Available online at www.sciencedirect.com
ScienceDirect
Int. J. Electron. Commun. (AEÜ) 64 (2010) 99–111
International Journal AEÜ of Electronics and Communications
www.elsevier.de/aeue

Energy-efficient utility maximization for wireless networks with/without multipath routing

Jingqiao Zhang[a,*], Heung-No Lee[b]

[a]Department of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
[b]ECE Department, University of Pittsburgh, Pittsburgh, PA 15260, USA

Received 7 May 2007; accepted 10 December 2008

Abstract

In wireless networks, end-to-end communication depends on link capacities which, in turn, are determined by transmit powers of interfering links. Optimal network performance and energy efficiency can be achieved by jointly optimizing congestion control and power control. In this paper, we study this joint optimization problem by formulating it into convex programming, i.e., we maximize a compound function which is a network utility function minus a factor, named tradeoff factor, of the associated power cost. We prove that this tradeoff factor is essential for good energy efficiency while maintaining the network throughput at a satisfactory level. The problem is solved by a distributed dual-decomposition based algorithm energy efficient jointly optimal congestion and power control (EJOC). EJOC tackles the power control problem in a recursive manner, operating as easily as the steepest descent method but converging much faster. This optimization framework is further extended to networks where each data source may have multiple alternative paths to its destination. Simulation results show that the proposed algorithm converges faster than other algorithm and is capable of significantly improving the energy efficiency of the network.

Keywords: Cross-layer optimization; Power control; Congestion control; Energy efficiency; Multipath routing

1. Introduction

A plethora of different tradeoff relations exist in wireless networks because wireless medium is open by nature. A wireless network can be modeled as a graph whose level of connectivity changes as the transmit powers are varied: it is loosely connected if the powers barely able to attach to the next hop nodes are used, or well connected if sufficiently large powers are used. A well-connected graph may imply the availability of multiple routes to destination; however, unlike the wired counterpart, a well-connected graph does not always indicate good because signal interference may get too large. In addition, each individual node sees different channel and thus channel capacity varies at each link. As a result, the problem of selecting power level at each link and that of selecting traffic injection rate at each source are tightly coupled.

We aim to use the network utility maximization (NUM) framework by Kelly, Maulloo, and Tan (KMT) [1] for joint power and congestion control. It is a convex optimization method developed for optimal congestion control of data traffic over wired networks, such as the Internet . Recently, this work has been utilized in many cross-layer wireless networking researches [2–9]. The framework lends itself to a Lagrange-dual based solution with which a gradient descent type distributive control at each link and at each source individually can drive the network to a globally optimal state.

* Corresponding author. Tel.: +1 518 276 8210; fax: +1 518 276 4897.
E-mail addresses: zhangj14@rpi.edu, jqzhang@amazon.com (J. Zhang), hnlee@ee.pitt.edu (H.-N. Lee).

1434-8411/$ - see front matter © 2008 Published by Elsevier GmbH
doi:10.1016/j.aeue.2008.12.001

Examples include the study of rate-reliability tradeoff [3], joint congestion and power control in wireless networks [4,5], joint routing and power allocation [6], joint opportunistic scheduling and congestion control [7], joint power adaptation, scheduling, and routing [8], and joint congestion control and medium access control [9].

In this paper, we are interested in the jointly optimal control at both the transport and the physical layer. Congestion control is conducted at the transport layer to support high network throughput, while power control is done at each link, aiming to achieve high energy efficiency. They interact through the link capacity constraints, i.e., the traffic flow routed on any particular link is feasible only when there is enough capacity of the link which can be varied by power control. We carefully model this coupling between the layers and obtain a distributed and energy-efficient joint congestion and power control algorithm for wireless networks.

The optimization frameworks in the literature can be categorized into single-path utility maximization, where each source node has a single path to the destination [1–3], and multipath utility maximization, where the traffic from each source to its destination can pass through multiple alternative paths [1,10–12]. In [1], the authors briefly discuss the multipath problem, and use a penalty function approach to produce generally approximate solutions. The single-path utility maximization in [10] is extended to the multipath case and produces exact solutions. However, both this method and that in [1] suffer from the convergence instability problem which is due to the lack of the strict concavity of the objective function in its flow-rate variables. The authors in [11] address this problem by adding a quadratic term to the objective function. This is similar to the proximal optimization method [13] which is used in [12] for wired networks and adopted in this paper as well in the wireless context.

Our objective in this paper is to consider the joint congestion and power control problem for wireless networks. The most related work in this direction includes Chiang's analysis [4], where the author has addressed the joint control in a NUM based flow-rate utility maximization framework and proposed a steepest descent method to optimize the flow rate. In this paper, we provide the following unique contributions for this problem:

1. First, different from the flow-rate utility maximization [4], we focus on the energy-efficiency issues by explicitly integrating both flow rate and power cost into the objective function. This integration of power cost turns out to be very important and necessary, because otherwise we prove that at least one link will transmit at its maximum power, leading to a rather high interference level in the network (see Proposition 1). By adjusting the tradeoff factor between the network utility and the power cost, the energy efficiency of the network can be significantly improved at the expense of a slight decrease in network throughput.
2. Second, we propose a novel recursive method to solve the optimization problem of power control, which operates as easily as the steepest descent method (as used in [4]) but converges much faster and does not require parameter training. The fast convergence and no need for training make the proposed method suitable for different wireless networks that come with vastly different characteristics and whose topology and link quality vary frequently over time. As a comparison, we notice the steepest descent method usually converges slowly and its control parameter (i.e., the step size used to updated variables at each iteration) needs to be tuned a priori for good performance. The optimality of the proposed method is proved (see Proposition 2).
3. Third, our analysis is extended to wireless networks which support multipath routing. The convergence instability resulting from multipath routing is dealt with by introducing a modified proximal optimization method.

The rest of the paper is organized as follows. Section 2 formulates the joint optimization problem. Section 3 develops a distributed algorithm in which the power control subproblem is solved by a new recursive algorithm. In Section 4, we extend the analysis to the multipath routing network. This is followed by discussions on simulation and numerical results in Section 5. Finally, we conclude the paper in Section 6.

2. Problem formulation

Consider a wireless multihop network consisting of N nodes and L directed links. Some nodes are traffic sources and some act as relay nodes. Assume each source s has a single path to the destination, which is composed of a sequence $L(s)$ of connected links l, $l \in L(s)$. Accordingly, we denote $S(l)$ as the set of data flows that pass through a certain link l. Notice that $l \in L(s)$ if and only if $s \in S(l)$. Denote x_s as the flow rate originated from source s, and P_l as the transmit power of link l. It is assumed in this section that each source has a single path to its destination. The multipath problem will be analyzed later. In an interference-limited wireless network, the information-theoretic capacity c_l of each link l is not fixed, but instead can be considered as a function of the transmit powers and channel conditions:

$$c_l(\mathbf{P}) = 1/T \cdot \log(1 + \mathrm{SINR}_l), \tag{1}$$

where T is the symbol period and is assumed to be one unit without loss of generality; $\mathbf{P}$ is the vector of the transmit powers P_l, and SINR_l denotes the signal-to-interference-and-noise ratio (SINR) at the receiver of link l. We have $\mathrm{SINR}_l = G_{l,l}P_l/(\sum_{k\neq l} G_{k,l}P_k + n_l)$, where n_l is the power density spectrum of the additive white Gaussian noise at the receiver of link l, and $G_{k,l}$ is the channel fading from the transmitter of link k to the receiver of link l. For simplicity, we incorporate into $G_{l,l}$ other constant coefficients, such as the spreading gain in CDMA systems. Expression (1) can be approximated by $c_l(\mathbf{P}) = \log(\mathrm{SINR}_l)$ when SINR_l is much

larger than 1 (e.g., $\geqslant 5$). This is reasonable either when the interference is much less than the received signal power, or when the spreading gain is large, e.g., in a CDMA system. In this case, the link capacity c_l can be approximated by

$$c_l(\mathbf{P}) = \log\left(\frac{G_{l,l}P_l}{\sum_{k\neq l}G_{k,l}P_k + n_l}\right), \tag{2}$$

where the sum of interference terms over $k \neq l$ can be conducted in practice only over active links in the two-hop neighborhood. Expressions (1) and (2) from information theory can be easily extended to the systems with practical modulation schemes. For example, it is known for M-QAM modulation schemes, the link capacity can be expressed as $\log(1 + K \cdot \mathrm{SINR}_l(\mathbf{P}))$, where K is a constant determined *a priori* by the modulation scheme and the required bit error rate [14]. In this case, we can simply redefine $G_{l,l}$ as K times the original $G_{l,l}$, and thus absorb the constant K into expression (1) and (2).

