP) to improve the performance of image captioning. MAD used image features as well as word embedding to improve attribute detection accuracy. A concise attribute was predicted with SAP every time to reduce the diversity of image attributes. Yang et al. [19] proposed a Dual generator generative adversarial network (DGGAN) based image captioning model. This technique ensemble the generation-based and retrieval-based image captioning models. Yuan et al. [20] captioned the images using multi-label attribute graph convolution and multi-level attention. The attention module focused on both specific spatial and scale features. For image captioning, attribute features were learned using the attribute graph convolution module. Huang et al. [21] utilized the Denoising-based multi-scale feature fusion (DMSFF) technique to caption the images. Monay et al. [22] captioned the images using the probabilistic latent semantic analysis (PLSA) model. The textual and visual modalities are assumed equally by an expectation-maximization algorithm. Yu et al. [23] proposed an image captioning DCNet using Multimodal Transformer (MT) model. Multi-view visual features were also introduced to improve the performance. Xian et al. [24] used multimodal LSTM to caption the images. Amirian et al. [25] proposed a deep learning-based image captioning model. Li et al. [26] designed an efficient image captioning model that can be used to extract potential information from digital images. It has shown effective performance over competitive models. But this model suffers from the over-fitting issue.

Park et al. [27] designed a multi-difference average pooling LSTM (mDiAP-LSTM)-based medical image captioning model. The most appropriate method for capturing the differences was determined through a study of feature representation methods. Hou et al. proposed an efficient Full-adversarial reinforcement learning (Full-ARL) model for medical image captioning. Li et al. [28] designed an efficient knowledge-driven encode, retrieve, paraphrase (KERP) model to obtain better medical image captions. It decomposes captions into explicit disease-related abnormality graphs, which were then analyzed using natural language modeling. Hou et al. [29]. An overall score based on accuracy and fluency was provided by two additional discriminators as the evaluator. A report generator was implemented, which produces discrete sequences of words based on decision probabilities, as opposed to generative adversarial networks (GANs) used in image generation. Furthermore, it prevented

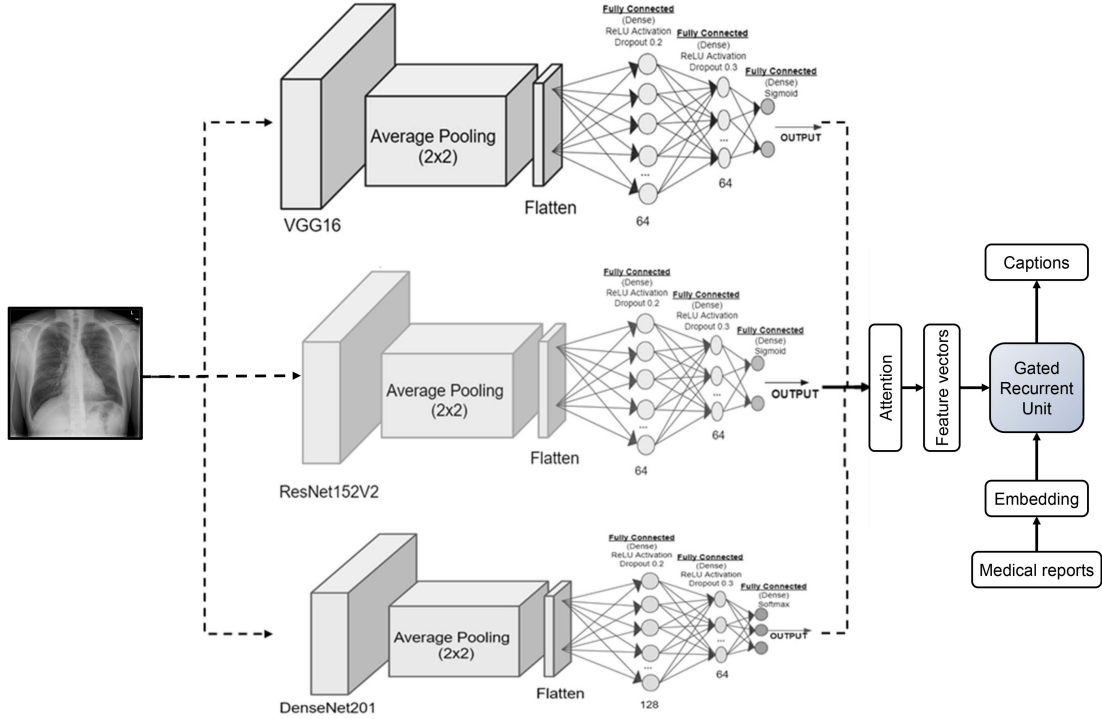


Fig. 1. Proposed ensemble deep transfer network with gated recurrent unit.

the gradients from being transmitted between discriminators and generators.

Wang et al. [30] designed an efficient relational-topic retrieval and generation framework (R-paraNet). Reports incorporated semantically consistent medical terms and encourage the generation of sentences for rare abnormal descriptions. Li et al. [31] designed a hybrid retrieval-generation reinforced agent (HRGR-Agent) that integrated retrieval-based strategies with sophisticated learning techniques to provide robust image captions. Hierarchical decision-making was employed by HRGR-Agent. To obtain a caption, a low-level generation module invoked either a high-level retrieval policy or a high-level retrieval policy module. Reinforcement learning guided the HRGR-Agent's updates via rewards at the word and sentence levels. Jing et al. [32] designed a multitask learning model that predicted tags and paragraphs simultaneously. It utilized co-attention to locate abnormal regions and generate narrations for them. The hierarchical LSTM (HLSTM) was proposed for generating long paragraphs.

It is observed that the existing models suffer from hyper-parameters tuning problems. Moreover, the designed model exhibits over-fitting and gradient vanishing problems.

III. PROPOSED MODEL

This section discusses the proposed medical image captioning model.

A. Ensemble Deep Transfer Model

An ensemble deep transfer network (DCNet) is proposed. VGG16 [33], ResNet152V2 [34], and DenseNet201 [35] models

have been used to develop the proposed model. Fig. 1 shows the EDC-Net with a gated recurrent unit. Ensembling of models provides better results by preventing over-fitting [36], [37]. It also enhances the extraction of features and performance of supervised models [38], [39]. Fig. 1 demonstrates the proposed ensemble deep transfer model. 128 neurons have been utilized for input dense layer [40]. VGG16 [33], ResNet152V2 [34], and DenseNet201 [35] models are used to obtain potential features. These models have been trained using 20 epochs with a batch size of 8. Fully connected layers [41] having size 128 neurons along with dropouts of 0.2 and 0.25, respectively have been utilized to prevent memorization problems with the competitive models. $l_r = 0.001$ has been utilized as learning rate.

B. Gated Recurrent Unit

Cho et al. [42] designed gated recurrent unit (GRU). GRU allows every recurrent unit to dynamically obtain dependencies of different time scales. Similar to LSTM, GRU has gating units that modulate the knowledge flow within the unit, but, without utilizing any additional memory cells.

Activation (α_n^k) of GRU at time t is a linear interpolation among candidate activation ($\tilde{\alpha}_n^k$) and succeeding activation (α_{t-1}^k). α_n^k can be computed as:

$$\alpha_n^k = (1 - \beta_n^k)\alpha_{t-1}^k + \beta_n^k\tilde{\alpha}_n^k, \quad (1)$$

Here, update gate (β_n^k) monitors and controls the activation. An update gate can be evaluated as:

$$\beta_n^k = \sigma(W_z\gamma_n + U_z\alpha_{t-1}^k). \quad (2)$$

Here, $W_r \gamma_n$ represents the weighted matrix. GRU is unable to limit the degree to which its state is exposed. However, it can expose the whole state during every iteration.

A candidate activation ($\tilde{\alpha}_n^k$) can be computed [43] as:

$$\tilde{\alpha}_n^k = \tanh(W_r \gamma_n + U(\mathbf{r}_n \odot \alpha_{t-1}))^k \quad (3)$$

Here, \odot shows an element-wise multiplication. \mathbf{r}_n contains a group of reset gates.

When r_n^k approaches 0, the reset gate can efficiently develop a unit act as if it is utilizing the first symbol of the input sequence by forgetting the earlier evaluated state.

Reset gate (r_n^k) can be evaluated according to the update gate as:

$$r_n^k = \sigma(W_r \gamma_n + U_r \alpha_{t-1})^k. \quad (4)$$

C. Adaptive Differential Evolution

The proposed DCNet suffers from the hyper-parameters tuning issue. Therefore, in this paper, a differential evolution variant is used to evolve the proposed model. Bilingual Evaluation Understudy (BLUE) [44] is used as an objective function. Differential evolution (DE) is used in various fields due to its advantages such as strong robustness, good performance, and simple structure to solve optimization problems. The performance of DE mainly depends on the selection of control parameters (crossover rate CR , scale factor F , and population size NZ) and trial vector generation strategy (crossover and mutation). According to the problem nature, these parameters should be selected for better optimization results. The setting of these parameters is a challenging task for any problem. The solution to this problem is given by Cui et al. [45] by proposing self-adaptive parameter control-based DE (SAPCDE). It is based on the idea that good parameters should be propagated from generation to generation and the bad parameters should learn from the good parameters. In SAPCDE, two populations are utilized i.e., the solution population and the parameter population. Every solution has its control parameters. Parameter population is also evolved with each generation. SAPCDE is a combination of basic DE and parameter self-adaptation control methods.

Lets suppose that initial parameter population for DCNet is represented as $C^0 = \{C_1^0, C_2^0, \dots, C_{NZ}^0\}$ with $C_i^0 = \{F_{i,1}^0, CR_{i,2}^0\}$, where NZ denotes the population size. The initial solution population is represented as $S^0 = \{S_1^0, S_2^0, \dots, S_{NZ}^0\}$ with $S_i^0 = \{s_{i,1}^0, s_{i,2}^0, \dots, s_{i,D}^0\}$, where D represents the number of variables. G_n represents the number of generations. The parameter population evolved in the same way as the solution population in DE. Initially, parameter population is generated uniformly and randomly between $[0, 0]$ and $[1, 1]$. Thereafter, for each individual $C_i^{G_n}$, mutation parameter ($VC_i^{G_n}$) is generated using mutation operator such as

$$VC_i^{G_n} = C_{r1}^{G_n} + CF \cdot (C_{r2}^{G_n} - C_{r3}^{G_n}) \quad (5)$$

$$VC_i^{G_n} = C_{r1}^{G_n} + CF \cdot (SC_k^{G_n} - C_{r2}^{G_n}) \quad (6)$$

Here, C_{r1} , C_{r2} , and C_{r3} are selected from parameter population randomly. $SC_k^{G_n}$ denotes a good parameter that is selected

randomly. Next, the trailing parameter ($TC_i^{G_n}$) is generated using crossover operator such as

$$TC_i^{G_n} = \begin{cases} VC_{i,j}^{G_n}, & \text{if } (rand_{i,j} \leq CCR \text{ or } j == j_{rand}) \\ C_{i,j}^{G_n}, & \text{Otherwise} \end{cases} \quad (7)$$

where $j = \{1, 2\}$ and $i = \{1, 2, \dots, NZ\}$. $rand_{i,j} \in [0, 1]$ represents the uniform random number. $CCR \in [0, 1]$ denotes the new CR. Lastly, the selection operator is applied to select the good parameter for the next generation. In SAPCDE, good parameter individual $C_i^{G_n}$ is one which helps the $S_i^{G_n}$ to produce better offspring $TS_i^{G_n}$. Otherwise, $C_i^{G_n}$ is considered as a bad control parameter. The selection operator for a good parameter is defined as

if $C_i^{G_n}$ is a good parameter

$$C_i^{G_{n+1}} = \begin{cases} C_i^{G_n}, & \text{if } rand(0, 1) < \lambda_1 \\ TC_i^{G_{n+1}}, & \text{Otherwise} \end{cases} \quad (8)$$

else

$$C_i^{G_{n+1}} = \begin{cases} TC_i^{G_{n+1}}, & \text{if } rand(0, 1) < \lambda_2 \\ C_i^{G_n}, & \text{Otherwise} \end{cases} \quad (9)$$

where λ_1 and λ_2 control the values of parameters to explore the new values and keep the previous values. The basic working of SAPCDE is illustrated in Algorithm 1. Initially, solution and parameter populations are generated in line 1 and line 2, respectively. In line 3, number of generation (G_n) is initialized to 1. F_{eval} represents the number of function evaluations. Line 5 represents the termination condition of the algorithm when F_{eval} reaches F_{max} . GP stores the good parameter individuals (line 6). It is initialized with 0. The solution population is evolved through lines 7 to 16. In line 7, the mutation operator is applied to generate a mutant vector ($SV_i^{G_n}$) using parameter individual $C_i^{G_n}$ such as

$$SV_i^{G_n} = S_{r1}^{G_n} + F_{i,1}^{G_n} \cdot (S_{r2}^{G_n} - S_{r3}^{G_n}) \quad (10)$$

where S_{r1} , S_{r2} , and S_{r3} are randomly selected from solution population. $F_{i,1}^{G_n} \in C_i^{G_n}$ represents the scale factor. In line 8, crossover operator is applied to obtain a trial vector ($TS_i^{G_n}$) using $C_i^{G_n}$ such as

$$TS_i^{G_n} = \begin{cases} SV_{i,j}^{G_n}, & \text{if } (rand_{i,j} \leq CR_{i,2} \text{ or } j == j_{rand}) \\ S_{i,j}^{G_n}, & \text{Otherwise} \end{cases} \quad (11)$$

where $CR_{i,2} \in C_i^{G_n}$ and $i = \{1, 2, \dots, NZ\}$. A selection operator is applied to select the best solution (lines 10–15). If fitness of $TS_i^{G_n}$ is better than $S_i^{G_n}$, the corresponding $C_i^{G_n}$ is considered as a good parameter individual and flagged as 1 (i.e., $val = 1$) (line 11). It is also added in the GP in line 12. If $C_i^{G_n}$ is a bad parameter, then it is flagged as 0 (line 14). Next, the parameter population is evolved using mutation, crossover, and selection operators (lines 17–37). If $GP = 0$ at any G_n , then bad parameters are initialized randomly (line 31) to explore the new values.

Algorithm 1: Self-Adaptive Parameter Control-Based Differential Evolution for Evolving Ensemble Model.

```

1 Generate initial solution ( $S^0$ ) and parameter population ( $C^0$ )
2 Set  $G_n = 1$  and  $F_{eval} = 0$ 
3 while  $F_{eval} < F_{max}$  do
4   Set  $GP = 0$ 
5   for  $i = 0$  to  $NZ$  do
6     Mutant vector  $SV_i^{G_n}$  is obtained using Eq. 10
       and  $C_i^{G_n}$ 
7     Trial vector  $TS_i^{G_n}$  is obtained using Eq. 11 and
        $C_i^{G_n}$ 
8     if  $f(TS_i^{G_n}) \geq f(S_i^{G_n})$  then
9        $S_i^{G_{n+1}} = TS_i^{G_n}$ ,  $val(i) = 1$ 
10      Put  $C_i^{G_n}$  into  $GP$ 
11     else
12        $S_i^{G_{n+1}} = S_i^{G_n}$ ,  $val(i) = 0$ 
13     end
14   end
15   for  $i = 1$  to  $NZ$  do
16     if  $val(i) == 1$  then
17       if  $rand(0, 1) < \lambda_1$  then
18          $C_i^{G_{n+1}} = C_i^{G_n}$ 
19       else
20         Generate  $TC_i^{G_n}$  using Eq. (5) and Eq.
           (7)
21          $C_i^{G_{n+1}} = TC_i^{G_n}$ 
22       end
23     else
24       if  $rand(0, 1) < \lambda_2$  then
25         if  $GP \neq 0$  then
26           Generate  $TC_i^{G_n}$  using Eqs. (6) and
             (7)
27            $C_i^{G_{n+1}} = TC_i^{G_n}$ 
28         else
29           initialize  $C_i^{G_{n+1}}$  randomly
30         end
31       else
32          $C_i^{G_{n+1}} = C_i^{G_n}$ 
33       end
34     end
35   end
36    $G_n = G_{n+1}$ 
37 end

```

IV. PERFORMANCE ANALYSIS

In this paper, a dataset is obtained from Open-i [46]. It contains chest X-ray images and radiology text reports. Each image is paired with respective captions. 7,471 chest X-ray images are available with frontal and lateral views of given patients. Additionally, VGG16 [33], ResNet152V2 [34], and DenseNet201 [35] models are also ensembled using majority voting (MV), min, and max combiner. These models are

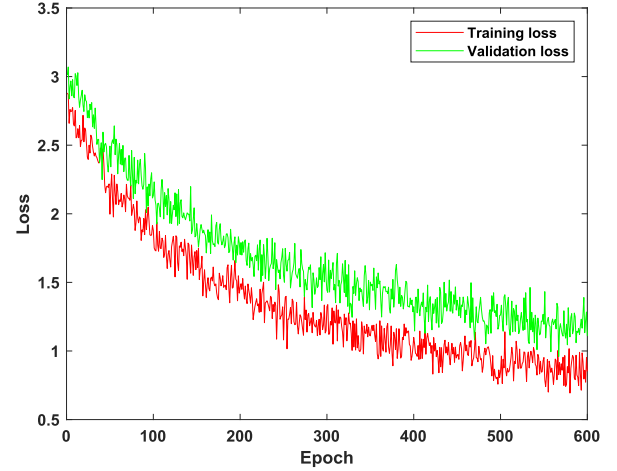


Fig. 2. Training and validation loss analysis of VGG16.

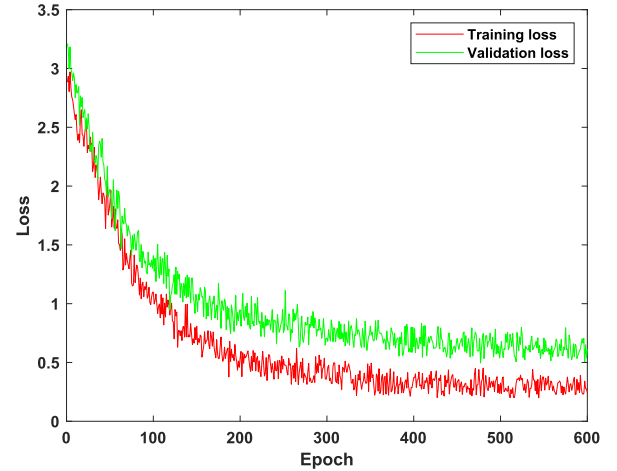


Fig. 3. Training and validation loss analysis of ResNet152V2.

renamed as MVE_1, Min_4, and Max_5, respectively. Also, inspired from [47], during the evolution phase, validation data is used to evaluate the performance of EDC-Net. The final trained EDC-Net is then tested on the training dataset.

A. Training and Validation Loss Analysis

This section discusses the training and validation analysis using the loss curves. Fig. 2 shows the training and validation loss analysis of VGG16 with respect to a number of epochs. It is found that the VGG16 achieves the best training and validation loss values as 0.7387 and 1.3481, respectively. It shows that there is an impact of over-fitting too.

Fig. 3 demonstrates the training and validation loss analysis of ResNet152V2. It is found that RESNET152 V achieves the best training and validation loss values of 0.6297 and 1.1726, respectively. It shows that there is an impact of over-fitting. But it shows better performance than VGG-16. Fig. 4 shows the training and validation loss analysis of DenseNet201. It is found that the best training and validation loss values are 0.3207 and 0.5218, respectively), thus DenseNet201 is least affected by the

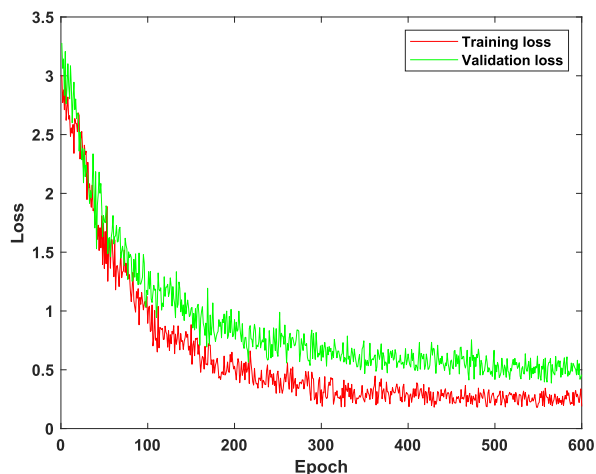


Fig. 4. Training and validation loss analysis of DenseNet201.

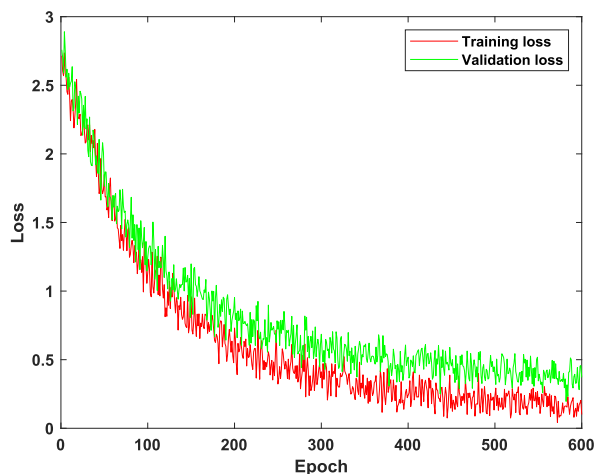


Fig. 5. Training and validation loss analysis of the proposed EDC-Net.

over-fitting problem. But still, there is room for improvement in the convergence curve. Fig. 5 demonstrates the training and validation loss analysis of EDC-Net. It is found that EDC-Net approaches toward minimum possible training loss. Also, there is a lesser difference between best training and validation loss values, (i.e., 0.2189 and 0.4368, respectively), thus EDC-Net is least affected by the over-fitting problem.

B. Visual Analysis

Fig. 6 shows the correctly obtained captions. It is found that the predicted captions are identical to the actual captions. So, for such captions, we received a maximum BLUE score. Since we have considered it as a classification problem, therefore, it represents the truly predicted class too. It clearly shows that EDC-Net can effectively provide captions for medical images.

Fig. 7 demonstrates the incorrectly obtained captions. It is found that the predicted captions are far away from the actual captions. So, for such captions, we received a minimum BLUE score. Since we have considered it as a classification problem, therefore, it represents the falsely predicted class

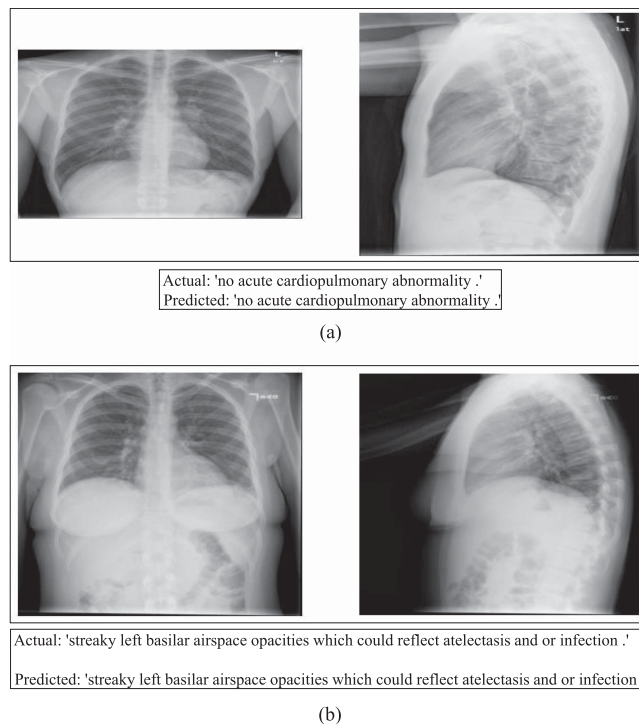


Fig. 6. Correctly classified captions.

TABLE I
BLUE SCORE AND KAPPA STATISTICS (KS) ANALYSIS OF THE EDC-NET

Model	BLUE-1	BLUE-2	BLUE-3	BLUE-4	KS
RMN	0.563	0.342	0.229	0.125	0.919
VGG16	0.568	0.364	0.249	0.126	0.923
CNN-RNN	0.511	0.351	0.253	0.123	0.912
DMSFF	0.539	0.353	0.244	0.126	0.924
DGGAN	0.563	0.365	0.238	0.127	0.932
ResNet152V2	0.571	0.309	0.256	0.119	0.896
DenseNet201	0.574	0.327	0.227	0.128	0.938
MVE_1	0.573	0.363	0.252	0.127	0.934
Min_1	0.571	0.361	0.250	0.124	0.915
Max_1	0.569	0.359	0.249	0.121	0.908
EDC-Net	0.577	0.367	0.258	0.129	0.952

too. It clearly shows that in certain cases when the visibility of the images is poor then EDC-Net fails to provide correct captions.

C. Quantitative Analysis

This section discusses the performance analyses using the confusion matrix. Fig. 8 shows the confusion matrix obtained from the VGG16 on the medical image captioning dataset. It is found that the VGG16 achieves 93.5% accuracy. Fig. 9 shows the confusion matrix obtained from the ResNet152V2 on the medical image captioning dataset. It is found that the ResNet152V2 achieves 94.8% accuracy. Fig. 10 shows the confusion matrix obtained from the DenseNet201 on the medical image captioning dataset. It is found that the DenseNet201 achieves 95.4% accuracy. Fig. 11 shows the confusion matrix obtained from EDC-Net on the medical image captioning dataset. It is found

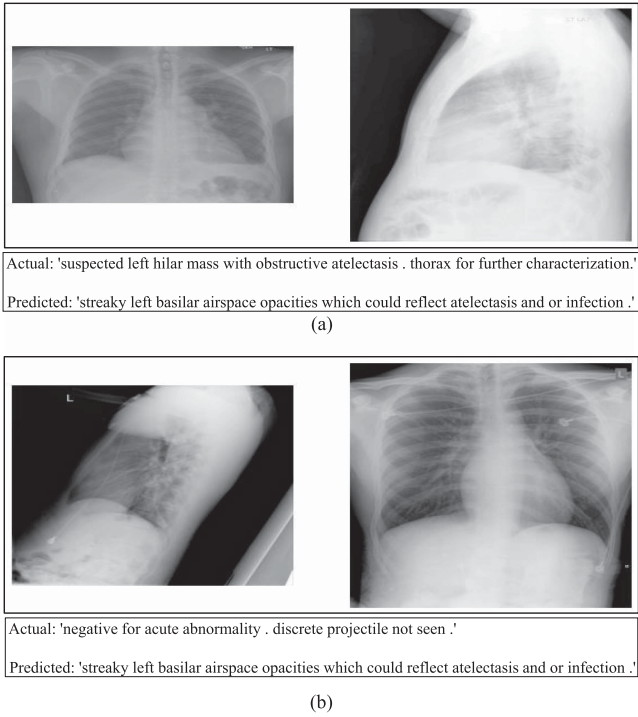


Fig. 7. Incorrectly classified captions.

Output Class	Class 1	Class 2	Class 3	Class 4	
Class 1	22 14.4%	4 2.6%	0 0.0%	1 0.7%	81.5% 18.5%
Class 2	5 3.3%	31 20.3%	0 0.0%	0 0.0%	86.1% 13.9%
Class 3	0 0.0%	0 0.0%	41 26.8%	0 0.0%	100% 0.0%
Class 4	0 0.0%	0 0.0%	0 0.0%	49 32.0%	100% 0.0%
	81.5% 18.5%	88.6% 11.4%	100% 0.0%	98.0% 2.0%	93.5% 6.5%
	Class 1	Class 2	Class 3	Class 4	

Fig. 8. Confusion matrix analysis of VGG16 on medical image captioning dataset.

that EDC-Net achieves 97.4% accuracy. Thus, EDC-Net outperforms the competitive models with an average improvement of 2.0%.

D. Discussion

Table I shows the BLUE score [44] analysis of EDC-Net. Comparisons are drawn between the proposed and the

Output Class	Class 1	Class 2	Class 3	Class 4	
Class 1	26 17.0%	1 0.7%	0 0.0%	1 0.7%	92.9% 7.1%
Class 2	1 0.7%	34 22.2%	0 0.0%	3 2.0%	89.5% 10.5%
Class 3	0 0.0%	0 0.0%	41 26.8%	2 1.3%	95.3% 4.7%
Class 4	0 0.0%	0 0.0%	0 0.0%	44 28.8%	100% 0.0%
	96.3% 3.7%	97.1% 2.9%	100% 0.0%	88.0% 12.0%	94.8% 5.2%
	Class 1	Class 2	Class 3	Class 4	

Fig. 9. Confusion matrix analysis of ResNet152V2 on medical image captioning dataset.

Output Class	Class 1	Class 2	Class 3	Class 4	
Class 1	27 17.6%	4 2.6%	0 0.0%	0 0.0%	87.1% 12.9%
Class 2	0 0.0%	28 18.3%	0 0.0%	0 0.0%	100% 0.0%
Class 3	0 0.0%	0 0.0%	41 26.8%	0 0.0%	100% 0.0%
Class 4	0 0.0%	3 2.0%	0 0.0%	50 32.7%	94.3% 5.7%
	100% 0.0%	80.0% 20.0%	100% 0.0%	100% 0.0%	95.4% 4.6%
	Class 1	Class 2	Class 3	Class 4	

Fig. 10. Confusion matrix analysis of DenseNet201 on medical image captioning dataset.

competitive models such as RMN [13], VGG16 [33], CNN-RNN [7], DMSFF 9057472, DGGAN [19], ResNet152V2 [34], and DenseNet201 [35] using BLUE scores [44] and kappa statistics (KS) [48], [49]. It is found that DenseNet201 [35] achieved the highest BLUE-1 as 0.574 as compared to the other competitive models. DGGAN [19] and VGG16 [33] models have achieved highest BLUE-2 values as 0.365 and 0.364, respectively. ResNet152V2 [34] achieved highest BLUE-3 value as 0.56 than the existing models. DenseNet201 [35]. Ensemble-based models, i.e., MVE_1, Min_4, and Max_5 achieved better results than most of the existing models. Out of these models,

Output Class	Class 1	Class 2	Class 3	Class 4	
Class 1	24 15.7%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Class 2	3 2.0%	35 22.9%	1 0.7%	0 0.0%	89.7% 10.3%
Class 3	0 0.0%	0 0.0%	40 26.1%	0 0.0%	100% 0.0%
Class 4	0 0.0%	0 0.0%	0 0.0%	50 32.7%	100% 0.0%
	88.9% 11.1%	100% 0.0%	97.6% 2.4%	100% 0.0%	97.4% 2.6%
	Class 1	Class 2	Class 3	Class 4	
	Target Class				

Fig. 11. Confusion matrix analysis of EDC-Net on medical image captioning dataset.

Source	SS	df	MS	F	Prob>F
Columns	2.267	10	0.2267	37.7833	2.016e-04
Error	0.294	49	0.0060		
Total	2.561	59			

Fig. 12. ANOVA table of BLUE-1.

Source	SS	df	MS	F	Prob>F
Columns	2.836	10	0.2836	32.9767	0.0156
Error	0.423	49	0.0086		
Total	3.259	59			

Fig. 13. ANOVA table of BLUE-2.

Source	SS	df	MS	F	Prob>F
Columns	0.53724	10	0.3301	47.8405	0.0023
Error	0.44408	20	0.0069		
Total	0.98132	24			

Fig. 14. ANOVA table of BLUE-3.

MVE_1 outperformed the Min_4 and Max_5. In terms of KS, DGGAN achieved remarkable performance over the existing models. From Table I, it is found that EDC-Net outperforms the existing models by achieving higher values of BLUE scores. Bold values indicate higher performance. It is found that the proposed EDC-Net outperforms the existing models in terms

Source	SS	df	MS	F	Prob>F
Columns	3.198	10	0.3198	53.3000	0.0003
Error	0.297	49	0.0060		
Total	3.495	59			

Fig. 15. ANOVA table of BLUE-4.

Source	SS	df	MS	F	Prob>F
Columns	1.748	10	0.1748	22.4102	1.435e-03
Error	0.385	49	0.0078		
Total	2.133	59			

Fig. 16. ANOVA table of kappa statistics (KS).

TABLE II
COMPARATIVE ANALYSIS OF EDC-NET WITH THE STATE-OF-THE-ART MODELS

Model	BLUE-1	BLUE-2	BLUE-3	BLUE-4
mDiAP-LSTM [27]	0.373	0.226	0.147	0.101
Full-ARL [29]	-	-	-	0.125
R-paraNet (VGG-19)[30]	0.505	0.329	0.230	0.168
R-paraNet (DneseNet-19)[30]	0.503	0.333	0.236	0.175
HRGR-Agent [31]	0.438	0.298	0.208	0.151
HLSTM [32]	0.455	0.288	0.205	0.154
KERP [28]	0.216	0.214	0.087	0.066
EDC-Net	0.516	0.348	0.238	0.178

of BLUE-1, BLUE-2, BLUE-3, BLUE-4, and KS by 1.258%, 1.185%, 1.289%, 1.098%, and 1.548%, respectively. Also, additional ANOVA analyses are performed on each performance metric which has shown that the EDC-Net significantly outperforms the competitive models.

For statistical analysis, ANOVA is used. Each performance metric has the following hypotheses:

$$\begin{cases} H_0 & \mu M_1 = \mu M_2 = \dots = \mu M_{11}, \\ H_A & \text{Means are not equal.} \end{cases} \quad (12)$$

Here, Null and alternate hypotheses are defined by H_0 and H_A , respectively. μM_i represents EDC-Net and the existing image captioning models. M_{11} depicts EDC-Net. The ANOVA table contains degrees of freedom, a sum of squares, the mean sum of squares, F-statistics (F_V), and P-value (p). If p value of F_V falls below the significance level, we will reject H_0 and state that there is a significant difference between the models. For all the metrics analysis presented in Table I, H_A is accepted (see Figs. 12–16). It shows that there is a significant difference in performance between different models.

Table II shows the comparative analysis between the EDC-Net and state-of-the-art models on IU X-RAY [50]. It is found that EDC-Net outperforms the state-of-the-art models in terms of BLUE-1, BLUE-2, BLUE-3, and BLUE-4 by 1.357%, 1.249%, 1.115%, and 1.031%, respectively.

V. CONCLUSION

To provide explanatory information to the doctors and patients, an efficient deep ensemble medical image captioning network (DCNet) was proposed. DCNet has ensembled three well-known pre-trained models such as VGG16, ResNet152V2, and DenseNet201. Ensembling of these models has shown better results by preventing over-fitting. DCNet can efficiently extract the potential features of biomedical images, but it is sensitive to its control parameters. Therefore, to tune the control parameters, evolving DCNet (EDC-Net) was proposed. Evolution was achieved using the self-adaptive parameter control-based differential evolution. Comparative analysis has shown that EDC-Net achieves higher performance than the existing models to provide explanatory information for biomedical images. Comparative analysis has shown that EDC-Net achieves 97.4% accuracy. According to the results obtained using Open-i dataset, the proposed EDC-Net outperformed the existing models in terms of BLUE-1, BLUE-2, BLUE-3, BLUE-4, and KS by 1.258%, 1.185%, 1.289%, 1.098%, and 1.548%, respectively. Additionally, the EDC-Net model outperformed the state-of-the-art models on IU X-RAY dataset in terms of BLUE-1, BLUE-2, BLUE-3, and BLUE-4 by 1.357%, 1.249%, 1.115%, and 1.031%, respectively.

In the near future, to handle the poor visibility, noise, poor registration, etc. kind of problems with medical images, we will integrate various preprocessing techniques such as image filtering, image registration, visibility restoration, etc. with the proposed EDC-Net. Since preprocessing techniques may reduce the performance of EDC-Net, therefore, a novel selective preprocessing model will be designed which will decide whether preprocessing is required or not. If preprocessing is required then which operation(s) should be applied on the given image(s).

Data Availability Statement: The data collected during the data collection phase are available from the corresponding authors upon request.

Conflicts of Interest: The authors would like to confirm there are no conflicts of interest regarding the study.

REFERENCES

- [1] M. Zhang, Y. Yang, H. Zhang, Y. Ji, H. T. Shen, and T.-S. Chua, "More is better: Precise and detailed image captioning using online positive recall and missing concepts mining," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 32–44, Jan. 2019.
- [2] X. Zeng, L. Wen, Y. Xu, and C. Ji, "Generating diagnostic report for medical image by high-middle-level visual information incorporation on double deep learning models," *Comput. Methods Programs Biomed.*, vol. 197, 2020, Art. no. 105700.
- [3] Z. Zhang, W. Zhang, W. Diao, M. Yan, X. Gao, and X. Sun, "VAA: Visual aligning attention model for remote sensing image captioning," *IEEE Access*, vol. 7, pp. 137355–137364, 2019.
- [4] R. Sharma, A. Kumar, D. Meena, and S. Pushp, "Employing differentiable neural computers for image captioning and neural machine translation," in *Proc. Int. Conf. Smart Sustain. Intell. Comput. Appl. Procedia Comput. Sci.*, 2020, vol. 173, pp. 234–244.
- [5] X. Zeng, L. Wen, B. Liu, and X. Qi, "Deep learning for ultrasound image caption generation based on object detection," *Neurocomputing*, vol. 392, pp. 132–141, 2020.
- [6] J. Yang, Y. Sun, J. Liang, B. Ren, and S.-H. Lai, "Image captioning by incorporating affective concepts learned from both visual and textual components," *Neurocomputing*, vol. 328, pp. 56–68, 2019.
- [7] Z. Babar, T. van Laarhoven, F. M. Zanzotto, and E. Marchiori, "Evaluating diagnostic content of AI-generated radiology reports of chest x-rays," *Artif. Intell. Med.*, vol. 116, 2021, Art. no. 102075.
- [8] D. Zhao, Z. Chang, and S. Guo, "A multimodal fusion approach for image captioning," *Neurocomputing*, vol. 329, pp. 476–485, 2019.
- [9] N. Zakharov, H. Su, J. Zhu, and J. Gläscher, "Towards controllable image descriptions with semi-supervised vae," *J. Vis. Commun. Image Representation*, vol. 63, 2019, Art. no. 102574.
- [10] S. Yang, J. Niu, J. Wu, Y. Wang, X. Liu, and Q. Li, "Automatic ultrasound image report generation with adaptive multimodal attention mechanism," *Neurocomputing*, vol. 427, pp. 40–49, 2021.
- [11] A. E. Kavur, L. I. Kuncheva, and M. A. Selver, "Basic ensembles of vanilla-style deep learning models improve liver segmentation from ct images," 2020, *arXiv:2001.09647*.
- [12] N. Yu, X. Hu, B. Song, J. Yang, and J. Zhang, "Topic-oriented image captioning based on order-embedding," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2743–2754, Jun. 2019.
- [13] B. Wang, X. Zheng, B. Qu, and X. Lu, "Retrieval topic recurrent memory network for remote sensing image captioning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 256–270, 2020.
- [14] L. Zhou, Y. Zhang, Y.-G. Jiang, T. Zhang, and W. Fan, "Re-caption: Saliency-enhanced image captioning through two-phase learning," *IEEE Trans. Image Process.*, vol. 29, pp. 694–709, 2020.
- [15] W. Zhao, X. Wu, and J. Luo, "Cross-domain image captioning via cross-modal retrieval and model adaptation," *IEEE Trans. Image Process.*, vol. 30, pp. 1180–1192, 2021.
- [16] M. Yang et al., "Multitask learning for cross-domain image captioning," *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 1047–1061, Apr. 2019.
- [17] G. Hoxha, F. Melgani, and B. Demir, "Toward remote sensing image retrieval under a deep image captioning perspective," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4462–4475, 2020.
- [18] Y. Huang, J. Chen, W. Ouyang, W. Wan, and Y. Xue, "Image captioning with end-to-end attribute detection and subsequent attributes prediction," *IEEE Trans. Image Process.*, vol. 29, pp. 4013–4026, 2020.
- [19] M. Yang et al., "An ensemble of generation- and retrieval-based image captioning with dual generator generative adversarial network," *IEEE Trans. Image Process.*, vol. 29, pp. 9627–9640, 2020.
- [20] Z. Yuan, X. Li, and Q. Wang, "Exploring multi-level attention and semantic relationship for remote sensing image captioning," *IEEE Access*, vol. 8, pp. 2608–2620, 2020.
- [21] W. Huang, Q. Wang, and X. Li, "Denoising-based multiscale feature fusion for remote sensing image captioning," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 3, pp. 436–440, Mar. 2021.
- [22] F. Monay and D. Gatica-Perez, "Modeling semantic aspects for cross-media image indexing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1802–1817, Oct. 2007.
- [23] J. Yu, J. Li, Z. Yu, and Q. Huang, "Multimodal transformer with multi-view visual representation for image captioning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 12, pp. 4467–4480, Dec. 2020.
- [24] Y. Xian and Y. Tian, "Self-guiding multimodal LSTM—when we do not have a perfect training dataset for image captioning," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5241–5252, Nov. 2019.
- [25] S. Amirian, K. Rasheed, T. R. Taha, and H. R. Arabnia, "Automatic image and video caption generation with deep learning: A concise review and algorithmic overlap," *IEEE Access*, vol. 8, pp. 218386–218400, 2020.
- [26] X. Li et al., "COCO-CN for cross-lingual image tagging, captioning, and retrieval," *IEEE Trans. Multimedia*, vol. 21, no. 9, pp. 2347–2360, Sep. 2019.
- [27] H. Park, K. Kim, S. Park, and J.-W. Choi, "Medical image captioning model to convey more details: Methodological comparison of feature difference generation," *IEEE Access*, vol. 9, pp. 150560–150568, 2021.
- [28] C. Y. Li, X. Liang, Z. Hu, and E. P. Xing, "Knowledge-driven encode, retrieve, paraphrase for medical image report generation," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, pp. 6666–6673.
- [29] D. Hou, Z. Zhao, Y. Liu, F. Chang, and S. Hu, "Automatic report generation for chest x-ray images via adversarial reinforcement learning," *IEEE Access*, vol. 9, pp. 21236–21250, 2021.
- [30] F. Wang, X. Liang, L. Xu, and L. Lin, "Unifying relational sentence generation and retrieval for medical image report composition," *IEEE Trans. Cybern.*, vol. 52, no. 6, pp. 5015–5025, Jun. 2022.
- [31] Y. Li, X. Liang, Z. Hu, and E. P. Xing, "Hybrid retrieval-generation reinforced agent for medical image report generation," *Adv. Neural Inf. Process. Syst.*, vol. 31, 2018.

- [32] B. Jing, P. Xie, and E. Xing, "On the automatic generation of medical imaging reports," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 2577–2586.
- [33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
- [34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [35] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.
- [36] K. Siwek, S. Osowski, and R. Szupiluk, "Ensemble neural network approach for accurate load forecasting in a power system," *Int. J. Appl. Math. Comput. Sci.*, vol. 19, no. 2, pp. 303–315, 2009.
- [37] J. Islam and Y. Zhang, "Brain MRI analysis for alzheimer's disease diagnosis using an ensemble system of deep convolutional neural networks," *Brain Informat.*, vol. 5, no. 2, 2018, Art. no. 2.
- [38] H. Alshazly, C. Linse, E. Barth, and T. Martinetz, "Ensembles of deep learning models and transfer learning for ear recognition," *Sensors*, vol. 19, no. 19, 2019, Art. no. 4139.
- [39] D. Singh, V. Kumar, M. Kaur, M. Y. Jabarulla, and H.-N. Lee, "Screening of covid-19 suspected subjects using multi-crossover genetic algorithm based dense convolutional neural network," *IEEE Access*, vol. 9, pp. 142566–142580, 2021.
- [40] D. Singh, V. Kumar, V. Yadav, and M. Kaur, "Deep neural network-based screening model for covid-19-infected patients using chest x-ray images," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 35, no. 03, 2021, Art. no. 2151004.
- [41] D. Singh, V. Kumar, and M. Kaur, "Densely connected convolutional networks-based COVID-19 screening model," *Appl. Intell.*, vol. 51, no. 5, pp. 3044–3051, 2021.
- [42] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," 2014, *arXiv:1409.1259*.
- [43] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.
- [44] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 311–318.
- [45] L. Cui, G. Li, Z. Zhu, Z. Wen, N. Lu, and J. Lu, "A novel differential evolution algorithm with a self-adaptation parameter control method by differential evolution," *Soft Comput.*, vol. 22, no. 18, pp. 6171–6190, 2018.
- [46] U. L. of Medicine, "Open-i service of the national library of medicine," 2022. [Online]. Available: <https://openi.nlm.nih.gov>
- [47] A. E. Kavur et al., "Chaos challenge - combined (CT-MR) healthy abdominal organ segmentation," *Med. Image Anal.*, vol. 69, 2021, Art. no. 101950.
- [48] T. Toprak, B. Belenlioglu, B. Aydın, C. Guzelis, and M. A. Selver, "Conditional weighted ensemble of transferred models for camera based onboard pedestrian detection in railway driver support systems," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5041–5054, May 2020.
- [49] M. A. Selver, "A robotic system for warped stitching based compressive strength prediction of marbles," *IEEE Trans. Ind. Informat.*, vol. 16, no. 11, pp. 6796–6805, Nov. 2020.
- [50] H. Park, K. Kim, J. Yoon, S. Park, and J. Choi, "Feature difference makes sense: A medical image captioning model exploiting feature difference and tag information," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics: Student Res. Workshop*, 2020, pp. 95–102.

Received 3 March 2022; revised 4 April 2022; accepted 22 April 2022.
Date of publication 28 April 2022; date of current version 12 May 2022.

Digital Object Identifier 10.1109/JTEHM.2022.3171078

Artificial Intelligence-Based Cyber-Physical System for Severity Classification of Chikungunya Disease

DILBAG SINGH¹, (Senior Member, IEEE), MANJIT KAUR¹, (Member, IEEE),
VIJAY KUMAR², MOHAMED YASEEN JABARULLA¹, (Member, IEEE),
AND HEUNG-NO LEE¹, (Senior Member, IEEE)

¹School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea

²Department of Computer Science and Engineering, NIT Hamirpur, Hamirpur 177005, India

CORRESPONDING AUTHOR: HEUNG-NO LEE (heungno@gist.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIP) under Grant NRF-2021R1A2B5B03002118; and in part by the Ministry of Science and ICT (MSIT), South Korea, through the Information Technology Research Center (ITRC) support program supervised by the Institute of Information and Communications Technology Planning and Evaluation (IITP) under Grant IITP-2021-0-01835.

ABSTRACT Background: Artificial intelligence techniques are widely used in solving medical problems. Recently, researchers have used various deep learning techniques for the severity classification of Chikungunya disease. But these techniques suffer from overfitting and hyper-parameters tuning problems. Methods: In this paper, an artificial intelligence-based cyber-physical system (CPS) is proposed for the severity classification of Chikungunya disease. In CPS system, the physical components are integrated with computational algorithms to provide better results. Random forest (RF) is used to design the severity classification model for Chikungunya disease. However, RF suffers from overfitting and poor computational speed problems due to complex architectures and large amounts of connection weights. Therefore, an evolving RF model is proposed using the adaptive crossover-based genetic algorithm (ACGA). Results: ACGA can efficiently optimize the architecture of RF to achieve better results with better computational speed. Extensive experiments are performed by utilizing the Chikungunya disease dataset. Conclusion: Performance analysis demonstrates that ACGA-RF achieves higher performance as compared to the competitive models in terms of F-measure, accuracy, sensitivity, and specificity. The proposed CPS system can prevent users from visiting hospitals and can render services to patients living far away from hospitals.

INDEX TERMS Artificial intelligence, cyber-physical system, automated diagnosis, Chikungunya disease, random forest, adaptive crossover, genetic algorithm, severity classification.

Clinical translation statement: The proposed model can be utilized for the severity classification of Chikungunya disease. The research findings are impactful as the proposed model can prevent users from visiting hospitals and can render services to patients living far away from hospitals.

I. INTRODUCTION

The Healthcare system plays a vital role in the development of any nation. Government is responsible to design proper healthcare policies to protect their citizens from any outbreak of diseases [1]. Hence, the outbreak of any new disease such as coronavirus is a major challenge for the healthcare system. Several viruses exist that can affect both animals and human beings. Chikungunya is one of the viruses that can be spread very rapidly and may create a big problem for the health system. Two basic types of infected mosquitoes

i.e., *Aedes albopictis* and *Aedes aegypti* transmit this virus in the human body [2]. The symptoms of Chikungunya are joint pain, sudden high fever, and rash. Some infected persons have headaches, fatigue, digestive complaints, and conjunctivitis [3]. The symptoms of Chikungunya are very similar to dengue because the same mosquito carries both viruses [4]. But in Chikungunya, joint pain is more severe as well as redness of the eyes. The symptoms of sore throat are different from dengue infection. Chikungunya may not cause death. As per literature, it is found that the patient

recovers within a week of this disease [5]. But, the joint pain may last for a few months. The doctors start the treatment by perceiving the symptoms of patients. However, the exact measurement of these symptoms is not possible. Therefore, the treatment of patients may not be effective.

Instead of symptoms found in the patients, reverse transcription-polymerase chain reaction (RT-PCR) and serological tests are used to diagnose Chikungunya. Both tests require blood samples of patients [6]. However, these tests are unable to provide reliable performance for this disease. Supervised learning techniques such as machine learning and deep learning can be used to evaluate the severity of this disease by considering the symptoms of patients and laboratory tests [7]. The severity classification of Chikungunya infected persons is still an ill-posed problem.

In [8], a fog-based framework for Chikungunya disease diagnosis was designed. J48 was utilized to classify Chikungunya infected patients. In [9], wearable internet of things (IoT) and fog-based framework for classification and controlling the Chikungunya disease was proposed. Fuzzy-C means (FCM) classifier was utilized for Chikungunya classification. But J48 [8] and FCM [9] suffer from over-fitting and hyper-parameters tuning problems. In [10], a particle swarm optimization-based ANFIS (PANFIS) model was implemented for the diagnosis of Chikungunya disease. Initially, an adaptive neuro-fuzzy inference system (ANFIS) classifier was used to classify the infected patients. Thereafter, particle swarm optimization (PSO) was utilized to overcome the parameter tuning problem with ANFIS. It achieved remarkable results compared to artificial neural networks (ANN). But, PANFIS [10], [11] suffers from the over-fitting problem. Also, sometimes PSO may be stuck in local optima and suffers from premature convergence problems [12], [13].

Therefore, to overcome the over-fitting and hyper-parameters tuning problems, an efficient evolving Random forest (RF) model is proposed for the severity classification of Chikungunya disease. The main contributions of this paper are as follows:

- 1) A cyber-physical system (CPS) based severity classification model is proposed for Chikungunya disease. In CPS system, the physical components are integrated with computational algorithms to attain better results.
- 2) Evolving RF model is proposed for severity classification of Chikungunya disease. An adaptive crossover-based genetic algorithm (ACGA) is utilized to evolve RF model.
- 3) Deep learning model is also implemented and compared with ACGA-RF for severity classification of Chikungunya disease.

The remainder of this paper is organized as follows. Section II presents the related work. The proposed ACGA-RF model is described in Section III. Performance analyses are presented in Section IV. The concluding remarks are discussed in Section V.

II. RELATED WORK

Artificial intelligence techniques are widely used in solving medical problems. Recently, researchers used machine learning techniques for the classification of Chikungunya disease. Hossain *et al.* [5] utilized the different symptoms of patients for the accurate assessment of Chikungunya disease. Their proposed framework collected the data from the interviews of patients. They used a belief-based rule system for predicting the level of Chikungunya. Their model attained an accuracy of 92%. Yang [14] developed a decision system for the diagnosis of Chikungunya disease. The neural network was used for classification by considering the uncertainty of the disease's symptoms. However, the uncertainty of some symptoms is not considered in this approach. Ganesan *et al.* [15] presented three different models to diagnose the Chikungunya disease. However, these models require human intervention for the assessment of this disease. Caicedo-Torres *et al.* [16] proposed a machine learning-based classifier for differentiating the dengue and Chikungunya patients. Their classifier was tested on 447 patients. The logistic regression model outperformed the other models. The accuracy obtained from logistic regression was 87%.

Ibrahim *et al.* [17] presented the backpropagation method for predicting the epidemic disease. They used epidemic disease factors for prediction. Thereafter, these factors were applied to the clustering technique. Their method is capable to identify the epidemic disease using feature classification. Coelho *et al.* [18] used a transfer learning model for predicting mosquito-borne diseases. They used time-series data from two Brazilian cities. Both the long short-term memory neural network model and random quantile forest model provided the same prediction performance. Caicedo-Torres *et al.* [19] utilized the machine learning techniques for envisaging the morbidity of Chikungunya in Colombia. Kernel ridge regression was used for forecasting the Chikungunya cases. Cross-validation and mean absolute error were used.

Sippy *et al.* [20] developed two prediction models on the Machala dataset. The first model namely the severity index for suspected arbovirus model (SISA) that utilized demographic data. Another model namely the severity index for suspected arbovirus (SISA1) with laboratory utilized the laboratory data. The accuracies obtained from SISA and SISA1 were 91% and 95%, respectively. Both models are capable to envisage arbovirus hospitalization.

Shimpi *et al.* [21] used a backpropagation algorithm to predict the Chikungunya disease. Five gradient-based optimization techniques were used. The pre-processed features were applied to the backpropagation algorithm for classification. The accuracy obtained from this model was 95%. However, a small dataset was used for validation purposes. Eng *et al.* [22] used machine learning techniques for predicting the binding affinity of T-cell epitopes of Chikungunya. They built prediction models for identifying binders and non-binder. This model will be helpful for vaccine development.

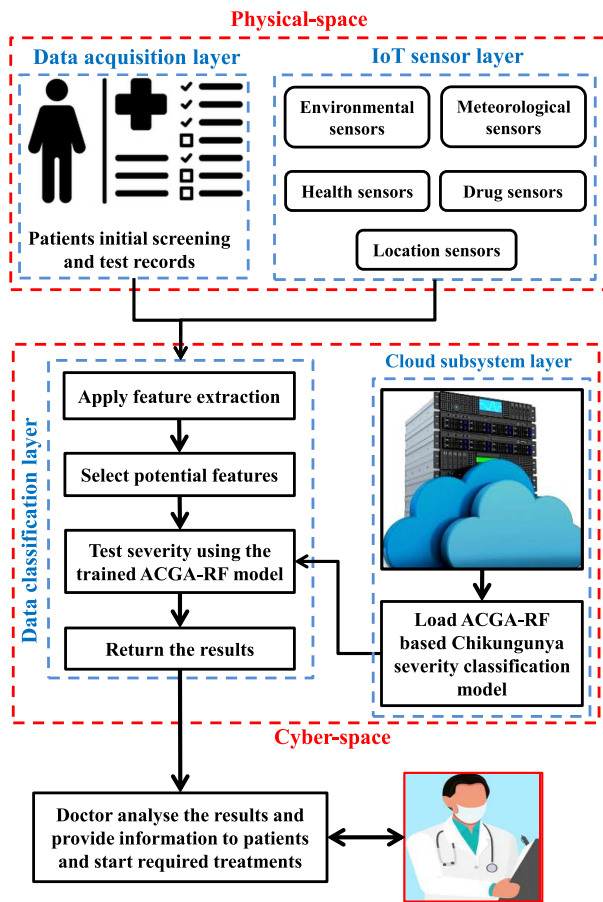


FIGURE 1. Proposed ACGA-RF model based Chikungunya severity classification model.

From the existing literature, it is found that the majority of the existing models suffer from hyper-parameters tuning, (i.e., optimization of initial control parameters), over-fitting and data insensitivity problems [23], [24]. Hence, there is a need to develop an efficient model for severity classification for Chikungunya disease.

III. PROPOSED FRAMEWORK

Motivated from [8], [10], an efficient model is proposed in this paper for diagnosis of Chikungunya disease. Figure 1 shows the proposed artificial intelligence-based cyber-physical system (CPS) for the diagnosis of Chikungunya disease. There are two main components in the proposed model, i.e., physical space and cyberspace. In physical space, the data related to users’ health is collected and forwarded to cyberspace for predicting the severity of the Chikungunya virus [25], [26]. In cyberspace, there are two sub-layers, i.e., the cloud subsystem layer and data classification layer [27], [28]. In the cloud subsystem layer, data and the proposed trained model are stored. At the data classification layer, the severity prediction of the Chikungunya virus is done in real-time by applying the proposed ACGA-RF model after extracting the potential features. Finally, doctors are involved for further assessment of the results.

A. PHYSICAL SPACE

In physical space, there is a data acquisition layer that collects users’ personal data, symptoms related to Chikungunya, and records of initial screening with the help of sensors.

The Chikungunya dataset is obtained from [8] and [9]. Sood and Mahajan [8] and [9] built the dataset by taking the symptoms-based dataset from [29] that contains eleven health features, i.e., abdominal pain, muscle pain, bleeding disorder, fatigue, eyes pain, itching, nausea, sore throat, joint pain, skin rash, and fever, of 2367 patients. Dataset with 5032 cases comprising environmental variables was taken from [30] and [31]. Monthly climate variables, i.e., rainfall, temperature, and humidity, were taken from [32]–[34]. The symptoms-based data was combined with climate and environmental features to validate ACGA-RF (for further details refer to [8] and [9]).

TABLE 1. Data attributes related to Chikungunya virus.

Attribute type	Attribute Name	Sensors used
User Personal data	UID (unique identification), Name, Age, Gender, Location of user workplace, Location of user residence, and Mobile number	GPS sensors
Health-related data	Joint pain, Muscle Pain, Rashes on the skin, Appetite loss, Fever, Redness in the eyes, Headache, nausea, sore throat, fatigue, vomiting	Biosensors
Environmental and location data	Water quality, air temperature, humidity, mosquito density, rainfall, mosquito breeding sites, level of carbon dioxide	Mosquito and climatic sensors

Table 1 shows the common features of Chikungunya virus. It contains several sensing devices to observe the health of patients. The observed information is then transferred to the data classification layer to classify the health of a particular client. The description of the dataset is mentioned in Table 1 (refer [8], [10]).

- 1) User Personal Data: This comprises the user’s personal data. GPS sensors are used to collect data such as UID, name, age, address, and mobile number.
- 2) Health-related Data: This set consists of vital signs. It provides information related to rashes on the skin, muscle pain, appetite loss, joint pain, fever, redness in the eyes, headache, nausea, sore throat, fatigue, and vomiting. Biosensors are used to collect this information.
- 3) Environmental and location Information: It provides the position of patients, susceptible users, uninfected users, and positions of mosquito dense areas, water quality, humidity, and mosquito breeding sites. These positions are evaluated by Global Positioning System (GPS) sensors to obtain the travel history of every patient. Radio Frequency Identification (RFID) tags and mosquito sensors are also utilized to store the proximity interactions between infected/uninfected/susceptible users, mosquito densities, and breeding sites.

Table 2 depicts the personal-parameters of registered users. It presents a brief description of the user’s personal parameters such as name, age, gender, address, mobile number, and contact details of guardians (refer [8], [10]). Table 3 shows

TABLE 2. Chikungunya's user personal-parameters.

Sr. No.	Parameters	Explanation
1.	Name	Patient name
2.	Age	Age of user in years
3.	Gender	Male or Female
4.	GPS	Geographic location of user
5.	Address	Address of the registered patient
6.	Telephone No.	Mobile number
7.	Patient Guardian's	Contact details of guardians

TABLE 3. Chikungunya symptoms related-dataset.

Sr. No.	Attributes	Decision	Description
1	Headache	Binary	Whether a user has a headache
2	Exposure to risky area	Binary	Whether a user works or lives in risk area
3	Nausea	Binary	User is feeling tired and sleepy
4	Fatigue	Binary	User feels weak and loss weight
5	Rash and Vomiting	Binary	Itching problem and has body marks
6	Muscular Pain	Binary	User suffers from muscular pain
7	Joints Swelling	Binary	User suffers from swelling
8	Joint pain	Binary	User suffers from joint pain symptom
9	Fever	Integer	Temperature in °C
10	Initiation post infection	Yes	3 to 7 days

Chikungunya symptoms such as headache, exposure to the risky area, nausea, fever, rash, vomiting, etc.

B. CYBER SPACE

Cyberspace is comprised of two sub-layers, i.e., the cloud subsystem layer and the data classification layer. The detail of registered users and the proposed trained model is stored in cloud subsystem layers. Whereas, at the data classification layer, diagnosis of the Chikungunya virus is done by applying ACGA-RF to the potential features. Also, RF-based model is used as it has the abilities of generalization, learning, fault tolerance, and adoption. However, RF model is sensitive to its control parameters. An efficient tuning of these control parameters can improve the performance of RF. Therefore, to automatically optimize control parameters of RF, an ACGA-RF model is proposed. Finally, the predicted severity report of a patient is transmitted to doctors for further treatment.

IV. PROPOSED EVOLVING RANDOM FOREST MODEL

This section discusses the proposed evolving random forest model. Initially, RF model is briefly discussed to understand its basic notations and hyper-parameters. Random forest (RF) is an ensemble of classification/regression trees [35], where every tree shows a mapping from feature space to the response. Trees can be obtained either using a subsampled data set of actual data or bootstrapped. Every tree is conditionally independent of one another. However, RF model is sensitive to its architecture and hyper-parameters. Thus, an adaptive crossover genetic algorithm (ACGA) is utilized to obtain the optimized architecture and hyper-parameters of RF.

In literature, genetic algorithm is widely accepted to optimize various classifiers such as deep learning models [23], [36]. Genetic algorithms utilize crossover and mutation operators during evolution phases to obtain the final solution. It has been found that the selection of efficient crossover and mutation operators is a challenging problem. To efficiently select crossover operator(s), Xue *et al.* [37] proposed an adaptive crossover genetic algorithm (ACGA). A group of crossover operators was utilized during the evolution process. Based on the performance of crossover operators, roulette wheel selection was utilized to select a specific crossover. In this paper, ACGA [37] is utilized to form evolving RF model. Three different crossover operators are used. The working of the proposed ACGA-RF is depicted in Algorithm 1.

init_P(): Population P is created using normal distribution by creating M vectors. Each vector represents the architecture and hyper-parameters-related values of RF.

init_A(): Adaptive crossover selector (A_S) is utilized by assigning probabilities to each crossover as $1/C$. C shows a number of utilized crossovers.

init_R(): Given crossover is selected according to roulette wheel selection and probabilities obtained from A_S .

init_C(): Apply selected crossover on parents' offspring to form child offsprings.

init_M(): Mutation operator is utilized to obtain child offspring. Compute child offsprings are saved in the offspring population (P_δ).

init_CA(): Dominated offsprings are then evaluated by using the actual and child offsprings. The respective outcomes are saved in rW and pL . P_δ is computed during $M/2^{th}$ step.

init_S(): Crowded distance [38] and Non-dominated sorting [39] are used to obtain M solutions from $R (P \cup P_\delta)$.

init_D(): To allocate reward/ penalty to selected offsprings, dominance comparison is utilized.

Penalty and reward of offsprings is saved in $nP_{IT \times C}$ and $nR_{IT \times C}$, respectively. After IT number of phases, A_S is updated by considering $nP_{IT \times C}$ and $nR_{IT \times C}$. All steps are repeated until the termination criterion (i.e., FE) is satisfied.

The succeeding subsections discuss the steps of ACGA.

A. FITNESS FUNCTION

Fitness function is designed to optimize RF by using sensitivity and specificity. It is defined as:

$$\max F(X) = \{f_1(X) \text{ and } f_2(X)\} \quad (1)$$

Here, X is an offspring. f_1 and f_2 represent the sensitivity and specificity parameters, respectively.

B. CROSSOVER OPERATORS

Three different crossover operators, i.e., single-point [38], chaotic crossover [40], and reduced surrogate [41], [42] are utilized. Single-point [38] has significant results to solve many computationally hard problems. It has shown better

Algorithm 1 ACGA Based RF

Input: Max population (M), A_S , iterative threshold (IT), No. of crossovers (C), and No. of fitness evaluations (FE)

Output: Optimized values for RF

```

1:  $P \leftarrow \text{init\_P}(M)$ 
2: Initialize  $rW$ ,  $pL$ ,  $nR_{IT \times C}$ , and  $nP_{IT \times C}$ .
3:  $\hat{P} = p_1, p_2, \dots, p_C \leftarrow \text{init\_A}(C)$ 
4:  $nFE \leftarrow 0$  and  $k \leftarrow 0$ 
5:  $P_\delta \leftarrow \phi$ 
6: while  $nFE < FE$  do
7:   for  $i = 1$  to  $M/2$  do
8:      $a_v \leftarrow \text{init\_R}(\hat{P})$ 
9:     Two offsprings are selected as parents:  $O_p$ 
10:     $O_c \leftarrow \text{init\_C}(O_p, a_v)$ 
11:     $O_c \leftarrow \text{init\_M}(O_c)$ 
12:     $nFE \leftarrow nFE + 2$ 
13:     $[rW, pL] \leftarrow \text{init\_CA}(O_p, O_c)$ 
14:    Add  $O_c$  to  $P_\delta$ 
15:  end for
16:   $k \leftarrow k + 1$ 
17:  Update  $rW$  to  $k^{\text{th}}$  row of  $nR_{IT \times C}$ 
18:  Update  $pL$  to  $k^{\text{th}}$  row of  $nP_{IT \times C}$ 
19:  if  $k = IT$  then
20:     $\hat{P} \leftarrow \text{init\_A}(nR_{IT \times C}, nP_{IT \times C})$ 
21:     $k = 0$ 
22:  end if
23:   $R \leftarrow P \cup P_\delta$ 
24:   $P \leftarrow \text{init\_S}(R)$ 
25:  Assign non-dominated offsprings in  $P$  to  $T_p$ 
26:  Optimized values  $\leftarrow T_p$ 
27: end while
28: return Optimized values

```

computational speed compared to the existing crossover operators [37].

Reduced surrogate [41], [42] can avoid unnecessary crossover operations when parents have similar offspring. Initially, it evaluates parents and forms a group of crossover points where both parents have different genes. In the absence of such a crossover point, no crossover operator is implemented. Chaotic crossover [40] can obtain a better converged and distributed group of Pareto-optimal offspring.

C. PENALTY AND REWARD

Penalties and rewards are allocated to offsprings by utilizing two matrices namely pL and rW as:

$$rW = [0 \dots 0]_{1 \times C} \quad (2)$$

$$pL = [0 \dots 0]_{1 \times C} \quad (3)$$

Pareto optima among the offsprings is used to modify rW and pL .

Algorithm 2 Credit Allocation (init_CA ())

Input: Parents (ρ), Children (δ), crossover selected using init_A (q)

Output: rW , pL [n_d , d_s] $\leftarrow \text{init_D}(\rho)$ // d_s and n_d define dominated and non-dominated offsprings, respectively.

```

1: Dominated parent (assume  $\rho_1 < \rho_2$ ).
2: if  $d_s \neq \phi$  then
3:   for  $i = 1$  to 2 do
4:     if  $\rho_1 < \delta_i$  then
5:        $pL_q \leftarrow pL_q + 1$ 
6:     else
7:        $rW_q \leftarrow rWrW_q + 1$ 
8:     end if
9:   end for
10: else
11:   // If parent is non-dominated.
12:   for  $i = 1$  to 2 do
13:     if  $\rho_1 \not< \delta_i$   $\rho_2 \not< \delta_i$  then
14:        $rW_q \leftarrow rWrW_q + 1$ 
15:     else
16:        $pL_q \leftarrow pL_q + 1$ 
17:     end if
18:   end for
19: end if
20:  $P \leftarrow \text{init\_P}(M)$ 
21: return Optimized values for RF

```

1) PARENT IS NON-DOMINATED

Pareto optima is evaluated between child and respective parent offsprings. If child offsprings are dominated by parents, then append pL_q by 1, otherwise append rW_q by 1. The pseudocode of updation of penalty and reward is depicted in Algorithm 2.

2) DOMINATED PARENT

If parent 1 (ρ_1) is dominated by parent 2 (ρ_2), then Pareto optima of each child offspring is compared with ρ_2 . If child offspring is dominated by ρ_2 , then append pL_q by 1. Otherwise, update rW_q by 1.

D. UPDATION OF ADAPTIVE CROSSOVER SELECTOR

A_S is utilized to update the crossover selection probabilities. During evolution process, it is implemented after every IT steps (refer [43]). Two matrices, i.e., $nP_{IT \times C}$ and $nR_{IT \times C}$ are used to hold the values of pL and rW , respectively. Recently updated IT 's pL and nR values are utilized to modify A_S . To evaluate the probability for q^{th} ($q = 1, 2, \dots, C$) crossover, addition of q^{th} column is utilized (refer [43]).

V. PERFORMANCE ANALYSIS

To analyze the efficiency of proposed model, health-related attributes are collected from [8], [34], [44], [45]. It mainly consists of attributes such as age, sex, location, fever, skin

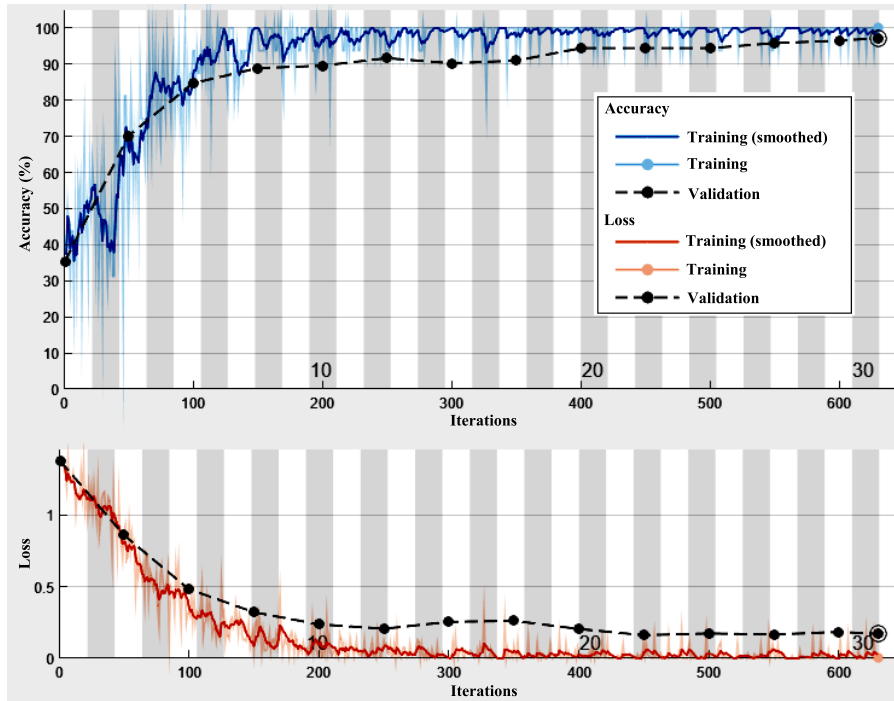


FIGURE 2. Accuracy and loss analysis of the deep learning-based severity classification of Chikungunya disease.

rashes, and joint pain. It is collected for approximately 10000 users. Around 1805 users are at no risk while 3890 are at normal risk and 4305 are at high risk. The acquired data is stored in cloud storage and is used for recognition by an optimized RF model. ACGA-RF is compared with J48, SVM, ANN, RF, adaptive neuro-fuzzy inference system (ANFIS), PANFIS [10], and deep learning (DL).

To implement DL model, various layers (i.e., feature input layer, fully connected layer, batch normalization layer [46], [47], ReLU layer [46], [47], softmax layer [46], [47], and classification layer) are utilized. For normalizing the input data, Z-score normalization [48] is used. Minibatch size is set to be 8. Adam [49] a stochastic gradient descent optimizer is used to achieve the better convergence of DL model.

A. TRAINING ANALYSIS

Figure 2 shows the accuracy and loss analysis of the deep learning-based severity classification model for Chikungunya disease. It clearly shows that the deep learning achieves significantly better convergence speed. It achieves 98.3% validation accuracy, therefore, least affected by the impact of over-fitting as training accuracy is 100%.

The performance of RF achieves the best training and validation accuracy values are 99.5% and 97.6%, respectively. Although it shows remarkable results, the performance of the proposed approach is still far from the optimal results. It shows an over-fitting problem as there is a high difference between training and validation accuracy values. The performance of ACGA-RF model achieves the best training and

TABLE 4. Confusion matrix analysis among the proposed ACGA-RF, DL, GA-RF, PSO-RF, and RF models.

Model	RF	DL	GA-RF	PSO-RF	ACGA-RF
TP	203	206	207	207	209
TN	314	315	315	317	318
FP	7	4	3	3	1
FN	6	5	5	3	2
Accuracy	0.9754	0.9831	0.9837	0.9855	0.9943
Sensitivity	0.9712	0.9763	0.9732	0.9855	0.9905
Specificity	0.9781	0.9874	0.9690	0.9902	0.9968
F-measure	0.9747	0.9818	0.9829	0.9902	0.9936

validation accuracy values are 100.0% and 99.6%, respectively. Thus, ACGA-RF model achieves remarkable results than both deep learning and RF models. Also, ACGA-RF is least affected by the over-fitting issue as there is only a 0.4% difference between the training and validation accuracy which are 1.7 and 1.9 for deep learning and RF, respectively. Also, ACGA-RF shows an enhancement in validation accuracy over deep learning and RF as 1.3% and 2.0%, respectively.

B. CONFUSION MATRIX ANALYSIS

To evaluate the performance of ACGA-RF, confusion matrix analyses are also achieved. Table 4 depicts the confusion matrix analysis proposed ACGA-RF, RF, and deep learning (DL) models. It is found that RF achieves the testing accuracy of 97.54%. It is found that the deep learning (DL) model achieves the testing accuracy of 98.31%. DL shows an average improvement of 0.77% over RF model. It is found

that ACGA-RF achieves an accuracy of 99.43%. Thus, the proposed ACGA-RF achieved an average improvement of 1.89% and 1.12% over RF and deep learning-based severity classification models, respectively.

C. COMPARATIVE ANALYSIS

Boxplot and ANOVA are used for statistical analysis. The hypotheses for every performance measure can be defined as:

$$\begin{cases} H_0 & \mu M_1 = \mu M_2 = \dots = \mu M_7, \\ H_A & \text{Means are not equal.} \end{cases} \quad (4)$$

where μM_i shows various severity classification models for Chikungunya disease. M_7 shows the proposed ACGA-RF. H_0 and H_A define the null and alternate hypotheses, respectively. ANOVA table consists of various attributes such as a sum of squares (SS), degrees of freedom (df), the mean sum of squares (MS), F-statistics (F), and P-value (P). If P value of F is lesser than the level of significance, then we can reject H_0 and conclude that the models are significantly different from each other.

ANOVA analysis of Testing Accuracy					
Source	SS	Df	MS	F	Prob>F
Columns	0.00177	6	0.0003	43.25	$1.639e^{-20}$
Error	0.00043	63	0.00001		
Total	0.00212	69			

FIGURE 3. ANOVA analysis of severity classification models for Chikungunya disease in terms of testing accuracy.

ANOVA analysis of Testing Sensitivity					
Source	SS	df	MS	F	Prob>F
Columns	0.00207	6	0.00034	55.29	$3.057e^{-23}$
Error	0.00039	63	0.00001		
Total	0.00246	69			

FIGURE 4. ANOVA analysis of severity classification models for Chikungunya disease in terms of testing sensitivity.

ANOVA analysis of Testing Specificity					
Source	SS	Df	MS	F	Prob>F
Columns	0.00247	6	0.00041	43.55	$1.376e^{-20}$
Error	0.00059	63	0.00001		
Total	0.00306	69			

FIGURE 5. ANOVA analysis of severity classification models for Chikungunya disease in terms of testing specificity.

Figures 3 - 6 show that H_A is accepted for all the considered performance metrics as the evaluated p - values are lower than 0.01. Thus, the performance of different models is significantly different from each other. But, it is not possible

ANOVA analysis of Testing F-measure					
Source	SS	df	MS	F	Prob>F
Columns	0.00221	6	0.00037	49.16	$2.057e^{-21}$
Error	0.00043	63	0.00001		
Total	0.00272	69			

FIGURE 6. ANOVA analysis of severity classification models for Chikungunya disease in terms of testing F-measure.

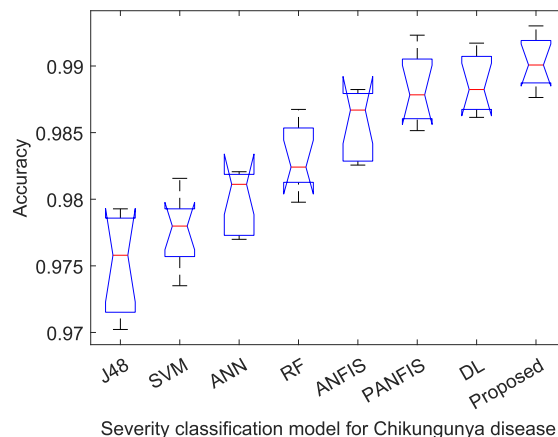


FIGURE 7. Accuracy analysis of ACGA-RF model.

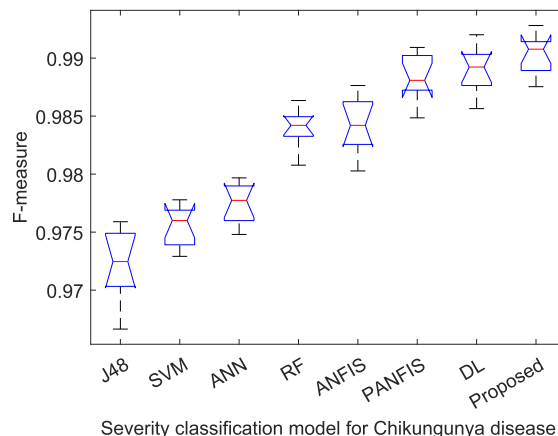


FIGURE 8. F-measure analysis of ACGA-RF model.

to find which technique outperforms the others. Thus, the boxplots are obtained to evaluate which technique performs significantly better than the others (see Figures 7 to 10).

Figure 7 demonstrates the accuracy analysis among ACGA-RF and the existing recognition Chikungunya disease recognition models. It is clearly shown that ACGA-RF achieves remarkably significant and consistent accuracy values. Compared to the existing Chikungunya disease recognition models, ACGA-RF achieves 1.3822% improvement in terms of accuracy.

Figure 8 shows F-measure analysis among ACGA-RF and the existing Chikungunya disease recognition models. ACGA-RF outperforms the existing Chikungunya disease recognition models by an average improvement of 1.4145%.

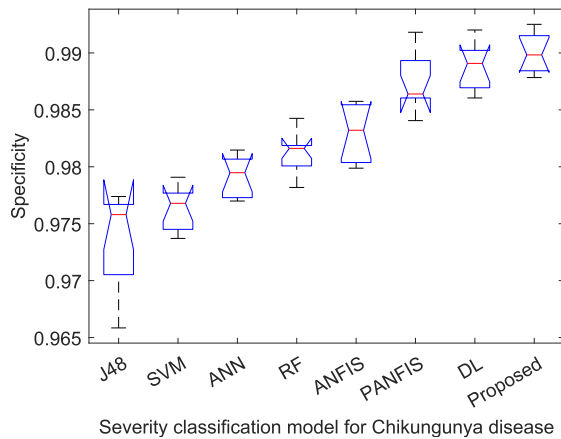


FIGURE 9. Specificity analysis of ACGA-RF model.

