Robust Iterative Tree-Pruning Detection and LDPCC Decoding

Heung-No Lee, *Member, IEEE,* and Xinde Hu, *Student Member, IEEE*

***Abstract*—A novel suboptimal low-complexity equalization and turbo-iterative decoding scheme is proposed in this paper. The scheme is developed for multiple transmit- and multiple receive-antenna systems operating over severe *frequency-selective* fading *intersymbol interference* channels. The signal-processing complexity may be of a concern for such systems. The complexity of a full-search equalization grows in a power-law manner $O(M^{N_t L})$, where M denotes for M-ary channel symbols, N_t the number of transmit-antennas, and L the number of delay channel taps. A low-complexity solution can be obtained by pruning an equalizer tree. The two main operations include a *sphere list detection* and a *threshold-based tree-search*. In the operation of extracting extrinsic messages from the pruned tree, a set of explored paths with different survival lengths poses a fairness problem: a longer-lived path naturally builds a larger discrepancy-metric than a shorter lived path does. A novel survival-length *compensation-rule* is devised so that all explored paths with different survival lengths are utilized fairly in generating the output message. Simulation results are obtained for multi-input and multi-output systems equipped with four transmit and four receive antennas. They indicate the performance of the receiver is very robust.**

***Index Terms*—Joint equalization and decoding, low-density parity-check (LDPC) codes, maximum *a posteriori* (MAP), multi-input and multi-output (MIMO) systems, reduced complexity receiver, turbo-iteration.**

I. Introduction

SINCE the landmark work by Telatar [1] and Foschini and Gans [2] on the capacity of multi-input and multi-output (MIMO) fading channels, design of wireless communication systems utilizing multiple antennas at both the transmitter and receiver sides, has become very important. A large body of publications are available today which propose a variety of different system-design ideas in order to attain the promised capacity as closely as possible. According to Zheng and Tse, [3], this additional degree-of-freedom allowed by antenna multiplicity can be utilized in achieving either a higher spectral-efficiency or a higher diversity benefit.

When targeting for spectrum-efficiency improvement, MIMO communication systems are anticipated to bring a ten to twentyfold increase in channel capacities in spectrum-limited joint tactical radio system (JTRS) bands over conventional single-input–single-output techniques [4]. In addition, this improvement is intended for dynamic urban nonline-of-sight multipath channel conditions, where the networked forces are frequently deployed for carrying out tactical missions. Among many design challenges for MIMO systems operating in such environment, the most notable issue relevant to this paper is how to design computationally efficient MIMO signal processing algorithms while maintaining a robust receiver performance. The appropriate size of an antenna-array is considered to be four transmit and four receive antennas for a vehicle mounted system.

The receiver studied in this paper uses a joint turbo-iterative equalization and decoding scheme with a low-density parity-check code (LDPCC) as the coding scheme. The turbo-iterative receiver paired with the powerful outer block code at the transmitter for robust performance has a high potential for achieving the optimal capacity-diversity tradeoff performance. Several turbo-like code-based space–time coded modulation systems have been reported in the literature and their performances have been found to be robust [5]–[8]. However, for higher spectral-efficiency (obtained by increasing the constellation size) and for robust performance over intersymbol interference (ISI) limited channels, a new receiver scheme capable of handling the signal processing complexity is still in great demand.

In [5], a low complexity decoding and equalization scheme based on a novel signal separation and per-antenna turbo-equalization receiver has been proposed. We notice, however, that this per-antenna scheme still requires a large amount of computations. The complexity, measured in terms of a number of states in an ISI trellis as an example, increases exponentially fast as the memory of the channel increases. The number of states in the ISI trellis is $M^{N_t(L-1)}$, where M denotes the M-ary signal symbols. The number of trellis-edges is $M^{N_t L}$. With the per-antenna equalizer, the number of per-antenna trellis states is only M^{L-1}. Nevertheless, the complexity at the signal-separation part remains as high as that of the full-complexity maximum *a posteriori* (MAP) algorithm, which is on the order of $O(M^{N_t(L-1)})$.

In this paper, a low-complexity solution is approached from the standpoint of a tree search. The optimal full-search algorithm, such as the Viterbi algorithm and the Bahl–Cocke–Jelinek–Raviv (BCJR) algorithm, runs on an ISI trellis. In principle, both algorithms utilize the entire duration of a received sequence and generate the final decision variables. The underlying structure of these algorithms ensures that the likelihoods of every possible sequence are utilized for the generation of the decision output, thus allowing for the selection of

Manuscript received April 1, 2004; revised January 20, 2005. This work was supported in part by the Pittsburgh Digital Greenhouse in its fifth round funding.

H.-N. Lee is with the Electrical and Computer Engineering, Department, University of Pittsburgh, Pittsburgh, PA 15261 USA (e-mail: hnlee@ee.pitt.edu).

X. Hu was with the Electrical and Computer Engineering Department, University of Pittsburgh, Pittsburgh, PA 15261 USA. He is now with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15231 USA (e-mail: xih1+@pitt.edu).

Digital Object Identifier 10.1109/JSAC.2005.845419

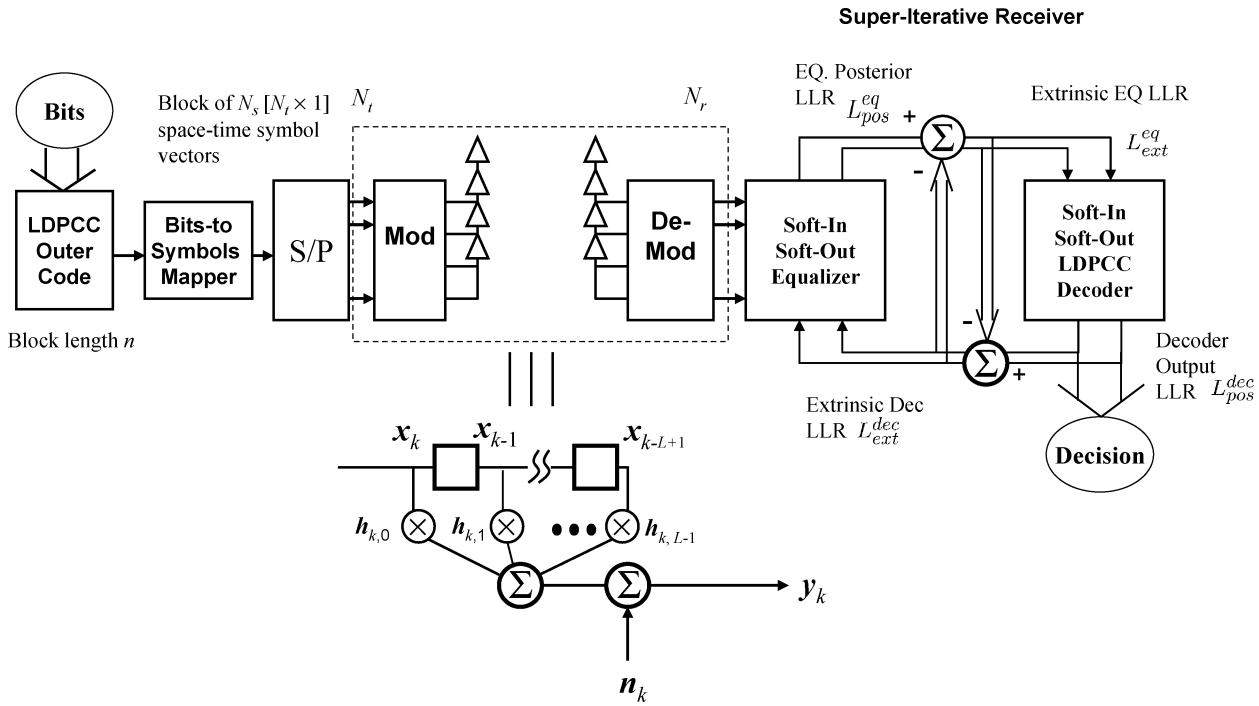

Fig. 1. Proposed transceiver systems over frequency-selective fading channels. Frequency-selective channels are modeled as a finite-impulse response filter with $[N_r \times N_t]$ matrix-valued taps. Input and output symbol vectors have dimensions $[N_t \times 1]$ and $[N_r \times 1]$, respectively. A gray constellation map is used at the BSM. The receiver employs a superiterative receiver scheme in which the soft-in soft-output equalizer and the soft LDPCC decoder exchange the extrinsic information.

the best possible transmitted symbol sequence. The reduction in signal-processing complexity in the proposed approach is to be achieved by maintaining only a small subset of most *posterior-probable* sequences by a tree pruning and is, thus, suboptimal. These are the only sequences used in the generation of the soft-output messages to be exchanged in turbo-iterative fashion to the decoder.

For a high spectral-efficiency system such as the one targeted in the JTRS MIMO system, the use of tree pruning alone would be insufficient to reduce the computational cost. The ISI tree is M^{N_t}-ary. There are M^{N_t} candidates which spawn from a single survivor node. It is certainly not practical to explore all of them, but it is sound to explore the most probable candidates. The proposed tree search algorithm is, therefore, assisted with the employment of a sphere list detection (SLD) algorithm [7], [9] at the stage of node expansion. This strategy seems to make good sense, but so far a design of a receiver based on such a strategy and its performance evaluation have not been explored.

It should be noticed that the proposed tree search algorithm makes forward moves only and is, thus, different from the well-known Fano *sequential search* algorithm [10]. It is rather similar to the T-algorithm [11]. One of its novelties is the introduction of a simple *compensation rule*. It provides a sense of fairness among different survivor paths having different survival lengths in the pruned tree. The longer a path stays alive, the more probable the path is. However, when compared with a shorter lived path, the distance metric of a longer lived path tends to be larger than that of a shorter lived path. This is simply because the longer path has more chances to build up distance. Two options can be considered here. One is to ignore all paths which do not survive to the end. This is undesirable since all information contained in such paths would be lost. The second option is to compensate for the difference in length and incorporate every explored path in generating soft-output messages. Note that the distance metrics of explored paths which are calculated during the path expansion phase can be saved.

The rest of this paper is organized as follows. In Section II, baseband equivalent system descriptions will be given. In Section III, we will discuss the pairwise error probability for MIMO fading ISI channels. The discussion of pair-wise error probability for low-density parity-check (LDPC) code modulated signals will also be given. In Section IV, the overall iterative equalization-decoding system is discussed. In Section V, reduced-complexity equalization using the tree search algorithm and the compensation rule is presented. Sphere list detection is discussed in Section VI. In Section VII, the simulation results for both uncoded and coded transmissions are shown. In Section VIII, a short discussion on system computational complexity is presented. Section IX contains the conclusion.