2.1. Problem formulation

One objective of this paper is to improve the energy efficiency of the network, i.e., to maximize the utility of the flow rates at the expense of appropriate transmit powers. The problem can be formulated as follows:

$$\mathbf{P1}: \text{maximize} \quad \sum_s U_s(x_s) - \beta\sum_l V_l(P_l)$$

$$\text{subject to} \quad \sum_{s\in S(l)} x_s \leqslant c_l(P), \quad \forall l, \tag{3}$$

$$x_s \geqslant 0 \quad \text{and} \quad 0 \leqslant P_l \leqslant P_l^{max}, \ \forall s, l, \tag{4}$$

where $U_s(x_s)$ is the *utility function* of traffic rate x_s, which depends on the type of service at source s; $V_l(P_l)$ is the *cost function* of the transmit power P_l, which is determined by the energy requirement; $\beta \geqslant 0$ is the tradeoff factor which can be interpreted as the 'knob' to balance the network utility of flow rates and the cost of transmit powers. The second constraint (4) states that each link l has a fixed power limit P_l^{max}.

For simplicity, the cost function is considered as a weighted value of the transmit power, i.e.,

$$V_l(P_l) = w_l P_l, \quad w_l \geqslant 0. \tag{5}$$

Different shapes of utility functions lead to different types of throughput fairness. For example, a family of utility functions parameterized by $[p_s, \alpha_s](p_s \geqslant 0, \alpha_s \geqslant 0)$ is proposed in [15]:

$$U_s(x_s) = \begin{cases} p_s \log x_s, & \alpha_s = 1, \\ p_s(1-\alpha_s)^{-1}x_s^{1-\alpha_s}, & \alpha_s \neq 1, \alpha_s \geqslant 0, \end{cases} \tag{6}$$

where p_s represents the weight (or priority) of different flows. If $\alpha_s = 0$, (6) is used to maximize the network throughput. If $\alpha_s = 1$, proportional fairness among flows is attained; if $\alpha_s = 2$, then harmonic mean fairness; and if $\alpha_s \to \infty$, then max–min fairness. It is clear, the general types of utility functions in (6) facilitate us to analyze different types of services and different efficiency-fairness tradeoffs.

2.2. Energy efficiency

In the literature, only the network utility of flow rates appears in the objective function, although the transmit powers are also considered as variables to be optimized. For example, the flow-rate utility maximization addressed in [4] can be considered as a special case of P1 by setting $\beta = 0$. As the main concern of this paper is *energy-efficient* network utility maximization, we prove that a positive tradeoff factor β is crucial to the energy efficiency of the whole system.

Proposition 1. *In the case of $\beta = 0$, the optimal solution to P1 is such that at least one link l transmits at its maximum power*, i.e., $P_l^* = P_l^{max}$ *for some l.*

Proof. (By contradiction) Denote $(\mathbf{x}^*, \mathbf{P}^*)$ as the optimal solution to P1 when $\beta = 0$, where $\mathbf{x}^*$ is the vector of the optimal flow rates x_s^* and $\mathbf{P}^*$ the vector of the optimal transmit powers P_l^*.

Suppose the optimal transmit power is strictly within the power constraints, i.e., $P_l^* < P_l^{max}, \forall l$. It is clear that there exists a positive constant δ such that

$$(1+\delta)P_l^* \leqslant P_l^{max}, \quad \forall l.$$

For example, one possible value is $\delta = \min_l\{(P_l^{max} - P_l^*)/P_l^*\}$. According to the expression of the link capacity (2), it is clear that a proportional increase in all transmit powers leads to greater link capacities, i.e., $c_l(\mathbf{P}^*) < c_l((1+\delta)\mathbf{P}^*)$.

In response to the increased capacity, the flow rates can be increased by a certain amount without violating the capacity constraints. This implies a better solution than $(\mathbf{x}^*, \mathbf{P}^*)$ exists – a consequence of $\beta = 0$ since there is no penalty on the increase in the transmit powers – but it contradicts the optimality of $(\mathbf{x}^*, \mathbf{P}^*)$. □

The statement in Proposition 1 is consistent with intuition. A link increases its transmit power to get a higher capacity. However, the higher the power of this link, the higher the interference to other links; hence the other links raise their current power levels to a certain degree to at least maintain their current channel capacities. As a result, all links end up increasing their transmit powers so as to not reduce their link capacities, until some links reach their maximal power limit. The increase in transmit powers (and therefore interference), however, leads to progressively more marginal increase in capacity since the capacity is logarithmic in SINR. This will greatly deteriorate the energy efficiency of the system.

3. Optimization approach

In this section, we start with a standard dual-based approach to decompose the energy-efficient optimization to congestion control and power control subproblems. We show that the congestion control subproblem can be solved in a closed form; we then propose a new method to solve the power control subproblem which converges fast and requires no a priori tuning on control parameters.

3.1. Standard approach for problem solving

P1 appears to be a non-convex programming problem, since the link constraints (3) correspond to a non-convex region. However, a simple variable transformation $\tilde{P}_l = \log(P_l)$ can be used to transform the problem into an equivalent convex optimization problem,

$$\mathbf{P2}: \text{maximize} \quad \sum_s U_s(x_s) - \beta \sum_l \tilde{V}_l(\tilde{P}_l)$$

$$\text{subject to} \quad \sum_{s \in S(l)} x_s \leqslant \tilde{c}_l(\tilde{\mathbf{P}}), \quad \forall l, \tag{7}$$

$$x_s \geqslant 0, \ \forall s \quad \text{and} \quad \tilde{P}_l \leqslant \tilde{P}_l^{\max}, \ \forall l, \tag{8}$$

where $\tilde{\mathbf{P}}$ is the vector of $\tilde{P}_l$, and $\tilde{V}_l(\tilde{P}_l)$ and $\tilde{c}_l(\tilde{\mathbf{P}})$ denote the corresponding transformations of $V_l(P_l)$ and $c_l(\mathbf{P})$. Notice that $\tilde{c}_l(\tilde{\mathbf{P}}) = \log(G_{ll}) + \tilde{P}_l - \log(\sum_{k \neq l} G_{k,l} \exp(\tilde{P}_k) + n_l)$ is the summation of a linear function and a negative log–sum–exp function. It is easy to prove that $\tilde{c}_l(\tilde{\mathbf{P}})$ is strictly convex in $\tilde{\mathbf{P}}$. For the sake of simplicity, we will use $\mathbf{x} \in \mathbf{X}$ and $\tilde{\mathbf{P}} \in \mathbf{Z}$ to denote the two sets of constraints (8).

Define the Lagrangian as

$$L(\mathbf{x}, \tilde{\mathbf{P}}, \lambda) = \sum_s U_s(x_s) - \beta \sum_l \tilde{V}_l(\tilde{P}_l) - \sum_l \lambda_l \left(\sum_{s \in S(l)} x_s - \tilde{c}_l(\tilde{\mathbf{P}}) \right),$$

where λ is the vector of Lagrange multipliers λ_l associated with capacity constraints (7). λ_l's are called dual variables (or *link prices* in the literature of NUM), while P_l's and x_s's primal variables. The dual problem P2 can be formulated as follows:

$$\mathbf{D2}: \min_{\lambda \geqslant 0} D(\lambda),$$

where the dual function $D(\lambda)$ is given by

$$D(\lambda) = \max_{\mathbf{x} \in \mathbf{X}, \tilde{\mathbf{P}} \in \mathbf{Z}} L(\mathbf{x}, \tilde{\mathbf{P}}, \lambda) = \max_{\mathbf{x} \in \mathbf{X}} L_1(\mathbf{x}, \lambda) + \max_{\tilde{\mathbf{P}} \in \mathbf{Z}} L_2(\tilde{\mathbf{P}}, \lambda). \tag{9}$$

The first term in (9) is formulated as

$$\max_{\mathbf{x} \in \mathbf{X}} L_1(\mathbf{x}, \lambda) = \sum_s \max_{x_s \geqslant 0} [U_s(x_s) - x_s \lambda^s], \tag{10}$$

where we denote $\lambda^s := \Sigma_{l \in L(s)} \lambda_l$. Notice that λ_l can be interpreted as the price one has to pay for the usage of link l. Then, λ^s is the total price for the route the source data s traverses. The optimization of (10) can be used to regulate the flow rate at each source. The second term in (9) is formulated as

$$\max_{\tilde{\mathbf{P}} \in \mathbf{Z}} L_2(\tilde{\mathbf{P}}, \lambda) = \max_{\tilde{\mathbf{P}} \in \mathbf{Z}} \sum_l [\lambda_l \tilde{c}_l(\tilde{\mathbf{P}}) - \beta \tilde{V}_l(\tilde{P}_l)]. \tag{11}$$

Notice that this term aims to maximize the weighted sum capacity less the power cost given fixed link prices. Thus, Eq. (11) serves as a tool for power control at each individual link.