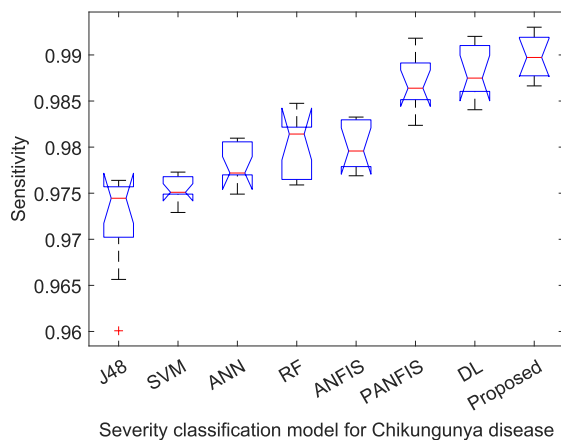


FIGURE 10. Sensitivity analysis of ACGA-RF model.

Figure 9 shows the specificity analysis among the existing models and the proposed Chikungunya disease recognition model. ACGA-RF achieves a 99.21% (median) value, which is significantly better than the competitive models by an average improvement of 1.3972%. Therefore, ACGA-RF can efficiently diagnose normal risk Chikungunya disease-infected patients. Figure 10 shows the sensitivity analysis among the existing models and proposed Chikungunya disease recognition model. ACGA-RF achieves 99.24% (median) value, which is significantly better than the competitive models by 1.4172%. Therefore, ACGA-RF can efficiently diagnose the high risk Chikungunya disease patients.

It is found that ACGA-RF takes on an average of 1 hour 36 minutes during the training process. During the testing process, it takes only 2.746 minutes to achieve the results. Additionally, ACGA-RF provides the testing results on an average of 1.674 seconds for a single patient. Therefore, ACGA-RF can be used for real-time applications.

D. DISCUSSION

Table 5 shows the comparison among the proposed model with state-of-the-art machine learning and deep learning models. It is found that in [8], J48 was utilized for severity

TABLE 5. Comparative analysis among the proposed ACGA-RF, DL, and state-of-the-art models.

Model	Accuracy	Sensitivity	Specificity	F-measure
J48 [8]	0.9278	0.935	0.883	0.827
FCM [9]	0.934	0.904	0.912	—
ANN [10]	0.9674	0.9438	0.9617	0.9328
PANFIS [10]	0.9871	0.9578	0.9787	0.9431
DL	0.9831	0.9763	0.9874	0.9818
Proposed ACGA-RF	0.9943	0.9905	0.9968	0.9936

classification of Chikungunya disease. It achieved the sensitivity, specificity, F-measure, and accuracy values as 0.935, 0.965, 0.827, and 92.7865, respectively. Fuzzy-C means (FCM) [9] achieved the sensitivity, specificity, and accuracy values as 0.867, 0.888, and 0.934, respectively. PANFIS [10] achieved sensitivity, specificity, F-measure, and accuracy values as 0.9578, 0.9787, 0.9431, and 0.9871, respectively. DL based Chikungunya diagnosis model achieved sensitivity, specificity, F-measure, and accuracy values as 0.9831, 0.9763, 0.9874, and 0.9818, respectively. Compared to these models, ACGA-RF achieved a sensitivity, specificity, F-measure, and accuracy values of 0.9943, 0.9905, 0.9968, and 0.9936, respectively. Therefore, the proposed ACGA-RF achieved significantly better results than the existing models in terms of accuracy, specificity, sensitivity, and F-measure by 1.3822%, 1.3972%, 1.4172%, and 1.4145%, respectively.

VI. CONCLUSION

In this paper, a cloud-based CPS is designed and implemented for the recognition of Chikungunya disease. The proposed system is divided into two main categories, i.e., physical space and cyberspace. Once, the data related to user-health are collected, it is stored in the cloud sub-system layer. An evolving RF model was proposed for the severity classification of Chikungunya disease by using ACGA. ACGA can efficiently optimize RF architecture to achieve better results with better computational speed. The comparative analysis demonstrates that ACGA-RF achieves significantly better testing performance than the existing models in terms of accuracy, specificity, sensitivity, and F-measure by 1.3822%, 1.3972%, 1.4172%, and 1.4145%, respectively. Thus, the proposed Chikungunya disease recognition model is beneficial for real-time medical applications.

In near future, the deep transfer learning models can be used to obtain more efficient results. Further novel meta-heuristic techniques can be designed to efficiently tune the deep learning architectures. Also, the proposed model can be applied to other kinds of datasets.

CONFLICT OF INTEREST

The authors declare that no conflict of interest.

REFERENCES

- [1] S. C. Weaver and M. Lecuit, "Chikungunya virus and the global spread of a mosquito-borne disease," *New England J. Med.*, vol. 372, no. 13, pp. 1231–1239, Mar. 2015.
- [2] Y. Lin, X. Lu, F. Fang, and J. Fan, "Personal health care monitoring and emergency response mechanisms," in *Proc. 1st Int. Symp. Future Inf. Commun. Technol. Ubiquitous HealthCare (Ubi-HealthTech)*, Jul. 2013, pp. 1–5.

- [3] B. Wahid, A. Ali, S. Rafique, and M. Idrees, "Global expansion of chikungunya virus: Mapping the 64-year history," *Int. J. Infectious Diseases*, vol. 58, pp. 69–76, May 2017.
- [4] M. Kaur et al., "Coinfection of chikungunya and dengue viruses: A serological study from north western region of Punjab, India," *J. Lab. Physicians*, vol. 10, no. 4, pp. 443–447, Oct. 2018.
- [5] M. S. Hossain, Z. Sultana, L. Nahar, and K. Andersson, "An intelligent system to diagnose chikungunya under uncertainty," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl.*, vol. 10, no. 2, pp. 37–54, 2019.
- [6] V. Lakshmi et al., "Clinical features and molecular diagnosis of chikungunya fever from south India," *Clin. Infectious Diseases*, vol. 46, no. 9, pp. 1436–1442, May 2008.
- [7] S. K. Mardekian and A. L. Roberts, "Diagnostic options and challenges for dengue and chikungunya viruses," *BioMed Res. Int.*, vol. 2015, pp. 1–8, Oct. 2015.
- [8] S. K. Sood and I. Mahajan, "A fog-based healthcare framework for chikungunya," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 794–801, Oct. 2018.
- [9] S. K. Sood and I. Mahajan, "Wearable IoT sensor based healthcare system for identifying and controlling chikungunya virus," *Comput. Ind.*, vol. 91, pp. 33–44, Oct. 2017.
- [10] S. Kaur and K. K. Chahal, "Prediction of chikungunya disease using PSO-based adaptive neuro-fuzzy inference system model," *Int. J. Comput. Appl.*, pp. 1–9, Jan. 2021.
- [11] A. Rezaeipannah and M. Mojarad, "Modeling the scheduling problem in cellular manufacturing systems using genetic algorithm as an efficient meta-heuristic approach," *J. Artif. Intell. Technol.*, vol. 1, pp. 228–234, Oct. 2021.
- [12] G. D. Singh, M. Prateek, S. Kumar, M. Verma, D. Singh, and H.-N. Lee, "Hybrid genetic firefly algorithm-based routing protocol for VANETs," *IEEE Access*, vol. 10, pp. 9142–9151, 2022.
- [13] S. Ndichu, S. Kim, and S. Ozawa, "Deobfuscation, unpacking, and decoding of obfuscated malicious Javascript for machine learning models detection performance improvement," *CAAI Trans. Intell. Technol.*, vol. 5, no. 3, pp. 184–192, Sep. 2020.
- [14] J. B. Yang, "Rule and utility based evidential reasoning approach for multiattribute decision analysis under uncertainties," *Eur. J. Oper. Res.*, vol. 131, no. 1, pp. 31–61, 2001.
- [15] V. Ganesan, B. Duan, and S. Reid, "Chikungunya virus: Pathophysiology, mechanism, and modeling," *Viruses*, vol. 9, no. 12, p. 368, Dec. 2017.
- [16] W. Caicedo-Torres, A. Paternina-Caicedo, H. Pinzón-Redondo, and J. Gutiérrez, "Differential diagnosis of dengue and chikungunya in Colombian children using machine learning," in *Proc. Ibero-Amer. Conf. Artif. Intell.*, Cham, Switzerland: Springer, 2018, pp. 181–192.
- [17] N. Ibrahim, N. S. M. Akhir, and F. H. Hassan, "Predictive analysis effectiveness in determining the epidemic disease infected area," in *Proc. AIP Conf.*, 2017, Art. no. 020064.
- [18] F. C. Coelho, N. L. D. Holanda, and B. M. Coimbra, "Transfer learning applied to the forecast of mosquito-borne diseases," *MedRxiv*, to be published, doi: 10.1101/2020.02.03.20020164.
- [19] W. Caicedo-Torres, D. Montes-Grajales, W. Miranda-Castro, M. Fennix-Agudelo, and N. Agudelo-Herrera, "Kernel-based machine learning models for the prediction of dengue and chikungunya morbidity in Colombia," in *Proc. Colombian Conf. Comput.*, Cham, Switzerland: Springer, 2017, pp. 472–484.
- [20] R. Sippy et al., "Severity index for suspected arbovirus (SISA): Machine learning for accurate prediction of hospitalization in subjects suspected of arboviral infection," *PLoS Neglected Tropical Diseases*, vol. 14, no. 2, 2020, Art. no. e0007969.
- [21] P. Shimpi, S. Shah, M. Shroff, and A. Godbole, "An artificial neural network approach for classification of vector-borne diseases," in *Proc. Int. Conf. Comput. Methodolog. Commun. (ICCMC)*, Jul. 2017, pp. 412–415.
- [22] C. L. P. Eng, T. W. Tan, and J. C. Tong, "T-cell epitope prediction of chikungunya virus," in *Chikungunya Virus*. Cham, Switzerland: Springer, 2016, pp. 201–207.
- [23] D. Singh, V. Kumar, M. Kaur, M. Y. Jabarulla, and H.-N. Lee, "Screening of COVID-19 suspected subjects using multi-crossover genetic algorithm based dense convolutional neural network," *IEEE Access*, vol. 9, pp. 142566–142580, 2021.
- [24] X. Zhang and G. Wang, "Stud pose detection based on photometric stereo and lightweight yolov4," *J. Artif. Intell. Technol.*, vol. 2, no. 1, pp. 32–37, 2022.
- [25] Y. Zhang, M. Qiu, C.-W. Tsai, M. M. Hassan, and A. Alamri, "Health-CPS: Healthcare cyber-physical system assisted by cloud and big data," *IEEE Syst. J.*, vol. 11, no. 1, pp. 88–95, Mar. 2017.
- [26] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE ICNN*, vol. 4, Nov./Dec. 1995, pp. 1942–1948.
- [27] M. Janalipour and A. Mohammadzadeh, "Building damage detection using object-based image analysis and ANFIS from high-resolution image (case study: BAM earthquake, Iran)," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 5, pp. 1937–1945, May 2016.
- [28] S. A. Hosseini and I. Esmaili Paen Afrakoti, "Evaluation of a new neutron energy spectrum unfolding code based on an adaptive neuro-fuzzy inference system (anfif)," *J. Radiat. Res.*, vol. 59, no. 4, pp. 436–441, 2017.
- [29] *Data Catalog. Now Cast Predictions for Local Transmission of Chikungunya Virus*. Accessed: Oct. 8, 2017. [Online]. Available: <https://catalog.data.gov/dataset?tags=chikungunya>
- [30] *DHS Program Demographic and Health Surveys*. Survey Dataset Files. Accessed: Oct. 8, 2017. [Online]. Available: https://dhsprogram.com/data/dataset/Ethiopia_StandardDHS_2016.cfm
- [31] *DHS Program Demographic and Health Surveys*. Survey Dataset Files. Accessed: Oct. 8, 2017. [Online]. Available: https://dhsprogram.com/data/dataset/Angola_StandardDHS_2015.cfm
- [32] *Data Catalog. Surface Air Temperature*. Accessed: Oct. 8, 2017. [Online]. Available: <https://data.gov.sg/dataset/surface-air-temperaturemonthly-mean>
- [33] *Un Data: A World of Information*. Relative Humidity. Accessed: Oct. 8, 2017. [Online]. Available: <http://data.un.org/Data.aspx?d=CLINO&f=ElementCode%3a11>
- [34] *Data Catalog. Monthly Total Rainfall*. Accessed: Oct. 8, 2017. [Online]. Available: <https://data.gov.sg/dataset/rainfall-monthly-total>
- [35] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [36] I. M. El-Hasnony, S. I. Barakat, and R. R. Mostafa, "Optimized ANFIS model using hybrid metaheuristic algorithms for Parkinson's disease prediction in IoT environment," *IEEE Access*, vol. 8, pp. 119252–119270, 2020.
- [37] Y. Xue, H. Zhu, J. Liang, and A. Slowik, "Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification," *Knowl.-Based Syst.*, vol. 227, Sep. 2021, Art. no. 107218.
- [38] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Aug. 2002.
- [39] M. Premkumar, P. Jangir, R. Sowmya, H. H. Alhelou, A. A. Heidari, and H. Chen, "MOSMA: Multi-objective slime mould algorithm based on elitist non-dominated sorting," *IEEE Access*, vol. 9, pp. 3229–3248, 2020.
- [40] L. dos Santos Coelho and P. Alotto, "Multiobjective electromagnetic optimization based on a nondominated sorting genetic approach with a chaotic crossover operator," *IEEE Trans. Magn.*, vol. 44, no. 6, pp. 1078–1081, Jun. 2008.
- [41] D. Whitley, "An executable model of a simple genetic algorithm," in *Foundations of Genetic Algorithms*, vol. 2. Amsterdam, The Netherlands: Elsevier, 1993, pp. 45–62.
- [42] H. Elmaraghy, V. Patel, and I. B. Abdallah, "Scheduling of manufacturing systems under dual-resource constraints using genetic algorithms," *J. Manuf. Syst.*, vol. 19, no. 3, pp. 186–201, 2000.
- [43] Y. Xue, B. Xue, and M. Zhang, "Self-adaptive particle swarm optimization for large-scale feature selection in classification," *ACM Trans. Knowl. Discovery From Data*, vol. 13, no. 5, pp. 1–27, Oct. 2019.
- [44] (Jun. 8, 2018). *Centers for Disease Control and Prevention*. Chikungunya Diseases. [Online]. Available: <https://www.cdc.gov/chikungunya/geo/default.html>
- [45] (May 16, 2018). *Pan American Health Organization*. Chikungunya Disease. [Online]. Available: <https://www.paho.org/data/index.php/en/mnu-topics/chikv-en.html>
- [46] R. Vecchi, S. Scardapane, D. Comminiello, and A. Uncini, "Compressing deep-quaternion neural networks with targeted regularisation," *CAAI Trans. Intell. Technol.*, vol. 5, no. 3, pp. 172–176, Sep. 2020.
- [47] Y. Yang, L. Juntao, and P. Lingling, "Multi-robot path planning based on a deep reinforcement learning DQN algorithm," *CAAI Trans. Intell. Technol.*, vol. 5, no. 3, pp. 177–183, 2020.
- [48] S. G. K. Patro and K. K. Sahu, "Normalization: A preprocessing stage," 2015, *arXiv:1503.06462*.
- [49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

• • •

Received 20 June 2022, accepted 4 July 2022, date of publication 15 July 2022, date of current version 25 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3191429

RESEARCH ARTICLE

DKNet: Deep Kuzushiji Characters Recognition Network

DILBAG SINGH¹, (Senior Member, IEEE), C. V. ARAVINDA²,
MANJIT KAUR¹, (Senior Member, IEEE), MENG LIN³, JYOTHI SHETTY²,
VIKRAM RAJU REDDICHERLA², AND HEUNG-NO LEE¹, (Senior Member, IEEE)

¹School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea

²N.M.A.M. Institute of Technology, Nitte, Karkala 574110, India

³Department of Electronic and Computer Engineering, The Graduate School of Science and Engineering, Ritsumeikan University, Kusatsu Shiga 525-8577, Japan

Corresponding author: Heung-No Lee (heungno@gist.ac.kr)

This work was supported in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant by the Korean Government through MSIT (Artificial Intelligence Graduate School Program—GIST), under Grant 2019-0-01842; and in part by the Ministry of Science and ICT (MSIT), South Korea, under the Information Technology Research Center (ITRC) Support Program supervised by the Institute of Information and Communications Technology Planning and Evaluation (IITP) under Grant IITP-2021-0-01835.

ABSTRACT Kuzushiji, a cursive writing style, had been extensively utilized in Japan for over a thousand years starting from the 8th century. In 1900, Kuzushiji was not included in regular school curricula due to the change in the Japanese writing system. Nowadays Japanese natives are unable to read historical books that were written using Kuzushiji language. Therefore, libraries and museums have decided to build digital copies of the documents and books that were written in Kuzushiji language. Due to a limited number of trained experts, researchers have utilized machine and deep learning models to convert historical documents and books into a modern script that can be easily read by human beings. However, the existing deep learning techniques suffer from over-fitting and gradient vanishing problems. To overcome these problems, an efficient deep Kuzushiji characters recognition network (DKNet) is proposed. Initially, to remove noise from the training images, a trilateral joint filter is applied. Contrast limited adaptive histogram equalization (CLAHE) is then applied to enhance the visibility of filtered images. Thereafter, a pre-trained MobileNet is utilized to extract the features of Kuzushiji characters. MobileNet's final layers are removed, including the fully connected layer and softmax. The flatten layer is then applied to the input. A fully connected classification layer is then used with Rectified linear units (ReLU) and dropouts. Dropouts are used to generalize the model, thus preventing the over-fitting problem. Finally, the softmax activation function is employed to provide the recognition results. To test the proposed model, actual documents are first segmented by using the proposed Maximally stable extremal regions (MSERs) and convexhull-based segmentation approach. Segmented characters are then recognized using the trained DKNet. Extensive comparative analyses reveal that DKNet achieves better performance than the competitive models in terms of various performance metrics. An efficient Application Programming Interface (API) is also designed for Japanese Kuzushiji ancient heritage character recognition to help the end-users.

INDEX TERMS Maximally stable extremal regions, deep learning models, Kuzushiji character recognition, MobileNet, InceptionNet, ResNet.

I. INTRODUCTION

Kuzushiji, a cursive writing style, had been extensively utilized in Japan for over a thousand years starting from

The associate editor coordinating the review of this manuscript and approving it for publication was Donato Impedovo¹.

the 8th century. But, in 1900, Kuzushiji was not included in regular school curricula due to the change in Japanese writing system. Nowadays Japanese natives are unable to read historical books that were written using Kuzushiji language. Therefore, libraries and museums have decided to build digital copies of the documents and books that were written

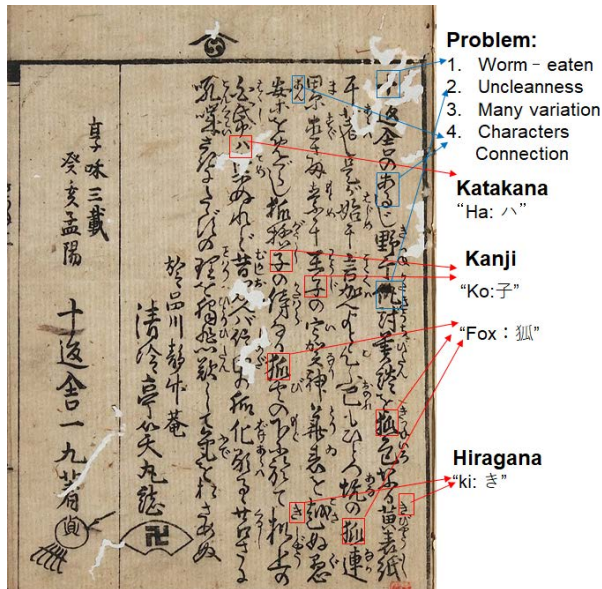


FIGURE 1. Typical page of a classical Japanese book.

in Kuzushiji language. But, due to a limited number of trained experts, researchers have utilized machine and deep learning models to convert historical documents and books into a modern script that can be easily read by human beings [1], [2].

To convert a single document or book page at a time, it is required to efficiently segment each character at a time [3]–[5]. In some cases, text detection becomes even more important than just text extraction until it can detect the shapes of regular or irregular shapes in the given images. Adding two Chinese syllabaries and a set of Chinese logograms into a complex logosyllabary system is the Japanese writing pattern, which is a fascinating study of tradition and innovation. The Japanese language uses incurred verbs and postpositions. It utilizes suffixes and particles to create words and clauses in sentences. Chinese characters (kanji) have been extensively utilized in a long history of use dating back centuries. The Japanese also used some Chinese characters with sound values for grammatical purposes. These characters were not instantly interpreted as phonetic or logographic representations of meaning [6], [7].

There were many variants of Kuzushiji characters and in some documents and books, these characters were connected with each other. Aging causes the documents to become unclean and some have been infected with worms [8]. Figure 1 shows a page of an early Japanese book that contains Kuzushiji characters.

- **Hiragana:** Manyoshu describes a system of writing native words in ancient Japan that uses manyogana. In Figure 1, Ki illustrates how these signs were reduced and simplified into sogana and finally to hiragana.
- **Katakana:** According to the Chinese Buddhist scriptures, the word kanji is an essential pronunciation aid. This character accumulated as grammatical suffixes,

particles, and postpositions. But the literal meaning of kanji did not change. In Figure 1, Ha indicates that katakana is widely used to write non-Chinese words.

- **Allophones:** In Figure 1, Ko and Fox illustrate what may appear to be allophones in the Japanese syllabograms. People speaking the same language perceive these sounds as similar sounds.

A. CHALLENGES

Figure 1 demonstrates challenges and problems associated with Kuzushiji character recognition.

- 1) **Modern and Kuzushiji characters:** Some Kuzushiji characters are written in such a way that only context, particularly the identity of preceding characters can be used to recognize them. A simple line, for example, might signify a variety of characters based on the characters used before it. KuroNet can disambiguate by using the surrounding characters since it employs both local and global contexts. However, for efficient classification of characters, it is required to segment each character prior to the recognition process.
- 2) **Variation:** Although Kuzushiji has a huge number of characters (e.g., used dataset comprises 4328), their distribution is long-tailed. A significant portion of the characters only appears once or twice in the sample. This is owing to the Japanese language's peculiar structure that has two sorts of character sets, i.e., a phonetic alphabet (with simple and extremely frequent characters) and non-phonetic Kanji characters. Kanji is made up of both common and uncommon characters, some are very intricate and others are merely one or two straight lines.
- 3) **Hentaigana characters:** One feature of classical Hiragana or Hentaigana (the word literally translates to “Character Variations”) that has a significant impact on recognition. Many characters that can be written exclusively in modern Japanese can be written in a variety of ways in pre-modern Japanese language [35].
- 4) **Aging processing:** It is hard to recognize the unclean characters due to infestation by worms.

Recently, researchers have utilized various deep learning models to improve the performance of Kuzushiji character recognition. Some of the popular models are as LeNet [16], AlexNet [17], GoogLeNet [18], VGG16, VGG19 [19], ResNet152V2 [21], InceptionNet [24], InceptionResNetV2 [20], XceptionNet [25], and MobileNet [26]. But most of these models suffer from over-fitting and gradient vanishing problems. Additionally, an efficient approach is required to segment the characters prior to the recognition process.

B. CONTRIBUTIONS

To overcome the aforementioned problems, an efficient deep Kuzushiji characters recognition network (DKNet) is proposed. The main contributions of this paper are as follows:

- To overcome the over-fitting and gradient vanishing problems, an efficient deep Kuzushiji characters recognition network (DKNet) is proposed.
- To remove noise from the training images, a trilateral joint filter is employed. Contrast limited adaptive histogram equalization (CLAHE) is also applied to enhance the visibility of filtered images.
- In DKNet, a MobileNet's final layers are removed. The flatten layer is then applied to the input. Thereafter, a fully connected classification layer is implemented with Rectified linear units (ReLU) and dropouts. Finally, the softmax activation function is employed to provide the recognition results.
- To test the proposed model, actual documents are first segmented by using the proposed Maximally stable extremal regions MSERs) and convexhull-based segmentation approach. Segmented characters are then recognized using the trained DKNet.
- An efficient Application Programming Interface (API) is also designed for Japanese Kuzushiji ancient heritage characters recognition to help the end-users.

The remaining paper is organized as follows. Section II discusses the literature review. Section III discusses the proposed DKNet. Section IV presents the proposed MSER and convexhull-based segmentation approach. Experimental results are presented in Section V. Section VI concludes the paper.

II. LITERATURE SURVEY

In [9], an adaptive neural network was utilized for character recognition by considering features of directional elements as feature vectors. In [11], line-by-line recognition was used to classify characters according to the attention mechanism. Lyu *et al.* [12] detected the text lines prior to the recognition process. Image processing operations and ARU-Net were used to identify text lines. It has achieved precision, recall, and F-score values as 95.2%, 98.3%, and 96.6%, respectively. Le *et al.* [13] designed a Recognition system using an attention-based encoder-decoder (RAED). It re-process data by detecting the line's start and recognizing each character until the line has been completed.

Li *et al.* [10] utilized three and five-cross point non-directed graphs of Oracle-bone inscriptions (OBIs) such as blocks and holes. Another recognition method was also used (refer to [14]). Li *et al.* [15] developed a DNA-based model for detection of OBIs. However, Meng *et al.* [5] and Liu *et al.* [28] proved that all OBIs needed to be cut in advance which is not convenient for practical applications. Meng *et al.* [5] extended the Single shot multibox detector (SSD) for OBI detection and achieved precision, recall, and F values as 98%, 83%, and 88%, respectively. Clanuwat *et al.* [40] recognized the Kuzushiji characters using a deep learning model. Residual U-Net architecture (RU-Net) was applied to recognize the entire page with identification of all characters given on it. In [41], Japanese

historical text was recognized by attention based encoder-decoder (AED) model. Multiscale features were extracted using a dense convolutional neural network. The target text was generated using Long short-term memory (LSTM) decoder.

In [45], handwritten Japanese text was recognized using Attention augmented convolutional recurrent network (AACRN). The performance of the model was evaluated on Kuzushiji and TUAT Kondate datasets. In [46], the Deep learning model(DLM) was applied to recognize the Kuzushiji characters. Ueki *et al.* [47] proposed a recognition system for the Kuzushiji characters by utilizing Multiple softmax outputs (MSOs). In [48], a 2-dimensional Context box proposal network (2DCbnp) was proposed to identify the Kuzushiji Characters. Features were extracted using VGG16 and then features were processed through a 2-D context. Thereafter, horizontal and vertical contexts were explored using Bidirectional LSTM (BLSTM).

Bing *et al.* [42] proposed a method to understand the ancient Japanese books using conventional image processing and ARU-Net. These methods were used for the detection of frames and segmentation of lines. Character recognition was done using by applying AlexNet. Nguyen *et al.* [43] used Gaussian mixture mode, LSTM, and CNN to recognize the Japanese kanji characters. Septiana *et al.* [44] proposed template matching algorithm to recognize the handwritten Japanese characters. It achieved the accuracy of 89.8%.

Horie *et al.* [6] recognized OBIs using deep learning models. While they achieved good accuracy, the experimental data were drawn from recent research, not the original rubbings. OBIs are a kind of hieroglyphs that were the original version of the modern Chinese characters that are widely used in China, Japan, and other Asian countries. One OBI character is one hieroglyph graph, with a meaning corresponding to one thing or one object. Lyu *et al.* [12] developed a Japanese historical character recognition system for the reading support system for historical documents by using the directional elements as feature vectors and designing a modular neural network as a pattern classification model.

Nguyen *et al.* [36] segmented the image into patches for specific characters and then classified each patch separately. This strategy is computationally better, but due to the contextual nature of many characters, it is unsuited for the overall Kuzushiji recognition job. Le *et al.* created datasets with separately segmented Kazushi characters but did not take into account the Kuzushiji recognition [13].

Unger [23] considered several characters that can be exclusively rewritten for current Japanese in different ways in pre-modern Japanese scripts. It seems to be one element of classical Hiragana or Hentaigana (meaning "Character Variations") that has a considerable impact on recognition. Japanese MNIST dataset [13] is significantly more difficult than the original MNIST dataset [16]. Because many pre-modern Japanese characters can be written in numerous ways. Therefore, there is a need to develop good models that

must be able to reflect the multi-modal distribution of each class.

III. DEEP KUZUSHIJI CHARACTERS RECOGNITION NETWORK (DKNet)

Deep neural networks can be made lightweight using pre-trained models. Pre-trained models can reduce the number of parameters significantly when compared to networks with regular convolutions. To train the proposed model, MobileNet is used as a transfer learning model. MobileNet was designed by Google and it is an open-sourced model. This section initially briefly discusses various pre-trained models. Finally, the proposed deep Kuzushiji characters recognition network (DKNet) is discussed.

A. PRE-TRAINED LEARNING MODELS

This section discusses various pre-trained learning models which were trained on the ImageNet dataset.

1) RESIDUAL NEURAL NETWORK (ResNet)

It was presented at ILSVRC 2015. This architecture utilized the concept of skip connections. It comes up with 152 layers that allow us to train very deep neural networks without suffering from the gradient vanishing problem. Skipping connections are also called gated-recurrent units and are similar to recurrent neural networks based on recent successes [11], [24].

2) INCEPTION

The main aim of this architecture is to reduce the computational workload of deep neural networks. Originally proposed by GoogleNet, this model architecture was later applied to both Inception-V2 and Inception-V3. It can manipulate similar inputs in parallel to achieve better convergence [21], [25].

3) MobileNet

MobileNet is a well-known pre-trained model. It is an open-sourced model developed by Google as shown in the Table 1. To develop a lightweight model, it utilizes depth-wise separable convolutions. It minimizes the number of attributes than the network with regular convolutions with similar depth. Thus, it is a lightweight deep learning model [23], [26]. Two different hyper-parameters are used that can efficiently trade-off between accuracy and latency.

4) XceptionNet

It is an extension of the inception model which can be viewed as an intermediate step between regular convolutions and depth-wise separable convolutions [22].

5) LeNet

It has five convolution layers. 32 images having 32×32 size for improving the classification performance. However, it has relatively few layers and simple architecture. LeNet cannot work properly for higher-resolution images.

TABLE 1. Mobile-net model architecture.

Type	Filter	Input
Conv/S2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv/dw/s1	$3 \times 3 \times 32dw$	$112 \times 112 \times 32$
Conv/s1	$1 \times 1 \times 32 \times 64dw$	$112 \times 112 \times 32$
Conv/dw/s2	$3 \times 3 \times 64dw$	$112 \times 112 \times 64$
Conv/s1	$1 \times 1 \times 128$	$56 \times 56 \times 64$
Conv/s1	$3 \times 3 \times 128dw$	$56 \times 56 \times 128$
Conv/s1	$1 \times 1 \times 128dw$	$56 \times 56 \times 128$
Conv/dw/s2	$3 \times 3 \times 128dw$	$56 \times 56 \times 128$
Conv/s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv/dw/s1	$3 \times 3 \times 256dw$	$28 \times 28 \times 256$
Conv/s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5 \times \text{Conv/s1/dw}$	$3 \times 3 \times 512$	$14 \times 14 \times 512$
Avgpool/s1	Pool 7×7	$7 \times 7 \times 1024$
FC/s1	1024×1000	$1 \times 1 \times 1024$
Softmax/s1	Classifier	$1 \times 1 \times 1000$

6) VGG

This was proposed by ILSVRC in 2014. It consists of 16 convolutional layers for uniform architecture. It is similar to AlexNet and has many filters and 138 million parameters. Hence, it has more computation power. Later, VGG-16 is extended to VGG-19 layers.

7) GoogLeNet

It achieved an error rate close to the human level in 2014. It consists of 22 layers with 224×224 sized images. The parameters are reduced from 60 million to 4 million [15].

8) InceptionNet

InceptionNet is created to reduce the computational burden of deep neural networks while maintaining performance. The idea is firstly proposed in GoogLeNet and then extended to Inception-V2, V3, and Inception-Res (Inception-V4). The key idea of InceptionNet is to compute multiple transformations over the same input map in parallel and provide the results into a single output. It has a compressed version of the spatial information that reduces the computation cost [16].

B. JOINT TRILATERAL FILTER

A joint trilateral filter is a well-known edge-preserving filter used to filter the Kuzushiji images. It considers a guided image G_{im} to filter the input image. Generally, the actual image I_m acts as a guided image. It can be implemented as:

$$J_{tf}(I_m) = \frac{\sum_{q \in \Omega} \rho^{pq}(G_{im}) \times I_q \times \sigma^2(I_q, G_{im})}{\sum_{q \in \Omega} \rho^{pq}(G_{im})} \quad (1)$$

Here, Ω is a kernel window at coordinate k dependent on the bilateral filter [37], [38].

A kernel weight, i.e., $\rho^{pq}(G_{im})$ can be computed as:

$$\rho^{pq}(G_{im}) = \frac{1}{|n|^2} \sum_{n:(p,q) \in \Omega} \left(1 + \frac{(G_{imp} - \mu_n)(G_{imq} - \mu_n)}{\sigma_n^2 + \epsilon} \right) \quad (2)$$

where $|n|$ shows pixels in window. μ_n and σ_n^2 are average and variance of G_{im} in local Ω . The weight associated to pixel q is maximum, if G_{imp} and G_{imq} are on similar sides of an edge, otherwise, a minimum weight is assigned to q .

C. CONTRAST LIMITED ADAPTIVE HISTOGRAM EQUALIZATION

The precision of Kuzushiji recognition is generally low if the images have poor visibility. The most of the enhancement methods are not effective for uneven lightening circumstances due to over-enhancement of images [39]. Therefore, CLAHE is used that can prevent excessive amplification of noise by limiting the contrast. It can be evaluated as:

$$f(D) = (1 - \omega_y)((1 - \omega_x)f_{ul}(D) + \omega_x f_{bl}(D)) + \omega_y((1 - \omega_x)f_{ur}(D) + \omega_x f_{br}(D)) \quad (3)$$

The mapping of four adjacent values of the histogram cumulative distribution function to pixel are desirable for every pixel. ω_x and ω_y indicate the distance among center of the left upper mask and pixels. $f()$ is cumulative distribution function. f_{br} , f_{bl} , f_{ul} , and f_{ur} represent below right, below left, upper left and upper right values in current patch, respectively. D represents the pixel coordinates.

D. PROPOSED DKNet

Deep neural networks can be made lightweight using MobileNet’s depth-separable convolutions. It reduces the number of parameters significantly when compared with networks with regular convolutions. Figure 2 shows the proposed DKNet model. Initially, joint trilateral filter is used to remove the noise from training images. Thereafter, contrast limited adaptive histogram equalization (CLAHE) is used to enhance the visibility of filtered images. Thereafter, MobileNet is used to extract the potential filters. Final layers of MobileNet, i.e., fully connected layer and softmax are removed. Flatten layer is then used to flatten the input. Thereafter, a fully connected classification layer is used. It utilizes two fully connected layers with a Rectified linear unit (ReLU) and dropouts are used. Dropouts are used to generalized the model, thus can prevent the over-fitting problem. Finally, the softmax activation function is used to provide the recognition results.

E. VALIDATION ON SELECTED DOCUMENTS

The trained DKNet model is tested on Kuzushiji documents using the following steps.

- 1) Initially, MSER and convexhull are used to extract the characters.
- 2) Apply joint trilateral filter on each extracted character image.
- 3) Apply CLAHE on the filtered character image.
- 4) Use trained DKNet model to recognize the character.

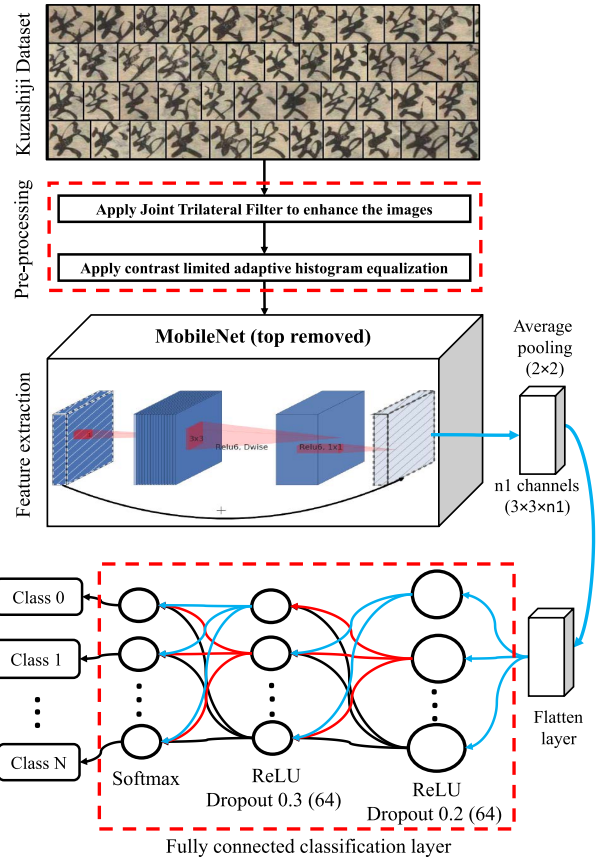


FIGURE 2. Diagrammatic flow of the proposed DKNet model.

IV. PROPOSED SEGMENTATION APPROACH FOR KUZUSHIJI CHARACTERS

This section discusses the proposed segmentation approach for Kuzushiji characters. Maximally stable extremal regions (MSER) and convexhull approaches are used to segment the characters from a given document.

Like a Scale-invariant feature detector (SIFT), MSER is a feature detector. It decomposes a given document to several co-variant regions, i.e., stable connected components or so-called MSERs. Figure 3 shows the diagrammatic flow of MSER technique.

To identify individual MSERs by characteristics of the Kuzushiji sample, MSER is applied to each document. To develop this pattern, binarization is achieved using OTSU threshold. The Pixel region of an area control is obtained by calculating the pixels region and constructing an ellipse using MinDiversity, MaxVariation, and Delta techniques. The contour is drawn around the arbitrary shape of the image using an elliptical frame that transposes the row index into a column index and vice versa. Based on a region seed, it returns a list of pixels within the region.

A. REGION OF GEOMETRIC PROPERTIES

The process of character detection and recognition can be achieved using following steps.

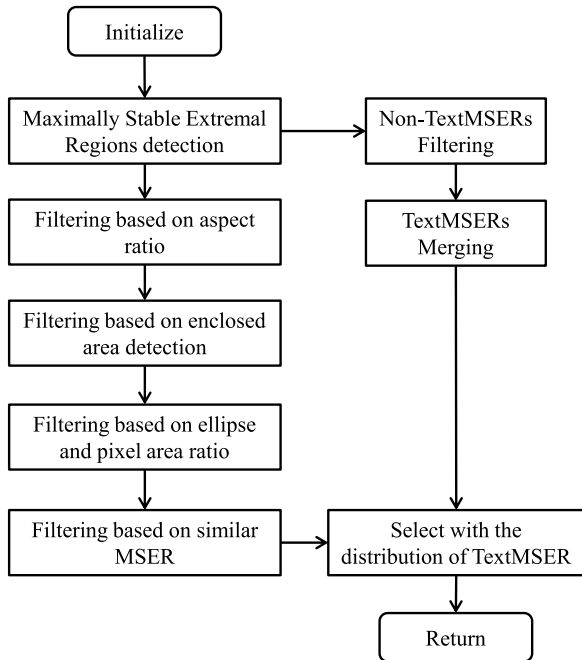


FIGURE 3. Diagrammatic flow of maximally stable extremal regions technique.

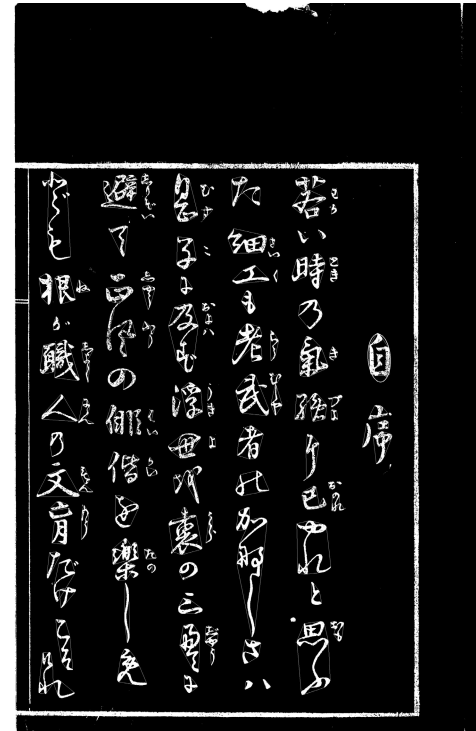


FIGURE 4. Region of image on which geometric properties are calculated.

- **Pre-processing:** Various conversion processes are used to convert RGB images to binary as shown in Figure 4.
- **Process:** An image was processed with each MSER algorithm as feature detector.
- **Properties:** Geometric attributes are employed to discriminate between non-text regions and text regions in the first pass for deleting the non-text regions from the obtained features (see Figure 5(a)).
- **Text regions:** To compute the remaining text area, the overlapping bounding boxes of these regions are combined to generate a bounded text region (see Figures 5(b) and 5(c)).
- **Detected Image:** The identified text bounding boxes in an image are examined using deep learning models to evaluate the actual and the predicted text.
- **Aspect ratio:** The bounding box ratio between width and height of the character is calculated using aspect ratio.
- **Eccentricity:** A given character’s circularity could be found using Eccentricity.
- **Extent:** To identify a region for character encoding, the rectangle enclosing text is determined to be a given value using Extent.
- **Solidity:** The convex hull area ratio is calculated for a given region of a character based on the pixels in the convex hull area.

The aspect ratio, eccentricity, extent, and solidity values are set to be 0.48130841121495327, 0.8839539479029316, 0.8026494873423464, and 0.07339475469138594, respectively.

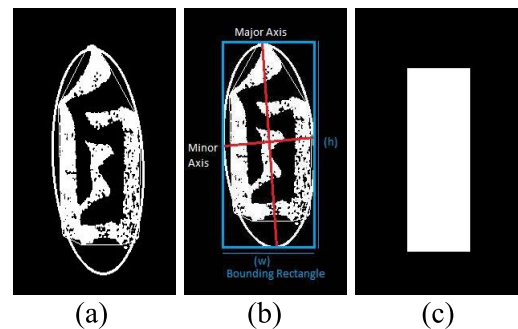


FIGURE 5. Geometric analysis using MSERs: (a) visualization of geometric properties, (b) rectangle image mask, and (c) sample image to calculate geometric properties using MSERs.

MSER calculates the mean square error of the pixels of an image over its background that have a constant intensity over its background. Text characters are assumed to have a comparable intensity and color contrast to their background. Some characters can be ignored if these characters have a small font size or represent the font’s pronunciation.

B. PRE-PROCESSING OF MSER

- Initially, white to black intensity thresholds are swept by using a simple luminance thresholding algorithm.
- External regions can be calculated as

$$S_r = p|I(p) > I(q)|\forall p \in S_r, \forall q \in B(S_r) \quad (4)$$

The gray-scale image I is indexed by p and q . $B(S_r)$ contains the boundaries of S_r as mentioned in Eq. (4).

An external region is maximally stable if it satisfies the following conditions, reject otherwise.

- The threshold δ , is the number of intensity threshold levels that must be reached for a region to be considered maximally stable during MSER computation.
- MSERs demonstrate a minimum level of variation which implies a high level of stability than the regions from previous threshold levels. It can be computed as follows:

$$V = \frac{area(R + \delta) - area(R)}{area(R)} \quad (5)$$

Here, R represents a threshold value for excising the region.

- A region beyond the area range or area size of the region in pixels should be excluded. For instance, an object whose area size is beyond 90% range of the whole image or less than 5% range would be rejected since it has no relevance for the actual test scenario as mentioned in Eq. (5).
- At different intensity thresholds, the maximum area variation exists between extremal regions. Therefore, the stability of results is limited. The variation in intensity thresholds leads to providing stable regions with tiny areas.

C. MSER REGIONS

The output of MSER algorithm is a set of pixels representing MSER regions. Ellipses can also be calculated to represent MSER regions. To represent a pixel list, it is intuitive to draw a minimum rectangle that encloses all of its coordinates. To represent ellipses, some mathematical operators are used to obtain the bounding box of each ellipse.

In the first method, a minimum enclosing rectangle for pixels is obtained within the region starting from different representations of MSER region. In the second method, a sampling set is used to find ROI coordinates. A bounding box of (x_{min}, y_{min}) and (x_{max}, y_{max}) can be easily derived for the entire list of pixels. These parameters identify the minimum and maximum coordinate values for the horizontal and vertical axes at any point. By geometrical transformation of MSER ellipses, ROI can be obtained in the form of bounding boxes as follows:

$$width = \max(2a \cos \theta, 2b \sin \theta) \quad (6)$$

$$height = \max(2a \sin \theta, 2b \cos \theta) \quad (7)$$

Here, a and b define the center values of extracted region. θ shows the rotation transformation.

To generate bounding boxes for MSER regions, ellipses are used. MSER regions often contain extra pixels near the text especially when text blends with its surroundings.

D. CHARACTER DETECTION APPROACHES

- **Approach 1: Morphological Image Operations** The characters in this approach are enlarged to a specified threshold limit. It means that all the single characters are grouped and the character's pronunciations are ignored.

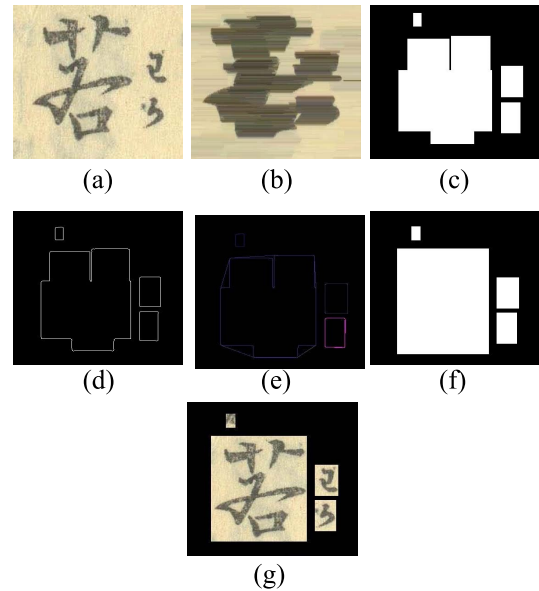


FIGURE 6. Analysis of morphological operation: (a) Input image, (b) dilatation, (c) bounding box extension, (d) canny edge detection, (e) contour technique, (f) mask applied on new contours, and (g) character extracted after operations.

Figure 6(a) and 6(b) illustrate the failed attempt to find the dilated characters for the given image and a sample of such a failure is shown below.

- **Approach 2: Contour technique** It is found that the gap between a single disconnected character is approximately in the range of 7 to 10 pixels as opposed to the gap between the character and its pronunciation which is approximately 14 pixels. An extension of the bounding box is made for this image about $7px$ to match the other disconnected component of the same character. The boundary mask area is drawn outward by $7px$ and the sample is shown in Figures 6(c), 6(d), and 6(e), respectively. Approximately 90% of all disconnected characters within the boundary mask meet (see Figures 6(f) and 6(g)).

V. EXPERIMENTAL RESULTS

Experiments are performed on Xeon E5 16200 v4 CPU and GPU GeForce GTX1080Ti $\times 4$ with memory as 64 GB and operating system is Ubuntu14.4. 3212 classes of Kuzushiji characters are used for building the DKNet model. The total number of training images is more than 600, 000. Kuzushiji dataset ID is obtained from the Center for Open Data in the Humanities (CODH) (refer to [29]). The parameter setting of DKNet and competitive models is presented in Table 2.

A. DATA AUGMENTATION

To augment the training data, three different augmentation techniques are also used. These techniques are random rotate, random blur, and random crop. The main objective is to augment the amount of training data to prevent the

TABLE 2. Parameters setting of DKNet and competitive models.

Parameter	Range
Padding pixels	0.1
Stride size	2
Filter size	7
Number of filters	6
Learning rate	0.01
epochs	100

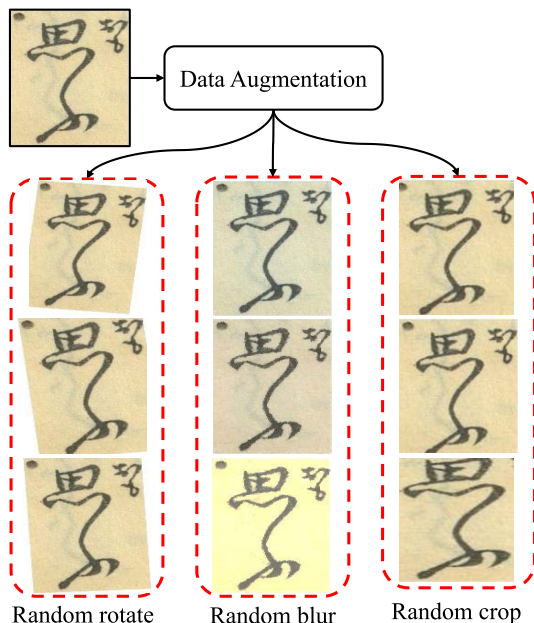


FIGURE 7. Impact of data augmentation techniques.

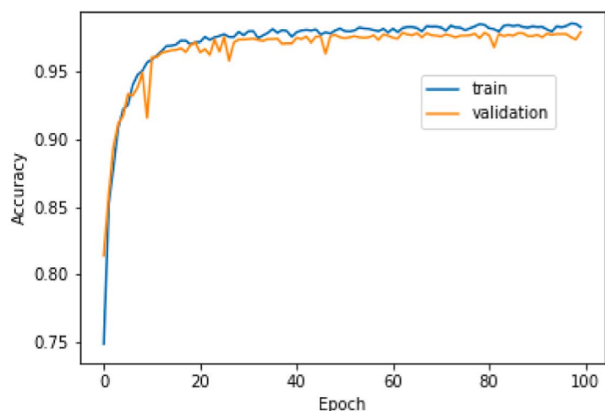


FIGURE 8. Training and validation accuracy of MobileNet.

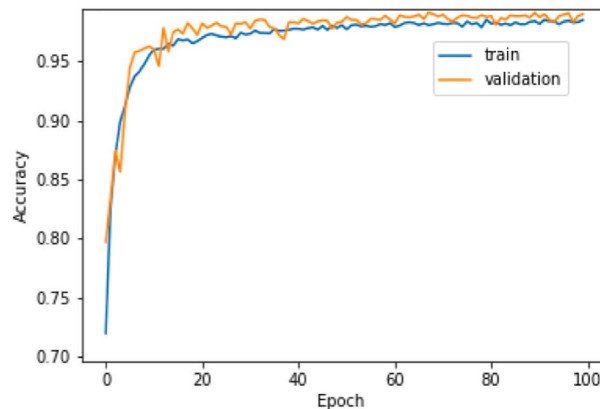


FIGURE 9. Training and validation accuracy of DKNet.

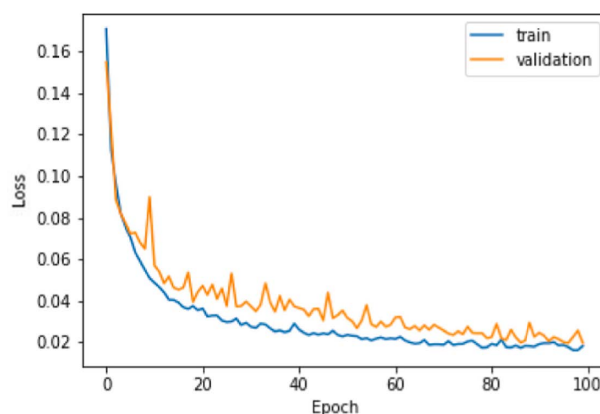


FIGURE 10. Training and validation loss of MobileNet.

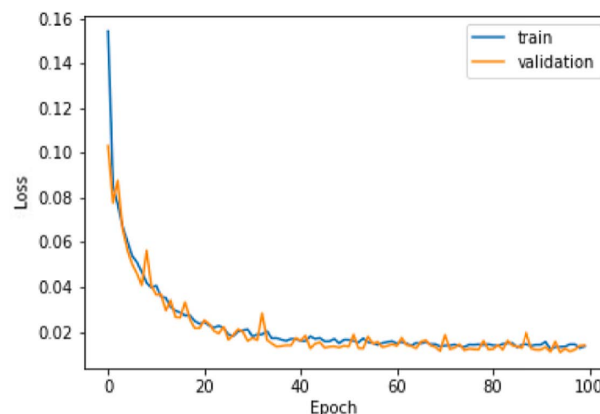


FIGURE 11. Training and validation loss of DKNet.

over-fitting and gradient vanishing problems. Figure 7 shows the impact of data augmentation techniques on a Kuzushiji character image.

B. TRAINING AND TESTING ANALYSIS

The training and validation accuracy analysis of MobileNet and DKNet are shown in Figures 8 and 9, respectively. The training and validation loss analysis of MobileNet and DKNet

are shown in Figures 10 and 11, respectively. It is found that DKNet has achieved the better training and validation accuracy as compared to MobileNet. Hence, DKNet is selected and implemented in our online API.

C. PERFORMANCE ANALYSIS

The performance of DKNet is evaluated using confusion matrix-based metrics such as accuracy, F-measure, and area

TABLE 3. Training analysis of DKNet.

Model	AUC	CER	F-measure	SER	Accuracy
LeNet	94.15 ± 1.81	20.48 ± 1.58	94.87 ± 1.14	58.59 ± 1.43	94.91 ± 1.45
GoogLeNet	97.45 ± 1.23	20.14 ± 1.18	97.34 ± 1.52	57.18 ± 1.54	97.32 ± 1.34
VGG16	97.12 ± 1.34	19.29 ± 1.23	94.95 ± 1.11	56.08 ± 0.98	98.03 ± 0.99
VGG19	97.72 ± 0.44	18.04 ± 0.64	95.49 ± 0.44	55.72 ± 0.34	97.17 ± 0.32
ResNet152V2	97.59 ± 0.94	17.52 ± 0.93	97.21 ± 0.79	54.25 ± 0.74	97.24 ± 0.72
InceptionNet	97.41 ± 1.13	16.19 ± 1.53	97.25 ± 1.07	53.51 ± 0.95	97.53 ± 0.93
InceptionResNetV2	97.81 ± 1.03	15.84 ± 0.93	97.82 ± 0.74	52.94 ± 0.78	97.94 ± 0.84
XceptionNet	97.81 ± 1.33	14.48 ± 1.01	97.83 ± 0.78	51.85 ± 1.04	97.88 ± 1.08
MobileNet	98.42 ± 0.68	13.41 ± 0.52	98.47 ± 0.63	50.74 ± 0.75	98.21 ± 0.85
Proposed DKNet model	99.59 ± 0.39	11.36 ± 0.21	99.64 ± 0.35	48.61 ± 0.29	99.67 ± 0.31

TABLE 4. Testing analysis of DKNet.

Model	AUC	CER	F-measure	SER	Accuracy
LeNet	94.75 ± 1.01	21.62 ± 1.72	94.74 ± 1.93	59.58 ± 1.92	94.71 ± 1.78
GoogLeNet	93.61 ± 1.31	21.54 ± 1.33	93.57 ± 1.64	58.51 ± 1.21	93.64 ± 1.23
VGG16	93.91 ± 1.24	20.77 ± 1.49	93.72 ± 1.21	57.74 ± 1.44	93.62 ± 1.44
VGG19	95.49 ± 1.41	19.19 ± 1.41	93.94 ± 1.41	56.21 ± 1.27	93.54 ± 1.22
ResNet152V2	94.31 ± 1.19	18.13 ± 1.14	94.15 ± 1.44	55.53 ± 1.43	93.51 ± 1.52
InceptionNet	95.21 ± 1.09	17.05 ± 1.14	93.44 ± 1.13	54.25 ± 1.09	94.83 ± 1.13
InceptionResNetV2	95.41 ± 1.17	16.81 ± 0.47	95.21 ± 0.89	53.29 ± 0.81	95.31 ± 0.22
XceptionNet	95.51 ± 1.25	15.12 ± 1.22	93.94 ± 1.22	52.02 ± 1.22	95.12 ± 1.46
MobileNet	96.63 ± 1.15	14.76 ± 1.51	95.32 ± 1.52	51.54 ± 0.42	96.34 ± 0.76
Proposed DKNet model	98.42 ± 0.81	12.78 ± 0.17	98.75 ± 0.47	49.53 ± 0.83	98.57 ± 0.95

under the curve (AUC). For evaluating handwriting recognition systems, we also use the Sequence Error Rate (SER) and Character Error Rate (CER) metrics [13]. DKNet's performance is evaluated through medians and uncertainty values (i.e., median ± $IQR \times 1.5$).

The nine models such as LeNet, GoogLeNet, VGG16, VGG19, ResNet152V2, Inception, InceptionResNetV2, XceptionNet, MobileNet, and DKNet are evaluated and the results are summarized in Tables 3 and 4.

Tables 3 and 4 illustrate the performance analysis of DKNet. DKNet achieves 99.67 and 98.57 accuracy values during training and testing. This means that the overfitting issue is not an issue. Further, DKNet was able to achieve an AUC value of 99.59 during training and 98.42 during testing. The proposed model is therefore least affected by the false positives and false negatives.

D. DISCUSSION

From the existing literature, it has been found that machine learning and deep learning techniques have been successfully used for Kuzushiji character recognition. Many models have been designed and implemented to recognize Kuzushiji characters from Japanese historical documents. Some well-known recognition models are RU-Net [40], AED [41], AACRN [45], 2DCbnp [48], MSOs [47], and RAED [13]. Although these models have shown significant results, the majority of these models have shown poor performance analysis. Therefore, an efficient deep Japanese recognition model using MSER is proposed. Table 5 demonstrates the performance analyses of DKNet and existing models.

RU-Net provides 86.83% accuracy and 85.53% F-measure. 2DCbnp provides better accuracy i.e., 93.32%,

TABLE 5. Performance comparison of DKNet with existing models.

Model	Database	Accuracy	F-measure	CER	SER
RU-Net [40]	CODH	86.83	85.48	-	-
AED [41]	-	-	-	32.34	-
AACRN [45]	CODH	-	-	21.48	94.97
2DCbnp [48]	CODH	93.32	92.42	-	-
MSOs [47]	CODH	94.67	-	-	-
RAED [13]	CODH	-	-	13.07	53.81
DKNet	CODH	98.57	98.75	12.78	49.53

and F-measure i.e., 92.42 as compared to RU-Net. The accuracy of MSOs is 94.67% which is significantly better than RU-Net and 2DCbnp. DKNet provides 98.57% accuracy and 98.57% F-measure. Hence, it can be seen that DKNet outperforms the existing techniques in terms of accuracy and F-measure. The CER of AED is 32.34%. CER and SER of AACRN are 21.49% and 94.97%. CER of AACRN is better than AED. RAED provides 13.07% CER and 53.81% SER which is significantly better than AED and AACRN. DKNet provides 12.78% CER and 49.53% SER. It can be seen that DKNet outperforms AED, AACRN, and ADE in terms of CER and SER. Therefore, the proposed DKNet model is an efficient recognition system. In future, performance of DKNet can be improved by tuning its hyper-parameters using metaheuristic techniques such as bat algorithm [50], particle swarm optimization [49], [51], etc.

E. EXPERIMENTAL ANALYSIS OF API

The recognition system is equipped on sever for online access. The CPU is Intel(R) Xeon(R) CPU E5-1410 v2 @ 2.80 GHz, RAM is 8G, and OS is Ubuntu 18.04.3 LTS. Apache HTTP Server was used for designing the API website.

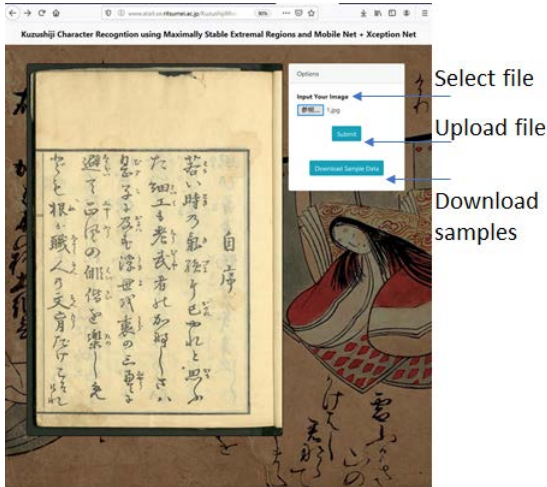


FIGURE 12. Interface of API.

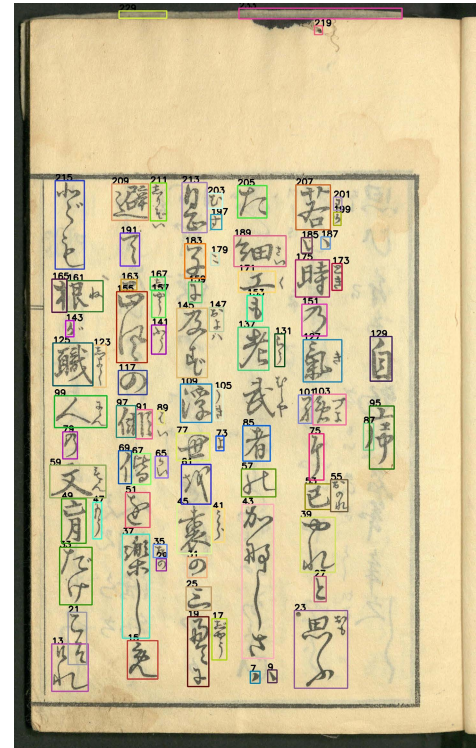


FIGURE 14. Segmentation of characters.



FIGURE 13. Masking MSER Technique.

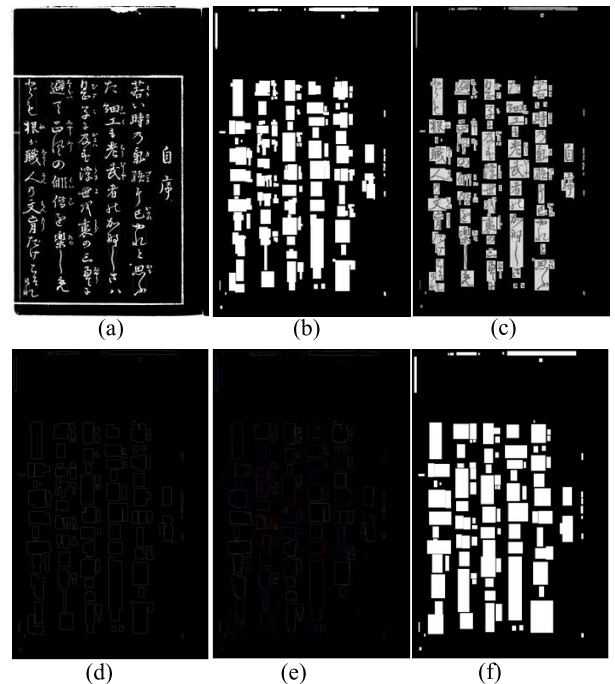


FIGURE 15. Extraction of characters: (a) MSER resultant image, (b) bounding mask drawn around possible characters, (c) text extracted with mask on the input image, (d) canny edge extracted from image mask, (e) convex hulls drawn around the canny edge extracted image, and (f) extraction of bounding mask from the convex hulls that contains characters.

Figure 12 shows the designed API, the target file can be selected from the local computer and submitted to our server. The user may download the sample files for testing the API.

Figures 13 and 14 show the obtained results of the proposed API. It is found that lots of characters are segmented correctly. The segmented characters are shown in the bottom of interface (see Figures 15 and 16). It is found that MSER with a bounding mask can achieve better results when there are significant gaps between characters. The proposed MSER and convex hull-based segmentation approach can segment

characters when there is no or lesser gap between the characters (refer Fig. 16(a)). Whenever the user clicks the

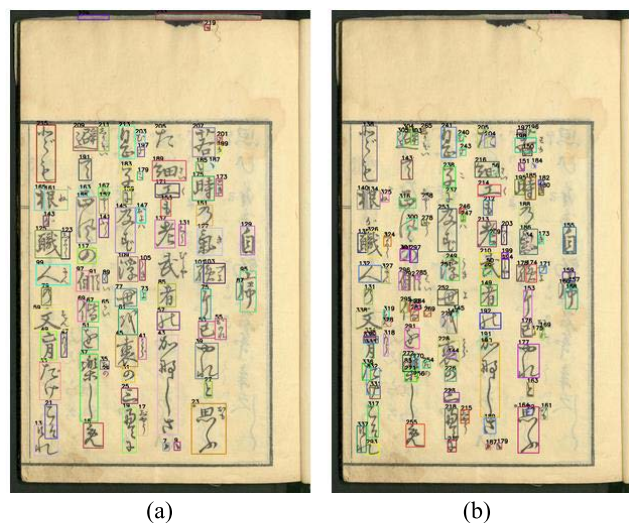


FIGURE 16. Extraction of single character: (a) extraction of text using Fig. 15(b) and (b) extraction of text using Fig.15(f).

characters, they will be recognized by the proposed model (refer Fig. 16(b)).

VI. CONCLUSION

A deep Kuzushiji characters recognition network (DKNet) was proposed to overcome over-fitting and gradient vanishing problems. First, the training images were filtered with a joint trilateral filter to remove noise. The filtered images were then enhanced with a contrast limited adaptive histogram equalization (CLAHE). A pre-trained MobileNet was then used to extract features from Kuzushiji characters. MobileNet's final layers, such as the fully connected layer and softmax, were removed. The flatten layer was then applied to the input. It was followed by a fully connected classification layer with rectified linear units (ReLU) and dropouts. Dropouts were used to generalize the model, thus preventing the over-fitting problem. Finally, the recognition results were obtained by applying the softmax activation function. To test the proposed model, actual documents were segmented using the proposed MSERs and a convexhull-based segmentation approach. Segmented characters were then recognized by the trained DKNet. Analyses of the competitive models revealed that DKNet surpassed the competitive models in terms of several performance metrics such as accuracy, F-measure, AUC, CER, and SER by 2.1675%, 1.9842%, 2.1075%, 1.6914%, and 1.7265%, respectively. An efficient Application Programming Interface (API) was also developed for Japanese Kuzushiji ancient heritage character recognition to help the end-users.

REFERENCES

- [1] W. H. Bangyal and J. Q. A. Abbas, "Recognition of off-line isolated handwritten character using counter propagation network," *Int. J. Eng. Technol.*, vol. 5, no. 2, p. 227, 2013.
- [2] Q. Abbas, W. H. Bangyal, and J. Ahmad, "Analysis of learning rate using BP algorithm for hand written digit recognition application," in *Proc. Int. Conf. Inf. Emerg. Technol.*, Jun. 2010, pp. 1–5.
- [3] N. Fujisaki, Y. Ishikawa, M. Takata, and K. Joe, "Crawling low appearance frequency character images for early-modern Japanese printed character recognition," in *Advances in Parallel & Distributed Processing, and Applications*. Cham, Switzerland: Springer, 2021, pp. 683–695.
- [4] C. V. Aravinda, M. Lin, A. Masahiko, and P. G. Amar, "A complete methodology for Kuzushiji historical character recognition using multiple features approach and deep learning Model," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 8, pp. 1–7, 2020.
- [5] L. Meng, B. Lyu, Z. Zhang, C. V. Aravinda, N. Kamitoku, and K. Yamazaki, "Oracle bone inscription detector based on SSD," in *Proc. Int. Conf. Image Anal. Process*. Cham, Switzerland: Springer, 2019, pp. 126–136.
- [6] F. Horie, H. Goto, and T. Suganuma, "Synthetic scene character generator and ensemble scheme with the random image feature method for Japanese and Chinese scene character recognition," *IEICE Trans. Inf. Syst.*, vol. 104, no. 11, pp. 2002–2010, Nov. 2021.
- [7] B. Lyu, Z. Wang, H. Li, A. Tanaka, K. Funumoto, and L. Meng, "Deep leaning based medicine packaging information recognition for medication use in the elderly," *Proc. Comput. Sci.*, vol. 187, pp. 194–199, Jan. 2021.
- [8] A. Sayeed, J. Shin, M. A. M. Hasan, A. Y. Srizon, and M. M. Hasan, "BengaliNet: A low-cost novel convolutional neural network for Bengali handwritten characters recognition," *Appl. Sci.*, vol. 11, no. 15, p. 6845, Jul. 2021.
- [9] T. Horiuchi and S. Kato, "A study on Japanese historical character recognition using modular neural networks," *Int. J. Innov. Comput., Inf. Control*, vol. 7, no. 8, pp. 5003–5014, 2005.
- [10] F. Li and P.-Y. Woo, "The coding principle and method for automatic recognition of Jia Gu Wen characters," *Int. J. Hum. Comput. Stud.*, vol. 53, no. 2, pp. 289–299, Aug. 2000.
- [11] X. Hu, M. Inamoto, and A. Konagaya, "Recognition of Kuzushi-Ji with deep learning method a case study of Kiritsubo chapter in the Tale of Genji," in *Proc. 33rd Annu. Conf. Jpn. Soc. Artif. Intell.*, 2019, pp. 1–2.
- [12] B. Lyu, R. Akama, H. Tomiyama, and L. Meng, "The early Japanese books text line segmentation base on image processing and deep learning," in *Proc. Int. Conf. Adv. Mech. Syst. (ICAMechS)*, Aug. 2019, pp. 299–304.
- [13] A. D. Le, T. Clanuwat, and A. Kitamoto, "A human-inspired recognition system for pre-modern Japanese historical documents," *IEEE Access*, vol. 7, pp. 84163–84169, 2019.
- [14] Q. Li, Y. Yang, and A. Wang, "Recognition of inscriptions on bones or tortoise shells based on graph isomorphism," *Comput. Eng. Appl.*, vol. 47, no. 8, pp. 112–114, 2011.
- [15] Q. S. Li and Y. X. Yang, "Sticker DNA algorithm of Oracle-bone inscriptions retrieving," *Comput. Eng. Appl.*, vol. 44, no. 28, pp. 140–142, 2008.
- [16] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
- [18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
- [19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
- [20] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-V4, inception-ResNet and the impact of residual connections on learning," 2016, *arXiv:1602.07261*.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*.
- [22] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2016, pp. 4700–4708.
- [23] K. Brown, *Encyclopedia of Language & Linguistics*, J. M. Unger, Ed., 2nd ed. Amsterdam, The Netherlands: Elsevier, 2006, pp. 95–102, doi: 10.1016/B0-08-044854-2/04566-1.
- [24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
- [25] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017.

- [26] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
- [27] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottleneck," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4510–4520.
- [28] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," 2015, *arXiv:1512.02325*.
- [29] Center for Open Data in the Humanities (CODH). *Japanese Classic Cursive Script Data Set Book Title List [Data Set]*. Accessed: Feb. 26, 2020. [Online]. Available: <http://codh.rois.ac.jp/char-shape/book/>
- [30] C. V. Aravinda, M. Lin, and P. G. Amar. *Kuzashi Recognition API*. Accessed: Feb. 26, 2020. [Online]. Available: <http://www.atait.se.ri.tsumei.ac.jp/KuzushijiMser/>
- [31] L. Meng, C. V. Aravinda, K. R. U. K. Reddy, T. Izumi, and K. Yamazaki, "Ancient Asian character recognition for literature preservation and understanding," in *Proc. Euro-Mediterranean Conf. Cham, Switzerland: Springer*, 2018, pp. 741–751.
- [32] C. V. Aravinda and H. N. Prakash, "A review on various handwritten cursive characters segmentation techniques," in *Proc. 2nd Int. Conf. Appl. Theor. Comput. Commun. Technol. (iCATect)*, 2016, pp. 647–651.
- [33] C. V. Aravinda and H. N. Prakash, "Template matching method for Kannada handwritten recognition based on correlation analysis," in *Proc. Int. Conf. Contemp. Comput. Informat. (ICI)*, Nov. 2014, pp. 857–861.
- [34] L. Meng, "Recognition of Oracle bone inscriptions by extracting line features on image processing," in *Proc. 6th Int. Conf. Pattern Recognit. Appl. Methods (ICPRAM)*, 2017, pp. 606–611, doi: [10.5220/0006225706060611](https://doi.org/10.5220/0006225706060611).
- [35] L. Meng, "Two-stage recognition for Oracle bone inscriptions," in *Proc. ICIAP*, 2017, pp. 672–682, doi: [10.1007/978-3-319-68548-9](https://doi.org/10.1007/978-3-319-68548-9).
- [36] H. T. Nguyen, N. T. Ly, K. C. Nguyen, C. T. Nguyen, and M. Nakagawa, "Attempts to recognize anomalously deformed Kana in Japanese historical documents," in *Proc. 4th Int. Workshop Historical Document Imag. Process.*, 2017, pp. 31–36.
- [37] D. Singh and V. Kumar, "Dehazing of remote sensing images using improved restoration model based dark channel prior," *Imag. Sci. J.*, vol. 65, no. 5, pp. 282–292, 2017.
- [38] M. Kaur, D. Singh, V. Kumar, and K. Sun, "Color image dehazing using gradient channel prior and guided L_0 filter," *Inf. Sci.*, vol. 521, pp. 326–342, Jun. 2020.
- [39] A. M. Reza, "Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement," *J. VLSI Signal Process. Syst. Signal, Image Video Technol.*, vol. 38, no. 1, pp. 35–44, 2004.
- [40] T. Clanuwat, A. Lamb, and A. Kitamoto, "KuroNet: Pre-modern Japanese kuzushiji character recognition with deep learning," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 607–614.
- [41] A. D. Le, D. Mochihashi, K. Masuda, and H. Mima. (2018). *An Attention-Based Encoder–Decoder for Recognizing Japanese Historical Documents*. Pattern Recognition and Machine Understanding (PRMU). [Online]. Available: <https://www.ieice.org/ken/paper/20181213H11M/eng>
- [42] L. Bing, H. Tomiyama, and L. Meng, "Frame detection and text line segmentation for early Japanese books understanding," in *Proc. 9th Int. Conf. Pattern Recognit. Appl. Methods*, 2020, pp. 600–606, doi: [10.5220/0009179306000606](https://doi.org/10.5220/0009179306000606).
- [43] H. T. Nguyen, T. Nakamura, C. T. Nguyen, and M. Nakagawa, "Online trajectory recovery from offline handwritten Japanese kanji characters of multiple strokes," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 8320–8327.
- [44] Y. Septiana, A. Mulyani, D. Kurniadi, and H. Hasanudin, "Handwritten recognition of Hiragana and Katakana characters based on template matching algorithm," in *Proc. IOP Conf. Mater. Sci. Eng.*, vol. 1098, no. 3, Mar. 2021, Art. no. 032093.
- [45] N. T. Ly, C. T. Nguyen, and M. Nakagawa, "Attention augmented convolutional recurrent network for handwritten Japanese text recognition," in *Proc. 17th Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Sep. 2020, pp. 163–168.
- [46] L. Sichao and H. Miwa, "Algorithm using deep learning for recognition of Japanese historical characters in photo image of historical book," in *Proc. Int. Conf. Intell. Netw. Collaborative Syst.* Cham, Switzerland: Springer, 2019, pp. 181–189.
- [47] K. Ueki, T. Kojima, R. Mutou, R. S. Nezhad, and Y. Hagiwara, "Recognition of Japanese connected cursive characters using multiple softmax outputs," in *Proc. IEEE Conf. Multimedia Inf. Process. Retr. (MIPR)*, Aug. 2020, pp. 127–130.
- [48] A. D. Le, "Detecting Kuzushiji characters from historical documents by two-dimensional context box proposal network," in *Proc. Int. Conf. Future Data Secur. Eng.* Cham, Switzerland: Springer, 2019, pp. 731–738.
- [49] W. H. Bangyal, A. Hameed, J. Ahmad, K. Nisar, M. R. Haque, A. A. A. Ibrahim, J. J. P. C. Rodrigues, M. A. Khan, D. B. Rawat, and R. Etengu, "New modified controlled bat algorithm for numerical optimization problem," *Comput., Mater. Continua*, vol. 70, no. 2, pp. 2241–2259, 2022.
- [50] S. Pervaiz, Z. Ul-Qayyum, W. H. Bangyal, L. Gao, and J. Ahmad, "A systematic literature review on particle swarm optimization techniques for medical diseases detection," *Comput. Math. Methods Med.*, vol. 2021, pp. 1–10, Sep. 2021.
- [51] W. H. Bangyal, A. Hameed, W. Alosaimi, and H. Alyami, "A new initialization approach in particle swarm optimization for global optimization problems," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–17, May 2021.

• • •

Predicting the Performance of Cooperative Wireless Networking Schemes With Random Network Coding

Jin-Taek Seong and Heung-No Lee, *Senior Member, IEEE*

Abstract—In this paper, we consider a cooperative wireless network in which there are multiple sources and multiple relays. Owing to unreliable wireless channels, the quality of network links between nodes can vary. This results in the failure of intermediate nodes that generate linear combinations of incoming messages in network coding schemes. We propose an analytical framework to evaluate the recovery performance of source messages at the base station. To this end, we consider a random transmission matrix in which each element of the transmission matrix is processed a random variable, where its distribution is a function of the outage probability. We derive an upper bound for the reconstruction performance, i.e., decoding failure probability and nullity. The proposed framework provides an evaluation tool that enables us to investigate the impact of a large number of sources and relays, as well as the field size of the network codes on system performance.

Index Terms—Cooperative network, network coding, upper bound, rank.

I. INTRODUCTION

CHANNEL fading is one of the underlying causes of the performance degradation in wireless networks. One naive approach to combat channel fading is to increase the transmit power. A more advanced approach is to utilize modern diversity techniques, which can be performed without increasing the transmit power. To date, numerous diversity techniques have been proposed and employed in the time, frequency, and space domains. Cooperative networking is one of the current approaches that aim to utilize spatial diversity via user cooperation. Each user participates collaboratively, and shares the benefit of a virtual antenna array in transceiver messages that are available through another user's antenna [1].

Network coding [2] first proposed by Ahlswede *et al.* is shown to achieve maximum information flow in a single source multicast network. Numerous efforts have subsequently been attempted; these efforts focused on elucidating if network coding can provide additional advantages compared to other cooperative networking schemes. For example, network coding over the binary field [3], [4] has shown to improve diversity gain and provide higher spectral efficiency in wireless networks,

Manuscript received October 10, 2013; revised March 11, 2014 and May 18, 2014; accepted June 4, 2014. Date of publication June 13, 2014; date of current version August 20, 2014. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MEST) through the Do-Yak Research Program under Grant 2010-0017944. The associate editor coordinating the review of this paper and approving it for publication was T. Tsiftsis.