II. Baseband Equivalent Systems

The complete baseband-equivalent transceiver system is depicted in Fig. 1. We first describe the MIMO fading *intersymbol* interference (ISI) channel model, followed by the transmitter and the receiver operations.

As shown in Fig. 1, we use the tapped delay line (TDL) as an input–output model to describe the MIMO *frequency-selective* channel seen from the modulator (Mod) to the demodulator (De-mod). Without loss of generality, the symbol-spaced sampling at the demodulator output is assumed and, thus, the taps are symbol spaced. The TDL channel model has L channel taps and each channel tap is a $[N_r \times N_t]$ matrix, where N_t is the

number of transmitter antennas and N_r the number of receiver antennas. The channel is also modeled as a time-varying TDL filter. Thus, $\boldsymbol{h}_{k,l}$ is an $[N_r \times N_t]$ matrix denoting the lth tap, $l = 0, 1, \ldots, L-1$, at the kth symbol epoch $k = 1, \ldots, N_s$ in the matrix-valued TDL channel-model.

The bits-to-symbol mapper (BSM) at the transmitter, shown in Fig. 1, represents a symbol modulator which takes a string of coded bits and maps it into a set of N_t channel symbols. The N_t channel symbols can be stored in the $[N_t \times 1]$ symbol vector $\boldsymbol{x}_k$. The element of each $[N_t \times 1]$ vector $\boldsymbol{x}_k$ is taken from the base digital constellation $\mathcal{S}$ such as binary phase-shift keying (BPSK) ($M = 2$), 4-PSK ($M = 4$), and M-ary quadrature amplitude modulation (QAM). Thus, the BSM maps a string of total $N_t \log_2(M)$ coded bits into a vector-symbol $\boldsymbol{x}_k \in \mathcal{S}^{N_t}$ at each time epoch k. Each of the $N_t \log_2(M)$ coded bits at the kth epoch is denoted as $c_{k,p,q}$, for $p = 1, 2, \ldots, N_t$; $q = 1, 2, \ldots, \log_2(M)$. The channel is used N_s times for a block transmission. Thus, we note that there are a total of $N_t N_s \log_2(M)$ coded bits per block transmission. We assume in this paper the length n of the block code is $n = N_t N_s \log_2(M)$.

The N_r receive symbols at the demodulator output can be grouped into the $[N_r \times 1]$ receive symbol vector $\boldsymbol{r}_k$. Thus, the MIMO relationship can be written as

$$\begin{aligned} \boldsymbol{r}_k &= \sum_{l=0}^{L-1} \boldsymbol{h}_{k,l} \boldsymbol{x}_{k-l} + \mathbf{n}_k \\ &= \boldsymbol{y}_k + \mathbf{n}_k \end{aligned} \tag{1}$$

where we have defined the following variables:

$$\boldsymbol{r}_k := \begin{pmatrix} r_k^{(1)} \\ \vdots \\ r_k^{(N_r)} \end{pmatrix}, \quad \boldsymbol{x}_k := \begin{pmatrix} x_k^{(1)} \\ \vdots \\ x_k^{(N_t)} \end{pmatrix}, \quad \mathbf{n}_k := \begin{pmatrix} n_k^{(1)} \\ \vdots \\ n_k^{(N_r)} \end{pmatrix},$$
$$\boldsymbol{h}_{k,l} := \begin{pmatrix} h_{k,l}^{(1,1)} & \cdots & h_{k,l}^{(1,N_t)} \\ \vdots & \ddots & \vdots \\ h_{k,l}^{(N_r,1)} & \cdots & h_{k,l}^{(N_r,N_t)} \end{pmatrix}. \tag{2}$$

In the second line in (1), $\boldsymbol{y}_k$ is defined as the *clean* channel output. In (2), $x_k^{(j)}$, $j = 1, 2, \ldots, N_t$, denotes a channel symbol at the jth transmitter antenna. Thus, we have $x_k^{(j)} \in \mathcal{S}$ and $\boldsymbol{x}_k \in \mathcal{S}^{N_t}$. The noise $n_k^{(i)}$, $i = 1, 2, \ldots, N_r$, is a circularly symmetric complex Gaussian (CSCG) with zero mean and variance N_0, where N_0 is the one-sided power spectral density of the additive white Gaussian noise. In addition, it is independent and identically distributed for each i and k. Each $h_{k,l}^{(i,j)}$ denotes the CSCG fading channel tap with zero mean and a certain variance which can be set according to a multipath power-delay profile. We use the *uniform* multipath power-delay profile, thus, the average power of a single tap $h_{k,l}^{(i,j)}$ is set to $1/L$ for all i, j and l. The channel $h_{k,l}^{(i,j)}$ is independent and identically distributed for each combination of transmitter and receiver pair i and j. Moreover, on the same antenna, different delay taps are also independently distributed from each other. Each fading tap is time-varying whose rate of change can be set according to the maximum Doppler fading rate f_{dm}.

For simulation, we need to generate as many as $N_r N_t L$ number of independently fading taps. Virtually, any number of independent time-varying channel tap processes can be produced by a filtered Gaussian process method (FGPM). In FGPM, a number of independently generated complex-valued Gaussian random processes are filtered with a low-pass filter. We take the approach of frequency-domain filtering using the fast Fourier transform (FFT). The deterministic shape of the low-pass filter can be set according to a desired Doppler power spectrum on which the maximum Doppler frequency can be defined (we use the Jakes outdoor Doppler spectrum [12] throughout the paper). The bandwidth of the low-pass filter determines the maximum Doppler shift. Finally, the time-varying channel tap processes can be obtained by taking the inverse FFT of the filtered Gaussian random sample paths. For further information, interested readers may find it useful to refer to [13, Ch. 5, p. 222]. Throughout the paper, we assume the channel estimation is perfectly done at the receiver.

We assume the use of the outer LDPC code which drives the BSM shown in Fig. 1. As mentioned earlier, the block length of the code is chosen as $n = N_t N_s \log_2(M)$. Then, by collecting N_s receive-symbol vectors into an overall receive-vector $\boldsymbol{r}$ of length $N_s N_r$, the input–output relationship can be written as (3)

$$\boldsymbol{r} := \begin{pmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \\ \vdots \\ \boldsymbol{r}_{N_s} \end{pmatrix} = \underbrace{\begin{pmatrix} \boldsymbol{h}_{1,0} & 0 & 0 & 0 & \cdots \\ \boldsymbol{h}_{2,1} & \boldsymbol{h}_{2,0} & 0 & 0 & \cdots \\ 0 & 0 & \ddots & \ddots & \vdots \\ \boldsymbol{h}_{L,L-1} & \cdots & \boldsymbol{h}_{L,0} & 0 & \cdots \\ 0 & 0 & \ddots & \ddots & \vdots \\ 0 & \cdots & \boldsymbol{h}_{N_s,L-1} & \cdots & \boldsymbol{h}_{N_s,0} \end{pmatrix}}_{:=\boldsymbol{h}} \times \underbrace{\begin{pmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_{N_s} \end{pmatrix}}_{:=\boldsymbol{x}} + \underbrace{\begin{pmatrix} \boldsymbol{n}_1 \\ \vdots \\ \boldsymbol{n}_{N_s} \end{pmatrix}}_{:=\boldsymbol{n}} \tag{3}$$

or as

$$\boldsymbol{r} = \boldsymbol{h}\boldsymbol{x} + \mathbf{n} \tag{4}$$

where $\boldsymbol{h}$ denotes the overall channel matrix of size $[N_s N_r \times N_s N_t]$.

We note that this coded modulation and decoding system would achieve a maximum throughput of $R_t N_t \log_2(M)$ bits per channel use where R_t is the rate of the outer code.

III. Pairwise Error Probability

We derive the *pairwise error* probability averaged over the frequency-selective independent fading channels. The primary purpose of the derivation is to evaluate the order of diversity achievable by the LDPC coded modulation system over the fading channels. Due to the space–time transmission of the coded bits as well as the frequency-selective channels, the system is capable of exploiting signal-diversities available in all space-, time-, and frequency-domains. This pairwise error

result can be used in the calculation of maximum likelihood union upper bounds [14].

The probability of transmitting a word $\boldsymbol{x}$ and deciding in favor of another word $\tilde{\boldsymbol{x}}$ can be approximated by the Chernoff upper bound

$$P(\boldsymbol{x} \to \tilde{\boldsymbol{x}}|\boldsymbol{h}) \leq \exp\left(-\frac{d^2(\boldsymbol{x},\tilde{\boldsymbol{x}})E_s}{4N_0}\right) \tag{5}$$

where $d(\boldsymbol{x},\tilde{\boldsymbol{x}})$ is the overall Euclidean distance between $\boldsymbol{y} = \boldsymbol{h}\boldsymbol{x}$ and $\tilde{\boldsymbol{y}} = \boldsymbol{h}\,\tilde{\boldsymbol{x}}$. The distance can be rewritten as

$$d^2(\boldsymbol{x},\tilde{\boldsymbol{x}}) = \sum_{k=1}^{N_s}\sum_{i=1}^{N_r}\sum_{l=0}^{L'(k)-1}\left|\boldsymbol{h}_{k,l}^{(i,:)}\Delta\boldsymbol{x}_k\right|^2 \tag{6}$$

$$= \sum_{k=1}^{N_s}\sum_{i=1}^{N_r}\sum_{l=0}^{L'(k)-1}\boldsymbol{h}_{k,l}^{(i,:)}\Delta\boldsymbol{x}_k\Delta\boldsymbol{x}_k\boldsymbol{h}_{k,l}^{(i,:)^*} \tag{7}$$

where the superscript $(\cdot)^*$ denotes the Hermitian (conjugate-transpose) of a matrix (or a vector) and $\boldsymbol{h}_{k,l}^{(i,:)}$ denotes the ith row—thus, a dimension of $[1 \times N_t]$—of the kth time epoch and the lth channel tap matrix. We use the $[N_t \times 1]$ error-vector $\Delta\boldsymbol{x}_k := \boldsymbol{x}_k - \tilde{\boldsymbol{x}}_k$.

As k approaches toward N_s, we expect to see an "edge" effect: the number of nonzero column entries in $\boldsymbol{h}$ reduces one-by-one to 1 from L. This happens only near the end of the block. We can make the representation to be as precise as we want by defining and using a new variable, say $L'(k) = L$ for $k = 1,2,\ldots,N_s - L + 1$ and $L'(N_s - L + j) = L - j$ for $j = 1,\ldots,L-1$. However, this edge effect is negligible at the perspective of the transmission of a whole block of length $N_s \gg L$. Thus, we will ignore this edge-effect in the sequel and assume $L(k) = L$ throughout all k.