It is clear that the energy efficiency issue can be explicitly dealt with in (11). With the positive tradeoff factor β, we can obtain an optimal balance between the link capacity $\tilde{c}_l(\tilde{\mathbf{P}})$ and the power cost $\tilde{V}_l(\tilde{P}_l)$. This provides a useful tool in trading off the network utility of flow rates with the energy efficiency of the system.

Problem D2 can be solved by minimization of the dual function $D(\lambda)$, and the maximization of the two sub-problems (10) and (11). First, note that Lagrangian $L(\mathbf{x}, \tilde{\mathbf{P}}, \lambda)$ is a strictly concave function of $(\mathbf{x}, \tilde{\mathbf{P}})$. According to [16, Prop. 6.1.1], $D(\lambda)$ is continuously differentiable everywhere and its unique subgradient

$$\frac{\partial D(\lambda)}{\partial \lambda_l} = \tilde{c}_l(\tilde{\mathbf{P}}(\lambda)) - \sum_{s \in S(l)} x_s(\lambda)$$

is indeed its derivative. Therefore, the minimization of the dual function $D(\lambda)$ can be done by a steepest descent method

$$\lambda_l(t+1) = \left[\lambda_l(t) + \gamma_1 \left(\sum_{s \in S(l)} x_s(t) - c_l(P(t)) \right) \right]^+, \tag{12}$$

where $[x]^+ = \max\{0, x\}$, and γ_1 is a constant step size.

Second, given λ, both (10) and (11) are strictly convex programming and therefore each has a unique solution. Denote the respective solution as $x_s^*(\lambda)$ and $\tilde{P}_l^*(\lambda)$. According to the first-order necessary optimality condition, we have the solution to (10):

$$x_s^*(\lambda) = (\lambda^s / p_s)^{-1/\alpha_s}. \tag{13}$$

$\tilde{P}_l^*(\lambda)$ does not have a closed-form expression and are usually solved by a steepest descent method, as used in the JOCP algorithm proposed in [4]. As in JOCP, we can write the gradient of $L_2(\tilde{\mathbf{P}}, \lambda)$ with respective to $\mathbf{P}$ as follows:

$$\frac{\partial L_2(\tilde{\mathbf{P}}, \lambda)}{\partial P_l} = \frac{\lambda_l}{P_l} - \left(\sum_{j \neq l} \frac{\lambda_j G_{l,j}}{m_j(\mathbf{P})} + \beta w_l \right), \quad \forall l, \tag{14}$$

where

$$m_j(\mathbf{P}) = \sum_{k \neq j} G_{k,j} P_k + n_j \tag{15}$$

represents the interference level at the receiver of link j. Thus, $\tilde{P}_l^*(\lambda)$ or $P_l^*(\lambda)$ can be updated as follows:

$$P_l(t+1) = \left[P_l(t) + \gamma_2 \left(\frac{\lambda_l(t)}{P_l(t)} - \left(\sum_{j \neq l} \frac{\lambda_j(t) G_{l,j}}{m_j(\mathbf{P}(t))} + \beta w_l \right) \right) \right]_l^{\pm}, \quad (16)$$

where γ_2 is the step size of small value and where $[x]_l^{\pm} = \min\{\max\{0, x\}, P_l^{\max}\}$.

3.2. A new approach for power control

The steepest decent algorithm (14), or the ones used in [4], suffers from slow convergence, especially when the step size γ_2 is not appropriately selected. An optimal γ_2 in general can be found for a certain fixed network scenario through extensive simulation trials. However, it becomes difficult when the algorithm needs to be applied on-the-fly since it takes a certain amount of training time to unearth the optimal step size. It becomes even more challenging when the network scenario varies over time. This motivates us to study a novel approach which is as simple as the steepest decent algorithm, but converges faster without any parameter tuning.

Proposition 2. *The power control subproblem* (11) *can be solved by alternately updating transmit powers of all links* ($l = 1, 2, \ldots, L$) *as follows*:

$$P_l(t+1) = \left[\lambda_l(t) \left(\sum_{j \neq l} \frac{\lambda_j(t) G_{l,j}}{m_j(\mathbf{P}^l(t))} + \beta w_l \right)^{-1} \right]_l^{\pm}, \quad (17)$$

where

$$\mathbf{P}^l(t) = (P_1(t+1), \ldots, P_{l-1}(t+1), P_l(t), P_{l+1}(t), P_L(t)) \quad (18)$$

is a vector of newly updated transmit powers of links 1 *to* $l-1$ *at iteration* $t+1$ *and the previous transmit powers of links* l *to* L *at iteration* t.

Proof. The convergence of expression (17) is proved in Appendix B by showing that it increases the upper-bounded objective function $L_2(\tilde{\mathbf{P}}, \lambda)$ at each iteration. □

It is worthwhile to provide some insight into (17). Let us write the derivative of $L_2(\tilde{\mathbf{P}}, \lambda)$ with respect to $\tilde{\mathbf{P}}$ as follows:

$$\frac{\partial L_2(\tilde{\mathbf{P}}, \lambda)}{\partial \tilde{P}_l} = \lambda_l - \left(\sum_{j \neq l} \frac{\lambda_j G_{l,j}}{m_j(\mathbf{P})} + \beta w_l \right) P_l, \quad \forall l. \quad (19)$$

It is clear that Eq. (17) implicitly solve the equations $\partial L_2(\tilde{\mathbf{P}}, \lambda)/\partial \tilde{P}_l = 0$ in a recursive manner. This is similar to the well-known power control algorithm in [17, Eq. (18)], in the sense that the latter also uses a recursive method to solve a set of power control equations.

The new method does not involve a step size in optimizing the power control subproblem and is thus robust in various network scenarios. Also, it benefits from the simple expression, which is comparable to the steepest decent method in (16). This enables a distributed implementation by exchanging messages $\lambda_j/m_j(\mathbf{P})$ among interfering links. Note that $m_j(\mathbf{P})$, as defined in (15), represents the interference level at the receiver of link j and therefore can be easily measured. What is more important, similar to the power control algorithm in [17], the new method converges much faster than the steepest descent method does as shown in our simulation (see Section 5 for details). In addition, all transmit powers are alternately (not simultaneously) updated in (17) and the sequence of the update does not affect the convergence of the method. The alternate update may be beneficial in a wireless network which provides only limited-quality time synchronization. However, it is worth more investigation to see how the performance of the algorithm is affected in such a network as the quality of time synchronization is degraded.

Based on the above analysis, we propose an iterative algorithm named energy efficient jointly optimal congestion and power control (EJOC), to solve the dual problem D2.

Algorithm EJOC.

1. At each link l
   (a) Update the link price $\lambda_l(t+1)$ according to (12), and communicate it to all sources using link l.
   (b) Measure the interference level, $m_l(\mathbf{P})$, at the receiver of link l, and broadcast $\lambda_l/m_l(\mathbf{P})$ to all interfering nodes in the neighborhood.
   (c) Receive messages $\lambda_j/m_j(\mathbf{P})$ from all interfering nodes j in the neighborhood and repeat the power update process (17) K times.
2. At each source s
   (a) Receive from the reverse path the sum ($\lambda^s = \sum_{l \in L(s)} \lambda_l$) of link prices λ_l.
   (b) Compute the new flow rate $x_s(t+1)$ using (13), and Communicate it to all links on its path.
3. Repeat Steps 1 and 2 until the flow rates and the transmit powers converge.

In EJOC, the iteration proceeds in two timescales: K primal variable updates are executed after each dual variable update. A large K offers convergence with very small residual error to the optimal point, but is likely to slow down the convergence of the entire algorithm. Hence, a small K is desired as long as the residual error is acceptable. Simulation results show that $K = 1$ renders good

performance due to the fast convergence of the proposed recursive method (17).

4. Multipath routing

In this section, we extend our analysis to the situation where multiple alternative paths may be established between some source–destination pairs, and the sources are able to explicitly adjust the traffic rates on these paths. When multiple paths are available, the source traffic is simply split into these paths without any redundancy. We do not consider balancing the traffic among the multiple paths. As a complementary technique to what is considered in this paper, interested readers may refer to [18] where energy efficiency problem is dealt with by spreading traffic and energy burdens over multiple paths.

In the following, we formulate the energy-efficient network utility maximization for networks with multipath routing. We tackle the instability problem caused by the multipath routing. In addition, we show that the NUM framework lends itself to an optimal water-filling rate allocation among multiple paths within each source–destination pair.