The authors are with the School of Information and Communications, Gwangju Institute of Science and Technology, Gwangju 500-712, Korea (e-mail: jtseong@gist.ac.kr; heungno@gist.ac.kr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCOMM.2014.2330825

whereas network coding with a nonbinary field further increase those benefits [5]–[9]. In particular, numerous studies have investigated the extent to which network coding can improve the performance of media access control and routing protocols, in terms of energy efficiency [22], transmission delay [24], and throughput [23], [25], compared to traditional forward-and-relay only based designs [19]–[21]. The performance of cooperative wireless networking schemes with network coding has been analyzed, and compared with erasure channel models [5]–[8], and error propagation models [30]–[33]. We will further address recent cooperative communication schemes in Section II by categorizing them with respect to their decoding techniques, spectral efficiencies, and cooperative strategies.

Xiao and Skoglund [5], [6] recently proposed a network coding scheme called Dynamic Network Codes (DNC) to handle a dynamic network topology. The inherent nature of wireless channels implies that links are unreliable and that link failures will occur randomly in the inter-user channels. DNC is performed successfully over such a dynamic network channel topology, in conjunction with techniques such as enhanced diversity order. In the DNC schemes, multiple network code matrices are used; each one is designed to handle a particular occurrence of link outages. In particular, an intermediate node in a network may fail to decode some of the messages received from the other nodes. The intermediate node creates, and later on forwards it to the base station, a linear combination of the messages which it could only successfully decode, and then forwards it to the base station. That is, a certain occurrence of link outages results in a particular restriction to the elements of the network code matrix. Thus, each network code matrix in the DNC scheme, referred to as the *transmission matrix* in this paper, is designed carefully so as to work effectively for the occurrence of a specific set of link failures. In addition, Rebelatto *et al.* in [7] and [8] extended the two-phase transmission framework of the DNC to multiple phases, in the Generalized Dynamic Network Codes (GDNC), to further enhance the transmission rate and the diversity order.

In this paper, our goal is to focus on the system models of the DNC and GDNC schemes, and provide a novel analysis framework for them. As noted earlier, there are other recently studied cooperative communications schemes, each with an advantage in a different perspective such as spectral efficiency and higher decoding performance. These will be discussed further in Section II. The DNC and GDNC schemes are found to be interesting due to at least to the following two aspects: i) DNC is the first network coding schemes designed for dynamic network topology. To the best of the authors' knowledge, DNC is the first attempt to adaptively use different network code

matrix as the link failure varies, and is designed to achieve the so-called the *min-cut capacity* of randomly changing links [5], ii) GDNC is shown to achieve full diversity order and increases the transmission rate [7]. While a series of performance analyses for DNC and GDNC.

Although a series of performance analyses for DNC and GDNC are provided in [5]–[8], the authors rely on the exhaustive investigation of all individual network code matrices to determine if the resultant transmission matrix at the base station is sufficiently able to decode the source messages. This is an exceedingly time-consuming and tedious process; thus, it cannot be extended to larger and more general networks where the link outage probabilities throughout the networks are, in general, different from each other.

In particular, the performance analyses in [5]–[8] to determine the probability of successful decoding at the base station are performed only for small and non-general networks. A successful decoding is assumed to be achieved when the network code matrix at the base station has a sufficient number of linearly independent vectors that at least equals the number of unknown source messages. The successful event begins by determining whether the rank of the network code matrix at the base station is full. Then, the success probability is obtained by adding all individual probabilities of such events over all possible link failures. To achieve this outcome, the authors, using Theorem 1 in [5] and Section V in [7], followed the approach of tracking down each network code matrix individually, and determining if each was full in rank. This is an exhaustive process. When the number of nodes in a network increases, it is evident that this approach becomes intractable, because of the exponential increase in possible combinations. For example, the total number of distinct $N \times N$ random matrices with full rank is $\prod_{i=1}^N (q^N - q^{i-1})$ for the finite field of size q [10].

As a result, the analyses performed in [5]–[8] are limited to small and homogeneous networks, i.e., a network of fewer than 10 nodes, with link failure probabilities set to be equal throughout the network. These methods are not suitable for analyzing networks where nodes are randomly deployed in an area of interests, and wireless networks present heterogeneous link outage probabilities. The lack of a general and systematic performance analysis framework to deal with such networks has motivated this study.

In this paper, our main goal is to propose a novel evaluation framework for cooperative network coding schemes. The contributions of this work are summarized as follows.

- (*Design of random transmission matrix*) We model a random transmission matrix with uniform and maximum distance separable (MDS) distributions in (6)–(8) of Section IV. The elements of this matrix are represented with random variables as a function of the outage probability of each wireless link. This new system model, enables us to avoid the exhaustive counting of each network code metric occurrence.
- (*Tight upper bounds to probability of decoding failure*) We derive a series of tight upper bounds on the probability of failure. In particular, the dimension of the nullspace of the random transmission matrix is used to derive an

TABLE I
SUMMARY OF RECENT TECHNIQUES FOR
COOPERATIVE COMMUNICATIONS

	Recent technique	References
Decoding techniques	Maximum ratio combining	[4], [13]–[15]
	Rank-based decoding	[5]–[8], [10], [11]
High spectral efficiency	Multuser detection	[17]
	Interference cancellation	[18]
	Non-orthogonal channel	[16]
Cooperative strategies	Amplify-and-Forward	[1], [35]
	Decode-and-Forward	[1], [36]
	Compute-and-Forward	[34]

upper bound, as discussed in Proposition 3. It is then linked to the decoding failure probability where the rank of the network code matrix is not full, which is shown in Theorem 4. The upper bounds have proven to be considerably tight in comparison to simulation results.

- (*Generality and scalability*) The developed analysis framework is general and scalable, offering the capability of analyzing large wireless networks with random deployments where all outage probabilities of wireless links are different. For example, consider the network scenario of randomly deployed nodes shown in Fig. 6 in Section VI. In addition, the developed framework can handle a large cooperative network that has more than 100 nodes. To the best of our knowledge, this scale of wireless networks is unprecedented in DNC and GDNC performance evaluations. The proposed framework enables us to investigate its impact on the successful reconstruction of source messages based on varying outage probabilities, and on various key parameters such as the number of relays and the field sizes in DNC and GDNC schemes.

The remainder of this paper is organized as follows. We review other recent cooperative communications techniques in Section II. We explain our cooperative model for wireless networks in Section III. In Section IV, we model a transmission matrix in terms of an outage probability. In Section V, we calculate upper bounds for the reconstruction performance of cooperative frameworks with various types of link connectivity. Finally, we evaluate the proposed framework in Section VI, and conclude the paper in Section VII.

II. OTHER RECENT WORKS AND RELATION TO OUR WORK

In this section, and in Table I, we provide an overview of the prevalent cooperative communications schemes believed to be closely related to the network coding schemes considered in this paper.

Two decoding approaches in cooperative wireless communications have been recently considered. The first is the Maximum Ratio Combining (MRC) decoding scheme [4], [13]–[15]. The main idea is to configure the network to coordinate the transmissions, and repeat a signal with a weak signal-to-noise ratio (SNR) multiple times over independent fading channels. In this manner, MRC allows the destination to maximize SNR. To implement an MRC scheme, all decoding information, as well as the success or failure of each source message at the base station, should be identified and forwarded to the relays. This is required to determine the source of the SNR, for which

retransmission was required. To enable its deployment in large-scale networks, the scheduling issue must be resolved.

Recently, two research groups have proposed advanced cooperative network schemes to achieve high spectral efficiency. In [16], Youssef and Amat have proposed the use of non-orthogonal channel allocation to improve spectral efficiency. For wireless networks where a multiuser detection receiver is utilized at the base station, cooperative transmission protocols with high spectral efficiency have been developed [17]. Furthermore, the work describe in [18] has aimed at improving the spectral efficiency of cooperative systems using superposition coding and iterative detection methods.

We believe our analysis framework could be utilized in [16] with a necessary but simple modification. The first aspect to consider is that all wireless channels should not be modeled independently from each other anymore. The outage probabilities are not independent from each other. This can be achieved by designing a joint probability distribution for the random matrix. The probability that the random transmission matrix is not full rank can be obtained by considering such an event over the joint probability distribution. For more details, we will show Example 3 for channel correlation cases in Section V-B of this paper.

Using network coding on lattice codes, Nazer and Gastpar recently proposed the compute-and-forward (CF) relaying scheme [34]. In CF, a relay is configured with a linear combination of multiple codeword signals, which are simultaneously transmitted and superposed in the air. The key idea of using CF relaying in network coding is to utilize the property of the lattice code property stating that the *integer combination* of lattice codewords remains a lattice codeword. Thus, the relay receives a codeword with additive noise. After a denoising step, the relay retransmits the decoded lattice codeword to the base station. Therefore, the benefit of using CF relaying in network coding is evident. Because the transmissions from all sources to the relay are performed simultaneously, the spectral efficiency is significantly enhanced.

There are two widely recognized cooperative relaying strategies, referred to as amplify-and-forward (AF) [1], [35] and decode-and-forward (DF) [1], [36]. AF and DF cooperative relaying strategies perform effectively in either low or high SNR regimes, while CF approach offers advantages in moderate SNR regimes where both interference and noise are significant factors [34]. The DNC and GDNC schemes considered in this paper are categorized as DF based network coding strategies.

Cooperative wireless communications with multiple sources and multiple relays closely related to our work have attracted significant attention because of their higher achievability rate [26], better error performance [28], [29] and diversity-multiplexing tradeoffs [9], [27]. There are two types of error propagation models worth that should be considered here. The authors in [30]–[33] assume a network channel model where erroneous messages are permitted to propagate throughout the network. For the erasure channel model, [5]–[8], erroneous messages at the relays are discarded, to avoid unnecessary error propagation caused by encoding and forwarding operations.

Recent studies [5]–[12] closely related to this paper have focused on the performance analysis and the design of network

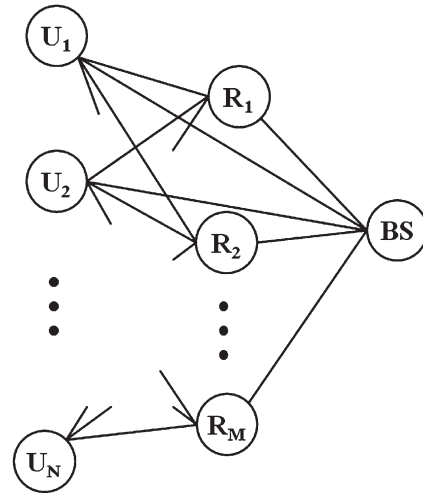


Fig. 1. An (N, M) cooperative network with N sources and M relays.

code matrices for cooperative networks with erasure channel models. In particular, in order to maximize diversity order in a multiple-access network, the problem of designing network code matrices subject to link failures was studied [5]–[8], to maximized diversity order in a multiple-access network. In [7], it is shown that the design of network code matrices is equivalent to the design of linear block codes for erasure correction coding. It was shown that maximum diversity order is guaranteed if an MDS code generator matrix of MDS codes is utilized as the network code matrix. In addition, Nguyen *et al.* [11] have defined upper and lower bounds on GDNC scheme recovery performance. For random linear network coding schemes, Trullols-Cruces *et al.* [10] have derived the exact decoding probability of obtaining network codes of full rank.

III. COOPERATIVE NETWORK

We consider an (N, M) cooperative scheme for wireless networks as shown in Fig. 1, in which there are N sources, $\{U_1, U_2, \dots, U_N\}$, and M relays, $\{R_1, R_2, \dots, R_M\}$. There are two cooperating transmission phases: *broadcasting* and *relaying*. In the broadcasting phase, each source transmits its message to the base station (BS). Owing to the nature of wireless channels, the relays in this phase can, in general, receive and successfully decode the messages from the sources. In the relay phase, each relay can generate a parity message constructed from a linear combination of these messages, and forward it to the BS. In this work, we assume that the received message for a single channel is considered either completely corrupted—an outage and therefore not available at the receiver—or error-free, i.e., no outage. For more complicated cooperative communications error models that have been studied in [30]–[33], we have included a discussion in Section II.

For both transmission types, we assume that in both transmissions all transmitters send their signals through orthogonal channels using either time- or frequency-division multiple access, and that all channels are spatially independent, flat-faded, and perturbed by additive white Gaussian noise (AWGN). We further assume that the channel gains are independent in both the broadcasting and relay phases. A discussion is included in

Section II in which no orthogonal transmissions are utilized and inter-user channels are not independent.

In the broadcasting phase, the signal y_{u,d_1} received at node d_1 for $d_1 \in \{R_1, R_2, \dots, R_M, BS\}$ is given by $y_{u,d_1} = \sqrt{P_u} h_{u,d_1} x_{u,d_1} + n_{u,d_1}$, where u denotes the transmitter node, i.e., $u \in \{U_1, U_2, \dots, U_N\}$; P_u denotes the transmit power at node u ; h_{u,d_1} denotes the channel gain between the two nodes u and d_1 , which is a circular symmetric complex-valued Gaussian random variable with zero mean and variance $\sigma_{u,d_1}^2/2$ per dimension; x_{u,d_1} is the signal transmitted from node u ; and noise n_{u,d_1} denotes the complex-valued AWGN with zero mean and variance $N_0/2$ per dimension. In the relay phase, the signal y_{r,d_2} received at the BS is $y_{r,d_2} = \sqrt{P_r} h_{r,d_2} x_{r,d_2} + n_{r,d_2}$, where r denotes the relay node, i.e., $r \in \{R_1, R_2, \dots, R_M\}$; d_2 denotes the BS; P_r denotes the transmit power at relay node r ; h_{r,d_2} denotes the channel gain between the relay node r and the BS, which is a circular symmetric complex-valued Gaussian random variable with zero mean and variance $\sigma_{r,d_2}^2/2$ per dimension; x_{r,d_2} is the signal transmitted from relay node r ; and noise n_{r,d_2} denotes the same AWGN as in the broadcasting phase. For Rayleigh fading channels, the variances of channel gains are defined as $\sigma_{u,d_1}^2 := \rho_{u,d_1}^{-\eta}$ and $\sigma_{r,d_2}^2 := \rho_{r,d_2}^{-\eta}$, letting ρ_{u,d_1} and ρ_{r,d_2} be the distances for u -to- d_1 and r -to- d_2 , respectively, and η be the path-loss exponent, i.e., $2 \leq \eta \leq 6$ [37]. Throughout this paper, we use $\eta = 3$. The instantaneous SNRs of the two channels are denoted as $\gamma_{u,d_1} := |h_{u,d_1}|^2 P_u / N_0$ and $\gamma_{r,d_2} := |h_{r,d_2}|^2 P_r / N_0$.

Let R_{th} be the predefined threshold of the spectral efficiency in bits/s/Hz. For both phases, we utilize the following outage probabilities based on [1] and [38],

$$\delta_{u,d_1} = \Pr \{ \log(1 + \gamma_{u,d_1}) < R_{th} \}, \quad (1)$$

and

$$\delta_{r,d_2} = \Pr \{ \log(1 + \gamma_{r,d_2}) < R_{th} \}. \quad (2)$$

Throughout this paper, we use $R_{th} = 1$ bit/s/Hz. Each of the outage probabilities is a function of the instantaneous SNR and the distance between two nodes. We can use these outage probabilities to model the elements of a transmission matrix.

IV. MODELING OF TRANSMISSION MATRICES

A. Transmission Matrix

We utilize the outage probabilities defined in Section III to model the elements of the transmission matrix. A random transmission matrix can be used to represent a family of network coding matrices for an (N, M) cooperative scheme shown in Fig. 1 in which two transmissions occur over a multiple access network with N sources and M relays. Let \mathbb{F}_q be a finite field of size q . Let $\mathbf{x} \in \mathbb{F}_q^{N \times 1}$ denote the $N \times 1$ vector of transmitted messages, $\mathbf{y} \in \mathbb{F}_q^{(N+M) \times 1}$ denote the $(N+M) \times 1$ vector of messages received at the BS, and $\mathbf{A} \in \mathbb{F}_q^{(N+M) \times N}$ denote the $(N+M) \times N$ transmission matrix. The vector \mathbf{y} received at the BS is then given by

$$\mathbf{y} = \mathbf{A}\mathbf{x}, \quad (3)$$

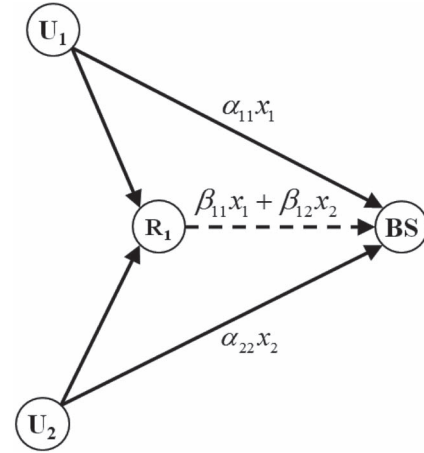


Fig. 2. The $(2,1)$ cooperative wireless network with $N = 2$ and $M = 1$. The solid lines indicate the broadcasting phase, and the dashed line indicates the relay phase.

where the transmission matrix \mathbf{A} consists of the $N \times N$ direct matrix \mathbf{D} and the $M \times N$ combination matrix \mathbf{P} ; i.e., $\mathbf{A} := \begin{bmatrix} \mathbf{D} \\ \mathbf{P} \end{bmatrix}$. Note that all of the arithmetic operations are performed over finite fields.

The direct matrix \mathbf{D} can be modeled as a diagonal matrix, i.e., one with zeroes for all off-diagonal elements. If there are no outage events for the channel links between the sources and the BS, the diagonal elements of this direct matrix are all set to one; otherwise, the corresponding elements are set to zero. Let α_{ii} denote the i th diagonal element of \mathbf{D} , i.e., $\alpha_{ii} \in \{0, 1\}$ for $i \in \{1, 2, \dots, N\}$, then the i th element $y_{i,1}$ of \mathbf{y} is represented as $y_{i,1} = \alpha_{ii} x_i$, where $x_i \in \mathbb{F}_q$ denotes the i th element of \mathbf{x} , and 1 in the subscript of $y_{j,1}$ indicates the broadcasting phase for $j \in \{1, 2, \dots, M\}$. Let $\beta_{ji} \in \mathbb{F}_q$ denote an element of \mathbf{P} , then the $(N+j)$ th element $y_{j,2}$ of \mathbf{y} is represented by $y_{j,2} = \sum_{i=1}^N \beta_{ji} x_i$, where 2 in the subscript of $y_{j,2}$ indicates the relay phase. As the BS receives $N+M$ messages from N sources and M relays, we can represent the transmission matrix as

$$\begin{bmatrix} y_{1,1} \\ \vdots \\ y_{N,1} \\ y_{1,2} \\ \vdots \\ y_{M,2} \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \alpha_{NN} \\ \beta_{11} & \cdots & \beta_{1N} \\ \vdots & \ddots & \vdots \\ \beta_{M1} & \cdots & \beta_{MN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}. \quad (4)$$

Note that all elements of the transmission matrix \mathbf{A} are random variables except for the off-diagonal terms of \mathbf{D} . The following simple example illustrates the method for determining the elements of the transmission matrix.

Example 1: Consider two sources (U_1 and U_2) and one relay (R_1) in a $(2, 1)$ cooperative wireless network shown in Fig. 2. Let the size of the finite field for the network coding be 2, $q = 2$. In the broadcasting phase, source U_1 transmits message x_1 to the BS and relay R_1 , while U_2 transmits message x_2 to the BS and R_1 . The relay overhears, decodes, and then linearly combines the decoded messages to generate a parity message that is forwarded to the BS. Thus, the BS receives three messages: x_1 and x_2 from the respective sources and a

TABLE II
DETERMINATION OF THE TRANSMISSION MATRIX FOR ALL CASES
OF FAILURES FOR $q = 2$, WHERE "O" INDICATES NO OUTAGE,
"×" INDICATES AN OUTAGE, AND "-" INDICATES DON'T CARE

U_1 -BS	U_2 -BS	D	U_1 - R_1	U_2 - R_1	R_1 -BS	P
O	O	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	O	O	O	(1 1)
O	×	$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$	O	×	O	(1 0)
×	O	$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$	×	O	O	(0 1)
×	×	$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$	×/-	×/-	O/×	(0 0)

parity message from the relay. This transmission mechanism is depicted in Fig. 2.

The transmission matrix in this example is given by

$$\begin{bmatrix} y_{1,1} \\ y_{2,1} \\ y_{1,2} \end{bmatrix} = \begin{bmatrix} \alpha_{11} & 0 \\ 0 & \alpha_{22} \\ \beta_{11} & \beta_{12} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \quad (5)$$

Here, the connectivities of the channel links (U_1 -BS and U_2 -BS) are represented by α_{11} and α_{22} in the transmission matrix. If a channel link (U_1 -BS) or (U_2 -BS) incurs an outage, then its associated connectivity, α_{11} or α_{22} , will be set to zero; otherwise, this element is set to one. Similarly, two elements β_{11} and β_{12} represent the joint factors of the qualities of the three channel links: ($U_1 - R_1$), ($U_2 - R_1$), and (R_1 -BS). If both links, ($U_2 - R_1$) and (R_1 -BS), do not simultaneously incur outages, β_{12} is set one; conversely, if either of the two links undergoes an outage, β_{11} will be set to zero. Thus, the transmission matrix will be determined by the condition of all five channel links in the wireless network, and if the transmission matrix at a given condition has full rank, the BS can successfully decode the two source messages x_1 and x_2 . Table II summarizes all outage events of the transmission matrix for the (2,1) cooperative wireless network in which there are two sources and one relay. ■

B. Modeling of Random Elements

In this subsection, we provide techniques for defining the elements of the transmission matrix as random variables. We assume that all outage events are mutually independent from each other, which is reasonable for typical wireless networks. Then, the probability distribution of the elements can be determined based on the outage probabilities of the wireless channels as follows. First, the probability distribution for each diagonal element of **D** is modeled using the outage probability of the source-to-BS channels. Second, by simultaneously considering the outage events in both channels (i.e., source-relay and relay-BS), we determine the probability distribution for each element of **P**.

To model each diagonal element of **D**, the probability of the i th diagonal element α_{ii} , $i \in \{1, 2, \dots, N\}$, can be defined from the set of possible outage events between the sources and the BS in the broadcasting phase as:

$$\Pr\{\alpha_{ii} = \theta\} = \begin{cases} \delta_{U_i,BS} & \text{if } \theta = 0, \\ 1 - \delta_{U_i,BS} & \text{if } \theta = 1, \end{cases} \quad (6)$$

where $\delta_{U_i,BS}$ the outage probability defined in (1) where an outage occurs in the single link between the i th source U_i and the BS.

Next, to model each element of the combination matrix **P**, we consider two types of probability distributions. The first is to permit the nonzero values of each element in **P** to be uniformly distributed. This distribution is reasonable, considering the recent result in [39] where it is acknowledged that a uniform distribution for linear network coding provides various benefits, including decentralized operation and robustness to network changes or link failures in multisource, multicast networks. The second is to allow the nonzero value to be predetermined. This specific value can be set using MDS codes [40] in coding theory. It is well known that MDS codes achieve the Singleton bounds. This supports the consideration of MDS codes for optimum reconstruction performance. In the latest literature, Rebelatto *et al.* [7] have proved that a systematic MDS code generator matrix, operating over sufficiently large finite fields such as the transmission matrix, is sufficient for obtaining full diversity in cooperative networks. However, MDS codes use a large field size so that may result in excessive complexity, especially in the cases where the dimension of the code is large.

1) *Uniform Distribution*: When modeling the elements of the combination matrix **P**, we have to consider the outage events in both the source-to-relay and relay-to-BS links, as the occurrence of either or both of these events will prevent the relay from delivering the source message to the BS. Let $\bar{\mathcal{E}}_j$ and \mathcal{E}_j denote the nonoccurrence and occurrence of an outage from the j th relay R_j , $j \in \{1, 2, \dots, M\}$, to the BS, respectively. Thus, both probabilities are: $\Pr\{\bar{\mathcal{E}}_j\} = 1 - \delta_{R_j,BS}$ and $\Pr\{\mathcal{E}_j\} = \delta_{R_j,BS}$. Because the outage event from a source to a relay is independent of any other outage events, the conditional probability $\Pr\{\beta_{ji} = \theta | \bar{\mathcal{E}}_j\}$ of this element of the combination matrix **P** can be modeled as

$$\Pr\{\beta_{ji} = \theta | \bar{\mathcal{E}}_j\} = \begin{cases} \delta_{U_i,R_j} & \text{if } \theta = 0, \\ (1 - \delta_{U_i,R_j}) / (q - 1) & \text{if } \theta \neq 0, \end{cases} \quad (7)$$

where δ_{U_i,R_j} denotes the probability that the outage occurs from the i th source U_i to the j th relay R_j . Each outage probability δ_{U_i,R_j} can be determined independently in (1).

In (7), the elements of **P** are nonzero when both outage events do not occur simultaneously; however, when an outage event from the j th relay to the BS occurs, i.e., when \mathcal{E}_j is true, the conditional probability is set as $\Pr\{\beta_{ji} = 0 | \mathcal{E}_j\} = 1$, regardless of the condition of the outage event (source-relay).

2) *MDS Distribution*: Next, we consider modeling the elements of **P** based on the systematic generator matrix of MDS codes. The difference from the aforementioned uniform distribution is that the nonzero value of each element should be taken from the pertinent value of a predefined MDS code. In this subsection, we refer to this as the MDS distribution. By considering the MDS distribution, we can compare its reconstruction performance to that of the uniform distribution given in (7). For the MDS distribution, the conditional probability $\Pr\{\beta_{ji} = \theta | \bar{\mathcal{E}}_j\}$ defined similarly to that in (7) is given by

$$\Pr\{\beta_{ji} = \theta | \bar{\mathcal{E}}_j\} = \begin{cases} \delta_{U_i,R_j} & \text{if } \theta = 0, \\ 1 - \delta_{U_i,R_j} & \text{if } \theta = \chi, \\ 0 & \text{otherwise,} \end{cases} \quad (8)$$

where χ denotes the coefficient that is predefined from the systematic generator matrix of MDS codes. To generate this code, we used the software application SAGE [41]. For $N = 8$ and $M = 4$, for example, the 4×8 submatrix of the systematic MDS code is:

$$\begin{bmatrix} 9 & 13 & 14 & 7 & 2 & 15 & 13 & 12 \\ 15 & 3 & 9 & 12 & 12 & 10 & 12 & 2 \\ 14 & 9 & 12 & 7 & 8 & 1 & 3 & 7 \\ 4 & 5 & 5 & 10 & 9 & 3 & 4 & 1 \end{bmatrix}. \quad (9)$$

In example (9), the conditional probability $\Pr\{\beta_{11} = 9|\bar{\mathcal{E}}_1\}$ is $1 - \delta_{U_1, R_1}$, and for any $\theta \in \mathbb{F}_{16} \setminus \{0, 9\}$, it is set to zero, i.e., $\Pr\{\beta_{11} = \theta|\bar{\mathcal{E}}_1\} = 0$. Based on this, we can investigate the improvement in the reconstruction performance that is achieved when using MDS codes in a cooperative wireless network.

Remark 1: In this work, we assume that all the inter-node channels are independent from each other. Thereby, probability distributions of random elements are defined independently. If channel correlations are considered, the distributions, i.e., (6)–(8), should be modeled as a joint distribution corresponding to the channel correlation. Using joint distributions, we can evaluate the performance of correlated wireless network coding schemes. Example 3 in Section V-B of this paper shows that the proposed framework can be extended to correlation cases. A generalized version of the proposed framework for channel correlations will be another direction of future research.

V. UPPER BOUND ON RECONSTRUCTION OF MESSAGES

If a transmission matrix for a dynamic network topology randomly generated using the probability distributions given in (6)–(8) has full rank, the BS can uniquely decode all messages from all sources. In this section, we aim to derive an upper bound on the decoding failure probability, and the dimension of the nullspace of the random transmission matrix of an (N, M) cooperative wireless network. We then connect them to investigate the manner in which network coding performance varies based on wireless channel conditions, the number of relays, field sizes, and the positions of nodes deployed in a 2D space. Throughout this paper, we use the random transmission matrix as a bold face, i.e., \mathbf{A} , while the realized transmission matrix in sans-serif style, i.e., A .

We define the dimension of the nullspace within the column space of a transmission matrix as follows. Let \mathbf{A} be an $(N+M) \times N$ matrix over the finite field with size q as \mathbb{F}_q . Based on linear algebra theory, the columns A_1, \dots, A_N of \mathbf{A} are linearly dependent if and only if a vector $\mathbf{c} = (c_1, \dots, c_N) \in \mathbb{F}_q^N$ such that exists, with at least one nonzero c_i , such that

$$\sum_{i=1}^N c_i A_i = 0. \quad (10)$$

Definition 1. (Number of Nonzero Coefficient Vectors): Let $L(\mathbf{A})$ be the number of all such nonzero vectors \mathbf{c} belonging to the nullspace of the given matrix \mathbf{A} . Let the column rank of

a realized transmission matrix be $\text{rank}(\mathbf{A})$. Thus, $L(\mathbf{A})$ can be represented as

$$L(\mathbf{A}) = q^{N-\text{rank}(\mathbf{A})} - 1. \quad (11)$$

Definition 2. (Nullity): Let $\text{nullity}(\mathbf{A})$ be the dimension of the nullspace in the column space of \mathbf{A} .

Proposition 3: For a random matrix \mathbf{A} , the expectation of the nullity of \mathbf{A} is upper bounded by $\mathbb{E}[\text{nullity}(\mathbf{A})] \leq \log_q(\mathbb{E}[L(\mathbf{A})] + 1)$, where $\mathbb{E}[\cdot]$ denotes the expectation.

Proof: For any $(N+M) \times N$ matrix \mathbf{A} , we follow $\text{nullity}(\mathbf{A}) = N - \text{rank}(\mathbf{A})$, known as the rank-nullity theorem of linear algebra [42]. Considering the expectation for a random transmission matrix in both sides of (11), we obtain the following upper bound using Jensen’s inequality:

$$\begin{aligned} \mathbb{E}[\text{nullity}(\mathbf{A})] &:= N - \mathbb{E}[\text{rank}(\mathbf{A})] \\ &= \mathbb{E}[\log_q(L(\mathbf{A}) + 1)] \\ &\leq \log_q(\mathbb{E}[L(\mathbf{A})] + 1). \end{aligned} \quad (12)$$

The proof of Proposition 3 is complete. ■

Theorem 4: Let P_{fail} be the decoding failure probability for the reconstruction of source messages. Then, $P_{fail} \leq \min\{1, 1/(q-1)\mathbb{E}[L(\mathbf{A})]\}$.

Proof: The probability P_{fail} is defined and upper bounded by

$$\begin{aligned} P_{fail} &= \Pr\{\text{rank}(\mathbf{A}) < N\} \\ &= \Pr\left\{\exists \mathbf{c} : \sum_{i=1}^N c_i A_i = 0\right\} \\ &\stackrel{(a)}{\leq} \sum_{\mathbf{c} \in \mathbb{F}_q^N \setminus \{0^T\}} \Pr\{\mathbf{A}\mathbf{c} = 0^T\} \\ &= \mathbb{E}[L(\mathbf{A})]. \end{aligned} \quad (13)$$

where inequality (a) is due to the union bound; note that $\mathbb{E}[L(\mathbf{A})] = \sum_{\mathbf{c} \in \mathbb{F}_q^N \setminus \{0^T\}} \Pr\{\mathbf{A}\mathbf{c} = 0^T\}$. Then, the upper bound on the probability P_{fail} can be tightened as

$$P_{fail} \leq \min\left\{1, \frac{1}{q-1}\mathbb{E}[L(\mathbf{A})]\right\}, \quad (14)$$

where the $1/(q-1)$ factor is due to the following reason. Suppose a nonzero vector \mathbf{c} exists such that $\mathbf{A}\mathbf{c} = 0^T$. Then, other $q-2$ nonzero vectors $\theta\mathbf{c}, \theta^2\mathbf{c}, \dots, \theta^{q-2}\mathbf{c}$ certainly exist for a primitive element $\theta \in \mathbb{F}_q \setminus \{0\}$, where each satisfies $\mathbf{A}\theta^i\mathbf{c} = 0^T$ for $i \in \{1, \dots, q-2\}$. Then, we note $\bigcup_{\mathbf{c}_1 \in \{\theta\mathbf{c}, \dots, \theta^{q-2}\mathbf{c}\}} \{\mathbf{A} : \mathbf{A}\mathbf{c}_1 = 0^T\} = \{\mathbf{A} : \mathbf{A}\mathbf{c} = 0^T\}$. ■

Remark 2: Proposition 3 and Theorem 4 provide a ground-work novel performance evaluation framework of cooperative wireless network coding schemes. These results enable us to calculate the decoding failure probability without an exhaustive search of all possible individual cases. They are new key steps that enable the evaluation framework to be computationally efficient, are not available in the literature, for example [5]–[8].

Next, we will derive $\mathbb{E}[L(\mathbf{A})]$ for three types of cooperative wireless networks.

A. Homogeneous and Heterogeneous Connectivity

In this subsection, we aim to find $\mathbb{E}[L(\mathbf{A})]$ for two cases: i) *homogeneous connectivity* in which all outage probabilities in the wireless network are equal; i.e., $\delta = \delta_{U_i,BS} = \delta_{U_i,R_j} = \delta_{R_j,BS}$ given in (1) and (2) for $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$, assuming that all the channel qualities in networks are equal, and ii) *heterogeneous connectivity* in which two types of outage probabilities exist, i.e., $\delta_1 = \delta_{U_i,BS} = \delta_{U_i,R_j}$ and $\delta_2 = \delta_{R_j,BS}$, with each outage probability assumed to be merely a function of transmit power. Note that each element of a random matrix follows the probability distributions defined in (6) and (7).

Let S_k denote the probability, $S_k := \Pr\{\sum_{i=1}^k \beta_{ji} = 0\}$, for the sum of the first k random elements, $k \in \{1, 2, \dots, N\}$, in the j th row of a combination matrix \mathbf{P} . For the homogeneous case, Lemma 5 provides this probability S_k .

Lemma 5: For the homogeneous connectivity with the distributions (6) and (7), the probability S_k is given by

$$S_k = \delta + (1 - \delta) \left(q^{-1} + (1 - q^{-1}) \left(1 - \frac{1 - \delta}{1 - q^{-1}} \right)^k \right). \quad (15)$$

Proof: See Appendix A. \blacksquare

Before attempting to derive $\mathbb{E}[L(\mathbf{A})]$ of a random matrix \mathbf{A} from Lemma 5, recall that $L(\mathbf{A})$ is the number of all nonzero vectors \mathbf{c} satisfying the linear dependency in (10). The following Proposition 6 gives $\mathbb{E}[L(\mathbf{A})]$ for the homogeneous (N, M) wireless cooperative network.

Proposition 6: Given an (N, M) cooperative network with the homogeneous connectivity based on some outage probability δ , $\mathbb{E}[L(\mathbf{A})]$ of a $(N + M) \times N$ random transmission matrix \mathbf{A} over the finite field \mathbb{F}_q is

$$\mathbb{E}[L(\mathbf{A})] = \sum_{k=1}^N \binom{N}{k} (q-1)^k \delta^k \left[\delta + (1 - \delta) \left(q^{-1} + (1 - q^{-1}) \left(1 - \frac{1 - \delta}{1 - q^{-1}} \right)^k \right) \right]^M. \quad (16)$$

Proof: See Appendix B. \blacksquare

Consider $\mathbb{E}[L(\mathbf{A})]$ under the heterogeneous case; i.e., $\delta_1 = \delta_{U_i,BS} = \delta_{U_i,R_j}$ and $\delta_2 = \delta_{R_j,BS}$. In the following section, we aim to study the manner in which the outage probabilities δ_1 and δ_2 affect the recovery performance, assuming that these outage probabilities rely on the transmit power at the sources and relays.

Proposition 7: Given the heterogeneous (N, M) cooperative network defined by the two outage probabilities δ_1 and δ_2 , $\mathbb{E}[L(\mathbf{A})]$ of a $(N + M) \times N$ random transmission matrix \mathbf{A} over finite fields \mathbb{F}_q is given by

$$\mathbb{E}[L(\mathbf{A})] = \sum_{k=1}^N \binom{N}{k} (q-1)^k \delta_1^k \left[\delta_2 + (1 - \delta_2) \left(q^{-1} + (1 - q^{-1}) \left(1 - \frac{1 - \delta_1}{1 - q^{-1}} \right)^k \right) \right]^M. \quad (17)$$

The proof is omitted. However, it can be proved by following the formalism given in Proposition 6, using two outage

probabilities, δ_1 and δ_2 , instead of single outage probability as in Proposition 6.

B. General Connectivity

Thus far, we have obtained $\mathbb{E}[L(\mathbf{A})]$ of a random transmission matrix \mathbf{A} for homogeneous and heterogeneous cases. In this subsection, we extend it to a more general case where δ_{U_i,R_j} , $\delta_{R_j,BS}$, and $\delta_{U_i,BS}$ are used as defined in Section III. Outage probabilities for the wireless links are obtained using (1) and (2), which are functions of transmit power and variance of the channel gain. After obtaining the outage probability for the each link, we determine the probability distributions for all elements in \mathbf{A} . We call this the *general connectivity* case. Since it involves an exhaustive search of combinations of column vectors in a random matrix, the general connectivity requires a more complicated computation than that of the homogeneous and heterogeneous connectivity to derive $\mathbb{E}[L(\mathbf{A})]$ where the proposed approach using the upper bound on the dimension of nullspace will be most effective.

Proposition 8: Given an (N, M) cooperative network with the general connectivity based on the outage probabilities defined in (1) and (2), $\mathbb{E}[L(\mathbf{A})]$ of a $(N + M) \times N$ random transmission matrix \mathbf{A} over finite fields \mathbb{F}_q is

$$\mathbb{E}[L(\mathbf{A})] = \sum_{k=1}^N (q-1)^k Q_k, \quad (18)$$

where $Q_k := \sum_{l=1}^{|\mathcal{L}_k|} Q_{k,l}$, $l \in \{1, 2, \dots, |\mathcal{L}_k|\}$, $|\mathcal{L}_k| := \binom{N}{k}$, and $\mathcal{L}_{k,l}$ is the l th entry of a set \mathcal{L}_k . Let \mathcal{L}_k denote the collection of the sets of k distinct indices among $[N] := \{1, 2, \dots, N\}$, i.e., $\mathcal{L}_k := \{\{\lambda_1, \lambda_2, \dots, \lambda_k\} : \lambda_i \in \{1, 2, \dots, N\}, \lambda_i \neq \lambda_j, i \neq j\}$. $Q_{k,l} := \Pr\{\sum_{i \in \mathcal{L}_{k,l}} c_i A_i = 0\}$.

Proof: See Appendix C. \blacksquare

We use Proposition 8 to obtain $\mathbb{E}[L(\mathbf{A})]$ for $q = 2$ in a $(2, 1)$ cooperative wireless network as follows.

Example 2: Let us consider a $(2, 1)$ cooperative wireless network for $q=2$, $N=2$, and $M=1$. There are three nonzero vectors \mathbf{c} in \mathbb{F}_2^2 : (10), (01), and (11). For each nonzero vector, we obtain the probability $Q_{k,l}$ as follows. First, the probability $Q_{1,1}$ is

$$\begin{aligned} Q_{1,1} &= \Pr\{c_1 A_1 = 0\} \\ &= \Pr\{\alpha_{11} = 0\} \Pr\{\beta_{11} = 0\} \\ &= \delta_{U_1,BS} (\delta_{R_1,BS} + (1 - \delta_{R_1,BS}) \delta_{U_1,R_1}). \end{aligned} \quad (19)$$

The probability $Q_{1,2}$ is

$$\begin{aligned} Q_{1,2} &= \Pr\{c_2 A_2 = 0\} \\ &= \Pr\{\alpha_{22} = 0\} \Pr\{\beta_{12} = 0\} \\ &= \delta_{U_2,BS} (\delta_{R_1,BS} + (1 - \delta_{R_1,BS}) \delta_{U_2,R_1}). \end{aligned} \quad (20)$$

The probability $Q_{2,1}$ is

$$\begin{aligned} Q_{2,1} &= \Pr\{c_1 A_1 + c_2 A_2 = 0\} \\ &= \Pr\{\alpha_{11} = 0\} \Pr\{\alpha_{22} = 0\} \Pr\{\beta_{11} + \beta_{12} = 0\} \\ &= \delta_{U_1,BS} \delta_{U_2,BS} (\delta_{R_1,BS} \\ &\quad + (1 - \delta_{R_1,BS}) \Pr\{\beta_{11} + \beta_{12} = 0 | \bar{\mathcal{E}}_1\}). \end{aligned} \quad (21)$$

In this example, $\mathbb{E}[L(\mathbf{A})]$ is then given by

$$\mathbb{E}[L(\mathbf{A})] = Q_{1,1} + Q_{1,2} + Q_{2,1}. \quad (22)$$

In addition, it would be intriguing to determine if the proposed evaluation framework developed thus far can be extended to cases where the outages between different links are not independent, but correlated. Such cases may occur when the channels between two nodes are not perfectly orthogonal. Then, such a case becomes an interesting problem to show how the proposed evaluation framework can be utilized to compute the decoding failure probability in the correlated link outages cases. In general, this is a difficult task and would require an additional research paper to effectively provide all details. These details will be provided in a future work. In this paper, we aim to show that the framework is extendible to correlated link outage cases.

For this purpose, we again utilize Proposition 8 to compute $\mathbb{E}[L(\mathbf{A})]$ and extend Example 2 for correlated cases. The outage probabilities are not independent from each other. This can be addressed by considering a joint probability distribution for the random matrix. Using the joint probability distribution, we can again compute the last line of (44) in Appendix, instead of the product of probabilities. This is the main change that allows correlated cases to extend the proposed evaluation framework. In Example 3, given the joint probability distributions, we compute $Q_{1,1}$, $Q_{1,2}$, and $Q_{2,1}$ as shown in Example 2.

Example 3: Consider a (2, 1) cooperative wireless network for $q = 2$, $N = 2$, and $M = 1$. There are two sets of channel correlations that are assumed in this example. The first set of correlated channels is between $U_1 - R_1$ and $U_2 - R_1$; the second set of correlated channels is between U_1 -BS and U_2 -BS. We assume that all other combinations of channels are mutually independent. Note that both sets of correlations occur in the broadcasting phase. A pair of two outage events, U_1 -BS and U_2 -BS, create a joint probability as $\Pr\{\alpha_{11} = \theta_1, \alpha_{22} = \theta_2\} = \Theta_{\theta_1, \theta_2}$ for each $(\theta_1, \theta_2) \in \mathbb{F}_2^2$, where $\sum_{\theta_1, \theta_2} \Theta_{\theta_1, \theta_2} = 1$. For example, when both channels are simultaneously successful during the broadcasting phase, we can set the particular probability as $\Pr\{\alpha_{11} = 1, \alpha_{22} = 1\} = \Theta_{1,1}$. Similarly, other probabilities can be defined according to the conditions of the two outage events, U_1 -BS and U_2 -BS. Note that the conditional joint probability is set as $\Pr\{\beta_{11} = 0, \beta_{12} = 0 | \mathcal{E}_1\} = 1$ since the two elements, β_{11} and β_{12} , are zero when the channel outage between R_1 and the BS occurs. In addition, a set of the two outage events, $U_1 - R_1$ and $U_2 - R_1$, can determine the values of the two random elements β_{11} and β_{12} once the channel outage between R_1 and BS does not occur, i.e., when $\bar{\mathcal{E}}_1$ is true. In this case, let the conditional joint probability distribution be known acknowledged and given as $\Pr\{\beta_{11} = \gamma_1, \beta_{12} = \gamma_2 | \bar{\mathcal{E}}_1\} = \Gamma_{\gamma_1, \gamma_2}$ for each $(\gamma_1, \gamma_2) \in \mathbb{F}_2^2$, where $\sum_{\gamma_1, \gamma_2} \Gamma_{\gamma_1, \gamma_2} = 1$. For $q = 2$, Table III summarizes this conditional joint probability distribution according to the conditions of the two outage events $U_1 - R_1$ and $U_2 - R_1$.

For three nonzero vectors \mathbf{c} in \mathbb{F}_2^2 , we can again compute $Q_{1,1}$, $Q_{1,2}$, and $Q_{2,1}$. The computation of $Q_{1,1}$ and $Q_{1,2}$ is

TABLE III
DETERMINATION OF THE CONDITIONAL JOINT PROBABILITY FOR THE TWO ELEMENTS β_{11} AND β_{12} , WHERE "O" INDICATES NO OUTAGE AND "X" INDICATES AN OUTAGE

(U_1-R_1, U_2-R_1)	(β_{11}, β_{12})	$\Pr\{\beta_{11} = \gamma_1, \beta_{12} = \gamma_2 \bar{\mathcal{E}}_1\}$
(O, O)	(1, 1)	$\Gamma_{1,1}$
(O, X)	(1, 0)	$\Gamma_{1,0}$
(X, O)	(0, 1)	$\Gamma_{0,1}$
(X, X)	(0, 0)	$\Gamma_{0,0}$

straightforward because of our assumption that the two sets of channel correlations are independent. The results are

$$Q_{1,1} = (\Theta_{0,0} + \Theta_{0,1}) (\delta_{R_1, BS} + (1 - \delta_{R_1, BS}) (\Gamma_{0,0} + \Gamma_{0,1})) \quad (23)$$

and

$$Q_{1,2} = (\Theta_{0,0} + \Theta_{1,0}) (\delta_{R_1, BS} + (1 - \delta_{R_1, BS}) (\Gamma_{0,0} + \Gamma_{1,0})) \quad (24)$$

The computation of $Q_{2,1}$ is given as follows:

$$\begin{aligned} Q_{2,1} &= \Pr\{c_1 A_1 + c_2 A_2 = 0\} \\ &= \Pr\{\alpha_{11} = 0, \alpha_{22} = 0, \beta_{11} + \beta_{12} = 0\} \\ &= \Pr\{\alpha_{11} = 0, \alpha_{22} = 0, \beta_{11} + \beta_{12} = 0 | \mathcal{E}_1\} \Pr\{\mathcal{E}_1\} \\ &\quad + \Pr\{\alpha_{11} = 0, \alpha_{22} = 0, \beta_{11} + \beta_{12} = 0 | \bar{\mathcal{E}}_1\} \Pr\{\bar{\mathcal{E}}_1\} \\ &\stackrel{(a)}{=} \Pr\{\alpha_{11} = 0, \alpha_{22} = 0\} \\ &\quad \times (\Pr\{\beta_{11} + \beta_{12} = 0 | \mathcal{E}_1\} \Pr\{\mathcal{E}_1\} \\ &\quad + \Pr\{\beta_{11} + \beta_{12} = 0 | \bar{\mathcal{E}}_1\} \Pr\{\bar{\mathcal{E}}_1\}) \\ &= \Theta_{0,0} (\delta_{R_1, BS} + (\Gamma_{0,0} + \Gamma_{1,1}) (1 - \delta_{R_1, BS})), \quad (25) \end{aligned}$$

where equality (a) is based on the fact that the relation between the two sets, $(\alpha_{11}, \alpha_{22})$ and (β_{11}, β_{12}) , is independent. We finally obtain $\mathbb{E}[L(\mathbf{A})] = Q_{1,1} + Q_{1,2} + Q_{2,1}$ for correlated cases by using the proposed evaluation framework. ■

C. Asymptotic Nullity

In practice, the computation of (18) requires a significant amount of time, because all combinations of column vectors must be collected as the number of sources and relays increases. In larger networks, this process is complicated; therefore, it will be beneficial to reduce the resources required for this computation. In this subsection, we aim to obtain an asymptotic form of (18) for utilization in large-scale networks.

As previously mentioned, the homogeneous connectivity scheme is a specific case among general connectivity schemes. We can exhibit a simple form of $\mathbb{E}[L(\mathbf{A})]$ in terms of Q_k for the homogeneous topology of cooperative networks. Based on this approach, we can obtain an asymptotic result of (18) in general connectivity schemes. Let us consider $\mathbb{E}[L(\mathbf{A})]$ for $q = 2$ in the homogeneous connectivity. Thus, $\mathbb{E}[L(\mathbf{A})] = \sum_{k=1}^N Q_k$ in (18). Using (44), Q_1 is given by

$$Q_1 = N \delta S_1. \quad (26)$$

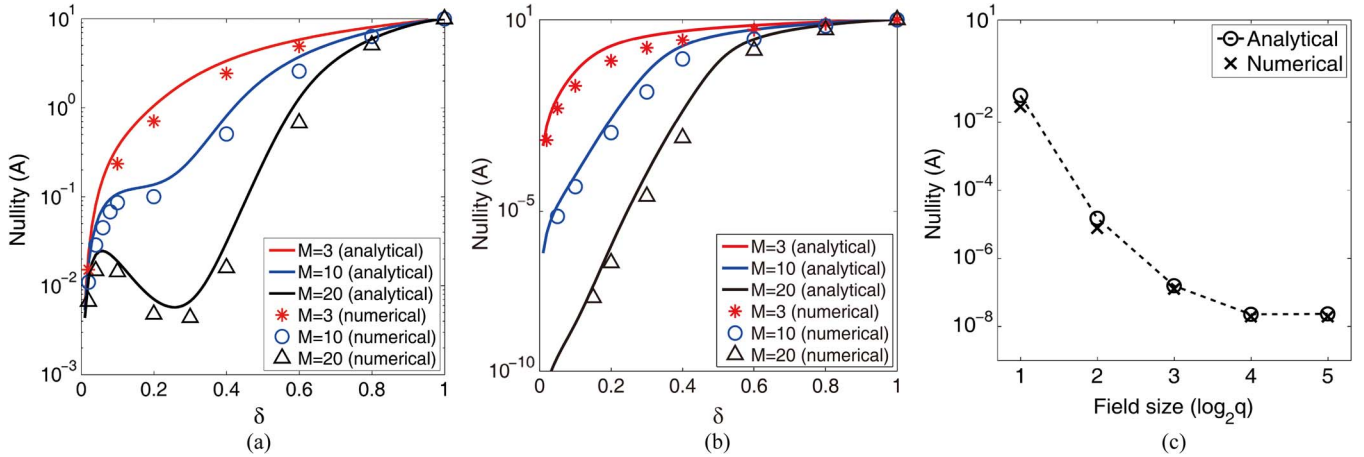


Fig. 3. The nullity of random matrix \mathbf{A} for a homogeneous $(10, M)$ cooperative wireless network with $N = 10$ and $M = 3, 10,$ and 20 . Solid lines indicate the upper bounds on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ using Proposition 6, and markers indicate numerically simulated results of $\mathbb{E}[\text{nullity}(\mathbf{A})]$, respectively: (a) $q = 2$ and (b) $q = 4$. Fig. 3(c) shows $\mathbb{E}[\text{nullity}(\mathbf{A})]$ with different field sizes from $q = 2$ to 32 under fixed $N = 10, M = 10,$ and $\delta = 0.05$.

For $k = 2$, we have $Q_2 = \binom{N}{2} \delta^2 S_2$. We further obtain $Q_3 = \binom{N}{3} \delta^3 S_3$ for $k = 3$. The general expression of Q_k is given by

$$Q_k = \binom{N}{k} \delta^k S_k. \quad (27)$$

In this case, $\mathbb{E}[L(\mathbf{A})]$ in (18) is

$$\mathbb{E}[L(\mathbf{A})] = \sum_{k=1}^N \binom{N}{k} \delta^k S_k. \quad (28)$$

In high SNR regions, assuming δ is small, an approximation form of (28) is obtained as

$$\begin{aligned} \mathbb{E}[L(\mathbf{A})] &= \sum_{k=1}^N Q_k \\ &\stackrel{(a)}{\approx} \binom{N}{1} \delta S_1 + \binom{N}{2} \delta^2 S_2, \end{aligned} \quad (29)$$

where (a) is based on the fact that the order of Q_k for $k \geq 3$ is greater than two with respect to δ . This approximation indicates that for the computation of $\mathbb{E}[L(\mathbf{A})]$, two terms Q_1 and Q_2 are sufficient in high SNR regions. Therefore, in the high SNR regions, $\mathbb{E}[L(\mathbf{A})]$ converges to the second order of the transmit SNR. For any finite field and the general connectivity, this approximation is satisfied.

Corollary 9: Given an (N, M) cooperative network with general connectivity having the distributions (6) and (7), $\mathbb{E}[L(\mathbf{A})]$ is simplified in the high SNR regime

$$\mathbb{E}[L(\mathbf{A})] \approx (q-1)Q_1 + (q-1)^2 Q_2. \quad (30)$$

Remark 3: Proposition 8 provides a closed-form solution to the expectation of the number of nonzero vectors in the nullspace of the random transmission matrix \mathbf{A} . This enables us to evaluate the performance of a general network with randomly deployed nodes, without separately processing each count of transmission matrix count. Corollary 9 is a closed-form approximation of (18) that is useful for performance evaluations of large size networks.

VI. NUMERICAL AND SIMULATION RESULTS

In this section, the performance of source message reconstruction at the BS is investigated by utilizing the proposed evaluation framework, i.e., $\mathbb{E}[\text{nullity}(\mathbf{A})]$ and P_{fail} . For the homogeneous connectivity scheme, we employ Proposition 6 to analytically derive the upper bound on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ defined in Section V as a function of the outage probability of the single channel link. We compare the upper bounds with the numerically simulated results of $\mathbb{E}[\text{nullity}(\mathbf{A})]$ as well as P_{fail} . Subsequently, we employ Proposition 8 to investigate the upper bounds of a general cooperative network in which sources and relays are deployed in a 2D space. Furthermore, we show the results of the upper bound on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ and P_{fail} for a given transmission matrix to investigate the impact of the number of relays and the field size of network coding.

Fig. 3 shows analytically obtained upper bounds and numerically averaged results of $\mathbb{E}[\text{nullity}(\mathbf{A})]$ for a random transmission matrix in a $(10, M)$ cooperative wireless network given the homogeneous connectivity scheme, where $N = 10$ and $M = 3, 10,$ and 20 for $q = 2$ in Fig. 3(a), and 4 in Fig. 3(b). We observe that $\mathbb{E}[\text{nullity}(\mathbf{A})]$ increases as the outage probability slightly increases. Based on Fig. 3(b), it is evident that a nonbinary network coding scheme provides superior reconstruction performance for source messages at the BS when compared to binary coding; moreover, increasing the field size of network coding also improves recovery performance. As the outage probability is reduced to zero, $\mathbb{E}[\text{nullity}(\mathbf{A})]$ approaches zero for all field sizes.

Fig. 4 shows the analytically derived upper bounds using Proposition 7 for a heterogeneously connected $(20, 20)$ cooperative wireless network. Regardless of the value of δ_2 , when the outage probability δ_1 approaches one, $\mathbb{E}[\text{nullity}(\mathbf{A})]$ barely reaches 20 for both field sizes, i.e., $q = 2$ and 4 . This indicates that all channel links undergo outage events, causing all elements of the transmission matrix to become zero. In Fig. 4(a), for $q = 2$, there is an oscillation in the proximity of $\delta_1 = 0.3$ such that $\mathbb{E}[\text{nullity}(\mathbf{A})]$ decreases as δ_1 increases to 0.3 , and beyond this point $\mathbb{E}[\text{nullity}(\mathbf{A})]$ increases. This oscillation also appears in Fig. 3(a). This behavior results from the fact that the rows of \mathbf{P} tend to be identical as the outage probabilities

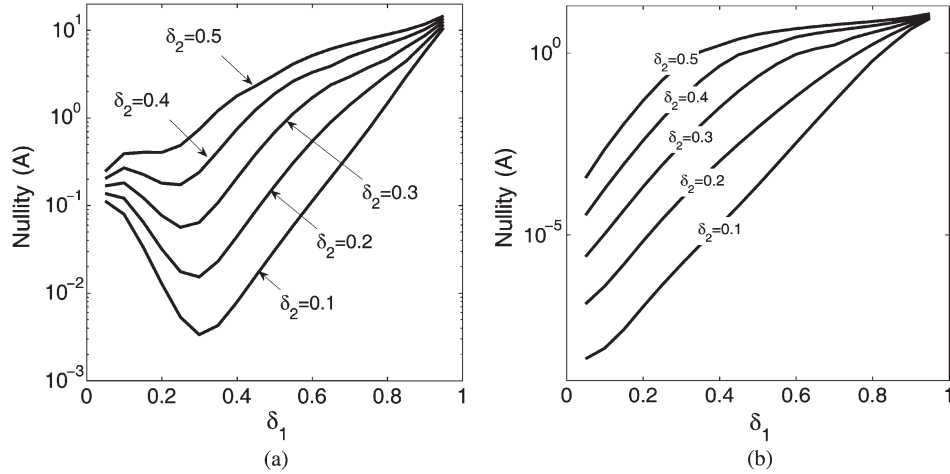


Fig. 4. Upper bounds on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ using Proposition 3 and 7 in a heterogeneous (20, 20) cooperative wireless network with two outage probabilities δ_1 and δ_2 for (a) $q = 2$, and (b) $q = 4$.

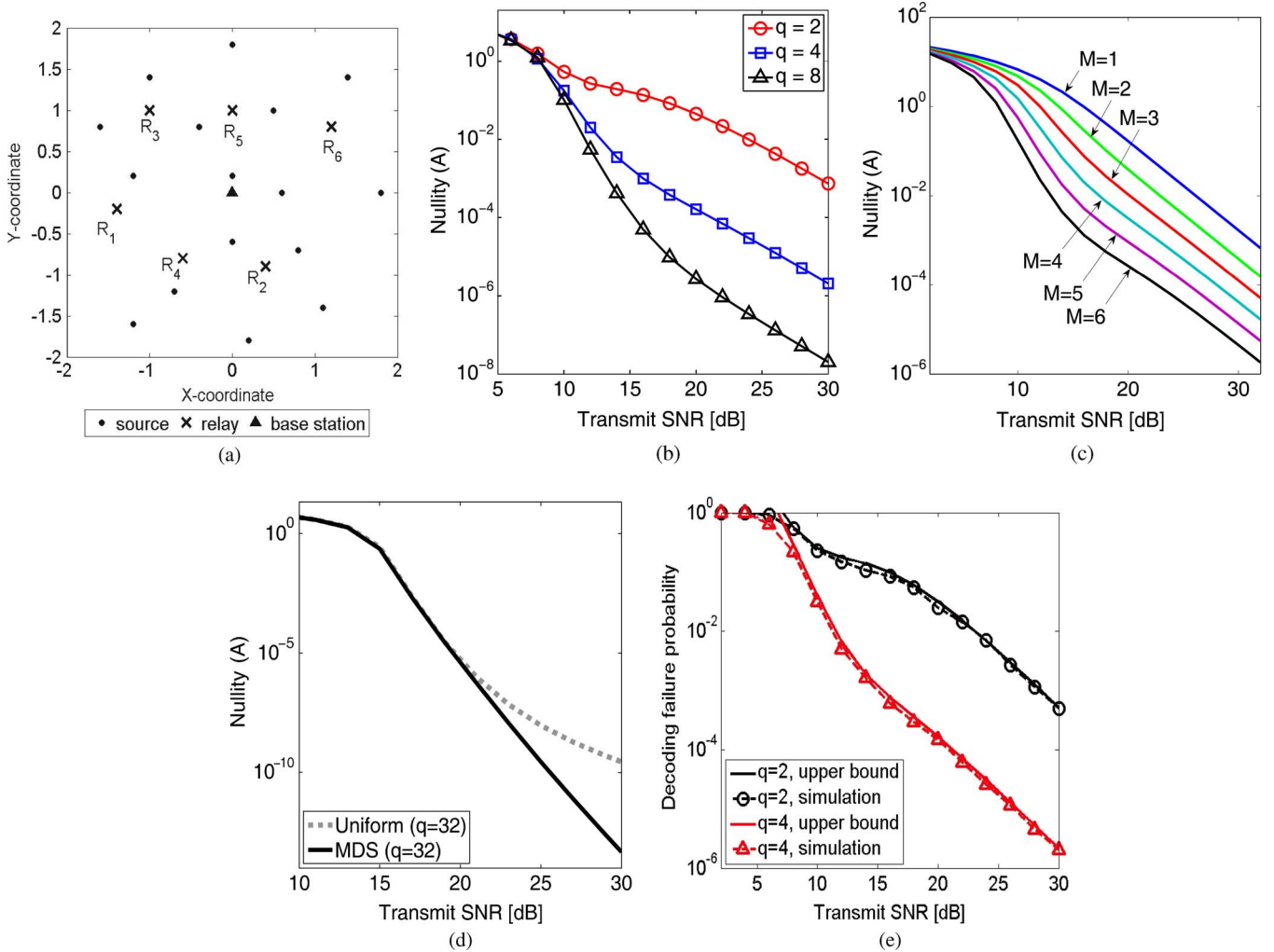


Fig. 5. (a) Location of 16 sources and 6 relays in 2D space for an (16, 6) cooperative wireless network. (b) Results of upper bounds on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ with differing network coding field sizes $q=2, 4$, and 8 , (c) varying the number of relays at $q=4$. (d) Comparison of upper bounds on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ for the uniform and MDS distributions. (e) Comparison of the decoding failure probabilities with the numerical simulation and the upper bound using Proposition 8 for $q=2$ and 4 .

δ_1 and δ_2 approach zero. For $q=4$, however, this behavior disappears owing to the extension of the field size from binary to quaternary.

Now, let us consider the (16, 6) cooperative wireless network shown in Fig. 5(a), in which there are 16 sources and 6 relays:

R_1 through R_6 . We randomly deploy these relays in a 2D space. We assume that all the transmit powers of sources and relays in both transmission phases are equal. Fig. 5(b) shows the upper bound on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ of the transmission matrix for $q = 2, 4$, and 8 ; the benefit of increasing the field size of

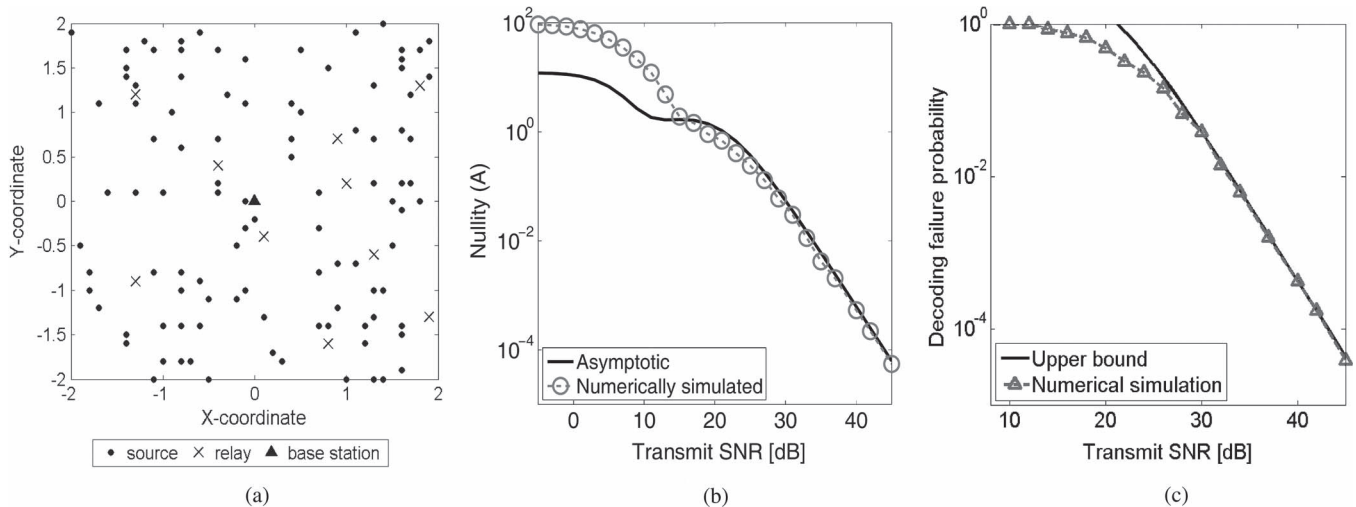


Fig. 6. (a) Locations of 100 sources and 10 relays in a 2D space for an (100, 10) cooperative wireless network, (b) Comparison of $\mathbb{E}[\text{nullity}(\mathbf{A})]$ with numerically simulated result and the upper bound using Corollary 9 with $q = 2$ and the uniform distribution. (c) Comparison of the decoding failure probability with the numerical simulation and the upper bound using Corollary 9.

network coding appears in this scheme. Fig. 5(c) shows the upper bound on $\mathbb{E}[\text{nullity}(\mathbf{A})]$ with respect to the number of relays. When $M = 1$, R_1 is used, while R_1 and R_2 are used as relays for $M = 2$, and relays R_1 , R_2 , and R_3 are used for $M = 3$. For $M = 4, 5$, and 6, we deploy one relay in order. We investigate the impact of the number of relays, as shown in Fig. 5(c), where increasing the number of relays contributes to the increasingly high likelihood of deriving random transmission matrices of full rank.

Other intriguing results indicate that the value of $\mathbb{E}[\text{nullity}(\mathbf{A})]$ differs slightly for the uniform and MDS distributions of the combination matrix defined in Section IV-B; furthermore, the recovery performance obtained using the MDS distribution is superior to that of the uniform distribution in high SNR regions. Comparative results of $\mathbb{E}[\text{nullity}(\mathbf{A})]$ for the two cooperative networks are shown in Fig. 5(d). We observe that there are minimal differences between the uniform and MDS distributions for the recovery performance in the low SNR regions. In particular, the benefit of using the systematic generator of MDS codes appears only in the high SNR regions.

To validate the usefulness of our asymptotic nullity, we consider an (100, 10) cooperative wireless network as a large-scale network in which 100 sources and 10 relays are deployed in the 2D space shown in Fig. 6(a). For Corollary 9, we show that in the high SNR regions, the asymptotic nullity of (30) is similar to the numerical results that were obtained from randomly generated transmission matrices. A comparison of those results is shown in Fig. 6(b). Using the asymptotic nullity, the complexity of (18) can be significantly reduced. In addition, the nullity of the random transmission matrix can be obtained efficiently. Our proposed framework provides the ability to evaluate reconstruction performance in large-scale networks.

Fig. 5(e) shows the comparison of numerically simulated decoding failure probabilities and upper bounds using (14) for an (16, 6) cooperative wireless network with $q = 2$ and 4. The gap between the results is evident in small SNR regions. However, the upper bound on the decoding failure probability is tight in high SNR regions. This behavior is shown in Fig. 6(c)

in which the upper bound is obtained from the approximation form of $\mathbb{E}[\text{nullity}(\mathbf{A})]$ in (30). Based on those results, we show that predicting the performance of source message reconstruction for an (N, M) cooperative wireless network is straightforwardly possible in large-scale networks.

VII. CONCLUSION

In this paper, we considered a cooperative wireless network where N sources are assisted with M relays in two phase transmissions. Our main goal was to propose a new performance analysis framework for evaluating the reconstruction performance of source messages at the BS. To handle dynamic network topologies, we modeled the elements of the transmission matrix as random variables. This enabled us to develop a systematic approach, to avoid the exhaustive evaluations used in DNC and GDNC schemes [5]–[8]. To complete the performance evaluation, we derived two tight upper bounds on the expected dimension of the nullspace of the random transmission matrix, as well as the decoding failure probability. The result is a framework that is more effective than the rank-based method proposed in the previous literature.

Three types of connectivity schemes are considered in this paper, as they make the framework to be general and scalable. They enabled us to show the reconstruction performance of our proposed framework using multiple sources and multiple relays randomly deployed in a 2D space. In addition, it enables us to investigate the impact of the number of relays and the field size of network coding on the system performance; an example is shown in Figs. 5 and 6 for example. In particular, the ability to generate a precise prediction of network coding performance for a network with a large number of sources and relays is a significant benefit. We can formulate challenging scenarios and generate accurate response in an efficient manner, without resorting to extensive computer simulations. For example, we can determine the advantage that using an MDS code, rather than random code, provides when designing the transmission matrices, as relays are added and field sizes are increased; we

can also determine how the position of relays and sources, with respect to the base station locations, affect the performance of cooperative communications. These questions are important engineering inquiries in terms of wireless networks design. These questions, which could not be readily answered in the past, but can now be answered precisely using the proposed framework describe in this paper. In addition, we demonstrate that the proposed framework can be extended to channel correlation cases; however, it is necessary to generalize for any categorization of cooperative wireless network coding schemes.

APPENDIX A
PROOF OF LEMMA 5

For the homogeneous connectivity, we use $\delta = \delta_{U_i, BS} = \delta_{U_i, R_j} = \delta_{R_j, BS}$. The conditional probability $\Pr\{\beta_{ji} = \theta | \bar{\mathcal{E}}_j\}$ of each element β_{ji} is defined using (7), and the other conditional probability can be set as $\Pr\{\beta_{ji} = 0 | \mathcal{E}_j\} = 1$. Using the total probability theorem, the probability $\Pr\{\sum_{i=1}^k \beta_{ji} = 0\}$ can be decomposed by the condition of the outage event \mathcal{E}_j , and then given as follows:

$$\begin{aligned} S_k &= \Pr\{\mathcal{E}_j\} \Pr\left\{\sum_{i=1}^k \beta_{ji} = 0 \middle| \mathcal{E}_j\right\} \\ &\quad + \Pr\{\bar{\mathcal{E}}_j\} \Pr\left\{\sum_{i=1}^k \beta_{ji} = 0 \middle| \bar{\mathcal{E}}_j\right\} \\ &\stackrel{(a)}{=} \delta + (1 - \delta) \Pr\left\{\sum_{i=1}^k \beta_{ji} = 0 \middle| \bar{\mathcal{E}}_j\right\}, \end{aligned} \quad (31)$$

where (a) follows from the fact that $\Pr\{\sum_{i=1}^k \beta_{ji} = 0 | \mathcal{E}_j\} = 1$, as $\Pr\{\beta_{ji} = 0 | \mathcal{E}_j\} = 1$ (note that the conditional probability $\Pr\{\beta_{ji} = \theta | \bar{\mathcal{E}}_j\}$ is independent of this). Let f_k be the probability, i.e., $f_k := \Pr\{\sum_{i=1}^k \beta_{ji} = 0 | \bar{\mathcal{E}}_j\}$. Given the conditional probability defined in (7) and $f_0 = 1$, the probability f_k can be rewritten by [43]

$$\begin{aligned} f_k &= \Pr\left\{\sum_{i=1}^{k-1} \beta_{ji} = 0 \middle| \bar{\mathcal{E}}_j\right\} \Pr\{\beta_{jk} = 0 | \bar{\mathcal{E}}_j\} \\ &\quad + \sum_{\theta \in \mathbb{F}_q \setminus \{0\}} \Pr\left\{\sum_{i=1}^{k-1} \beta_{ji} = \theta \middle| \bar{\mathcal{E}}_j\right\} \Pr\{\beta_{jk} = -\theta | \bar{\mathcal{E}}_j\} \\ &= f_{k-1} \delta + (1 - f_{k-1}) \frac{1 - \delta}{q - 1}. \end{aligned} \quad (32)$$

Let $g_k := f_k - q^{-1}$. By rewriting (32) as a function of g_k , we have a simple closed form:

$$g_k = g_{k-1} \left(1 - \frac{1 - \delta}{1 - q^{-1}}\right). \quad (33)$$

Applying a geometric series to (33), we obtain f_k as

$$f_k = q^{-1} + (1 - q^{-1}) \left(1 - \frac{1 - \delta}{1 - q^{-1}}\right)^k. \quad (34)$$

Finally, the probability S_k can be obtained by substituting (34) into (31) as:

$$S_k = \delta + (1 - \delta) \left(q^{-1} + (1 - q^{-1}) \left(1 - \frac{1 - \delta}{1 - q^{-1}}\right)^k \right). \quad (35)$$

APPENDIX B
PROOF OF PROPOSITION 6

Let us consider a vector $\mathbf{c} = (c_1, \dots, c_N) \in \mathbb{F}_q^N$ in which the first k entries (and only the first k entries) are nonzero, i.e., $\mathbf{c} = (c_1, \dots, c_k, 0, \dots, 0)$. Let P_k be the probability that the sum of the first k column vectors is zero, i.e., $P_k := \Pr\{\sum_{i=1}^k c_i A_i = 0\}$. As (13), $\mathbb{E}[L(\mathbf{A})]$ is given by

$$\begin{aligned} \mathbb{E}[L(\mathbf{A})] &= \sum_{\mathbf{c} \in \mathbb{F}_q^N \setminus \{0^T\}} \Pr\{\mathbf{A}\mathbf{c} = 0^T\} \\ &= \sum_{k=1}^N \binom{N}{k} (q-1)^k P_k. \end{aligned} \quad (36)$$

Since all links in the wireless network are assumed to be spatially and temporally independent, the rows of the transmission matrix are also independent. Thus, P_k is given by

$$\begin{aligned} P_k &= \Pr\left\{\sum_{i=1}^k c_i A_i = 0\right\} \\ &= \prod_{i=1}^k \Pr\{c_i \alpha_{ii} = 0\} \prod_{j=1}^M \Pr\left\{\sum_{i=1}^k c_i \beta_{ji} = 0\right\}. \end{aligned} \quad (37)$$

Let H_k be the probability as $H_k := \Pr\{\sum_{i=1}^k c_i \beta_{ji} = 0\}$. For $k = 1$, it is easy to show $\Pr\{c_1 \beta_{j1} = 0\} = \Pr\{\beta_{j1} = 0\}$ for $c_1 \in \mathbb{F}_q \setminus \{0\}$ because of the multiplication property in finite fields. Next, we prove that $H_k = S_k$ for $k \geq 2$ where $c_1, c_2, \dots, c_k \in \mathbb{F}_q \setminus \{0\}$ denote the k nonzero elements. The probability H_k is represented by

$$\begin{aligned} H_k &= \sum_{\theta \in \mathbb{F}_q} \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = \theta, c_k \beta_{jk} = -\theta\right\} \\ &= \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = 0, c_k \beta_{jk} = 0\right\} \\ &\quad + \sum_{\theta \in \mathbb{F}_q \setminus \{0\}} \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = \theta, c_k \beta_{jk} = -\theta\right\}. \end{aligned} \quad (38)$$

Decomposing the outage event \mathcal{E}_j , (38) can be rewritten by

$$\begin{aligned} H_k &= \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = 0, c_k \beta_{jk} = 0 \middle| \bar{\mathcal{E}}_j\right\} \Pr\{\bar{\mathcal{E}}_j\} \\ &\quad + \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = 0, c_k \beta_{jk} = 0 \middle| \mathcal{E}_j\right\} \Pr\{\mathcal{E}_j\} \\ &\quad + \sum_{\theta \in \mathbb{F}_q \setminus \{0\}} \left(\Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = \theta, c_k \beta_{jk} = -\theta \middle| \bar{\mathcal{E}}_j\right\} \Pr\{\bar{\mathcal{E}}_j\} \right. \\ &\quad \left. + \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = \theta, c_k \beta_{jk} = -\theta \middle| \mathcal{E}_j\right\} \Pr\{\mathcal{E}_j\} \right). \end{aligned} \quad (39)$$

Since $\Pr\{\beta_{ji} = 0 | \mathcal{E}_j\} = 1$, (39) can be represented by

$$\begin{aligned} H_k &= \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = 0, c_k \beta_{jk} = 0 \middle| \bar{\mathcal{E}}_j\right\} \Pr\{\bar{\mathcal{E}}_j\} + \Pr\{\mathcal{E}_j\} \\ &\quad + \sum_{\theta \in \mathbb{F}_q \setminus \{0\}} \Pr\left\{\sum_{i=1}^{k-1} c_i \beta_{ji} = \theta, c_k \beta_{jk} = -\theta \middle| \bar{\mathcal{E}}_j\right\} \Pr\{\bar{\mathcal{E}}_j\}. \end{aligned} \quad (40)$$

Noting that wireless channels are independent from each other under the condition of $\bar{\mathcal{E}}_j$, (40) can be decomposed by

$$\begin{aligned}
 H_k &= \Pr \left\{ \sum_{i=1}^{k-1} c_i \beta_{ji} = 0 \middle| \bar{\mathcal{E}}_j \right\} \Pr \{ c_k \beta_{jk} = 0 | \bar{\mathcal{E}}_j \} \Pr \{ \bar{\mathcal{E}}_j \} \\
 &+ \Pr \{ \mathcal{E}_j \} + \sum_{\theta \in \mathbb{F}_q \setminus \{0\}} \Pr \left\{ \sum_{i=1}^{k-1} c_i \beta_{ji} = \theta \middle| \bar{\mathcal{E}}_j \right\} \\
 &\times \Pr \{ c_k \beta_{jk} = -\theta | \bar{\mathcal{E}}_j \} \Pr \{ \bar{\mathcal{E}}_j \}. \tag{41}
 \end{aligned}$$

Using recursion, the probability H_k is given by

$$\begin{aligned}
 H_k &= \Pr \left\{ \sum_{i=1}^{k-1} \beta_{ji} = 0 \middle| \bar{\mathcal{E}}_j \right\} \Pr \{ \beta_{jk} = 0 | \bar{\mathcal{E}}_j \} \Pr \{ \bar{\mathcal{E}}_j \} \\
 &+ \Pr \{ \mathcal{E}_j \} + \sum_{\theta \in \mathbb{F}_q \setminus \{0\}} \Pr \left\{ \sum_{i=1}^{k-1} \beta_{ji} = \theta \middle| \bar{\mathcal{E}}_j \right\} \\
 &\times \Pr \{ \beta_{jk} = -\theta | \bar{\mathcal{E}}_j \} \Pr \{ \bar{\mathcal{E}}_j \} \\
 &= \sum_{\theta \in \mathbb{F}_q} \Pr \left\{ \sum_{i=1}^{k-1} \beta_{ji} = \theta, \beta_{jk} = -\theta \right\} \\
 &= \Pr \left\{ \sum_{i=1}^k \beta_{ji} = 0 \right\}. \tag{42}
 \end{aligned}$$

Thus, we simply rewrite P_k as follows

$$P_k = \delta^k S_k^M. \tag{43}$$

The proof of Proposition 6 is complete.