We note that the $[N_t \times N_t]$ matrix $\Delta\boldsymbol{x}_k\Delta\boldsymbol{x}_k^* =: \boldsymbol{P}_k$ is Hermitian symmetric and, thus, can be diagonalized with nonnegative eigenvalues and unitary eigenvectors. That is, we can find a unitary matrix $\boldsymbol{V}_k$ and a real diagonal matrix $\boldsymbol{Q}_k$ such that $\boldsymbol{P}_k = \boldsymbol{V}_k\boldsymbol{Q}_k\boldsymbol{V}_k$. It should be noted that this diagonalization can be done for each $k = 1,\ldots,N_s$.

Proceeding further with the distance (7), we have

$$d^2(\boldsymbol{x},\tilde{\boldsymbol{x}}) = \sum_{k=1}^{N_s}\sum_{i=1}^{N_r}\sum_{l=0}^{L-1}\boldsymbol{h}_{k,l}^{(i,:)}\boldsymbol{P}_k\boldsymbol{h}_{k,l}^{(i,:)^*} \tag{8}$$

$$= \sum_{k=1}^{N_s}\sum_{i=1}^{N_r}\sum_{l=0}^{L-1}\underbrace{\boldsymbol{h}_{k,l}^{(i,:)}\boldsymbol{V}_k}_{=:\boldsymbol{g}_{k,l}^{(i,:)}}\boldsymbol{Q}_k\boldsymbol{V}_k^*\boldsymbol{h}_{k,l}^{(i,:)^*} \tag{9}$$

$$= \sum_{k=1}^{N_s}\sum_{i=1}^{N_r}\sum_{l=0}^{L-1}\boldsymbol{g}_{k,l}^{(i,:)}\boldsymbol{Q}_k\boldsymbol{g}_{k,l}^{(i,:)^*}. \tag{10}$$

For the purpose of continuing discussion, it is beneficial to mention some properties of $\boldsymbol{g}_{k,l}^{(i,:)}$, $\boldsymbol{Q}_k$ and $\boldsymbol{V}_k$.

- $\boldsymbol{V}_k\boldsymbol{V}_k^* = \boldsymbol{V}_k^*\boldsymbol{V}_k = \boldsymbol{I}_{N_t\times N_t}$, where $\boldsymbol{I}_{N_t\times N_t}$ is the $[N_t \times N_t]$ identity matrix. The columns of $\boldsymbol{V}_k$ form a *complete* basis of the N_t dimensional space.
- The elements of the diagonal matrix $\boldsymbol{Q}_k$ are the eigenvalues of $\boldsymbol{P}_k$, which are nonnegative reals. Since the rank of $\boldsymbol{P}_k$ is either one or zero, all but a single eigenvalue—say the first one—are zeros. Depending on the two vectors $\boldsymbol{x}_k$ and $\tilde{\boldsymbol{x}}_k$, the value of the first eigenvalue is either zero or $||\Delta\boldsymbol{x}_k||^2$. Denoting this first eigenvalue as d_k, we note that

$$d_k = \begin{cases} ||\Delta\boldsymbol{x}_k||^2, & \text{when } \boldsymbol{x}_k \neq \tilde{\boldsymbol{x}}_k \\ 0, & \text{o.w.} \end{cases} \tag{11}$$

- The matrix $\boldsymbol{P}_k$ and its eigenvalue d_k are different with respect to different k.
- Note that the statistical property of the $[N_t \times 1]$ vector $\boldsymbol{g}_{k,l}$ remains the same as that of $\boldsymbol{h}_{k,l}^{(i,:)}$ since $\boldsymbol{V}_k$ is a unitary transformation.
- Now, note that the term, $\boldsymbol{g}_{k,l}^{(i,:)}\boldsymbol{Q}_k\boldsymbol{g}_{k,l}^{(i,:)^*}$, is a real-valued random variable $d_k|\bar{g}_{k,l}^{(i,1)}|^2$. Each $|\bar{g}_{k,l}^{(i,1)}|^2$ is the χ^2-random variable with two degrees of freedom with mean $1/L$.

Finally, the distance square can be written as

$$d^2(\boldsymbol{x},\tilde{\boldsymbol{x}}) = \sum_{k=1}^{N_s}\sum_{i=1}^{N_r}\sum_{j=0}^{L-1}d_k\left|g_{k,l}^{(i,1)}\right|^2. \tag{12}$$

Thus, substituting (12) into (5), we obtain

$$P(\boldsymbol{x} \to \tilde{\boldsymbol{x}}|\boldsymbol{h}) \leq \prod_{k=1}^{N_s}\exp\left(-d_k\frac{E_s}{4N_0}\sum_{i=1}^{N_r}\sum_{l=0}^{L-1}\left|g_{k,l}^{(i,1)}\right|^2\right) \tag{13}$$

assuming the channel taps $\boldsymbol{h}_{k,l}$ are independent over time k.

For the MIMO ISI fading channel, (12) is a general result which can be used to approximate the error probability for any given channel matrix $\boldsymbol{h}$. It should be noticed that (13) also incorporates the delay-diversity taps so that frequency-selective diversity can be analyzed as well.

Now, with a further assumption of independence with respect to each receiver antenna and each channel-delay tap, we note that all $h_{k,l}^{(i,1)}$ and, thus, all $g_{k,l}^{i,1}$ for each and every combination of indices i,l, are independent and identically distributed. Every element in the row vector $\boldsymbol{g}_{k,l}^{(i,:)}$, is a mutually independent and complex Gaussian with variance $1/L$ and zero mean. Under this set of conditions, the pairwise error ensemble averaged over the channel can be written as

$$P_e(\Delta\boldsymbol{x}) := P(\boldsymbol{x} \to \tilde{\boldsymbol{x}}) \leq \prod_{k=1}^{N_s}\prod_{i=1}^{N_r}\left[\prod_{l=0}^{L-1}\frac{1}{1+d_k\frac{E_s}{4N_0}}\right] \tag{14}$$

$$= \prod_{k=1}^{N_s}\left[\frac{1}{1+d_k\frac{E_s}{4N_0}}\right]^{N_rL}. \tag{15}$$

Comparing (15) with the pairwise error probability for MIMO *flat* fading channels from [15], we notice that the diversity order is improved by a factor of L. This is expected since the channel has L independently fading delay taps. This is in addition to the space diversity N_r from the receiver antennas. A similar result can be obtained also for the slow fading channel which is beyond the scope of this paper and will not be discussed.

Let w_H denote the cardinality of the set of the first eigenvalues d_k, such that

$$\mathcal{D} := \{d_k : d_k \neq 0, k = 1,2,\ldots,N_s\}. \tag{16}$$

Then, we note that the diversity order $w_H N_r L$ can be achieved with a careful design of the outer code. For $(E_s/4N_0) \gg 1$, (15) becomes

$$P_e(\Delta \boldsymbol{x}) \leq \prod_{k \in \mathcal{D}} \left[d_k \frac{E_s}{4N_0} \right]^{-N_r L} \tag{17}$$

$$= \left(\prod_{k \in \mathcal{D}} d_k^{-N_r L} \right) \left[\frac{E_s}{4N_0} \right]^{-N_r L w_H}. \tag{18}$$

This leads to the *Distance* and *Product* criteria in [15] for rapidly fading channels. In our case, the space–time coding block is a simple modulation block (the BSM shown in Fig. 1) which maps the binary coded-sequence into a symbol sequence. The binary coded-sequence is the codewords of the outer LDPCC.

We use Gallager's codes [16], [17] which is a linear block code. An ensemble of a (n, w_c, w_r) Gallager code can be fully defined by its parity-check matrix $\boldsymbol{F}$ in which there are n columns. w_c denotes the Hamming weight of each column and w_r that of each row. The code-rate R_t is $R_t \approx 1 - w_c/w_r$. A particular parity-check matrix is obtained randomly under a given weight constraint. Similar to the turbo decoder, the decoder of the LDPCC uses iterations to exchange soft log-likelihood-ratio (LLR) messages [16].

As can be hinted from (18), the minimum Hamming weight $D_{\min}$ of this linear code plays a key role in determining the error performance of the MIMO system. For Gallager codes, the minimum distance $D_{\min}$ is proportional to the code length $D_{\min} = \sigma n$ [16], [17], where σ is a parameter set by a given column weight w_c and a row weight w_r. In particular, from [16, Th. 2.4] (or [17, Fig. 3]) we note that for an (n, w_c, w_r) ensemble, the probability of selecting a code whose minimum distance is smaller than $n\sigma$ would be very small. In fact, it is close to $(w_r - 1)/(2n^{w_c - 2})$. For example, $\sigma = 0.023$ for the $(n, w_c = 3, w_r = 6)$ ensemble. Thus, $D_{\min}$ for $n = 1024$ would be around 24; and that of a (4096, 3, 6) code is 94. The probability of random selection of a code from the ensemble with $D_{\min}$ smaller than $n\sigma$ is, thus, $5/(2 \times 1024) = 0.0024$ for $n = 1024$ and $5/(2 \times 4096) = 6 \times 10^{-4}$, respectively.

Assuming $D_{\min} = n\sigma$ and binary channel symbols $(M = 2)$, we note that, w_H in (18) ranges from $\lfloor (D_{\min}/N_t) \rfloor$ to $D_{\min}$. The former is the case when all differences between a pair of codewords happens to occur consecutively. In other words, the cardinality of the set $\{k : \boldsymbol{x}_k \neq \tilde{\boldsymbol{x}}_k, k = 1, \ldots, N_s\}$ is $\lfloor (D_{\min}/N_t) \rfloor$. The latter refers to the case in which there is no consecutive difference: Each and every $D_{\min}$ coded-bit difference occurs sporadically across the entire block and, thus, there is no single event with N_t consecutive coded-bit differences. Therefore, any nonzero d_k in (15) is due to a single coded-bit difference. From the discussion so far, we note that the diversity order in the range of the minimum $N_r L \lfloor (D_{\min}/N_t) \rfloor$ to the maximum $N_r L D_{\min}$ is anticipated for independently fading channels.

IV. Superiterative Equalization and Decoding System

In the receiver part shown in Fig. 1, the demodulated sequence of $[N_r \times 1]$ receive-signal vectors $\boldsymbol{r}_k$, $k = 1, \ldots, N_s$, is feed-forwarded to the soft-in soft-out (SISO) sum-product equalizer block. The equalizer generates soft-output messages on the coded bits. The decoder takes the soft messages and runs its own iterative message passing operation on the bipartite code graph. After a certain number of iterations, the decoder generates a block of posterior *log-likelihood ratios* (LLRs) on the coded bits and feeds them back to the equalizer. The equalizer takes the extrinsic part and runs the equalization operation again. In this paper, we will refer to this iteration between the decoder and the equalizer as the "superiteration."