Assume each source s has a set $\pi(s)$ of alternative paths to the destination, each of which is composed of a sequence of connected links $l \in L(s, p)$, for $p \in \pi(s)$. Denote $SP(l)$ as the set of all source–path pairs (s, p) that pass through link l. Notice that $l \in L(s, p)$ if and only if $(s, p) \in SP(l)$. Let $x_{s,p}$ be the flow rate on each path; the total traffic rate x_s originated from source s is equal to $\sum_{p\in\pi(s)} x_{s,p}$. Similar to P2, the problem can be formulated as follows:

$$\textbf{P3}: \text{maximize} \quad \sum_s U_s\left(\sum_{p\in\pi(s)} x_{s,p}\right) - \beta\sum_l \tilde{V}_l(\tilde{P}_l)$$

$$\text{subject to} \quad \sum_{(s,p)\in SP(l)} x_{s,p} \leqslant \tilde{c}_l(\tilde{\mathbf{P}}), \quad \forall l, \tag{20}$$

$$x_{s,p} \geqslant 0, \ \forall s, p \quad \text{and} \quad \tilde{P}_l \leqslant \tilde{P}_l^{\max}, \ \forall l. \tag{21}$$

Notice that, different from that of P2, the objective function of P3 is not strictly concave once some sources have multiple alternative paths. This may cause an instability problem in the convergence of an iterative algorithm – a persistent oscillation of the flow rate around the optimal value. This implies that although the dual variables may converge, the more important primal variables, flow rates and transmit powers, may not. Such behavior was observed in wired networks [10]. To deal with this instability problem, we borrow an idea from proximal optimization algorithms [13, p. 232]. That is, instead of P3, we try to solve an equivalent problem by introducing a quadratic term of some auxiliary variables $y_{s,p}$ so that the optimization problem becomes strictly concave with respect to $x_{s,p}$,

$$\textbf{P4}: \text{maximize} \quad \sum_s U_s\left(\sum_{p\in\pi(s)} x_{s,p}\right) - \beta\sum_l \tilde{V}_l(\tilde{P}_l) - \frac{c}{2}\sum_s I_s \sum_{p\in\pi(s)} (x_{s,p} - y_{s,p})^2$$

$$\text{subject to} \quad (20) \text{ and } (21),$$

where c is a positive constant, I_s is used to indicate whether source s has multiple alternative paths or not, i.e.,

$$I_s = \begin{cases} 0, & |\pi(s)| = 1, \\ 1, & |\pi(s)| \geqslant 2, \end{cases}$$

where $|\cdot|$ denotes the cardinality of the set.

Lagrangian of P4 can be written as

$$L'(\mathbf{x}, \mathbf{y}, \tilde{\mathbf{P}}, \lambda) = \sum_s U_s(x_s) - \frac{c}{2}\sum_s I_s \sum_{p\in\pi(s)} (x_{s,p} - y_{s,p})^2 - \beta\sum_l \tilde{V}_l(\tilde{P}_l) - \lambda_l\left[\sum_{(s,p)\in SP(l)} x_{s,p} - \tilde{c}_l(\tilde{\mathbf{P}})\right],$$

where $\mathbf{x}$ and $\mathbf{y}$ are the vectors of $x_{s,p}$ and $y_{s,p}$, respectively.

The dual problem P4 is formulated as follows:

$$\textbf{D4}: \min_{\lambda\geqslant 0} D'(\lambda),$$

where the dual function $D'(\lambda)$ is given by

$$D'(\lambda) = \max_{\mathbf{x}\in\mathbf{X}, \tilde{\mathbf{P}}\in\mathbf{Z}} L'(\mathbf{x}, \mathbf{y}, \tilde{\mathbf{P}}, \lambda) = \max_{\mathbf{x}\geqslant 0, \mathbf{y}} L_1'(\mathbf{x}, \mathbf{y}, \lambda) + \max_{\tilde{\mathbf{P}}\in\mathbf{Z}} L_2(\tilde{\mathbf{P}}, \lambda), \tag{22}$$

where the second optimization term in (22) is the same as that defined in (11), and therefore can be solved by the method proposed in (17). The first optimization in (22) is given by

$$\max_{\mathbf{x}\geqslant 0, \mathbf{y}} L_1'(\mathbf{x}, \mathbf{y}, \lambda) = \sum_s \max_{x_s\geqslant 0, y_s}\left[\sum_s U_s\left(\sum_{p\in\pi(s)} x_{s,p}\right) - \sum_{p\in\pi(s)}\left(x_{s,p}\lambda^{s,p} + \frac{c}{2} I_s (x_{s,p} - y_{s,p})^2\right)\right], \tag{23}$$

where $\lambda^{s,p} = \sum_{l\in L(s,p)} \lambda_l$ and $\mathbf{x}_s$ and $\mathbf{y}_s$ are the vectors of $x_{s,p}$ and $y_{s,p}$ for $p \in \pi(s)$, respectively.

If source s has a single path (i.e., $I_s = 0$), the solution to (23) is similar to (13), i.e.,

$$x_s^*(\lambda) = x_{s,1}^*(\lambda) = (\lambda^{s,1}/p_s)^{-\alpha_s}. \tag{24}$$

Otherwise, the optimization (23) can be solved by a nonlinear Gauss–Seidel method which alternately (i) maximizes

the objective function over $\mathbf{x}_s$ while keeping $\mathbf{y}_s$ fixed, then (ii) maximizes it over $\mathbf{y}_s$ while keeping $\mathbf{x}_s$ fixed, and repeats. As is proved in Appendix B, the maximization of (i) has a water-filling solution:

$$x_{s,p}(t+1) = \frac{1}{c}[p_s v^{-\alpha_s} - \lambda^{s,p} + c y_{s,p}(t)]^+, \tag{25}$$

where v is the unique solution to the equation $f(v)=0$ with

$$f(v) = v - \sum_{p\in\pi(s)} [p_s v^{-\alpha_s} - \lambda^{s,p} + c y_{s,p}(t)]^+$$

being a strictly increasing function. Note that $f(0^+)=-\infty$, and $f(+\infty)=+\infty$. Hence the equation can be solved efficiently by one-dimensional line search methods. The solution to maximization (ii) is merely

$$y_{s,p}(t+1) = x_{s,p}(t+1). \tag{26}$$

Similar to EJOC, we propose the following algorithm EJOC*m* to solve the multipath routing problem.

Algorithm EJOC*m*.

1. At each link l, the operations are the same as EJOC except that the link price $\lambda_l(t+1)$ is updated as follows:
$$\lambda_l(t+1)=\lambda_l(t)+\gamma_2\left[\sum_{p\in\pi(s)} x_{s,p}(t)-c_l(\mathbf{P}(t))\right]^+. \tag{27}$$
In addition, if $\mathrm{SINR}_l = 1$ in the last $K_1 = 50$ iterations, set $P_l = 0$ and declare link l as inactive.
2. At each source s
   (a) Receive from the reverse path the sum $\lambda^{s,p} = \sum_{l\in L(s,p)} \lambda_l$ of link prices λ_l.
   (b) Update the flow rate $x_{s,p}(t+1)$ using (24) or (25).
   (c) Communicate $x_{s,p}(t+1)$ to all links on path p.
   (d) In addition, if $x_{s,p} = 0$ in the last $K_2 = 50$ iterations, declare path (s, p) as inactive.
3. Repeat Steps 1 and 2 until the flow rates and the transmit powers converge.

It is worthwhile to provide some explanation of the link- and path-removal procedures of EJOC*m* (i.e., the declaration of a link or a path as inactive). Any solution to P4 or D4 must satisfy $\mathrm{SINR}_l \geqslant 1$ so that $c_l(\mathbf{P}) \geqslant 0$. The links with $\mathrm{SINR}_l = 1$ have zero capacity but are allocated non-zero powers. Thus, we can remove these links and still guarantee the feasibility of the solution. In addition, the removal of these links reduces the interference level at other links and contributes to higher link capacities. Hence, the link- and path-removal procedures is capable of improving the algorithm's performance.

5. Simulation investigation

In this section, we investigate the performance of the proposed iterative algorithms in some sample wireless networks. For simplicity, we present our simulation results for a dumbbell-shaped network shown in Fig. 1. This network is simple yet useful to model the usual bottleneck situation and multipath routing in a network. There are six nodes, A–F, and five wireless links A–C, B–C, C–D, D–E, and D–F for the single-path routing or two additional links A–E and B–F in the case of multipath routing. These nodes and links are named as node 1, 2, …, 6, and link 1, 2, …, 7, respectively. The single-path routing problem is analyzed in Sections 5.1–5.3, where paths 1–3 in Table 1 are used for data transmission. The multipath routing problem is then addressed in Section 5.4, where all the five paths in Table 1 are considered.