APPENDIX C
PROOF OF PROPOSITION 8

For general connectivity, each element of \mathbf{A} have a different the probability distribution. This result in different probabilities $Q_{k,l}$ for that any k column vectors of \mathbf{A} that are linearly dependent. The total number of $Q_{k,l}$ is $|\mathcal{L}_k| := \binom{N}{k}$. We have to consider all different probabilities $Q_{k,l}$ with respect to all sets $\mathcal{L}_{k,l}$. The probability Q_k should be summed over all different probabilities $Q_{k,l}$, i.e., $Q_k := \sum_{l=1}^{|\mathcal{L}_k|} Q_{k,l}$ for $l \in \{1, 2, \dots, |\mathcal{L}_k|\}$ and $k \in \{1, 2, \dots, N\}$. Thus, all $Q_{k,l}$ are enumerated and collected to obtain the probability Q_k , which is derived as follows:

$$\begin{aligned}
 Q_k &= \sum_{l=1}^{|\mathcal{L}_k|} Q_{k,l} = \sum_{l=1}^{|\mathcal{L}_k|} \Pr \left\{ \sum_{i \in \mathcal{L}_{k,l}} c_i A_i = 0 \right\} \\
 &= \sum_{l=1}^{|\mathcal{L}_k|} \prod_{m=1}^k \Pr \{ \alpha_{l_m l_m} = 0 \} \prod_{j=1}^M \Pr \left\{ \sum_{m=1}^k \beta_{jl_m} = 0 \right\}, \tag{44}
 \end{aligned}$$

where l_m is the m th entry of the set $\mathcal{L}_{k,l}$, $m \in \{1, 2, \dots, k\}$. As similarly obtained in (31), (44) can be rewritten as

$$\begin{aligned}
 Q_k &= \sum_{l=1}^{|\mathcal{L}_k|} \prod_{m=1}^k \Pr \{ \alpha_{l_m l_m} = 0 \} \prod_{j=1}^M \left(\delta_{R_j, BS} \right. \\
 &\left. + (1 - \delta_{R_j, BS}) \Pr \left\{ \sum_{m=1}^k \beta_{jl_m} = 0 \middle| \bar{\mathcal{E}}_j \right\} \right). \tag{45}
 \end{aligned}$$

In order to determine $\mathbb{E}[L(\mathbf{A})]$, we count the number of vectors \mathbf{c} having the first k nonzero elements, i.e., $(q - 1)^k$. Finally, we perform the summation over all k , and obtain $\mathbb{E}[L(\mathbf{A})]$ as follows,

$$\mathbb{E}[L(\mathbf{A})] = \sum_{k=1}^N (q - 1)^k Q_k. \tag{46}$$

The proof of Proposition 8 is complete.

REFERENCES

- [1] J. L. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.
- [2] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.
- [3] Y. Chen, S. Kishore, and J. Li, "Wireless diversity through network coding," in *Proc. IEEE WCNC*, Las Vegas, NV, USA, Apr. 2006, pp. 1681–1686.
- [4] D. H. Woldegebreal and H. Karl, "Network-coding based adaptive decode and forward cooperative transmission in a wireless network: Outage analysis," in *Proc. 13th Eur. Wireless Conf.*, Paris, France, Apr. 2007, pp. 1–6.
- [5] M. Xiao and M. Skoglund, "Multiple-user cooperative communications based on linear network coding," *IEEE Trans. Commun.*, vol. 58, no. 12, pp. 3345–3351, Dec. 2010.
- [6] M. Xiao and M. Skoglund, "Design of network codes for multiple-user multiple-relay wireless networks," in *Proc. IEEE ISIT*, Seoul, Korea, Jun. 2009, pp. 2562–2566.
- [7] J. L. Rebelatto, B. F. Uchoa-Filho, Y. Li, and B. Vucetic, "Multiuser cooperative diversity through network coding based on classical coding theory," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 916–926, Feb. 2012.
- [8] J. L. Rebelatto, B. F. Uchoa-Filho, Y. Li, and B. Vucetic, "Adaptive distributed network-channel coding," *IEEE Trans. Wireless Commun.*, vol. 10, no. 9, pp. 2818–2822, Sep. 2011.
- [9] H. Topakkaya and Z. Wang, "Wireless network code design and performance analysis using diversity-multiplexing tradeoff," *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 488–496, Feb. 2011.
- [10] O. Trullols-Cruces, J. M. Barcelo-Ordinas, and M. Fiore, "Exact decoding probability under random linear network coding," *IEEE Commun. Lett.*, vol. 15, no. 1, pp. 67–69, Jan. 2011.
- [11] H. V. Nguyen, S. X. Ng, and L. Hanzo, "Performance bounds of network coding aided cooperative multiuser systems," *IEEE Signal Process. Lett.*, vol. 18, no. 7, pp. 435–438, Jul. 2011.
- [12] J.-T. Seong and H.-N. Lee, "4-ary network coding for two nodes in cooperative wireless networks: Exact outage probability and coverage expansion," *EURASIP J. Wireless Commun. Netw.*, vol. 2012, p. 366, Dec. 2012.
- [13] T. Wang and G. B. Giannakis, "Complex field network coding for multiuser cooperative communications," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 3, pp. 561–571, Apr. 2008.
- [14] T. Islam, A. Nasri, R. Schober, R. K. Mallik, and V. K. Bhargava, "Network coded multi-source cooperative communication in BICM-OFDM networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 9, pp. 3180–3193, Sep. 2012.
- [15] X.-T. Vu, M. D. Renzo, and P. Duhamel, "BER analysis of joint network/channel decoding in block Rayleigh fading channels," in *Proc. IEEE Int. Symp. PIMRC*, London, U.K., Sep. 2013, pp. 698–702.
- [16] R. Youssef and A. Graell i Amat, "Distributed serially concatenated codes for multi-source cooperative relay networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 1, pp. 253–263, Jan. 2011.
- [17] Z. Han, X. Zhang, and H. V. Poor, "High performance cooperative transmission protocols based on multiuser detection and network coding," *IEEE Trans. Wireless Commun.*, vol. 8, no. 5, pp. 2352–2361, May 2009.
- [18] T.-W. Yune, D. Kim, and G.-H. Im, "Iterative detection for spectral efficient user cooperative transmissions over multipath fading channels," *IEEE Trans. Commun.*, vol. 58, no. 4, pp. 1121–1128, Apr. 2010.
- [19] S. Katti et al., "XORs in the air: Practical wireless network coding," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 497–510, Jun. 2008.
- [20] A. Argyriou, "Wireless network coding with improved opportunistic listening," *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 2014–2023, Apr. 2009.
- [21] J. Zhang, Y. P. Chen, and I. Marsic, "MAC-layer proactive mixing for network coding in multi-hop wireless networks," *Comput. Netw.*, vol. 54, no. 2, pp. 196–207, Feb. 2010.

- [22] A. Antonopoulos, C. Verikoukis, C. Skianis, and O. B. Akan, "Energy efficient network coding-based MAC for cooperative ARQ wireless networks," *Ad Hoc Netw.*, vol. 11, no. 1, pp. 190–200, Jan. 2013.
- [23] X. Wang, J. Li, and F. Tang, "Network coding aware cooperative MAC protocol for wireless ad hoc networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 167–179, Jan. 2014.
- [24] M. H. Firooz, Z. Chen, S. Roy, and H. Liu, "Wireless network coding via modified 802.11 MAC/PHY: Design and implementation on SDR," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 8, pp. 1618–1628, Aug. 2013.
- [25] S. Wang, Q. Song, X. Wang, and A. Jamalipour, "Distributed MAC protocol supporting physical-layer network coding," *IEEE Trans. Mobile Comput.*, vol. 12, no. 5, pp. 1023–1036, May 2013.
- [26] J. Li, J. Yuan, R. Malaney, M. Azmi, and M. Xiao, "Network coding based LDPC code design for a multi-source relaying system," *IEEE Trans. Wireless Commun.*, vol. 10, no. 5, pp. 1538–1551, May 2011.
- [27] C. Wang, M. Xiao, and M. Skoglund, "Diversity-multiplexing tradeoff analysis of coded multi-user relay networks," *IEEE Trans. Commun.*, vol. 59, no. 7, pp. 1995–2005, Jul. 2011.
- [28] M. Xiao and T. Aulin, "On the bit error probability of noisy channel networks with intermediate node encoding," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 5188–5198, Nov. 2008.
- [29] M. Xiao and T. Aulin, "Optimal decoding and performance analysis of a noisy channel network with network coding," *IEEE Trans. Commun.*, vol. 57, no. 5, pp. 1402–1412, May 2009.
- [30] H.-T. Lin, Y.-Y. Lin, and H.-J. Kang, "Adaptive network coding for broadband wireless access networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 1, pp. 4–18, 2013.
- [31] M. D. Renzo, M. Iezzi, and F. Graziosi, "Error performance and diversity analysis of multi-source multi-relay wireless networks with binary network coding and cooperative MRC," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2883–2903, 2013.
- [32] M. D. Renzo, M. Iezzi, and F. Graziosi, "On diversity order and coding gain of multisource multirelay cooperative wireless networks with binary network coding," *IEEE Trans. Veh. Technol.*, vol. 62, no. 3, pp. 1138–1157, Mar. 2013.
- [33] J. Li, J. Yuan, R. Malaney, M. Xiao, and W. Chen, "Full-diversity binary frame-wise network coding for multiple-source multiple-relay networks over slow-fading channels," *IEEE Trans. Veh. Technol.*, vol. 61, no. 3, pp. 1346–1360, Mar. 2012.
- [34] B. Nazer and M. Gastp, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6463–6486, Oct. 2011.
- [35] S. Borade, L. Zheng, and R. Gallager, "Amplify-and-forward in wireless relay networks: Rate, diversity, and network size," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3302–3318, Oct. 2007.
- [36] G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *IEEE Trans. Inf. Theory*, vol. 51, no. 9, pp. 3037–3063, Sep. 2005.
- [37] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.
- [38] J. N. Laneman and G. W. Wornell, "Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2415–2425, Oct. 2003.
- [39] T. Ho *et al.*, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.
- [40] F. J. Macwilliams and N. J. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North Holland, 1977.
- [41] SAGE, Open Source Mathematics Software. [Online]. Available: <http://www.sagemath.org>
- [42] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. Philadelphia, PA, USA: SIAM, 2000.
- [43] J.-T. Seong and H.-N. Lee, "Necessary and sufficient conditions for recovery of sparse signals over finite fields," *IEEE Commun. Lett.*, vol. 17, no. 10, pp. 1976–1979, Oct. 2013.



Jin-Taek Seong received the B.S. degree from the University of Seoul, Seoul, Korea, in 2006 and the M.S. degree from Gwangju Institute of Science and Technology (GIST), Gwangju, Korea, in 2008. He is currently working toward the Ph.D. degree at the School of Information and Communications, GIST. He worked for LG Electronics as a Junior Engineer from 2008 until he started his Ph.D. program in 2010. His research interests include cooperative communications, compressed sensing, and network coding.



Heung-No Lee (SM'13) received the B.S., M.S., and Ph.D. degrees from the University of California, Los Angeles, CA, USA, in 1993, 1994, and 1999, respectively, all in electrical engineering. He then moved to HRL Laboratories, LLC, Malibu, CA, USA, and worked there as a Research Staff Member from 1999 to 2002. In 2002, he was appointed an Assistant Professor with the University of Pittsburgh, Pittsburgh, PA, USA, where he stayed until 2008. In 2009, he then moved to the School of Information and Communications, Gwangju Institute of Science and Technology, Gwangju, Korea, where he is currently affiliated. His general areas of research include information theory, signal processing theory, communications/networking theory, and their application to wireless communications and networking, compressive sensing, future internet, and brain-computer interface. He has served as a member of technical program committees for several IEEE conferences, including the IEEE International Conference on Communications and the IEEE Global Communications Conference (Globecom). He served as the Lead Guest Editor for the European Association for Signal Processing *Journal on Wireless Communications and Networking* in 2010–2011. He has served as an Area Editor for the AEU *International Journal of Electronics and Communications* since January 2013. His research efforts have been recognized with prestigious national awards, including the Top 100 National Research and Development Award from the Korean Ministry of Science, ICT, and Future Planning in 2012, the Top 50 Achievements of Fundamental Researches Award from the National Research Foundation of Korea in 2013, and the Science/Engineer of the Month (January 2014) from the National Research Foundation of Korea.

Computation of an Equilibrium in Spectrum Markets for Cognitive Radio Networks

Sang-Seon Byun, *Member, IEEE*, Ilanko Balasingham, *Senior Member, IEEE*, Athanasios V. Vasilakos, *Senior Member, IEEE*, and Heung-No Lee, *Member, IEEE*

Abstract—In this paper, we investigate a market equilibrium in multichannel sharing cognitive radio networks (CRNs): it is assumed that every subchannel is orthogonally licensed to a single primary user (PU), and can be shared with multiple secondary users (SUs). We model this sharing as a spectrum market where PUs offer SUs their subchannels with limiting the interference from SUs; the SUs purchase the right to transmit over the subchannels while observing the interference limits set by the PUs and their budget constraints. Moreover, we consider each SU limits the total interference that can be invoked from all other SUs, and assume that every transmitting SU marks the interference charges to other transmitting SUs. The utility function of SU is defined as least achievable transmission rate, and that of PU is given by the net profit. We define a market equilibrium in the context of extended Fisher model, and show that the equilibrium is yielded by solving an optimization problem, Eisenberg-Gale convex program. To make the solutions of the convex program meet the market equilibrium, we apply monotone-transformation to the utility function of each SU. Furthermore, we develop a distributed algorithm that yields the stationary solutions asymptotically equivalent to the solutions given by the convex program.

Index Terms—Cognitive radio networks, Fisher model, market equilibrium, Eisenberg-Gale convex program, distributed algorithm

1 INTRODUCTION

IN general markets, when the demand and supply depend upon current price only, there exists an equilibrium under certain conditions, called market equilibrium, where 1) all the traders (i.e., suppliers and consumers) can achieve maximum utilities at least in the Pareto sense; 2) the total demand for each commodity is equal to the total supply of that commodity; and 3) all the budgets possessed by the consumers are spent completely [1].

Recently, market-based approaches have started being deployed to various cognitive radio network (CRN) scenarios since the behaviors of the wireless users in CRN can be cast easily into those of the traders in the general market. Furthermore, the market equilibria comply with the key requirement of the CRN, i.e., spectral efficiency; therefore, lately the US Federal Communications Commission (FCC) has employed policies and procedures to bring spectrum trading into CRN, and analogous regulatory efforts are commenced by EU [2], [3], [4].

In the market-based approaches for CRNs, spectra and interference are regarded as marketable products. In words,

primary users (PUs) offer the interference on their licensed spectra to transmitting secondary users (SUs)¹ with collecting certain monetary rewards from SUs; SUs purchase the offered interference on each spectrum by adjusting their transmission powers. Furthermore, every PU has a limitation in her production (i.e., spectra and interference); meanwhile, every SU has limited budget.

In this work, we consider a multichannel sharing CRN where the frequency range is divided into multiple subchannels, and each subchannel is orthogonally licensed to a single PU. Each PU offers the interference on her subchannels to SUs with limiting the interference from the SUs; the SUs purchase the offered interference observing their budget constraints and the interference limits set by the PUs. Moreover, we reflect the interference among SUs to the spectrum market: every SU limits the total interference from other SUs, and every transmitting SU is required to pay charges for interfering with other SUs. The utility function of SU is defined as the least achievable transmission rate, and the utility of PU is given by the net profit the PU makes.

During the history of market theory, there have been lots of market models developed in accordance with various market scenarios: for instance, the *Cournot model* for oligopoly, the *Stackelberg model* considering leadership in the production, the *Bertrand model* for describing the fierce competition among the sellers, and the *Edgeworth model* considering a capacity-limited market.² In our work, we apply the market model developed in [6] where the authors have extended *Fisher model* for a market consisting of multisellers as well as multibuyers, and divisible goods. Considering concave utility functions for buyers and linear utility functions for sellers, the authors have proved that we can achieve an equilibrium that clears the market by solving

• S.-S. Byun is with Daegu-Gyeongbuk Medical Innovation Foundation, 701-310, Dong-Nae Ro 88, Dong-Gu, Daegu, Korea. E-mail: ssbyun@dgmiif.re.kr.

• I. Balasingham is with the Intervention Center, Rikshospitalet, Oslo University Hospital, Oslo N0027, Norway. E-mail: ilangkob@medisin.uio.no.

• A.V. Vasilakos is with the Department of Computer and Telecommunications Engineering, University of Western Macedonia, Parko Agiou Dimitriou, Kozani GR50100, Greece. E-mail: vasilako@ath.forthnet.gr.

• H.-N. Lee is with the School of Information and Communications, Gwangju Institute of Science and Technology, 500-712, Oh-Ryong-Dong, Buk-Gu, Gwangju, Korea. E-mail: heungno@gist.ac.kr.

Manuscript received 14 Mar. 2012; revised 9 Aug. 2012; accepted 12 Aug. 2012; published online 28 Aug. 2012.

Recommended for acceptance by V. Eramo.

For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-2012-03-0193.

Digital Object Identifier no. 10.1109/TC.2012.211.

1. Hereafter, SU implies transmitting SU.

2. Refer to [5] for the details.

a convex optimization problem called *Eisenberg-Gale convex program*. However, in our spectrum market, the purchasing amount is restricted by other buyers as well as sellers unlike the model in [6]: the transmission rate of SU is restricted by the interference limits defined by other SUs as well as PUs. To derive an equilibrium with this restriction, the buyers (i.e., SUs) should pay additional charges to other buyers.

Including the interference limit among SUs into the market model in [6], we define the market equilibrium such that the following conditions hold:

1. The transmission power vector yielded by the market equilibrium maximizes the joint utility of SUs in the Pareto sense satisfying the constraints of budget, interference to PUs, and interference to other SUs.
2. The transmission power vector yielded by the market equilibrium maximizes the utility of every PU, and completely consumes the interference exactly up to the limits set by the PUs.
3. With the prices and charges yielded by the market equilibrium, the sum of the initial budget possessed by all SUs equals the sum of the profits made by all PUs plus the sum of the interference charges gathered by all the SUs.

We model our spectrum market using the Eisenberg-Gale convex program whose objective is to maximize the joint utility function of all SUs, which is given by log-sum-utility, over a convex region defined via a set of linear constraints, and prove that the convex program yields the equilibrium holding the above-mentioned conditions. Meanwhile, this convex program becomes *Nash bargaining problem* whose solution satisfies the weak Pareto optimality, and its Lagrangian dual variables turn out to be the prices and charges given by the equilibrium. Furthermore, we show that the equilibrium satisfies the *core stability* of PUs. However, the utility functions in the convex program should be concave and homogeneous of degree one. Since we formulate the joint utility function of all SUs as a concave function, we apply a monotone-transformation that transforms a concave function into an equivalent function that is homogeneous of degree one with maintaining the concavity.

To find the Lagrangian dual variables, we should solve the system of the linear equations that consist of the Karush-Kuhn-Tucker (KKT) optimality conditions of the convex program. However, the system is generally inconsistent, and which implies that it is impossible to find the exact Lagrangian variables. For this reason, we solve the system with a certain small precision bound. In numerical experiments, we show that approximate Lagrangian variables are found with quite small precision error, and yield an approximate equilibrium quite close to the exact one.

We also consider a distributed implementation for solving the convex program, which enables PUs and SUs to make their decisions autonomously as follows: given the price of each channel, 1) each SU computes her optimal transmission power vector using *best response dynamics*; 2) using linear dynamics, each PU updates the price of her each subchannel in proportion to the interference invoked from the SUs, and, meanwhile, each SU updates the interference charge on each subchannel in proportion to the interference invoked from all the other SUs; and

3) repeat “1” and “2” until the linear dynamics become stationary. We show that the linear dynamics are *asymptotically stable* with any initial points, and the solutions yielded at the stable state is equivalent to the solutions given by the convex program (i.e., *market equilibrium*). By numerical evaluations, we illustrate the market equilibrium given by the distributed algorithm is quite close to the equilibrium yielded by the convex program.

Recently, a distributed algorithm for finding the market equilibrium in the Fisher model has been studied in [7]; the authors have applied proportional response dynamics and proved its convergence. To our best knowledge, there is no previous work that has developed a distributed implementation for finding the equilibrium in the extended model given in [6]. Furthermore, in this paper, we consider the situation where the purchasing amount is restricted by buyers (SUs) as well sellers (PUs).

The remainder of this paper is organized as follows: In Section 2, we summarize some important characteristics of the market model studied in [6], and survey some recent market-based approaches for CRNs. In Section 3, we give our spectrum market model. In Section 4, we define the market equilibrium in our spectrum market. In Section 5, we give the Eisenberg-Gale convex program for our spectrum market, and investigate its properties. In Section 6, we present the distributed algorithm for solving the convex program, and analyze its stability and asymptotic optimality. In Section 7, we give several results of the numerical evaluations. Finally, we conclude this paper in Section 8.

2 RELATED WORK

2.1 Market-Based Approaches for CRNs

Recently, the market-based approach has been deployed to CRNs by several research efforts.

Xing et al. [8] have considered a spectrum market where different consumers evaluate the same supplier differently according to their applications and locations. Considering limited information, they have developed price dynamics with a stochastic learning algorithm to find the optimal price yielding maximum benefit of the suppliers. However, they have not addressed the utility of the consumers.

Hong and Garcia [9] have proposed a fully distributed algorithm—no collaboration among SUs and PUs—that achieves a market equilibrium in multichannel sharing CRN such that the supply of the spectrum equals to its demand, and the network of SU is stable. They have investigated and presented the convergence condition of the algorithm in terms of the channel gain. Unlike other approaches, they have considered the utilities of PUs as well as SUs. However, they have not investigated the social optimality of the market equilibrium.

Niyato and Hossain [10], [11] have modeled a multilevel bandwidth sharing in CRN into an interrelated market. They have proposed a price-demand decision algorithm that guarantees the convergence to a market equilibrium with which all the primary and secondary services are satisfied. However, they have assumed that the price-demand decision algorithm is directed by a central authority in each level, and have not considered the way of allocating the spectra to users in each level.

Xu et al. [12] have proposed a secondary network where SUs trade among themselves their channels purchased from PUs in the direction of asymptotic optimal spectrum utilization. To this end, they have devised dynamic double auction mechanism that is conducted by a centralized spectrum broker, and proved the truthfulness and asymptotic efficiency in the total social welfare.

Li et al. [13] have addressed a spectrum auction mechanism between SUs and spectrum owners without any central auctioneer. They have deployed an iterative matching algorithm that achieves the price set in core where no SUs and spectrum owners can negotiate to do better for both. However, they have assumed fixed transmission power, and there is no consideration of the market clearance.

In [14], Xu et al. have handled a two-tier market: spectrum contracts from a PU to SUs in Tier-1, and spectrum redistribution among SUs to satisfy SUs dynamic traffic demands in Tier-2. They have applied Nash bargaining solution in Tier-1 market to achieve the fairness between the utility of the PU and the aggregate utility of all the SUs. For Tier-2 market, they have deployed random matching and bilateral bargaining. However, they have considered a single PU, and have not addressed the market clearance.

Xie et al. [15] have addressed the spectrum trading between wireless users—that can be regarded as SUs in a CRN—and a single price manager—that can be regarded as a PU or spectrum broker in a CRN, and investigated a market equilibrium where the market clears, and the budgets of the wireless users are completely consumed. Unlike our work, they have addressed the actual interference among the wireless users in the utility function. Then they have shown that the market equilibrium is given by the solution of a linear complementarity problem, and under the symmetric channel gain and low-rank conditions, they have proved that this problem becomes equivalent to the problem of finding KKT points of a quadratic program. Furthermore, they have developed a decentralized tatonnement process that converges to the equilibrium. However, they have not included the manager's utility in the market equilibrium. Moreover, the KKT points of the quadratic program do not guarantee the optimality, and due to this reason, it is not verified whether the distributed algorithm (tatonnement process) converges to the optimal solution even asymptotically.

Koutsopoulos and Iosifidis [16] have surveyed various auction mechanisms for spectrum allocation in CRN. They have indicated that auction mechanisms (including double auction mechanisms) involve a single seller and multiple buyers, and have no interaction among buyers.

2.2 Brief of the Extended Market Model

The Fisher model considers a market consisting of multiple buyers and divisible goods. In the Fisher model, the budget for each buyer and the amount of each commodity need to be specified, and the utility functions of buyers are assumed to be concave. Then the market equilibrium in the Fisher model is given by the price of each commodity that yields optimal utilities of buyers at least in Pareto sense and clears the market: there should be neither surplus nor deficiency in any of the commodities and the budgets [17].

In 1959, Eisenberg and Gale gave a convex program for

computing market equilibrium for the Fisher model of linear utility functions [18], and in 1961, Eisenberg generalized this to concave homogeneous functions of degree one [19]. In 1954, Nobel laureates Arrow and Debrue generalized the Fisher model considering agents who come to the market with initial endowments of goods, and at any set of prices, want to sell all their goods and buy optimal bundles at these prices. The problem again is to find market clearing prices [20].

Jain et al. [6] have extended the Fisher model considering the utilities that are homothetic, quasiconcave, and homogeneous functions of arbitrary degree; they also have included sellers' utilities into their model. They have applied the Eisenberg-Gale convex program with the buyers' utilities monotone-transformed. They show that the equilibrium price, given by the Lagrangian dual variables of the convex program, maximizes sellers' utilities as well as buyers' utilities, and clears the market.

In our work, we employ this extended model to our spectrum market since it deals with the market where the buyers are clearly distinguished from sellers, and considers the utilities of sellers as well as those of buyers. However, in our spectrum market, the purchasing amount is restricted by other buyers as well as sellers: that is, the amount of interference that can be purchased by each SU is restricted by the interference limits imposed by other SUs as well as PUs. To derive an equilibrium with this restriction, we consider that each buyer (i.e., SU) pays additional charges to all the other buyers. Furthermore, we envisage a distributed implementation of our market model.

We emphasize that the extended Fisher model is a unique market model that enables the following considerations altogether:

1. strictly distinguished multiple sellers (PUs) from multiple buyers (SUs);
2. guarantee of Pareto optimality for SUs' utility and the core-existence for PUs' utilities;
3. guarantee of the market clearance;
4. distributed implementation with no central authority; and
5. providing the way of allocating the spectra to SUs.

3 SPECTRUM MARKET MODEL

We consider a CRN where all the PUs and SUs are located within a limited geographical region. Then we address the spectral resource on the frequency domain taking the multichannel sharing into account—that is, the whole frequency range is split into multiple subchannels, and each subchannel can be shared by multiple users. In this work, we premise that each subchannel is exclusively licensed to a single PU; it, however, can be shared with multiple SUs concurrently unless the SUs invoke interference larger than certain limits (see Fig. 1).

Applying a market concept to our CRN scenario, the subchannels and interference are interpreted into the types of commodity and the quantity of each commodity, respectively. Upon the current prices, each SU decides subchannels and the amount of interference she would like to purchase; each PU updates the price on every her subchannel according to the interference invoked from the SUs.

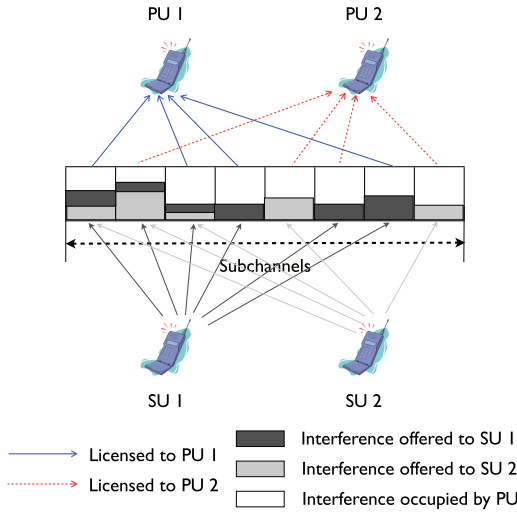


Fig. 1. Multichannel sharing model with two PUs, two SUs, and eight subchannels. Each subchannel is orthogonally allocated to a single PU, and can be shared with all the SUs unless the SUs invoke less interference than the PUs can tolerate.

Ideally, SUs' utility function should reflect actual interference from all the other SUs as well as interference from PUs, which, however, makes the problem nonconvex if the utility function is given as Shannon capacity [21]. Therefore, we consider the maximum allowable interference and reflect it to the utility function; the utility function is given by least achievable transmission rate. Then we let SUs make and charge for every subchannel—in proportion to the amount of interference from all the other SUs. Moreover, we assume that the maximum allowable interference is given by a central authority to let SUs share the interference fairly.

The price as well as the interference charge are marked on every subchannel, and given as a *price per unit interference*.

As a supplier in general market has a limitation on the net supply of its commodity, we envisage that every PU has a limitation on the interference she can offer to SUs over every her subchannel. In addition, like consumers in general markets, every SU cannot spend more budget than she possesses on purchasing the interference from the PUs and paying the interference charges to all the other SUs.

Prior to giving the formal definition of the market equilibrium, we define the following denotations:

- \mathcal{I} : Set of transmitting SUs.
- \mathcal{L} : Set of PUs, and we let $m := |\mathcal{L}|$.
- \mathcal{J} : Set of subchannels, and we let $n := |\mathcal{J}|$
- $u_i : \mathbf{R}_+^n \rightarrow \mathbf{R}_+$: Utility function of SU $i \in \mathcal{I}$.
- $\mathbf{p}_i = [p_{i1}, \dots, p_{in}]^T$: Transmission power vector of SU i .
- p_{ij} : SU i 's transmission power on subchannel $j \in \mathcal{J}$.
- \mathcal{J}_l : Set of subchannels licensed to PU $l \in \mathcal{L}$.
- $v_l : \mathbf{R}^n \rightarrow \mathbf{R}$: Utility function of PU l .
- π_{lj} : Price marked by PU l on subchannel j .
- $\boldsymbol{\pi}_l = [\pi_{l1}, \dots, \pi_{ln}]^T$: Price vector of PU l .
- η_{ij} : Interference charge decided by SU i on subchannel j .
- $\boldsymbol{\eta}_i = [\eta_{i1}, \dots, \eta_{in}]^T$: Vector of interference charges decided by SU i .
- y_{lj} : Limit of allowable interference from all the SUs to PU l on subchannel j .

- γ_{ij} : Maximum allowable interference from all the other SUs to SU i on subchannel j .

Now, we develop the spectrum market model with the following key considerations:

1. Each SU $i \in \mathcal{I}$ has a concave, scalable utility function u_i with respect to its transmission power vector \mathbf{p}_i . In this paper, the utility of SU is defined as the summation of least achievable rate on every subchannel, that is,

$$u_i(\mathbf{p}_i) = \sum_{j \in \mathcal{J}} B_j \log_2 \left(1 + \frac{p_{ij} G_{ij}}{N_0 + \gamma_{ij} + \Gamma_{ij}} \right), \quad (1)$$

where B_j is the bandwidth of subchannel $j \in \mathcal{J}$, G_{ij} is the channel gain on subchannel j between SU i and its target, Γ_{ij} is the interference invoked from the PU who owns subchannel j to SU i , and N_0 is the thermal noise.

2. SUs cannot purchase the interference larger than the limits set by PUs. That is, $\forall l \in \mathcal{L}$ and $\forall j \in \mathcal{J}_l$,

$$\sum_{i \in \mathcal{I}} p_{ij} G_{ij}^l \leq y_{lj}, \quad (2)$$

where G_{ij}^l is the channel gain on subchannel j between SU i and PU l . We assume that each PU has a minimum QoS requirement and sets the interference limit (i.e., y_{lj}) in order that the QoS may be guaranteed.

3. Each SU i has an initial endowment of budget $e_i > 0$, and the total budget spent on the purchase of subchannels and the interference charges cannot exceed e_i .
4. Each PU l offers her subchannels, i.e., subchannels in \mathcal{J}_l , to SUs with price π_{lj} , and its utility function v_l is defined as the net profit it makes, i.e.,

$$v_l = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}_l} \pi_{lj} p_{ij} G_{ij}. \quad (3)$$

5. In every SU, the total interference on subchannel j from all the other SUs cannot exceed the maximum allowable interference: $\forall i \in \mathcal{I}$ and $\forall j \in \mathcal{J}$

$$\sum_{k \in \mathcal{I}, k \neq i} p_{kj} G_{kj}^i \leq \gamma_{ij}, \quad (4)$$

where G_{kj}^i is the channel gain on subchannel j between SU k and the target of SU i . In this paper, we assume a symmetric channel gain among SUs: that is, $G_{kj}^i = G_{ij}^k$ where $i \neq k$.

The channel gain reflects the free space path loss defined by Friis transmission equation [22]:

$$G_{ij} = \frac{g_i g_k \lambda_j^2}{(4\pi)^2 d^2 L}, \quad (5)$$

where g_i is the transmitter antenna gain, g_k is the receiver antenna gain, d is the transmitter-receiver separation distance in meters, L is the system loss factor not related to propagation, and λ_j is the wavelength in meters on subchannel j . Since the channel gain reflects the wave length of each subchannel, the higher frequency a subchannel has,

the less amount of information a user can transmit on it given a fixed transmission power and subchannels with equal bandwidth [8]. To let every subchannel have equal opportunity of being purchased, we assume that the frequency range is divided into subchannels in the way that every subchannel yields similar transmission rate given a constant transmission power and separate distance.

Additional assumptions are the following:

1. All PUs and SUs can access noncontiguous subchannels in parallel.
2. All SUs have the transmission power enough to fully utilize the interference offered by PUs.
3. The spectrum trade occurs on every predefined epoch, and all SUs perform their transmissions ceaselessly during the epoch.
4. All PUs perform their transmission ceaselessly without any changes in their transmission powers during the epoch.

4 MARKET EQUILIBRIUM

Henceforth, we let y_{lj} have a very large number for all $l \in \mathcal{L}$ and $j \in \mathcal{J}$ but $j \notin \mathcal{J}_l$.³ Based on the key considerations mentioned in Section 3, we define the equilibrium in the spectrum market following [6]: the equilibrium is defined as a pair of a nonnegative price vector $\pi = [\pi_1, \dots, \pi_m]^T$ and a vector of interference charges $\eta = [\eta_1, \dots, \eta_{|\mathcal{I}|}]^T$ at which there exists a transmission power vector p_i for each SU i such that the following conditions hold:

1. The vector \mathbf{p}_i maximizes the utility of SU i given her initial endowment of budget e_i and the equilibrium π and η , that is, \mathbf{p}_i maximizes u_i over all $\mathbf{p}_i \in \mathbf{R}_+^n$ subject to

$$\sum_{l \in \mathcal{L}} \sum_{j \in \mathcal{J}} \pi_{lj} p_{ij} G_{ij}^l + \sum_{j \in \mathcal{J}} p_{ij} \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{ij}^k \leq e_i, \quad (6)$$

and constraint (4).

2. For each PU l , the vector \mathbf{p}_i maximizes the profit v_l subject to constraint (2).
3. The total interference offered by all PUs equals the total interference consumed by all SUs, that is, $\sum_{l \in \mathcal{L}} \sum_{j \in \mathcal{J}} y_{lj} = \sum_{i \in \mathcal{I}} \sum_{l \in \mathcal{L}} \sum_{j \in \mathcal{J}} p_{ij} G_{ij}^l$.
4. The sum of the initial budget possessed by SU i equals the sum of the prices paid to all PUs by SU i plus the sum of the interference charges paid to all the other SUs by SU i , that is,

$$e_i = \sum_{l \in \mathcal{L}} \sum_{j \in \mathcal{J}} \pi_{lj} p_{ij} G_{ij}^l + \sum_{j \in \mathcal{J}} p_{ij} \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{ij}^k.$$

Therefore, the market equilibrium in this model is also known as market clearing equilibrium since it clears not only all the commodities offered by suppliers but also all the initial budget possessed by consumers; that is, it clears all the budget possessed by SUs as well as all the interference offered by PUs.

3. Over subchannels that a PU does not own, we set the interference limit very large numbers. Then we can drop the subscript l from \mathcal{J}_l . Surely, $\forall j \notin \mathcal{J}_l$, y_{lj} does not belong to the commodities of PU l .

However, if SU's maximum allowable interference, i.e., γ_{ij} for any i and j , is set significantly small or null in the worst case, the interference offered by PUs may not be consumed entirely. Then the market clearance cannot be guaranteed. We present this problem in Section 5 in detail.

5 EISENBERG-GALE CONVEX PROGRAM

In this section, we develop a convex program, called Eisenberg-Gale convex program [6], [23], that yields the market equilibrium defined in the previous section. The convex program is to maximize the joint utility function of all SUs given by log-sum-utility over a convex region defined via a set of linear constraints.

The Eisenberg-Gale convex program can compute the market equilibrium only when the utility functions are homogeneous of degree one as well as concave. As given in (1), the utility function is concave but not homogeneous of degree one. Therefore, prior to presenting the convex program, we give a formal method of transforming a nonhomogeneous function into an equivalent function homogeneous of degree one.

5.1 Monotone-Transformation of the Utility Function

Most of all, we need the following definition [24]:

Definition 1. Given a function $u : \mathbf{R}_+^n \rightarrow \mathbf{R}_+$:

1. u is strictly monotonic if for any $\mathbf{p}, \bar{\mathbf{p}} \in \mathbf{R}_+^n$, $\mathbf{p} > \bar{\mathbf{p}}$ implies that $u(\mathbf{p}) > u(\bar{\mathbf{p}})$;
2. let u be a strictly monotonic function. Then u is homothetic if for any $\mathbf{p}, \bar{\mathbf{p}} \in \mathbf{R}_+^n$ and any $\alpha > 0$, $u(\mathbf{p}) \geq u(\bar{\mathbf{p}})$ iff $u(\alpha\mathbf{p}) \geq u(\alpha\bar{\mathbf{p}})$; and
3. u is homogeneous of degree one if for any $\mathbf{p} \in \mathbf{R}_+^n$ and any $\alpha > 0$, $u(\alpha\mathbf{p}) = \alpha u(\mathbf{p})$.

It is not difficult to check whether u_i given by (1) is continuous, strictly monotonic and homothetic, but not homogeneous of degree one. Therefore, we apply a *monotone-transformation* [6] that preserves strict monotonicity, concavity, and homotheticity. The monotone-transformation is given by the following theorem:

Theorem 1. Let $u : \mathbf{R}_+^n \rightarrow \mathbf{R}_+$ be a continuous, strict monotonic, concave, homothetic function. Then there is a monotone-transformation yielding a function $f : \mathbf{R}_+^n \rightarrow \mathbf{R}_+$ that is homogeneous of degree one, and preserves continuity, strict monotonicity, concavity, and homotheticity, and satisfies:

1. If $u(\mathbf{p}) = 0$, then $f(\mathbf{p}) = 0$.
2. If $u(\mathbf{p}) \neq 0$, then there exists a unique $\alpha \in \mathbf{R}_+$ such that $u(\mathbf{p}/\alpha) = 1$, and $f(\mathbf{p}) = \alpha$.

Consequently, the monotone-transformation can be done by finding α that satisfies the following nonlinear equation:

$$u(\mathbf{p}/\alpha) = 1. \quad (7)$$

5.2 Convex Program Yielding Market Equilibrium

By the monotone-transformation, we can transform u_i to f_i that is homogeneous of degree one. Then we develop the Eisenberg-Gale convex program that yields the market equilibrium as follows:

$$\text{maximize } \sum_{i \in \mathcal{I}} e_i \ln(f_i) \quad (8)$$

subject to

$$\sum_{k \in \mathcal{I}, k \neq i} p_{kj} G_{kj}^i \leq \gamma_{ij}, \quad \forall i \in \mathcal{I} \text{ and } \forall j \in \mathcal{J}, \quad (9)$$

$$\sum_{i \in \mathcal{I}} p_{ij} G_{ij}^l \leq y_{lj}, \quad \forall l \in \mathcal{L} \text{ and } \forall j \in \mathcal{J}. \quad (10)$$

As we have addressed in Section 4, if γ_{ij} is set too small, the market clearance cannot be guaranteed. Considering nonzero channel gains, we present the problem in detail:

- In case $\gamma_{ij} = 0$ for all i and j , all p_{ij} should be null. Then, the conditions for the market equilibrium (i.e., conditions 3 and 4) can never be met.
- In case γ_{ij} for all i and j is nonzero, but it has significantly small value, p_{ij} should have also small value to meet the constraint given in (9). Then the constraint (10) may not be tight always. The tightness of the constraint (10) is one of the condition for the market equilibrium, i.e., 3). If the tightness of the constraint (10) is not met, the price of the subchannels that are not entirely purchased by SUs is given as zero according to the KKT condition given in (15), and in turn the condition 4) may not hold either.

To prevent the above problem, we need to add the following necessity condition for the existence of the market equilibrium: every $\gamma_{ij} \forall i \in \mathcal{I}$ and $\forall j \in \mathcal{J}$ is nonzero and has a value that makes a solution vector \mathbf{p} satisfy $\sum_{i \in \mathcal{I}} p_{ij} G_{ij}^l = y_{lj} \forall l \in \mathcal{L}$ and $\forall j \in \mathcal{J}$, and $\sum_{k \in \mathcal{I}, k \neq i} p_{kj} G_{kj}^i \leq \gamma_{ij} \forall i \in \mathcal{I}$ and $\forall j \in \mathcal{J}$ always. Intuitively, SUs will always utilize the available interference entirely bounded by constraint (10) to maximize their collaborative utility given in (8) if constraint (10) is set tighter than constraint (9), and therefore the market clearance can be guaranteed. As an instance, we consider a spectrum market where the channel gains over all SUs, PUs, and channels have an identical value, i.e., identical G_{ij}^l and G_{kj}^i for all $i, k \in \mathcal{I}$, $j \in \mathcal{J}$, and $l \in \mathcal{L}$. Additionally, we let $\gamma_{ij} = c$ and $y_{lj} = f$ for all $i \in \mathcal{I}$, $j \in \mathcal{J}$, and $l \in \mathcal{L}$. Then, if $c \geq f$, constraint (10) over all l and j becomes always tight to achieve the maximum utility.

For all l and $j \notin \mathcal{J}_l$, we give a large value to y_{lj} . In this case, π_{lj} should be zero to satisfy the optimality condition given in (15), and this setting is reasonable. Then we draw the following remark from the convex program:

Remark 1. The solution of the above Eisenberg–Gale convex program (henceforth, convex program) is often regarded as the *Nash bargaining solution* [25] with zero disagreement point. Therefore, it has a unique solution vector that satisfies *Pareto optimality* due to the strict concavity of the objective function and linearity of the constraints [26].

Let \tilde{p}_{ij} denote the optimal solutions to the convex program. Notice that $f_i(\tilde{\mathbf{p}}_i) > 0$ for all i . Now, we have the following KKT optimality conditions with the corresponding

Lagrangian multipliers η_{ij} and π_{lj} :

$$\tilde{p}_{ij} \left(\frac{e_i}{f_i(\tilde{\mathbf{p}}_i)} \frac{\partial f_i(\tilde{\mathbf{p}}_i)}{\partial p_{ij}} - \sum_{l \in \mathcal{L}} \pi_{lj} G_{ij}^l - \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{kj}^i \right) = 0, \quad (11)$$

$$\forall i \in \mathcal{I} \text{ and } \forall j \in \mathcal{J},$$

$$\frac{e_i}{f_i(\tilde{\mathbf{p}}_i)} \frac{\partial f_i(\tilde{\mathbf{p}}_i)}{\partial p_{ij}} - \sum_{l \in \mathcal{L}} \pi_{lj} G_{ij}^l - \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{kj}^i \leq 0, \quad (12)$$

$$\forall i \in \mathcal{I} \text{ and } \forall j \in \mathcal{J},$$

$$\sum_{k \in \mathcal{I}, k \neq i} \tilde{p}_{kj} G_{kj}^i \leq \gamma_{ij}, \quad \forall i \in \mathcal{I} \text{ and } \forall j \in \mathcal{J}, \quad (13)$$

$$\sum_{i \in \mathcal{I}} \tilde{p}_{ij} G_{ij}^l \leq y_{lj}, \quad \forall l \in \mathcal{L} \text{ and } \forall j \in \mathcal{J}, \quad (14)$$

$$\pi_{lj} \left(\sum_{i \in \mathcal{I}} \tilde{p}_{ij} G_{ij}^l - y_{lj} \right) = 0, \quad \forall l \in \mathcal{L} \text{ and } \forall j \in \mathcal{J}, \quad (15)$$

$$\eta_{ij} \left(\sum_{k \in \mathcal{I}, k \neq i} \tilde{p}_{kj} G_{kj}^i - \gamma_{ij} \right) = 0, \quad \forall i \in \mathcal{I} \text{ and } \forall j \in \mathcal{J}, \quad (16)$$

and (11) and (12).

Subsequently, we establish the following linear program for each PU l :

LP1:

$$\text{maximize } \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \pi_{lj} p_{ij} G_{ij}^l \quad (17)$$

subject to

$$\sum_{i \in \mathcal{I}} p_{ij} G_{ij}^l \leq y_{lj}, \quad \forall j \in \mathcal{J}. \quad (18)$$

We prove that π_l and η_i is the equilibrium price, and $\tilde{\mathbf{p}}$ is the optimal solution of LP1 as well. This proof begins with Euler's theorem [24]:

Theorem 2 (Euler's Theorem). *Let $f(\mathbf{p})$ be a homogeneous function of degree 1 on \mathbf{R}_+^n . Then, for all \mathbf{p} ,*

$$p_1 \frac{\partial f(\mathbf{p})}{\partial p_1} + p_2 \frac{\partial f(\mathbf{p})}{\partial p_2} + \cdots + p_n \frac{\partial f(\mathbf{p})}{\partial p_n} = f(\mathbf{p}). \quad (19)$$

Then the following core theorem holds:

Theorem 3. *The optimal solution to the convex program optimizes the utility of each SU i and the utility of each PU l , and the Lagrangian multipliers, π and η , are the equilibrium. In addition, the interference offered by all PUs is entirely exhausted by SUs, and the initial budget possessed by all SUs is fully spent and precisely equals to the total profit earned by the PUs plus the total interference charges. Namely, the market clears.*

Proof. Summing (11) over $j \in \mathcal{J}$, we get

$$\sum_{j \in \mathcal{J}} \frac{e_i}{f_i(\tilde{\mathbf{p}}_i)} \frac{\partial f_i(\tilde{\mathbf{p}}_i)}{\partial p_{ij}} \tilde{p}_{ij} - \sum_{j \in \mathcal{J}} \tilde{p}_{ij} \sum_{l \in \mathcal{L}} \pi_{lj} G_{ij}^l - \sum_{j \in \mathcal{J}} \tilde{p}_{ij} \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{kj}^i = 0, \quad \forall i \in \mathcal{I}. \quad (20)$$

By Euler's theorem, the first term in the left side of (20) is reduced to e_i simply. Then, since we assume $G_{kj}^i = G_{ij}^k$, $\forall i, k \in \mathcal{I}$ and $\forall j \in \mathcal{J}$ where $i \neq k$,

$$e_i = \sum_{j \in \mathcal{J}} \tilde{p}_{ij} \sum_{l \in \mathcal{L}} \pi_{lj} G_{ij}^l + \sum_{j \in \mathcal{J}} \tilde{p}_{ij} \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{ij}^k = 0, \quad \forall i \in \mathcal{I}, \quad (21)$$

which implies that each SU spends her initial budget completely under the equilibrium π and η . The first term in the right side of (21) indicates the total price to be paid to all PUs by SU i , and the second term indicates the total interference charge to be paid to all other SUs by SU i .

We next consider the dual program of LP1 for each PU l with dual variables w :

LP2:

$$\text{minimize } \sum_{j \in \mathcal{J}} y_{lj} w_{lj} \quad (22)$$

subject to

$$w_{lj} G_{ij}^l = \pi_{lj} G_{ij}^l, \quad \forall i \in \mathcal{I} \text{ and } \forall j \in \mathcal{J}, \quad (23)$$

$$w_{lj} \geq 0, \quad \forall j \in \mathcal{J}. \quad (24)$$

Let $\bar{\mathbf{p}}$ be the optimal solution of LP2. By the complementary slackness condition [27], the following equation holds as well:

$$w_{lj} \left(\sum_{i \in \mathcal{I}} \bar{p}_{ij} G_{ij}^l - y_{lj} \right) = 0, \quad \forall l \in \mathcal{L} \text{ and } \forall j \in \mathcal{J}. \quad (25)$$

Assuming that $G_{ij}^l \neq 0$ for all i, j , and l , (23) becomes $w_{lj} = \pi_{lj}$. Thus,

$$\pi_{lj} \left(\sum_{i \in \mathcal{I}} \bar{p}_{ij} G_{ij}^l - y_{lj} \right) = 0, \quad \forall l \in \mathcal{L} \text{ and } \forall j \in \mathcal{J}. \quad (26)$$

We see that $\bar{\mathbf{p}}$ also satisfies (26), and thus it can be the optimal solution of the LP2 as well. Hence, together with Remark 1, we draw a conclusion that all the PUs and SUs can achieve maximum utilities with respect to the equilibrium π and η .

Finally, we show that the market clears. Summing (21) over all $i \in \mathcal{I}$, we get

$$\sum_{i \in \mathcal{I}} e_i = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \tilde{p}_{ij} \sum_{l \in \mathcal{L}} \pi_{lj} G_{ij}^l + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \tilde{p}_{ij} \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{ij}^k, \quad (27)$$

and by (15),

$$\sum_{i \in \mathcal{I}} e_i = \sum_{j \in \mathcal{J}} \sum_{l \in \mathcal{L}} \pi_{lj} y_{lj} + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \tilde{p}_{ij} \sum_{k \in \mathcal{I}, k \neq i} \eta_{kj} G_{ij}^k. \quad (28)$$

Eventually, we conclude that the equilibrium given by the Lagrangian multipliers clears the market. \square

The equilibrium can be computed by solving the system of the linear equations (11), (12), (15), and (16) using the optimal solutions of the convex program. It is known that the system of the linear equations has a unique solution if it is consistent [17]. However, the system is normally inconsistent since there are more equations than the unknown

variables. Thus, we need to provide a certain precision bound on each linear equation to achieve an approximate equilibrium at least. The precision bound is dependent on the instance of the problem such as the sizes of PUs, SUs, subchannels, and nonzero solutions. In Section 7, we evaluate numerically the quality of the solution varying the precision bound given a CRN instance.

Besides, to solve the system of the linear equations, we need to compute the partial derivatives of the monotone-transformed function (required in (11) and (12)), and it is given by the following lemma [6]:

Lemma 1. *If we let $f(\mathbf{p}) = \alpha$, then the partial derivatives to p_j of f is given by*

$$\frac{\partial f(\mathbf{p})}{\partial p_j} = \alpha \frac{\partial u(\mathbf{p}/\alpha)/\partial p_j}{\nabla u(\mathbf{p}/\alpha)^T \mathbf{p}}. \quad (29)$$

5.3 Core Stability of the Production Vector

In this section, we show that the solution $\tilde{\mathbf{p}}$ yielded by the convex program is in the *core* of nontransferable utility (NTU) game in the context of cooperative game theory [28].

In a cooperative game, the existence of nonempty core guarantees that no player will break away from the grand coalition (i.e., the cooperation of all the players) since the payoffs achieved by the cooperation within any subcoalition are not larger than the payoff yielded by the cooperation of the grand coalition. Therefore, if the solution $\tilde{\mathbf{p}}$ is in the core of the spectrum market, we guarantee that no PU will leave this market.

Denoting the core of the spectrum market as $C(V)$, the core of an NTU game is defined as:

Definition 2. *For a $S \subseteq \mathcal{L}$, let $V(S) = \{v_l(\mathbf{p}) : l \in S\}$. Then the core of the spectrum market, $C(V)$ is defined as the set of all undominated imputations, i.e., $\tilde{\mathbf{p}} \in C(V)$, if and only if there is neither $S \subset \mathcal{L}$, $S \neq \emptyset$, nor \mathbf{p} such that $v_l(\mathbf{p}) > v_l(\tilde{\mathbf{p}})$ for all $l \in S$.*

Theorem 4. *The solution vector $\tilde{\mathbf{p}}$ of the convex program belongs to $C(V)$.*

Proof. In this proof, we apply the proof by contradiction. Thus, this proof begins with supposing $\tilde{\mathbf{p}} \notin C(V)$. Then there exists $S \subset \mathcal{L}$, $S \neq \emptyset$, and some allocation $\hat{\mathbf{p}}$ such that

$$v_l(\hat{\mathbf{p}}) > v_l(\tilde{\mathbf{p}}), \quad (30)$$

for all $l \in S$. That is,

$$\sum_{j \in \mathcal{J}} \pi_{lj} \sum_{i \in \mathcal{I}} \hat{p}_{ij} G_{ij}^l > \sum_{j \in \mathcal{J}} \pi_{lj} \sum_{i \in \mathcal{I}} \tilde{p}_{ij} G_{ij}^l, \quad \forall l \in S. \quad (31)$$

Then, by (15),

$$\sum_{j \in \mathcal{J}} \pi_{lj} \sum_{i \in \mathcal{I}} \hat{p}_{ij} G_{ij}^l > \sum_{j \in \mathcal{J}} \pi_{lj} y_{lj}, \quad \forall l \in S. \quad (32)$$

To satisfy (32), the following relation should be satisfied in any l and j :

$$\sum_{i \in \mathcal{I}} \hat{p}_{ij} G_{ij}^l > \sum_{j \in \mathcal{J}} y_{lj}, \quad \forall l \in S. \quad (33)$$

However, the inequality (33) violates the constraint of the total interference. This contradiction proves that $\tilde{\mathbf{p}}$ is undominated, and therefore belongs to $C(V)$. \square

Consequently, it is guaranteed that the interference solutions given by the convex program let no PU break away from the spectrum market.

6 DISTRIBUTED ALGORITHM

In this section, we develop a distributed approach whose stationary point is asymptotically equivalent to the optimal solution given by the convex program.

6.1 The Algorithm

A natural class of dynamics in multiplayer noncooperative system is the *best-response dynamics* where each player updates her strategy to maximize her utility given the strategies of other players [29]. In this algorithm, each SU deploys the best-response dynamics to maximize her individual utility given the vectors of price and interference charges. That is, the best response of SU i is given by

$$\beta_i(\pi, \eta) = \arg \max_{\mathbf{p}_i; \text{s.t. (6)}} e_i \ln f_i(\mathbf{p}_i). \quad (34)$$

Accordingly, the algorithm with the best-response dynamics is given as follows:

- 1: Initialize $\pi(0)$ and $\eta(0)$;
- 2: $t \leftarrow 0$;
- 3: **loop**
- 4: **for** each $i \in \mathcal{I}$ **do**
- 5: Find the best response $\beta_i(t)$ given $\pi(t)$ and $\eta(t)$;
- 6: **end for**
- 7: **for** each $l \in \mathcal{L}$ and $j \in \mathcal{J}$ **do**
- 8: **if** $j \notin \mathcal{J}$ **then**
- 9: $\pi_{lj} = 0$;
- 10: **else**
- 11: Update the price such that

$$\dot{\pi}_{lj} = \alpha \left(\sum_{i \in \mathcal{I}} p_{ij}(t) G_{ij}^l - y_{lj} \right)_{\pi_{lj}}^+; \quad (35)$$
- 12: **end if**
- 13: **end for**
- 14: **for** each $i \in \mathcal{I}$ and $j \in \mathcal{J}$ **do**
- 15: **if** $p_{ij}(t) = 0$ **then**
- 16: $\eta_{ij} = 0$;
- 17: **else**
- 18: Adjust the interference charge by

$$\dot{\eta}_{ij} = \alpha \left(\sum_{k \in \mathcal{I}, k \neq i} p_{kj}(t) G_{kj}^l - \gamma_{ij} \right)_{\eta_{ij}}^+ \quad (36)$$
- 19: **end if**
- 20: **end for**
- 21: **if** $\dot{\pi}_{lj} < \epsilon$ and $\dot{\eta}_{ij} < \epsilon$ for all $l \in \mathcal{L}$ and $j \in \mathcal{J}$ **then**
- 22: Terminate the loop;
- 23: **else**
- 24: $t \leftarrow t + 1$;
- 25: Continue the loop;
- 26: **end if**
- 27: **end loop**

In this algorithm, α and ϵ indicate the adjustment speed of the linear dynamics and the termination condition of the algorithm, respectively. If we set the adjustment speed too small, the algorithm requires more iterations until it converges. On the other hand, setting it too large can make the algorithm oscillate. There is no formal way of finding an adequate value for the adjustment speed except trial-and-error. In this paper, an adequate adjustment speed is found by testing the algorithm with various adjustment speed. $(a)_b^+$ implies $\max(a, 0)$ if $b = 0$, and equal to a if $b > 0$. In every time t , by (34), each SU finds the transmission power vector that is her best response to the current price and interference charges. When each SU computes its best response dynamics, she should know the amount of interference from PUs as well as the channel gain to her target. By (35), each PU updates the price on each her subchannel according to the interference from all SUs. By (36), each SU updates her interference charge on each subchannel according to the interference from other SUs. We note that PUs and SUs update their interference charges with only the amount of interference they observe. Then the price updated by each PU (i.e., π) is delivered to all SUs, and the price updated by each SU (i.e., η) is delivered to other SUs. These price lists are the only feedbacks that should be delivered.

6.2 Stability Analysis

In this section, we prove the asymptotic stability of the linear dynamics given by (35) and (36). To this end, we develop the following theorem.

Theorem 5. *The linear dynamics given by (35) and (36) are globally asymptotically stable.*

Proof. See Appendix A, which can be found on the Computer Society Digital Library at <http://doi.ieeecomputersociety.org/10.1109/TC.2012.211>. \square

We see that, as the distributed algorithm proceeds, SUs' utilities approach to those yielded by the convex program.

6.3 Investigation of Asymptotic Equivalence to the Convex Program

We have shown the linear dynamics are asymptotically stable. Then, assuming the system of the linear equations (11), (12), (15), and (16) is consistent, we can prove that the distributed algorithm yields the solutions *asymptotically equivalent* to the solutions of the convex program. That is, the solutions yielded at $t = \infty$ is equivalent to the solutions of the convex program. However, as discussed in Section 5.2, it is not easy to validate the equivalence because of the inconsistency in the system of the linear equations. Nonetheless, by numerical experiments, we illustrate that the utilities of SUs determined by the distributed algorithm are almost identical to those yielded by the convex program in spite of the inconsistency, and which will be explained in Section 7. We also show that the equilibrium yielded by the distributed algorithm also meets the KKT conditions within a small tolerance.

We give the following theorem.

Theorem 6. *Postulating that the system of the linear equations (11), (12), (15), and (16) are consistent, the solutions and equilibrium obtained by the distributed algorithm are asymptotically equivalent to those yielded by the convex program and its KKT conditions.*

Proof. We denote the best response of SU i at $t = \infty$ as $\tilde{\beta}_i$, the price made by PU l on subchannel j at $t = \infty$ as $\tilde{\pi}_{lj}$, and the interference charge made by SU i on subchannel j at $t = \infty$ as $\tilde{\eta}_{ij}$, then the KKT conditions (37)-(39) hold with Lagrangian multiplier $\kappa_i \geq 0$:

$$\tilde{\beta}_{ij} \left(\frac{e_i}{f_i(\tilde{\beta}_i)} \frac{\partial f_i(\tilde{\beta}_i)}{\partial p_{ij}} - \kappa_i \left(\sum_{l \in \mathcal{L}} \tilde{\pi}_{lj} G_{ij}^l + \sum_{k \in \mathcal{I}, k \neq i} \tilde{\eta}_{ij} G_{ij}^k \right) \right) = 0, \quad \forall j \in \mathcal{J}, \quad (37)$$

$$\frac{e_i}{f_i(\tilde{\beta}_i)} \frac{\partial f_i(\tilde{\beta}_i)}{\partial p_{ij}} - \kappa_i \left(\sum_{l \in \mathcal{L}} \tilde{\pi}_{lj} G_{ij}^l + \sum_{k \in \mathcal{I}, k \neq i} \tilde{\eta}_{ij} G_{ij}^k \right) \leq 0, \quad \forall j \in \mathcal{J}, \quad (38)$$

$$\kappa_i \left(\sum_{j \in \mathcal{J}} \tilde{\beta}_{ij} \sum_{l \in \mathcal{L}} \tilde{\pi}_{lj} G_{ij}^l + \sum_{j \in \mathcal{J}} \tilde{\beta}_{ij} \sum_{k \in \mathcal{I}, k \neq i} \tilde{\eta}_{ij} G_{ij}^k - e_i \right) = 0, \quad (39)$$

$$\sum_{j \in \mathcal{J}} \sum_{l \in \mathcal{L}} \tilde{\pi}_{lj} \tilde{\beta}_{ij} G_{ij}^l + \sum_{j \in \mathcal{J}} \tilde{\beta}_{ij} \sum_{k \in \mathcal{I}, k \neq i} \tilde{\eta}_{ij} G_{ij}^k \leq e_i. \quad (40)$$

Summing (37) over all $j \in \mathcal{J}$, we get (41), and, by Euler's theorem,

$$\sum_{j \in \mathcal{J}} \frac{e_i}{f_i(\tilde{\beta}_i)} \frac{\partial f_i(\tilde{\beta}_i)}{\partial p_{ij}} \tilde{\beta}_{ij} = \kappa_i \left(\sum_{j \in \mathcal{J}} \sum_{l \in \mathcal{L}} \tilde{\pi}_{lj} \tilde{\beta}_{ij} G_{ij}^l + \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{I}, k \neq i} \tilde{\eta}_{ij} \tilde{\beta}_{ij} G_{ij}^k \right), \quad (41)$$

$$e_i = \kappa_i \left(\sum_{j \in \mathcal{J}} \sum_{l \in \mathcal{L}} \tilde{\pi}_{lj} \tilde{\beta}_{ij} G_{ij}^l + \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{I}, k \neq i} \tilde{\eta}_{ij} \tilde{\beta}_{ij} G_{ij}^k \right). \quad (42)$$

Substituting e_i in (39) with (42), we get the following equation:

$$\kappa_i (1 - \kappa_i) \left(\sum_{j \in \mathcal{J}} \sum_{l \in \mathcal{L}} \tilde{\pi}_{lj} \tilde{\beta}_{ij} G_{ij}^l + \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{I}, k \neq i} \tilde{\eta}_{ij} \tilde{\beta}_{ij} G_{ij}^k \right) = 0. \quad (43)$$

For all $i \in \mathcal{I}$, at least one of $\tilde{\beta}_i$ should be nonzero to make $f_i(\tilde{\beta}_i) > 0$. Moreover, due to the strict monotonicity of f_i , the partial derivatives of f_i should be always positive. Therefore, at least one element of $\tilde{\pi}$ and $\tilde{\eta}$ should be nonzero to make (37) hold. We also set all the channel gains nonzeros. Subsequently, to make both (37) and (43) hold for all i , κ_i should be 1. Then, the KKT condition (37) and (38) turn out to be identical with (11) and (12), respectively, if we let $\tilde{\beta}_i = \tilde{\mathbf{p}}_i$ for all i , $\tilde{\pi} = \pi$, and $\tilde{\eta} = \eta$.

Since the two linear dynamics (i.e., (35) and (36)) are stable, there exist some $l \in \mathcal{L}$ and $j \in \mathcal{J}$ such that

$$\sum_{i \in \mathcal{I}} \tilde{\beta}_{ij}(t) G_{ij}^l = y_{lj}, \quad (44)$$

if $\tilde{\pi}_{lj} > 0$, or

$$\sum_{i \in \mathcal{I}} \tilde{\beta}_{ij}(t) G_{ij}^l < y_{lj}, \quad (45)$$

if $\tilde{\pi}_{lj} = 0$; in a similar way, there exist some $i \in \mathcal{I}$ and $j \in \mathcal{J}$ such that

$$\sum_{k \in \mathcal{I}, k \neq i} \tilde{\beta}_{ij}(t) G_{kj}^i = \gamma_{ij}, \quad (46)$$

if $\tilde{\eta}_{ij} > 0$, and

$$\sum_{k \in \mathcal{I}, k \neq i} \tilde{\beta}_{ij}(t) G_{kj}^i < \gamma_{ij}, \quad (47)$$

if $\tilde{\eta}_{ij} = 0$. We notice that (44) and (45) are equivalent to the KKT condition (13), and (46) and (47) are equivalent to the KKT condition (16) if we let $\tilde{\beta}_i = \tilde{\mathbf{p}}_i$ for all i , $\tilde{\pi} = \pi$, and $\tilde{\eta} = \eta$. Finally, (44)-(47) satisfy both (13) and (14) as well if we let $\tilde{\beta}_i = \tilde{\mathbf{p}}_i$ for all i .

For the reasons stated above, the stable solutions and the equilibrium yielded by the distributed algorithm satisfies the KKT condition of the convex program. Since the equilibrium as well as the solutions of the convex program are unique, we confirm $\tilde{\beta}_i = \tilde{\mathbf{p}}_i$ for all i , $\tilde{\pi} = \pi$, and $\tilde{\eta} = \eta$. Consequently, we conclude that the solutions and equilibrium yielded by the distributed algorithm are asymptotically equivalent to those given by the convex program and its KKT conditions. \square

7 NUMERICAL EVALUATIONS

7.1 Experimental Setup

We generate a CRN within a 500×500 square, and consider the frequency range of 54-862-MHz TV band following the IEEE 802.22 standard [30]. As mentioned in Section 3, we divide the frequency range into subchannels in the way that every subchannel yields equal transmission rate as possible given a constant transmission power. We vary the sizes of SUs, PUs, and subchannels according to the type of the experiment we perform. Besides, we use interior point optimizer (IPOPT) [31] for solving the convex program and the best-response dynamics, and GNU linear programming kit (GLPK) [32] for solving the system of the linear equations. Additional experimental parameters are the following:

- e_i : randomly chosen from (0, 1.0].
- y_{lj} : $\forall l \in \mathcal{L}$ and $\forall j \in \mathcal{J}_l$, 1e-08.
- γ_{ij} : $\forall i \in \mathcal{I}$ and $\forall j \in \mathcal{J}$, 1e-08.
- Initial π_{lj} for the distributed algorithm: $\forall l \in \mathcal{L}$ and $\forall j \in \mathcal{J}_l$, 6e06.
- Initial η_{ij} for the distributed algorithm: $\forall i \in \mathcal{I}$ and $\forall j \in \mathcal{J}$, 6e06.
- Antenna gain: 1.0 for both transmitter side and receiver side.
- System loss factor: 1.0.
- Speed of light: 3e08 m/s.
- Thermal noise: 1e-10.
- PU's transmission power: 0.1W for all PUs.

7.2 Illustration of the Strict Monotonicity

First, we illustrate the strict monotonicity of the monotone-transformed utility function, i.e., f_i . To this end, we consider two subchannels and compute the function values of f_i varying the transmission power on each subchannel. Fig. 2b shows the results. We plot also the function values of the original utility function, i.e., u_i in Fig. 2a. As shown in

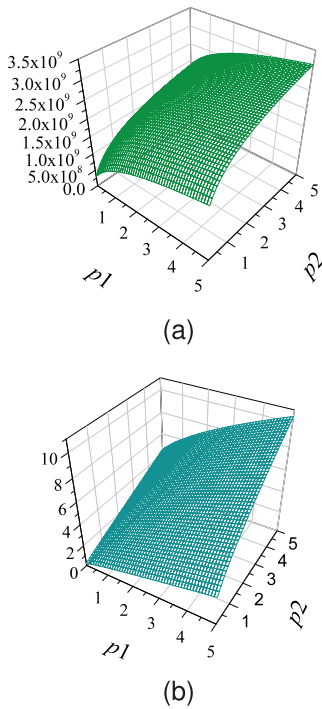


Fig. 2. Illustrations of the strict monotonicity of u_i and f_i .

Fig. 2b, the function value increases strictly monotonically as the transmission power on each subchannel increases, and which illustrates the strict monotonicity.

7.3 Illustration and Evaluation of Equilibrium Price

Next, we measure the transition of the total interference demand according to the change of the prices given by PUs when the function (8) is maximized subject to the budget constraint (i.e., (6)). For these experiments, we locate three SUs and two PUs accommodating two subchannels,⁴ and ignore the limit of the interference from other SUs, that is ignore (9). Thus, all the budget possessed by the SUs will be paid to the PUs. Setting $y_{lj} = 8e-08$ for all l, j , the measured results are shown in Fig. 3, and it is observed that the demand decreases as the prices increase. By the definition, the equilibrium price is obtained when the total interference demand equals $2 \times \sum_{l,j} y_{lj}$, that is, $1.6e-07$.

The next set of experiments is done to evaluate the precision of the equilibrium price obtained by solving the system of the linear equations (11)-(16) under various precision bound. For this set of experiments, we locate eight SUs and eight PUs, and arrange 32 subchannels. Then we measure the absolute gap between the initial budget and the payment of each SU under three different precision bounds. Fig. 4 shows the results. If the precision bound is made lower than $1.4508e-5$, then the system becomes inconsistent. As shown in the graph, the absolute gaps are measured at most around $4.0e-4$ with the smallest feasible precision bound.

7.4 Evaluation of the Distributed Algorithm

In this section, we illustrate the convergence process of the distributed algorithm, and evaluate it in terms of convergence speed and solution quality.

4. We let each PU have one subchannel.

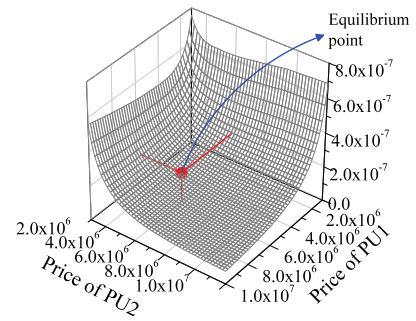


Fig. 3. The transition of the total interference demand according to the prices when we locate three SUs, two PUs, and eight subchannels. This figure also plots the equilibrium point; the price pair given by the equilibrium point is the equilibrium price.

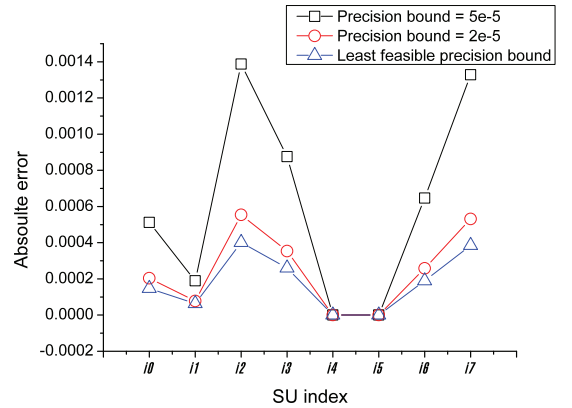


Fig. 4. The absolute gap between the initial budget of each SU and her payment decided by the KKT conditions of the convex program.

7.4.1 Illustration of the Convergence Process

First, we illustrate the convergence process to the equilibrium point with the distributed algorithm. In this set of experiments, we consider three SUs, three PUs, and eight subchannels, and apply the constant adaptive size of $1e13$. In Fig. 5, we plot the convergence trajectories as the iteration proceeds. Each axis on each graph indicates the utility of each SU (see Fig. 5a), the utility of each PU (see Fig. 5b), the total price gathered by each PU (see Fig. 5c), and the total interference charge gathered by each SU (see Fig. 5d). It is observed that, as the algorithm approaches to the equilibrium point, the amount of update in each iteration decreases, and which illustrates the asymptotic stability of the algorithm.

7.4.2 Illustration of the Convergence Speed and the Solution Quality

Next, we evaluate the distributed algorithm in terms of the convergence speed and the solution quality. In these experiments, eight SUs and eight PUs are located accommodating 32 subchannels. First, we measure the number of iterations required to reach the termination condition. Here, we define the termination condition as the errors in the KKT optimality conditions. The results are shown in Fig. 6 where we plot the objective function value of (8) on every iteration. In addition, the speed of adjustment is set $2e13$. As shown in the graph, as the KKT error is larger, the distributed algorithm converges faster. The function value yielded by the convex program is 79.53536. With KTT error of $1e-2$, the algorithm terminates after 33 iterations, and the value of

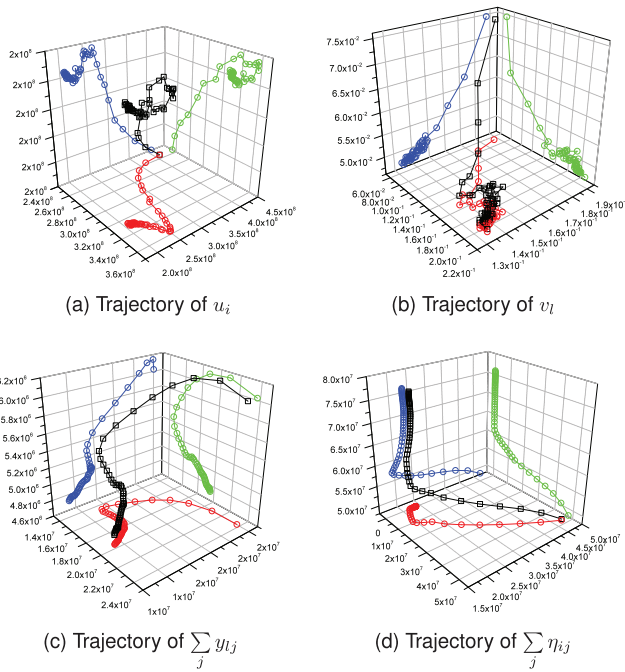


Fig. 5. The trajectories of the transitions of the utilities of SUs and PUs, prices, and interference charges as the distributed algorithm proceeds. We also plot the projection of each trajectory to the surfaces.

the objective function measured 79.43998. With KKT error of $4e-3$, the algorithm stops at 188th iteration, and the value is measured 79.59514. With KKT error of $2e-3$, the algorithm terminates at 303rd iteration with the function value of 79.53107. Therefore, we notice the tradeoff between the convergence speed and the solution quality.

Furthermore, we measure the utilities of the SUs obtained by the distributed algorithm and those yielded by the convex program. Fig. 7 plots the results. The largest difference is measured around $3e6$ at $i6$, but in percentage, it is 0.67 percent.

We also measure the absolute gaps between the initial budget of each SU and her payment yielded by the distributed algorithm as we have done with the KKT conditions of the convex program in Section 7.3. The measured results are plotted in Fig. 8 together with the results with the KKT conditions. We see that the distributed algorithm yields larger gaps than the KKT conditions, and the largest absolute gap is measured $3.69e-4$ at most.

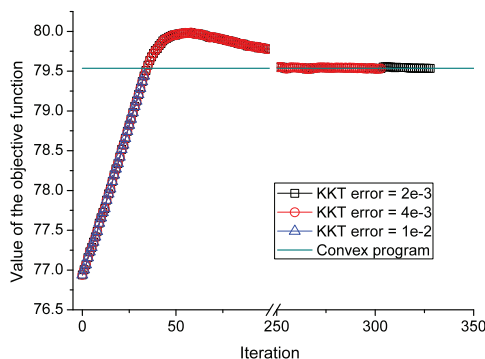


Fig. 6. Graphical presentation of the number of iterations with various termination conditions (i.e., KKT error). The solid green line indicates the optimal value of the objective function yielded by the convex program. Authorized licensed use limited to: Kwangju Institute of Science and Technology. Downloaded on January 11, 2024 at 05:40:35 UTC from IEEE Xplore. Restrictions apply.

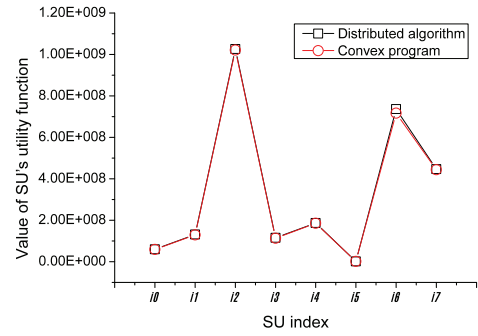


Fig. 7. The comparison of SUs utility.

8 CONCLUDING REMARK

In this paper, we consider a market equilibrium in multi-channel sharing CRN. PUs and SUs act as suppliers and purchasers, respectively: PUs offer their subchannels to SUs with bounding the total amount of interference invoked from SUs transmissions, and SUs purchase the offered subchannels observing their budget constraints and the interference bounds given by the PUs. Moreover, we consider that each SU sets the constraint of interference from other SUs. Accordingly, SUs pay the interference charges to other SUs she interferes. The utility functions of SUs and PUs are given as the least achievable transmission rates and the net profits, respectively.

The market equilibrium not only optimizes all traders' (PUs and SUs) utilities but also achieves the market clearance. We show that the market equilibrium is yielded by solving the optimization problem called the Eisenberg-Gale convex program, and the equilibrium price is given by the Lagrangian dual variables of the convex program. The convex program yields the equilibrium only when the utility functions of SUs are homogeneous of degree one. Therefore, we apply a monotone-transformation to SUs utility functions with maintaining the strict monotonicity and concavity.

We also develop a distributed algorithm with which the traders can reach the market equilibrium without any central authority. However, it is impossible to yield the exact equilibrium price since the system of the linear equations—that are composed of the KKT conditions of the convex program—are normally inconsistent, and the convergence behavior of the distributed algorithm is asymptotic. For these reasons, we provide the system of the linear

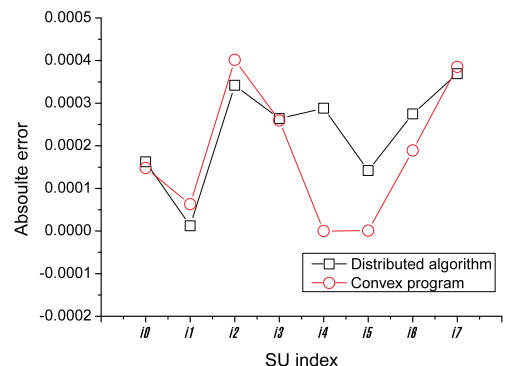


Fig. 8. The comparison of the absolute gaps between the initial budget of each SU and her total payment.

equations with a certain precision bound that makes the system consistent, and give a termination threshold to the distributed algorithm.

By the numerical experiments, we illustrate the strict monotonicity of the monotone-transformed function, and present graphically the existence of the equilibrium and the convergence process of the distributed algorithm. We also measure the absolute errors in the solutions and KKT conditions, which is yielded by the precision bound in the system of the linear equations and the asymptotic optimality of the distributed algorithm. The measured results show that the solutions achieved by the distributed algorithm are quite close to those of the convex program.