In a classical context, an equalizer is employed to mitigate the intersymbol interference (ISI) which may be present in frequency-selective channels. The outputs of the proposed equalizer, however, are the LLRs on the coded bits. Thus, one may ask a question: what are the entities that are equalized? In fact, there is nothing that is equalized. We will, however, continue to use the term "equalizer" in this paper for the lack of a better term.

The equalizer in fact takes the role of a SISO de-mapper. At the transmitter, the BSM is used to map a string of coded bits to a channel symbol vector. As mentioned in Section II, each element of the vector $\boldsymbol{x}$ is taken from the base digital-constellation $\mathcal{S}$ such as BPSK, 4-PSK, or M-ary QAM. Thus, the BSM maps a string of $N_t \log_2(M)$ coded bits into a single-channel symbol vector $\boldsymbol{x}_k \in \mathcal{S}^{N_t}$. Meanwhile, the equalizer takes a received-signal vector $\boldsymbol{r}_k$ and generates soft-output messages on the $N_t \log_2(M)$ coded bits per each $k = 1, 2, \ldots, N_s$. Therefore, the equalizer is closer in its role as a soft-out de-mapper which generates the soft messages on the coded bits by processing the received-signal sequence. When operating in the superiteration mode, the equalizer also make use of the soft-input messages (LLRs) generated from the decoder.

The process of superiteration can be elaborated now. As shown in Fig. 1, the sequence of posterior LLRs is denoted by $L^{\text{eq}}_{\text{pos}} := (L^{\text{eq}}_{\text{pos},1} L^{\text{eq}}_{\text{pos},2} \ldots L^{\text{eq}}_{\text{pos},n})$. The extrinsic LLRs, denoted in a similar manner as $L^{\text{eq}}_{\text{ext}}$, can be obtained by removing the LLR's $L^{\text{dec}}_{\text{ext}}$ from $L^{\text{eq}}_{\text{pos}}$. That is

$$L^{\text{eq}}_{\text{ext},j} = L^{\text{eq}}_{\text{pos},j} - L^{\text{dec}}_{\text{ext},j} \tag{19}$$

for $j = 1, 2, \ldots, n$. All LLRs discussed in this paper are on the coded bits and, thus, the length of the LLR sequences is fixed to be n (which is the length of the LDPCC). As mentioned, we assume $n = N_t \times N_s \times \log_2(M)$ throughout this paper without loss of generality. The LDPCC decoder takes the sequence $L^{\text{eq}}_{\text{ext}}$ as its input and generates the posterior LLR's $L^{\text{dec}}_{\text{pos}}$. The extrinsic part is forwarded to the equalizer, which is obtained by

$$L^{\text{dec}}_{\text{ext},j} = L^{\text{dec}}_{\text{pos},j} - L^{\text{eq}}_{\text{ext},j} \tag{20}$$

for $j = 1, 2, \ldots, n$. The equalizer utilizes the extrinsic LLR's $L^{\text{dec}}_{\text{ext},k}$ as prior LLRs, and updates its output, say $L^{\text{eq}}_{\text{pos}}$. This completes a single cycle of superiteration. In the beginning of the cycle, the sequence $L^{\text{dec}}_{\text{ext}}$ from the decoder is not available and, thus, set to zero. In the next section, the generation of the posterior equalizer output LLR's $L^{\text{eq}}_{\text{pos}}$ will be discussed. We use the standard message passing algorithm to generate the decoder output posteriors $L^{\text{dec}}_{\text{pos}}$ [17]–[19].

After a certain number of superiterations, hard decisions are made at the output of the decoder. A detailed design of a LDPC encoder and an iterative decoder can be found in [16] and [20].

V. Reduced Complexity Equalizer

The SISO equalizer is depicted in Fig. 1. As mentioned in Section IV, the equalizer plays a role as a de-mapper which takes the receive signal from the demodulator and generates the posterior LLRs on the coded bits of length n ($n = N_t N_s \log_2(M)$). The sum-product algorithm is used at the equalizer to generate the posterior LLRs. For optimal processing, this sum-product algorithm should be applied to the full list of all possible candidates of space–time symbol sequences. For a low complexity solution, it can be applied to a smaller subset of most probable candidates. This section describes how to obtain this smaller set and how to apply the sum-product algorithm to the set.

A Q-ary tree, $Q := M^{N_t}$, can be used as a model to track all possible space–time symbol sequences $\boldsymbol{x}$ of length N_s. We first describe the application of the sum-product algorithm on the full tree for ease of explanation and then the sum-product algorithm applied on a pruned tree.

A. Sum-Product Algorithm and the Distance Measure

Using the input–output relationship given in (4), the posterior probability can be written as

$$\begin{aligned}\Pr(\boldsymbol{x}|\boldsymbol{r}) &= P(\boldsymbol{x},\boldsymbol{r})P(\boldsymbol{r}) \\ &\propto P(\boldsymbol{r}|\boldsymbol{x})\Pr(\boldsymbol{x}) \\ &\propto \prod_{k=1}^{N_s} P(\boldsymbol{r}_k|\boldsymbol{x}_k,\cdots,\boldsymbol{x}_{k-L+1})Pr(\boldsymbol{x}_k) \end{aligned} \tag{21}$$

where the definition of conditional probability, the Bayes' theorem, and the assumption of white additive Gaussian noise are used, respectively, in each line.

We note that the likelihood for each time epoch k in (21) can be obtained by

$$P(\boldsymbol{r}_k|\boldsymbol{x}_k,\cdots,\boldsymbol{x}_{k-L+1}) \propto \underbrace{\exp\left(-\frac{1}{N_o}||\boldsymbol{r}_k - \boldsymbol{y}_k||^2\right)}_{=:p(\boldsymbol{y}_k)} \tag{22}$$

where a likelihood metric $p(\boldsymbol{y}_k)$ is defined for the quantity on the left-hand side. In addition, the prior probabilities $Pr(\boldsymbol{x}_k)$ can also be obtained. Recalling our BSM scheme shown in Fig. 1, we note that there are $N_t \log_2(M)$ number of coded bits associated with each $\boldsymbol{x}_k$. Further recalling our notation that the coded bits at the kth string are denoted by $c_{k,p,q}$ for $p = 1,\ldots,N_t$; $q = 1,\ldots,\log_2(M)$. Thus, the prior $Pr(\boldsymbol{x}_k)$ in (21) can be obtained by

$$Pr(\boldsymbol{x}_k) = \prod_{p=1}^{N_t}\prod_{q=1}^{\log_2(M)} p(c_{k,p,q}). \tag{23}$$

Recalling that the BSM block maps the string of $N_t \log_2(M)$ coded bits to a space-symbol vector $\boldsymbol{x}_k$ at each symbol-epoch k, we obtain the prior probabilities on the coded bits from the extrinsic message $L^{\text{dec}}_{\text{ext},k}$ from the decoder such that

$$\begin{aligned} p(c_{k,p,q}=1) &= \frac{\exp\left(L^{\text{dec}}_{\text{ext},k}\right)}{1+\exp\left(L^{\text{dec}}_{\text{ext},k}\right)} \\ p(c_{k,p,q}=0) &= \frac{1}{1+\exp\left(L^{\text{dec}}_{\text{ext},k}\right)} \end{aligned} \tag{24}$$

for $k = 1,\ldots,N_s$, $p = 1,\ldots,N_t$, and $q = 1,\ldots,\log_2(M)$.

The posterior LLR for each coded bit, $c_{k,p,q}$ for $k = 1,\ldots,N_s$; $p = 1,\ldots,N_t$; $q = 1,\ldots,\log_2(M)$, can be obtained as

$$\begin{aligned} L^{\text{eq}}_{\text{pos}}(c_{k,p,q}) &= \log\frac{\Pr(c_{k,p,q}=1|\boldsymbol{r})}{\Pr(c_{k,p,q}=0|\boldsymbol{r})} \\ &= \log\left(\frac{\sum_{\boldsymbol{x}:c_{k,p,q}=1} P(\boldsymbol{r}|\boldsymbol{x})\Pr(\boldsymbol{x})}{\sum_{\boldsymbol{x}:c_{k,p,q}=0} P(\boldsymbol{r}|\boldsymbol{x})\Pr(\boldsymbol{x})}\right). \end{aligned} \tag{25}$$

Note in (24) and (25) that there are a total of n LLRs in $L^{\text{eq}}_{\text{pos}}$, i.e., $L^{\text{eq}}_{\text{pos}}(c_{k,p,q})$ for $k = 1,\ldots,N_s$; $p = 1,\ldots,N_t$; and $q = 1,\ldots,\log_2(M)$. Defining an auxiliary index j, i.e., $j := p+(q-1)\log_2(M) + (k-1)N_t\log_2(M)$ and, thus, $j = 1\ldots,n$, we note the LLRs given in (19) can be fully defined, i.e., $L^{\text{eq}}_{\text{pos},j} \equiv L^{\text{eq}}_{\text{pos}}(c_{k,p,q})$.

For the purpose of the tree-pruning operation, we found it useful to define the log version of the product algorithm. By taking the log on both sides of (21), we have

$$\begin{aligned} \alpha(\boldsymbol{x},\boldsymbol{r}) &:= \log\left(\Pr\{\boldsymbol{x}|\boldsymbol{r}\}\right) \\ &\propto \sum_{j=1}^{N_s}\left\{\log\left(p(\boldsymbol{y}_j)\right) + \log\left(\Pr(\boldsymbol{x}_j)\right)\right\}. \end{aligned} \tag{26}$$

$$\begin{aligned} &= -\frac{1}{N_o}\sum_{j=1}^{N_s}\underbrace{\left\{||\boldsymbol{r}_j - \boldsymbol{y}_j||^2 - N_0\log\left(\Pr(\boldsymbol{x}_j)\right)\right\}}_{=:d(\boldsymbol{r}_j,\boldsymbol{y}_j)} \\ &= -\frac{1}{N_o}\sum_{j=1}^{N_s} d(\boldsymbol{r}_j,\boldsymbol{y}_j). \end{aligned} \tag{27}$$

For convenience, let us define α_k^m as the cumulative measure of a particular path, say the mth and at a certain tree-depth k. As shown in Fig. 2, the survived paths are counted from the top of the tree and indexed as $m = 0, 1, 2, \ldots$ Then, we have

$$\alpha_k^m := \sum_{j=1}^{k}\underbrace{d\left(\boldsymbol{r}_j,\boldsymbol{y}_j(m)\right)}_{=:\gamma_j^m} = \sum_{j=1}^{k}\gamma_j^m \tag{28}$$

where γ_j^m denotes the jth branch metric of the mth path. It can be imagined that the tree will explode quickly. Therefore, the idea of tree-pruning is used.