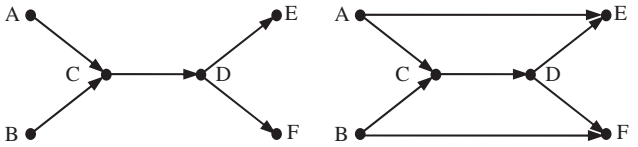

Fig. 1. Logical topology and connections for an illustrative example. The coordinates of nodes A–F are $(-(\sqrt{3}+1)/2, 1)$, $(-(\sqrt{3}+1)/2, 1)$, $(-1, 0)$, $(1, 0)$, $((\sqrt{3}+1)/2, 1)$, and $((\sqrt{3}+1)/2, -1)$, respectively.

Table 1. The source, sink, and path of the flows in Fig. 1.

Path/flow	Source	Destination	Links on the path
1	A	E	A–C, C–D, D–E
2	B	F	B–C, C–D, D–F
3	C	E	C–D, D–E
4	A	E	A–E
5	B	F	B–F

In our simulation, we verified that the proposed algorithms work well for different network configurations such as those in [4,12] and that the proposed distributed algorithm achieves the same optimum solution as the MATLAB solver *fmincon*[1] does. Due to space limitation, our discussion focuses on the network shown in Fig. 1.

If not otherwise specified, the simulation scenario is set up as follows. The system is initialized by setting all transmit powers P_l to be 0.1, and the link prices λ_l to be 1. These link prices are used to calculate the initial flow rate using (13). It is assumed that the maximal transmit power P_l^{max} of each link is 1, and the noise power n_l at each receiver 0.001. The parameters p_s and w_l are set to be 2 and 1, respectively. All the results reported below are obtained by averaging over 100 independent simulations, in which the channel fading between any two nodes of distance d is randomly generated according to a uniform distribution on $[0.2d^{-4}, 2d^{-4}]$.

The *energy efficiency* of the system is defined as the ratio of the total flow rate to the total transmit power.

[1] Mathwork Inc. http://www.mathwork.com.

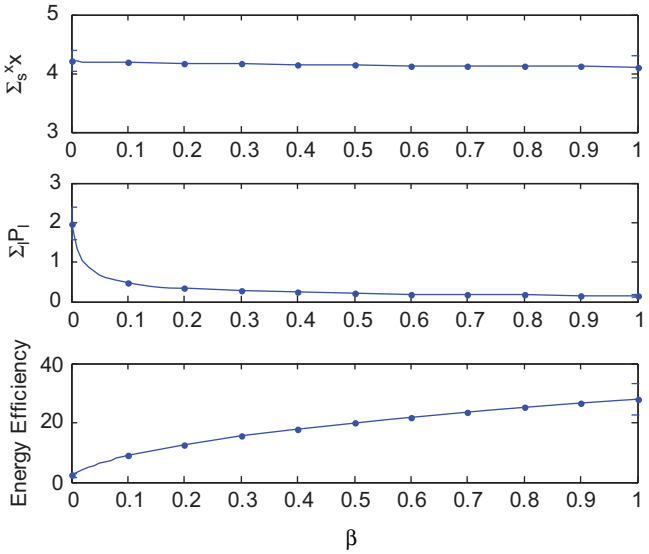

Fig. 2. The effect of the tradeoff factor β ($\alpha_s = 1,\ p_s = 2,\ w_l = 1$). From top to bottom, the three graphs show the total flow rate, the total transmit power, and the corresponding energy efficiency, respectively. The curve shows the mean and the standard error calculated based on 100 independent experiments.

5.1. The effect of β on the energy efficiency

As discussed in Section 2, the tradeoff factor β plays an important role in improving the energy efficiency of the system while maintaining the network throughput at a high level. The comparison of different β is illustrated in Fig. 2. It is interesting to note that as β increases from 0 to 1, the total flow rate decreases by 2.28% while the total transmit power goes down dramatically from 1.97 to 0.15. Accordingly, the energy efficiency of the system goes up from 2.24 to 28.02.

For further investigation, we plot the typical convergence behaviors of EJOC when $\beta = 0$ or 0.1 in plots (a)–(d) of Fig. 3. The channel fading is assumed to be d^{-4} in both cases. It is clear that the flow rates converge in a similar way to the optimum after 40 iterations. The convergence of transmit powers, however, are quite different in the two cases. When $\beta = 0$, all transmit powers keep (roughly proportionally) increasing from 40 to 100 iterations until link 3 reaches its limit, i.e., $P_3 = 1$. This is consistent with the statement of Proposition 1. Recall the expression of the link capacity (2). It is clear a roughly proportional increase in all transmit powers can increase link capacities. The capacity increase, however, is getting more and more marginal due to its logarithmic expression (2). As a result, the energy efficiency is significantly degraded due to the steady increase in transmit powers. On the other hand, when $\beta = 0.1$, each transmit power converge to a moderate level after 40 iterations. This leads to much higher energy efficiency since the total flow rates are almost the same for $\beta = 0$ and 0.1.

5.2. The effect of α_s

Now, let us consider the effect of various utility functions which are defined as a family (6) and differ from one another by the factor α_s. As is known, this factor has a strong affect on the fairness of the optimal flow rates. For the sake of quantitative comparison, we define the fairness index as $f(x) = E_s(x_s)^2/E_s(x_s^2)$ [19], where $E_s(x_s)$ and $E_s(x_s^2)$ denote the average values of x_s and x_s^2, respectively. The fairness index ranges from 0 to 1 and a higher value implies a higher degree of fairness.

The simulation results are presented in Table 2 and Fig. 4. It is clear from Table 2 that the higher the factor α_s, the higher the fairn ess among the flows rates. It leads to nearly identical flow rates when α_s is equal to 10. Fig. 4 shows that α_s has some effect on the energy efficiency of the network: the energy efficiency is improved because the transmit powers decrease faster than the flow rates do. It is not always the case, however. For example, if $\beta = 0$, the energy efficiency tends to slightly decrease as α_s goes up. This is because the total transmit power remains at a high level if β=0 (Proposition 1), while the total flow rate steadily decreases as α_s increases.

5.3. Comparison of power control algorithms

In practical applications, we prefer an algorithm which can be easily implemented in a distributed manner and converge fast. In this subsection, we validate the effectiveness of the simultaneous primal and dual updates of EJOC by setting $K = 1$, and compare EJOC to the JOCP algorithm proposed in [4].

Denote $L_2(\mathbf{P}^K, \lambda)$ as the objective function value L_2 of power control sub-problem in Eq. (11) after K primal updates, and $\mathrm{RRE}(K) = |L_2(\mathbf{P}^K, \lambda) - L_2(\mathbf{P}^*(\lambda), \lambda)|/|L_2(\mathbf{P}^*(\lambda), \lambda)|$ as the relative residual error (RRE). In Fig. 5, we plot the RRE after different number of dual updates. It is clear