REFERENCES

- [1] A. O'Sullivan and S.M. Sheffrin, *Economics: Principles in Action*. Prentice Hall, 2003.
- [2] A. Al Daoud, M. Alanyali, and D. Starobinski, "Pricing Strategies for Spectrum Lease in Secondary Markets," *IEEE/ACM Trans. Networking*, vol. 18, no. 2, pp. 462-475, Apr. 2010.
- [3] "Secondary Markets Initiative. Federal Communications Commission," <http://wireless.fcc.gov/licensing/secondarymarkets>, 2013.
- [4] "Radio Spectrum Policy Group," <http://rspg.groups.eu.int>, 2013.
- [5] K. Binmore, *Playing for Real: A Text on Game Theory*. Oxford Univ. Press, 2007.
- [6] K. Jain, V.V. Vazirani, and Y. Ye, "Market Equilibria for Homothetic, Quasi-Concave Utilities and Economies of Scale in Production," *Proc. ACM-SIAM Symp. Discrete Algorithms*, pp. 63-71, Jan. 2005.
- [7] B. Birnbaum, N.R. Devanur, and L. Xiao, "Distributed Algorithms via Gradient Descent for Fisher Markets," *Proc. ACM Conf. Electronic Commerce*, pp. 127-136, June 2011.
- [8] Y. Xing, R. Chandramouli, and C. Cordeiro, "Price Dynamics in Competitive Agile Spectrum Access Markets," *IEEE J. Selected Areas Comm.*, vol. 25, no. 3, pp. 613-621, Apr. 2007.
- [9] M. Hong and A. Garcia, "Competitive Sharing of the Spectrum in Cognitive Radio Network: A Market Equilibrium Framework," *Proc. Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, pp. 40-49, June 2010.
- [10] D. Niyato and E. Hossain, "Hierarchical Spectrum Sharing in Cognitive Radio: A Microeconomic Approach," *Proc. IEEE Wireless Comm. and Networking Conf. (WCNS)*, pp. 3822-3826, Mar. 2007.
- [11] D. Niyato and E. Hossain, "A Microeconomic Model for Hierarchical Bandwidth Sharing in Dynamic Spectrum Access Networks," *IEEE Trans. Computers*, vol. 59, no. 7, pp. 865-877, July 2010.
- [12] H. Xu, J. Jin, and B. Li, "A Secondary Market for Spectrum," *Proc. IEEE INFOCOM*, Mar. 2010.
- [13] D. Li, Y. Xu, J. Liu, X. Wang, and Z. Han, "A Market Game for Dynamic multiBand Sharing in Cognitive Radio Networks," *Proc. IEEE Int'l Conf. Comm. (ICC)*, pp. 1-5, May. 2010.
- [14] D. Xu, X. Liu, and Z. Han, "A Two-Tier Market for Decentralized Dynamic Spectrum Access in Cognitive Radio Networks," *Proc. IEEE Sensor Mesh and Ad Hoc Comm. and Networks (SECON)*, pp. 1-9, June 2010.
- [15] Y. Xie, B. Armbruster, and Y. Ye, "Dynamic Spectrum Management with the Competitive Market Model," *IEEE Trans. Signal Processing*, vol. 58, no. 4, pp. 2442-2446, Apr. 2010.
- [16] I. Koutsopoulos and G. Iosifidis, "Auction Mechanisms for Network Resource Allocation," *Proc. Int'l Symp. Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, pp. 554-563, 2010.
- [17] N.R. Devanur, C.H. Papadimitriou, A. Saberi, and V.V. Vazirani, "Market Equilibrium via a Primal-Dual Algorithm for a Convex Program," *J. ACM*, vol. 55, no. 5, pp. 22:1-22:18, Nov. 2008.
- [18] E. Eisenberg and D. Gale, "Consensus of Subjective Probabilities: The Pari-Mutuel Method," *Annals of Math. Statistics*, vol. 30, pp. 165-168, 1959.
- [19] E. Eisenberg, "Aggregation of Utility Functions," *Management Sciences*, vol. 7, no. 4, pp. 337-350, 1961.
- [20] K. Arrow and G. Debreu, "Existence of an Equilibrium for a Competitive Economy," *Econometrica*, vol. 22, no. 3, pp. 265-290, 1954.
- [21] M. Chiang, C.W. Tan, D.P. Palomar, D. O'Neill, and D. Julian, "Power Control by Geometric Programming," *IEEE Trans. Wireless Comm.*, vol. 6, no. 7, pp. 2640-2651, July 2007.
- [22] M. Rice, *RF and Microwave Wireless Systems*. Wiley, 2009.
- [23] B. Codenotti and K. Varadarajan, "Computation of Market Equilibria by Convex Programming," *Algorithmic Game Theory*, chapter 6, N. Nisan T. Roughgarden E. Tardos, and V.V. Vazirani, eds., pp. 135-158, Cambridge Univ. Press, 2007.
- [24] C.P. Simon and L. Blume, *Mathematics for Economists*. Norton, 1994.
- [25] J.F. Nash, "The Bargaining Problem," *Econometrica*, vol. 28, pp. 155-162, 1950.
- [26] R.B. Myerson, *Game Theory: Analysis of Conflict*. Harvard Univ. Press, 1991.
- [27] A. Ravindran, D.T. Philips, and J.J. Solberg, *Operations Research: Principles and Practice*, second, ed. Wiley, 2007.
- [28] G. Owen, *Game Theory*, third, ed. Emerald, 2008.
- [29] U. Candogan, I. Menache, A. Ozdaglar, and P. Parrilo, "Near-Optimal Power Control in Wireless Networks: A Potential Game Approach," *Proc. IEEE INFOCOM*, pp. 1-9, Mar. 2010.
- [30] C.-K. Yu, K.-C. Chen, and S.-M. Cheng, "Cognitive Radio Network Tomography," *IEEE Trans. Vehicular Technology*, vol. 59, no. 4, pp. 1980-1997, May 2010.
- [31] A. Wächter and L.T. Biegler, "On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming," *Math. Programming*, vol. 106, no. 1, pp. 25-57, Jan. 2006.
- [32] "GNU Linear Programming Kit (GLPK)," <http://www.gnu.org/software/glpk/>, 2013.



Sang-Seon Byun received the BS, MS, and PhD degrees in computer science from Korea University, Seoul, in 1996, 2002, and 2007, respectively. He was an assistant research professor of the Graduate School of Embedded Software, Korea University, in 2007. From 2007 to 2011, he has served as a postdoctoral researcher and research scientist of the Department Electronics and Telecommunications, Norwegian University of Science and Technology, Trondheim, Norway. He has also worked as a research professor in the School of Information and Communications, Gwanju Institute of Science and Technology, Korea. Currently, he is a senior researcher of Daegu-Gyeongbuk Medical Innovation Foundation, Korea. His research interests include the fields of cognitive radio networks and software-defined radio architecture. He is a member of the IEEE.



Ilango Balasingham received the MSc and PhD degrees in signal processing from the Department of Telecommunications, Norwegian University of Science and Technology (NTNU), Trondheim, Norway, in 1993 and 1998, respectively. He performed the master's degree thesis at the Department of Electrical and Computer Engineering, University of California, Santa Barbara. From 1998 to 2002, he worked as a research scientist at Fast Search and Transfer ASA, Oslo, Norway. Since 2002, he has been with the Intervention Centre, Oslo University Hospital, Norway, as a senior research scientist, where he heads the Wireless Sensor Network Research Group. He was appointed as a professor in signal processing in medical applications at NTNU in 2006. His research interests include super robust short-range communications for both in-body and on-body sensors, body area sensor network, microwave short-range sensing of vital signs, and short-range localization and tracking sensors, catheters, and micro robots. He is the general chair of the 2012 Bodynets conference in Norway and serves an area editor of the Elsevier Nano Communication Networks. He is a senior member of the IEEE.



Athanasios V. Vasilakos is currently a professor at the Department of Computer and Telecommunications Engineering, University of Western Macedonia, Greece, and a visiting professor at the Graduate Programme of the Department of Electrical and Computer Engineering, National Technical University of Athens. He has authored or coauthored more than 200 technical papers in major international journals and conferences. He is author/coauthor of five books, 20 book chapters in the areas of communications.

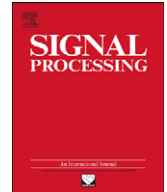
He served as a general chair, TPC chair, and symposium chair for many international conferences. He served or is serving as an editor or/and guest editor for many technical journals, i.e., *IEEE TNSM*, *IEEE TSMC-Part B*, *IEEE TITB*, and *JSAC*. He is founding editor-in-chief of the journals: *International Journal of Adaptive and Autonomous Communications Systems* (IJAACS, <http://www.inderscience.com/ijaacs>), *International Journal of Arts and Technology* (IJART, <http://www.inderscience.com/ijart>). He is the chairman of the European Alliance for Innovation. He is a senior member of the IEEE.



Heung-No Lee received the BS, MS, and PhD degrees in electrical engineering from the University of California at Los Angeles, in 1993, 1994, and 1999, respectively. He then moved to HRL Laboratory, Malibu, California, and worked there as a research staff member from 1999 to 2002. He was then appointed as an assistant professor at the University of Pittsburgh, Pennsylvania, in 2002 where he stayed until 2008. He then moved to Gwangju

Institute of Science and Technology in 2009. His research interests include information theory, signal processing theory, and communications/networking theory, and their application to wireless communications and networking, compressive sensing, future Internet, and brain computer interface. He has served as the lead guest editor for *EURASIP Journal on Wireless Communications and Networking* from 2010 to 2011. He has also served as a member of technical program committees for several renowned IEEE conferences, including the IEEE International Conference on Communications, and Globecom. He is a member of the IEEE.

▷ **For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.**



Fast communication

MB iterative decoding algorithm on systematic LDGM codes: Performance evaluation

Cheng-Chun Chang^a, Zhi-Hong Mao^a, Heung-No Lee^{b,*}^a Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, PA, USA^b Gwangju Institute of Science and Technology (GIST), Republic of Korea

ARTICLE INFO

Article history:

Received 9 May 2008

Received in revised form

22 May 2009

Accepted 25 May 2009

Available online 31 May 2009

Keywords:

Systematic LDGM codes

Majority-rule decoding

Performance bounds

ABSTRACT

We investigate the performance of regular systematic low-density generator matrix (LDGM) codes under the majority rule based (MB) iterative decoding algorithm. We derive a recursive form which can be used to extract the error performance of the code. Based on the recursive expression, we derive a tight non-recursive lower bound. These results can serve as efficient tools to evaluate the performance of the code for different degrees.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Systematic low-density generator matrix (LDGM) codes with moderate code length are of interest not only because they can provide satisfying performance at moderate block length while maintaining low encoding and decoding complexities [1–4], but also because the systematic form of LDGM codes makes the code useful in new applications such as cooperative wireless multiple access relay network [5] and joint source-channel encoding systems [6].

In this paper, we are interested in the performance of the majority rule based (MB) iterative decoding algorithm for systematic LDGM codes. Although the MB algorithm is based on hard-decision and thus its performance cannot match those based on soft-decision, it has drawn significant interest in the past owing to its simplicity and low computation complexity, which allow fast decoding [4,7–9].

We investigate the asymptotic performance of the code consisting of regular systematic LDGM codes and the MB

iterative decoding algorithm. By assuming infinite block length, we derive a recursive expression which predicts both the threshold and error floor behaviors of the code. Gallager has analyzed an MB iterative decoding for low-density parity-check (LDPC) codes; we build our analysis on systematic LDGM codes by extending his results reported in [9]. Based on the recursive expression, we further derive a non-recursive lower bound expression which is simply a function of the *degree of variable nodes*. We show that the bound is tight in simulation, and thus it can be useful to quickly assess the performance of the code for given degrees.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the systematic LDGM codes and the majority rule based iterative decoding algorithm. The recursive expression and the lower bound expression are derived in Section 3. In Section 4, computer simulation results and analysis results are compared and discussion is provided. Finally, we make a conclusion in Section 5.

2. Systematic LDGM codes and MB algorithm

Similar to the well known LDPC codes, systematic LDGM codes can be represented by sparse matrices,

* Corresponding author. Tel.: +82 62 970 2237; fax: +82 62 970 2204.

E-mail addresses: chc55@pitt.edu (C.-C. Chang), maozh@engr.pitt.edu (Z.-H. Mao), heungno@gist.ac.kr (H.-N. Lee).

see [10]. We briefly give the definition of systematic LDGM codes, and then introduce the majority rule based iterative decoding algorithm.

2.1. Systematic LDGM codes

Systematic LDGM codes are linear block codes with parity check matrix $\mathbf{H} = [\mathbf{P}; \mathbf{I}]$, where \mathbf{P} is an $(n-k)$ by k sparse matrix and \mathbf{I} is the $(n-k)$ by $(n-k)$ identity matrix. The positive integer k denotes the number of input bits and n denotes the number of output bits of a systematic LDGM encoder. The matrix \mathbf{P} of ones and zeros can be generated at random. A systematic LDGM code will be called *regular* if both the number of 1's in column in the \mathbf{P} matrix and that in row stay fixed for all columns and rows. Though irregularity can provide performance improvement, regularity could lead to simplified modular implementation in hardware realization. We will study the regular version only in this paper. We denote the *degree* of a variable node as d_v , which is the number of ones in each column in the \mathbf{P} matrix. Similarly, the degree of a check node, d_c , represents the number of ones in each row in the \mathbf{H} matrix. The code can be completely specified by a bipartite graph [11] consisting of check nodes and variable nodes. Since a systematic codeword is composed of message bits and parity-check bits, the variable nodes can be further separated into *message-bit variable (MV) nodes* and *parity-check-bit variable (PV) nodes*. Based on the structure of the \mathbf{H} matrix, the code rate R of (d_v, d_c) -regular systematic LDGM codes is given as $R = 1/(d_v/(d_c - 1) + 1)$.

It should be noticed that, when a code has a generator matrix with rows of constant weight, the code contains code words of the specified constant weight, and hence the minimum distance of the code could have been specified. In the case of regular systematic LDGM codes, the minimum distance of the code is not larger than d_v+1 , i.e., the weight of the rows of the generator matrix.

2.2. The majority-rule based iterative decoding algorithms

There are two steps in each iteration for the MB iterative decoding algorithm. The first step is done in a check node. The output binary message from the i th check node, toward the j th of its d_c variable nodes, is the result of the XOR operation on the rest of d_c-1 incoming binary messages. That is, $c_{i,j} = \oplus_{\sum_{k=1, (k \neq j)}^{d_c-1} (v_{k,j})}$, where the summation is done in modular-2 addition and $v_{k,j}$ is the binary message from the k th variable to the i th check node. The second step is done in a variable node at which the majority rule is applied. Let $f_j, f_j \in \{0, 1\}$, denote the hard-decision binary value of the received signal for the j th bit transmission. The output binary message from the j th variable node, toward the i th of its d_v check nodes, is obtained from the rest of d_v-1 incoming messages, and is given by

$$v_{j,i} = \begin{cases} \tilde{f}_j & \text{if } \left(\sum_{k=1, (k \neq i)}^{d_v-1} \text{XOR}(f_j, c_{k,j}) \right) \geq m, \\ f_j & \text{o.w.} \end{cases} \quad (1)$$

That is, if m or more incoming messages are violated, then the message $v_{j,i}$ is the complement of f_j ; otherwise, it holds the value of f_j . At the last iteration, the j th bit is decoded to be \tilde{f}_j if $(\sum_{k=1}^{d_v} \text{XOR}(f_j, c_{k,j})) \geq m$; otherwise, the j th bit is decoded to be f_j . In the algorithm, the weight m is an integer between 0 and d_v . The weight m needs to be carefully chosen in each iteration as it affects the performance of the MB iterative decoding algorithm. From the above description, we note that the MB iterative decoding algorithm is extremely simple.

3. Error performance analysis

In this section, we derive the recursive expression (2) and the tight lower bound expression (11). These expressions serve as efficient tools to extract the performance of the codec for given degrees of systematic LDGM codes.

Due to the hard decision characteristic of the MB decoding algorithm, we may assume all the coded bits are transmitted through a binary symmetric channel with error probability P_0 . Consider the error performance on an MV node. Assume infinite code length and unfold the MB iterative decoding onto a cycle free decoding tree. Then, the error probability for the message on the MV node after the i th iteration can be expressed by the recursive form

$$P_{i+1} = P_0(1 - f(m, P_i)) + (1 - P_0)g(m, P_i). \quad (2)$$

where

$$f(m, x) = \sum_{l=m}^{d_v-1} \binom{d_v-1}{l} \left(\frac{1 + (1 - 2P_0)(1 - 2x)^{d_c-2}}{2} \right)^l \times \left(\frac{1 - (1 - 2P_0)(1 - 2x)^{d_c-2}}{2} \right)^{d_v-1-l} \quad (3)$$

and

$$g(m, x) = \sum_{l=m}^{d_v-1} \binom{d_v-1}{l} \left(\frac{1 - (1 - 2P_0)(1 - 2x)^{d_c-2}}{2} \right)^l \times \left(\frac{1 + (1 - 2P_0)(1 - 2x)^{d_c-2}}{2} \right)^{d_v-1-l}. \quad (4)$$

The first term in (2) represents the probability of an event that the MV node was in the error state originally and the error correction mechanism of MB algorithm is not triggered because less than m extrinsic messages, out of d_v-1 total, are in violation. Thus the error in the variable node remains unchanged. The second term represents the probability of an event that the MV node was in the correct state but the error correction mechanism of MB algorithm is falsely triggered—because of m or more extrinsic messages in violation—and forces an error.

It is interesting to compare this recursion result (2) to Gallager's result on regular LDPC codes [9, p. 46]. The difference is that we have $(1 - 2P_0)(1 - 2x)^{d_c-2}$ in the recursion equation, instead of $(1 - 2x)^{d_c-1}$. This belongs to one of the characteristic results of the systematic LDGM codes. It is caused by the one and only one connection made by each PV node to the corresponding check node in the bipartite graph. This causes the error floor effect in systematic LDGM codes.

For a given channel error probability P_0 , the weight m and the degrees d_v and d_c determine the behavior of the recursive process (2) and hence the error performance. The optimal weight m which minimizes P_{i+1} in (2) for the i th iteration can be found by exhaustively searching for the integer between 0 and d_v , or by solving the smallest integer m which satisfies the following inequality [9]:

$$\frac{1 - P_0}{P_0} \leq \left(\frac{1 + (1 - 2P_0)(1 - 2P_i)^{d_c - 2}}{1 - (1 - 2P_0)(1 - 2P_i)^{d_c - 2}} \right)^{2m - d_v + 1}. \quad (5)$$

In the following, we derive the lower bound expression based on the recursive expression (2). Taking the partial derivative of (2) with respect to P_i , we obtain

$$\frac{\partial P_{i+1}}{\partial P_i} = -P_0 \frac{\partial f}{\partial P_i} + (1 - P_0) \frac{\partial g}{\partial P_i}, \quad (6)$$

where

$$\frac{\partial f}{\partial P_i} = \binom{d_v - 1}{m} (m \zeta^{+m-1} \zeta^{-d_v-1-m} \eta^-) \quad (7)$$

and

$$\frac{\partial g}{\partial P_i} = \binom{d_v - 1}{m} (m \zeta^{-m-1} \zeta^{+d_v-1-m} \eta^+). \quad (8)$$

In (7) and (8), the notations ζ^+ , ζ^- , η^+ , and η^- are defined as $\zeta^+ = (1 + (1 - 2P_0)(1 - 2P_i)^{d_c - 2})/2$, $\zeta^- = (1 - (1 - 2P_0)(1 - 2P_i)^{d_c - 2})/2$, $\eta^+ = (d_c - 2)(1 - 2P_0)(1 - 2P_i)^{d_c - 3}$, and $\eta^- = -(d_c - 2)(1 - 2P_0)(1 - 2P_i)^{d_c - 3}$.

Without loss of generality, we may assume that P_0 and P_i are restricted in the interval $[0, 0.5]$. Then, we observe that (6) is always non-negative, i.e., $\partial P_{i+1}/\partial P_i \geq 0$. This shows that (2) is a monotone increasing function of P_i . Therefore, by substituting $P_i = 0$ into (2), we can obtain a lower bound expression of P_{i+1} .

The weight m used in the lower bound expression can be determined via (5). By substituting $P_i = 0$ into (5), we have

$$\frac{1 - P_0}{P_0} \leq \left(\frac{1 + (1 - 2P_0)}{1 - (1 - 2P_0)} \right)^{2m - d_v + 1}. \quad (9)$$

If P_0 is restricted within the interval $[0, 0.5]$, then $(1 - P_0)/P_0$ is not less than 1. The inequality is satisfied if and only if the exponent of the right hand side is greater than 1, i.e., $2m - d_v + 1 \geq 1$. The smallest integer that satisfies this inequality is $m = \lceil d_v/2 \rceil$, where $\lceil \cdot \rceil$ is the ceiling operation. Notice that, at the last iteration, the number of available extrinsic messages for an MV node is d_v , instead of $d_v - 1$. Hence, we choose the weight m^* for a given (d_v, d_c) regular systematic LDGM as

$$m^* = \left\lceil \frac{d_v + 1}{2} \right\rceil. \quad (10)$$

Therefore, the lower bound expression, which is only a function of the degree of variable nodes d_v , is given by

$$P_{LB} = P_0 \left(1 - \sum_{l=m^*}^{d_v} \binom{d_v}{l} (1 - P_0)^l (P_0)^{d_v - l} \right) + (1 - P_0) \left(\sum_{l=m^*}^{d_v} \binom{d_v}{l} (P_0)^l (1 - P_0)^{d_v - l} \right). \quad (11)$$

We note that the results (2) and (11) are based on the cycle-free assumption, i.e., infinite code length. Thus, they render the best performance for given degrees of systematic LDGM codes. The tightness of the lower bound is illustrated through simulation in Section 4.

4. Simulation results and discussion

Assuming BPSK modulation over AWGN channels, the error probability of the equivalent binary symmetric channel (BSC) for the MB algorithm is obtained by $P_0 = 0.5 \operatorname{erfc}(\sqrt{RE_b/N_0})$, where R is the code rate, E_b is the energy per bit, and N_0 is the one sided power spectral density of the noise. We assess the best possible performance of the code by using the recursion form (2) in the following manner. While numerically evaluating the recursion expression (2), we test out all the possible choices of m in each iteration and then select the best value of m that results in the lowest error probability at the end of each iteration. In addition, we let a large number of iterations (more than 50) to ensure the convergence of the recursive form (2).

Fig. 1(a) shows the BER curves for rate half systematic LDGM codes with degree (8, 9), (9, 10), (10, 11), (11, 12), (12, 13), and (13, 14). Fig. 1(b) shows the BER curves for rate around 1/3 systematic LDGM codes with degree (9, 6), (10, 6), (11, 6), and (11, 7). The dashed curves in the figure are obtained from the recursive method (2), whereas the solid curves are obtained from the non-recursive lower bound (11). We note that the lower bound is asymptotically tight with respect to channel signal to noise ratio (SNR). This is expected because, at high SNR, P_i in (2) can evolve to a value very close to zero, and hence the assumption $P_i = 0$ we made to derive the lower bound becomes more valid. We also note that the lower bound expression predicts the performance well in the entire error floor region. Defining threshold as the SNR the waterfall starts, moreover, we note that a code with small degrees exhibits a high error floor but a small threshold; whereas a code with large degrees shows a low error floor but a larger threshold. Considering the trade-off relation between the error floor and the threshold behavior, the best degree d_v for systematic LDGM codes under MB iterative decoding algorithm can be selected. For example, we may select it to be 10 based on our results. Systematic LDGM codes with other degrees are not good, since they exhibit either a high error floor or a large threshold. In addition, we observe that, for rate half systematic LDGM codes, the curves of (d_v, d_v+1) and (d_v+1, d_v+2) converge asymptotically for even d_v . This is one of the characteristic behaviors of systematic LDGM codes which was also reported in [2].

Fig. 2 shows the BER curves obtained from the Monte Carlo computer simulation. Ten iterations are used in the MB algorithm. Two randomly constructed (8, 9) and (9, 10) systematic LDGM codes with length 6000 are used in simulation.

To draw best threshold behavior while maintaining a low error floor, we use the following strategy for selecting

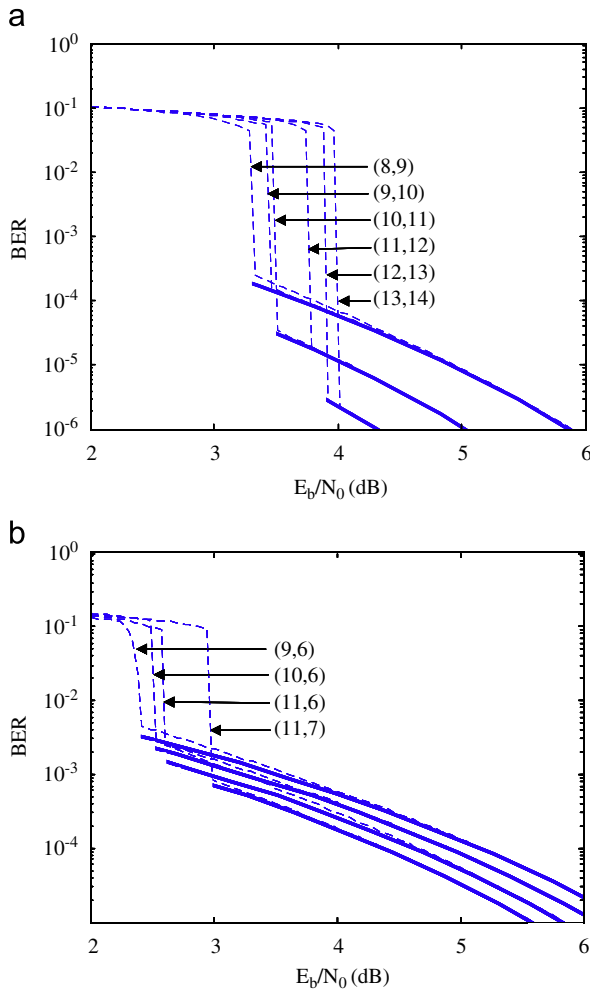


Fig. 1. BER performance of (a) rate half systematic LDGM codes and (b) rate around $\frac{1}{3}$ systematic LDGM codes. In the figure, the dashed curves represent the BER obtained from the recursive expression (2) with dynamic optimized weight m ; the solid curves represent the BER obtained by the non-recursive lower-bound expression (11).

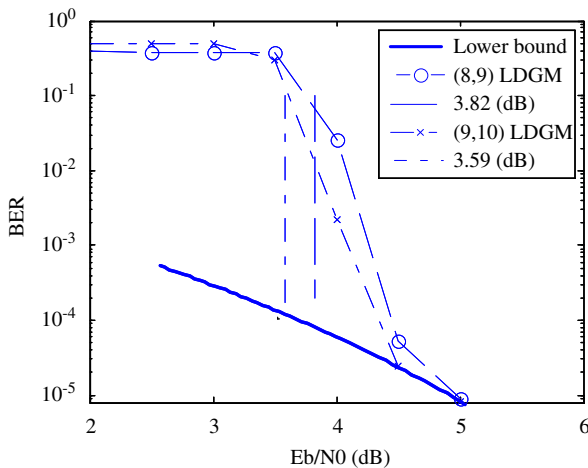


Fig. 2. Comparison between the Monte Carlo computer simulation results and the analytic results.

the weight m . Initially, we choose the weight m which has the smallest threshold. This initial weight is used all the way through the last iteration, and at the last iteration the weight calculated from (10) is used to push down the error floor. For the (8, 9) systematic LDGM code, the weight of the smallest threshold is $m = 5$ and the weight calculated from (10) is also $m = 5$. Hence, we select the weight to be 5 throughout the iterations. For the (9, 10) systematic LDGM codes, the weight of the smallest threshold is $m = 6$, whereas the weight calculated from (10) is $m = 5$. We choose $m = 6$ for the iterations all the way until the last one, and then choose $m = 5$ for the last iteration.

The simulation results show that both the (8, 9) systematic LDGM code and the (9, 10) systematic LDGM code not only can achieve the lower bound, but also can achieve the thresholds. In other words, the derived recursive expression and non-recursive lower bound are tight and can successfully serve as efficient tools to access the error performance of the codec.

5. Conclusion

Systematic LDGM codes and the majority-rule based iterative decoding algorithm may be of interest for communications system engineers because they render simple encoding and decoding complexities. The codec exhibits two eminent error performance behaviors, the threshold and the error floor. We have provided the analytic expression (2) for efficiently assessing the best possible performance of the codec for given degrees of systematic LDGM codes. Furthermore, a simple tight lower bound expression (11) has been derived, which can be readily evaluated once the degree of variable nodes is given.

References

- [1] J.F. Cheng, R.J. McEliece, Some high-rate near capacity codes for the Gaussian channel, in: Proceedings of 34th Allerton Conference on Communication, Control and Computing, Monticello, IL, October 1996.
- [2] J. Garcia-Frias, W. Zhong, Approaching Shannon performance by iterative decoding of linear codes with low-density generator matrix, *IEEE Communication Letters* 7 (6) (2003) 266–268.
- [3] L. Ping, S. Chan, K.L. Yeung, Iterative decoding of multi-dimensional concatenated single parity check codes, in: Proceedings of IEEE International Conference on Communication, Atlanta, GA, June 1998, pp. 131–135.
- [4] M.G. Luby, M. Mitzenmacher, M.A. Shokrollahi, D.A. Spielman, V. Stemann, Practical loss-resilient codes, in: Proceedings of 29th ACM Symposium on Theory of Computing, El Paso, TX, May 1997, pp. 150–159.
- [5] X. Bao, J. Li, Matching code-on-graph with network-on-graph: adaptive network coding for wireless relay networks, in: Proceedings of 43th Allerton Conference on Communication, Control, and Computing, Champaign, IL, September 2005.
- [6] W. Zhong, J. Garcia-Frias, LDGM codes for channel coding and joint source-channel coding of correlated sources, *EURASIP Journal on Applied Signal Processing* 2005 (1) (2005) 942–955.
- [7] P. Zarrinkhat, A.H. Banihashemi, Threshold values and convergence properties of majority-based algorithms for decoding regular low-density parity-check codes, *IEEE Transactions on Communication* 52 (12) (2004) 2087–2097.
- [8] L. Bazzi, T. Richardson, R. Urbanke, Exact thresholds and optimal codes for the binary symmetric channel and Gallager's decoding

- algorithm A, IEEE Transactions on Information Theory 50 (9) (2004) 2010–2021.
- [9] R.G. Gallager, Low-Density Parity-Check Codes, MIT Press, Cambridge, MA, 1963.
- [10] T. Richardson, R. Urbanke, Modern Coding Theory, Cambridge University Press, Cambridge, 2008.
- [11] R.M. Tanner, A recursive approach to low complexity codes, IEEE Transactions on Information Theory 27 (5) (1981) 533–547.

Date of submission xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.0322000

Enhancing Zero-Shot Crypto Sentiment with Fine-tuned Language Model and Prompt Engineering

RAHMAN S M WAHIDUR¹, ISHMAM TASHDEED², MANJIT KAUR³, (Senior Member, IEEE), and HEUNG-NO-LEE¹, (Senior Member, IEEE)

¹School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea

²Department of Computer Science and Engineering, Islamic University of Technology, Dhaka, Bangladesh.

³School of Computer Science and Artificial Intelligence, SR University, Warangal, Telangana, India-506371.

Corresponding author: Heung-No Lee (heungno@gist.ac.kr).

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2023-2021-0-00118, Development of decentralized consensus composition technology for large-scale nodes) and This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2023-2021-0-01835) supervised by the IITP (Institute of Information & Communications Technology Planning & Evaluation)

ABSTRACT Blockchain technology has revolutionized the financial landscape, with cryptocurrencies gaining widespread adoption for their decentralized and transparent nature. As the sentiment expressed on social media platforms can significantly influence cryptocurrency discussions and market movements, sentiment analysis has emerged as a crucial tool for understanding public opinion and predicting market trends. Motivated by the aim to enhance sentiment analysis accuracy in the cryptocurrency domain, this paper investigates fine-tuning techniques on large language models. This paper also investigates the efficacy of supervised fine-tuning and instruction-based fine-tuning on large language models for unseen tasks. Experimental results demonstrate a significant average zero-shot performance gain of 40% after fine-tuning, highlighting the potential of this technique in optimizing pre-trained language model efficiency. Additionally, the impact of instruction tuning on models of varying scales is examined, revealing that larger models benefit from instruction tuning, achieving the highest average accuracy score of 75.16%. In contrast, smaller-scale models may experience reduced generalization due to the complete utilization of model capacity. To gain deeper insight about how instruction works with these language models, this paper presents an experimental investigation into the response of an instruction-based model under different instruction tuning setups. The investigation demonstrates that the model achieves an average accuracy score of 72.38% for short and simple instructions. This performance significantly outperforms its accuracy under long and complex instructions by over 12%, thereby effectively highlighting the profound significance of instruction characteristics in maximizing model performance. Finally, this paper explores the relationship between fine-tuning corpus size and model performance, finding an optimal corpus size of 6,000 data points for achieving the highest performance across different language models. Significantly, a distillation model of BERT called MiniLM, published by the Microsoft research team, stands out for its exceptional data efficiency, effectively optimizing its performance while making efficient use of data. Conversely, Fine-tuned LAnguage Net, FLAN-T5 developed by the Google research team, impressively maintains consistent and reliable performance across diverse corpora, further affirming its robustness and versatility.

INDEX TERMS Zero-Shot Learning, In-context Learning, Supervised fine-tuning, Instruction Tuned, Prompt Engineering.

I. INTRODUCTION

IN the recent decade, cryptocurrency has gained much traction in finance and business for its decentralization, per-

missionless, and open nature built on the public blockchain [1]. This architecture can create an immutable and highly interoperable financial system with unprecedented trans-

parency, equal access rights, and little need for central authorities that smart contracts can control [2]. Cryptocurrencies employ cryptographic ciphers to facilitate financial transactions, distinguishing them from traditional forms of currency [3]. Cryptocurrencies are the first pure digital assets to be included by asset managers [4]. They mitigate double-spending by leveraging multiple verifications from neighboring nodes within the blockchain network. As the number of confirmations grows, the transaction gains enhanced reliability and become highly irreversible. Because of these favorable attributes and the widespread accessibility of cryptocurrencies, they can be a means of transaction and a store of wealth [4], [5].

In cryptocurrency discussions, social media networks have gained significant importance as information-sharing platforms. Communicating crypto events and market conditions through social media is widely recognized. Moreover, there is a prevalent belief in the correlation between altcoin prices and the sentiment expressed within the Twitter community [6]. The demand for cryptocurrency is intricately tied to people's trust in Bitcoin and its underlying technology. As people's trust plays a significant role in the growth of the cryptocurrency market, the sentiment of the general population has a substantial impact on the future market capitalization of cryptocurrencies [7]. The effect of social media on crypto discourse is growing [8]. The sentiment expressed on social media platforms can greatly influence cryptocurrency discussions, shaping public opinion, driving market movements, disseminating information, and generating buzz. In 2019, over 300 million active monthly users shared their emotions in multiple languages on various social media platforms, including Twitter, Facebook, and YouTube. Among these platforms, Twitter has emerged as one of the most influential social media networks [9], [10]. Twitter stands out as a unique platform due to its straightforward way of gauging individuals' sentiments toward a text. Performing real-time data analysis is also possible on the Twitter platform [11].

Sentiment analysis is a sub-research area of computational Natural Language Processing (NLP) studies and a widely used contextual mining technique for extracting valuable and subjective information from text-based data [6], [12]. There are several approaches to sentiment analysis, which is the process of determining the sentiment or emotion expressed in the text based on the type and size of the corpus. The rule-based approaches use a set of predefined rules, such as regular expressions and dictionaries, to classify text as positive, negative, or neutral [12]. Machine learning approaches try to find patterns from the provided data. This approach involves training a machine learning model on a labeled dataset to classify new text. These machine-learning techniques are further expanded over supervised and unsupervised. Wherein supervised approaches are trained on an annotated dataset, unsupervised does not require work on experience to improve accuracy [13]. However, pre-trained models can be Fine-tuned on specific datasets, reducing the need for labeled data and computational resources [14].

In text mining, sentence-level sentiment analysis has emerged as a burgeoning area of research. This analytical process encompasses six fundamental stages: data collection, data preprocessing, feature extraction, model training, model evaluation, and model deployment. Feature extraction, in particular, holds significant importance in optimizing the model's efficacy. Conventionally, Bag-of-Words (BoW) [15] and N-gram [16] techniques are widely employed for this purpose. However, their one-hot word representation approach creates high-dimensional feature spaces and scalability challenges, failing to capture word sequences and their syntactic and semantic nuances. Word embedding models like word2vec [17] and Glove [18] have gained popularity for their ability to capture word semantics in high-dimensional spaces, although they require substantial training data and are susceptible to data sparsity issues for rare or out-of-vocabulary (OVV) [12] words, which can impact performance.

To address the abovementioned limitations, a transformer-based NLP model was proposed [19] in the paper "Attention is All You Need." The transformer architecture is based on self-attention, which allows the model to weigh the importance of different words in a sentence when making predictions. In short, it creates contextual awareness between the words of a sentence. BERT [20], T5 [21], GPT-3 [22], and LLaMA [23] are a few examples among all pre-trained transformer-based models for NLP tasks. All these models have shown remarkable performance on various NLP tasks, especially in few-shot learning. However, they are less successful in zero-shot learning [24]. This paper explores a simple but powerful method called instruction tuned to improve the performance of zero-shot learning of Large Language Models (LLMs) for cryptocurrency sentiment classification tasks. Moreover, this paper utilizes an In-context learning (ICL) and prompt engineering method to generate effective instructions using LLMs that allow the creation of effective instructional datasets for fine-tuning the pre-trained language models to extract the cryptocurrency sentiment from social media data.

The key contributions of this paper are summarized as follows:

- **Improved Model Efficiency through fine-tuning:** The study demonstrates that supervised fine-tuning and instruction tuning significantly enhance pre-trained language models' performance on unseen tasks related to cryptocurrency sentiment analysis. This research experimentally shows that the average accuracy scores significantly increased after fine-tuning. The findings provide compelling evidence that fine-tuning enhances model efficiency, with an average performance gain of 40%. This contributes to the practical application of fine-tuning as a powerful tool for optimizing pre-trained language model performance.
- **Benefits of Instruction Tuning and Model Scale:** Building upon previous research, the study investigates the impact of instruction tuning on different-sized lan-

guage models. By analyzing FLAN-T5 models of varying scales, it was discovered that larger models benefit from instruction tuning by improving generalization to new tasks. However, for small-scale models, instruction tuning had a detrimental effect on generalization, potentially due to the complete utilization of model capacity for learning the mixture of instruction tuning tasks. This observation enhances our understanding of the interplay between instruction tuning and model scale, providing valuable insights for future model development and deployment.

- **Impact of model performance under different instruction tuning setups:** The conducted experimentation aimed to compare the quality of models under different instruction tuning setups, specifically focusing on the response of the instruction-based model. By introducing diverse instructions of varying lengths and complexities, the study provided insights into the model's handling of different instruction types, revealing its effectiveness in understanding and executing short and simple instructions compared to long and complex ones, thereby emphasizing the importance of considering instruction characteristics in instruction tuning setups.
- **Impact of fine-tuning Corpus Size on Model Performance:** The research explores the relationship between the size of the fine-tuning corpus and the performance of language models. By varying the sample size, the study investigates the influence of data availability on model performance. The findings highlight the optimal corpus size of 6000 data points for achieving the highest performance for each model and the data efficiency of MiniLM in leveraging limited data.

The remaining paper is organized as follows: Section II provides background information and reviews of related works. Section III presents the comprehensive architecture of the proposed model. Section IV showcases the analysis and metrics of the experimental results used for evaluation. Finally, in Section V, the paper concludes and summarizes the findings while also discussing the potential areas for future research and research limitations.

II. RELATED WORK

Recently, researchers have perceived a growing interest in leveraging sentiment analysis techniques for cryptocurrency. Several models are available for sentiment acquisition, each with varying precision and applicability [25]. Hasan et al. [26] investigated public sentiment analysis by employing a machine learning algorithm, namely the Support Vector Machine. The researchers also utilized Chi-square for feature selection to mitigate sentence noise. A similar approach was used by Satrya et al. [27] to determine the polarity of the sentiment. Additionally, they used TF-IDF weighting to transform the data from text to numeric values. The methodology employed by Padmalatha et al. [28] is based on Naive Bayes model to analyze social media opinions. Prasad et al. [29]

designed an ensemble classifier to classify YouTube comments based on cryptocurrency. They used Decision Tree, K Nearest Neighbors, Random Forest Classifier, XGBoost, and a Logistic Regression base classifier to create a stacked ensemble model. Sasmaz et al. [6] studied the feasibility of automated sentiment analysis for cryptocurrencies using the Random Forest Classifier.

The Valence Aware Dictionary for Sentiment Reasoning (VADER), a widely utilized and straightforward sentiment calculation model, is commonly employed to predict cryptocurrency prices by analyzing cryptocurrency-related news and Tweets. Suardi et al. [30] used VADER to investigate the predictive power of information contained in social media tweets on bitcoin market dynamics. The extent to which Twitter sentiment analysis can predict price fluctuations for cryptocurrencies was examined by Oikonomopoulos et al. [31]. They employed VADER for sentiment analysis in their study. Jagini et al. [32] intend to analyze the effect of tweets on the stock price of Bitcoin. To calculate the associated sentiment, they also used VADER. Parekh et al. [33] proposed a hybrid and robust DL-Gues framework for cryptocurrency price prediction. They also utilized a similar VADER technique in their framework to extract the polarity from the Twitter sentiment.

Regarding the recent superior performance transformers-based model, Dwivedi et al. [34] utilized the BERT (Bidirectional Encoder Representation) to predict the sentiments of cryptocurrency news articles. Kim et al. [35] introduced CBITS, a Fine-tuned version of BERT designed explicitly for cryptocurrency sentiment analysis, mainly focusing on the Korean crypto market. Widiyanto et al. [36] have created a BERT model to assess sentiment analysis on cryptocurrency and NFT by utilizing data crawling and pre-processing using Rapiminer. The findings of Ortu et al. [37] indicate that incorporating features derived from BERT-based emotion classification of comments on GitHub and Reddit results in a notable improvement in the predictability of Bitcoin and Ethereum's hourly and daily return direction. Despite extensive exploration of sentiment extraction methods, including rule-based and machine learning-based approaches, they often prove inadequate in cryptocurrencies due to domain-specific jargon, slang, and sentiment ambiguity not adequately covered by general-purpose texts [35]. Rule-based systems, although less accurate, are limited by the need for more powerful linguistic resources [12]. On the other hand, machine learning-based methods tend to achieve higher accuracy but require large amounts of labeled training data and significant computational resources. Existing literature reveals a notable gap in utilizing large language models for sentiment extraction. Additionally, there is a lack of research using fine-tuning techniques, despite their potential to enhance performance and adaptability.

This study proposes integrating large language models and utilizing fine-tuning techniques in sentiment extraction to address these research gaps. By leveraging the capabilities of large language models and optimizing their performance

through fine-tuning, this research aims to overcome the limitations of current approaches and advance the accuracy and applicability of sentiment extraction, especially in cryptocurrency. This contribution expects to facilitate the development of more effective and comprehensive sentiment analysis methodologies, ultimately enhancing decision-making processes in related industries and domains.

III. PROPOSED SYSTEM

This section illustrates and describes the entire procedure adopted in this paper. It is subdivided into five sub-sections. Subsection III-A summarizes the proposed model along with its visualization. The overview of the core concept related to the proposed model is discussed in the following subsections.

A. PROBLEM FORMULATION

Let X be the set of tweets related to cryptocurrency where each tweet $x \in X$ is represented as a feature matrix $x \in \mathbb{R}^{1 \times n}$ thus, X can be represented as

$$X \in \mathbb{R}^{m \times n} \quad (1)$$

where m is the number of tweets and n is the number of features used to represent each tweet. Let Y be a set of sentiments represented as a target matrix

$$Y \in \mathbb{R}^{m \times 1} \quad (2)$$

where m is the number of tweet labels. For this experiment, only *positive*, *negative* sentiment tweets are considered. Thus dataset D can be represented as

$$D = (X^{(i)}, Y^{(i)} | X^{(i)} \in \mathbb{R}^{(1 \times n)}, Y^{(i)} \in \mathbb{R}^{(1 \times 1)}) \quad (3)$$

which consisting of pairs $(X^{(i)}, Y^{(i)})$. We can then define the model

$$f : \mathbb{R}^{(m \times n)} \rightarrow \mathbb{R}^{(m \times 1)} \quad (4)$$

that maps each tweet matrix $X \in \mathbb{R}^{(m \times n)}$ to its sentiment matrix $Y \in \mathbb{R}^{(m \times 1)}$. The goal is to train the model f on the data set D such that

$$\forall x \in X : f(x, \Theta) \in Y \quad (5)$$

where Θ is the model parameters. We can use various mathematical techniques, such as gradient descent, backpropagation, and cross-entropy loss, to train/fine-tune the model f on D . Moreover, We can use various optimization algorithms, such as stochastic gradient descent or Adam, to find the optimal set of parameters Θ^* that minimizes the cross-entropy loss. The loss function can be defined as:

$$L(\Theta) = - \sum_{i=1}^m y^{(i)} \log(f(x^{(i)}, \Theta)) + (1 - y^{(i)}) \log(1 - f(x^{(i)}, \Theta)) \quad (6)$$

where m is the number of training examples, $x^{(i)}$ is the i^{th} tweet vector in the training set, $y^{(i)}$ is the corresponding sentiment label (either 0 or 1 for negative or positive sentiment, respectively), and $f(x^{(i)}, \Theta)$ is the predicted sentiment score

for $x^{(i)}$ given the current set of parameters Θ . We aim to minimize the cross-entropy loss $L(\Theta)$ by finding the optimal set of parameters Θ^* that maximizes the likelihood of the training data D given the model.

Once the model is trained/fine-tuned, we can use it to predict the sentiment of new tweets related to cryptocurrency by feeding their feature matrix representations into the model and obtaining the predicted sentiment matrix as output. Mathematically the final model can be represented as

$$M_{eval} = (f_{tuned}, D_{test}) \quad (7)$$

where f_{tuned} is the trained/Fine-tuned model and D_{test} is the evaluation dataset.

B. PROPOSED SYSTEM ARCHITECTURE

The design overview of our proposed model is shown in Figure 1. The proposed architecture begins with the initial user interaction step and prompt generation. The user engages with the OpenAI "text-davinci-003" model, initiating the prompt generation process. The users provide contextual information using prompt templates, which input the subsequent stages. The provided text is transmitted to the language model's backend through an API call, facilitating communication between the users and the large language mode. Next, a content moderator component is employed to evaluate the content. If the content is determined to be unsafe, the model responds with a default message. In the case of safe content, the system proceeds through a safety gateway to engage with the core parts of the large language model. This model encompasses several key components, like a response generator, context processor, knowledge domain, and system response generator. The system leverages its knowledge domain through interaction with these components to generate an effective response based on the provided input. The resulting response, an AI-generated prompt, is then delivered to the user. Upon receiving the AI-generated prompt, a filtering process takes place. Human feedback is vital in determining whether the prompt should progress to the subsequent stages.

The primary objective of this filtering is to generate adequate instructions by harnessing the advantages of in-context learning inherent in large language models and to prune low-quality and repeated instructions before adding them to the task pool. Finally, the prompt generation is refined by incorporating user feedback and capitalizing on the model's contextual understanding to enhance its efficacy. The process involving user interaction, prompt generation, filtering, and human feedback collectively constitutes the in-context learning and prompt engineering phase. The overall process of generating instructions using in-context learning can be seen in Algorithm 1. This phase aims to iteratively improve the prompt generation procedure by leveraging user feedback and maximizing the model's contextual comprehension capabilities.

Following the generation of effective instructions, the subsequent objective is to create an augmented dataset. This dataset is constructed by concatenating the introductions with

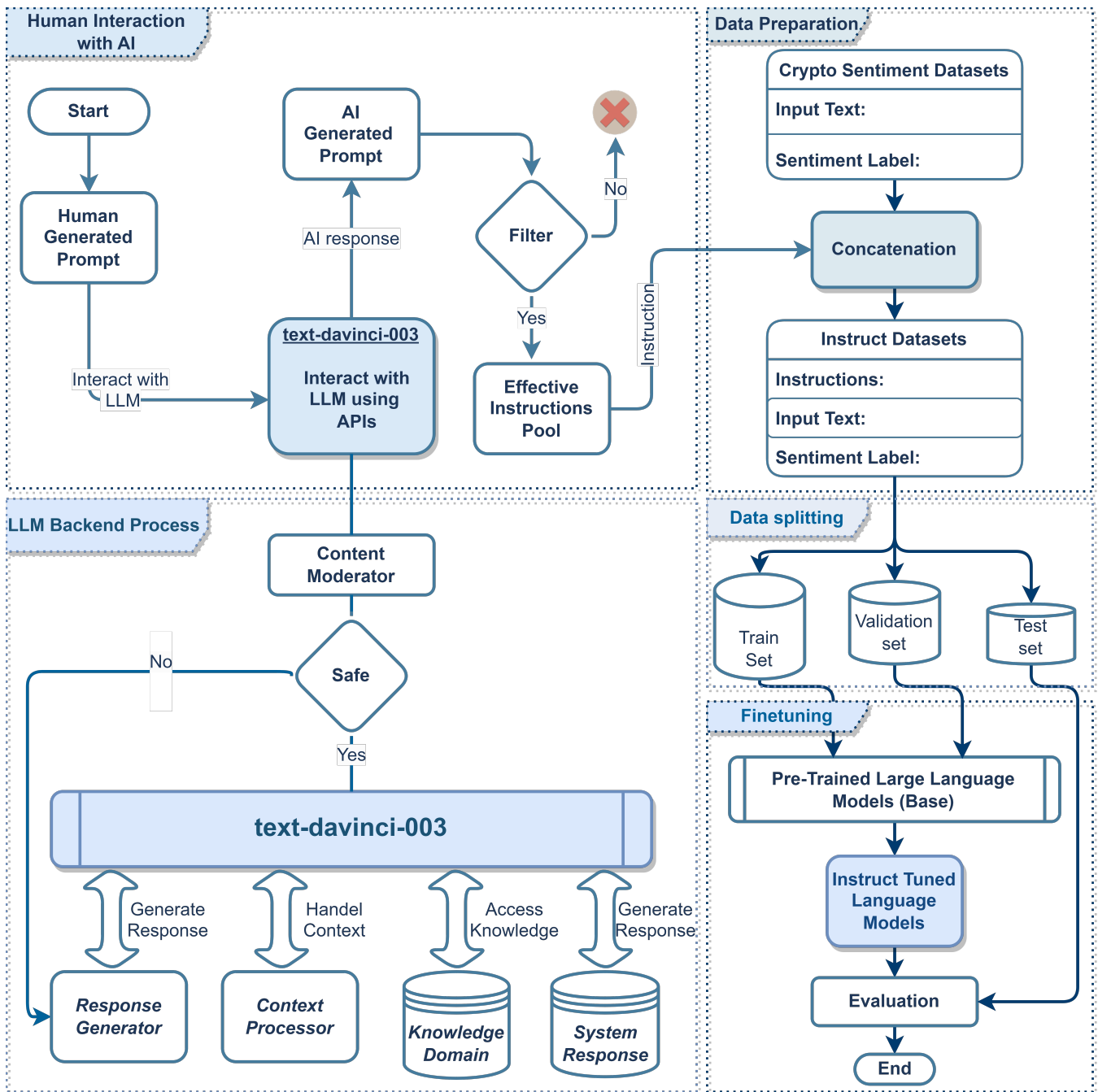


FIGURE 1. The Proposed System Model.

the original dataset, ensuring a comprehensive collection of relevant information for training purposes. Algorithm 2 shows the process of generating an augmented crypto sentiment dataset. The instructional dataset is then divided into separate training and validation sets. Three additional datasets are withheld to measure zero-shot performance, remaining untouched for evaluation. The large language models undergo fine-tuning at this stage, utilizing the instructional dataset. The initial model weights are modified, resulting in a newly instructed, Fine-tuned model version. The overall fine-tuning

process of a large language model can be observed in Algorithm 3. After the fine-tuning process, the performance of various models is evaluated. This evaluation provides valuable insights into the efficacy and effectiveness of the in-context learning and prompt engineering techniques employed in instructing fine-tuning for Zero-shot learning.

C. PRE-TRAINED LANGUAGE MODELS (PLMS)

Recently, there has been a significant emphasis on pre-trained language models (PLMs) that utilize self-supervised learning

Algorithm 1 Generating instructions using in-context learning.

Constant model, model, temp, max_len, top_p, penalty

Input A human-generated prompt $p \in P$.

Output An effective instructions pool Y .

- 1: Define the Language Model (LLM) as a function $G : P \rightarrow R$ that maps a given prompt $p \in P$ to a model-generated response $r \in R$.
 - 2: $r \leftarrow G(p; \text{mode}, \text{model}, \text{temp}, \text{max_len}, \text{top_p})$
 - 3: Pass the response through a filter $F : R \rightarrow (True, False)$
 - 4: **for** response $r \in R$ **do**
 - 5: **if** $F(R) = True$ **then**
 - 6: Add r to the effective instructions pool Y .
 - 7: **else**
 - 8: Discard the response r .
 - 9: **end if**
 - 10: **end for**
 - 11: **Return** Effective instructions pool Y
-

Algorithm 2 Generating augmented crypto sentiment dataset.

Input Effective instructions pool Y , and Crypto sentiment dataset X .

Output Augmented crypto sentiment dataset $X_{augmented}$.

- 1: Let function $C : x \rightarrow x$ which cleans a dataset entry.
 - 2: $X_{filtered} \subseteq X$ which contains only non-neutral sentiments.
 - 3: **for** $i \leftarrow (1 \dots \text{len}(X_{filtered}))$ **do**
 - 4: $X_{cleaned}^{(i)} \leftarrow C(X_{filtered}^{(i)})$.
 - 5: **end for**
 - 6: Select an instruction $y \in Y$.
 - 7: Let function $A : (x, y) \rightarrow x$ which augments each entry $x \in X$ with an instruction y .
 - 8: **for** $i \leftarrow (1 \dots \text{len}(X_{cleaned}))$ **do**
 - 9: $X_{augmented}^{(i)} \leftarrow A(X_{cleaned}^{(i)}, y)$.
 - 10: **end for**
 - 11: **Return** Augmented dataset $X_{augmented}$
-

on extensive raw text data [38]. Notable examples of such models include GPT-3 [22], PaLM [39], Chinchilla [40], LLaMA [23], and Falcon 40B [41]. By training on large-scale texts using self-learning tasks like masked word prediction, sentence sequence recognition, text completion, and text generation [19], [42] PLMs acquire a comprehensive understanding of language. In addition, these models enhance the semantic representation of words by considering contextual dynamics and provide a unified framework for various NLP tasks. Currently, there are three standard models [43] structures in PLMs: autoregressive language models, autoencoding language models, and hybrid language models. Representative models for each design are GPT [22], BERT [20], and T5 [21], respectively. Autoregressive language models follow a standard approach where language modeling is done decoder-only, predicting words one by one through one-way

Algorithm 3 Fine-tuning of a pre-trained language model.

Constant random_seed, input sequence size, number of layers, number of hidden layer nodes, number of classifier outputs.

Input Training set $(X_{train}^{(1)}, Y_{train}^{(1)}), \dots, (X_{train}^{(M)}, Y_{train}^{(M)})$.

Output Trained language model network parameters..

- 1: **for** $epoch \in \text{epochs}$ **do**
 - 2: **for** $batch \in \text{batches}$ **do**
 - 3: **for** $i \leftarrow (1 \dots \text{len}(batch))$ **do**
 - 4: $M_{train}^{(i)} \leftarrow \text{tokenizer}(X_{train}^{(i)})$
 - 5: $\hat{M}_{train}^{(i)} \leftarrow \text{model}(M_{train}^{(i)})$
 - 6: **end for**
 - 7: $loss \leftarrow E(Y_{train}^{(batch)}, \hat{Y}_{train}^{(batch)})$
 - 8: Calculate $\nabla \Theta$ for backpropagation.
 - 9: Adjust parameters using an optimizer to minimize the loss.
 - 10: **end for**
 - 11: **end for**
 - 12: **Return** Task-specific Fine-tuned model.
-

language encoding-decoding and token-by-token prediction of words. Autoencoding language models randomly mask words in a sentence, use bidirectional encoding to capture context, and then predict the masked words based on the encoded information. Finally, hybrid language models combine the approaches of the previous two models. They mask words randomly in a sentence, apply bidirectional encoding, and predict subsequent words step by step by inputting the earlier text in one direction [43].

The advancement of artificial intelligence technology has demonstrated that LLMs can acquire a deeper understanding of language and exhibit stronger capabilities in understanding and generating data. These models learn abstract knowledge from raw data, resulting in better generality and generalization. The autoregressive language model adopted by GPT-3 and its subsequent models, such as GPT3.5 and GPT4, has proven advantageous in utilizing natural language for various tasks in different fields [43]. Initially, it was believed that increasing the number of parameters in models would lead to better performance. However, recent research by Hoffmann et al. [23] has shown that smaller models trained on more data can achieve the best performances given a specific computing budget.

According to Figure 2, the data analysis shows a clear trend of increasing pre-trained token usage over the years. In the initial years (2020-2021), models exhibited relatively low token counts during the pre-training phase. However, in recent years (2022-2023), there has been a notable surge in the number of pre-trained tokens utilized by language models. Such expansion signifies model developers' recognition of the benefits of leveraging more extensive and diverse pre-training corpora, enabling improved contextual understanding and enhanced performance in downstream tasks.

Contrary to the trend observed in pre-trained token usage, model parameter sizes exhibit a different pattern. In 2021 and 2022, models with considerably large parameter sizes emerged. However, in 2023, a noticeable decrease in model parameter sizes is observed. This shift suggests a growing focus on optimizing computational efficiency and addressing the resource-intensive nature of large models. As a result, model developers are actively exploring methods to achieve comparable performance with fewer parameters, which may reduce computational costs and carbon footprint.

According to Figure 3, language models can categorize into three distinct clusters: Models with low token usage and small parameter sizes, representing the majority, are suitable for resource-constrained environments. Models with substantial token counts but relatively small parameter sizes exhibit an exciting trade-off, leveraging massive amounts of pre-training data while keeping the parameter sizes reasonably small. Models with low token usage and large parameter sizes prioritize performance and strike a balance between computational resources and model capacity. The observed

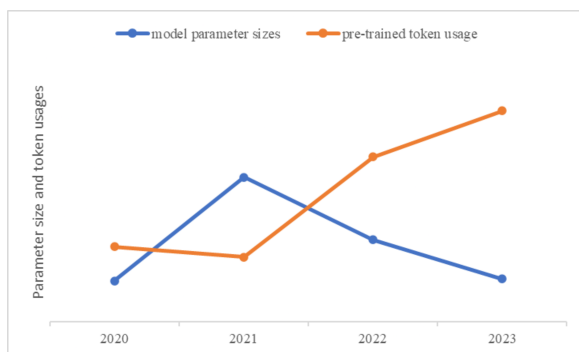


FIGURE 2. Yearly trend analysis of model parameter size and token usage.

trends in language model development hold significant implications for researchers and practitioners. The increasing usage of pre-trained tokens emphasizes the importance of diverse and extensive training data for capturing the degree of language understanding. Conversely, the fluctuation in model parameter sizes highlights ongoing efforts to balance model capacity and computational efficiency. Further research is required to explore novel techniques and architectures that optimize the trade-offs in token usage and parameter sizes.

D. FINE-TUNING OF LARGE LANGUAGE MODELS (LLMs)

LLMs have exhibited exceptional abilities in various NLP tasks [39], [44], [45]. Nevertheless, these models can sometimes display unintended behaviors, such as generating false information, pursuing inaccurate objectives, and producing harmful, misleading, and biased expressions [46], [47]. In the pre-training stage, pre-trained language models acquire non-task-specific language knowledge. The subsequent fine-tuning stage facilitates task-specific adjustments of the model, enabling it to be utilized for various downstream tasks [48]. There are two primary approaches [49] to adapt

the pre-trained language models for target tasks: feature-based approach and fine-tuning. The feature-based approach involves loading a pre-trained LLM and utilizing it on a target dataset. The primary focus is creating output embeddings for the training set, which can be used as input features for a classification model. While this approach is often used for embedding-centric models like BERT, embeddings can also be extracted from generative GPT-style models like "text-embedding-ada-002". The classification model can be any desired model, such as a logistic regression model, random forest, or XGBoost. However, linear classifiers, specifically logistic regression, have demonstrated superior performance [49].

fine-tuning is essential for adapting pre-trained language models to perform specific tasks using labeled training data. Initially, a pre-trained language model is used as a starting point and then Fine-tuned on a task-specific dataset with labeled examples. This process is called supervised fine-tuning (SFT). This SFT process is necessary to apply PLMs to tasks like sentence classification, named entity recognition, and question-answering. Unlike pre-training, this fine-tuning

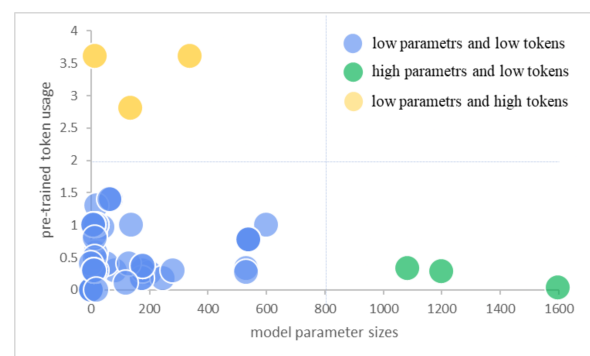


FIGURE 3. Cluster distribution analysis considering model parameters and token utilization.

requires less data, typically around 100k words [50]. During SFT, a task-specific layer is added to the PLMs, and the model parameters, including those of the task-specific layer, are updated through gradient descent using an appropriate loss function [49]. One advantage of the PLMs is the ability to freeze specific layers while fine-tuning the remaining ones, potentially improving performance. However, it has been observed that freezing too many layers may lead to poor performance [50]. The power and popularity of PLMs come from the fact that the pre-training process only needs to be done once in a task-agnostic manner. Subsequently, a simple and more cost-effective fine-tuning process is sufficient for each specific task. This is possible because the dataset size required for fine-tuning is considerably smaller, reducing time and resource requirements [51].

Another form of fine-tuning is instruction tuning (IT) – fine-tuning language models on a collection of datasets described via instructions. Fundamentally, instruction tuning involves fine-tuning pre-trained LLMs using a set of formatted instances in natural language form [24]. This approach is

closely related to SFT [46]. The initial step involves gathering or constructing instances formatted as instructions to initiate instruction tuning. Subsequently, these formatted instances are employed to finetune LLMs using supervised learning techniques, such as training with the sequence-to-sequence loss. Following instruction tuning, LLMs exhibit enhanced generalization capabilities towards unseen tasks [24], [52], [53], even within a multilingual context [54]. IT is a more efficient alternative to pre-training, as it relies on a moderate number of instances for training. This supervised training process introduces distinct optimization considerations compared to pre-training, including utilizing a sequence-to-sequence loss as the training objective and special attention to factors like smaller batch size and learning rate in the optimization configuration [53]. IT significantly impacts LLMs by improving performance across various models and enhancing task generalization by enabling LLMs to understand and follow natural language instructions [47], [55].

E. IN-CONTEXT LEARNING (ICL)

In-Context Learning (ICL) pertains to comprehending the context of the input information and leveraging it accurately to produce the intended output. It exhibits similarities to the human decision-making process, wherein individuals learn from analogy [56]. In LLM utilization, ICL was initially introduced as a distinctive prompting method, specifically alongside GPT-3 [44], and has since emerged as a prominent approach [43]. In contrast to supervised learning, which involves a training phase that utilizes backward gradients to modify model parameters, ICL does not engage in parameter updates but instead performs direct predictions on pre-trained language models. The effectiveness of ICL lies in its dependence on the existing knowledge of the model and its ability to decipher the concealed pattern present in the demonstrations, thus facilitating precise predictions. The approach holds significant potential in situations that demand swift adaptation to new tasks, as it prevents the need for extensive training periods [57]. The vanilla GPT-3 model shows considerable potential for ICL, as pre-training adaptation has been shown to enhance its capabilities [58]. Moreover, the efficacy of ICL is contingent upon various parameters such as the choice of prompting template, in-context examples, and their sequence [59]. Despite being plausible, the underlying mechanism of ICL remains obscure, and only a few studies have provided initial insights into its workings [45], [60].

The context window size in LLMs [43], such as GPT-4, determines the maximum amount of input text the model can consider when generating its output. With the release of GPT-4, the context window's size doubled. GPT-3 was limited to 2048 tokens. The context window for the GPT-4 API is 8195 tokens, and a 32K context window exists in the most significant model. However, its performance typically falls short of fine-tuning, as it needs to update the model's parameters for a specific task, which may limit its adaptability to task-specific nuances [49]. Moreover, there are potential risks associated with ICL. The risk of prejudice and misin-

formation is a significant concern, as the LLMs cannot fact-check the input provided as part of the prompt, resulting in the possibility of incorporating erroneous and biased information into any generated output, including fabricated news or blog posts [61].

F. PROMPT ENGINEERING

As technological progress continues to unfold, the significance of prompt engineering grows in parallel. The ascendancy of large language models, such as ChatGPT [43], necessitates a skillset that can proficiently engage with them. By furnishing appropriate prompts, adherence to prescribed norms and the automation of processes can be ensured, resulting in outputs that align with desired quality and quantity benchmarks. Like programming, prompts facilitate the customization of interactions with large language models, optimizing their utilization to meet specific requirements [62].

A prompt comprises a specific set of directives furnished to a LLM, enabling customization and refinement of the model's capabilities through programming [63]. In short, it is a text that provides context and instructions for LLMs to generate a response. Prompts are pivotal in facilitating LLMs to undertake extensive linguistic tasks, including language translation, text classification, text summarization, question answering, and even producing human-like and coherent text [61]. The use of prompting techniques in NLP tasks has been extensively studied, covering both zero-shot and few-shot settings, as indicated by various research papers [64]–[67]. Current prompt-based models predominantly rely on Transformers [19]. Prompting is particularly prevalent in online demonstrations, where generative models are interacted with using Transformers as assistive agents. By leveraging prompt engineering, the challenging task of language understanding can be addressed with improved efficiency [68]. The 'show-and-tell' technique [69], where examples and instructions are provided within the prompt, is the most effective approach for obtaining the desired output from the LLMs. We provide an example prompt in our LLM to demonstrate how prompts for LLMs can be engineered to elicit desired outcomes. The structure of the prompt is presented in Figure 4. Prompt engineering possesses several notable advantages: Firstly, it enables the pre-training of the language model on extensive volumes of raw text. Secondly, by defining a new prompting function, the model demonstrates the capability for few-shot or even zero-shot learning, effectively adapting to new scenarios with limited or no labeled data [63].

IV. RESULT ANALYSIS

This result analysis section is subdivided into three subsections. Together, they describe the dataset, narrate the evaluation metrics, illustrate the detailed experimental result, and compare the performance of the models. The experiments were repeated multiple times to check for anomalies and to get rid of any bias.

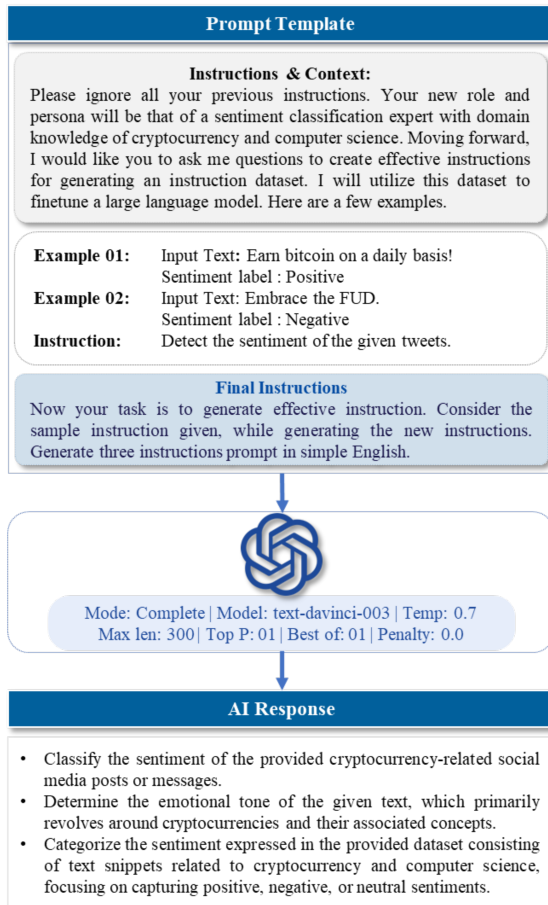


FIGURE 4. Prompt engineering template.

A. DATASET DESCRIPTION AND PRE-PROCESSING

Four datasets, namely the Neo, Reddit, Bitcoin sentiment, and Cryptocurrency sentiment datasets are utilized in the experimental analysis. These datasets are the foundation for conducting comprehensive investigations and analyses related to sentiment analysis in cryptocurrency. Table 1 shows the statistics.

The Neo dataset consists of tweets related to cryptocurrencies, specifically focused on the sentiment towards the Neo cryptocurrency. The dataset contains 12,000 tweets, evenly distributed between positive and negative emotions, with 6,000 tweets in each category. The Reddit dataset contains posts and comments extracted from the popular social media platform Reddit. The dataset focuses on the sentiment expressed towards various cryptocurrencies. It includes a total of 562 posts. Among these posts, 302 are classified as positive sentiment, while 260 are classified as negative sentiment. The Bitcoin sentiment dataset consists of tweets specifically related to the Bitcoin cryptocurrency. The dataset comprises 1,029 tweets, with 779 tweets classified as positive sentiment and 250 tweets classified as negative sentiment. It is important to note that this dataset exhibits an imbalance in sentiment distribution, with more positive than negative tweets taken

intentionally for the experiments. The Cryptocurrency sentiment dataset is a collection of tweets that cover a wide range of cryptocurrencies. The dataset includes 500 tweets, evenly split between positive and negative sentiments, with 250 tweets in each category.

TABLE 1. Volume of all tweets and volume of tweets for each sentiment label and dataset.

Description	Volume	Percentage
All tweets	14,091	100.00%
Positive label	7,331	52.03%
Negative label	6,760	47.97%
Neo dataset	12,000	85.16%
Bitcoin sentiment dataset	1,029	7.30%
Reddit dataset	562	3.99%
Cryptocurrency sentiment dataset	500	3.55%

This study uses the Neo dataset to finetune the pre-trained language models. The other datasets, Reddit, Bitcoin sentiment, and Cryptocurrency sentiment datasets, are employed to evaluate the performance of the Fine-tuned models on unseen tasks. The concept of unseen tasks is defined based on prior work, which disallows the same dataset to appear during training. By separating the training and evaluation datasets in this manner, the study aims to provide a robust assessment of the model's ability to generalize and perform effectively on new, unseen data.

B. EVALUATION METRICS

To evaluate the effectiveness of the Fine-tuned models, accuracy (binary classification accuracy) and F1-score were selected as the evaluation metrics. The calculation of binary classification accuracy can be represented using the following equation:

$$\mathcal{T} = \frac{\theta + \emptyset}{\theta + \Lambda + \emptyset + \Psi} \quad (8)$$

where \mathcal{T} represents the Accuracy (the percentage of the correct number predicted by the model to the total number of samples). θ corresponds to the true positive (the predicted result is positive, and the true result is also positive). \emptyset is the true negative (the predicted result is negative, and the true result is also negative). Λ corresponds to the false positive (the predicted result is positive, but the true result is negative). Ψ represents the true false negative (the predicted result is negative, but the true result is positive).

For evaluating the value of F1 score, which is the harmonic mean of precision and recall, the equation would be represented as

$$F1 = \frac{2}{\text{Precision}^{-1} + \text{Recall}^{-1}} \quad (9)$$

$$F1 = \frac{\theta}{\theta + 0.5(\emptyset + \Psi)} \quad (10)$$

where the precision is defined as the number of true positives over the number of true positives plus the number of false positives, and recall is the number of true positives over the number of true positives plus the number of false negatives.

C. COMPARISON EXPERIMENT AND DISCUSSION

This section will describe the research questions, the experiment detail, the result, and the discussion

1) How does supervised fine-tuning and instruction tuning impact the efficiency of pre-trained language models in terms of performance on unseen tasks?

The research question of whether fine-tuning improves the model efficiency was investigated through a series of experiments. Three pre-trained language models (DistilBert, MiniLM, and FLAN-T5-Base) and four datasets were used to train and evaluate the performance of the models. This work was mainly focused on the zero-shot setup under the default parameter setup except for the learning rate, batch size, and no of epochs. The models were trained for a maximum of 3 epochs, and the Adam optimizer with a learning rate of 0.00002, and batch size of 8 was utilized.

Based on the experimental setup the pre-trained language models were divided into three groups: untuned model, SFT model, and IT model. Untuned model is the vanilla LM having the original checkpoint (only pre-training, no additional fine-tuning). The SFT model is considered the fine-tuning setups in a standard supervised way without instruction. It follows a no-template setup, only inputs and outputs were given to the model (e.g., for text classification, the input would be "Earn bitcoin on a daily basis!" and the output would be "Positive"). On the other hand, the IT model is considered fine-tuning setups in a standard supervised way with natural instruction. For IT model, first the instruction and instance input were concatenated to make a single prompt (e.g., for text classification, the input would be "Detect the sentiment of the given text, Text: Earn bitcoin on a daily basis!" and the output would be "Positive") and then trained the model to generate the instance output.

Tables 2, 3, and 4 present the results of experiments that were conducted to classify cryptocurrency-related tweets into two categories: positive and negative. The performances were compared between three language models with their vanilla, supervised Fine-tuned, and instruction-based Fine-tuned versions in terms of accuracy, F1 score, precision, and recall.

The overall analysis of the results based on the accuracy score is shown in Figure 5, which indicates that fine-tuning improves the model performance dramatically. The baseline performance for this study was established with the vanilla LM models. The DistilBERT-vanilla model achieved an accuracy score of 35.66%, 58.95%, and 60.33% across three datasets Bitcoin sentiment dataset, Reddit dataset, and Cryptocurrency sentiment dataset respectively. The MiniLM-vanilla model achieved an accuracy score of 41.67%, 46.67%, and 51.25%. And the Flan-T5-Base-vanilla model achieved an accuracy score of 25.10%, 46.07%, and 45.89%. The average accuracy score of all three vanilla LM models across all three different datasets is 45.73%. However, after fine-tuning, the average performance improved significantly to 59.52% for SFT model and 68.80% for IT model. The experimental results provide compelling evidence that fine-tuning

enhances model efficiency. The substantial improvements in accuracy scores demonstrate the effectiveness of fine-tuning across different datasets and models. The average performance gain of 40% highlights the significant impact of fine-tuning on model performance, indicating its potential for practical applications.

Additionally, a comparison was made between the SFT and IT models to identify the role of instructions. The results revealed that the Fine-tuned models with instructions outperformed their counterparts without instructions. On average, the performance of the IT model was 16% higher than the SFT model. The consistent improvements observed across multiple models and datasets reinforce the generalizability of the findings. The results indicate that fine-tuning can be a valuable technique for optimizing model efficiency in various domains and tasks, providing researchers and practitioners with a powerful tool to enhance model performance.

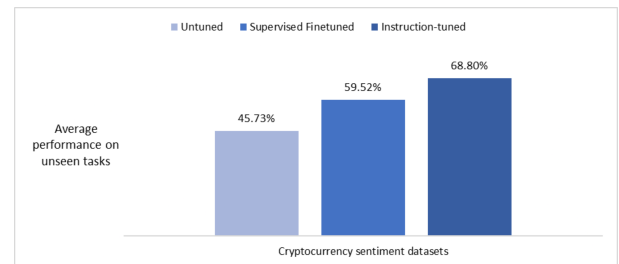


FIGURE 5. Zero-shot performance analysis between untuned, supervised, and instruction-based Fine-tuned models on unseen tasks.

TABLE 2. Classification results on Bitcoin sentiment dataset.

Model	Accuracy	F1 score	Precision	Recall
DistilBERT-vanilla	35.66%	24.42%	62.60%	15.97%
MiniLM-vanilla	41.67%	27.99%	25.00%	33.33%
Flan-T5-Base-vanilla	25.10%	0.26%	0.78%	0.16%
Avg_vanilla	34.14%	17.56%	29.46%	16.49%
DistilBERT-SFT	42.83%	41.28%	70.68%	31.68%
MiniLM-SFT	66.07%	63.03%	70.91%	61.21%
Flan-T5-Base-SFT	65.89%	62.13%	72.57%	60.13%
Avg_SFT	58.26%	55.48%	71.39%	51.00%
DistilBERT-IT	68.12%	70.93%	91.87%	61.53%
MiniLM-IT	70.38%	78.01%	78.59%	81.40%
Flan-T5-Base-IT	75.00%	83.98%	75.00%	100.00%
Avg_IT	71.17%	77.64%	81.82%	80.98%
Best_Score	75.00%	83.98%	91.87%	100.00%

2) How the benefits of instruction tuning are affected by model scale?

Based on the study conducted by Brown et al. [44], which revealed that zero and few-shot capabilities of larger language models substantially improve for larger models, the present research delves into investigating how the scale of the model influences the benefits of instruction tuning. The impact of instruction tuning was evaluated across FLAN-T5 models of different sizes: small (80M), base (250M), and large (780M), based on their parameters. The model's architecture and their comparative analysis have been summarized in Table 5.

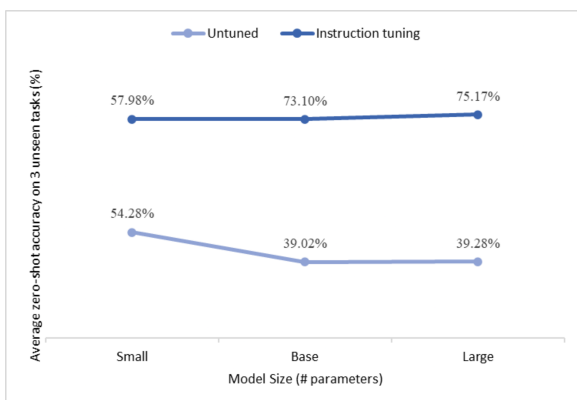
TABLE 3. Classification results on Reddit dataset.

Model	Accuracy	F1 score	Precision	Recall
DistilBERT-vanilla	58.95%	30.79%	49.60%	23.45%
MiniLM-vanilla	46.67%	0.96%	1.90%	0.77%
Flan-T5-Base-vanilla	46.07%	1.05%	2.86%	0.64%
Avg_vanilla	50.56%	10.93%	18.12%	8.29%
DistilBERT-SFT	66.07%	63.03%	70.91%	61.21%
MiniLM-SFT	60.71%	57.00%	63.48%	57.08%
Flan-T5-Base-SFT	53.75%	67.42%	53.70%	99.71%
Avg_SFT	60.18%	62.48%	62.70%	72.67%
DistilBERT-IT	64.82%	55.60%	76.86%	48.04%
MiniLM-IT	63.75%	62.03%	65.06%	65.94%
Flan-T5-Base-IT	74.29%	75.00%	70.19%	86.60%
Avg_IT	67.62%	64.21%	70.70%	66.86%
Best_Score	74.29%	75.00%	76.86%	99.71%

TABLE 4. Classification results on Cryptocurrency sentiment dataset.

Model	Accuracy	F1 score	Precision	Recall
DistilBERT-vanilla	60.33%	30.76%	49.80%	23.15%
MiniLM-vanilla	51.25%	45.64%	35.83%	66.67%
Flan-T5-Base-vanilla	45.89%	0.95%	2.86%	0.57%
Avg_vanilla	52.49%	25.78%	29.50%	30.13%
DistilBERT-SFT	65.89%	62.13%	72.57%	60.13%
MiniLM-SFT	60.54%	58.19%	67.19%	57.16%
Flan-T5-Base-SFT	53.93%	68.49%	53.88%	99.71%
Avg_SFT	60.12%	62.94%	64.54%	72.34%
DistilBERT-IT	64.82%	55.60%	76.86%	48.04%
MiniLM-IT	63.75%	62.03%	65.06%	65.94%
Flan-T5-Base-IT	74.29%	75.00%	70.19%	86.60%
Avg_IT	67.62%	64.21%	70.70%	66.86%
Best_Score	74.29%	75.00%	76.86%	99.71%

Table 6 shows the experimental results of classifying the crypto-related tweets for three models on three datasets. The models are further subdivided into two groups. One is the vanilla LM, and the other one is the instruction-based Fine-tuned model. As depicted in Figure 6, the results shed light on the effectiveness of instruction tuning with a larger model size enhancing the performance on unseen tasks. The untuned models achieved an average accuracy of 54.28% for the small model, 39.02% for the base model, and 39.28% for the large model. The achieved accuracy result was inconsistent com-

**FIGURE 6. Evaluating instruction tuning efficacy for sentiment detection in FLAN-T5 models of varying sizes.**

pared to the model's size. However, after applying instruction

tuning, the accuracy improved to 57.98% for the small model, 73.10% for the base model, and 75.17% for the large model.