B. Pruning by Threshold Test and Applying the Sum-Product on Survived Paths

A pruned tree is used in the sum-product algorithm to generate the posteriors, as given in (25). Depicted in Fig. 2, the binary tree ($Q = 2$) is used for the purpose of illustration. Associated with each path is a particular transmitted sequence of symbol-vectors $\boldsymbol{x}_k$, or more directly the clean channel output $\boldsymbol{y}_k$. Upon reception of the receive-signal vectors $\boldsymbol{r}_k$, the distance-metric (28) on each path can be computed.

We propose the use of a threshold-based tree pruning rule. While expanding the tree in the forward direction, some low-probability paths are not selected for inclusion into a list of survivor paths which are further explored. From the $(k-1)$st epoch to the kth epoch, for example, a single survivor path is expanded into $Q = M^{N_t}$ candidates. Suppose the number of survivors at $k-1$ is $N_{sv}(k-1)$, thus, the total number of candidates at k may reach up to $M^{N_t N_{sv}(k-1)}$. We may use the SLD method

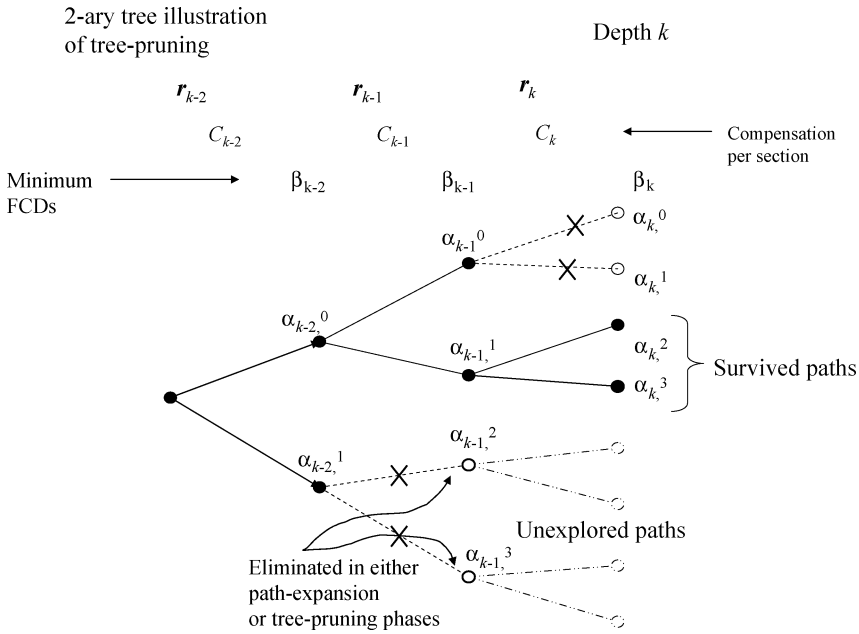

Fig. 2. Illustration of tree-pruning with $(Q = 2)$ binary tree. For MIMO settings, the tree is $Q = M^{N_t}$-ary.

at the expansion phase to further reduce the complexity. This is discussed in detail in the next section. For each expanded candidate path, the cumulative metric is updated. There are a total of $N_{sv}(k)$ cumulative metrics at the kth epoch. Thus, there is a set of $N_{sv}(k)$ number of metric elements, i.e., $\{\alpha_k^m\}$, for $m = 0, 1, 2, \ldots, N_{sv}(k)-1$. The threshold-based rejection rule can then be applied and is described as follows.

1) Find the path whose metric is the smallest and set its metric value as β_k.
2) Prune all paths whose accumulated metric α_k^m is larger than $\beta_k + T$, where T is a threshold value to be optimized. (The optimal threshold value can be determined off-line through simulations and before being used in practice.)

Now, suppose there is a set of sequences which have been explored. For an optimal detection performance at a given complexity (e.g., a fixed threshold value T), we desire to make use of all the explored paths and their associated metrics in the sum-product algorithm. Then, the posterior LLR in (25) can be obtained as

$$L_{\text{pos}}^{\text{eq}}(c_{k,p,q}) = \log\left(\frac{\sum_{\boldsymbol{x}'\in\mathcal{W}^+} P(\boldsymbol{r}|\boldsymbol{x}')P(\boldsymbol{x}')}{\sum_{\boldsymbol{x}'\in\mathcal{W}^-} P(\boldsymbol{r}|\boldsymbol{x}')P(\boldsymbol{x}')}\right) = \log\left(\frac{\sum_{\boldsymbol{x}'\in\mathcal{W}^+} \exp\left(\alpha^m(\boldsymbol{x}', \boldsymbol{r})\right)}{\sum_{\boldsymbol{x}'\in\mathcal{W}^-} \exp\left(\alpha^m(\boldsymbol{x}', \boldsymbol{r})\right)}\right) \quad (29)$$

where we define two mutually exclusive sets of explored paths among the survivors whose length is at least k. $\mathcal{W}^+$ denotes the set of explored paths with $c_{k,p,q} = 1$, and $\mathcal{W}^-$ denotes the set of explored paths with $c_{k,p,q} = 0$.

While using all of the explored paths is desirable, a problem arises because of difference in path length—some paths are pruned earlier than others. Thus, one must decide how to fairly use these shorter paths in the sum-product algorithm for calculation of the posteriors (29). Such short-lived paths tend to have smaller distance metrics than longer lived paths do. One naive solution is to truncate all the sequences used in the summation phase at the same length. For example, one may choose to utilize only the handful of paths which survive to the last epoch. This option has the obvious disadvantage of not making use of the potentially important information available in some longer lived survivors. A more accurate and elaborate solution is to devise and utilize a rule which compensates for the difference in path lengths.

C. Compensation Part

As an integral part of the proposed transceiver system, a turbo-iterative detection between the equalizer and the decoder is desired. The quality of soft-output messages generated by the equalizer in the sum-product algorithm (29) would improve as the number of paths is increased. While making forward movement, the equalizer generates a list of explored paths along with their associated metrics which are used in making the tree-pruning decision. The metric of the explored paths can be saved for a later use in the sum-product algorithm generating the posterior LLRs.

However, utilizing every explored path with different survivor lengths in the sum-product algorithm poses a fairness problem. As mentioned, a shorter lived path has a smaller metric than a longer lived path does. A solution suggested in this paper is to *penalize* each path pruned early in the process for its missing tree sections. This can be accomplished by having a certain amount of distance metric added to the cumulative-metric of the path to *compensate* for the number of tree sections for which the distance metric is not accumulated. In this paper, we propose a simple *compensation rule* in which the compensation for the kth section C_k is calculated during the forward movement and added to the cumulative metric of the path per each missing branch in the sum-product algorithm. If a path is pruned at $k = N_p < N_s$, for example, a cumulative compensation $S_{N_p} = \sum_{k=N_p}^{N_s-1} C_k$ is added to those paths which have been eliminated at $k = N_p$.

The following two principles of compensation can be stated for a selection of C_k.

- *Fairness*: C_k must be large enough so that the compensated path would be forced to drop out if it were put into competition in the next time section.
- *Tightness*:) C_k should be appropriately small so that the pruned paths contribute in the sum-product algorithm as equally as any surviving path does.

Now, a compensation rule can be devised. The result is stated in the following lemma and theorem.

Lemma 1: Let β_{k+1} and β_k be the minimum forward-cumulative distances at the $(k+1)$-st and kth tree sections, respectively. Then, a compensation satisfying the following inequality:

$$C_k \geq \beta_{k+1} - \beta_k + T \tag{30}$$

for the kth missing tree section is sufficient for fairness such that if any path whose edge metric at the kth epoch is greater than or equal to the right-hand side, it is rejected by the threshold test.

A detailed proof is given in the Appendix.

We choose $C_k = \beta_{k+1} - \beta_k + T$ for tightness.

By considering a transmission block of length N_s, Lemma 1 leads to the following compensation rule.

Theorem 1: For an explored path pruned at $k = N_p$ where $1 \leq N_p < N_s$, a tightened compensation rule

$$S_{N_p} = \beta_{N_s} - \beta_{N_p} + (N_s - N_p) \times T \tag{31}$$

is fair.

Proof: From repeated application of Lemma 1, we note that

$$\begin{aligned} S_{N_p} &= C_{N_p} + C_{N_p+1} + \ldots + C_{N_s-1} && (32)\\ &\geq \left(\beta_{N_p+1} - \beta_{N_p} + T\right) + \left(\beta_{N_p+2} - \beta_{N_p+1} + T\right) + \\ &\quad \ldots + \left(\beta_{N_s} - \beta_{N_s-1} + T\right) \\ &= \beta_{N_s} - \beta_{N_p} + (N_s - N_p)T. && (33) \end{aligned}$$

Note that any choice of overall compensation S_{N_p} satisfying the inequality is fair. By selecting $C_k = \beta_{k+1} - \beta_k + T$, a tightened compensation is obtained which is $S_{N_p} = \beta_{N_s} - \beta_{N_p} + (N_s - N_p)T$. ∎

It should be noted that: 1) the per-section compensation C_k is the same for all paths of the same survival length and that 2) C_k varies from one tree section to another because the minimum forward distance varies from section-to-section.

The sum-product algorithm applied on the pruned-tree augmented with the compensation rule would render a robust receiver capable of handling a modest MIMO system up to a certain order of modulation size and a certain number of transmit antennas. In order to meet the high spectrum efficiency requirement set for future tactical military operations [4], it seems that the number of transmitter/receiver antennas desired is at least four and the constellation size may be selected up to 16 or even 64 QAM. We note that each individual survivor path is expanded into a set of M^{N_t} candidates. For $M = 16$, the number of candidates expanded from a single survivor is $16^4 = 65\,536$; if we have $M = 64$ and $N_t = 4$, the number is $64^4 \approx 17 \times 10^6$. It is not practical to consider all expandable paths. This motivates us to consider the sphere list detection algorithm at the tree-expansion phase. To only those selected candidates, the threshold-based tree-pruning can be applied.

VI. Sphere List Detection (SLD)

In the tree search operation, a survived node gets expanded into a number of candidates in the next time section; and the expanded paths are those candidates subject to the tree-pruning threshold test. When the constellation-size, or the number of transmit antennas—or both is large, the expansion step of the tree-search algorithm becomes a problem. The number of candidates per survived path M^{N_t} may become simply too large for a system to handle. For this, we aim to augment the tree-search algorithm with the introduction of the Fincke–Pohst path-expansion algorithm and attempt to reduce the number of candidates to a manageable size (i.e., on the order of 100 candidates per survivor on the average). Hochwald and Brink in [7] utilized the Fincke–Pohst algorithm [9], referring to it as the SLD algorithm, and demonstrated the robustness of the algorithms for the turbo-iterative decoding and demodulation problem in MIMO *flat*-fading channels. In [21], a class of the SLD algorithms developed in the field of applied mathematics have been investigated in the context of MIMO channels and a number of low-complexity list generation methods have been devised. In this paper, the Fincke–Pohst algorithm is applied for the MIMO ISI fading channels. As will be discussed in detail later, this marriage between the two algorithms works well.