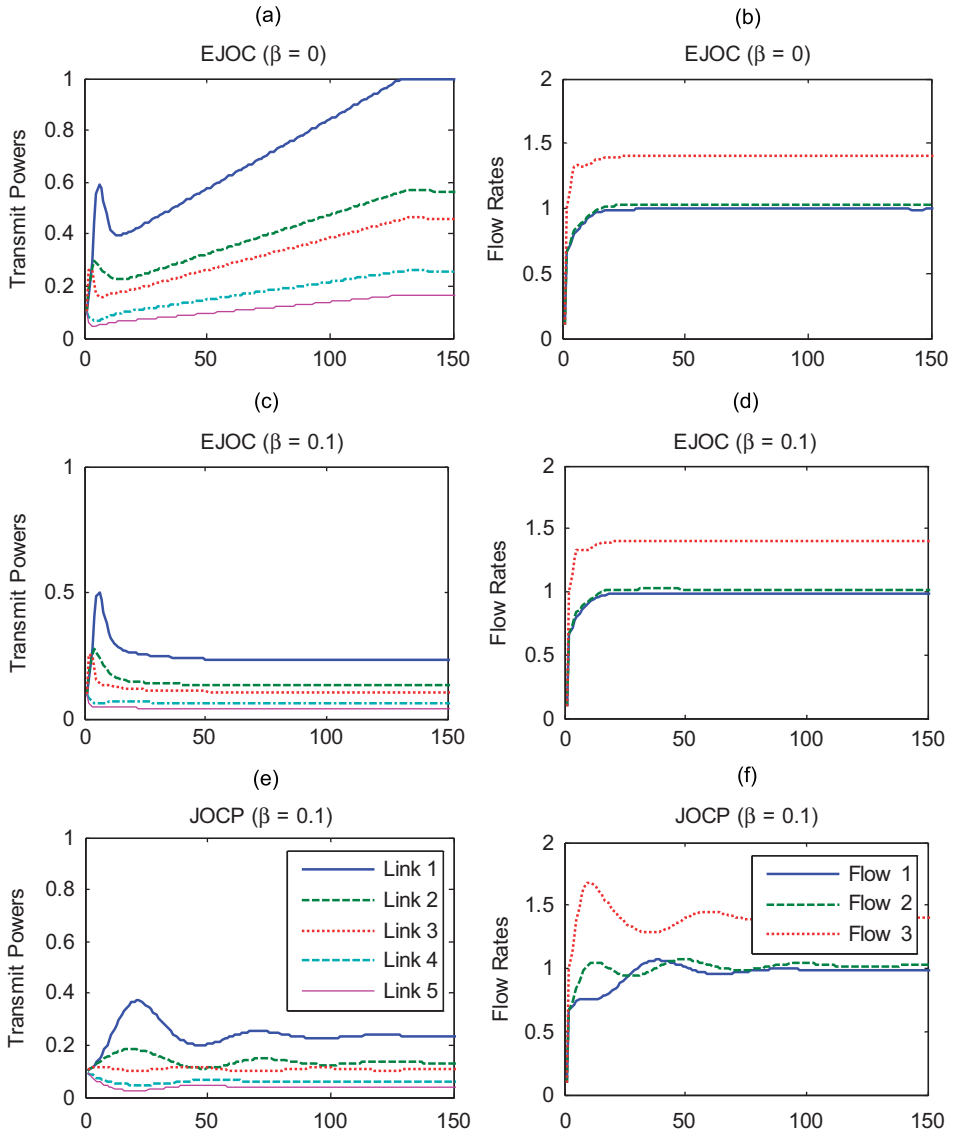

Fig. 3. The convergence behavior of joint congestion and power control ($\alpha_s = 1,\ p_s = 2,\ w_l = 1$). The x-axis shows the number of iterations (i.e., the number of dual updates). Plots (a)–(d) show simulation results obtained by EJOC when $\beta = 0$ and 0.1, respectively. Plots (e) and (f) are simulation results obtained by the steepest descent algorithm of JOCP proposed in [4] when $\beta = 0.1$.

Table 2. The effect of the factor α_s on the flow rates ($\beta = 0.1,\ p_s = 2,\ w_l = 1$).

α_s	x_1 mean (std)	x_2 mean (std)	x_3 mean (std)	$\Sigma_s x_s$ mean (std)	Fairness mean (std)
0	0.00 (0.0e+0)	0.71 (3.7e−1)	3.67 (3.8e−1)	4.38 (1.5e−1)	0.46 (7.0e−1)
1	1.28 (1.0e−1)	1.38 (9.3e−2)	1.54 (5.0e−2)	4.18 (2.0e−1)	0.99 (4.7e−3)
2	1.33 (8.6e−2)	1.38 (7.9e−2)	1.46 (5.8e−2)	4.17 (2.1e−1)	1.00 (1.4e−3)
5	1.35 (6.8e−2)	1.37 (6.5e−2)	1.41 (5.6e−2)	4.14 (1.9e−1)	1.00 (2.6e−4)
10	1.34 (6.0e−2)	1.35 (5.9e−2)	1.37 (5.4e−2)	4.06 (1.7e−1)	1.00 (6.6e−4)

that RRE can be reduced to 10^{-4} after a few dual updates even with $K = 1$, which is attributed to recursive update of the proposed method (17) of EJOC. A larger K can further reduce the RRE but will inevitably lower the convergence rate of the algorithm. It is thus reasonable to set $K = 1$ in EJOC.

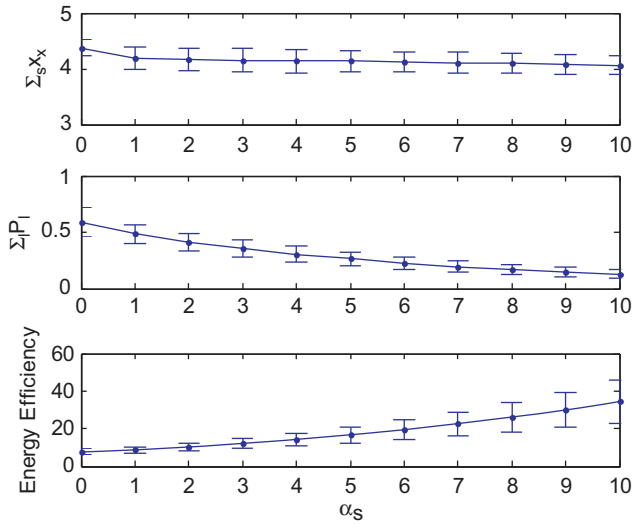

Fig. 4. The effect of the fairness factor α_s (β=0.1, p_s=2, w_l=1). From top to bottom, the three graphs show the total flow rate, the total transmit power, and the corresponding energy efficiency, respectively. The curve shows the mean and the standard error calculated based on 100 independent experiments.

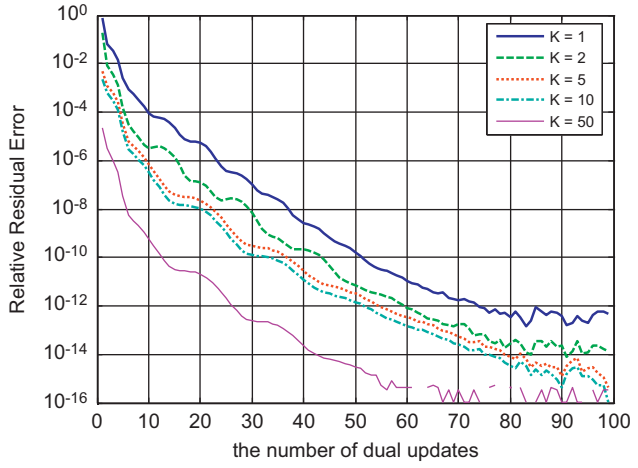

Fig. 5. The relative residual error vs. the number (K) of primal updates between any two successive dual updates in EJOC.

We next compare EJOC to the JOCP algorithm [4] where the steepest descent method is applied to update the transmit powers. The convergence behaviors of EJOC and JOCP are shown in plots (c)–(e) in Fig. 3. In JOCP, the step size of the steepest descent method is set to 0.1 to update the transmit powers. This value leads to fast convergence according to our experiments. However, even with the best step size, JOCP converges clearly slower than EJOC. Thus, EJOC has shown its superiority to JOCP for its faster convergence and no requirement of parameter training, which are beneficial in wireless networks of different time-variant characteristics.

We note that each iteration of EJOC needs end-to-end communication, and therefore requires a few round-trip times (RTTs). Each RTT is typically in the order of millisecond to tens of milliseconds. Considering that EJOC usually converges in less than 100 iterations, the overall convergence time of the algorithm is less than 1 s for this simple network.

5.4. Multi-path routing

To simulate the multipath routing problem, we use all the five paths listed in Table 1, i.e., both sources A and B have two alternative paths while source C has a single path. The simulation results are summarized in Table 3. Also, to illustrate the effect of the path- and link-removal procedure introduced in EJOC*m*, we consider an experiment where the procedure is disabled at the beginning of optimization and then enabled it after 100 iterations.

As is shown in Fig. 6, EJOC*m* converges to a small neighborhood of the optimum value in 50 iterations. At the optimum point, the SINR_1 is equal to one (and thus the capacity of link 1 is zero) because of the high interference level at the receiver of link 1. Implemented with the path- and link-removal procedures after the 100-th iteration, EJOC*m* removes this link from further operation. This reduces interference levels at the receivers of other links, and thus leads to higher link capacities and flow rates. Averaged over 100 simulations, the results in Table 3 indicate 11.77% reduction of total transmit power and 25.39% improvement of total flow rate. The energy efficiency is thus increased by 42.11% from 6.33 to 9.00.

6. Conclusion

In this paper, we have considered energy-efficient utility maximization in wireless networks by integrating both the network utility and the power cost in the objective function. A tradeoff factor is introduced to balance the network utility and the energy cost. We have shown that the proposed method significantly improves the energy efficiency of the system. For example, as the tradeoff factor increases from 0 to 1 in the investigated scenario, the network throughput has a slight decrease of 2.28% while the total transmit power is dramatically reduced by 92.39%. Accordingly, the energy efficiency increases from 2.24 to 28.02.