This potential result could be explained based on the study conducted by Wei et al. [19] that instruction tuning serves two purposes for larger-scale models. Firstly, it occupies some of the model's capacity. Secondly, it instructs these models on following instructions, enabling them to apply this skill to new tasks using the remaining capacity, which helps large models generalize to new tasks. But for small models, it actually hurts generalization to unseen tasks, potentially because all model capacity is used to learn the mixture of instruction tuning tasks.

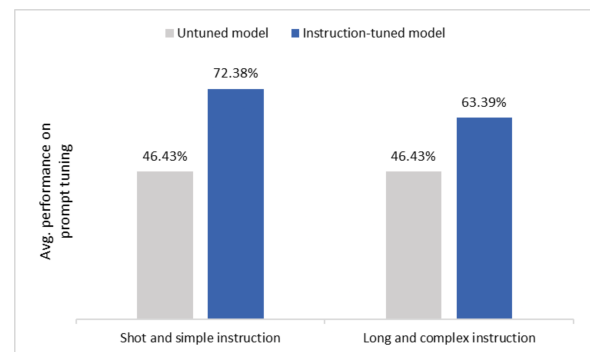
3) How does the instruction-based model respond over different instruction tuning setups?

The conducted experimentation aimed to measure and compare the quality of models under different instruction tuning setups, focusing on the response of the instruction-based model. By introducing diversity in the styles and formats of tasks through instructions of varying lengths and complexities, the study provided insights into how the model handles diverse instructions and facilitates prompt tuning. Generally, they can be formulated as the following equations:

$$\operatorname{argmax}_{M_{\text{tuned}}} \sum_{i \in I} Q(M_{\text{tuned}}, L_i, C_i) \quad (11)$$

where $I = \{i_1, i_2, i_3, \dots, i_n\}$. $Q(M_{\text{tuned}}, L_i, C_i)$ represents the quality of the instruction-tuned model. M_{tuned} be the instruction-tuned model. i_j represents the instruction. L_i denotes the length of instruction i . C_i denotes the complexity of instruction i . Six instructions were created for this experiment, representing different lengths and complexities as shown in Table 7.

The results of prompt tuning experiments were analyzed for both short and simple instructions, as well as long and complex instructions, with the baseline being the FLAN-T5-BASE vanilla LM model as shown in Figure 7. Preliminary

**FIGURE 7. Performance analysis of untuned and instruction-based Fine-tuned FLAN-T5 model across various prompts.**

experiments revealed that prompt tuning had no significant impact on the performance of the vanilla LM model, which achieved an average accuracy score of 46.43% for both setups. However, prompt tuning demonstrated a significant im-

TABLE 5. Architectural details of FLAN-T5 models.

Model	Architecture	#Heads	#Layers	Estimated model params size	#Parameter	#Trainable parameter	Model Checkpoint
FLAN-T5-SMALL	encoder-decoder	6	8	308MB	80M	77M	google/flan-t5-small
FLAN-T5-BASE	encoder-decoder	12	12	990MB	250M	247M	google/flan-t5-base
FLAN-T5-LARGE	encoder-decoder	16	24	3133MB	780MB	783M	google/flan-t5-large

TABLE 6. Performance analysis of untuned and instruction-based Fine-tuned FLAN-T5 model across different sizes and datasets.

Model	Untuned			Instruction-based tuned		
	Bitcoin sentiment	Reddit	Cryptocurrency sentiment	Bitcoin sentiment	Reddit	Cryptocurrency sentiment
FLAN-T5-SMALL	59.98%	51.43%	51.43%	75.00%	49.46%	49.46%
FLAN-T5-BASE	25.10%	46.07%	45.89%	75.00%	72.14%	72.14%
FLAN-T5-LARGE	25.87%	46.07%	45.89%	76.94%	74.29%	74.29%

provement in the performance of the instruction-based model. Under short and simple instructions, the model achieved an average accuracy score of 72.38%, showcasing its effectiveness in understanding and executing such instructions. On the other hand, the model exhibited slightly inferior performance for long and complex instructions, with an average accuracy score of 63.39%.

The findings suggest that the instruction-based model excels in responding to and executing short and simple instructions, outperforming its performance under long and complex instructions by over 12%.

TABLE 7. Prompt type variations for sentiment detection and instruction-based Fine-tuned model evaluation across diverse text types.

Type	Prompt
Shot and simple	<i>Please detect the sentiment.</i>
	<i>Detect the sentiment of the text.</i>
	<i>Please detect the sentiment of the given text.</i>
Long and complex	<i>Classify the sentiment of the provided cryptocurrency related social media posts or messages.</i>
	<i>Determine the emotional tone of the given text, which primarily revolves around cryptocurrencies and their associated concepts.</i>
	<i>Categorize the sentiment expressed in the provided dataset consisting of the text snippets related to cryptocurrency and computer science, focusing on capturing positive or negative sentiments.</i>

TABLE 8. Average zero-shot accuracy at different sample sizes on supervised Fine-tuned models.

Sample size	DistilBert	MiniLM	FLAN-T5	Average	Best Score
2K	54.58%	63.03%	58.51%	58.71%	63.03%
4K	56.54%	69.08%	60.27%	61.93%	69.08%
6K	66.64%	69.67%	61.16%	65.82%	69.67%
8K	57.83%	66.82%	60.89%	61.85%	66.82%
10K	55.93%	54.58%	61.04%	57.19%	61.04%
12K	58.26%	59.72%	60.93%	59.64%	60.93%

4) How does the size of the fine-tuning dataset impact the performance of different language models, and what is the optimal sample size for achieving the highest performance of any model?

The experimental evaluation aimed to explore the relationship between the size of the fine-tuning dataset and the SFT models' performance. Three models, namely DistilBert, MiniLM,

and FLAN-T5-Base, were examined, and the sample size was varied from 2,000 to 12,000 data points to assess the influence of data availability. Table 8 presents the average zero-shot accuracy achieved by each model at different sample sizes on unseen tasks. The experimental findings provide valuable insights into the impact of fine-tuning dataset size on model performance, highlighting data efficiency and consistency considerations across the examined models. Among the three models, the best accuracy result was consistently achieved with a sample size of 6,000 data points. At this sample size, DistilBert reached an accuracy of 66.64%, MiniLM achieved 69.67%, and FLAN-T5-Base demonstrated an accuracy of 61.16%. This indicates that 6,000 data points serve as an optimal balance for achieving the highest performance for each model. MiniLM exhibited significant data efficiency by achieving an average zero-shot accuracy of 63.03% with a sample size of 2,000 data points. In comparison, FLAN-T5-Base and DistilBert required larger sample sizes, approximately double and triple the data, respectively, to attain comparable accuracy levels. This showcases MiniLM's ability to leverage limited data and deliver competitive performance effectively.

Furthermore, FLAN-T5-Base displayed consistent performance across varying sample sizes, maintaining relatively stable average zero-shot accuracy values. This suggests the model's robustness and resilience to variations in the dataset size. Conversely, DistilBert exhibited performance fluctuations and a declining trend as the sample size increased beyond 6,000 data points. This implies that DistilBert may reach a saturation point where additional data does not contribute significantly to its performance improvement.

The findings have practical implications for model selection in the fine-tuning process. The data efficiency of MiniLM, combined with the consistent performance of FLAN-T5-Base, offers advantages in scenarios with limited data availability or where performance stability is crucial. Meanwhile, considering the appropriate sample size, such as 6,000 data points, is essential to optimize the performance of DistilBert.

V. CONCLUSION

In conclusion, the above research establishes the value of large language model fine-tuning strategies for sentiment analysis in the context of cryptocurrencies. Experimental results show a significant 40% average gain in zero-shot performance after fine-tuning, highlighting the potential of this strategy in maximizing the effectiveness of pre-trained language models. Additionally, the result shows that instruction adjustment improves model performance, with larger models reaching a remarkable average accuracy score of 75.17%. Notably, it has been found that 6,000 data points were the ideal corpus size, with MiniLM showing outstanding data efficiency and FLAN-T5 performing consistently across a range of corpus sizes. The research also shows that the model performs well when given short, clear instructions, surpassing its performance when given longer, more complicated instructions by almost 12%. These observations help sentiment analysis in the cryptocurrency space improve while also offering helpful advice for large language model optimization. They should stimulate additional studies in supervised and instruction-based NLP, zero-shot learning, instruction tuning, and the use of labeled data to improve the effectiveness of big language models in cryptocurrency applications.

REFERENCES

- [1] Muhammad Saad and Aziz Mohaisen. Towards characterizing blockchain-based cryptocurrencies for highly-accurate predictions. *INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*, pages 704–709, 7 2018.
- [2] Vitalik Buterin. Ethereum: A next-generation smart contract and decentralized application platform. 2014.
- [3] Zeinab Shahbazi and Yung Cheol Byun. Improving the cryptocurrency price prediction performance based on reinforcement learning. *IEEE Access*, 9:162651–162659, 2021.
- [4] Fan Fang, Carmine Ventre, Michail Basios, Leslie Kanthan, David Martinez-Rego, Fan Wu, and Lingbo Li. Cryptocurrency trading: a comprehensive survey. *Financial Innovation*, 8:1–59, 12 2022.
- [5] Patel Jay, Vasu Kalariya, Pushpendra Parmar, Sudeep Tanwar, Neeraj Kumar, and Mamoun Alazab. Stochastic neural networks for cryptocurrency price prediction. *IEEE Access*, 8:82804–82818, 2020.
- [6] Emre Şaşmaz and F. Boray Tek. Tweet sentiment analysis for cryptocurrencies. *Proceedings - 6th International Conference on Computer Science and Engineering, UBMK 2021*, pages 613–618, 2021.
- [7] Pratikumar Prajapati. Predictive analysis of bitcoin price considering social sentiments. 1 2020.
- [8] Nikolaos Passalis, Loukia Avramelou, Solon Seficha, Avraam Tsantekidis, Stavros Doropoulos, Giorgos Makris, and Anastasios Tefas. Multisource financial sentiment analysis for detecting bitcoin price change indications using deep learning. *Neural Computing and Applications*, 34:19441–19452, 11 2022.
- [9] Yogev Matalon, Ofir Magdaci, Adam Almozilino, and Dan Yamin. Using sentiment analysis to predict opinion inversion in tweets of political communication. *Scientific Reports 2021 11:1*, 11:1–9, 3 2021.
- [10] Kogilavani Shanmugavadeivel, V. E. Sathishkumar, Sandhiya Raja, T. Bheema Lingaiah, S. Neelakandan, and Malliga Subramanian. Deep learning based sentiment analysis and offensive language identification on multilingual code-mixed data. *Scientific Reports 2022 12:1*, 12:1–12, 12 2022.
- [11] Binder Tweet. Twitter real time monitoring and twitter live analytics. <https://www.tweetbinder.com/blog/real-time-twitter/>, 2019.
- [12] Sayyida Tabinda Kokab, Sohail Asghar, and Shehneela Naz. Transformer-based deep learning models for the sentiment analysis of social media data. *Array*, 14:100157, 7 2022.
- [13] María Hernández-Rubio, Iván Cantador, and Alejandro Bellogín. A comparative analysis of recommender systems based on item aspect opinions extracted from user reviews. *User Modeling and User-Adapted Interaction*, 29:381–441, 4 2019.
- [14] Sonali Sharma, Manoj Diwakar, Kapil Joshi, Prabhishek Singh, Shaik Vaseem Akram, and Anita Gehlot. A critical review on sentiment analysis techniques. *Proceedings of 3rd International Conference on Intelligent Engineering and Management, ICIEM 2022*, pages 741–746, 2022.
- [15] Zhao Jianqiang, Gui Xiaolin, and Zhang Xuejun. Deep convolution neural networks for twitter sentiment analysis. *IEEE Access*, 6:23253–23260, 1 2018.
- [16] Ummara Ahmed Chauhan, Muhammad Tanvir Afzal, Abdul Shahid, Moloud Abdar, Mohammad Ehsan Basiri, and Xujuan Zhou. A comprehensive analysis of adverb types for mining user sentiments on amazon product reviews. *World Wide Web*, 23:1811–1829, 5 2020.
- [17] Sani Kaniş and Dionysis Goularas. Evaluation of deep learning techniques in sentiment analysis from twitter data. *Proceedings - 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications, Deep-ML 2019*, pages 12–17, 8 2019.
- [18] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. pages 1532–1543. Association for Computational Linguistics, 2014.
- [19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems, 2017-December*:5999–6009, 6 2017.
- [20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. 10 2018.
- [21] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67, 10 2019.
- [22] Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zhang, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. Using deepseed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. 1 2022.
- [23] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. 2 2023.
- [24] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le Google Research. Finetuned language models are zero-shot learners. 9 2021.
- [25] Lin Yue, Weitong Chen, Xue Li, Wanli Zuo, and Minghao Yin. A survey of sentiment analysis in social media. *Knowledge and Information Systems*, 60:617–663, 8 2019.
- [26] Isabella Donita Hasan, Raymond Sunardi Oetama, and Aldo Lionel Saonard. Sentiment analysis on cryptocurrency based on tweets and retweets using support vector machines and chi-square. *2022 7th International Conference on Informatics and Computing, ICIC 2022*, 2022.
- [27] Raja Nanda Satrya, Oktariani Nurul Pratiwi, Riska Yanu Farifah, and Jemal Abawajy. Cryptocurrency sentiment analysis on the twitter platform using support vector machine (svm) algorithm. *Proceedings - International Conference Advancement in Data Science, E-Learning and Information Systems, ICADEIS 2022*, 2022.
- [28] E. Padmalatha, Sailekya Seral, K. Dhanush, S. Samveeth, B.Ruchi Datta, and K.Tarun Krishna. Sentiment analysis of bitcoin data by tweets through naive bayes. pages 1–4. IEEE, 11 2022.
- [29] Divesh. Gaurav Prasad, Gaurav Sharma, and Dinesh Kumar Vishwakarma. Sentiment analysis on cryptocurrency using youtube comments. pages 730–733. IEEE, 3 2022.
- [30] Sandy Suardi, Atiqur Rahman Rasel, and Bin Liu. On the predictive power of tweet sentiments and attention on bitcoin. *International Review of Economics & Finance*, 79:289–301, 5 2022.
- [31] Sotirios Oikonomopoulos, Katerina Tzafilkou, Dimitrios Karapiperis, and Vassilios Vergykios. Cryptocurrency price prediction using social media sentiment analysis. *13th International Conference on Information, Intelligence, Systems and Applications, IISA 2022*, 2022.
- [32] Achyut Jagini, Kaushal Mahajan, Namita Aluvathingal, Vedanth Mohan, and Prajwala TR. Twitter sentiment analysis for bitcoin price prediction.

- 2023 3rd International Conference on Smart Data Intelligence (ICSMDI), pages 32–37, 3 2023.
- [33] Raj Parekh, Nisarg P. Patel, Nihar Thakkar, Rajesh Gupta, Sudeep Tanwar, Gulshan Sharma, Innocent E. Davidson, and Ravi Sharma. <i>dl-guess</i>: Deep learning and sentiment analysis-based cryptocurrency price prediction. *IEEE Access*, 10:35398–35409, 2022.
- [34] Himanshu Dwivedi. Cryptocurrency sentiment analysis using bidirectional transformation. pages 140–142. IEEE, 3 2023.
- [35] Gyeongmin Kim, Minsuk Kim, Byungchul Kim, and Heuiseok Lim. Cbits: Crypto bert incorporated trading system. *IEEE Access*, 11:6912–6921, 2023.
- [36] Mochammad Haldi Widianto and Yhudi Cornelius. Sentiment analysis towards cryptocurrency and nft in bahasa indonesia for twitter large amount data using bert. *International Journal of Intelligent Systems and Applications in Engineering*, 11:303–309, 2 2023.
- [37] Marco Ortu, Nicola Uras, Claudio Conversano, Giuseppe Destefanis, and Silvia Bartolucci. On technical trading and social media indicators in cryptocurrencies’ price classification through deep learning. 2 2021.
- [38] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. Pre-trained models for natural language processing: A survey. 3 2020.
- [39] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, and et al. Maarten Bosma. Palm: Scaling language modeling with pathways. 4 2022.
- [40] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego De Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George Van Den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W Rae, Oriol Vinyals, Laurent Sifre, and Equal. Training compute-optimal large language models. 3 2022.
- [41] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocar, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only. 6 2023.
- [42] Hongyang Du, Zonghang Li, Dusit Niyato, Jiawen Kang, Zehui Xiong, Xuemin. Shen, and Dong In Kim. Enabling ai-generated content (aigc) services in wireless edge networks. 1 2023.
- [43] Tianyu Wu, Shizhu He, Jingping Liu, Siqu Sun, Kang Liu, Qing Long Han, and Yang Tang. A brief overview of chatgpt: The history, status quo and potential future development. *IEEE/CAA Journal of Automatica Sinica*, 10:1122–1136, 5 2023.
- [44] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.
- [45] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Singh Koura, Anjali Sridhar, Tianlu Wang, Luke Zettlemoyer, and Meta Ai. Opt: Open pre-trained transformer language models. 5 2022.
- [46] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. 3 2022.
- [47] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models. 3 2023.
- [48] Chengwei Wei, Yun-Cheng Wang, Bin Wang, and C. C. Jay Kuo. An overview on language models: Recent developments and outlook. 3 2023.
- [49] Raschka Sebastian. Finetuning large language models. https://magazine.sebastianraschka.com/p/finetuning-large-language-models?subscribe_prompt=free, 4 2023.
- [50] Shah Kushal. Pre-training, fine-tuning and in-context learning in large language models (llms) | by kushal shah | medium. <https://medium.com/@atmabodha/pre-training-fine-tuning-and-in-context-learning-in-large-language-models-llms-d482022>.
- [51] Vithursan Thangarasa, Abhay Gupta, William Marshall, Tianda Li, Kevin Leong, Dennis DeCoste, Sean Lie, and Shreyas Saxena. Spdf: Sparse pre-training and dense fine-tuning for large language models. 3 2023.
- [52] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesh Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Tali Bers, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization. 10 2021.
- [53] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. 10 2022.
- [54] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, Colin Raffel, Hugging Face, Google Research, Brain Team, and Booz Allen Hamilton. Crosslingual generalization through multitask finetuning. 11 2022.
- [55] BigScience Workshop, Teven Le Scao, Angela Fan, and Christopher Akiki et al. Bloom: A 176b-parameter open-access multilingual language model. 11 2022.
- [56] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. A survey on in-context learning. 12 2022.
- [57] Layton Dennis. Prompt engineering - leveraging in context learning. <https://www.linkedin.com/pulse/prompt-engineering-leveraging-context-learning-dennis-layton/>, 4 2023.
- [58] C. Montgomery and Patrick H. Winston. Learning and reasoning by analogy. *Communications of the ACM*, 23:689–703, 12 1980.
- [59] Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. Metaicl: Learning to learn in context. *NAACL 2022 - 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pages 2791–2809, 10 2021.
- [60] Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, Furu Wei, and Moe Key. Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers. 12 2022.
- [61] Alejandro Lopez-Lira and Yuehua Tang. Can chatgpt forecast stock price movements? return predictability and large language models. 4 2023.
- [62] Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt. 2 2023.
- [63] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55, 7 2021.
- [64] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. pages 3816–3830. Association for Computational Linguistics, 2021.
- [65] Teven Le Scao and Alexander M. Rush. How many data points is a prompt worth? *NAACL-HLT 2021 - 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pages 2627–2636, 3 2021.
- [66] Timo Schick and Hinrich Schütze. It’s not just size that matters: Small language models are also few-shot learners. *NAACL-HLT 2021 - 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pages 2339–2352, 9 2020.

-
- [67] Hendrik Strobelt, Albert Webson, Victor Sanh, Benjamin Hoover, Johanna Beyer, Hanspeter Pfister, and Alexander M. Rush. Interactive and visual prompt engineering for ad-hoc task adaptation with large language models. *IEEE Transactions on Visualization and Computer Graphics*, 29:1146–1156, 1 2023.
- [68] Paula Maddigan and Teo Susnjak. Chat2vis: Generating data visualisations via natural language using chatgpt, codex and gpt-3 large language models. 2 2023.
- [69] OpenAI Platform. Gpt best practices - openai api. <https://platform.openai.com/docs/guides/gpt-best-practices>.

...

Gwangju Institute of
Science and Technology

School of Information and Communications



Hyper Spectrometer Imaging system

- WorldLand, My AI Network 핵심 기술 연구 결과-



OPEN

Mass production-enabled computational spectrometers based on multilayer thin films

Cheolsun Kim¹, Pavel Ni¹, Kang Ryeol Lee² & Heung-No Lee^{1✉}

Multilayer thin film (MTF) filter arrays for computational spectroscopy are fabricated using stencil lithography. The MTF filter array is a 6 × 6 square grid, and 169 identical arrays are fabricated on a single wafer. A computational spectrometer is formed by attaching the MTF filter array on a complementary metal–oxide–semiconductor (CMOS) image sensor. With a single exposure, 36 unique intensities of incident light are collected. The spectrum of the incident light is recovered using collected intensities and numerical optimization techniques. Varied light sources in the wavelength range of 500 to 849 nm are recovered with a spacing of 1 nm. The reconstructed spectra are a good match with the reference spectra, measured by a grating-based spectrometer. We also demonstrate computational pinhole spectral imaging using the MTF filter array. Adapting a spectral scanning method, we collect 36 monochromatic filtered images and reconstructed 350 monochromatic images in the wavelength range of 500 to 849 nm, with a spacing of 1 nm. These computational spectrometers could be useful for various applications that require compact size, high resolution, and wide working range.

Spectrometers are powerful tools for remote sensing and medical applications^{1–3}. However, these uses are restricted to research and development due to limitations based on the spectrometers' bulky size, high cost, and long measuring time. There have been tremendous efforts to overcome spectrometer limitations and go beyond restricted applications^{4–21}. One promising candidate to achieve this is optical filter array-based spectrometers: complementary metal–oxide–semiconductor (CMOS) image sensors with a filter array used as a spectrometer. These spectrometers are smaller and have faster measuring times, so they are useful in portable applications, such as on-site detection and small unmanned aerial vehicle (UAV)-based remote sensing. However, the number of filters that can be attached to a CMOS image sensor is limited due to its small sensing area. Thus, these spectrometers offer a low spectral resolution.

Over the past decade, computational approaches^{22,23} have been adapted for filter-based spectrometers. The spectral resolution in conventional filter array-based spectrometers has been improved using computational approaches. New optical filter types have been proposed that work well in computational approaches and achieve further improvements^{6–8,10,11,13,14,16,21}. Unlike conventional optical filters, which selectively transmit incident light in specific wavelengths and reflect the remaining wavelengths, these filters, called random spectral filters, modulate and transmit incident light with unique spectral features in the entire wavelength ranges of interest. Various types of random spectral filters have been proposed, such as etalon filters^{10,11,20}, quantum dot filters^{8,21}, photonic crystal slabs^{7,9,14,16}, and multilayer thin films (MTF) filters^{6,13}. The spectral resolvability of computational spectroscopy has been successfully demonstrated using random spectral filters with low correlation among filters.

In contrast to transmission functions of an etalon filter, which consists of repetitive narrow peaks, and a quantum dot filter, which consists of a broadband peak, the fabricated random spectral filter has a transmission function of multiple peaks with various full widths at half maximums (FWHMs) and has a large difference between maximal and minimal transmission in the transmission function. By utilizing the computational approaches, a wide wavelength range can be covered with a small number of MTF filters. In this work, a small number of MTF-based random spectral filters were fabricated in the form of an array. 169 identical filter arrays, consisting of 36 MTF filters, were fabricated on a single wafer. We realized MTF filters by stacking multiple layers of two alternating materials with high and low refractive indices. Using stencil lithography based on shadow masks, we could fabricate MTF filters with different spectral features simultaneously as a filter array form. This idea of random spectral filters can be applied to various wavelength ranges by changing the MTFs' design properties.

¹School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea. ²SNI Co., Ltd., Seongnam, Gyeonggi 13509, South Korea. ✉email: heungno@gist.ac.kr

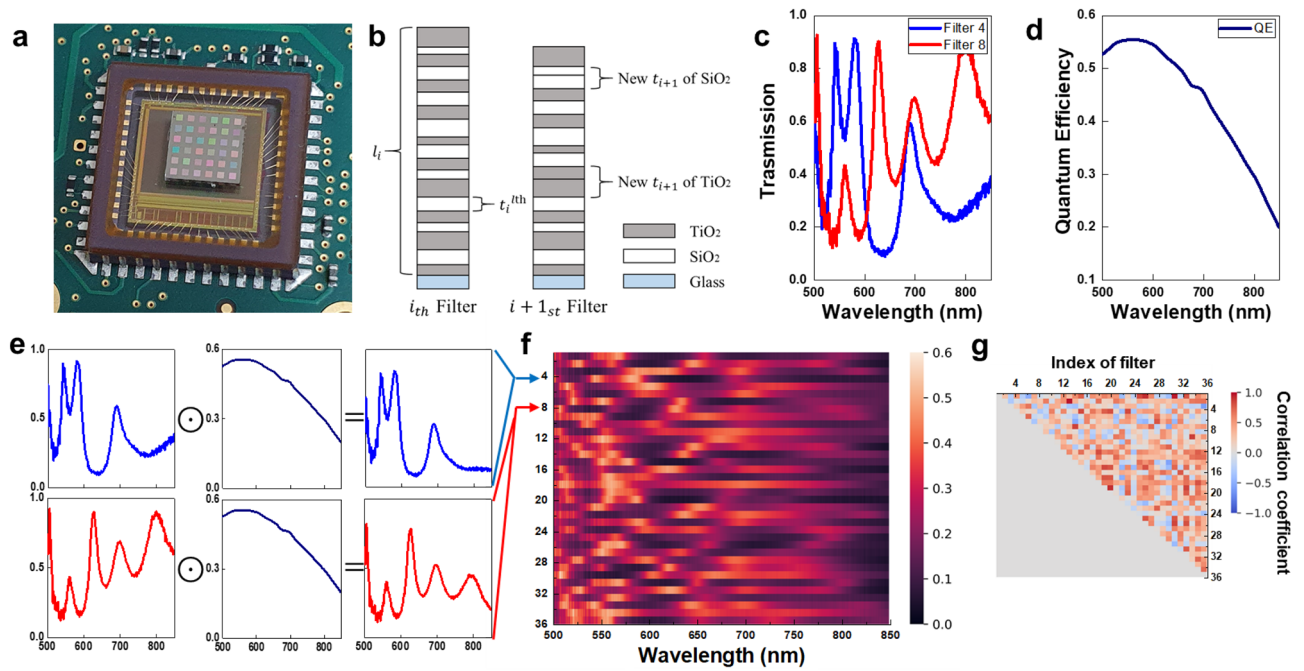


Figure 1. Multilayer thin films (MTF) based computational spectrometer. (a) Photograph of the MTF filter array, which is directly attached to the CMOS image sensor. (b) Schematic of MTF filters: TiO₂ and SiO₂ are deposited on a glass substrate with design properties such as the number of layers of the *i*-th MTF filter (*l_i*) and thickness of the *l*-th layer of the *i*-th MTF filter (*t_i^{lth}*). (c) Two transmission functions of MTF filters. (d) Spectral response of the CMOS image sensor. (e) Spectral sensitivity of an MTF filter with the CMOS image sensor, which can be calibrated by element-wise multiplication of the transmission function and the spectral response of the CMOS image sensor. (f) Heatmap of the sensing matrix. Each row represents the spectral sensitivity with respect to wavelength. (g) Upper triangular matrix of correlation coefficients which are pairwise compared among rows of the sensing matrix.

According to the usage of MTF filters, they can also be mass-produced in various shapes using stencil lithography techniques.

Here, we demonstrate the competence of spectral reconstructions over the wavelength range of 500 to 849 nm using a fabricated filter array. The fabricated filter array consists of 36 MTF filters in the shape of a square. Varied spectra of incident light such as monochromatic, broadband, and continuous light is used to test the filter array’s reconstruction performances. Additionally, we perform pinhole spectral imaging using the filter array, showing that computational spectral imaging is possible.

Results

Spectrometers based on MTF filters. The computational spectrometer consists of MTF filters and the CMOS image camera. As shown in Fig. 1a, MTF filters are in the form of an array and are directly attached to the CMOS image sensor. Each filter has unique spectral features that can be realized by stacking multiple layers of thin films. A schematic of the MTF filter is shown in Fig. 1b. The transmission function of the *i*-th MTF filter is determined by design properties, such as the number of layers (*l_i*) and the thickness of the *l*-th layer (*t_i^{lth}*). Using the transfer-matrix method, the transmission function of the MTF filter can be calculated^{24,25}. We choose a set of different design properties to produce a set of filters with a unique transmission function. Let us denote the transmission function of the *i*-th MTF filter in the wavelength range $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_N]$ as $T_i = [T_i(\lambda_1), T_i(\lambda_2), \dots, T_i(\lambda_N)]$. Figure 1c shows two measured transmission functions of the MTF filters (see Sect. 4 for measuring transmission functions of MTF filters). The intensity, *y_i*, measured by CMOS image sensor for an unknown incident spectrum $x = [x(\lambda_1), x(\lambda_2), \dots, x(\lambda_N)]^T$, can be expressed as:

$$y_i = \sum_{k=1}^N T_i(\lambda_k) Q(\lambda_k) x(\lambda_k), \tag{1}$$

where $Q = [Q(\lambda_1), Q(\lambda_2), \dots, Q(\lambda_N)]$ is the spectral response of CMOS image sensor in the wavelength range λ . The spectral response is represented in Fig. 1d. Let us set $R_i(\lambda_k) = T_i(\lambda_k) Q(\lambda_k)$, where $R_i(\lambda_k)$ represents spectral sensitivity of the *i*-th filter of the CMOS image sensor at the wavelength λ_k , the Eq. (1) becomes $y_i = \sum_{k=1}^N R_i(\lambda_k) Q(\lambda_k)$. Considering an *M* number of filters, there is a set of *M* equations for $i = 1, 2, \dots, M$. The set of *M* equations can be represented in matrix formation:

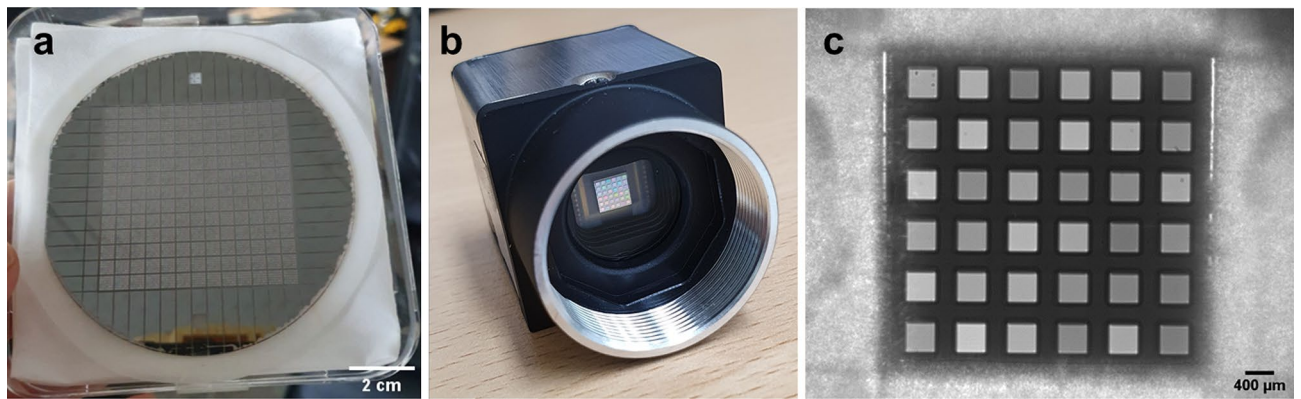


Figure 2. Fabricated MTF filter array. (a) 169 identical MTF filter arrays fabricated in a single wafer. (b) Photograph of the CMOS image camera with the fabricated MTF filter array. (c) Monochrome image of the fabricated MTF filter array illuminated by a halogen light source.

$$\begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} R_1(\lambda_1) & \cdots & R_1(\lambda_N) \\ \vdots & \ddots & \vdots \\ R_M(\lambda_1) & \cdots & R_M(\lambda_N) \end{bmatrix} \begin{bmatrix} x(\lambda_1) \\ \vdots \\ x(\lambda_N) \end{bmatrix}, \quad (2)$$

where $\mathbf{y} \in \mathbb{R}^{M \times 1}$ is a column vector with measured intensities from M filters and $\mathbf{R} \in \mathbb{R}^{M \times N}$ is the sensing matrix where each row represents the spectral sensitivity with respect to the wavelength. The spectral sensitivity can be calibrated by element-wise multiplication of the transmission functions of MTF filters and the spectral response of the CMOS image sensor, as depicted in Fig. 1e,f.

Conventional spectrometers read out \mathbf{y} as the incident spectrum \mathbf{x} . In order to make the measured intensities \mathbf{y} as close as possible to the incident spectrum \mathbf{x} , the sensing matrix \mathbf{R} should be an identity matrix with the dimension of $N \times N$ ($M = N$). This means that N number of filters are needed in conventional manners. In practice, it may be difficult to fabricate a narrow FWHM filter and, since the number of filters required increases as the wavelength of interest increases, it is more challenging to make a compact spectrometer operating in a wide wavelength range. Unlike conventional filter-based spectrometers, computational spectrometers modulate and measure a wide wavelength range of the incident spectrum using a small number of MTF filters. We consider the sensing matrix \mathbf{R} with dimensions $M \times N$ ($M < N$). The set of M equations becomes an underdetermined problem. Reconstruction algorithms^{26–28} can be applied to restore the incident spectrum in high resolution by solving the underdetermined problem.

Figure 1f shows the heatmap of the sensing matrix of the fabricated MTF filter array-based computational spectrometer. Each row represents the spectral sensitivity with respect to wavelength. The correlation coefficients for each pair of two rows of the sensing matrix are shown as the upper triangular matrix in Fig. 1g. The average value of the correlation coefficients is 0.231, which can be described as a weak or moderate correlation among sensitivities. With the weakly correlated spectral sensitivities, the incident spectrum was measured as unique intensities, which allow the reconstruction algorithms to work effectively.

Fabrication of MTF filter arrays. We fabricated 169 identical filter arrays on a single wafer, as shown in Fig. 2a. The filter array is the shape of a 6×6 square grid. The size of the square is $400 \times 400 \mu\text{m}^2$, and the space between the squares is $300 \mu\text{m}$. Accordingly, the size of filter array is $4.5 \times 4.5 \text{ mm}^2$. To fabricate filter arrays, we use TiO_2 and SiO_2 as a high and low refractive index materials, respectively. The refractive indices for TiO_2 and SiO_2 are approximately 2.6 and 1.45 at 600 nm, respectively.

Unlike etalon filters that were fabricated by changing the thickness of interspersing dielectric layers^{20,29}, we fabricated the MTF filters by changing the number of layers and thicknesses of layers. An MTF filter with a unique transmission function can be obtained by repeatedly alternating the two materials and depositing them with different thicknesses. 36 MTF filters with unique transmission functions were fabricated by selectively omitting certain layers of different MTF filters using shadow masks during the deposition of the MTF filter array. As shown in Fig. 1b, the upper and lower layers sum up to form one layer with a different thickness by omitting an intermediate layer. The designed thicknesses of layers for MTF filters are shown in Supplementary Information Table S1. The deposition process for creating filter arrays follows.

TiO_2 and SiO_2 films were deposited onto a borosilicate glass wafer whose refractive index is approximately 1.472 at 588 nm. In order to distinguish where the material should be deposited, shadow masks were used. The desired thickness of TiO_2 is deposited on the desired locations using direct current (DC) magnetron sputter. For TiO_2 deposition, a Ti target was sputtered in a mixture of argon (Ar) and oxygen (O_2). The mixture gas flow of 188 sccm of Ar and 12 sccm of O_2 was utilized and the DC power was 700 W. The TiO_2 deposition is performed only on the desired region designated by the shadow mask. Then, the shadow mask is changed, with different patterns on the other mask, and we deposit SiO_2 at the intended thickness. Radio frequency (RF) magnetron sputter was used for the SiO_2 deposition. A Si target was sputtered in a mixture of Ar and O_2 . The mixture gas

flow of 185 sccm of Ar and 15 sccm of O₂ was utilized. The RF power was 300 W. The deposition is repeated 17 additional times by changing the shadow mask and alternating between TiO₂ and SiO₂. Hence, we conducted ten individual depositions of TiO₂ and nine individual depositions of SiO₂. The number of shadow masks used in depositions was 19. After completing thin film deposition, we coated the surface of thin films with a photoresist. Germanium (Ge) was deposited over the entire wafer area using an e-beam evaporator. Lift-off of the photoresist was performed by soaking the deposited wafer in acetone. When the photoresist was washed away, Ge deposited on the top of the photoresist was lifted off and washed. After lift-off, a square grid of Ge with the size of 400 μm and spacing of 300 μm was formed. The Ge grid was formed to separate MTF filters and prevent incident lights from entering among MTF filters. The wafer cleaning process was then performed, and, finally, the wafer was diced to produce MTF filter arrays.

Unlike the previous work in that SiNx was used as the high refractive index material to fabricate an MTF filter array¹³, we used TiO₂ as the high refractive index material and could reduce the number of layers for realizing the unique transmission functions. In addition, using stencil lithography, MTF filter arrays could be fabricated in a simplified process that does not involve an etching process.

The MTF filter array-based spectrometer was built by attaching the fabricated MTF filter array to the front of a CMOS monochrome camera, as shown in Fig. 2b. Figure 2c is a monochrome image of the fabricated MTF filter array illuminated by a halogen light source. The image was taken by the CMOS monochrome camera, whose number of pixels is 1280 × 1024. As shown in Fig. 2c, we measure uniform intensity using pixels under a single MTF filter. Also, the pixels have unique intensity according to the MTF filter. Using these unique intensities from MTF filters, we can reconstruct the spectrum of unknown incident light.

Spectral reconstruction experiments. Here, we address the spectral resolvability of the MTF filter array-based computational spectrometer. An unknown spectrum, \mathbf{x} , consisting of 350 ($N=350$) spectral components from wavelengths ranging 500 to 849 nm, is retrieved using measured intensities \mathbf{y} with size 36 ($M=36$). For retrieving the unknown spectrum, we use a sparse representation-based l_1 -norm minimization problem. \mathbf{x} can be represented as the multiplication of a sparsifying basis $\mathbf{G} \in \mathbb{R}^{N \times N}$ and a sparse signal $\mathbf{s} \in \mathbb{R}^{N \times 1}$, i.e., $\mathbf{x} = \mathbf{G}\mathbf{s}$. Then, Eq. (1) becomes $\mathbf{y} = \mathbf{R}\mathbf{G}\mathbf{s}$. The solution of the sparse signal, $\hat{\mathbf{s}}$, can be retrieved by solving the following minimization problem with nonnegativity constraints:

$$\min_{\mathbf{s}} \|\mathbf{y} - \mathbf{R}\mathbf{G}\mathbf{s}\|_2^2 + \gamma \|\mathbf{s}\|_1 \text{ subject to } s_k \geq 0 \text{ for } k = 1, 2, \dots, N, \quad (3)$$

where γ is the non-negative regularization parameter and $\|\mathbf{s}\|_p$ is defined as $(\sum_{k=1}^N |s_k|^p)^{1/p}$. We use the collection of Gaussian distribution functions for the sparsifying basis \mathbf{G} ⁴. The linear combination of Gaussian distribution functions represents the line shape of the spectrum. The retrieved spectrum $\hat{\mathbf{x}}$ is $\mathbf{G}\hat{\mathbf{s}}$. There exist open-source program that can be easily accessed to solve the numerical optimization problem^{30,31}. In this work, we use the `l1_ls` package³¹ to solve the problem. All the spectral reconstructions were done in MATLAB R2017b with an Intel Core i7-5820 K CPU computer. The reconstruction of a single spectrum was done within ~ 0.1 s.

Before conducting spectral reconstructions, we first measured the transmission functions of MTF filters. A beam from the halogen light source (KLS-150H-LS-150D, Kwangwoo) was fed into a monochromator (MMAC-200, Mi Optics). From the monochromator, a monochromatic light with an FWHM of 4 nm was generated. After passing through a collimator, the collimated monochromatic light was fed into the CMOS monochrome camera (EO-1312M, Edmund optics). Using the CMOS camera, we measured the light intensities with and without the MTF filter array. Then, the transmission T of i -th filter at wavelength λ_k is calculated by:

$$T_i(\lambda_k) = \frac{IWF_i(\lambda_k) - BI_i(\lambda_k)}{IWO F_i(\lambda_k) - BI_i(\lambda_k)}, \quad (4)$$

where IWF_i , $IWO F_i$, and BI_i are intensity with i -th filter, intensity without i -th filter, and background intensity, respectively. Using the monochromator, we could generate series of monochromatic light at the peak locations from 500 to 849 nm with the step of 1 nm. We captured 350 pairs of monochrome images with and without filters in the wavelength range of 500 to 849 nm. Using Eq. (4), we could obtain transmission functions of 36 MTF filters. The transmission functions were calibrated by element-wise multiplication of the spectral response of the CMOS image sensor as shown in Fig. 1e.

To analyze the dual-peak resolution of the fabricated MTF filter array spectrometer, we conducted simulations of dual-peak spectra reconstructions. Figure 3a shows an example of a dual-peak spectrum. The recovery performances were investigated for noisy environments ranging from 10 to 35 dB of signal-to-noise ratios (SNRs), more details are in the Methods section. The root mean squared error (RMSE), which is defined as $\sqrt{\|\mathbf{x}_{refer} - \mathbf{x}_{recon}\|_2^2 / N}$, was used to evaluate the performances. The result of dual-peak spectra reconstructions with respect to SNRs is shown in Fig. 3b. We considered four kinds of dual-peak spectra. The FWHMs of a peak were 1 and 2 nm, respectively and the gaps between peaks were 2 and 3 nm, respectively. For each kind of dual-peak, spectra were created by changing the location of dual-peak in the wavelength range of 500 to 849 nm. The reconstructions were performed on all these spectra, and RMSEs were calculated. We averaged the RMSEs and regarded the average RMSE value as the performance of the fabricated MTF filter array to reconstruct dual-peak in noisy environments. As shown in Fig. 3b, the average RMSEs of dual-peak with the FWHM of 1 nm and the gap of 2 nm were 0.0268 for 30 dB, 0.0481 for 20 dB, and 0.0643 for 10 dB, respectively. Similar performances were obtained from the other three kinds of dual-peak spectra. We could find that the fabricated MTF filter array performs well to reconstruct dual-peak spectra in noisy environments.

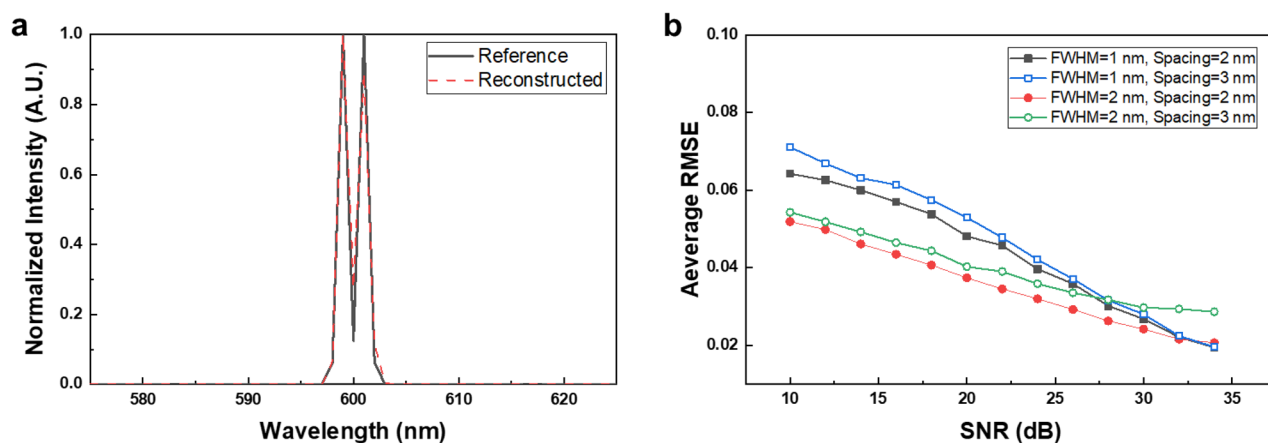


Figure 3. Simulation result of dual-peak spectra reconstructions using MTF filter array. (a) An example of dual-peak spectra with an FWHM of 1 nm with 2 nm apart. (b) Reconstruction performances of dual-peak spectra with respect to signal-to-noise ratios.

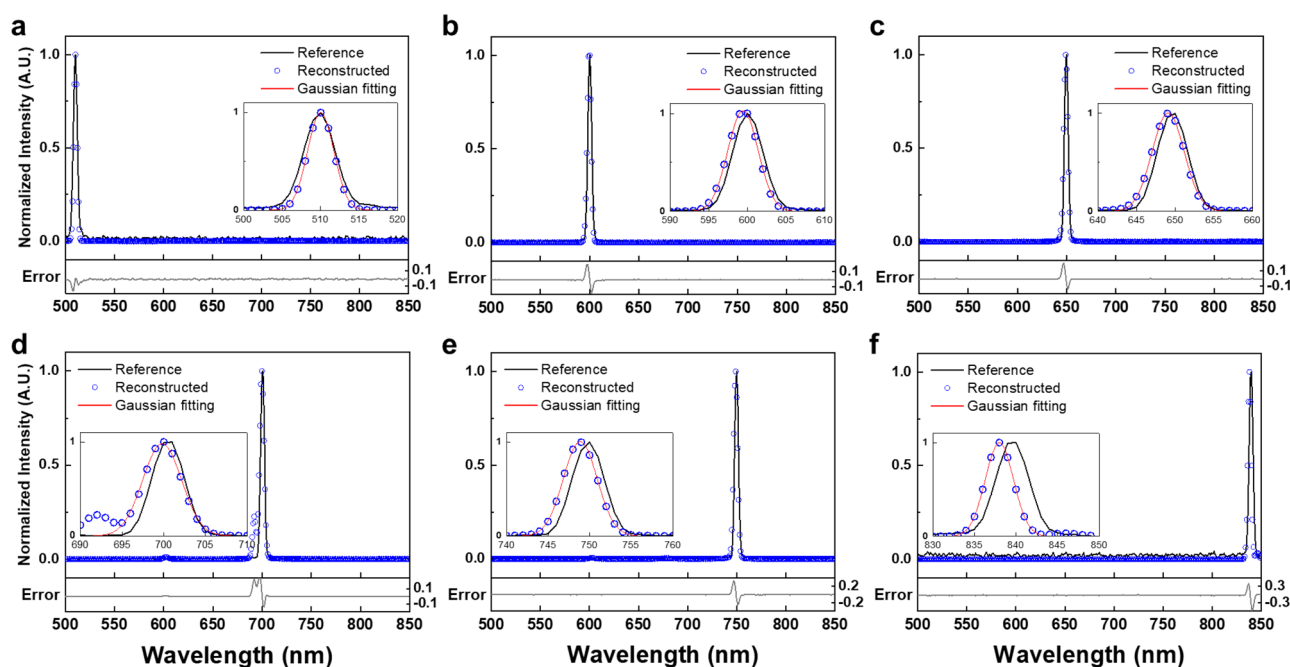


Figure 4. Spectral reconstruction of monochromatic light sources. Monochromatic light with an FWHM of 4 nm at peak wavelengths of (a) 510 nm, (b) 600 nm, (c) 650 nm, (d) 700 nm, (e) 750 nm, and (f) 840 nm. Solid black lines represent reference spectra which are measured by the grating-based spectrometer. Blue circles represent reconstructed spectra using the MTF filter array spectrometer. Solid red lines represent the results of Gaussian fitting. Solid light gray lines represent error between reconstructed and reference spectra.

After analyzing the dual-peak resolution of the MTF filter array in simulations, we tested monochromatic lights by varying the peak wavelength. The CMOS monochrome camera was used for measuring the intensities of test lights. The pixel size of the CMOS image sensor is $5.2 \times 5.2 \mu\text{m}$. Underneath each filter, there are approximately 60×60 pixels. However, considering a case where the layer's location mismatch may occur during the fabrication process of the MTF filter, we excluded the boundary pixels. The averaged intensities from 30×30 pixels at the center of each filter were used for the spectral reconstruction experiments. Using a grating-based spectrometer (Black-Comet, StellarNet), monochromatic lights were measured for use as a reference.

Figure 4 shows the reconstruction results for the monochromatic light. For ease of comparison, reference spectra and reconstructed spectra are normalized. Solid black lines and blue circles in Fig. 4 represent reference spectra and reconstructed spectra, respectively. Reference spectra have peak wavelengths at 510, 600, 650, 700, 750 and 840 nm with FWHMs of 4 nm. As depicted in the inset enlarged graph, the reconstructed spectra using the MTF filter array spectrometer matched the reference spectra. More specifically, differences of peak wavelengths between reference and reconstructed spectra were within 2 nm. The RMSEs were 0.023, 0.023,

Monochromatic light	510 nm	600 nm	650 nm	700 nm	750 nm	840 nm
Peak Center (nm)	509.995	599.438	649.092	699.808	748.878	838.001
Peak shift (nm)	0.005	0.562	0.908	0.192	1.122	1.999
FWHM (nm)	4.008	4.597	4.887	5.486	4.79	4.003

Table 1. Evaluation of monochromatic lights reconstructions using the Gaussian curve fitting.

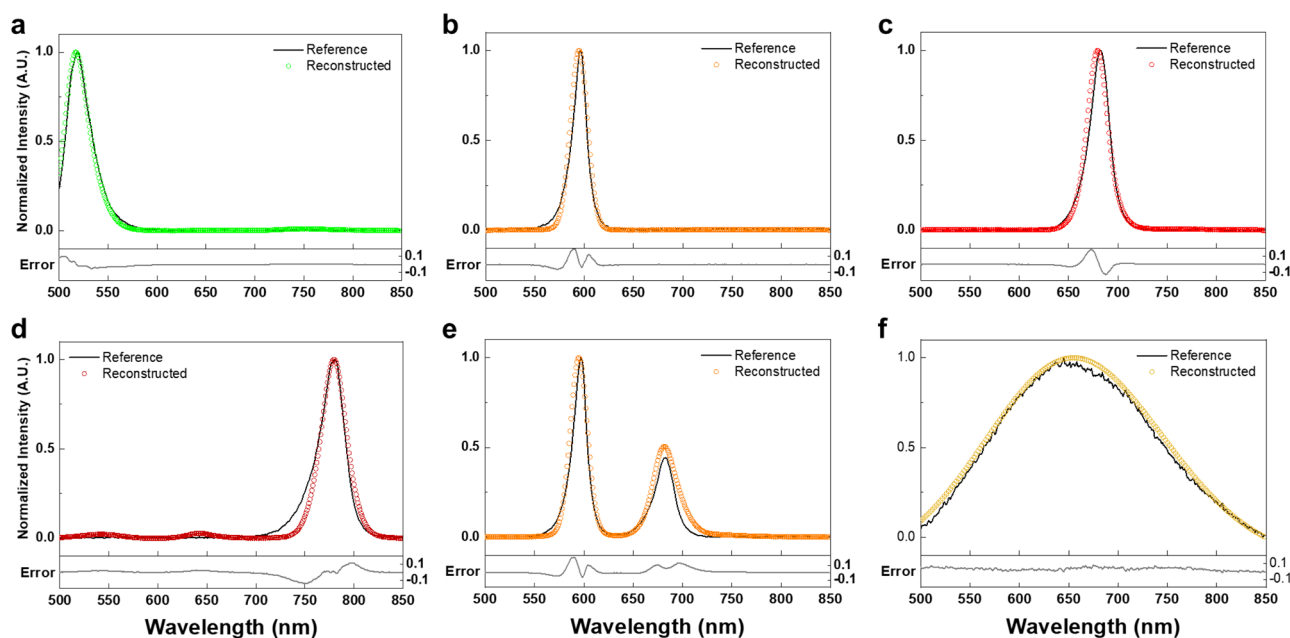


Figure 5. Spectral reconstructions of LEDs and a halogen light source. (a) a green LED, (b) an orange LED, (c) a red LED, (d) an infrared LED, (e) Combined two LEDs (orange and red), and (f) a halogen light source. Solid black lines represent reference spectra which are measured by the grating-based spectrometer. Colored circles represent reconstructed spectra using the MTF filter array spectrometer. Solid light gray lines represent error between reconstructed and reference spectra.

0.021, 0.035, 0.035, and 0.061, for wavelengths 510, 600, 650, 700, 750, and 840 nm, respectively, as shown in Fig. 4a–f. Table 1 presents the evaluation of monochromatic light reconstructions using Gaussian fittings. Over monochromatic lights, peak shifts and FWHMs were within 2 nm and 5.5 nm, respectively. Spectral reconstruction performance seems to degrade in the long-wavelength range due to the low spectral response of the CMOS image sensor and the monotonous spectral features of MTF filters.

We further explored the performance of the MTF filter array spectrometer using broadband light sources, such as LEDs and a halogen light source. Figure 5 shows the spectral reconstruction results. Solid black lines represent reference spectra, which are measured using the grating-based spectrometer. Colored circles represent reconstructed spectra using the MTF filter array spectrometer. Three single-color visible LEDs and one single-color infrared LED were used for spectral reconstruction experiments, as shown in Fig. 5a–d. A green LED (LED 525E, Thorlabs) with an FWHM of 32 nm was reconstructed with an RMSE of 0.021. An orange LED (LED 600L, Thorlabs) with an FWHM of 12 nm was reconstructed with an RMSE of 0.034. A red LED (LED 680L, Thorlabs) with an FWHM of 16 nm was reconstructed with an RMSE of 0.035. An infrared LED (LED 780E, Thorlabs) with an FWHM of 25 nm was reconstructed with an RMSE of 0.044. Similar to experimental results of monochromatic light, the reconstruction performance is relatively poor for spectrum in the long-wavelength range. In addition, we conducted the spectral reconstruction for combined LEDs (an orange LED and a red LED), as shown in Fig. 5e. A beam splitter is used to measure the light of the combined LED. The combined LED was reconstructed with an RMSE of 0.044. Finally, the spectral reconstruction of the halogen light source was conducted. The halogen light source, with an FWHM of 180 nm, was reconstructed with an RMSE of 0.034, as shown in Fig. 5f. As evidenced by low RMSE values, reconstructed spectra agree well with reference spectra measured by the grating-based spectrometer.

Computational pinhole spectral imaging. Furthermore, we demonstrated spectral imaging using the MTF filter array. As shown in Fig. 6a, the pinhole imaging system was constructed by combining a pinhole (Edmund optics), whose aperture diameter is 150 μm , with the monochrome CMOS image camera. The MTF filter array was placed in front of the pinhole. A single filter was adjusted to the pinhole to allow an incident image to pass through the filter and pinhole, and the filtered image was measured by the CMOS image sensor.

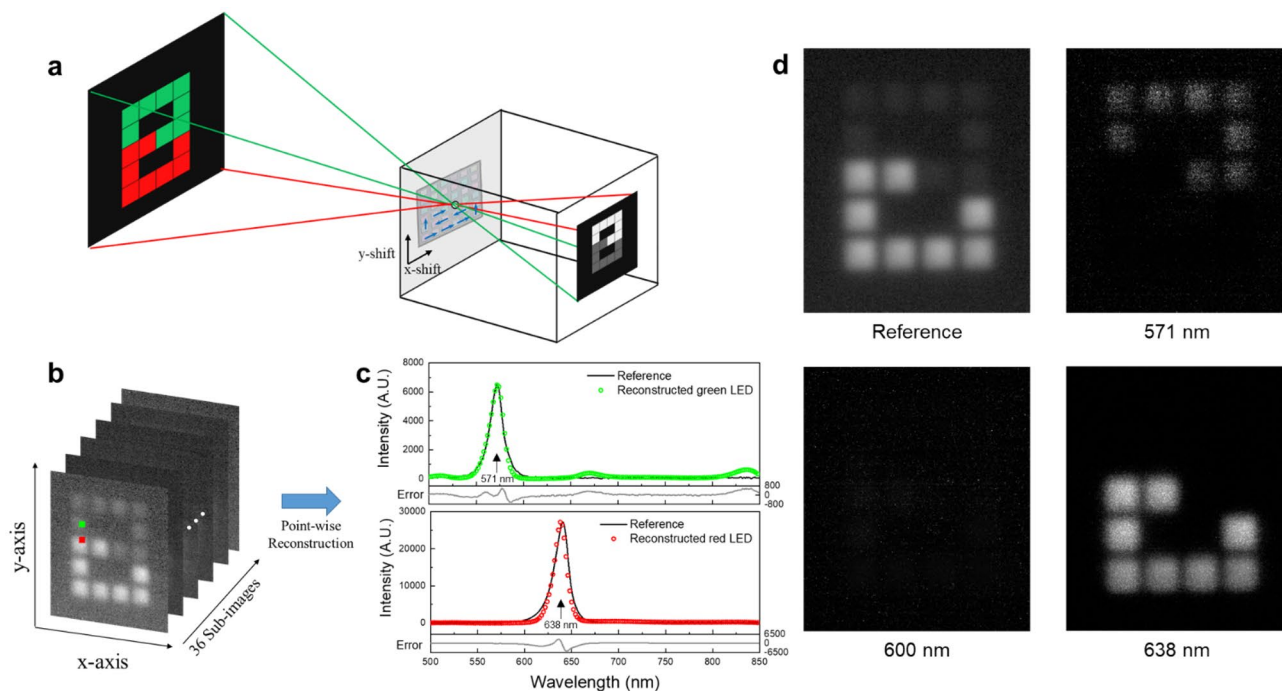


Figure 6. Computational pinhole spectral imaging. **(a)** Schematic of pinhole imaging; The MTF filter array is placed in front of the pinhole camera. A single filter is adjusted to the pinhole and the filtered image is acquired. By changing filters, 36 filtered images are obtained. **(b)** 36 filtered images of 8×8 LED matrix showing the number “8”. The upper part consists of green LEDs and the lower part consists of red LEDs. **(c)** Point-wise spectral reconstruction: a pixel of a green LED block and a red LED block which are denoted in **(b)**. Solid light gray lines represent error between reconstructed and reference. **(d)** Monochrome image of reference and reconstructed monochrome images at 571, 600 and 638 nm, respectively.

By changing filters using a linear translation stage (Newport), 36 filtered images are obtained. Bi-Color 8×8 LED matrix (Adafruit) was used to generate a target. We made a small display by connecting the LED matrix to an Arduino Uno (Arduino) and by controlling the color of the 64 blocks. The number “8” was represented by the LED cube. The upper blocks consist of green LEDs, and the lower blocks consist of red LEDs. Figure 6b shows a stack of the filtered 36 sub-images. A 1280×1024 size image was reduced to a sub-image size of 350×300 by discarding unnecessary pixels. Thus, a data cube with $350 \times 300 \times 36$ in size was obtained. Spectral reconstruction was performed for each pixel, and the data cube was restored with a size of $350 \times 300 \times 350$. It took ~ 1.8 h to reconstruct the data cube. As shown in Fig. 6c, the reference spectra measured by the grating-based spectrometer are shown in solid black lines. As denoted pixels in Fig. 6b, the reconstructed spectra of a pixel in the green LED block and a pixel in the red LED block are represented in Fig. 6c as green circles and red circles, respectively. The RMSE was calculated after normalizing reference spectra and reconstructed spectra. The green LED with an FWHM of 15 nm was reconstructed with an RMSE of 0.0315. The red LED with an FWHM of 20 nm was reconstructed with an RMSE of 0.0370. Figure 6d shows the monochrome image of reference and reconstructed monochrome images at 571, 600, and 638 nm. The pinhole imaging system also measured the monochrome image of reference without the MTF filter array. Since the spectral component of the red LED does not exist at 571 nm, only the upper blocks of the number “8” are shown in the reconstructed monochrome image at 571 nm. On the other hand, only the lower blocks of the number “8” are shown in the reconstructed monochrome image at 638 nm, where the spectral component of the green LED does not exist. Finally, nothing is displayed in the reconstructed monochrome image at 600 nm, where spectral components of the green and red LED do not exist.

As proof-of-principle of the spectral imaging, we implemented the spectral scanning method on the pinhole imaging system. While reconstructed spectra of the pinhole spectral imaging match well with reference spectra, there are improvements to consider. In the spectral scanning method, the data cube acquisition time is long so that spectral smearing may occur in the case of a moving target. The non-scanning method, such as in snapshot spectral imaging systems, can solve these problems by acquiring the data cube in a single exposure. We assume that it is possible to construct a snapshot spectral imaging system combining the MTF filter array and a thin observation module by bound optics (TOMBO)^{32–34} structure. This spectral imaging system requires a microlens array and a single separator but does not need the MTF filter to be as small as pixel size. Rather, the MTF filter should be made large so that many pixels are underlying the filter. The MTF filter array can be fabricated in scalable using stencil lithography techniques according to the spatial resolution of the spectral imaging system.

Discussion

In conclusion, we mass-produced MTF filter arrays using stencil lithography and experimentally demonstrated the spectral resolvability of an MTF filter array-based computational spectrometer. 169 identical filter arrays with 36 MTF filters were fabricated on a single wafer. Although the MTF filter size was larger than that of the photonic crystal slabs^{14,16}, it can be improved to a smaller size by using advanced lithography techniques and facilities. In addition, by using a higher refractive index material, the number of layers of the MTF filter can be reduced so that manufacturing efficiency can be improved.

Using the random spectral features of MTF filters and numerical optimization techniques, we recover varied spectra from the visible range to the near-infrared range (500 to 849 nm) with 1 nm spacing. The spectral reconstruction performance in the near-infrared range is relatively inferior to the visible range, but it can be further improved by using a CMOS image sensor with a high spectral response in the near-infrared region. Also, computational spectral imaging with the MTF filter array was demonstrated using the spectral scanning method. The reconstructed data cube was found to match well with spatial and spectral references. However, to use the spectral imaging system in mobile applications, a shorter data cube acquisition time is required. By utilizing the TOMBO structure with the MTF filter array, it is possible to construct a snapshot spectral imaging system that has a short acquisition time.

Finally, the production of the MTF filter arrays is an important step towards the industrialization and practical uses of computational spectrometers. This study will be helpful for computational spectroscopy to be used in various applications where compact size, high resolution, and wide working range are required.

Methods

Simulation details. Using Gaussian distribution functions, we generated a dual-peak spectrum \mathbf{x} as shown in Fig. 3a. The spectrum \mathbf{x} is computationally measured as \mathbf{y} by multiplying the sensing matrix \mathbf{R} , i.e., $\mathbf{y} = \mathbf{R}\mathbf{x}$. In addition, we made noisy measurement $\tilde{\mathbf{y}}$ by adding additive noise \mathbf{n} as $\tilde{\mathbf{y}} = \mathbf{y} + \mathbf{n} = \mathbf{R}\mathbf{x} + \mathbf{n}$. SNR in decibels is defined as $10 \log (\|\mathbf{x}\|_2^2 / N\sigma^2)$ where σ is the standard deviation of the noise.

Data availability

Raw data is available from the corresponding author upon reasonable request.

Received: 6 September 2021; Accepted: 8 February 2022

Published online: 08 March 2022

References

- Clark, R. N. & Roush, T. L. Reflectance spectroscopy: Quantitative analysis techniques for remote sensing applications. *J. Geophys. Res. Solid Earth* **89**, 6329–6340 (1984).
- Bacon, C. P., Mattley, Y. & DeFrece, R. Miniature spectroscopic instrumentation: Applications to biology and chemistry. *Rev. Sci. Instrum.* **75**, 1–16 (2004).
- Kim, S. *et al.* Smartphone-based multispectral imaging: System development and potential for mobile skin diagnosis. *Biomed. Opt. Express* **7**, 5294–5307 (2016).
- Kurokawa, U., Choi, B. I. & Chang, C.-C. Filter-based miniature spectrometers: Spectrum reconstruction using adaptive regularization. *IEEE Sens. J.* **11**, 1556–1563 (2011).
- Oliver, J., Lee, W., Park, S. & Lee, H.-N. Improving resolution of miniature spectrometers by exploiting sparse nature of signals. *Opt. Express* **20**, 2613–2625 (2012).
- Oliver, J., Lee, W.-B. & Lee, H.-N. Filters with random transmittance for improving resolution in filter-array-based spectrometers. *Opt. Express* **21**, 3969–3989 (2013).
- Wang, Z. & Yu, Z. Spectral analysis based on compressive sensing in nanophotonic structures. *Opt. Express* **22**, 25608–25614 (2014).
- Bao, J. & Bawendi, M. G. A colloidal quantum dot spectrometer. *Nature* **523**, 67–70 (2015).
- Yang, T. *et al.* Miniature spectrometer based on diffraction in a dispersive hole array. *Opt. Lett.* **40**, 3217–3220 (2015).
- Huang, E., Ma, Q. & Liu, Z. Etalon array reconstructive spectrometry. *Sci. Rep.* <https://doi.org/10.1038/srep40693> (2017).
- Oiknine, Y., August, I., Blumberg, D. G. & Stern, A. Compressive sensing resonator spectroscopy. *Opt. Lett.* **42**, 25–28 (2017).
- Cerjan, B. & Halas, N. J. Toward a nanophotonic nose: A compressive sensing-enhanced, optoelectronic mid-infrared Spectrometer. *ACS Photonics* **6**, 79–86 (2018).
- Kim, C., Lee, W.-B., Lee, S. K., Lee, Y. T. & Lee, H.-N. Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy. *Opt. Lasers Eng.* **115**, 53–58 (2019).
- Wang, Z. *et al.* Single-shot on-chip spectral sensors based on photonic crystal slabs. *Nat. Commun.* **10**, 1020 (2019).
- Yang, Z. *et al.* Single-nanowire spectrometers. *Science* **365**, 1017–1020 (2019).
- Zhu, Y., Lei, X., Wang, K. X. & Yu, Z. Compact CMOS spectral sensor for the visible spectrum. *Photonics Res.* **7**, 961–966 (2019).
- Yang, Z., Albrow-Owen, T., Cai, W. & Hasan, T. Miniaturization of optical spectrometers. *Science* **371**, 0722 (2021).
- August, Y. & Stern, A. Compressive sensing spectrometry based on liquid crystal devices. *Opt. Lett.* **38**, 4996–4999 (2013).
- Kwak, Y., Park, S. M., Ku, Z., Urbas, A. & Kim, Y. L. A pearl spectrometer. *Nano Lett.* **21**, 921 (2020).
- Liu, C. & Sun, Z. Design and fabrication of a metallic irregular F-P filter array for a miniature spectrometer. *Appl. Opt.* **60**, 4948–4953 (2021).
- Li, H. *et al.* A near-infrared miniature quantum dot spectrometer. *Adv. Opt. Mater.* **9**, 2100376 (2021).
- Baraniuk, R. G. Compressive sensing [lecture notes]. *IEEE Signal Process. Mag.* **24**, 118–121 (2007).
- Donoho, D. L. Compressed sensing. *IEEE Trans. Inf. Theory* **52**, 1289–1306 (2006).
- Barry, J. R. & Kahn, J. M. Link design for nondirected wireless infrared communications. *Appl. Opt.* **34**, 3764–3776 (1995).
- Macleod, H. A. *Thin-Film Optical Filters* (CRC Press, 2001).
- Koh, K., Kim, S.-J. & Boyd, S. An interior-point method for large-scale l_1 -regularized logistic regression. *J. Mach. Learn. Res.* **8**, 1519–1555 (2007).
- Bioucas-Dias, J. M. & Figueiredo, M. A. A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image Process.* **16**, 2992–3004 (2007).
- Wright, S. J., Nowak, R. D. & Figueiredo, M. A. Sparse reconstruction by separable approximation. *IEEE Trans. Signal Process.* **57**, 2479–2493 (2009).

29. Wang, S.-W. *et al.* 128 channels of integrated filter array rapidly fabricated by using the combinatorial deposition technique. *Appl. Phys. B* **88**, 281–284 (2007).
30. Candes, E. & Romberg, J. *l1-magic: Recovery of Sparse Signals Via Convex Programming*, Vol. 4, 14 (2005) www.acm.caltech.edu/l1magic/downloads/l1magic.pdf.
31. Koh, K., Kim, S. & Boyd, S. *l1 ls: A Matlab Solver for Large-Scale ℓ_1 -Regularized Least Squares Problems* (2008).
32. Tanida, J. *et al.* Thin observation module by bound optics (TOMBO): Concept and experimental verification. *Appl. Opt.* **40**, 1806–1813 (2001).
33. Gupta, N., Ashe, P. R. & Tan, S. Miniature snapshot multispectral imager. *Opt. Eng.* **50**, 033203 (2011).
34. Geelen, B., Tack, N. & Lambrechts, A. A snapshot multispectral imager with integrated tiled filters and optical duplication. In *Advanced Fabrication Technologies for Micro/Nano Optics and Photonics VI*, Vol. 8613, 861314 (International Society for Optics and Photonics, 2013).

Acknowledgements

This work was supported by a National Research Foundation of Korea (NRF) Grant funded by the Korean government (MSIP) (NRF-2021R1A2B5B03002118).

Author contributions

C.K. and H.-N.L. conceptualized the idea. C.K., P.N. and K.R.L. performed the design and fabrication of the MTF filter arrays. C.K. and P.N. conducted optical experiments and reconstructions. C.K. visualized the experiments results. All authors wrote the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1038/s41598-022-08037-y>.

Correspondence and requests for materials should be addressed to H.-N.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022

Article

Compressive Sensing Spectroscopy Using a Residual Convolutional Neural Network

Cheolsun Kim, Dongju Park and Heung-No Lee *

School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, Korea; csk0315@gist.ac.kr (C.K.); toriving@gist.ac.kr (D.P.)

* Correspondence: heungno@gist.ac.kr; Tel.: +82-62-715-2237

Received: 20 December 2019; Accepted: 20 January 2020; Published: 21 January 2020



Abstract: Compressive sensing (CS) spectroscopy is well known for developing a compact spectrometer which consists of two parts: compressively measuring an input spectrum and recovering the spectrum using reconstruction techniques. Our goal here is to propose a novel residual convolutional neural network (ResCNN) for reconstructing the spectrum from the compressed measurements. The proposed ResCNN comprises learnable layers and a residual connection between the input and the output of these learnable layers. The ResCNN is trained using both synthetic and measured spectral datasets. The results demonstrate that ResCNN shows better spectral recovery performance in terms of average root mean squared errors (RMSEs) and peak signal to noise ratios (PSNRs) than existing approaches such as the sparse recovery methods and the spectral recovery using CNN. Unlike sparse recovery methods, ResCNN does not require *a priori* knowledge of a sparsifying basis nor prior information on the spectral features of the dataset. Moreover, ResCNN produces stable reconstructions under noisy conditions. Finally, ResCNN is converged faster than CNN.

Keywords: spectroscopy; compressed sensing; deep learning; inverse problems; sparse recovery; dictionary learning

1. Introduction

There has been considerable interest in producing compact spectrometers having a high spectral resolution, wide working range, and short measuring time. Such a spectrometer can be used in a broad range of fields such as remote sensing [1], forensics [2], and medical applications [3]. Spectrometers that exploit advanced signal-processing methods are promising candidates. The compressive sensing (CS) [4,5] framework makes it possible for a spectrometer to improve its spectral resolution while retaining its compact size. CS spectroscopy comprises two parts: Capturing a spectrum with a small number of compressed measurements and reconstructing the spectrum from the compressed measurements using reconstruction techniques.

To date, for effective signal recovery in CS spectroscopy, three requirements should be satisfied. First, the spectrum should be a sparse signal or capable of sparse representation on a certain basis. Second, the sensing patterns of optical structures should be designed to have a small mutual coherence [6]. Third, appropriate reconstruction algorithms are required. Note that several sparsifying bases have been used in CS spectroscopy such as a family of orthogonal Daubechies wavelets [7], a Gaussian line shape matrix [8,9], and a learned dictionary [10]. Furthermore, numerous optical structures have been proposed to attain the necessary small mutual coherence for sensing patterns such as thin-film filters [11,12], a liquid crystal phase retarder [13], Fabry–Perot filters [7,14], and photonic crystal slabs [15,16]. As algorithms for reconstructing the original signal, two types of basic reconstruction techniques have been developed: greedy iterative algorithms [17,18] and convex relaxation [19,20]. In CS spectroscopy, the reconstruction algorithms have been used with a sparsity

constraint. Additionally, a non-negativity constraint is used in Reference [16,21]. Combining these three considerations, CS spectrometers have shown stable performance for light-emitting diodes (LEDs) and monochromatic lights.

Since not all signals can be represented as sparse on a fixed basis, prior information on structural features of the spectral dataset is therefore required to generate a best-fit sparsifying basis. Furthermore, a high computational cost is required for reconstruction techniques. Recently, deep learning [22] has been emerging as a promising alternative framework for reconstructing the original signal from the compressed measurements.

Mousavi et al. [23] was the first study on image recovery from structured measurements using deep learning. Moreover, a deep-learning framework for inverse problems has been applied in biomedical imaging for imaging through scattering media [24], magnetic resonance imaging [25,26], and X-ray computed tomography [27]. Kim et al. [28] reported the first attempt to use deep learning in CS spectroscopy. They trained a convolutional neural network (CNN) to output the reconstructed signal from the network. From here on the network reported by Kim et al. will be referred to as CNN.

Unlike CNN [28] in which learnable layers were simply stacked and trained to directly reconstruct the original spectrum, we make a residual connection [29] between the input and output of CNN and train the network to reconstruct the original spectrum by referring the input of the network. As a result, the network learns residuals between the input of the network and the original spectrum. It has been reported that it is easier to train a network when using residual connections than to train a plain network that was simply stacked with learnable layers [25,29]. Lee et al. [25] analyzed the topological structure of magnetic resonance (MR) images and the residuals of MR images. They showed that the residuals possessed a simpler topological structure, thus making learning residuals easier than learning the original MR images. In addition, He et al. [29] demonstrated with empirical results that the residual networks are easy to optimize and they achieved improvements in image-recognition tasks. From these works, we gain insights such that adding residual connections to CNN would improve the spectral reconstruction performance in CS spectroscopy.

In this paper, we aim to propose a novel residual convolutional neural network (ResCNN) for recovering an input spectrum from the compressed sensing measurements in CS spectroscopy. The novelty lies in the proposed ResCNN structure, with a moderate depth of learnable layers and a single residual connection, which provides the desired spectral reconstruction performance. The desired performance here means that the proposed ResCNN offers a performance which is better than that of CNNs as well as that of CS reconstruction with its sparsifying base known. In CS reconstruction, the prior knowledge of a fixed sparsifying basis is useful and offers good sparse representation results. However, in general it is a difficult problem to identify a sparsifying basis for various kinds of spectra and apply the identified basis to have the recovery performance improved. In this regard, it is an important advance to find a simple ResCNN which offers good enough performance. It is also worth to note that the proposed ResCNN is tested with the array type CS spectroscopy, discussed in Section 2, which we have designed with an array of multilayer thin-film filters.

The previous works on CS spectroscopy [7,11,13,14,16] have shown decent reconstruction performance but on limited simple sources such as LEDs and monochromatic lights. Using ResCNNs, we are now able to reconstruct more complex spectra, such as spectra with multiplicity of peaks mixed with a gradual rise-and-fall.

The remainder of this paper is organized as follows. In Section 2, we model the optical structure which is used for CS spectroscopy. In Section 3, we describe the system of CS spectroscopy and the proposed ResCNN. In Section 4, simulated experiments are described. Section 5 presents the results of experiments. In Section 6, we discuss the results. Finally, we conclude this paper in Section 7.

2. Optical Structure

Numerous optical structures have been proposed for CS spectroscopy. It has been reported that CS spectrometers, which have various spectral features in the transmission spectrum, show high

spectral-resolving performance [16]. In this work, we used thin-film filters to model CS spectrometers. Thin-film filters demonstrate a variety of spectral features depending on the materials used, the number of layers, and the thicknesses of the layers. Once the structure of thin-film is determined, a transmission value at a given wavelength λ is defined as follows [30]:

$$T(\lambda) = 1 - \frac{1}{2} \left(|\rho_{TE}(\lambda)|^2 + |\rho_{TM}(\lambda)|^2 \right), \quad (1)$$

where $\rho_{TE}(\lambda)$ and $\rho_{TM}(\lambda)$ are amplitude reflection coefficients. The coefficients represent the fraction of the power reflected by a multilayer thin-film in the transverse electric (TE) and transverse magnetic (TM) modes of an incident light, respectively. We summarized recursive processes for calculating amplitude reflection coefficients in Algorithm 1 [11,12,31].

Algorithm 1: Recursive processes for amplitude reflection coefficients.

Input:	λ Structure parameters: $\theta_1, \mathbf{n} = \{n_1, n_2, \dots, n_l\}, \mathbf{d} = \{d_2, d_3, \dots, d_l\}$.
Step 1:	Calculate θ_k, β_k , and N_k using structure parameters. $\theta_k = \sin^{-1} \left(\frac{n_{k-1}}{n_k} \sin \theta_{k-1} \right)$, for $k = 2, 3, \dots, l$. $\beta_k = 2\pi \cos(\theta_k) n_k d_k / \lambda$, for $k = 2, 3, \dots, l$. $N_k = \begin{cases} n_k / \cos \theta_k & \text{for TE} \\ n_k \cos \theta_k & \text{for TM} \end{cases}$, for $k = 2, 3, \dots, l$.
Step 2:	Obtain η_2 by setting $\eta_l = N_l$. For $k = l-1$ to 2 $\eta_k = N_k \frac{\eta_{k+1} \cos \beta_k + j N_k \sin \beta_k}{N_k \cos \beta_k + j \eta_{k+1} \sin \beta_k}$.
Step 3:	Compute $\rho = (N_1 - \eta_2) / (N_1 + \eta_2)$.
Output:	ρ

Here, θ_k is the angle of an incident light passing from k^{th} to $k+1^{\text{th}}$ layer. The refractive index of k^{th} layer is denoted as n_k . d_k denotes the thickness of the k^{th} layer. Given a wavelength vector $\lambda = (\lambda_1 \lambda_2 \dots \lambda_N) \in \mathbb{R}^{1 \times N}$ in the range of interest, i.e., $\lambda_{\max} - \lambda_{\min}$. Let $\Delta\lambda = \frac{\lambda_{\max} - \lambda_{\min}}{N}$. Then, evaluating the function at the integer multiple of $\Delta\lambda$, i.e., $T(\lambda = \lambda_{\min} + n\Delta\lambda)$ for $n = 0, 1, \dots, N-1$, we obtain the vector of transmission spectrum $\mathbf{T}_m \in \mathbb{R}^{1 \times N}$ for the wavelength range. Then, the sensing pattern matrix of optical structures $\mathbf{T} \in \mathbb{R}^{M \times N}$ is obtained by repeating the calculation of \mathbf{T}_m for $m = 1, 2, \dots, M$.