SLD is applied at the expansion phase of the tree search routine. Note that each survivor node is the forefront of exploration along a particular path. A path of a certain length, say N_p, is defined by the N_p hypothetical channel symbol vectors stored in the path. Each survivor node in the tree gets expanded into a set of candidates of length $N_p + 1$. With SLD, only those candidates within the sphere of a prechosen radius R from the channel output are to be expanded. Further note that the ISI channel has L-taps. A received signal vector at a particular time epoch depends on $L-1$ previous channel symbol vectors. These previous channel symbol vectors are stored in the path.

It is also useful to note that by using these channel symbol vectors, the contribution of the previous symbols can be canceled out from the receiver vector. This is called the per-survivor processing technique. This cancellation is exact on the correct path, while it causes a further distortion and subsequent increases in the distance-measure of incorrect paths. As a consequence, this per-survivor ISI cancellation helps increase the number of candidates spawning from the correct path at the SLD path expansion phase; while it increases the distance measure of the candidates spawning from all incorrect paths, thus helping to eliminate the offspring of incorrect paths from the survivor lists much earlier.

For description of the cancellation step, we use $\boldsymbol{r}'_k$ to denote the ISI-cancelled received signal of a certain path at the kth epoch

$$\begin{aligned} \boldsymbol{r}'_k &:= \boldsymbol{r}_k - \sum_{l=1}^{L-1} \boldsymbol{h}_{k,l} \bar{\boldsymbol{x}}_{k-l} && (34)\\ &= \boldsymbol{h}_{k,0} \boldsymbol{x}_k + \boldsymbol{D}_k + \boldsymbol{n}_k && (35) \end{aligned}$$

where $\bar{\boldsymbol{x}}_{k-l}$ denotes the hypothetical channel symbol vectors stored in the path and $\boldsymbol{D}_k$ denotes the cancellation error such that $\boldsymbol{D}_k = \sum_{l=1}^{L-1} \boldsymbol{h}_{k,l}(\boldsymbol{x}_k - \bar{\boldsymbol{x}}_k)$. Note that $\boldsymbol{D}_k = 0$ on the correct path.

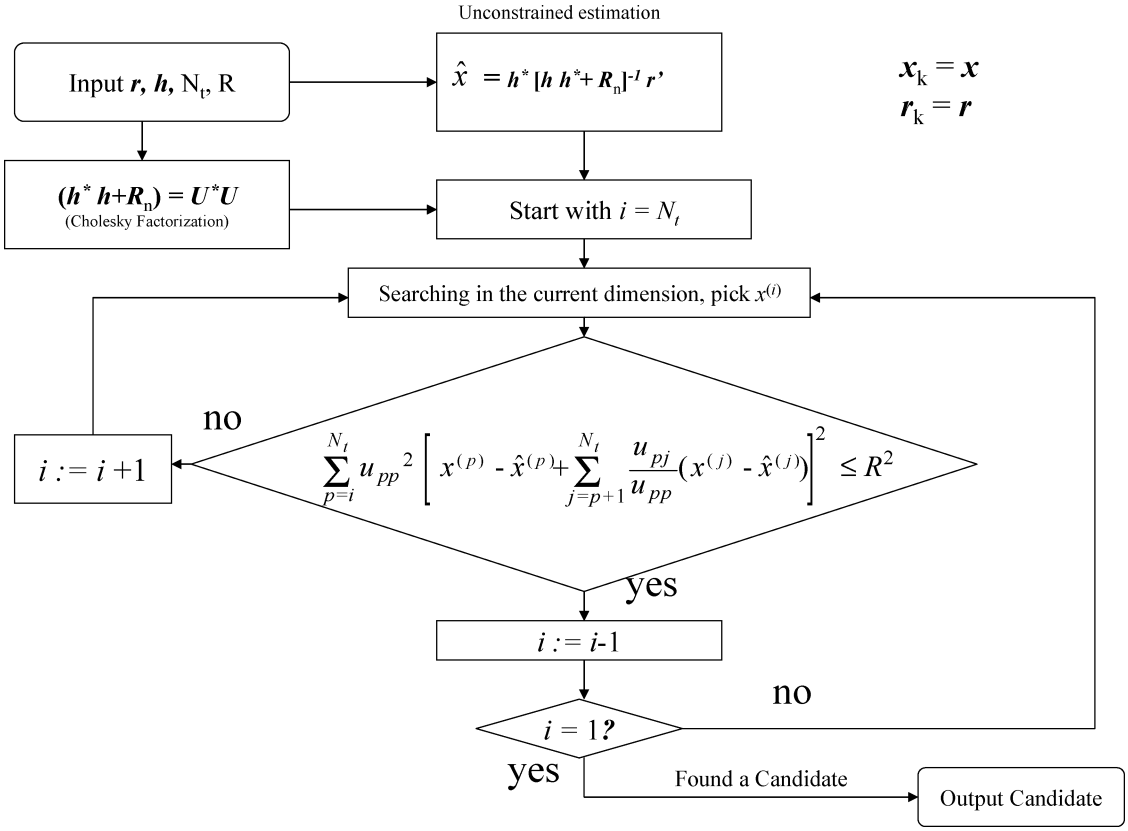

Fig. 3. Flowchart of the SLD.

A. Expansion of Each Survivor Using SLD

The aim of SLD is to generate a list of the most posterior-probable candidates given the received signal. After the cancellation process (35), the likelihood function of a particular symbol vector $\boldsymbol{x}_k$ is determined by the Euclidean distance (since the noise is Gaussian) such as

$$\|\boldsymbol{r}'_k - \boldsymbol{h}_{k,0}\boldsymbol{x}_k\|^2 \tag{36}$$

where $\boldsymbol{r}'_k$ is the received signal due only to the current input vector $\boldsymbol{x}_k$. We use this distance measure to find the list. Every possible signal vector $\boldsymbol{x}_k$ which results in clean channel outputs $\boldsymbol{h}_{k,0}\boldsymbol{x}_k$ within a certain distance from $\boldsymbol{r}'_k$ should be selected and stored into the list. With the Fincke–Pohst algorithm [9], we can accomplish this task. This process of sphere list detection shown in the flowchart in Fig. 3 starts with the unconstrained estimation of $\boldsymbol{x}_k$. Hochwald and Brink [7] used the ML estimator

$$\begin{aligned}\hat{\boldsymbol{x}}_{k,ML} &:= \arg\max_{\boldsymbol{x}_k \in \mathcal{C}^{N_t}} \mathrm{P}\left(\boldsymbol{r}'_k | \boldsymbol{x}_k\right) \\ &= \left(\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0}\right)^{-1}\boldsymbol{h}^*_{k,0}\boldsymbol{r}'_k\end{aligned} \tag{37}$$

where $\mathcal{C}$ denotes the set of the complex number. The ML estimator requires the random channel matrix $\boldsymbol{h}_{k,0}$ to be full rank. If it is not a full rank matrix, the estimator is not well defined and may incur a large estimation error. This motivates us to devise and use a regularized estimator. We propose the minimum mean-square estimator (MMSE)

$$\hat{\boldsymbol{x}}_k := \arg\min_{\boldsymbol{x}_k \in \mathcal{C}^{N_t}} E\left\{\|\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k\|^2\right\} \tag{38}$$

which can be written as

$$\hat{\boldsymbol{x}}_k = \left(\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0} + \boldsymbol{R}_n\right)^{-1}\boldsymbol{h}^*_{k,0}\boldsymbol{r}'_k \tag{39}$$

where $\boldsymbol{R}_n = N_0\boldsymbol{I}_{N_t \times N_t}$ is the noise covariance matrix. With the MMSE estimator $\hat{\boldsymbol{x}}_k$, the distance criteria can be revised as

$$\begin{aligned}\|\boldsymbol{r}'_k - \boldsymbol{h}_{k,0}\boldsymbol{x}_k\| &= (\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^* \left(\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0} + \boldsymbol{R}_n\right)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) \\ &\quad + \boldsymbol{r}'^*_k \left(I - \boldsymbol{h}_{k,0}\left(\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0} + \boldsymbol{R}_n\right)^{-1}\boldsymbol{h}^*_{k,0}\right)\boldsymbol{r}'_k.\end{aligned} \tag{40}$$

The second term in (40) is a constant with respect to different $\boldsymbol{x}_k$. Thus, the SLD criteria to find every candidate $\boldsymbol{x}_k$ can be summarized as

$$(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^* \left(\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0} + \boldsymbol{R}_n\right)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) \le R^2 \tag{41}$$

where R is the radius of the sphere. Using this criterion, the Fincke–Pohst algorithm can be applied.

We use the Cholesky factorization on $(\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0} + \boldsymbol{R}_n)$: $\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0} + \boldsymbol{R}_n = \boldsymbol{U}^*\boldsymbol{U}$, where $\boldsymbol{U}$ is an upper triangular $[N_t \times N_t]$ matrix. Since the $(\boldsymbol{h}^*_{k,0}\boldsymbol{h}_{k,0} + \boldsymbol{R}_n)$ is a positive definite matrix (the sum of two positive definite matrices is still positive definite), the upper triangular matrix $\boldsymbol{U}$ always exists with all diagonal elements being positive real numbers. Therefore, (41) can be written as

$$\begin{aligned}&(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^*\boldsymbol{U}^*\boldsymbol{U}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k) \\ &\quad = \sum_{i=1}^{N_t} u_{ii}^2 \left[x_k^{(i)} - \hat{x}_k^{(i)} + \sum_{j=i+1}^{N_t}\frac{u_{ij}}{u_{ii}}\left(x_k^{(j)} - \hat{x}_k^{(j)}\right)\right]^2 \\ &\quad \le R^2.\end{aligned} \tag{42}$$

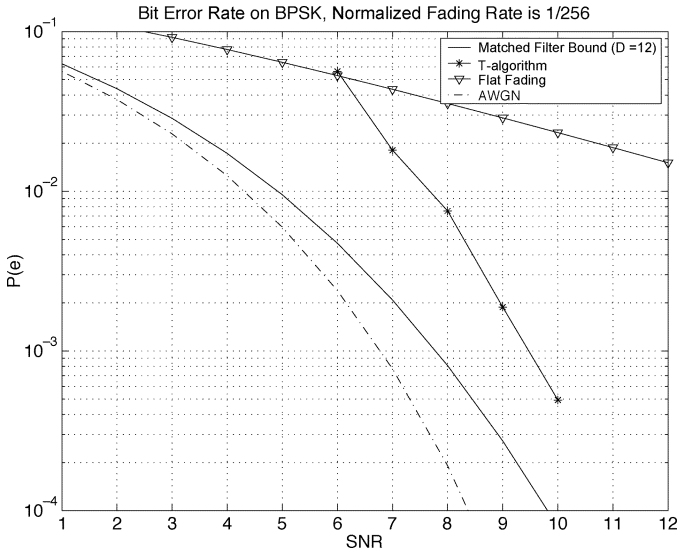

Fig. 4. ($N_t = 4$, $N_r = 4$, and $L = 3$.) MIMO system with BPSK ($M = 2$) modulation (no need for SLD). The threshold value for the tree search is $T = 4.0$.