We have solved this optimization problem by decomposing it into two sub-problems. To tackle the power control sub-problem, we have proposed a new iterative method which can be implemented as easily as the steepest descent method but converges much faster than the latter. To solve the congestion control sub-problem which no longer has a closed-form solution in the multipath routing case, we have derived a water-filling solution which simplifies the operation at each data source. The entire algorithm has been

Table 3. A comparison of EJOC*m* (a) without or (b) with the path- and link-removal procedures ($\alpha_s = 1,\ \beta = 1,\ p_s = 2,\ w_l = 1$).

Link	Average Tx power mean (std)		Flow	Average flow rate mean (std)	
	(a)	(b)		(a)	(b)
1	0.05 (2.5e−2)	0.01 (2.0e−2)	1	0.04 (1.2e−1)	0.02 (6.9e−2)
2	0.06 (2.5e−2)	0.03 (3.0e−2)	2	0.20 (2.5e−1)	0.23 (3.5e−1)
3	0.02 (8.3e−3)	0.02 (9.1e−3)	3	1.05 (2.3e−1)	1.23 (2.3e−1)
4	0.00 (2.7e−3)	0.00 (2.9e−3)	4	1.00 (3.2e−1)	1.33 (3.2e−1)
5	0.00 (1.4e−3)	0.00 (1.7e−3)	5	0.93 (4.7e−1)	1.24 (6.4e−1)
6	0.19 (6.1e−2)	0.21 (6.3e−2)			
7	0.19 (6.4e−2)	0.19 (8.0e−2)			
Total	0.51 (9.7e−2)	0.45 (8.8e−2)	Total	3.23 (6.9e−1)	4.05 (8.1e−1)

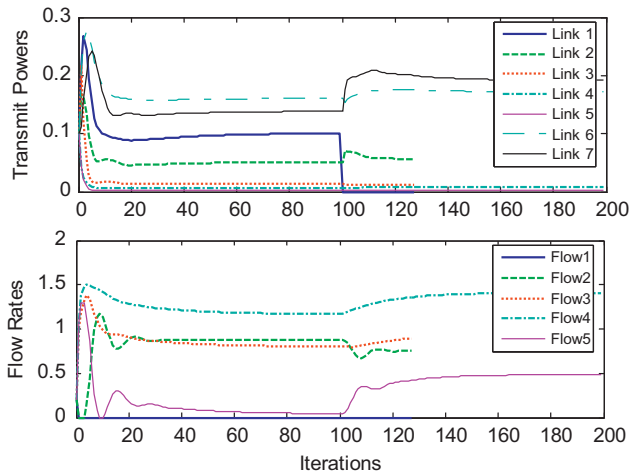

Fig. 6. An illustrative example of EJOC*m*. The path- and link-removal procedures are disabled at the beginning and enabled at the 100-th iteration.

implemented in a distributed manner and simulation results have been provided to confirm our claims.

Acknowledgments

We would like to thank anonymous reviewers for their valuable comments and suggestions.

Appendix A. Proof of convergence (17)

Lemma 3. *Suppose* λ_l*'s are all positive. Let* $\mathbf{P}(t)$ *be the sequence generated by* (17), i.e.,

$$P_l(t+1) = \left[\lambda_l(t)\left(\sum_{j\neq l}\frac{\lambda_j(t)}{P_l(t)+n'_j}+\beta w_l\right)^{-1}\right]_l^{\pm}, \quad (A.1)$$

where

$$n'_j = G_{l,j}^{-1}\left(\sum_{k\neq j,k<l} G_{k,j}P_k(t+1) + \sum_{k\neq j,k>l} G_{k,j}P_k(t) + n_j\right).$$

We have

$$L_2(\tilde{\mathbf{P}}^1(t),\lambda) \leqslant L_2(\tilde{\mathbf{P}}^2(t),\lambda) \leqslant \cdots \leqslant L_2(\tilde{\mathbf{P}}^L(t),\lambda), \quad (A.2)$$

where

$$\tilde{\mathbf{P}}^l(t) = (\tilde{P}_1(t+1)\cdots\tilde{P}_{l-1}(t+1),\ \tilde{P}_l(t), \tilde{P}_{l+1}(t)\cdots\tilde{P}_L(t))^{\mathrm{T}} \quad (A.3)$$

and the equalities simultaneously hold if and only if $\tilde{\mathbf{P}}(t) = \tilde{\mathbf{P}}^*$.

Proof. Let

$$g_l(\xi) = L_2((\tilde{P}_1(t+1)\cdots\tilde{P}_{l-1}(t+1),\ \xi, \tilde{P}_{l+1}(t)\cdots\tilde{P}_L(t))^{\mathrm{T}},\lambda).$$

That is, $g_l(\tilde{P}_l(t)) = L_2(\tilde{\mathbf{P}}^l(t),\lambda)$ and $g_l(\tilde{P}_l(t+1)) = L_2(\tilde{\mathbf{P}}^{l+1}(t),\lambda)$. Thus, inequalities (A.2) are equivalent to

$$g(\tilde{P}_l(t)) \leqslant g(\tilde{P}_l(t+1)), \quad l = 1,2,\cdots,L-1. \quad (A.4)$$

Before proceeding, we note that the derivative of $g(\xi)$

$$\nabla g_l(\xi) = \lambda_l - \left(\sum_{j\neq l}\frac{\lambda_j}{\exp(\xi)+n'_j}+\beta w_l\right)\exp(\xi) \quad (A.5)$$

is a strictly decreasing function of ξ.

The positiveness assumption on all λ_l's implies the bracketed term in (A.1) is always positive. Thus, it is sufficient for us to consider two cases: (i) $0 < P_l(t) \leqslant P_l(t+1) \leqslant P_l^{max}$, (ii) $0 < P_l(t+1) \leqslant P_l(t) \leqslant P_l^{max}$ and $P_l(t+1) < P_l^{max}$.

In the first case, we have

$$\nabla g_l(\tilde{P}_l(t+1)) = \lambda_l - \left(\sum_{j\neq l}\frac{\lambda_j}{P_l(t+1)+n'_j}+\beta w_l\right) P_l(t+1)$$
$$\geqslant \lambda_l - \lambda_l\left(\sum_{j\neq l}\frac{\lambda_j}{P_l(t+1)+n'_j}+\beta w_l\right) \times \left(\sum_{j\neq l}\frac{\lambda_j}{P_l(t)+n'_j}+\beta w_l\right)^{-1}$$
$$\geqslant \lambda_l - \lambda_l = 0, \tag{A.6}$$

where the first inequality is obtained by replacing $P_l(t+1)$ with the bracketed term in (A.1), and the second inequality is merely due to $P_l(t)\leqslant P_l(t+1)$. Thus, we have $\nabla g_l(\xi) > 0$ for $\xi < \tilde{P}_l(t+1)$, because $\nabla g_l(\xi)$ is a strictly decreasing function. Furthermore, $g_l(\xi)$ is strictly increasing for $\xi < \tilde{P}_l(t+1)$ and thus $(\tilde{P}_l(t+1))\geqslant g(\tilde{P}_l(t))$ with equality if and only if $\tilde{P}_l(t) = \tilde{P}_l(t+1)$.

In the second case, $P_l(t+1)$ is the bracketed term in (A.1). Thus, we have

$$\nabla g_l(\tilde{P}_l(t+1)) = \lambda_l - \left(\sum_{j\neq l}\frac{\lambda_j}{P_l(t+1)+n'_j}+\beta w_l\right) P_l(t+1)$$
$$= \lambda_l - \lambda_l\left(\sum_{j\neq l}\frac{\lambda_j}{P_l(t+1)+n'_j}+\beta w_l\right) \times \left(\sum_{j\neq l}\frac{\lambda_j}{P_l(t)+n'_j}+\beta w_l\right)^{-1}$$
$$\leqslant \lambda_l - \lambda_l = 0 \tag{A.7}$$

where the inequality follows from $P_l(t+1)\leqslant P_l(t)$. It is clear $\nabla g_l(\xi) < 0$ for $\xi > \tilde{P}_l(t+1)$, because $\nabla g_l(\xi)$ is strictly decreasing. Furthermore, $g_l(\xi)$ is strictly decreasing for $\xi > \tilde{P}_l(t+1)$ and thus $g(\tilde{P}_l(t+1))\geqslant g(\tilde{P}_l(t))$ with equality if and only if $\tilde{P}_l(t) = \tilde{P}_l(t+1)$.