We have used SiNx and SiO₂ for high- and low-refractive index materials, respectively. We numerically generated thin-film filters by alternately stacking high- and low-refractive index materials, changing the number of layers, and varying the thickness of each layer. The number of layers in each filter is in the interval of (19, 24), and the thickness (nm) of each layer is in the interval of (50, 300). Initially, we randomly generated reference filters and compute the mutual coherence among the filters. Then, new filters were generated by changing thicknesses of the layers and the mutual coherence of the filters is compared to the mutual coherence of reference filters. Filters with a smaller mutual coherence then became the new reference filters. This process is repeated until reasonable reference filters with the required small mutual coherence are obtained.

Figure 1 shows the heatmap for the transmission spectra of the reference filters and two selected transmission spectra. In Figure 1a, each of the transmission spectra shows a unique sensing pattern because of the iterative modeling process of the reference filters based on mutual coherence. Figure 1b shows two transmission spectra that correspond to the 15th and 30th rows in the heatmap of reference filters. The transmission spectrum reveals a deep spectral modulation depth and various features such as broadband backgrounds, multiple peaks with a small full width at half maximums (FWHMs), and irregular fluctuations.

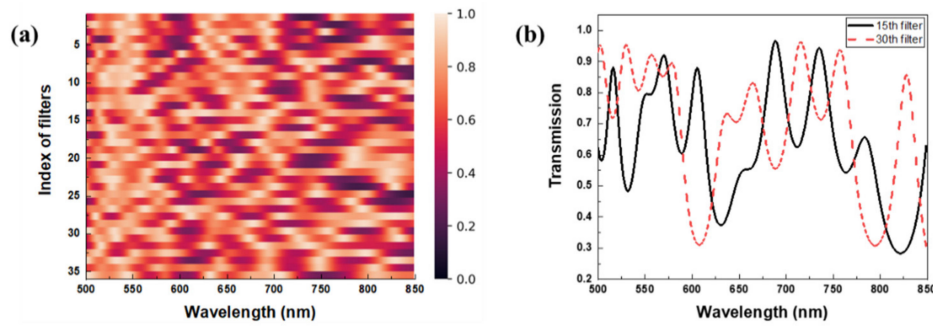


Figure 1. (a) Heatmap of the sensing matrix: each row represents the transmission spectrum of the designed thin-film filter. (b) Two transmission spectra corresponding to the 15th and 30th rows in the sensing matrix.

3. Compressive Sensing (CS) Spectrometers Using the Proposed Residual Convolutional Neural Network (ResCNN)

3.1. CS Spectrometers

In CS spectroscopy, the measurement column vector $\mathbf{y} \in \mathbb{R}^{M \times 1}$ is represented using the following relation:

$$\mathbf{y} = \mathbf{T}\mathbf{x}, \quad (2)$$

where $\mathbf{x} \in \mathbb{R}^{N \times 1}$ is the spectrum column vector of incident light and $\mathbf{T} \in \mathbb{R}^{M \times N}$ is the sensing matrix of the optical structure. Each row of \mathbf{T} represents a transmission spectrum. Because the length of the measurement vector is smaller than the length of the spectrum vector ($M < N$), the system is underdetermined. Conventionally, if \mathbf{x} is a sparse signal or can be sparsely represented in a certain basis, i.e., $\mathbf{x} = \Phi\mathbf{s}$, reconstruction algorithms can determine a unique sparse solution $\hat{\mathbf{s}}$ from the following optimization problem:

$$\min_{\mathbf{s}} \|\mathbf{T}\Phi\mathbf{s} - \mathbf{y}\|_2^2 + \tau\|\mathbf{s}\|_1, \quad (3)$$

where $\Phi \in \mathbb{R}^{N \times N}$ is a sparsifying basis and τ is a regularization parameter. Here, \mathbf{s} is a sparse signal whose components are zero except for a small number of non-zero components. Then, the recovered spectrum $\hat{\mathbf{x}}$ is $\Phi\hat{\mathbf{s}}$. In this paper, we refer to the methods of solving the optimization problem using Equation (3) as sparse recovery.

Typically, except for narrow-band spectra, a spectrum is not a sparse signal, and a fixed sparsifying basis cannot transform all spectra into sparse signals. Clearly, the use of a fixed basis may lead the sparse recovery to struggle, as no fixed basis will transform every signal into a sparse signal. In addition, the sparse recovery is time-consuming and takes a high computational cost.

Our goal is to overcome the limitations of the sparse recovery in CS spectroscopy and recover various kinds of spectra using ResCNN. Figure 2 shows the schematic of the CS spectroscopy system using ResCNN. This system consists of two parts: compressive sampling and dimension extension, and the reconstruction using ResCNN. In the compressive sampling and dimension extension, the measurement vector \mathbf{y} is obtained from Equation (1), which then transforms into $\tilde{\mathbf{x}} \in \mathbb{R}^{N \times 1}$ using a linear transformation. A transform matrix $\mathbf{A} \in \mathbb{R}^{N \times M}$ extends the M dimension of \mathbf{y} to N dimension of $\tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}}$ is a representative spectrum corresponding to \mathbf{x} . We used $\tilde{\mathbf{x}}$ as the input for the reconstruction. ResCNN learnt a non-linear mapping between $\tilde{\mathbf{x}}$ and \mathbf{x} , and afforded a reconstructed spectrum $\hat{\mathbf{x}} \in \mathbb{R}^{N \times 1}$. The dimension extension by the transform matrix was used to make it easier for ResCNN to extract features and reconstruct spectra from the non-linear mapping.

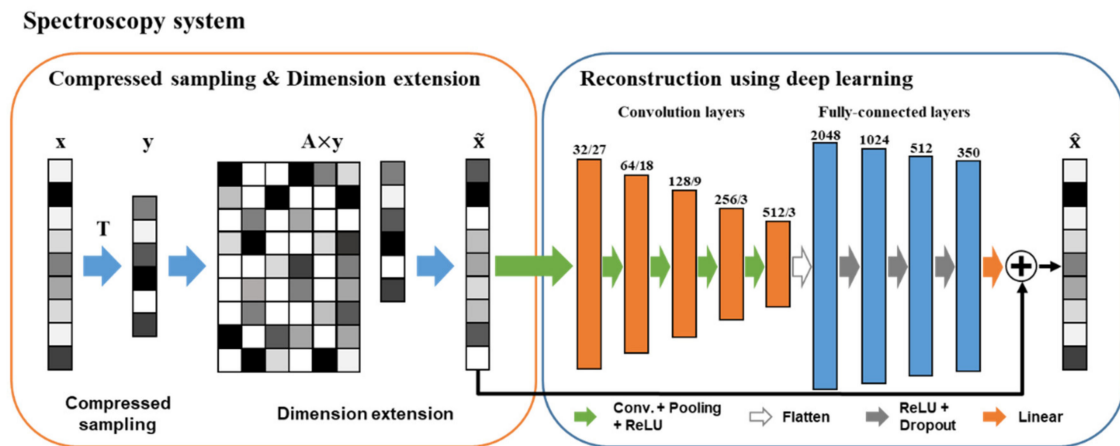


Figure 2. Overview of compressive sensing (CS) spectroscopy system including the proposed residual convolutional neural network (ResCNN): An input spectrum is compressively sampled by the sensing matrix, and the dimension of measurements is extended by the transform matrix. ResCNN is trained to recover the original spectrum from the extended measurements.

3.2. The Proposed ResCNN

As depicted in Figure 2, ResCNN comprises nine learnable layers, five of which are convolution layers, four are fully-connected layers, and one is a residual connection. Convolution layers are used for the feature extraction in the non-linear mapping between \tilde{x} and x . Fully-connected layers are used for the spectra reconstruction. Each of the convolution layers has a set of one-dimensional learnable kernels with specific window sizes. The number of kernels and the window sizes are indicated in Figure 2. After every convolutional layer, the rectified linear unit (ReLU) is used as an activation function, and the subsampling is then applied. We use non-overlapping max-pooling to down-sample the output of the activation function. We stack the convolutional layer, the ReLU, and the subsampling five times. The output of the last subsampling is flattened and then fed into the subsequent four fully-connected layers. The first three layers are followed by the ReLU and dropout in sequence. The dropout is introduced to reduce the overfitting of ResCNN. The output of the last fully-connected layer is fed into a linear activation function. The number of units in each of the fully-connected layers is noted in Figure 2. Unlike CNN [28] in which learnable layers are simply stacked, we make the residual connection that the representative spectrum \tilde{x} and the output of the linear activation function are added up to the reconstructed spectrum \hat{x} . Consequently, \hat{x} is trained to become x . Given training data $\{x_t^i\}_{i=1}^k$, we train ResCNN to minimize a loss function L . We use the mean squared error between the original x_t and recovered \hat{x}_t as the loss function:

$$L = \frac{1}{k} \sum_{i=1}^k \|x_t^i - \hat{x}_t^i\|_2^2. \quad (4)$$

The non-linear mapping that \tilde{x} becomes x can be defined as $H(\tilde{x}) = x$. Because of the residual connection in ResCNN, $H(\tilde{x})$ can be rewritten as $H(\tilde{x}) = F(\tilde{x}) + \tilde{x}$, where $F(\tilde{x})$ is the mapping of the learnable layers. The representative spectrum \tilde{x} is referenced by the residual connection, and then, $F(\tilde{x}) = H(\tilde{x}) - \tilde{x}$. In particular, the mapping of $F(\tilde{x})$ is called a residual mapping; therefore, the learnable layers learn the residual of x and \tilde{x} .

The previous researches [25,29] have used numerous residual connections in very deep neural networks in order to make networks converge faster by avoiding vanishing gradient problems. We use one residual connection between input and output of the moderate depth network. Figure 3 depicts the manner in which a spectrum is recovered in CNN and ResCNN. The learnable layers of CNN directly reconstruct the spectrum from the representative spectrum \tilde{x} . Alternatively, ResCNN reconstructs the

spectrum by passing the representative spectrum \tilde{x} through the residual connection shown in Figure 3b. Consequently, the learnable layers of ResCNN learn to reconstruct residuals.

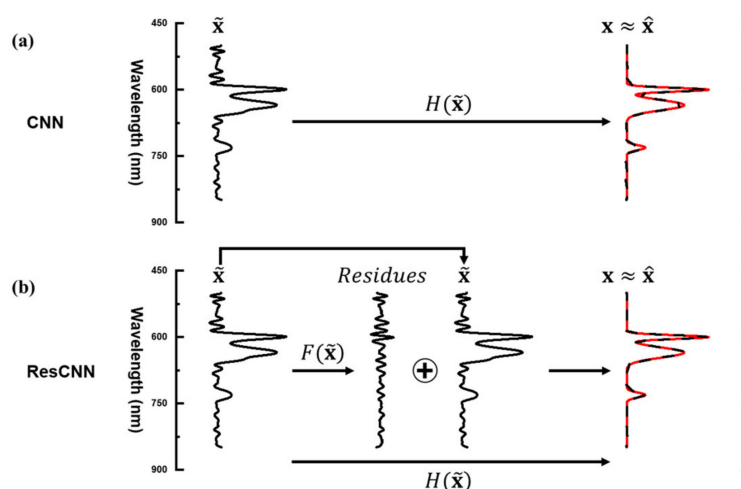


Figure 3. Descriptions of the spectrum recovery process: (a) convolutional neural network (CNN), (b) ResCNN.

4. Simulated Experiments

We reconstructed 350 spectral bands ($N = 350$) using 36 thin-film filters ($M = 36$) whose sensing patterns have a spacing of 1 nm for wavelengths from 500 to 850 nm. We determined the sensing matrix \mathbf{T} , assuming that the incident light falls onto the filters with normal incidence. As the transform matrix \mathbf{A} , we used the Moore–Penrose inverse of the sensing matrix \mathbf{T} , i.e., $\mathbf{A} = \mathbf{T}^T(\mathbf{T}\mathbf{T}^T)^{-1}$.

4.1. Spectral Datasets

To evaluate the performance of ResCNN, we used two synthetic spectral datasets and two measured spectral datasets. The first synthetic dataset is composed of Gaussian distribution functions while the other is composed of Lorentzian distribution functions. These two synthetic datasets were selected as generally these types of functions are used to represent spectral line shapes. As shown in Figure 4, component functions are added to produce the spectra. We generated 12,000 spectra for each dataset. For each spectrum, the number of component functions was generated using a geometric distribution with the probability parameter p set to 0.3. We added one to the number of component functions to prevent the number of component functions from becoming zero. Then, we randomly set a location, a height, and an FWHM of each peak. To set a peak location (nm), an integer number was randomly selected from a uniform distribution with the interval (500, 849). A random number from a uniform distribution in the interval (0, 1) was used for the height. An integer number for an FWHM (nm) was randomly drawn from a uniform distribution with the interval (2, 50). Finally, all of the component functions were summed to generate the spectrum. The height of each generated spectrum was normalized such that it was mapped from zero to one.

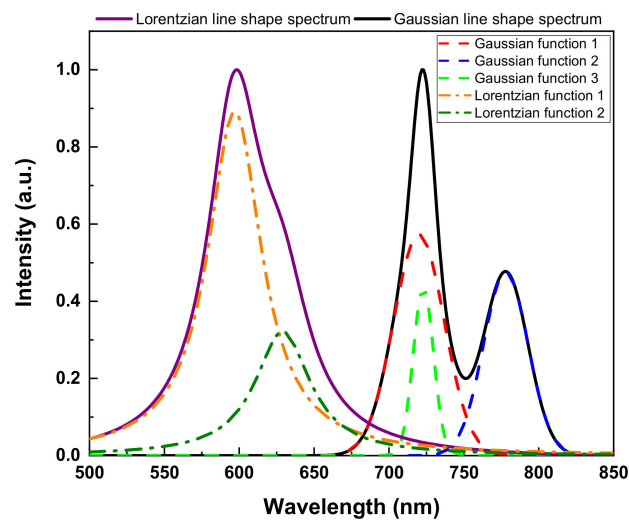


Figure 4. Examples of two synthetic spectra: the solid purple line is composed of two Lorentzian distribution functions (dash-dotted orange and olive lines), and the solid black line is composed of three Gaussian distribution functions (dashed red, blue, and green lines).

As measured datasets, we used the US Geological Survey (USGS) spectral library version 7 [32], and the glossy Munsell colors spectral dataset [33]. The USGS spectral library provides reflectance spectra for artificial materials, coatings, liquids, minerals, organic compounds, soil mixtures, and vegetation. We discarded any spectrum that has missing spectral bands. Then, we extracted the spectrum in the wavelength range of interest (500 to 849 nm) from the wavelength range of the original spectrum (350 to 2500 nm). The measured wavelength range for the glossy Munsell colors spectral dataset, which contains the reflectance spectra of the glossy Munsell color chips, was 380 to 780 nm. The wavelength range of the original spectrum was different from the wavelength range of interest. We decided to use the wavelength range from 400 to 749 nm to ensure each spectrum was set to 350 spectral bands. This selection of wavelengths is reasonable because the wavelengths were located in the center of the wavelength range of the original spectrum, and showed different spectral features with respect to each spectrum. In addition, our aim was to show the reconstruction performance with respect to various kinds of spectra. Finally, each spectrum was normalized such that the height varies from 0 to 1. Overall, 1473 spectra from USGS spectral dataset and 1600 spectra from Munsell color spectral dataset were used for our simulated experiments. Table 1 lists the details of each of the spectral datasets.

Table 1. Description of the spectral datasets.

Dataset	Training/Validation/Test	Avg. Number of Nonzero Values	Description
Gaussian dataset	8000/2000/2000	336.8/350	FWHM (nm) on the interval [2, 50], Height on the interval [0, 1]
Lorentzian dataset	8000/2000/2000	349/350	FWHM (nm) on the interval [2, 50], Height on the interval [0, 1]
US Geological Survey [32]	982/246/245	348.9/350	350–2500 nm, 2151 spectral bands (we use 350 spectral bands in 500–849 nm)
Munsell colors [33]	1066/267/267	349/350	380–780 nm, 401 spectral bands (we use 350 spectral bands in 400–749 nm)

4.2. Data Preprocessing and Training

Given the sensing matrix, the spectral data are compressively sampled as the measurement vector \mathbf{y} shown in Equation (1), and then transformed into the representative spectrum $\tilde{\mathbf{x}}$ by multiplying the transform matrix \mathbf{A} and \mathbf{y} .

In each spectral dataset, the number of training, validation, and test spectra are randomly assigned using a ratio of 4:1:1 for the synthetic and measured data sets, respectively. The validation spectra are used for estimating the number of epochs and tuning the hyper-parameters. To train ResCNN, we used the Adam optimizer [34] implemented in Tensorflow with the batch size of 16 and 250 epochs. The experiments were conducted on an NVIDIA GeForce RTX 2060 graphics processing unit (GPU). Training the architecture can be done in half an hour for each dataset.

4.3. Sparsifying Bases for Sparse Recovery

Using sparse recovery, we evaluated the performance of conventional CS reconstructions to benchmark the performance of ResCNN. As shown in Table 1, the spectra for both the synthetic and measured datasets are dense spectra. Therefore, we must transform the spectra into sparse signals to solve Equation (3). In this section, we considered methods to make a sparsifying basis Φ .

First, we considered a Gaussian line shape matrix as a sparsifying basis. Each column of the matrix comprises a Gaussian distribution function whose length is N . A collection of N Gaussian functions works as a sparsifying basis $\Phi \in \mathbb{R}^{N \times N}$. We generate two Gaussian line shape matrices. Figure 5a shows the heatmap images for two Gaussian line shape matrices. Seven different FWHMs are used to generate the Gaussian distributions. Given an FWHM, Gaussian distributions are generated by shifting the peak location using uniform spacing. To create a small dissimilarity between the two Gaussian line shape matrices, two of the seven FWHMs in Gaussian 1 were replaced with other FWHMs, thus producing Gaussian 2, as shown in Figure 5a.

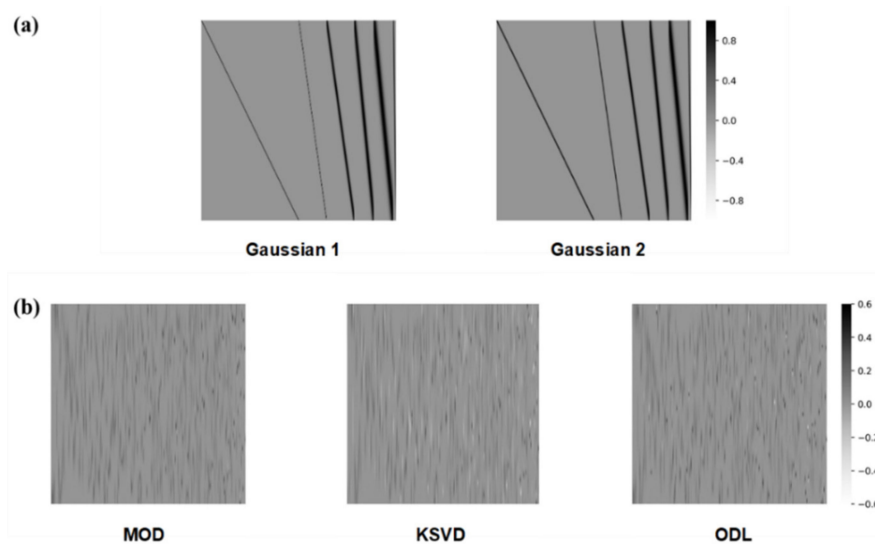


Figure 5. Heatmap images of sparsifying bases that were used in simulated experiments: (a) Gaussian line shape matrices, (b) the learned dictionaries which are from the Gaussian training dataset.

Second, a learned dictionary [35–38] is used as a sparsifying basis. Given a training dataset $\{\mathbf{x}_t^i\}_{i=1}^k$, we can derive a learned dictionary Φ that sparsely represents the training data \mathbf{x}_t by solving the following optimization problem, known as the dictionary learning problem:

$$\min_{\Phi, \mathbf{s}_1^1, \dots, \mathbf{s}_t^k} \sum_{i=1}^k \|\mathbf{x}_t^i - \Phi \mathbf{s}_t^i\|_2^2 + \tau \|\mathbf{s}_t^i\|_1, \quad (5)$$

where τ is a regularization parameter and \mathbf{s}_t^i is i th sparse signal over the training dataset. By fixing an initial guess for the dictionary Φ in Equation (5), we obtain a solution for the sparse signals $\{\mathbf{s}_t^i\}_{i=1}^k$. The dictionary is then updated by solving Equation (5) using the sparse signals obtained. This process is iteratively repeated until convergence is reached and we derive the learned dictionary. We used three dictionary learning methods: method of optimal directions (MOD) [36], K-SVD [37], and online dictionary learning (ODL) [38]. The learned dictionaries are generated for each of the training datasets, and the reconstruction performances are evaluated for each test dataset. Figure 5b shows learned dictionaries identified using the Gaussian training dataset. The learned dictionaries clearly depend on the dictionary-learning methods used. Nevertheless, each column of the dictionaries shows a learned spectral feature from the training dataset.

5. Results

To demonstrate the ability of ResCNN to reconstruct spectra, we evaluated its performance using three different datasets: Synthetic datasets, noisy synthetic datasets, and measured datasets. We used the same hyper-parameters of ResCNN for each of these datasets. Moreover, we adopted $l1_ls$ [39] as the fixed reconstruction algorithm in the sparse recovery. We compared the recovered signal with the original signal by calculating the root mean squared error (RMSE) and the peak signal to noise ratio (PSNR). In addition, the performance of five conventional sparse recovery methods, described in Section 4.3 and CNN was calculated.

5.1. Synthetic Datasets

The two synthetic data sets described in Table 1 were used to perform the signal recovery using sparse recovery and deep learning. Table 2 shows the average RMSE and PSNR for each of the seven methods evaluated. ResCNN shows the smallest average RMSE for both the Gaussian and Lorentzian datasets of 0.0094 and 0.0073, respectively. Moreover, ResCNN shows the largest average PSNR of 49.0 dB for the Lorentzian dataset. For the Gaussian dataset, the sparse recovery method with Gaussian 2 shows the largest average PSNR, 49.7 dB, which is slightly higher than the 47.2 dB for ResCNN. Note that the minor difference between the two Gaussian line shape matrices results in considerable performance difference. However, reconstruction using the learned dictionaries show similar performance across all of the synthetic datasets.

Table 2. Average root mean squared errors (RMSEs) and peak signal to noise ratios (PSNRs) over synthetic datasets.

Dataset	Sparse Recovery					Deep Learning	
	Gaussian 1	Gaussian 2	K-SVD	MOD	ODL	CNN	ResCNN
Gaussian dataset	0.0226 (43.1 dB)	0.0112 (49.7 dB)	0.0172 (40.3 dB)	0.0174 (40.3 dB)	0.0161 (41.1 dB)	0.0132 (40.5 dB)	0.0094 (47.2 dB)
Lorentzian dataset	0.0146 (44.9 dB)	0.0094 (47.5 dB)	0.0136 (42.3 dB)	0.0137 (42.3 dB)	0.0127 (42.9 dB)	0.0101 (42.8 dB)	0.0073 (49.0 dB)

Figure 6 shows the reconstructed test spectra from each of the synthetic datasets. The solid red line (i) is the input spectra from each dataset. ResCNN is shown in dashed black line (ii), while CNN is shown in solid orange lines (iii). The reconstructed spectra using sparse recovery with Gaussian 1 (iv), Gaussian 2 (v), and ODL (vi) are shown in solid green, blue, and purple lines in respectively. Because of the similar performance from each of the learned dictionaries, only the ODL method is shown. The RMSE and PSNR of ResCNN are 0.0138 (37.2 dB) for the spectrum from the Gaussian dataset and 0.0096 (40.4 dB) for the spectrum from the Lorentzian dataset. For the selected spectra, ResCNN achieves superior reconstruction performance compared with the other four reconstructions.

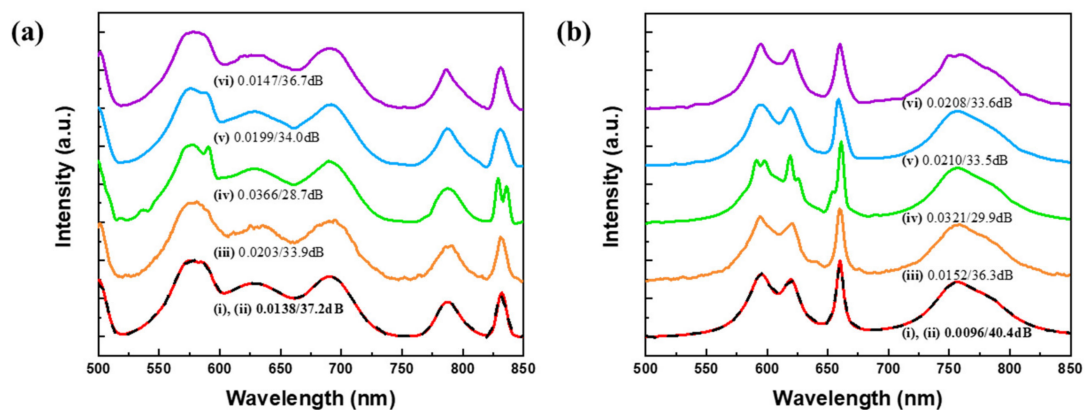


Figure 6. Spectral reconstructions of test spectra in synthetic datasets, (a) Gaussian dataset, (b) Lorentzian dataset. An input spectrum (solid red (i)) is compared with ResCNN (dashed black (ii)), CNN (orange (iii)), sparse recovery: Gaussian 1 (green (iv)), Gaussian 2 (blue (v)), and online dictionary learning (ODL) (purple (vi)). The baselines are shifted for clarity.

Only sparse recovery with Gaussian 1 fails to recover the fine details of the input spectrum. One example of the poor ability of sparse recovery with Gaussian 1 to resolve the signal is the recovery of the peak at ~830 and 590 nm being recovered as two neighboring peaks in Figure 6a,b, respectively. CNN was unable to capture the smoothness of the spectral features compared to the other methods.

5.2. Noisy Synthetic Datasets

To verify the stability of ResCNN, we evaluated the accuracy of the reconstruction at various noise levels. Gaussian white noise was added to the measurement vector $\mathbf{n} \in \mathbb{R}^{M \times 1}$ to Equation (2), i.e., $\mathbf{y} = \mathbf{T}\mathbf{x} + \mathbf{n}$. We considered six different noise levels whose signal-to-noise ratios (SNRs) are 15, 20, 25, 30, 35, and 40 dB. The SNR (dB) is defined as $10 \cdot \log_{10}(\|\mathbf{x}\|_2^2 / N\sigma^2)$, where σ is the standard deviation of the noise. Using Gaussian and Lorentzian datasets, we compared the reconstruction performance of ResCNN with the sparse recovery using Gaussian 2, which shows the best reconstruction performances among sparse recovery methods in synthetic datasets. ResCNN was evaluated with the same hyper-parameters that were used for the noise-free datasets. The average RMSE and PSNR for each of the six noise levels are shown in Table 3. While ResCNN was trained using noise-free data, it outperformed the sparse recovery with Gaussian 2 at every noise level, which indicates that ResCNN remains stable even with noisy datasets.

Table 3. Average RMSE and PSNR under various signal-to-noise ratios (SNRs, dB) with synthetic datasets.

Dataset	Method	SNR (dB)					
		15 dB	20 dB	25 dB	30 dB	35 dB	40 dB
Gaussian Dataset	Sparse recovery + Gaussian 2	0.0796 (22.7 dB)	0.0482 (27.1 dB)	0.0308 (31.2 dB)	0.0215 (34.8 dB)	0.0166 (37.9 dB)	0.0138 (40.7 dB)
	ResCNN	0.0671 (24.2 dB)	0.0401 (28.7 dB)	0.0251 (32.9 dB)	0.0171 (36.6 dB)	0.0130 (39.8 dB)	0.0110 (42.4 dB)
Lorentzian Dataset	Sparse recovery + Gaussian 2	0.0817 (22.6 dB)	0.0483 (27.1 dB)	0.0300 (31.2 dB)	0.0201 (35.0 dB)	0.0147 (38.5 dB)	0.0119 (41.4 dB)
	ResCNN	0.0689 (24.1 dB)	0.0404 (28.7 dB)	0.0243 (33.1 dB)	0.0157 (37.1 dB)	0.0113 (40.6 dB)	0.0091 (43.4 dB)

5.3. Measured Datasets

ResCNN was trained using the two measured datasets listed in Table 1, USGS and Munsell colors, and its reconstruction performance was evaluated. In addition, the signal reconstruction was performed using CNN and sparse recovery with five different sparsifying bases. Table 4 reports the average RMSE and PSNR for each of the seven methods. ResCNN achieves the smallest average RMSE and the largest average PSNR for both datasets. In the USGS dataset, the average RMSE and PSNR of ResCNN are 0.0048 and 52.4 dB, respectively. In addition, ResCNN achieves 0.0040 for the average RMSE and 50.0 dB for the average PSNR in the Munsell colors dataset. Similar to synthetic datasets, all of the learned dictionaries provided similar reconstruction performances. In addition, the small differences between Gaussian 1 and 2 show large differences in the RMSE and PSNR. The average RMSE and PSNR of the learned dictionary methods approach the values of ResCNN for Munsell colors dataset because the Munsell colors dataset has simpler spectral features than the other datasets.

Table 4. Average RMSEs and PSNRs for the measured datasets.

Dataset	Sparse Recovery					Deep Learning	
	Gaussian 1	Gaussian 2	K-SVD	MOD	ODL	CNN	ResCNN
USGS [32]	0.0081 (45.3 dB)	0.0061 (48.4 dB)	0.0070 (48.5 dB)	0.0081 (47.4 dB)	0.0074 (47.6 dB)	0.0116 (40.8 dB)	0.0048 (52.4 dB)
Munsell colors [33]	0.0068 (44.6 dB)	0.0050 (47.5 dB)	0.0040 (49.8 dB)	0.0040 (49.9 dB)	0.0042 (49.5 dB)	0.0076 (43.0 dB)	0.0040 (50.0 dB)

Figure 7 shows the reconstruction results of one test spectra from each of the measured datasets. The spectrum for the organic compound dibenzothiophene in the USGS dataset is reconstructed in Figure 7a. The spectrum of Munsell color 5 PB 2/2 is shown in Figure 7b. The solid red lines are the input spectra (i). ResCNN are shown in dashed black lines (ii), and CNN are shown in solid black lines (iii). The spectra of (iv) to (vi) are reconstructed spectra using the sparse recovery with Gaussian 1, Gaussian 2, and K-SVD. Because of the best performance of the K-SVD among the learned dictionaries only the K-SVD method is shown.

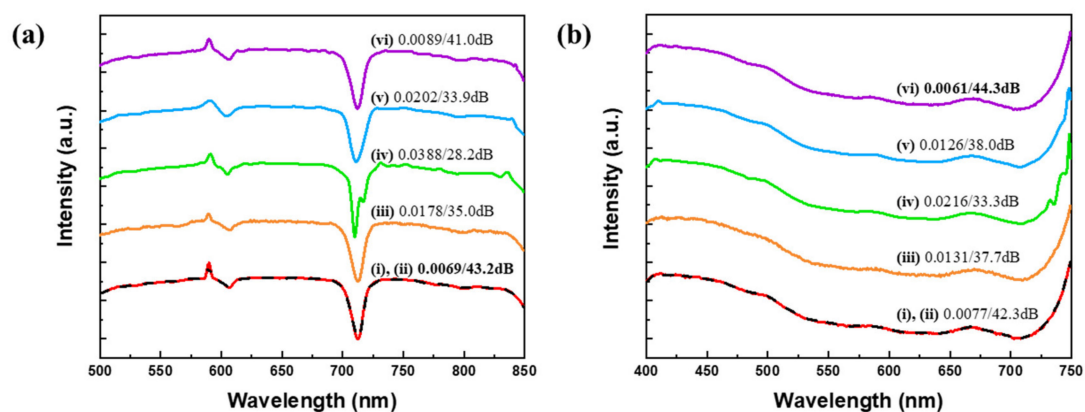


Figure 7. Spectral reconstructions of test spectra in measured datasets: (a) spectrum of organic compound dibenzothiophene in USGS dataset, (b) spectrum of Munsell color 5PB 2/2. The input spectrum (solid red line (i)) is compared with ResCNN (dashed black (ii)), CNN (orange (iii)), sparse recovery: Gaussian 1 (green (iv)), Gaussian 2 (blue (v)), and K-SVD (purple (vi)). The baselines are shifted for clarity.

The RMSE and PSNR for ResCNN are 0.0069 (43.2 dB) for the spectrum from the USGS dataset and 0.0077 (42.3 dB) for the spectrum from the Munsell colors dataset. ResCNN outperforms other approaches for the spectrum from USGS dataset. However, for the spectrum from Munsell colors

dataset, the sparse recovery with K-SVD outperforms ResCNN. ResCNN achieves slightly larger RMSE and smaller PSNR.

The performances of sparse recovery with Gaussian 2 is degraded for measured datasets compared with the performance for synthetic datasets. The measured datasets have rough spectral features unlike the smooth spectral features observed in the synthetic datasets. As a result, the sparse recovery with Gaussian 2 performs worse, because of its inability to represent rough spectral features using Gaussian distribution functions. The performance of sparse recovery with dictionary learning methods are improved for measured datasets compared with the performance of synthetic datasets. Because the number of spectra in measured datasets are smaller than the number of spectra in synthetic datasets. Therefore, finding the best-fit sparsifying basis for measured datasets is easier than finding the best-fit sparsifying basis for synthetic datasets using dictionary-learning methods. Meanwhile, ResCNN shows superior reconstruction performances regardless of spectral features of datasets and the size of datasets.

6. Discussion

As shown in the results, we demonstrate empirically that ResCNN outperforms the sparse recovery methods and the CNN over all datasets. The sparse recovery shows unstable performance because it is highly dependent on the sparsifying basis and spectral features of dataset. This is a direct result of being unable to identify a fixed sparsifying basis that can transform any spectra into a sparse signal, which means the *a priori* structural information such as line shapes and FWHMs is required to select a consistent sparsifying basis. Learned dictionaries are used to cope with the problem of identifying a consistent sparsifying basis. The columns of learned dictionaries are composed of learned spectral features from the training dataset. While this shows an improvement in measured datasets, a learned dictionary is still limited to representing all the spectral features in the large dataset (i.e., synthetic datasets) using linear combinations of columns of the learned dictionary.

Compression approaches for summarizing information with a small number of sensors were proposed in [40]. These approaches can be exploited to generate a sparsifying basis by reducing the loss of spectral information in large datasets.

To improve the reconstruction performance in sparse recovery, pre-defined structure information and side information of unknown target signals were used in [41,42]. The reconstruction of three-dimensional electrical impedance tomography was improved by updating three-dimensional structural correlations using pre-defined structured signals [41]. To recover multi-modal data, a reconstruction framework is proposed in [42] that uses side information in unrolled optimization. Unrolled optimization approaches using deep learning were proposed in [43,44]. Deep-learning architectures were used to train hyper-parameters, such as a gradient regularizer and a step size. Using learned hyper-parameters, it was shown optimized solutions can be obtained within a fixed number of iterations. These proposed approaches for image reconstruction have assumed random sensing matrix and structured or sparse signals. In this work, however, we consider dense spectra and the sensing matrix from thin-film filters for the real implementation. Moreover, the reconstruction performance may change to a sparsifying basis as shown in results because a reconstructed spectrum $\hat{\mathbf{x}}$ should be represented as a linear combination of columns of a fixed sparse basis Φ as $\Phi\hat{\mathbf{s}}$.

For recovering spectra, ResCNN does not require the *a priori* knowledge of a sparsifying basis or prior information of spectral features. During training, ResCNN learns the spectral features using learnable layers, which enable it to recover the fine details for various kinds of spectra without identifying a sparsifying basis.

ResCNN is directly compared with CNN for the synthetic Gaussian dataset in Figure 8a where the mean squared error (Equation (4)) is plotted with respect to the epoch. The mean squared error for CNN and ResCNN are shown in solid black line and solid red line with square symbols, respectively. ResCNN shows a lower mean squared error than that of CNN. Moreover, ResCNN converges faster than CNN, indicating that ResCNN optimizes the learnable layers quicker, as expected based on

previous research using residuals [25,29]. In contrast to the previous research that numerous residual connections were used in very deep neural networks to converge networks faster by avoiding vanishing gradient problem, we achieve spectral reconstruction improvements even with one residual connection in a moderate depth CNN.

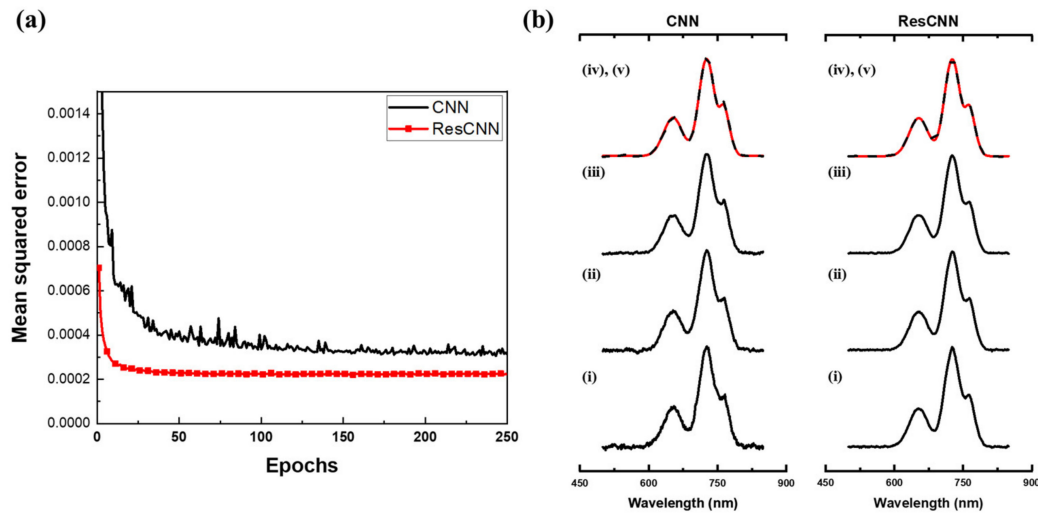


Figure 8. (a) Mean squared error of Gaussian dataset with respect to epochs. Solid black line denotes validation error of CNN, and solid red line with square symbols denotes validation error of ResCNN. (b) Reconstructions of a spectrum with respect to epochs where (i) to (iv) are epochs 1, 50, 150, and 250, respectively. Red line (v) denotes the original spectrum.

The reconstruction of an example spectrum with respect to the number of epochs is shown in Figure 8b. Black lines ((i) to (iv)) are the reconstructed spectra at 1, 50, 150, and 250 epochs, respectively. The solid red line (v) is the original spectrum, and the series of reconstructed spectrum for ResCNN show that the reconstruction converged earlier than CNN. The increased rate of convergence is because of the residual connection in ResCNN. Overall, the reconstruction performance of ResCNN is an improvement over CNN.

Note that both ResCNN and dictionary learning for sparse recovery require a training dataset and an optimization process to learn the spectral features. While this is a time-consuming process, remember that when using a learned dictionary to recover spectra, an iterative reconstruction algorithm is required, which needs additional time and incurs a high computational cost. The benefit of ResCNN is that it gives a reconstructed spectrum immediately once the training is completed.

7. Conclusions

In this paper, we propose a novel ResCNN for recovering the input spectrum from the compressed measurements in CS spectroscopy. As the optical structure for CS spectroscopy, we numerically generated multilayer thin-film filters which have a small mutual coherence. Therefore, we could compressively measure input spectra with unique sensing patterns. To reconstruct the input spectra from the compressively sampled measurements, we modeled ResCNN, which has a moderate-depth of learnable layers and a residual connection. We stacked nine learnable layers: five convolutional layers and four fully-connected layers with a single residual connection between the input and output of the learnable layers. The measurements were extended by a linear transformation and then fed into ResCNN. Finally, ResCNN reconstructed the input spectra. We demonstrated the empirical reconstruction results for ResCNN using synthetic and measured datasets. We compared the reconstruction performance of ResCNN with sparse recovery using five different sparsifying bases and CNN. Compared with sparse recovery methods, ResCNN shows better reconstruction performance without the *a priori* knowledge of either a sparsifying basis or any spectral features of the spectral datasets. On the other hand, the

sparse recovery methods show deviation of reconstruction performances to sparsifying bases and spectral datasets, meaning that a fixed sparsifying basis cannot represent all spectral features of input spectra. Furthermore, ResCNN shows stable reconstruction performances under noisy environments. Compared with CNN, ResCNN shows significant improvement in reconstruction performance and converges faster than CNN. In future work, we will explore compression approaches [40] and unrolled optimization approaches [43,44] for generating a sparsifying basis Φ from the training dataset to fully represent spectra without loss of spectral features.

Author Contributions: Conceptualization, C.K. and H.-N.L.; methodology, C.K.; software, C.K. and D.P.; formal analysis, C.K. and D.P.; investigation, C.K.; data curation, C.K.; writing—original draft preparation, C.K.; writing—review and editing, C.K., D.P. and H.-N.L.; project administration, H.-N.L.; funding acquisition, H.-N.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (NRF-2018R1A2A1A19018665).

Conflicts of Interest: The authors declare no conflict of interest.

References

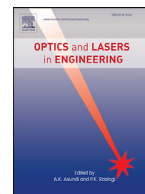
- Clark, R.N.; Roush, T.L. Reflectance spectroscopy: Quantitative analysis techniques for remote sensing applications. *J. Geophys. Res. Solid Earth* **1984**, *89*, 6329–6340. [[CrossRef](#)]
- Izake, E.L. Forensic and homeland security applications of modern portable Raman spectroscopy. *Forensic Sci. Int.* **2010**, *202*, 1–8. [[CrossRef](#)] [[PubMed](#)]
- Kim, S.; Cho, D.; Kim, J.; Kim, M.; Youn, S.; Jang, J.E.; Je, M.; Lee, D.H.; Lee, B.; Farkas, D.L.; et al. Smartphone-based multispectral imaging: System development and potential for mobile skin diagnosis. *Biomed. Opt. Express* **2016**, *7*, 5294–5307. [[CrossRef](#)] [[PubMed](#)]
- Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [[CrossRef](#)]
- Eldar, Y.C.; Kutyniok, G. *Compressed Sensing: Theory and Applications*; Cambridge University Press: Cambridge, UK, 2012.
- Candes, E.J.; Eldar, Y.C.; Needell, D.; Randall, P. Compressed sensing with coherent and redundant dictionaries. *Appl. Comput. Harm. Anal.* **2011**, *31*, 59–73. [[CrossRef](#)]
- Oiknine, Y.; August, I.; Blumberg, D.G.; Stern, A. Compressive sensing resonator spectroscopy. *Opt. Lett.* **2017**, *42*, 25–28. [[CrossRef](#)]
- Kurokawa, U.; Choi, B.I.; Chang, C.-C. Filter-based miniature spectrometers: Spectrum reconstruction using adaptive regularization. *IEEE Sens. J.* **2011**, *11*, 1556–1563. [[CrossRef](#)]
- Cerjan, B.; Halas, N.J. Toward a Nanophotonic Nose: A Compressive Sensing-Enhanced, Optoelectronic Mid-Infrared Spectrometer. *ACS Photonics* **2018**, *6*, 79–86. [[CrossRef](#)]
- Oiknine, Y.; August, I.; Stern, A. Multi-aperture snapshot compressive hyperspectral camera. *Opt. Lett.* **2018**, *43*, 5042–5045. [[CrossRef](#)]
- Kim, C.; Lee, W.-B.; Lee, S.K.; Lee, Y.T.; Lee, H.-N. Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy. *Opt. Lasers Eng.* **2019**, *115*, 53–58. [[CrossRef](#)]
- Oliver, J.; Lee, W.-B.; Lee, H.-N. Filters with random transmittance for improving resolution in filter-array-based spectrometers. *Opt. Express* **2013**, *21*, 3969–3989. [[CrossRef](#)] [[PubMed](#)]
- August, Y.; Stern, A. Compressive sensing spectrometry based on liquid crystal devices. *Opt. Lett.* **2013**, *38*, 4996–4999. [[CrossRef](#)] [[PubMed](#)]
- Huang, E.; Ma, Q.; Liu, Z. Etalon Array Reconstructive Spectrometry. *Sci. Rep.* **2017**, *7*, 40693. [[CrossRef](#)] [[PubMed](#)]
- Wang, Z.; Yu, Z. Spectral analysis based on compressive sensing in nanophotonic structures. *Opt. Express* **2014**, *22*, 25608–25614. [[CrossRef](#)] [[PubMed](#)]
- Wang, Z.; Yi, S.; Chen, A.; Zhou, M.; Luk, T.S.; James, A.; Nogan, J.; Ross, W.; Joe, G.; Shahsafi, A. Single-shot on-chip spectral sensors based on photonic crystal slabs. *Nat. Commun.* **2019**, *10*, 1020. [[CrossRef](#)]
- Pati, Y.C.; Rezaifar, R.; Krishnaprasad, P.S. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In Proceedings of the 27th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 1–3 November 1993; pp. 40–44.

18. Dai, W.; Milenkovic, O. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. Inf. Theory* **2009**, *55*, 2230–2249. [CrossRef]
19. Chen, S.S.; Donoho, D.L.; Saunders, M.A. Atomic decomposition by basis pursuit. *SIAM Rev.* **2001**, *43*, 129–159. [CrossRef]
20. Candes, E.; Tao, T. Decoding by linear programming. *arXiv* **2005**, arXiv:math/0502327. [CrossRef]
21. Oliver, J.; Lee, W.; Park, S.; Lee, H.-N. Improving resolution of miniature spectrometers by exploiting sparse nature of signals. *Opt. Express* **2012**, *20*, 2613–2625. [CrossRef]
22. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436. [CrossRef]
23. Mousavi, A.; Baraniuk, R.G. Learning to invert: Signal recovery via deep convolutional networks. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 2272–2276.
24. Li, Y.; Xue, Y.; Tian, L. Deep speckle correlation: A deep learning approach toward scalable imaging through scattering media. *Optica* **2018**, *5*, 1181–1190. [CrossRef]
25. Lee, D.; Yoo, J.; Ye, J.C. Deep residual learning for compressed sensing MRI. In Proceedings of the 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), Melbourne, VIC, Australia, 18–21 April 2017; pp. 15–18.
26. Mardani, M.; Gong, E.; Cheng, J.Y.; Vasanawala, S.S.; Zaharchuk, G.; Xing, L.; Pauly, J.M. Deep Generative Adversarial Neural Networks for Compressive Sensing MRI. *IEEE Trans. Med. Imaging* **2019**, *38*, 167–179. [CrossRef] [PubMed]
27. Jin, K.H.; McCann, M.T.; Froustey, E.; Unser, M. Deep convolutional neural network for inverse problems in imaging. *IEEE Trans. Image Process.* **2017**, *26*, 4509–4522. [CrossRef] [PubMed]
28. Kim, C.; Park, D.; Lee, H.-N. Convolutional neural networks for the reconstruction of spectra in compressive sensing spectrometers. In *Optical Data Science II*; International Society for Optics and Photonics: Bellingham, WA, USA, 2019; Volume 10937, p. 109370L.
29. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
30. Macleod, H.A. *Thin-Film Optical Filters*; CRC Press: Boca Raton, FL, USA, 2010.
31. Barry, J.R.; Kahn, J.M. Link design for nondirected wireless infrared communications. *Appl. Opt.* **1995**, *34*, 3764–3776. [CrossRef] [PubMed]
32. Kokaly, R.F.; Clark, R.N.; Swayze, G.A.; Livo, K.E.; Hoefen, T.M.; Pearson, N.C.; Wise, R.A.; Benzel, W.M.; Lowers, H.A.; Driscoll, R.L. *USGS Spectral Library Version 7 Data: US Geological Survey Data Release*; United States Geological Survey (USGS): Reston, VA, USA, 2017.
33. University of Eastern Finland. Spectral Color Research Group. Available online: <http://www.uef.fi/web/spectral/-spectral-database> (accessed on 2 August 2019).
34. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
35. Chen, G.; Needell, D. Compressed sensing and dictionary learning. *Finite Fram. Theory* **2016**, *73*, 201.
36. Engan, K.; Aase, S.O.; Husoy, J.H. Method of optimal directions for frame design. In Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258), Phoenix, AZ, USA, 15–19 March 1999; Volume 5, pp. 2443–2446.
37. Aharon, M.; Elad, M.; Bruckstein, A. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal. Process.* **2006**, *54*, 4311–4322. [CrossRef]
38. Mairal, J.; Bach, F.; Ponce, J.; Sapiro, G. Online dictionary learning for sparse coding. In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–17 June 2009; pp. 689–696.
39. Koh, K.; Kim, S.-J.; Boyd, S. An interior-point method for large-scale l1-regularized logistic regression. *J. Mach. Learn. Res.* **2007**, *8*, 1519–1555.
40. Martino, L.; Elvira, V. Compressed Monte Carlo for distributed Bayesian inference. *arXiv* **2018**, arXiv:1811.0505.
41. Liu, S.; Wu, H.; Huang, Y.; Yang, Y.; Jia, J. Accelerated Structure-Aware Sparse Bayesian Learning for 3D Electrical Impedance Tomography. *IEEE Trans. Ind. Inform.* **2019**. [CrossRef]
42. Tsiliogianni, E.; Deligiannis, N. Deep coupled-representation learning for sparse linear inverse problems with side information. *IEEE Signal. Process. Lett.* **2019**, *26*, 1768–1772. [CrossRef]

43. Diamond, S.; Sitzmann, V.; Heide, F.; Wetzstein, G. Unrolled optimization with deep priors. *arXiv* **2017**, arXiv:1705.08041.
44. Gilton, D.; Ongie, G.; Willett, R. Neumann Networks for Linear Inverse Problems in Imaging. *IEEE Trans. Comput. Imaging* **2019**. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).



Fabrication of 2D thin-film filter-array for compressive sensing spectroscopy

Cheolsun Kim, Woong-Bi Lee, Soo Kyung Lee, Yong Tak Lee, Heung-No Lee*

School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea

ARTICLE INFO

Keywords:

Spectroscopy
Thin films
Inverse problems
Compressive sensing

ABSTRACT

We demonstrate 2D filter-array compressive sensing spectroscopy based on thin-film technology and a compressive sensing reconstruction algorithm. To obtain different spectral modulations, we fabricate a set of multilayer filters using alternating low- and high-index materials and reconstruct the input spectrum using a small number of measurements. Experimental results show that the fabricated filter-array provides compatible spectral resolution performance with a conventional spectrometer in monochromatic lights and LEDs. In addition, the fabricated filter-array covers a wide range of wavelengths with a single exposure.

1. Introduction

The demand for spectrum information is increasing not only in research and development but also in the private sector. In response to this demand, researchers are trying to make spectrometers that are both small and inexpensive. These spectrometers could be used in various fields, such as medical systems, mobile applications, and remote sensing [1–3]. In particular, optical filter-based spectrometers do not need motorized or dispersive elements, and their filter-array can be directly attached to the detectors so that they can be easily miniaturized. However, there is a trade-off between size (for integrating filters) and spectral resolution with miniaturized spectrometers.

Over the years, numerous approaches to applying compressive sensing (CS) techniques have been proposed to reduce the size of spectrometers without reducing spectral resolution, or potentially even improving it. These approaches [4–7] include the following: band pass filters [4], random transmittance filters [5], photonic crystal slabs [6], and liquid crystal phase retarders [7]. Recently, Fabry–Perot (FP)-based CS spectroscopy methods have been presented [8,9]. To acquire differently modulated spectral measurements, a 2D array of FP resonators with different cavity depths has been tried [8] as well as a piezo-actuated device that changes the distance between two FP mirrors has been tried [9]. A hundred FP resonators are used to recover the input spectrum in [8], and the operational range of the piezo-actuator imposes mechanical limitations in [9].

The CS framework [10–12] is an efficient sampling and reconstruction scheme that requires fewer samples to reconstruct the signal than that required by conventional sampling. The CS framework can be applied to filter-based spectroscopy, offering the advantage of reducing

the number of filters and detectors required and allowing the system to be miniaturized.

In spectroscopy, the relation between the spectral components of the input light source $\mathbf{x} \in \mathbb{R}^{N \times 1}$ and the modulated signal $\mathbf{y} \in \mathbb{R}^{M \times 1}$ can be expressed as follows:

$$\mathbf{y} = \mathbf{T}\mathbf{x}, \quad (1)$$

where $\mathbf{T} \in \mathbb{R}^{M \times N}$ is the sensing matrix. Each row of the sensing matrix is to represent the transmission function (TF) of i -th filter, $T_m \in \mathbb{R}^{1 \times N}$ for $m = 1, 2, \dots, M$. In order to achieve miniaturization of spectroscope without degradation of spectral resolution, the CS framework is utilized in spectroscopy, where the number of filters is set to be smaller than the number of spectral components ($M < N$). Then, Eq. (1) becomes an underdetermined linear system. A sparse signal reconstruction algorithm with $L1$ norm minimization can be used to solve Eq. (1), if the input spectrum is either naturally sparse or can be sparsely represented in some basis $\Phi \in \mathbb{R}^{N \times N}$, i.e., $\mathbf{x} = \Phi\mathbf{s}$, where $\mathbf{s} \in \mathbb{R}^{N \times 1}$ is a sparse vector. Then, Eq. (1) becomes

$$\mathbf{y} = \mathbf{T}\Phi\mathbf{s} \quad (2)$$

The sparse signal \mathbf{s} can be estimated by solving the following $L1$ norm minimization problem:

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}} \|\mathbf{s}\|_1 \text{ subject to } \|\mathbf{y} - \mathbf{T}\Phi\mathbf{s}\|_2 \leq \epsilon \quad (3)$$

where ϵ is a small non-negative constant. The reconstructed input spectrum $\hat{\mathbf{x}}$ is then $\Phi\hat{\mathbf{s}}$.

In this paper, we demonstrate 2D filter-array CS spectroscopy. This uses a multilayer thin-film filter-array for spectral modulation, where each filter modulates the input spectrum using different sensing patterns. A CMOS image camera reads out the modulated signals with a

* Corresponding author.

E-mail address: heungno@gist.ac.kr (H.-N. Lee).

Table 1
Recursion for calculating reflection coefficients.

Input: $\lambda, \theta_1 = 0, \mathbf{n} = \{n_1, n_2, \dots, n_{l-1}, n_l\}, \mathbf{d} = \{d_2, d_3, \dots, d_{l-1}, d_l\}.$
Step 1: Obtain $\theta_k, \beta_k,$ and N_k $\theta_k = \sin^{-1} \left(\frac{n_{k-1}}{n_k} \sin \theta_{k-1} \right),$ for $k = 2, 3, \dots, l.$ $\beta_k = 2\pi \cos(\theta_k) n_k d_k / \lambda,$ for $k = 2, 3, \dots, l.$ $N_k = \begin{cases} n_k / \cos \theta_k & \text{for TE} \\ n_k \cos \theta_k & \text{for TM} \end{cases},$ for $k = 1, 2, \dots, l.$
Step 2: Set $\eta_l = N_l$
Step 3: Obtain η_2 Decrement k by 1 from $l - 1$ to 2 $\eta_k = N_k \frac{\eta_{k+1} \cos \beta_k + j N_{k+1} \sin \beta_k}{N_{k+1} \cos \beta_k + j \eta_{k+1} \sin \beta_k}$
return η_2
Step 4: Compute $\rho = (N_1 - \eta_2) / (N_1 + \eta_2).$
Output: ρ

single exposure, and then a reconstruction algorithm is applied that depends on the modulated signals and the sensing matrix, allowing the input spectrum to be recovered.

The research focus has been given to fabrication of the multilayer thin-film filters for actual CS spectroscopy implementation and verification experiments. For fabricating as a 2D filter-array, we use commonly available materials SiNx and SiO2 for high and low refractive index materials which are deposited alternately on the substrate with varying thicknesses. Furthermore, we come up with a practical way that set of filters can be deposited on a single substrate with different thicknesses of layers.

2. 2D filter-array

2.1. Multilayer thin-film filter

Thin films are a basic component that have been applied in a variety of areas, including semiconductor devices, optical coatings, and solar cells [13]. The theoretical TF of a multilayer thin-film filter is given by [14]

$$T(\lambda, \theta_1) = 1 - \frac{1}{2} \left(|\rho_{TE}|^2 + |\rho_{TM}|^2 \right), \quad (4)$$

where ρ_{TE} and ρ_{TM} are the reflection coefficients. Given a wavelength λ and the incident angle θ_1 , TF can be calculated using recursive routines shown in Table 1.

In Table 1, given the input of a wavelength λ , a vector of l refractive indices $\mathbf{n} = (n_1, n_2, \dots, n_{l-1}, n_l)$ and a vector of $l - 1$ layer thicknesses $\mathbf{d} = (d_2, d_3, \dots, d_{l-1}, d_l)$, a reflection coefficient ρ is generated. Note that there are l layers considered in total. The first one is the layer of the air and the last one is the layer of the substrate. The light is assumed to be arriving from the air to the second layer in normal incidence. The first index n_1 in the vector \mathbf{n} represents the refractive index of the air. The last one n_l in the vector \mathbf{n} represents the refractive index of the substrate. The refractive indices of the intermediate thin-film layers are denoted by n_2 to n_{l-1} . The thickness of the air does not need to be considered. The thickness of the substrate is denoted by d_l .

The thicknesses of the intermediate thin-film layers are denoted by d_2 to d_{l-1} . The incidence angle of the light passing from the k th to the $k + 1$ th layer is θ_k , and η_k is the effective complex-valued index of the k th layer. A TF for a single filter is obtained by considering all wavelengths in the range of interest. An array of TFs for the M filters can be obtained by repeating this process where each filter $T_m \in \mathbb{R}^{1 \times N}$ for $m = 1, 2, \dots, M$ in Eq. (1) is generated from a unique set of refractive index and thickness vectors.

2.2. Numerical design of 2D filter-array

To implement the proposed 2D filter-array, we numerically modeled the proposed spectroscopy method with reference to [14–16], and according to the following steps. (i) Generate the reference vector of layer

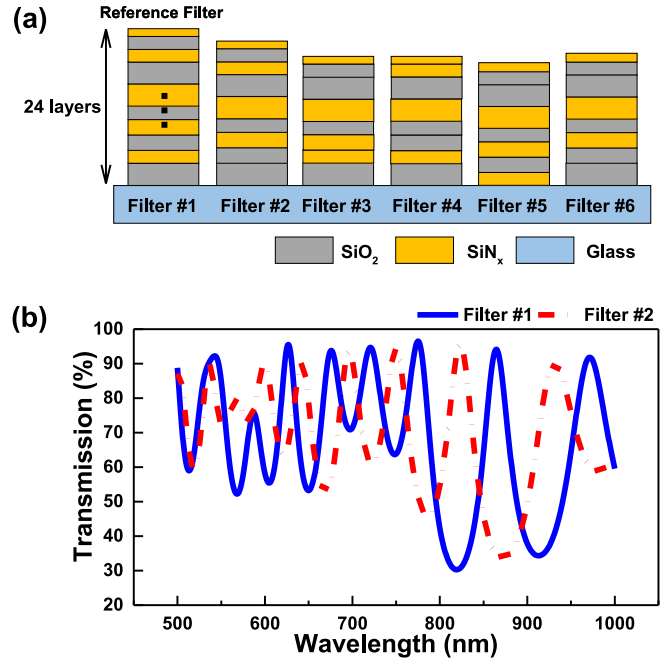


Fig. 1. (a) Schematic of the thin-film filter-array. (b) Example of two transmission functions for thin-film filters.

thicknesses, i.e. $\mathbf{d} = (d_2, d_3, \dots, d_{l-1}, d_l)$, for the reference filter. (ii) Generate a vector of thicknesses for the other filter by randomly removing one to five layer thicknesses from the reference vector. (iii) Repeat the step (ii) 35 times to create a total of 36 vectors of thicknesses. (iv) Use the recursion Table 1 and Eq. (4) to calculate the TFs for a new filter-array (sensing matrix). (v) Use the mutual coherence μ to quantify the goodness of the sensing matrix of the designed filter-array. Mutual coherence μ is defined as $\mu \triangleq \max_{i,j} |o_{ij}|$, where o_{ij} is the (i, j) th off-diagonal element of the Gram matrix, $\mathbf{T}^* \mathbf{T} \in \mathbb{R}^{N \times N}$. \mathbf{T}^* denotes the conjugate transpose of \mathbf{T} . With these steps, we can generate a single set of 36 filters. By repeating these steps, multiple sets of 36 filters can be obtained. Among these sets of filter-arrays, the set of filters with a smallest mutual coherence is selected.

In CS framework, a sensing matrix with a smaller mutual coherence is better than the one with a higher mutual coherence to capture the information of input signal to be reconstructed [5,17]. A schematic of the proposed filter-array is shown in Fig. 1(a). Each time a layer is removed, the layers above and those below come together to form a single layer with two thicknesses added up. We consider two materials, SiNx and SiO2 for the high- and low-refractive index materials with refractive indices of 2.02 for SiNx and 1.45 for SiO2. The thickness range of each layer is from 50 to 150 nm. Through the numerical design, we empirically found that removal of up to five layers from the 24-layer reference filter was possible to create a 6×6 filter-array with a low coherence.

Fig. 1(b) shows the TFs for two designed filters as examples. In conventional spectroscopy, the TFs with a large spectral depth and a narrow spectral peak are preferred in order to prevent interference among measurements. In compressive sensing spectroscopy, Each TF of the filter should be wide enough so that the set of the small number of filters fully senses the spectral information in the given wavelength range [9].

Each filter shows several spectral peaks and rapid changes of transmission value with respect to wavelength. Therefore, each filter has a high optical throughput that the energy (intensity) which passes through the filter is higher than that with the conventional bandpass filter approach. In addition, fewer filters can be used to cover the entire wavelength range with the proposed method. For example, suppose 250 bandpass filters are used to cover the wavelength range from 500 to

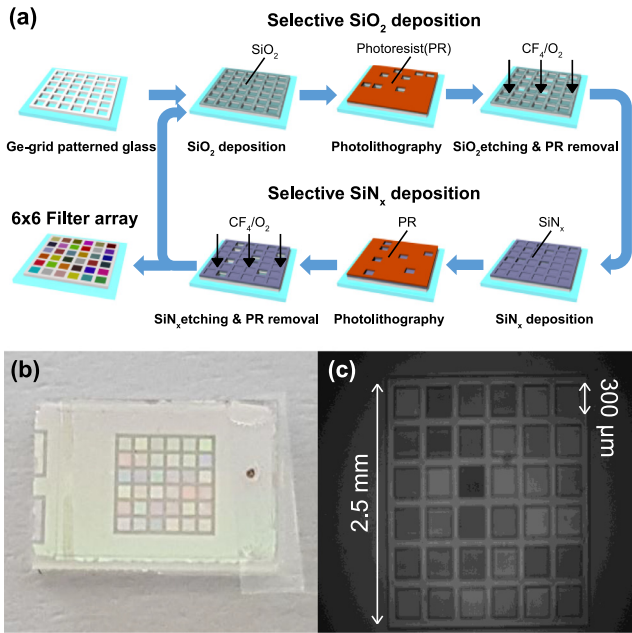


Fig. 2. (a) Schematic of the thin-film filter-array fabrication process. (b) Photograph of a fabricated thin-film filter-array. (c) Monochrome image of the thin-film filter-array taken at a wavelength of 700 nm.

1000 nm. Then, the bandwidth of TF is 2 nm, according to the conventional bandpass filter design. In the proposed approach, the same range of wavelength can be covered with only 36 proposed filters, subject to the use of a recovery algorithm present at the reconstruction end.

2.3. Filter-array fabrication

Fig. 2(a) shows the process in which a thin-film filter-array is fabricated. This comprises two main parts; one is SiO₂ film deposition and the other is SiN_x film deposition according to the specified thicknesses. Prior to depositing an SiO₂ film, a 6 × 6 germanium (Ge) grid with elements of size 300 μm and spacing 100 μm was formed on the glass using an e-beam evaporator to separate the filters. In this grid, SiO₂ and SiN_x layers were deposited with the width of 300 μm in each filter. Then, selective deposition was done as follow: An intentionally thick SiO₂ film was deposited on the glass patterned with the Ge grid using plasma-enhanced chemical vapor deposition. The regions where the film should not be deposited were then removed by conventional photolithography, namely CF₄/O₂ reactive ion etching. The process pressure and radio-frequency power were maintained at 50 mTorr and 50 W, respectively. The SiN_x film deposition process was performed in the same manner as for SiO₂. Finally, these two main steps, SiO₂ and SiN_x film deposition, were repeated 12 times each to lay down 24 layers. Fig. 2(b) and (c) show a photograph of a fabricated thin-film filter-array and a monochrome image of the filter-array, respectively. Each filter is composed of a different number of layers each with different thicknesses; therefore, each one has unique color due to its different TF, as shown in Fig. 1(b).

3. Experiments

3.1. Experimental setup

Optical setups for experimental verification of the proposed spectroscopy system are shown in Fig. 3. Fig. 3(a) depicts the optical setup for measuring TFs of a filter-array. The setup for testing the performance of the proposed system is shown in Fig. 3(b). The photographs of the optical setup and the CMOS image camera with the thin-film filter-array

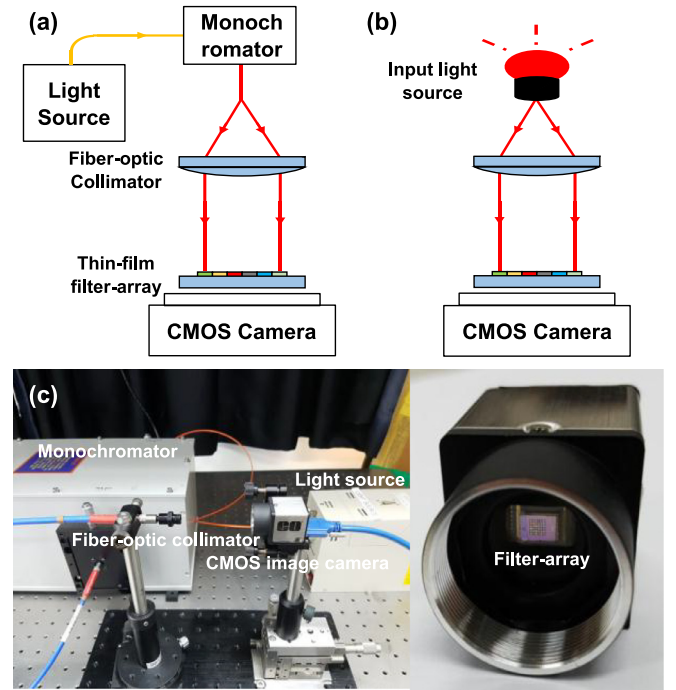


Fig. 3. (a) Schematic of the optical setup for measuring the sensing matrix. (b) Schematic of the optical setup for testing the performance of the proposed spectroscopy system. (c) Photographs of the optical setup and the CMOS image camera with the thin-film filter-array.

are shown in Fig. 3(c). During the optical experiments, we set the incident angle to filter-array as normal incidence. Using a linear stage, a rotational stage and optical mounting posts, we aligned the optical fiber with the CMOS image camera (E0-1312, Edmund Optics) for the normal incidence.

In Fig. 3(a), a halogen lamp (KLS-150H-LS-150D, Kwangwoo) was used to provide a continuous light spectrum. It was put into a monochromator (MMAC-200, Mi Optics) to produce a specific narrow wavelength band. Then, a fiber-optic collimator was used to form a beam of parallel light. The beam was fed into the CMOS image camera through the fabricated thin-film filter-array. With a single exposure, each filter modulated the light in a different pattern. The modulated light was read out by pixels of CMOS image camera, yielding $M = 36$ distinct output signals \mathbf{y} in Eq. (1). Each output signal was taken by summing up the modulated values of the pixels underneath the pertinent filter.

To apply CS reconstruction algorithms to the proposed system, the sensing matrix \mathbf{T} must be pre-determined. Let us denote the intensity which passes through the filter-array as $IF(m, \lambda)$ and the intensity without the filter-array as $IWF(m, \lambda)$, where m is the filter index and λ is the wavelength. The sensing matrix is then given by

$$T(m, \lambda) = \frac{IF(m, \lambda) - BI(m, \lambda)}{IWF(m, \lambda) - BI(m, \lambda)}, \quad (5)$$

where $BI(m, \lambda)$ is the background intensity. We took 500 wavelength samples, spaced 1 nm apart, in the range from 500 to 1000 nm. The measured sensing matrix $\mathbf{T} \in \mathbb{R}^{36 \times 500}$ obtained from the fabricated thin-film filter-array is shown in Fig. 4. Each TF of the filters, a row of the color map, is shown as a combination of colors, i.e., red (high transmission value) and blue (low transmission value). Different TFs show different places of high and low transmission values indicating mutual uncorrelation. As a set of 36 filters, the filter-array covers the entire wavelength range with high optical throughput.

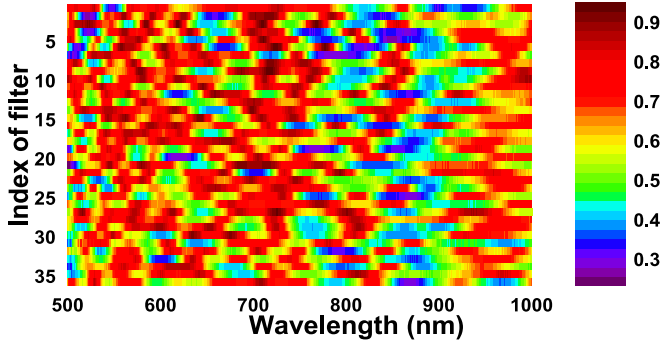


Fig. 4. Color map of the measured sensing matrix for the thin-film filter-array. Each row represents the TF of a filter with respect to wavelength.

3.2. Computational experiments

To quantify the performance and explore the two-point resolution of the fabricated filter-array, we conducted computational experiments. The two-point resolution is the ability to distinguish the spectral peaks which are closely spaced. For the experiments, we generated mono-peak spectra and two-peak spectra as input spectra using the Gaussian function. A generated input spectrum \mathbf{x} was numerically modulated by multiplying the measured sensing matrix \mathbf{T} as shown in Eq. (1). Then, using the M -modulated signals (measurements) and the sensing matrix \mathbf{T} , a reconstruction algorithm is used to recover the input spectrum. In the experiments, we considered that the input spectrum was a directly sparse signal. The mean-squared error (MSE) between the input spectrum \mathbf{x} and the reconstructed spectrum $\hat{\mathbf{x}}$ was calculated. The MSE is defined as $\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 / N$.

We firstly tested the spectral reconstruction performance of the fabricated filter-array with changing the full width at half maximum (FWHM) of the generated input signals. We made three noisy environments by adding the additive noise \mathbf{n} to Eq. (1) as $\mathbf{y} = \mathbf{T}\mathbf{x} + \mathbf{n}$ whose the signal to noise ratios (SNRs) were 20, 25, 30 dB. The SNR in decibels is defined as $10 \cdot \log_{10}(\|\mathbf{x}\|_2^2 / N\sigma^2)$, where σ is the standard deviation of the noise.

The spectral reconstruction performances with respect to the FWHMs are shown in Fig. 5(a). For the two-peak spectrum, the distance between two peaks was determined as $[1.5 \cdot \text{FWHM}]$, where $[\cdot]$ is the nearest integer function. We averaged all the MSEs of the spectrum over the peak-locations from 500 to 999 nm in a given FWHM. As shown in Fig. 5(a), the mono-peak spectrum is reconstructed better than two-peak spectrum. As the FWHM increased, the performance of spectral reconstruction is degraded. This is due to the increased sparsity of the spectrum.

Second, we verified the stability of noise along the SNR conditions for the fabricated filter-array. As shown in Fig. 5(b), the reconstruction performance on mono-peak spectrum is better than that of the two-peak spectrum. In addition, when the FWHM is 1 nm, the reconstruction performance is better than the FWHM with 2 nm. Despite the additive noise, the results show that the fabricated filter-array is robust to the noisy environments.

As depicted in Fig. 5, the reconstruction performance of the fabricated filter-array depends on the FWHM and the SNR. For the two-point resolution, the MSE has the smallest value when the FWHM of the two-peak spectrum is 1 nm. The overall MSEs are small enough to use the fabricated filter-array to conduct the optical experiments.

3.3. Optical experiments

Optical experiments were then conducted to evaluate the performance of the proposed system, as shown in Fig. 3(b). Narrow-band monochromatic lights and LEDs were used as input light sources. To gen-

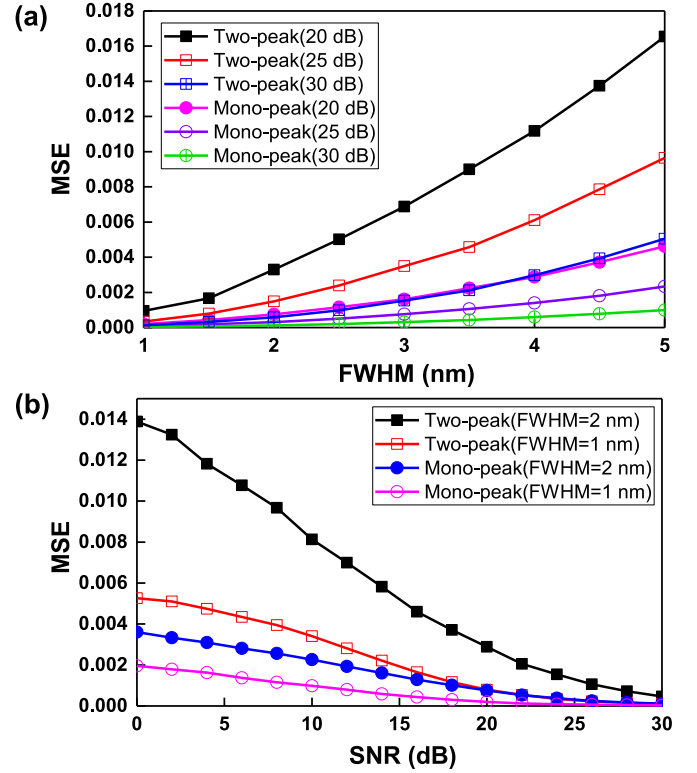


Fig. 5. (a) Computational reconstruction performance of the fabricated thin-film filter-array with respect to the FWHM. (b) Computational spectral reconstruction performance of the fabricated thin-film filter-array with respect to the SNR.

erate narrow-band light, a supercontinuum white light source (SuperK COMPACT, NKT Photonics) was placed in the monochromator, making a narrow band of light with a full width at half maximum (FWHM) of approximately 1 nm. These light sources were fed into the CMOS image camera through the filter-array, simultaneously capturing the M differently modulated signals. The M -modulated signals and the measured sensing matrix \mathbf{T} were then used to solve Eq. (3). We used a Gaussian kernel matrix as the sparsifying basis Φ . The spectral waveform can be represented as a linear combination of Gaussian kernels, and a Gaussian kernel can be easily generated with two parameters, namely the peak location and the FWHM value [4,18]. The *l1* *l*_s *noneg* algorithm [19] was used as a reconstruction algorithm to solve Eq. (3) with non-negativity constraints.

Fig. 6 shows the reconstruction results for monochromatic lights and LEDs. For comparison, the reference spectrum and the reconstructed spectrum were normalized to the range between zero and one.

The optical experimental results for monochromatic lights are shown in Fig. 6(a). In our optical experiment, depicted in Fig. 3(c), we use four different monochromatic spectra, with spectral peaks located at 600, 700, 800, and 900 nm, respectively. The reference spectra are measured using an optical spectrum analyzer (AQ-6315B, Ando) which indicate actual spectral peak locations at 598.7, 700.4, 800.5, and 900.4 nm, respectively. Using the fabricated filter-array CS spectroscopy with the reconstruction algorithm, the spectral peak locations are reconstructed at 599, 699, 799, and 901 nm, respectively. The mean FWHM of the reference spectra is approximately 1 nm, and the mean FWHM for the reconstructed spectra is approximately 1.4 nm.

Fig. 6(b) shows the spectral reconstructions of green (527 nm) and red (635 nm) LEDs. For the reference spectra, we measure the LEDs using a grating spectrometer (QE65000, Ocean Optics). The spectral peak locations for the reference LEDs are 527.6 nm (green LED) and 634.9 nm

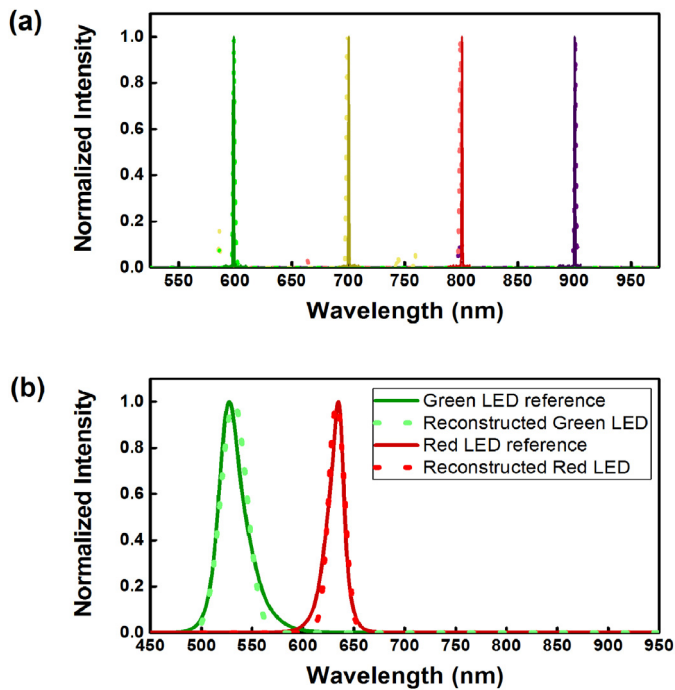


Fig. 6. Spectral reconstructions of several different input light sources. (a) Spectral reconstructions of monochromatic lights (dots) compared with reference spectra (solid lines): 600 nm (green), 700 nm (yellow), 800 nm (red), and 900 nm (purple). (b) Spectral reconstructions of LEDs (dots) compared with reference spectra (solid lines): green LED (527 nm), and red LED (635 nm).

(red LED), and the reconstructed spectral peak locations are 531 nm (green LED) and 633 nm (red LED). The peak signal-to-noise ratios are 28.3 dB (green LED) and 31.7 dB (red LED).

Discussing Fig. 6, the spectra of reconstructed monochromatic lights show several negligible spikes. This is probably due to background noise in the optical experiments. But overall, the reconstruction results of the proposed CS spectroscopy system for monochromatic lights and LEDs are similar to those of the grating spectrometer. Furthermore, the number of modulated signals is significantly small ($M=36$) that the measurement to wavelength sample ratio is 36:500 (ratio between M and N).