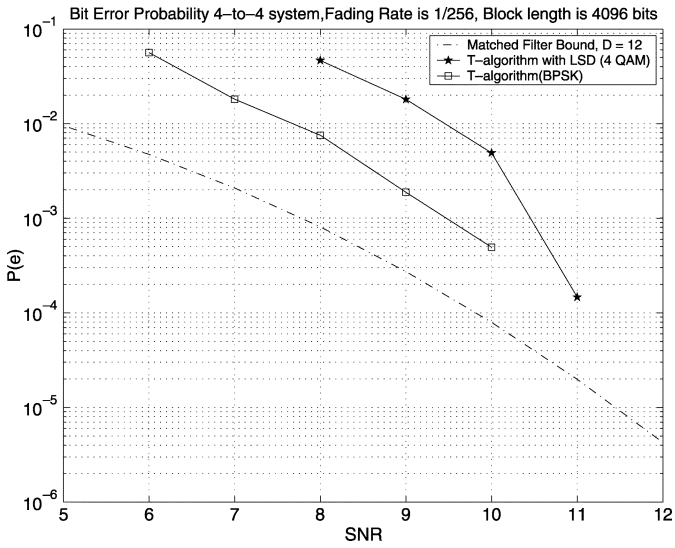

Fig. 5. ($N_t = 4$, $N_r = 4$, and $L = 3$.) MIMO system with 4-QAM ($M = 4$) modulation. The threshold value $T = 4.0$. The search radius R in SLD is 5.5.

It should be noticed that each ith summand, $i = 1, 2, \ldots N_t$, is nonnegative. The summation can be started from $i = N_t$ (without loss of generality) and down. As soon as the partial sum exceeds R^2 at a certain $i = t$, we note that the total sum will be bigger than R^2. Therefore, we can stop immediately. There is no need to proceed any further for all ith summands, for $i < t$.

Considering at $i = N_t$, we can choose all candidates for $x_k^{(N_t)}$ which satisfy

$$u_{N_t N_t}^2 \left(x_k^{(N_t)} - \hat{x}_k^{(N_t)} \right)^2 \leq R^2. \tag{43}$$

For each candidate $x_k^{(N_t)} \in \mathcal{S}$, we may continue to choose a candidate of $x_k^{(N_t-1)} \in \mathcal{S}$ by again using (42). Similarly, this process can be continued until $i = 1$. It is possible that no candidate could be found. The process can be continued by going one step back to the selection of $x_k^{(i+1)}$, until some $x_k^{(i)}$ is found. Fig. 3 shows the detailed flowchart of the proposed sphere list detection process.

VII. Simulation Results

Simulation experiments have been carried out to evaluate the performance of the proposed transceiver system. Several representative simulation results for ($N_t = 4$, $N_r = 4$, and $L = 3$) will be discussed in this section.

We let SNR denote the normalized signal-to-noise ratio (SNR) which is the SNR per information bit. Suppose SNR_r denote the SNR per received channel. Then, SNR is defined as

$$\mathrm{SNR} = \frac{N_r \mathrm{SNR}_r}{N_t R_t log_2(M)} \tag{44}$$

where R_t denotes the rate of the outer code. In addition, the Gray mapping from coded-bits to channel symbols is used.

A. Uncoded Results: MIMO System for BPSK/4-QAM Modulations ($N_t = 4$, $N_r = 4$)

Fig. 4 shows the performance of the proposed transceiver system for uncoded BPSK modulation. Without utilizing SLD, the tree-pruning algorithm, referred to as the T-algorithm, with threshold $T = 4.0$ is applied. The bit-error rate (BER) curve is drawn in comparison with the matched-filter bound (MFB) for a ($N_t = 4$, $N_r = 4$, and $L = 3$) system [5]. It can be observed from the BER curves that the diversity order of $N_r L = 12$ is achieved. Since uncoded modulation is used, no transmit diversity is achieved. An average number of around ten survivors is required to obtain this performance. We note that the BER obtained is within 2 dB of the ideal MFB.

The performance under 4-QAM system is shown in Fig. 5. In this case, the full complexity trellis has $M^{N_t(L-1)} = 4^6 = 65\,536$ states on each of which a posterior probability is needed to be calculated for each time epoch. Using the proposed tree search methods, it has been observed that the average number of candidate paths is around 20 at the output of the SLD routine and that of survivors in the tree-pruning algorithm is around 15. We note that the increase in computational complexity compared with the BPSK case is negligible. The BER is shown to be about 2 dB away from the MFB. In both cases, the channel simulated was slow fading with the normalized Doppler fading rate $f_{dm}T_s = 1/256$, where T_s is the period of channel symbol.

B. Coded Results: MIMO System With LDPCC ($N_t = 4$, $N_r = 4$)

We use the ($n = 4098$, $w_c = 4$, and $w_r = 8$) Gallager code as the LDPCC. For the simulation results given in this section, the channel is simulated as fast fading with the normalized Doppler fading rate $f_{dm}T_s = 1/16$. In addition, the threshold value has been set to $T = 4.0$ and the search radius of the SLD to $R = 5.5$.

We first evaluate the effectiveness of the compensation rule. Without the compensation rule applied, the equalizer does not have any other choice but to generate hard decision outputs. For the most part of the tree sections, there is a single path. Only toward the end of the tree, a number of candidates become available which can be used to formulate the soft-output messages. With the hard-output from the equalizer, the decoder makes the internal iteration for final decisions. There is no superiteration

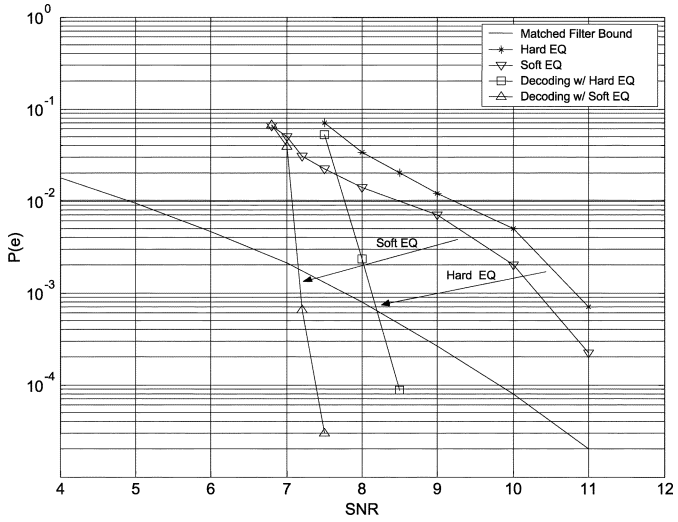

Fig. 6. BERs with hard- and soft-equalizer outputs.

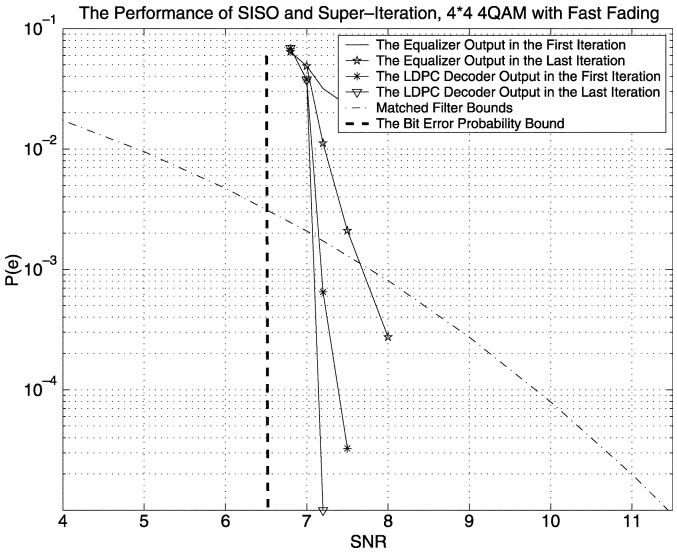

Fig. 7. BERs in three superiterations.

in this case. With the use of compensation rule, however, the equalizer can generate soft-output LLR messages which are utilized to initialize the message passing decoder. It is interesting to see the performance difference of these two schemes, which is in fact illustrated in Fig. 6. With the hard output equalizer, we note that, the LDPC decoder reaches 10^{-4} error rate at about 8.5 dB SNR. From the BER curves for the decoders using the soft equalizer outputs, we notice that, the BER is improved by about 1.0 dB.

As illustrated in Section IV and in (23) and (26), the equalizer based on the compensation rule makes use of the LLRs generated from the decoder and, thus, a superiterative receiver operation between the equalizer and the decoder can take place. Fig. 7 shows the effect of superiterations. After three superiterations, the error rate reaches 10^{-4} at around 7.2 dB.

As discussed in Section III, the upper bound on pairwise error-probability (18) can be used to generate an error-performance measure for the purpose of comparison with the simulation results. For the ensemble of ($n = 4096$, $w_c = 4$, and $w_r = 8$) LDPCCs, the exact calculation of the union bound for MIMO fading channels [14] can be obtained. However, an approximated error performance based only on the minimum-distance is generated in this paper. It is an ensemble average of all randomly selectable codes with a certain minimum distance. For an (4096,4,8) LDPCC, for example, the minimum distance $D_{\min} = \sigma n$ is 256 [16]. The pairwise error (18) can be averaged over an ensemble of channel symbol difference matrices $\Delta\boldsymbol{x}$. Each $[N_s N_t \times 1]$ vector $\Delta\boldsymbol{x}$ can be generated from the binary sequence of weight $D_{\min}$ which is randomly selected. For each matrix, the distance set can be found $\mathcal{D}$ in (18). This approximation is used as a comparison in Fig. 7.

VIII. System Computational Complexity Estimation

In this section, the relationship between the computational complexity and the system parameters is investigated. Analysis of computational complexity can help us determine the data processing speed required to obtain a particular performance result. For example, at a certain processing speed, a particular set of affordable system parameters such as the search radius R, the threshold T, the size of modulation M, and the numbers of transmit and receive antennas, N_t and N_r can be selected.