In both cases, we arrive at (A.4) and the equality simultaneously holds for each l if and only if $\tilde{\mathbf{P}}(t) = \tilde{\mathbf{P}}(t+1)$, i.e., $\tilde{\mathbf{P}}(t) = \tilde{\mathbf{P}}^*$. To see this, let us consider two cases: (i) $\tilde{P}_l(t) = \tilde{P}_l(t+1) < \tilde{P}_l^{\max}$ and (ii) $\tilde{P}_l(t) = \tilde{P}_l(t+1) = \tilde{P}_l^{\max}$. In the first case, we can simply follow (A.7) to show that $\nabla g_l(\tilde{P}_l(t)) = \nabla g_l(\tilde{P}_l(t+1)) = 0$, while in the second case, we have $\nabla g_l(\tilde{P}_l(t))\geqslant 0$ according to (A.6). That is,

$$\nabla g_l(\tilde{P}_l(t))(\xi - \tilde{P}_l(t))\leqslant 0 \quad \text{for } \xi\leqslant \tilde{P}_l^{\max}.$$

Notice that when $\tilde{\mathbf{P}}(t) = \tilde{\mathbf{P}}(t+1)$, $\nabla g_l(\tilde{P}_l(t))$ is indeed $\nabla_l L_2(\tilde{\mathbf{P}}(t), \lambda)$. Thus, we have

$$\nabla_l L_2(\tilde{P}(t), \lambda)(\xi - \tilde{P}_l(t))\leqslant 0 \quad \text{for } \xi\leqslant \tilde{P}_l^{\max},$$

which is the sufficient condition for $\tilde{\mathbf{P}}(t) = \tilde{\mathbf{P}}^*$. □

Theorem 4. *The sequence of* $\mathbf{P}$(t) *generated by* (17) *converges to the unique maximizor of problem* (11)

$$\max_{\tilde{\mathbf{P}}\in\mathbf{Z}} L_2(\tilde{\mathbf{P}}, \lambda) = \max_{\tilde{\mathbf{P}}\in\mathbf{Z}} \sum_l \left[\lambda_l \tilde{c}_l(\tilde{\mathbf{P}}) - \beta \tilde{V}_l(\tilde{P}_l)\right]. \tag{A.8}$$

Proof. Let us first consider the non-trivial case where λ_l's are all positive. According to Lemma 3, we have

$$L_2(\tilde{\mathbf{P}}(t), \lambda)\leqslant L_2(\tilde{\mathbf{P}}(t+1), \lambda),$$

where $\tilde{\mathbf{P}}(t) = \tilde{\mathbf{P}}^1(t)$, $\tilde{\mathbf{P}}(t+1) = \tilde{\mathbf{P}}^L(t)$ and the equality holds if and only if $\tilde{\mathbf{P}}(t) = \tilde{\mathbf{P}}^*$. Since $L_2(\tilde{\mathbf{P}}, \lambda)$ is bounded from above for $\tilde{P}_l\leqslant \tilde{P}_l^{\max}$, the sequence of $L_2(\tilde{\mathbf{P}}(t), \lambda)$ converges to the unique optimum, and accordingly $\tilde{\mathbf{P}}(t)$ converges to the unique maximizor $\tilde{\mathbf{P}}^*$.

If λ_l is equal to zero for some l, it is easy to see from (A.8) that the corresponding P_l^* is equal to zero. This is achieved by (17) or (A.1) in one step, so we can simply consider all the links with $\lambda_l = 0$ to be inactive in the network. Note that $P_l^* = 0$ implicitly removes itself from expression (A.1). Thus, process (A.1) can still be used for other links with $\lambda_l > 0$ to reach the optimum. This completes the proof. □

Appendix B. Proof of Eq. (25)

The derivative of the objective function in (23) with respect to $x_{s,p}$ is given by

$$\frac{p_s}{(\sum_{p\in\pi(s)} x_{s,p})^{\alpha_s}} - \lambda^{s,p} + cy_{s,p} - cx_{s,p}.$$

Thus, according to the first-order necessary optimality condition, we have

$$x^*_{s,p} = \tfrac{1}{c}[p_s v^{-\alpha_s} - \lambda^{s,p} + cy_{s,p}]^+, \tag{A.9}$$

where $v = \sum_{p\in\pi(s)} x^*_{s,p}$ in turn can be calculated by summing the both sides of (A.9) over $p\in\pi(s)$, i.e.,

$$v = \sum_{p\in\pi(s)} [p_s v^{-\alpha_s} - \lambda^{s,p} + cy_{s,p}]^+. \tag{A.10}$$

The iterative procedure (25) follows from (A.9) and (A.10).

References

[1] Kelly FP, Maulloo A, Tan D. Rate control for communication networks: shadow prices, proportional fairness and stability. Journal of Operations Research Society 1998;49.

[2] Johansson M, Xiao L, Boyd S. Simultaneous routing and resource allocation in CDMA wireless data networks. In: IEEE conference on decision and control, Anchorage, USA, 2003. p. 51–5.

[3] Lee J-W, Chiang M, Calderbank AR. Price-based distributed algorithm for optimal rate-reliability tradeoff in network utility maximization. IEEE Journal on Selected Areas in Commnications 2006;25.

[4] Chiang M. Balancing transport and physical layer in wireless multihop networks: jointly optimal congestion control and power control. IEEE Journal on Selected Areas in Communications 2005;23.
[5] Chiang M. Balancing supply and demand of bandwidth in wireless cellular networks: utility maximization over powers and rates. In: Proceedings of IEEE infocom conference, Phoenix, USA, 2004.
[6] Moberg P. Simultaneous routing and resource allocation in multihop wireless networks using optimization. In: IEEE global telecommunications conference, San Francisco, USA, 2006. p. 1–5.
[7] Akyol U, Andrews M, Gupta P, Hobby J, Saniee I, Stolyar A. Joint scheduling and congestion control in mobile ad-hoc networks. In: Proceedings of IEEE infocom conference, Phoenix, USA, 2008.
[8] Kim G, AR, Negi R. Joint power adaptation, scheduling and routing framework for wireless ad-hoc networks. In: IEEE 6th workshop on signal processing advances in wireless communications, New York, USA, 2005. p. 725–29.
[9] Zhang J, Lee H-N. Jointly optimal rate control and access control in random access wireless networks. Performance Evaluation, under second review.
[10] Lin X, Shroff NB. An optimization based approach for quality of service routing in high-bandwidth networks. In: Proceedings of IEEE infocom conference, Hong Kong, 2003.
[11] Wang MPWH, Low SH. Optimal flow control and routing in multi-path networks. Performance Evaluation 2003;52.
[12] Lin X, Shroff NB. Utility maximization for communication networks with multi-path routing. IEEE Transactions on Automatic Control 2006;51.
[13] Bertsekas DP, Tsitsiklis JN. Parallel and distributed computation: numerical methods. Englewood Cliffs, NJ: Prentice-Hall; 1998.
[14] Goldsmith A. Wireless communications. Cambridge: Cambridge University Press; 2005.
[15] Mo J, Walrand J. Fair end-to-end windows-based congestion control. IEEE/ACM Transactions on Networking 2000;8.
[16] Bertsekas DP. Nonlinear programming. 2nd ed, Athena Scientific; 1999.
[17] Foschini GJ, Miljanic Z. A simple distributed autonomous power control algorithm and its convergence. IEEE Transactions on Vehicular Technology 1993;42.
[18] Baek S-J, de Veciana G. Spatial energy balancing through proactive multipath routing in wireless multihop networks. IEEE/ACM Transactions on Networking 2007;15.
[19] Jain R. The art of computer systems performance analysis. New York: Wiley; 1991.

Jingqiao Zhang received his Ph.D., M.S., and B.S. degrees in Electrical Engineering from Rensselaer Polytechnic Institute in 2008, Tsinghua University in 2003, and Beijing University of Posts and Telecommunications in 2000, respectively. From June 2007 to May 2008, he served as an intern researcher in the Industrial Artificial Intelligence Lab in General Electric Global Re search Center in Niskayuna, NY. His research interests include mathematical programming, evolutionary optimization, machine learning, artificial intelligence, and their application in real-world problems. He has published over 20 journal and conference papers and filed one patent.

Heung-No Lee was born in Choong-Nam and raised in Seoul, South Korea. He received his Ph.D., M.S., and B.S. degrees in electrical engineering from UCLA, in 1999, 1994, and 1993, respectively. From March 1999 to December 2001, he was with the Network Analysis and Systems department in the Information Science Laboratory of HRL Laboratories in Malibu, Cali0-fornia. At HRL, he led a number of research projects as the principal investigator including Traffic modeling for tactical internet (under DARPA ATO ASPEN program), future tactical networking system, capacity analysis for satellite networks using realistic input traffic, and broadband wireless modem. He joined as a Faculty of the Electrical Engineering Department at the University of Pittsburgh, Pennsylvania, in 2002. He is interested in information and signal processing theories for wireless network and bio-medical applications.