To further explore the performance of the proposed CS spectroscopy, we conducted the computational experiment on the fabricated filter-array using a continuous light source, halogen lamp. For the experiment, we used the measured sensing matrix T . The conventionally measured spectrum of the halogen lamp was used as the input spectrum x . The modulated signals were generated by numerically multiplying the sensing matrix and the input spectrum. By solving Eq. (3), we reconstructed the continuous spectrum of light. In Fig. 7, we present computational spectral reconstruction of the halogen lamp. The peak signal-to-noise ratio is 43.8 dB. Due to the limitations of our optical components to reject the spectrum of the halogen lamp except for the wavelength range from 500 to 1000 nm, we could not perform the optical experiment on the continuous source. However, the computational reconstruction result of the halogen lamp indicates that the fabricated filter-array can be utilized for recovering the various kinds of spectra in the given wavelength range without limitations of the optical components.

Fabricating the proposed filter-array can be more difficult than fabricating Fabry–Perot structure due to the large number of layers for the proposed filter-array. However, the proposed spectroscopy is compact and it does not need motorized components which were used with the Fabry–Perot structure [9]. In addition, thanks to the 2D array structure, the proposed spectroscopy captures all measurements in a single exposure. But the Fabry–Perot spectroscopy [9] required a number of

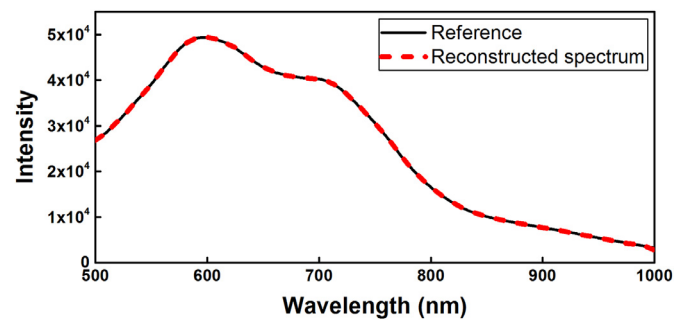


Fig. 7. Computational spectral reconstruction of a halogen lamp (red dash line) compared with the reference spectrum (black solid line) measured by a conventional spectrometer.

exposures as many times as the number of measurements. Compared to Fabry–Perot spectroscopy [8], the proposed spectroscopy utilizes 36 filters to cover the wavelength range from 500 to 1000 nm, but 100 filters were used in [8] to cover the range from 500 to 750 nm.

4. Conclusion

We have demonstrated a 2D array CS spectroscopy based on thin-film technology. A 2D thin-film filter-array is fabricated based on array processing. Using the fabricated filter-array, measurements are obtained to which the CS reconstruction algorithm is applied. Finally, demonstration of input spectrum reconstruction is successfully made. The proposed system is compact, portable, and obtains the necessary measurements in a single exposure thanks to its structural advantages. Moreover, it works over a wide spectral range, from the visible light region to the near-infrared region. Compared with conventional spectrometers (non-CS spectrometers), the proposed system has a high optical throughput and compatible spectral resolution performance in monochromatic lights and LEDs with significantly less number of measurements.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (NRF-2018R1A2A1A19018665).

References

- [1] Bacon CP, Mattley Y, DePrece R. Miniature spectroscopic instrumentation: applications to biology and chemistry. *Rev Sci Instrum* 2004;75:1–16.
- [2] Kim S, Cho D, Kim J, Kim M, Youn S, Jang JE, et al. Smartphone based multispectral imaging: system development and potential for mobile skin diagnosis. *Biomed Opt Express* 2016;7:5294–307.
- [3] Clark RN, Roush TL. Reflectance spectroscopy: quantitative analysis techniques for remote sensing applications. *J Geophys Res Solid Earth* 1984;89:6329–40.
- [4] Oliver J, Lee W, Park S, Lee H-N. Improving resolution of miniature spectrometers by exploiting sparse nature of signals. *Opt Express* 2012;20:2613–25.
- [5] Oliver J, Lee W-B, Lee H-N. Filters with random transmittance for improving resolution in filter-array-based spectrometers. *Opt Express* 2013;21:3969–89.
- [6] Wang Z, Yu Z. Spectral analysis based on compressive sensing in nanophotonic structures. *Opt Express* 2014;22:25608–14.
- [7] August Y, Stern A. Compressive sensing spectrometry based on liquid crystal devices. *Opt Lett* 2013;38:4996–9.
- [8] Huang E, Ma Q, Liu Z. Etalon array reconstructive spectrometry. *Sci Rep* 2017;7.
- [9] Oiknine Y, August I, Blumberg DG, Stern A. Compressive sensing resonator spectroscopy. *Opt Lett* 2017;42:25–8.
- [10] Donoho DL. Compressed sensing. *IEEE Trans Inf Theory* 2006;52:1289–306.
- [11] Baraniuk RG. Compressive sensing [lecture notes]. *IEEE Signal Process Mag* 2007;24:118–21.
- [12] Qaisar S, Bilal RM, Iqbal W, Naureen M, Lee S. Compressive sensing: from theory to applications, a survey. *J Commun Netw* 2013;15:443–56.

- [13] Macleod HA. Thin-film optical filters. CRC press; 2001.
- [14] Barry JR, Kahn JM. Link design for nondirected wireless infrared communications. *Appl Opt* 1995;34:3764–76.
- [15] Pedrotti FL, Pedrotti LS. Introduction to optics. 2nd ed. Prentice Hall; 1993.
- [16] Topasna DM, Topasna GA. Numerical modeling of thin film optical filters. *Proc. SPIE* 2009;9666:96661P.
- [17] Candes EJ, Eldar YC, Needell D, Randall P. Compressed sensing with coherent and redundant dictionaries. *Appl Comput Harmon Anal* 2011;31:59–73.
- [18] Kurokawa U, Choi BI, Chang C-C. Filter-based miniature spectrometers: spectrum reconstruction using adaptive regularization. *IEEE Sens J* 2011;11:1556–63.
- [19] Koh K, Kim S-J, Boyd S. An interior-point method for large-scale l_1 -regularized logistic regression. *J Mach Learn Res* 2007;8:1519–55.

Filters with random transmittance for improving resolution in filter-array-based spectrometers

J. Oliver, Woong-Bi Lee, and Heung-No Lee*

School of Information and Communications, Gwangju Institute of Science and Technology, South Korea
*heungno@gist.ac.kr

Abstract: In this paper, we introduce a method for improving the resolution of miniature spectrometers. Our method is based on using filters with random transmittance. Such filters sense fine details of an input signal spectrum, which, when combined with a signal processing algorithm, aid in improving resolution. We also propose an approach for designing filters with random transmittance using optical thin-film technology. We demonstrate that the improvement in resolution is 7-fold when using the filters with random transmittance over what was achieved in our previous work.

©2013 Optical Society of America

OCIS codes: (300.6320) Spectroscopy, high-resolution; (120.6200) Spectrometers and spectroscopic instrumentation; (100.6640) Super-resolution.

References and links

1. D. J. Brady, *Optical Imaging and Spectroscopy* (John and Wiley Sons, 2009).
2. S. W. Wang, C. Xia, X. Chen, W. Lu, M. Li, H. Wang, W. Zheng, and T. Zhang, "Concept of a high-resolution miniature spectrometer using an integrated filter array," *Opt. Lett.* **32**(6), 632–634 (2007).
3. W. L. Wolfe, *Introduction to Imaging Spectrometers* (SPIE, 1997).
4. H. N. Lee, *Introduction to Compressed Sensing* (Lecture notes; Spring Semester, GIST, Korea, 2011). http://inonet.gist.ac.kr/?page_id=843
5. J. Oliver, W. B. Lee, S. J. Park, and H. N. Lee, "Improving resolution of miniature spectrometers by exploiting sparse nature of signals," *Opt. Express* **20**(3), 2613–2625 (2012).
6. C. C. Chang, N. T. Lin, U. Kurokawa, and B. I. I. Choi, "Spectrum reconstruction for filter-array spectrum sensor from sparse template selection," *Opt. Eng.* **50**(11), 114402 (2011).
7. C. C. Chang and H. N. Lee, "On the estimation of target spectrum for filter-array based spectrometer," *Opt. Express* **16**(2), 1056–1061 (2008).
8. U. Kurokawa, B. I. Choi, and C.-C. Chang, "Filter-based miniature spectrometers: spectrum reconstruction using adaptive regularization," *IEEE Sens. J.* **11**(7), 1556–1563 (2011).
9. C. Bendjaballah, "Information rates in optical channels," *Opt. Commun.* **17**(1), 55–58 (1976).
10. Y. Aizu, K. Ogino, and T. Asakura, "A laser velocimeter using a random pattern," *Opt. Commun.* **64**(3), 205–210 (1987).
11. J. Ojeda-Castañeda and A. Saucedo, "Random gratings as correlator sensors," *Opt. Lett.* **22**(5), 257–258 (1997).
12. D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory* **52**, 1289–1306 (2006).
13. R. Baraniuk, "Compressive sensing," *IEEE Sig. Proc. Mag.* **24**(4), 118–121 (2007).
14. E. Candes and J. Romberg, "11-magic: Recovery of sparse signals via convex programming," Technical report (2005). <http://users.ece.gatech.edu/~justin/11magic/>
15. S. Park and H. N. Lee, "Designing an algorithm to solve basis pursuit denoising with a nonnegative constraint," *IEEE Sig. Proc. Letters*. (submitted to).
16. R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Stat. Soc., B* **58**, 267–288 (1996).
17. A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Ima. Sciences* **2**(1), 183–202 (2009).
18. A. Juditsky and A. Nemirovski, "First Order Methods for Nonsmooth Convex Large-Scale Optimization, I: General Purpose Methods," in *Optimization for Machine Learning*, S. Sra, S. Nowozin, and S.J. W. Write, eds. (MIT Press, 2011), pp. 1–28.
19. S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Founda. and Tren. Mach. Learn.* **3**(1), 1–122 (2010).
20. M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Sig. Proc. Mag.* **25**(2), 83–91 (2008).
21. R. Dikpal, A. Veeraraghavan, and R. Chellappa, "P2C2: Programmable pixel compressive camera for high speed imaging," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (IEEE, 2011)*, pp. 329–336.

22. C. Li, T. Sun, K. Kelly, and Y. Zhang, "A compressive sensing and unmixing scheme for hyperspectral data processing," Technical report. (<http://www.caam.rice.edu/~zhang/reports/tr1101.pdf>).
23. A. Rajwade, D. Kittle, T.-H. Tsai, D. Brady, and L. Carin, "Coded hyperspectral imaging and blind compressive sensing," submitted (2012).
24. K. Madanipour and M. T. Tavassoly, "Determination of modulation transfer function of a printer by measuring the autocorrelation of the transmission function of a printed Ronchi grating," *Appl. Opt.* **48**(4), 725–729 (2009).
25. J. R. Barry and J. M. Kahn, "Link design for non-directed wireless infrared communications," *Appl. Opt.* **34**(19), 3764–3776 (1995).
26. H. A. Macleod, *Thin-Film Optical Filters* (Institute of Physics Publishing, 2002).
27. Z. B. Haim, Y. C. Eldar, and M. Elad, "Coherence-based performance guarantees for estimating a sparse vector under random noise," *IEEE Trans Sig. Proc.* **58**, 5030–5043 (2010).
28. C. Z. Microscopy, "Fundamentals of mercury arc lamps," <http://zeiss-campus.magnet.fsu.edu/articles/lightsources/mercuryarc.html>.

1. Introduction

Miniature spectrometers are key instruments that are required in various academic and industrial applications such as bio-medical, chemical, and environmental engineering [1]. These spectrometers, built with integrated-filter arrays, provide superior portability, flexibility, and cost-effectiveness [2]. The spectrometers also provide fine details about the various spectral components of the incident light. These spectral components reveal a wealth of information concerning the composition and the structure of the various objects being observed [3]. The ability of a spectrometer to reveal fine information is usually attributed to its resolution. Thus, in recent years, researchers have focused on improving the resolution of spectrometers.

State-of-the-art filter-array-based spectrometers rely on digital signal processing (DSP) algorithms. These algorithms may run on a general-purpose computing platform of cellular phones and laptop computers, as shown in Fig. 1, before the spectrum estimate is displayed on the screen. They can also run on a specialized DSP chip integrated into the spectrometer unit in the case of stand-alone spectrometers. These algorithms process the raw spectrum acquired by the spectrometer to reduce noise and distortions and thereby aid in improving the resolution.

In the filter-array-based spectrometers, the factors that limit the resolution are the number of filters in the filter array and the shapes of the transmittance functions of these filters [4]. In practical miniature spectrometers, the number of filters is fixed. Thus, in order to improve the resolution, it is often not a realistic option to increase the number of filters. The second factor that affects the resolution is the transmittance functions of the filters. While the transmittance function (TF) is usually designed to meet certain criteria, in practice, owing to low-cost integrated-array fabrication, the TF of each of the filters is non-ideal. These non-ideal filters highly distort the raw spectrum and thereby necessitate the digital processing of the raw spectrum to determine information concerning the spectral components of the input signal spectrum. DSP algorithms or estimators have been reported in the literature [5–8]. The algorithms in [5, 6] are centered on L_1 -norm minimization, which is a modern optimization tool in DSP for solving underdetermined system of linear equations, whereas the algorithms in [7, 8] are based on conventional least squares and are designed with an aim to reconstruct the signal spectrum but not to improve resolution.

In this paper, we assume the use of the L_1 -norm-minimization-based DSP algorithm in [5] as the spectral estimator and aim to address the following relevant questions: Is it possible to enhance the resolution of the spectrometers by shaping the TFs in a particular way? Namely, is there a certain shape of the TF that is suitable for use with the L_1 -norm-minimization algorithm at the back end? If yes, what is that shape, and how can such a TF be designed? Is there a practical way to implement such a TF? Also of great importance is the discussion of suitable performance metrics to compare different TFs. This is necessary to reflect the use of the L_1 -norm-based spectral estimator we aim to use. With the new metric, one should be able to investigate the maximum possible resolution achieved by employing different TFs.

As per the discussion of the questions above, the role of TFs in improving the resolution of the spectrometers under the DSP-based spectral estimator needs to be

carefully examined. We observe that the TFs whose auto-covariance functions resemble that of a Dirac delta function acquire independent information about the closely spaced spectral components of the input signal spectrum under the DSP-based spectral estimator. Filters with random-TFs possess such auto-covariance. Based on our study, we show that it is possible to design filters with random-TFs using thin-film technology and improve the resolution up to 7-fold compared to the filters with non-ideal transmittances in [5].

The concept of random-TFs goes back to the 1970s. Different aspects of the random-TFs have been studied in the past [9–11]. In [9], the author has studied the maximum information transmission rate of an optical filter with random transmittance. A laser-based velocimeter is designed in [10] based on random transmittance patterns. In [11], the authors have designed a holographic-based sensor using random transmittances that are generated by using random gratings. As per our knowledge, however, no one has designed or reported filters with random transmittance for improving the resolution of a spectrometer.

This paper is organized as follows: Section 2 presents the system model, and Section 3 discusses in detail the need for designing filters with random transmittances for improving resolution. Section 4 discusses the design of random-transmittance filters using thin-film technology, and Section 5 presents the concept of the resolution of a spectrometer. Experimental results are discussed in Section 6, and Section 7 concludes the paper.

2. System description

We consider a spectrometer that consists of a planar filter array with M filters arranged in a 2D fashion as shown in Fig. 1. Each of these filters acts as a wavelength-selective device. Each filter is specified in terms of a transmittance function (TF), $T_i(\lambda)$, $i = 1, 2, \dots, M$, which indicates the fraction of input light that the filter allows at each unit of wavelength. We note that $T_i(\lambda)$ is a continuous (analog) function of wavelength. Each filter is attached to a CCD element that converts light into electrical energy. The output of each of the CCD elements is sampled to form an $M \times 1$ vector \mathbf{y} , which is called a raw spectrum.

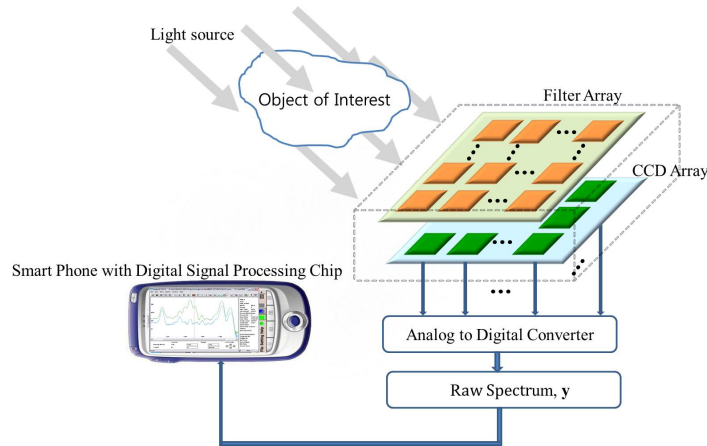


Fig. 1. Schematic of the proposed filter-array-based spectrometer.

Each sample of the raw spectrum denotes the projection (inner product) of the input signal spectrum onto a TF. That is, the i th sample of the raw spectrum is obtained as

$$y_i = \int x(\lambda) T_i(\lambda) d\lambda + w_i, \quad i = 1, 2, \dots, M, \quad (1)$$

where $x(\lambda)$ is the input signal spectrum and w_i denotes a Gaussian noise sample with zero-mean and variance σ^2 . Thus, each raw spectrum sample contains information about

the input signal spectrum acquired by a single filter in the filter array. We observe that the model in Eq. (1) for the raw spectrum can be used to describe the raw spectrum of a grating-based spectrometer as well. Grating-based spectrometers are a more conventional way of implementing spectrometers [1]. The raw-spectrum model for the grating-based spectrometer is given [1, p. 339, Eq. (9.5)] by

$$g_i = \int x(\lambda) h(\lambda - i\Delta) d\lambda, \quad i = 1, 2, \dots, M, \quad (2)$$

where g_i is the detected spectral intensity at the i th CCD element, $x(\lambda)$ is again the input signal spectrum, $h(\lambda)$ is the system impulse response or the instrumental function, and Δ is the sampling interval. The system impulse response $h(\lambda)$ denotes the overall impulse response of the complete system from the slit to the CCD sensor, including the transmittances of the slit and the grating. We note that the raw-spectrum models of the modern filter-based and conventional grating-based spectrometers are similar in their integral form input-output relationship. The role of $h(\lambda)$ in the grating-based spectrometer is taken by the TF, $T_i(\lambda)$, in the filter-array-based spectrometers.

Modern filter-array-based spectrometers [4–8] process the raw spectrum using a DSP algorithm or an estimator to recover the input signal spectrum. The data model for the raw spectrum, \mathbf{y} , which is an input to the DSP algorithm can be represented [4] as a system of linear equations:

$$\mathbf{y} = D\mathbf{x} + \mathbf{w} \quad \text{where} \quad \mathbf{x} \geq 0. \quad (3)$$

In Eq. (3), the $N \times 1$ vector \mathbf{x} contains the samples of the original signal spectrum having an operating bandwidth of W_λ , and the matrix D is an $M \times N$ TF matrix. Each row of D is the TF of a filter. We observe that the matrix D contains only non-negative values as light transmittance is always non-negative.

In this paper, since we are dealing with improving resolution, the value of N is set to be greater than M , so $N > M$. That is, the number of samples of \mathbf{x} is greater than the number of filters. The reason for setting $N > M$ is that resolution in filter-array-based spectrometers is limited by the number of filters M (as will be discussed in detail in Section 3.1). Thus, in order to increase resolution beyond this limit set by the fixed number of filters M , the value of N should be greater than M . By setting $N > M$, we note that the noise-free system of equations in Eq. (3) is underdetermined. We also note then that in Eq. (3), the spacing between the samples of \mathbf{x} is $\Delta\lambda_N = \frac{W_\lambda}{N}$, while the spacing between the samples of the raw spectrum \mathbf{y} is $\Delta\lambda_M = \frac{W_\lambda}{M}$.

Recently, sparse representation of the input signal spectrum, \mathbf{x} , has drawn considerable attention [4–6]. The aim of sparse representation is to decompose the signal spectrum into its constituent spectral (wavelength) components. For example, the signal spectrum of a low-power mercury lamp can be decomposed into its seven wavelength components, whereas the rest of the wavelength components are zero-valued. In general, any signal spectrum vector \mathbf{x} in Eq. (3) can be equivalently represented as a sparse vector \mathbf{s} using a linear transform matrix G , i.e., $\mathbf{x} = G\mathbf{s}$. The vector \mathbf{s} is called a K -sparse vector. In this paper, we refer to \mathbf{s} as a *sparse spectrum*. It contains K non-zero values and $N - K$ zero-valued components, $K \ll N$. The K non-zero values of \mathbf{s} indicate the intensities of the spectral (wavelength) components that constitute \mathbf{x} . The set of K wavelengths corresponding to those intensities is called the support set. Thus, each element of the support set indicates the wavelengths (in nm) that are present in the signal spectrum. The intensities of the remaining $N - K$ wavelength components of \mathbf{s} are zero.

The matrix G is called the kernel matrix, and it is of size $N \times N$. The columns of G are chosen as functions that preserve the natural shape of the signal spectrum. Gaussian kernels are widely used for signal spectrum modeling [8]. The other possible non-Gaussian kernels are secant hyperbolic and Lorentzian [1]. To construct G , we need to know only a single kernel function, the width of which is controlled by a parameter called

full-width at half-maximum (FWHM). When the FWHM is small, the width of the kernel function is also small, and vice versa. Once the kernel function is chosen, it is sampled, and the sampled version of the kernel function is considered as the first column of G . The remaining $N-1$ columns of G are just shifted versions of the first column [6]. We observe that the Gaussian kernel has a uniform width that models narrowband light sources. For modeling broadband light sources, one may require kernels with varying widths, as discussed in [6]. In this paper, we consider only narrowband sources. With the sparse modeling of the input signal spectrum, Eq. (3) can be written as

$$\mathbf{y} = \underbrace{DG}_{\mathbf{A}} \mathbf{s} + \mathbf{w} = \mathbf{A}\mathbf{s} + \mathbf{w} \quad \text{where } \mathbf{s} \geq 0. \quad (4)$$

The goal of the DSP algorithm or the spectral estimator is to obtain an estimate $\hat{\mathbf{s}}$ from the raw spectrum \mathbf{y} , given the matrices D and G .

3. Proposed transmittance functions

3.1. Introduction: Transmittance functions

The transmittance function (TF) of an optical filter is a waveform that has a shape that dictates the fraction of input light that the filter transmits at various wavelengths. The conventional approach is to attempt to design TFs with an ideal shape, i.e., a sharp brick wall as shown in Fig. 2(a). In Fig. 2(a), we show only three filters out of 40 in a filter array. Ideal filters allow light only for desired wavelengths, called the passband, and completely stop the remaining wavelengths, called the stopband. In addition, from Fig. 2(a), we note that the passbands of the ideal TFs do not overlap with each other.

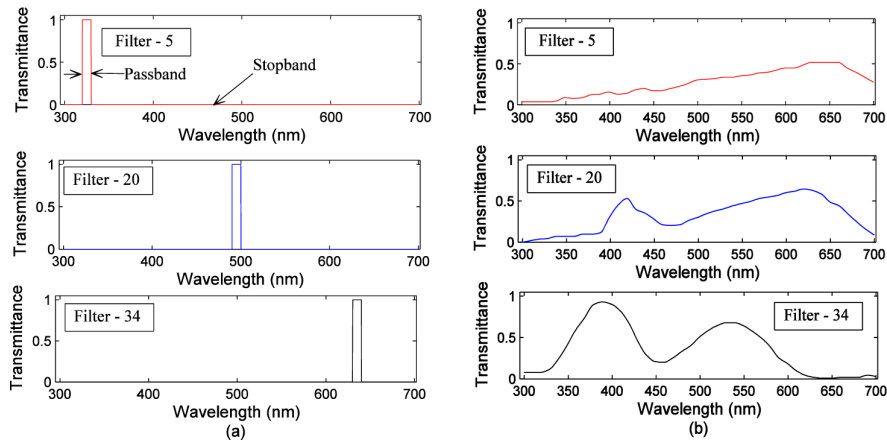


Fig. 2. Transmittances of (a) ideal filters (b) non-ideal filters.

Ideal filters are desired for two reasons. First, the raw spectrum (Eq. (3)) obtained by employing the ideal TFs is a direct estimate of the signal spectrum, and hence, there is no need for a DSP algorithm. This is because each sample of the raw spectrum is a projection (Sec. 2) of the input signal spectrum with an ideal TF. Since ideal TFs are non-overlapping by design, each sample of the raw spectrum only contains information about the spectral (wavelength) components that correspond to a single passband. Second, any two spectral components of the input signal spectrum that are one passband spacing apart are distinguishable (resolvable) in the raw spectrum. If there are any two spectral components separated by less than a passband width, they cannot be resolved. In order to resolve such spectral components, the passband width of each filter should be decreased. Consequently, this increases the number of TFs (number of filters) in order to cover the entire operating bandwidth. Thus, the resolution of the filter-array-based spectrometers with ideal TFs is limited by the number of filters, M . Since the spacing between the samples of the raw spectrum is $\Delta\lambda_M = \frac{W_s}{M}$, it was believed to be the limit on the

resolution of filter-array-based spectrometers. This limit is the same for the more conventional grating-based spectrometer with a CCD array consisting of a total of M pixels. That is, by using the ideal system impulse response ($h(\lambda) = \delta(\lambda)$, the Dirac delta), the spacing between the samples of the raw spectrum of a grating-based spectrometer is also $\Delta\lambda_M = \frac{w_s}{M}$.

However, filters with ideal TFs are not easy to realize in practice [2]. Therefore, non-ideal TFs are inevitable in spectrometers. The shapes of the practical non-ideal TFs employed in a typical filter-array spectrometer [7] are shown in Fig. 2(b). These TFs are not purposely designed in such shapes but are instead accidentally obtained in the micro-level processing of implementing the ideal filters. Unlike the ideal TFs, the waveforms of non-ideal TFs are smooth functions of wavelength. In addition, as seen from Fig. 2(b), the passband of each filter's TF leaks into the passbands of neighboring filters. Hence, non-ideal TFs have considerable overlap with each other. Non-ideal TFs are considered worse for the signal spectrum estimation process in conventional wisdom. The reason is that each sample of the raw spectrum now contains information about the spectral components not only from its own band but also from the neighboring filters' bands, *viz.* interferences. The interference from the use of non-ideal TFs causes severe distortion in the raw spectrum. Therefore, for the spectrometers with non-ideal TFs, post-processing of the raw spectrum using DSP algorithms is inevitable. These algorithms are designed with the aim of extracting the constituent spectral components (sparse spectrum) of the input signal spectrum from the raw spectrum \mathbf{y} , via Eq. (4).

A recent recovery algorithm, reported in [5], is tailor-made for spectrometers. This algorithm is based on a DSP optimization tool called L_1 -norm minimization. L_1 -norm minimization is a recent approach more suitable for solving a noise-corrupted underdetermined system of linear equations such as Eq. (4) than a classical least-squares approach [12–14]. The minimization problem for the recovery of the sparse signal spectrum \mathbf{s} in Eq. (4) can be expressed as:

$$\hat{\mathbf{s}} = \min_{\mathbf{s}} \|\mathbf{s}\|_1 \quad \text{subject to} \quad \|D\mathbf{G}\mathbf{s} - \mathbf{y}\|_2 \leq \varepsilon, \quad \mathbf{s} \geq 0 \quad (5)$$

where ε is a small positive constant.

Note that the problem in Eq. (5) is a bit different from the standard L_1 -norm minimization approach with the inclusion of the additional non-negative constraint, $\mathbf{s} \geq 0$. In order to find the non-negative spectral estimate $\hat{\mathbf{s}}$, the authors in [5] have used a DSP algorithm that was reported in [15]. In [15], the authors have derived an algorithm using the modern interior point method. They have shown that the performance of their algorithm is best when compared to the standard L_1 -norm minimization algorithms such as L_1 magic [14] and LASSO [16], which do not use the additional information. In addition, they have shown that their algorithm is robust against the observation noise, stable, and competitive with the existing algorithms. Using the algorithm in [15] for spectrometers, the authors in [5] have shown an improvement in resolution to about a factor of six below the resolution limit. In addition, the authors in [5] have provided a hint that the recovery performance (and hence the resolution) of the algorithm depends on the design of the TF matrix D .

Recently, fast algorithms for L_1 -norm minimization such as FISTA [17], FOM [18], and ADMM-based algorithms [19] have gained considerable interest among the imaging community. In imaging systems [20–23], the number of measurements (typically more than a few million) is much more than the number of measurements used in the miniaturized spectrometers. Consequently, these imaging systems should be capable of handling large amounts of data efficiently; for spectrometers, however, the accuracy is more important than the efficiency. It is well known that the interior-point-method-based algorithms provide good accuracy. Thus, in this paper, we use the same algorithm that was used in [5]. Readers who may be interested in implementing fast algorithms for spectrometers can refer to the approaches in [17–19].

In this paper, our central goal is to design the TF matrix D in order to enhance the resolution. That is, we use the spectral estimation algorithm in [5] and show how to design the TF matrix D (which in turn determines the filter transmittances) that helps the algorithm to improve resolution. We show that by using our proposed TF matrix, it is possible to enhance resolution 7-fold compared to what was achieved in [5].

3.2. Motivations for new TF design

In this paper, we aim to present a new approach to further enhance (as compared to what was achieved in [5]) the resolution of the spectrometers by designing TFs. The motivations behind designing new TFs are based on the following observations:

1. We recall that each sample of the raw spectrum is modeled as a projection of the input signal spectrum onto a filter TF. In spectrometers with non-ideal TFs, the projection captures not only the in-band but also the out-of-band information about spectral components in each sample of the raw spectrum. Traditionally, when no DSP is used, unintentional capturing of the out-of-band information that is mixed with the information of the desired band is considered as pure distortion of the spectral components. In a completely different viewpoint, however, it can be considered as additional source from which useful information can be extracted. Namely, because of the shapes of the non-ideal filters, each sample of the raw spectrum contains extra information about the entire signal spectrum rather than the only about a particular band like the sample of an ideal TF does. That is, each non-ideal filter collects information from the entire signal spectrum and maps it into a single sample of the raw spectrum. Since the shape of each non-ideal filter is different from that of all other non-ideal filters, we can get many such independent “holistic” views of the entire signal spectrum from each sample of the raw spectrum. This led us to the very natural question: What kind of TFs should provide more holistic and independent information about the spectral components in each sample of the raw spectrum?
2. The modern, L_1 -norm-minimization-based DSP spectral estimator [5] recovers the spectral components very well from the distorted raw spectrum, prompting us to investigate the reason for this unexpected level of performance. We found that the problem of resolving (identifying) the spectral components essentially reduces to the unique identification of s from y . The success of this identification depends on the matrices D and G in Eq. (4). For a given class of signal spectrum, the matrix G is fixed. However, the matrix D can be determined from the design and fabrication of the filter TFs. Hence, we conclude that a good design for the TFs is important in order for the DSP algorithm to recover the spectral components from the raw spectrum.

Thus, it is crucial to design TFs that 1) capture holistic and independent information about the constituent spectral components and 2) aid in recovering those spectral components.

3.3. Design of new TFs

We observe from the previous section that the amount of additional information acquired in each sample of the raw spectrum is solely proportional to the shape of the TFs. By shaping the TFs through a proper design procedure, it is possible to extract much information about the input signal spectrum. Therefore, the TF matrix D that senses the maximum amount of information about the signal spectrum appears to be the best choice for the spectrometer case, and hence, it is our design goal. We note here that a similar design notion arises in the emerging signal processing research area called compressive sensing (CS) where a sensing matrix [12, 13] takes the role of the TF matrix in acquiring a signal.

In CS, the raw signal samples acquired using the matrix D is modeled as $y = DGs + w = As + w$, which is the same as the raw spectrum model in Eq. (4). In CS,

the matrix D is referred to as the sensing matrix; the matrix G , the sparsifying basis; the vector \mathbf{s} , a K -sparse signal; and $A\mathbf{s}$ is called the compressed signal. In CS, the sensing matrix D is chosen such that it captures enough information for the unique identification of the signal \mathbf{s} with as few samples of $A\mathbf{s}$ (the compressed signal) as possible. In order to quantify the ability of the sensing matrix to acquire enough information about the signal in a minimum number of raw samples, the mutual coherence metric, μ , is used [12, 14]. The mutual coherence of the matrix $A = DG$ is defined as follows:

$$\mu = \max_{i \neq j} \left| \langle a_i, a_j \rangle \right| \quad i, j = 1, \dots, N \quad (6)$$

where a_i is the i th column of A . The μ is a measure of the maximum possible correlation among pairs of columns of A . The smaller μ is, the smaller the correlation among the columns of A is. The smaller the correlation is, the better the reconstruction accuracy of \mathbf{s} from \mathbf{y} is. Thus in CS, it is desirable to design the sensing matrix D , such that the matrix $A = DG$ has low coherence.

It is well known in CS that sensing matrices D , the entries of which are drawn from i.i.d. samples of a random variable, exhibit low coherence. Such matrices are called random sensing matrices. These matrices are capable of capturing enough information about the signal \mathbf{s} to perform reconstruction from a small number of samples of \mathbf{y} . Therefore, random sensing matrices are widely employed in CS-based applications.

It appears that the goal of designing a sensing matrix in CS resembles that of TF matrix design. That is, both the TF and the sensing matrix should be designed to capture sufficient information to permit the faithful recovery of the signal. Since both of these goals are met by random matrices [12], in this paper, we consider a random matrix as the TF matrix rather than using the non-ideal TF matrix as in [5–8]. Since each row of the TF matrix denotes a transmittance function of a filter, a row of the random TF matrix is termed a random transmittance function. That is, a filter with a transmittance that exhibits random fluctuations as opposed to possessing a pre-designed transmittance shape is termed a random transmittance function. Thus, our goal of improving resolution reduces to the design of a set of M filters with random TFs. A set of TFs $\{T_i(\lambda)\}$ is referred to as random when

1) The auto-covariance function (ACF) of each TF in the set is close to the Dirac-delta function. That is,

$$\delta(\Delta\lambda) \approx \int [T_i(\lambda) - m_i][T_i(\lambda + \Delta\lambda) - m_i] d\lambda, \quad i = 1, 2, \dots, M, \quad (7)$$

where $\delta(\cdot)$ is the Dirac-delta function, m_i is the mean of $T_i(\lambda)$, and $\Delta\lambda$ is the wavelength difference, and,

2) when the cross-covariance function (CCF) between each pair of TFs in the set is very small. That is, $\varepsilon_{i,j} = \int [T_i(\lambda) - m_i][T_j(\lambda) - m_j] d\lambda$, $i \neq j$, where $\varepsilon_{i,j}$ is a small number close to zero. We discuss in detail the design of filters with such random transmittance in Section 4 using thin-film technology.

3.4. Significance of the proposed approach

The ideas in this paper mirror some of the prior work in CS with regard to the use of random matrices for signal acquisition and recovery. We briefly compare our approach to other well-known CS imaging approaches such as the single-pixel camera (SPC) [20], the P2C2 video system [21], the hyperspectral design (HSD) [22], and the CASSI system [23].

First, we note that the SPC is a well-known CS-based imaging system [20]. In SPC, in order to compressively sample a scene by optical random linear projection, a digital micromirror device (DMD) was used together with a biconvex lens. The DMD is programmed with random 0/1 Walsh patterns [20]. With the availability of such optical devices, random projections of a scene were made almost effortlessly without going

through a pixel-by-pixel scanning, and as a result, it was possible to realize the concept of SPC in practice. The P2C2 video system [21] extends the SPC architecture to video acquisition, and hence it also employs the random 0/1 Walsh patterns. We observe that in both the systems the measurement matrices are 0/1 Walsh matrices.

HSD [22] and CASSI [23] are spectral imaging systems. These systems collect a scene as a set of spatial images. Each image represents a range of an electromagnetic spectrum also known as a spectral band. These spatial images are arranged along various spectral bands to form a three-dimensional structure called a hyperspectral data cube. In HSD, the data cube is modeled as a linear combination of the spectral signatures of endmembers (that is, objects in a scene such as a road, grass, trees, or a roof), where each spectral signature can be obtained from a publicly accessible database such as HYDICE Urban hyperspectral data as used in [22]. Then, the HSD exploits the spatial and spectral redundancies of the data cube by using CS. In particular, HSD adopts the concept of SPC for acquisition of the data cube, and hence they use the DMD with 0/1 Walsh patterns. In contrast to HSD, the CASSI system models the data cube as a linear combination of two-dimensional wavelets and then acquires the data cube using the coded aperture masks. Thus, the measurement matrices in the CASSI system are also random binary 0/1 code matrices. We observe that both the HSD and CASSI systems compressively acquire the spectral data by means of spatial domain random measurement matrices, each element of which is either 0 or 1.

In contrast to the above imaging systems, we note that in our proposed spectrometer, we acquire the spectral data or signal spectrum using random spectral filters (not spatial). That is, as discussed in Section 2, the sensing of the signal spectrum is performed using random analog optical TFs, which are non-negative continuous functions of wavelength. We observe that the set of optical TFs play a sensing role as significant as that of the DMD in single-pixel cameras or the coded aperture masks. In addition, the TFs have values that can be any real number between 0 and 1, in contrast to the 0/1 patterns in the imaging applications. Table 1 summarizes the measurement matrices and their implementation in various imaging applications.

Second, in optical applications the sensing of signals and their subsequent recovery is accomplished with a sensing matrix stored beforehand. For example, in [20, p. 85], the DMD array stores Walsh, Hadamard, or noiselet patterns. The best matrices or the matrices of interest were designed in the digital domain. They were then used to dither the micromirrors back and forth in the DMD. We call this the *digital design first* approach here, as depicted in Fig. 3. In spectrometers, the actual mechanism of sensing is done using the analog TFs, as per Eq. (1), and then the signal spectrum recovery is done using its digital model. We note here that the digital TFs can be losslessly obtained from the TFs of analog thin-film filters (discussed in the next section). Thus, once the analog TFs are designed, they can be digitally represented in the TF matrix in Eq. (4) for spectrum recovery. We call this the *analog design first* approach in this paper (see Fig. 3), in contrast to the *digital design first* approach. The comparison of the *analog design first* approach with the *digital design first* approach may unearth a noteworthy characteristic of the spectrometer designs. Let us elaborate a little further.

Table 1. Measurement Matrices and Their Implementation in Various Imaging Applications

Application	Measurement matrix / Implementation	Page number
Single-pixel camera (2008) [20]	Random 0/1 Walsh matrix	87
	DMD array	84
P2C2 system for sensing videos (2011) [21]	Random 0/1 Walsh matrix	331
	DMD array	335
Hyperspectral data processing (2009) [22]	Random 0/1 Walsh matrix	9
	DMD array	13
Coded aperture snapshot spectral imaging (CASSI) system for hyperspectral video (2012) [23]	Binary CASSI code matrix	6
	Binary coded mask	15

For spectrometers, it is desirable to design a set of random analog TFs with the ACF of each TF shaped as close as possible to the Dirac delta function. The narrower the width of the ACF is, the better the set is at resolving the fine details of the input spectrum. Thus, the resolution of a spectrometer is closely related to the width of the ACF of the TF. In the *analog design first* approach, due to the optical nature of the thin-film filters, there is an inevitable practical limit on how narrow the width of an ACF can be made.

We note that there is no such limit for the digital design. That is, in the digital case, the width of the ACF can be made as arbitrarily narrow as one wishes. For example, this can be achieved by dividing the operating wavelength range by the number of samples, say N , and hence obtaining any resolution we wish to have, and then by digitally generating a TF of length N by drawing independent samples from a probability distribution such as the 0/1 Walsh matrix with $N = 65,536$ as in [20] or as in P2C2 video systems [21]. Since the samples of the TF are independently and identically generated from the given distribution, the ACF of the TF is always as close as possible to the Dirac delta function. This means that one can increase the resolution just by increasing N . This may work for image recovery applications, subject to the creation of a DMD array with so many more elements, but not for spectrometers. For spectrometers, the inherent width of the ACF of the analog TFs must be preserved in its digital TFs. If we first design a random (white) digital TF with an ACF very close to the Dirac delta, the digital TF may not be realizable as the TF of an analog filter. Thus, the resolution in the digital domain is a useless concept unless a way of constructing the practical analog counterpart of the digital TF is given.

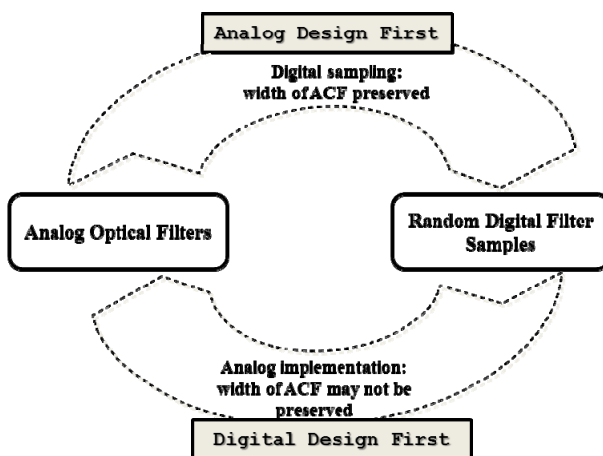


Fig. 3. Summary of *analog* and *digital design first* approaches.

Our main contribution is twofold. The first part shows a specific way to design a set of optical analog filters with TFs that are white and uncorrelated with each other. The second part illustrates that significant improvement in resolution can be achieved with such a filter design.

3.5. Characterizing random TFs

Usually, the degree of correlation of a function over an interval is characterized in terms of the auto-covariance function (ACF). A function can be correlated with a time lagged (or advanced) version of itself. Thus, the ACF is a function of the time difference (time lag). The ACF reveals the correlation among the samples of the function. Similarly, a random TF can be characterized in terms of their ACF, which is now a function of wavelength difference. The shape of the ACF reveals the degree of similarity (correlation) between the light intensities at two different wavelengths of the random TF. If the shape of the ACF is wide with slowly decaying tails, then the TF is said to be highly correlated. On the other hand, if the TF has low correlation at different wavelengths, then the shape of the ACF is narrow with rapidly decaying tails, resembling

a Dirac delta function. The shape of the Dirac delta function is ideal for an ACF. The width of the ACF measures the degree of correlation. That is, an ACF with narrow width exhibits low correlation, and vice versa. A Dirac-delta-like ACF is required in applications such as determining the modulation transfer function of an optical system [24] or detecting displacements or phase gradients in holography [11]. This evidence shows that filters with a Dirac-delta-like ACF are crucial in optical systems and can be designed in practice.

Filters having ACFs that resemble that of a Dirac delta are preferred for L_1 -norm-minimization-based spectrometers. The reason is that a Dirac-delta-like ACF has narrow width, and hence two spectral components that are separated by more than this width are sensed differently by the TF. This independent information about the spectral components aids the L_1 -norm-based algorithms to resolve closely spaced spectral components. Thus, the resolution of a spectrometer has a close connection to the width of the ACF. By using random TFs, we can reduce the width of the ACF and hence the spacing between two spectral components that are independently sensed. On the other hand, non-ideal TFs exhibit high correlation, and hence the widths of the ACFs are wider than those of the random TFs. Hence, information gathered about the two nearby spectral components is not too independent. This reduces the ability of the spectrometer to distinguish the two spectral components. Thus, our goal in this paper is to design the transmittances of filters with Dirac-delta-like ACFs. We call them *random* TFs, while the TFs in [5], *correlated* TFs, in the subsequent discussions.

4. Design of the proposed random transmittance

In the previous section, we introduced random TFs with the goal of improving the resolution of a spectrometer system. Towards this end, we ask ourselves the following question: is it possible, in practice, to design and implement *random* TFs in the analog domain? If yes, how should one be designed? We have found that filters with random transmittance can be designed and implemented by using thin-film optical filters [25, 26] in the way explained in this section.

A thin-film optical filter consists of multiple layers of high- and low-refractive index materials (dielectrics) deposited on a substrate [26]. Each layer has a thickness usually of one quarter-wavelength. Thin-film filters work based on the interference of light transmitted or reflected at the boundaries between the layers. This interference is wavelength-dependent; that is, depending on the number of layers, the index of refraction, and the thicknesses of the layers, the transmission (or reflection) of light through the filter changes with wavelength, and hence the overall transmittance changes. By controlling the thickness of each layer and the number of layers, optical engineers design various filter transmittances such as band-pass filters. In this section, we show how varying the thickness of each layer generates random transmittances using the design methodology in [25].

4.1 Thin-film-based random transmittance filter design

In this section, we first briefly discuss the generic design of thin-film filters to generate band-pass-like transmittances. We then motivate and show how to design random transmittances using the same thin-film structure by varying only the thicknesses of the layers. Let us consider a thin-film filter with m dielectric layers between the input medium (air) and the output medium (substrate). Let n_a and n_s denote the refractive indices of the input and output medium, while n_H and n_L refer to the refractive indices of the high-refractive-index and low-refractive-index intermediate dielectric layers, respectively, and d_k is the thickness of layer k . Assuming no dielectric losses, the transmittance of the thin-film filter as a function of wavelength (λ) and the angle of the incident light (θ) is given [25] by

$$T(\lambda, \theta) = 1 - \frac{1}{2} \left(|\rho_{TE}|^2 + |\rho_{TM}|^2 \right) \quad (8)$$

where ρ_{TE} and ρ_{TM} are power parameters that measure the powers that are contained in the TE and TM polarization modes of the input light.

The power parameters in Eq. (8) are calculated using a recursive set of equations shown in Table 2. We note from Table 2 that the power parameters (and hence the transmittance) depend on the thickness and the refractive index of the different media, as well as on the wavelength and angle of the incident light. Optical engineers use these recursive equations to design the transmittance shapes, such as the band-pass profile, that are required in practice. For example, the typical band-pass transmittances designed using Eq. (8) for various angles of incidence are provided in [25, p. 3765, Fig. (1)]. Also, it is shown in [25, p. 3767, Fig. (3)] that the band-pass transmittances designed using the equations in Table 2 are accurate and are close to the actual transmittances obtained in practice. In Table 2, θ_k denotes the angle that the incident light makes with the normal as it travels from layer k to $k + 1$, η_k is the effective complex-valued refractive index seen by the light as it enters layer k . In order to calculate ρ_{TE} and ρ_{TM} , we first start with $\eta_m = N_m$, and we apply the equation for η_k recursively until we arrive at η_2 , which when substituted into the equation for ρ in Table 2 yields either ρ_{TE} or ρ_{TM} .

In this paper, we adopt the design equations in Table 2 to generate the random transmittances. We are able to generate the random transmittances based on the following observation: We first observed that thin-film filters can provide band-pass-like transmittance because of the principle of interference of light beams. That is, the incident light, when traveling through the thin-film filter, suffers different phase shifts as a result of propagating through different layers. The phase shifts depend on the refractive indices, the thicknesses of the layers, and the wavelength of the incident light. After transmission through the layers, the different beams that reappear at the filter output combine (add) either constructively or destructively depending on their path lengths. The constructive addition leads to increased transmission, and destructive addition leads to decreased transmission of the incident light. Since these constructive and destructive additions are wavelength-dependent, optical engineers usually design the thicknesses of the layers such that constructive interference occurs only for certain wavelengths and thereby achieve band-pass-like transmittances.

Table 2. Recursive Equations for Calculating Power Parameters

$$\rho = \frac{N_1 - \eta_2}{N_1 + \eta_2}$$

$$N_k = \begin{cases} n_k / \cos \theta_k & \text{for TE} \\ n_k \cos \theta_k & \text{for TM} \end{cases}, \quad k \in \{2, \dots, m\}$$

$$\eta_k = N_k \frac{\eta_{k+1} \cos \beta_k + jN_k \sin \beta_k}{N_k \cos \beta_k + j\eta_{k+1} \sin \beta_k}, \quad k \in \{2, \dots, m\}$$

$$\theta_k = \sin^{-1} \left(\frac{n_{k-1} \sin \theta_{k-1}}{n_k} \right), \quad k \in \{2, \dots, m\}$$

$$\beta_k = 2\pi \cos(\theta_k) n_k d_k / \lambda$$

To avoid boundary reflections (and hence to obtain efficient transmission), it is preferred to have a 180° phase shift between the reflected beams at the boundary between the layers. This phase difference directly corresponds to a $\lambda/2$ phase shift of a sinusoidal wave, which can best be accomplished by setting the optical thickness of the layers to $\lambda/4$. Hence, in practice, stacks of quarter-wavelength layers are used as a basic building block [26] for many types of thin-film filters such as band-pass, long-wave-pass. For example, thin-film layers with thickness $\lambda_{\text{normal}}/4n$ are very common in practice, where λ_{normal} is the center wavelength of the band-pass filter with normal incidence and n is the

refractive index of the layer. Thus, we note that the filter transmittances have a direct relationship with the thicknesses of the layers. Based on these observations, we inferred that it is possible to create variations in the filter transmittances by just changing the thicknesses of the layers.

In this paper, in order to generate random transmittances, we propose to vary the thicknesses of the filters, in contrast with the conventional fixed quarter-wavelength thickness. Specifically, we vary the thicknesses of the layers as $\lambda_{\text{normal}}/U$, where U is a uniform random variable that has a mean and variance that are chosen such that the average thickness of a layer is on the order of μm , similar to the thickness of the quarter-wavelength layer. We note that once we generate the random thicknesses ($\lambda_{\text{normal}}/U$), they are inserted into the design, and hence the thicknesses are no longer random after they are generated. We use the same design equations shown in Table 2. The only parameter that changes in the design is β_k (in Table 2), which depends on the thickness. The other recursive equations stay the same. We note that we can produce M independent random filters by generating M different sets of random thicknesses.

4.2. Designed random transmittances and their ACF and CCF

Figure 4(a) shows the random TFs of three filters (5, 20, and 34) generated by the thin-film method for the specifications given in Section 6. It is apparent from Fig. 4(a) that it is possible to generate a random transmittance just by varying the thickness of the layers in the thin-film filters. Figure 4(b) shows the ACF of the transmittance of Filter-5. It is evident from Fig. 4(b) that the shape of the ACF of thin-film-based transmittances is Dirac-delta-like with near-zero correlations other than the zeroth lag, which is what we desire in the proposed spectrometers. Figure 4(c) shows the CCF between Filter-5 and Filter-20. We infer from Fig. 4(c) that the CCF values are near zero for all lags, as expected. We obtained similar ACFs and CCFs for the other filters as well. An additional advantage of using the random TFs is that no stringent filter design constraint, like keeping an exact quarter-wavelength thickness, has to be followed, and this opens the way for mass production. In Section 6, we demonstrate the improvement in resolution obtained by using the proposed random transmittances over the non-ideal transmittances in [5–8]. The implementation of the random transmittance filter array is currently underway in our laboratory.

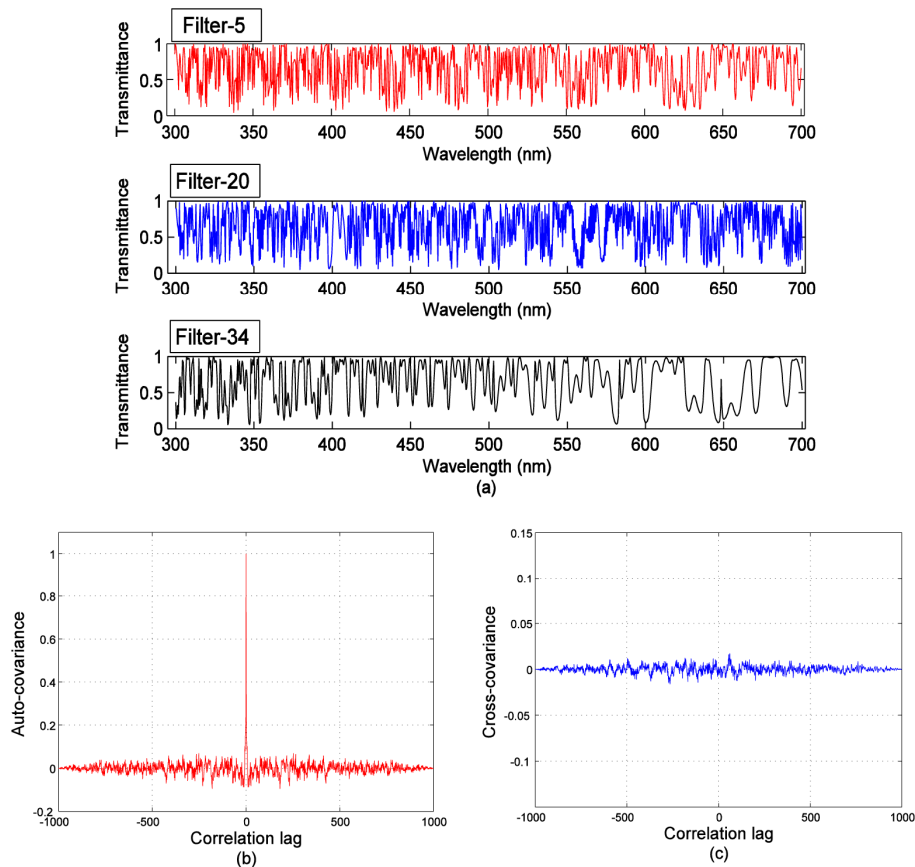


Fig. 4. (a) Random transmittances produced by the thin-film method. (b) ACF of Filter-5. (c) CCF between Filter-5 and Filter-20.

5. Resolution of filter-array-based spectrometers

Traditionally, the resolution of a spectrometer characterizes its ability to distinguish the peaks of two closely spaced spectral components of the input signal spectrum. The spectrometers that identify the closely spaced spectral components reveal fine details about the input signal spectrum. Thus, the quality of the spectrometers is usually specified in terms of this resolution. This was perhaps good enough for the conventional spectrometers but not appropriate for those that use a DSP algorithm. In this section, therefore, we first introduce a metric useful for defining the resolution that can be achieved by using various TF matrices introduced in Section 3. We then provide a new definition of resolution suitable for DSP-based filter-array spectrometers.

5.1. Performance metric useful for defining resolution

As discussed in Section 3.2, the problem of resolving distinct spectral components is equivalent to the exact recovery of the sparse spectral components of Eq. (4) by DSP algorithms. There are infinitely many different sparse spectra \mathbf{s} . Thus, to determine the resolution, it is required to test the recovery of as many sparse spectra \mathbf{s} as possible for a given TF matrix. Usually, a Monte Carlo simulation is used to evaluate if, on average, a specified resolution can be achieved by a TF matrix. The drawback of Monte Carlo simulation is that it is time-consuming. Therefore, in this section, we introduce a metric called genie-aided mean square error (g.MSE). It is a closed-form mean square error expression obtained assuming that the estimator knows the support set, i.e., a set of constituent wavelengths of the input signal spectrum. Thus, g.MSE can be used to indicate the minimum MSE obtainable by the L_1 -norm-based DSP estimator. We will

provide a comparison based on the more practical MSE as well and discuss in Section 6 how well the g.MSE predicts the MSE that is achieved in practice.

Consider again the data model $\mathbf{y} = DG\mathbf{s} + \mathbf{w} = A\mathbf{s} + \mathbf{w}$ given by Eq. (4). Given \mathbf{y} and A , the aim of the L_1 -norm-based DSP algorithms is to find an estimate $\hat{\mathbf{s}}$ of \mathbf{s} . The DSP algorithm should find: 1) a set of constituent wavelengths (the support set) and 2) the non-zero intensity values corresponding to those constituent wavelengths. Let us suppose that the estimator exactly knows the support set. Then, by removing all the non-support set elements from the data model, the raw spectrum can be written as $\mathbf{y} = A_k \mathbf{s}_k + \mathbf{w}$, where \mathbf{s}_k is a $K \times 1$ vector that contains the K non-zero intensity values of \mathbf{s} , and A_k is the $M \times K$ submatrix, the columns of which are the K corresponding columns of A . Note that this is an over-determined problem now, and finding $\hat{\mathbf{s}}$ essentially reduces to estimating the intensity vector \mathbf{s}_k from \mathbf{y} .

An estimator that has complete knowledge of the support set is called an oracle estimator [27]. By making use of this prior information, the oracle estimator determines \mathbf{s}_k from \mathbf{y} using the conventional least-squares approach. The minimum MSE of the oracle estimator is termed a genie-aided MSE (g.MSE). This g.MSE provides a tight lower bound on the accuracy of an estimator that finds \mathbf{s}_k from \mathbf{y} . In this paper, we use the g.MSE to predict the maximum achievable resolution, because

1. It is often used as a golden standard against which the performance of practical algorithms can be compared [27].
2. It can be pre-computed, with just the knowledge of A .

We now define the g.MSE. Let $\hat{\mathbf{s}}_k$ be the estimate of \mathbf{s}_k , the $K \times 1$ intensity vector. Let $\mathbf{e}_s = (\mathbf{s}_k - \hat{\mathbf{s}}_k)$ be the error vector. The co-variance of the error vector \mathbf{e}_s is then given by $C_s = \mathbb{E}[\mathbf{e}_s \mathbf{e}_s^T] = \sigma^2 (A_k^T A_k)^{-1}$ where A_k denotes an $M \times K$ submatrix of A obtained by taking only those K columns that correspond to K wavelengths in the support set. We note that the *expectation* operation is taken over the probability distribution of the noise vector. The matrix C_s is of size $K \times K$. Now, the average MSE, $\mathbb{E}[\|\mathbf{s}_k - \hat{\mathbf{s}}_k\|_2^2]$, is calculated as $\mathbb{E}[\|\mathbf{s}_k - \hat{\mathbf{s}}_k\|_2^2] = \mathbb{E}[\mathbf{e}_s^T \mathbf{e}_s] = \text{tr}(C_s)$, where $\text{tr}(C_s)$ is the trace of the matrix C_s . Since the input signal spectrum is $\mathbf{x} = G_k \mathbf{s}_k$ and $\hat{\mathbf{x}} = G_k \hat{\mathbf{s}}_k$, the minimum MSE between \mathbf{x} and $\hat{\mathbf{x}}$ with the known support set can be calculated. We define the error vector $\mathbf{e}_x = (\mathbf{x} - \hat{\mathbf{x}}) = G_k (\mathbf{s}_k - \hat{\mathbf{s}}_k)$. The co-variance of the error vector \mathbf{e}_x is then given by $C_x = \mathbb{E}[\mathbf{e}_x \mathbf{e}_x^T] = \sigma^2 G_k (A_k^T A_k)^{-1} G_k^T$. Now, the average genie-aided MSE is calculated as

$$\text{g.MSE}(A, \sigma^2, K) \triangleq \mathbb{E}[\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2] = \mathbb{E}[\mathbf{e}_x^T \mathbf{e}_x] = \text{tr}(C_x) = \text{tr}(\sigma^2 G_k (A_k^T A_k)^{-1} G_k^T). \quad (9)$$

For simplicity, we drop the arguments of g.MSE in the subsequent discussions. We note that the MSE that we would obtain by using any spectrum estimator is always bounded below by the g.MSE.

We note from Section 2 that a sparse spectrum \mathbf{s} contains only K non-zero wavelength components, and the remaining $N-K$ wavelength components are zero. In addition, these K non-zero wavelength components can be present anywhere among the N possible wavelength components. That is, for a given N and K , there are $\binom{N}{K}$ possible sets of locations (support sets) where each set contains K wavelengths. Therefore, we can obtain $\binom{N}{K}$ g.MSEs. Each g.MSE is for a signal spectrum with K wavelengths. We note that for a fixed K , increasing N increases the number of support sets exponentially. In the next section, we define the resolution of a DSP-based spectrometer based on the g.MSE. In this paper, we always express the g.MSE in decibels (dB) for convenience.

5.2. Resolution: A DSP-based definition

As discussed in Section 3.1, when no DSP is used, the resolution is determined by the number of filters, M , in the filter array. That is, $\Delta\lambda_M = \frac{W_\lambda}{M}$ is the minimum distance between two spectral components that can be distinguished by the spectrometer. The larger the number of filters is, the better the resolution is. However, in miniature spectrometers, the number of filters is fixed, and hence the resolution is limited. Thus, $\Delta\lambda_M$ is called the resolution limit of the filter-array-based spectrometers. In order to further improve the resolution (beyond the resolution limit), DSP estimators are used [5]. These estimators exploit the sparse nature of the signal spectrum (Section 2) and employ L_1 -norm-based estimators to improve the resolution. In this section, we introduce a new definition to measure the resolution that can be obtained by using the DSP estimators for the filter-array-based spectrometers.

We recall from Section 2 that the spacing between the samples of the input signal spectrum x (or equivalently s) is $\Delta\lambda_N = \frac{W_\lambda}{N}$ while the spacing between the samples of the raw spectrum is $\Delta\lambda_M = \frac{W_\lambda}{M}$. Since $M < N$, $\Delta\lambda_M > \Delta\lambda_N$, and therefore, the raw spectrum is in a low-resolution state ($\Delta\lambda_M$), while the signal spectrum x is in a high-resolution ($\Delta\lambda_N$) state. The high-resolution state can be realized and determined when a quality estimate of x is obtained for all possible x .

A way to define the resolution of a DSP-based spectrometer therefore is to find the minimal distance between the spectral components that can be accurately detected by the DSP estimator. Since the minimal distance is the distance between any two adjacent spectral components, viz. $\Delta\lambda_N = \frac{W_\lambda}{N}$, as N increases, the minimal distance decreases, and hence the resolution increases. We may come to the point of diminishing returns when going beyond a certain N . We call this N_{\max} . In addition, we define the minimum distance to be $\Delta\lambda_{\min} = \frac{W_\lambda}{N_{\max}}$. Thus, to find this ‘‘digital resolution,’’ it is sufficient to find N_{\max} .

We obtain N_{\max} using the g.MSE measure introduced in Section 4.1 as follows: We first fix the parameters N , M , K , G , σ^2 , and a TF matrix, which are required to calculate the g.MSE. We then check if the condition $\text{g.MSE} \leq \delta$ is met for a given percentage of the total $\binom{N}{K}$ possible support sets, where δ is an MSE specified by the user. If the condition is met, it indicates that the DSP estimator is able to detect all the spectral components. We then increase the value of N further and check the condition again. We keep repeating this process (that is, increasing the value of N and checking the condition) until the condition is no longer met. We finally declare the largest value of N that met the condition as N_{\max} . The minimum spacing and hence the maximum achievable resolution of the DSP-based spectrometer is then $\Delta\lambda_{\min} = \frac{W_\lambda}{N_{\max}}$. In this paper, we increase the value of N starting from M that corresponds to the resolution limit $\Delta\lambda_M = \frac{W_\lambda}{M}$ of filter-array-based spectrometers. We note that it is not possible to increase N indefinitely with the hope of obtaining a minimum spacing. There is a limit for increasing N . This limit depends only on the TF matrix for the fixed parameters M , K , G , and σ^2 . Since the g.MSE captures all these parameters, it is enough to define the resolution based on the g.MSE measure.

For a signal spectrum with K spectral components, we define the maximum resolution ($\Delta\lambda_{\min}$) of the spectrometer as follows:

$$N_{\max} := \max\{N \in \{M, M+1, \dots\} : \Pr\{\text{g.MSE} \leq \delta\} \geq \rho\}$$

$$\Delta\lambda_{\min} := \frac{W_\lambda}{N_{\max}} \quad (10)$$

where the user-defined MSE δ is specific to an application. It specifies the minimum MSE that should be guaranteed by the DSP algorithm. The notation $\Pr\{\text{g.MSE} \leq \delta\}$

denotes the probability of the event $\text{g.MSE} \leq \delta$. The sample space for the event is the set of all possible support sets. Each support set is mapped into a g.MSE value. The variable ρ , $0 \leq \rho \leq 1$, denotes the probability of the event $\text{g.MSE} \leq \delta$. That is, the variable ρ represents the fraction of the total support sets that satisfy the event. We express ρ in terms of percentage as $100 \times \rho\%$. If $\rho = 1$, then 100%; that is, all the $\binom{N}{K}$ support sets should satisfy the event. If $\rho = 0.95$, then 95% of the total number of support sets should satisfy the event. We say that the K spectral components are resolvable if we can find an N such that the probability of the event $\text{g.MSE} \leq \delta$ is greater than or equal to a specified ρ . We note that $\Delta\lambda_{\min}$ in Eq. (10) is the resolution of the DSP-based spectrometer whereas $\Delta\lambda_M = \frac{W_\lambda}{M}$ is the limit on the resolution of filter-array spectrometers.

The definition of the resolution given by Eq. (10) is general in nature in that it can include prior information. For example, if we know beforehand that the signal spectral components occupy only a certain range of the spectrum, then this definition automatically incorporates this prior information. Since prior information reduces the number of support sets and therefore the change is only in the event $\text{g.MSE} \leq \delta$, the definition of the resolution remains unaffected. Since we use g.MSE , the minimum possible MSE, the corresponding resolution obtained is the maximum possible by employing any kind of DSP-based estimator. In the next section, we evaluate the resolution that can be achieved by the correlated TF matrix in [5] and the proposed random TF matrix.

6. Results and discussion

In this section, we aim to investigate how much gain in terms of resolution improvement can be obtained by using random TFs in comparison with the correlated, non-ideal TFs in [5]. We first investigate the maximum possible resolution by using the definition in Eq. (10). We then use the L_1 -based DSP algorithm reported in [5] and see how much of this resolution can be realized in practice. We set the number of filters in the array to $M = 40$. The non-ideal TFs, for the 40 filters, have been obtained from [7], which are also used in [5]. The random TFs are generated using thin-film filters with $m = 4$ layers. We used the following refractive indices: $n_a = 1$, $n_s = 3$, $n_H = 4$, and $n_L = 1.2$ (Section 4.1). The angle of incidence of the light at the input medium is 45° . The center wavelength $\lambda_{\text{center}} = 850 \text{ nm}$. The thicknesses of the four layers are chosen as explained in Section 4.1, with a different thickness for each filter. After we generate the random TFs, we sample them, and arrange them in rows to form a random TF matrix.

We choose a Gaussian kernel with an FWHM of 1 nm to generate the kernel matrix G . The operating wavelength range of the spectrometer is from 300 nm to 700 nm, corresponding to a W_λ of 400 nm. We vary N in order to find the maximum possible resolution $\Delta\lambda_{\min}$ as per Eq. (10). The value of N begins with M (which corresponds to the resolution limit) and is set to increase in steps of 40, i.e., $N = 40, 80, 120$, etc. For $N = 40$, the minimum spacing between two adjacent spectral components is $\Delta\lambda_N = \frac{400 \text{ nm}}{40} = 10 \text{ nm}$ (from Section 3), which is the resolution limit (maximum possible resolution without using a DSP estimator). We define the signal-to-noise ratio (SNR) for the data model in Eq. (4) as $\text{SNR} = \frac{|DGs|^2}{M\sigma^2}$. We keep the SNR at 30 dB.

We consider a sparse signal spectrum with $K = 5$ spectral (wavelength) components. The locations of these spectral components within the spectrum are random. Thus, the total number of possible locations (support sets) is $\binom{N}{5}$ for a given N . We choose 50,000 support sets at random from this total population. For each of these support sets, we calculate the g.MSE as per Eq. (9). We set $\rho = 0.95$. As we vary N , we note the value of the g.MSE below which the g.MSE s of 95% of the support sets lie. Figure 5 shows the

plot of the 95th-percentile MSE versus N obtained for the TF matrix in [5] and for the random TF matrix.

We now set $\delta = -5$ dB and find N_{\max} . That is, we aim to find the maximum N such that the 95th-percentile MSE is less than -5 dB. We observe from Fig. 5 that for the TF matrix in [5], the 95th-percentile MSE is less than -5 dB when $N_{\max} = 62$. Thus, the maximum resolution of the DSP-based spectrometer obtained by using the TF matrix in [5] is $\Delta\lambda_{\min} = \frac{W_{\lambda}}{N_{\max}} = \frac{400 \text{ nm}}{62} = 6.5 \text{ nm}$. Therefore, the improvement in resolution over the resolution limit is 1.54 (10 nm / 6.5 nm). In the case of random TFs, $N_{\max} = 405$ and the maximum resolution obtained is $\Delta\lambda_{\min} = \frac{400 \text{ nm}}{405} = 0.99 \text{ nm}$. The improvement in resolution in this case is 10.1 (10 nm / 0.99 nm). We observe that the random TFs provide a resolution improvement of approximately a factor of 7 over that of the TFs in [5].

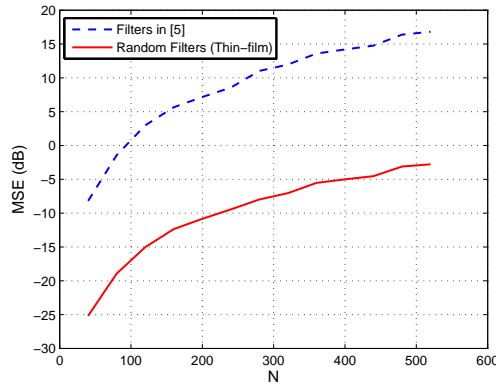


Fig. 5. g.MSE against resolution (N) for $K = 5$ and 50,000 support sets with $\rho = 0.95$.

Table 3 summarizes the resolution obtained by using various design approaches and algorithms for the same filter and signal settings as before. From Table 3, we note that the proposed random TFs achieve 7-fold resolution improvement compared to the TFs in [5] and 10-fold resolution improvement when compared to the ideal TFs.

We have also considered the non-Gaussian kernels such as Lorentzians and secant hyperbolic (mentioned in Section 2) for the same FWHM. Here, we report that we have obtained a similar resolution improvement to those obtained by using the Gaussian kernel. In addition, we note that comparing the resolution obtained by using ideal TFs with that of the random TFs is equivalent to comparing the resolution of ideal grating-based spectrometers with the random filter-array-based spectrometer. This is because, as discussed in Section 3.1, the resolution limit of both of the spectrometers is the same. Thus, we are in one sense indirectly comparing our proposed approach with the best possible conventional grating-based spectrometers.

Table 3. Summary of Filter Design Approaches and Corresponding Resolution

Methods	Design approach	Resolution	Recovery method
Conventional	Ideal brick-wall	10 nm	No DSP
Correlated TFs in [5]	Holistic design by accident	10 nm	Least-squares and adaptive regularization [7, 8]
Correlated TFs in [5]	Holistic design by accident	6.5 nm (a 1.5-fold improvement compared to the ideal brick wall filters)	L_1 DSP estimator [5]
Proposed Random TFs	Random design by purpose	0.99 nm (an improvement of 7-fold compared to the TFs in [5] and 10-fold compared to the ideal brick wall filters)	L_1 DSP estimator [5]

We now investigate the practical spectral estimate obtained by using the DSP estimator reported in [5] when both the non-ideal and the random TFs are used in the filter-array spectrometer. For this purpose, we consider a signal spectrum generated by a mercury arc lamp, commonly used in fluorescence microscopy applications. The original signal spectrum of the mercury arc lamp has five ($K = 5$) prominent spectral components located at 365 nm, 405 nm, 436 nm, 546 nm, and 579 nm [22]. We consider $N = 360$ and $M = 40$. Figure 6 shows the original signal spectrum of the mercury arc lamp and an estimate of the signal spectrum obtained by using the correlated TF matrix in [5]. Figure 7 shows the estimated signal spectrum obtained by using the proposed random TF matrix.

From Figs. 6 and 7, we observe the following: First, it is evident that the estimate obtained by using the random TFs is closer to the original signal spectrum than that of the estimate using the non-ideal, correlated TFs. Second, the five spectral components of the original signal spectrum are determined correctly by the random TFs. However, the non-ideal TFs miss all five of the spectral components. Both the non-ideal and random TFs detect additional spectral components that are not present in the original signal spectrum. These additional spectral components appear as small bumps in Figs. 6 and 7.

In order to investigate how much resolution can be obtained using the DSP algorithm, we computed the MSE of the spectral estimates shown in Figs. 6 and 7 and compared them with the theoretical MSE shown in Fig. 5. The MSE of the spectral estimate by the random TFs (in Fig. 7) is about -5.5 dB which is just 1.3 dB away from the theoretical 95th-percentile MSE of -6.8 dB (in Fig. 5) at $N = 360$. For the non-ideal filters, the calculated MSE of the spectral estimate is 14 dB, which is 1.1 dB away from the MSE (12.9 dB) shown in Fig. 5. Thus, we observe that the resolution improvement deduced from Fig. 5 is actually achievable by using the practical DSP algorithms. The difference in MSE between the theoretical and the practical estimates is because of the detection of additional spectral components by the DSP algorithm. The difference in the practical MSE between the non-ideal TF in [5] and the random TFs using the DSP algorithm is about 18 dB.

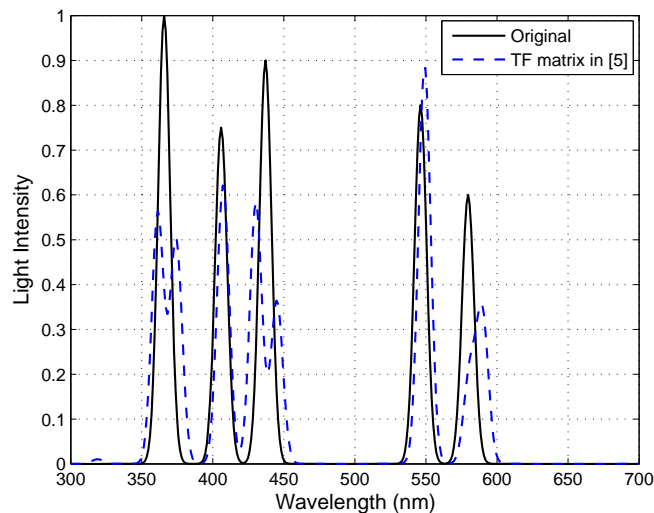


Fig. 6. Reconstruction of a mercury arc signal spectrum by a non-ideal TF matrix in [5].

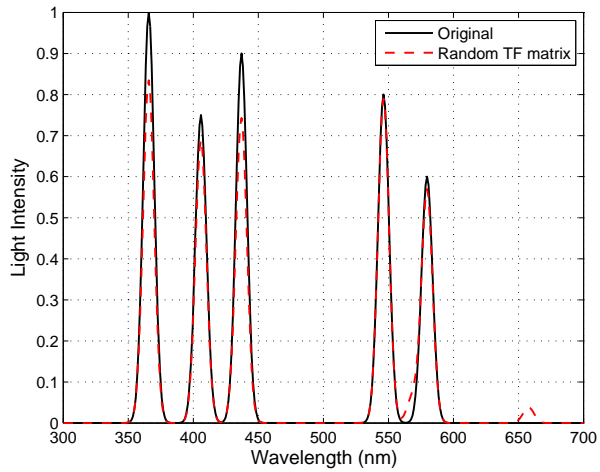


Fig. 7. Reconstruction of a mercury arc signal spectrum by the proposed random TF matrix.

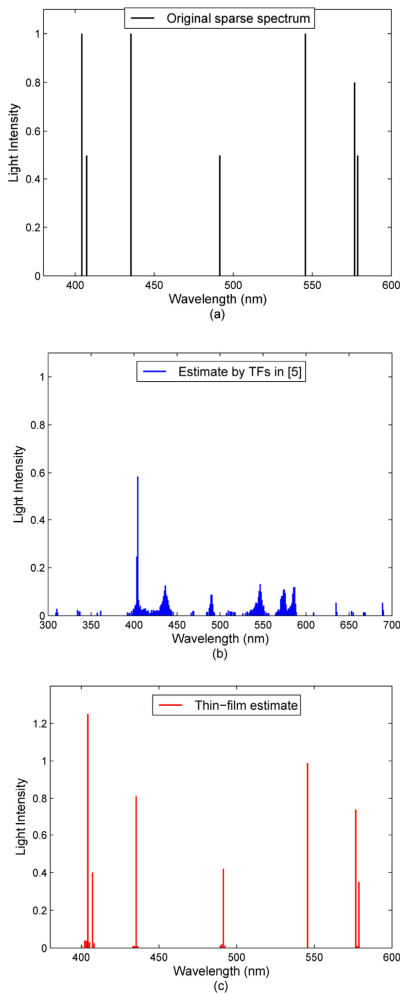


Fig. 8. (a) Original sparse spectrum of the mercury lamp. (b) Estimated sparse spectrum by using TFs in [5]. (c) Estimated sparse spectrum by thin-film-based random TFs.

In order to show that a resolution of 0.99 nm (see Table 3) is achievable by using random TFs, we consider an original mercury lamp spectrum [28] with prominent and weak spectral lines as follows: 404.656 nm, 407.781 nm, 435.835 nm, 491.604 nm, 546.074 nm, and a pair at 576.959 nm and 579.065 nm. Thus, the sparse spectrum of the mercury lamp contains seven spectral lines or components as shown in Fig. 8(a). We note that the least separation among these spectral lines or components as shown in Fig. 8(a). We note that the least separation among these spectral lines is 2.106 nm, which is between the pair 576.959 nm and 579.065 nm. We aim to resolve these components in the sparse domain. We consider the case of $N = 400$. Figures 8(b) and 8(c) show the estimate of the sparse spectrum using the TFs in [5] and using the proposed thin-film-based random TFs, respectively. It is apparent from Fig. 8(c) that random TFs correctly estimate the seven dominant wavelength components in the sparse spectrum. In addition, random TFs clearly resolve the pair of two closely spaced spectral components at 576.959 nm and 579.065 nm. This evidence supports the fact shown in Table 3 that random TFs are capable of resolving any two spectral components that are more than 0.99 nm apart. In contrast, the estimate obtained by using the TFs in [5] misses all seven of the dominant spectral components as shown in Fig. 8(b). In addition, the estimate contains spectral components that were not originally present in the signal spectrum. Based on these results and observations, we conclude that the filters with random TFs outperform the non-ideal TFs in terms of resolution and reconstruction performance.

7. Summary and conclusions

In this paper, we have proposed random-transmittance-based filters to improve the resolution of a spectrometer. We have shown that the optical filters with random TFs are designed to acquire *holistic* information about the signal spectrum, rather than the *localized* information that was the target of traditional designs. The holistic information captured by each filter when put together at the spectral estimator provides multiple independent views and helps the DSP algorithm to differentiate the details in the spectral components. We have shown via examples that spectrometers with random TFs provide 7-fold resolution improvement when compared to the use of filters that aim at implementing the ideal TFs. Rather than trying to design filters with the ideal brick wall TF, which has been the design paradigm in the past, our study shows that designing filters with random TFs should be the design paradigm for modern DSP-based spectrometers. This shift in the design paradigm not only brings forth an order-of-magnitude improvement in terms of resolution and MSE but also may relax the difficulty of the filter engineering process, resulting in less stringent design requirements.

Acknowledgments

The authors would like to thank the anonymous reviewers for their thorough review of the manuscript, for useful references, and for their role in improving the original manuscript. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MEST) (Do-Yak Research Program, No. 2012-0005656).