In the SLD phase, given a dimension $[N_t \times 1]$ of channel symbol vector $\boldsymbol{x}_k$ without loss of generality, we may choose to start the search from the N_tth coordinate and work our way down to the first one. Although most candidates searched would not appear on the final list, each candidate explored incurs a certain amount of computations. This overall operation of the algorithm can be represented as a tree, as was proposed in [22] and [23]. We may adopt and apply their mathematical framework to the MIMO frequency-selective fading channel. In each time epoch, instead of having a single path, a number of survivors are accepted, each of which leads to a certain amount of computations in the SLD process. Then, the overall complexity $V(N_t, N_r, N_0)$ can be estimated by

$$V(N_t,N_r,N_0)=\sum_{t=1}^{N_t} f_p(t)\sum_{l=0}^{\infty}\gamma\left(\frac{R^2}{2(N_o+l)},\frac{N_r-N_t+t}{2}\right)g_t^{(l)}$$

where $f_p(t) = 8t + 17$ which represents the number of floating point operations required for each candidate explored at the level t; $\gamma(\cdot,\cdot)$ denotes the incomplete Gamma function in the χ^2 distribution which represents the probability of a particular candidate having a distance less than R; $g_t^{(l)}$ denotes the number of constellation points in the t^{th} hyperspace with a distance of l in which the neighboring candidates have the unit distance. Making use of Euler's formula, $g_t^{(l)}$ can be computed for different base constellation sizes [22].

Fig. 8 shows the results for base constellation $\mathcal{S} = 4$-QAM with a different R. Similar curves can be obtained for $\mathcal{S} = 16$, 64-QAM. In Fig. 8, the number of computations used in systems with different N_t and N_r is shown. For the same search radius R, the computational complexity increases as the number of transmitter and receiver antennas increases as expected. In addition, for a particular system, increasing R will lead to a higher level of computational complexity as well. When the radius R reaches a certain limit, all constellation points are included in the sphere. Thus, any further increase of R will not increase the computational complexity.

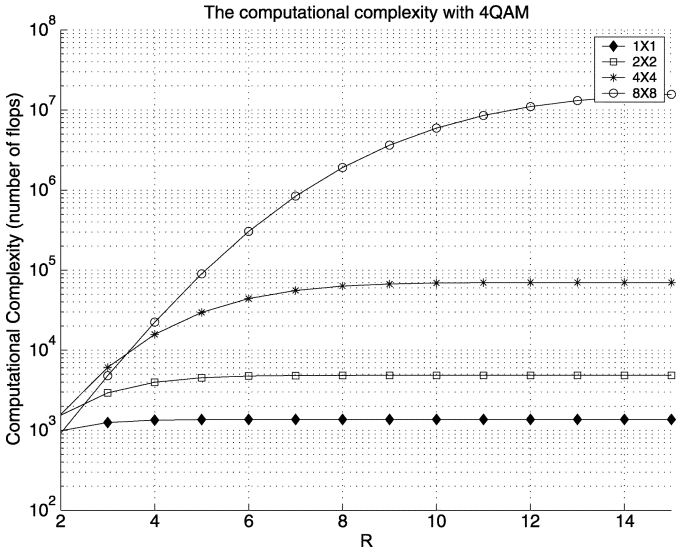

Fig. 8. Computational complexity as a function of the sphere radius R for the base constellation $\mathcal{S} = 4$-QAM. The SNR per transmit antenna is 10 dB.

IX. Conclusion

We have proposed a novel transmitter-receiver system design with a new reduced complexity tree search-based soft-input soft-output equalization method which can be combined with the graph decoder in a turbo-iterative manner. The simple but effective compensation rule allows the generation of more reliable soft-output messages and contributes to a significant reduction in the bit error rates. The proposed transmitter and receiver pair may be useful for future communication systems requiring very high spectrum efficiency operating over severely frequency-selective multiple transmit and receive antenna systems. The simulation results show that the low-complexity scheme performs robustly over the MIMO fading channels.

Appendix

In this section, we offer the proof of the compensation rule (Lemma 1).

Proof: Without loss of generality, we may use Fig. 2 in the proof paying particular attention to the last three tree sections of the pruned tree. Let β_k denote the minimum of the cumulative metrics at the kth tree section, i.e.,

$$\beta_k = \min_{m=1,2,\ldots,N_{sv}(k)} \{\alpha_k^m\} \tag{45}$$

where $N_{sv}(k)$ denotes the number of survivors at the kth tree section. As shown in Fig. 2, the pruned paths at a particular k are marked with "X." Thus, at the kth epoch, the zeroth and the first path are pruned because their cumulative metrics α_k^0 and α_k^1 are bigger than $\beta_k + T$, where T denotes a fixed constant for the threshold value of the tree-pruning algorithm. We note that since the zeroth and the first paths survived the threshold test at $k-1$, their, respectively, associated metrics α_{k-1}^0 and α_{k-1}^1 should satisfy the following inequalities:,

$$\alpha_{k-1}^0 \leq \beta_{k-1} + T$$
$$\alpha_{k-1}^1 \leq \beta_{k-1} + T.$$

Now, note that since all the offsprings of the zeroth node at $k-1$ are pruned at the kth section, the following inequality must be satisfied:

$$\alpha_{k-1}^0 + C_{k-1} \geq \beta_k + T \tag{46}$$

where C_{k-1} denotes the compensation applied to all paths pruned at the kth tree section.

On the other hand, we may write any cumulative metric survived at the $k-1$th section can be written as $\beta_{k-1} + \theta$, where $0 \leq \theta \leq T$. Thus, we have

$$\alpha_{k-1}^0 = \beta_{k-1} + \theta. \tag{47}$$

For instance, if $\theta = 0$, then $\alpha_{k-1}^0 = \beta_{k-1}$.

Substituting (47) into (46), we obtain

$$C_{k-1} \geq \beta_k + T - \beta_{k-1} - \theta. \tag{48}$$

Since $\theta \in [0, T]$, we have the following results:

$$C_{k-1} \geq \beta_k + T - \beta_{k-1}. \tag{49}$$

■

Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful comments.

References

[1] I. E. Telatar, "Capacity of multiantenna Gaussian channels," AT&T Bell Labs., Lucent Technologies, Tech. Memo, [Online]. Available: http://mars.bell-labs.com/papers/proof/, Oct. 1995.

[2] G. Foschini and M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Pers. Commun.*, vol. 6, pp. 311–335, Mar. 1998.

[3] L. Zheng and D. N. C. Tse, "Diversity and multiplexing: A fundamental tradeoff in multiple antenna channels," *IEEE Trans. Inf. Theory*, pp. 1073–1096, May 2003.

[4] J. A. Freebersyser, "Mobile network MIMO," DARPA ATO Briefings, Jun. 2003.

[5] V. Gulati and H.-N. Lee, "A low-complexity iterative per-antenna map equalizer for MIMO frequency-selective fading channels," in *Proc. IEEE GLOBECOM*, vol. 2, 2002, pp. 1118–1123.

[6] A. Stefanov and T. M. Duman, "Turbo-coded modulation for systems with transmit and receive antenna diversity over block fading channels: System model, decoding approaches, and practical considerations," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 5, pp. 958–968, May 2001.
[7] B. M. Hochwald and S. T. Brink, "Achieving near-capacity on a multiantenna channel," *IEEE Trans. Commun.*, vol. 51, pp. 389–399, May 2003.
[8] B. Dong, X. Wang, and A. Doucet, "A new class of soft MIMO demodulation algorithms," *IEEE Trans. Signal Process.*, vol. 51, no. 11, pp. 2752–2763, Nov. 2003.
[9] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complex analysis," *Math. Comput.*, vol. 44, no. 170, pp. 463–471, Apr. 1985.
[10] R. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Trans. Inf. Theory*, vol. IT–9, pp. 64–74, 1963.
[11] S. J. Simmons, "Breadth-first trellis decoding adaptive effort," *IEEE Trans. Commun.*, vol. 38, pp. 3–12, Jan. 1990.
[12] W. C. Jakes, *Microwave Mobile Communications*. Piscataway, NJ: IEEE Press, 1994.
[13] T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 2002.
[14] J. Zhang and H.-N. Lee, "Union bounds on LDPC-coded modulation over fast fading MIMO channels," *IEEE Commun. Lett.*, 2005, to be published.
[15] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space–time codes for high data rate wireless communications: Performance analysis and code construction," *IEEE Trans. Inf. Theory*, vol. 44, no. 5, pp. 744–765, Mar. 1998.
[16] R. G. Gallager, "Low-density parity-check codes," Ph.D. dissertation, MIT, Cambridge, MA, 1963.
[17] R. Gallager, "Low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.
[18] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electron. Lett.*, vol. 33, no. 6, pp. 457–458, Mar. 1997.
[19] T. J. Richardson, M. A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.
[20] H.-N. Lee. (2002) Channel coding theory and practice. [Online]. Available: http://crl.ee.pitt.edu/courses/ChannelCoding/
[21] M. O. Damen, H. El Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.
[22] B. Hassibi and H. Vikalo, "On sphere decoding algorithm I. Expected complexity," *IEEE Trans. Signal Process.*, to be published.
[23] ——, "On sphere decoding algorithm. II. Generalizations, second-order statistics and applications to communications," *Trans. Signal Process.*, submitted for publication.

Heung-No Lee (S'94–M'99) was born in Choong-Nam, Korea, and raised in Seoul, Korea. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of California, Los Angeles (UCLA), in 1993, 1994, and 1999, respectively.

From March 1999 to December 2001, he was with the Network Analysis and Systems Department, Information Science Laboratory of HRL Laboratories, Malibu, CA. At HRL, he led a number of research projects as the Principal Investigator including traffic modeling for tactical Internet (under the DARPA ATO ASPEN program), future tactical networking system, capacity analysis for satellite networks using realistic input using realistic input traffic and broadband wireless modem. His research interests during his tenure at UCLA include decision feedback equalization, trellis coded modulation, and channel estimation for fast time-varying delay-dispersive channels. He joined the faculty of the Electrical Engineering Department, University of Pittsburgh, Pittsburgh, PA, in January 2002. Since then, the primary areas of his research have been focused on advances in communication, information, and signal-processing theories for applications in future wireless networking systems. Specific research topics include iterative decoding and equalization, multiuser detection and its impact on network throughput, cross-layer ad hoc networking, network information theory, and channel-coding theorems for wireless networks.

Xinde Hu (S'03) received the B.S. degree in information science and electronic engineering from Zhejiang University, Zhejiang, China, in 2002 and the M.S. degree in electrical engineering from the University of Pittsburgh, Pittsburgh, PA, in 2004. He is currently working towards the Ph.D. degree in electrical and computer engineering at Carnegie Mellon University (CMU), Pittsburgh, PA.

His research interests at the University of Pittsburgh were iterative equalization and decoding methods for wireless transceivers. His current research interests at CMU are equalization for data storage and communication systems.