PARALLELISM/REGULARITY-DRIVEN MIMO DETECTION ALGORITHM DESIGN

Tong Zhang†, Yan Xin‡, and Sizhong Chen†

†Electrical, Computer and Systems Engineering Department
Rensselaer Polytechnic Institute, USA
‡Department of Electrical and Computer Engineering
National University of Singapore, Singapore

ABSTRACT

Efficient VLSI implementation of multiple-input multiple-output (MIMO) detectors plays an important role in the real-life implementation of MIMO communication systems. However, most high-performance MIMO detection algorithms developed so far largely lack the operational *parallelism* and *regularity* that are essential for high-throughput and low-power VLSI implementations. In this paper, following the theme of parallelism/regularity-driven algorithm design, we propose hard/soft-output MIMO detection algorithms that have high operational parallelism and regular/static data flow structure with fixed detection delay. Besides those properties desirable for VLSI implementations, such algorithms can achieve superior detection performance as demonstrated in the simulations.

1. INTRODUCTION

The signal detector is a key element in multiple-input multiple-output (MIMO) communication systems. Different detectors fall into two categories: hard-output detectors and soft-output detectors. Hard-output detectors only provide the hard estimation of the transmitted bits; soft-output detectors provide a posteriori probability (APP) information about each bit, which can be used to realize iterative detection in conjunction with an outer channel code such as low-density parity-check (LDPC) codes or Turbo codes.

In general the maximum-likelihood (ML) hard/soft-output MIMO detectors based on exhaustive search incurs prohibitive computational complexities, and therefore development of suboptimal detectors with reduced computational complexity attracted many attentions. One family of suboptimal detectors are linear detectors including detectors based on principles of minimum mean-square error (MMSE) and zero-forcing (ZF). Although they can dramatically reduce the computational complexity, they suffer from significant performance degradation. Their performance can be improved by certain techniques such as soft interference cancellation at the cost of increased computational complexity. To achieve a performance much closer to the optimal detection, researchers have proposed several nonlinear suboptimal hard/soft-output detectors that are based on the essentially same idea: approximate the ML exhaustive search by sequential non-exhaustive tree-search using a set of additive metrics. Different nonlinear suboptimal detectors mainly differ on how to perform the sequential search.

In this work, we are interested in the design of nonlinear suboptimal hard/soft-output detectors. As pointed out by ITRS (International Technology Roadmap for Semiconductors) [1], with the continuous scaling of CMOS technology, **parallelism** and **regularity** at the algorithm/architecture level are increasingly important for high-throughput and low-power VLSI implementations. Hence this work follows the theme of parallelism/regularity-driven algorithm design. Among various nonlinear suboptimal detectors, two different types of sequential tree-search have been used: *depth-first search* [2–4] using sphere decoding or stack algorithms and *breadth-first search* [5, 6] using *M*-algorithm [7]. From the VLSI implementation perspective, as demonstrated in [8], the latter is more favorable because of its higher operational parallelism and better regularity. In this work, inspired by the excellent parallelism/regularity of Viterbi algorithm for breadth-first trellis search, we re-formulate the original tree-search problem as a trellis-search problem, based on which hard-output and soft-output nonlinear suboptimal MIMO detectors are developed. They include the hard-output detector proposed in [5] as a special case when the parallelism is minimized. Besides the high parallelism/regularity, these algorithms can achieve very good detection performance as demonstrated in our simulations.

2. SYSTEM MODEL AND BACKGROUND

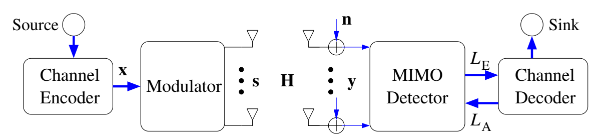

Fig. 1. Coded MIMO system model.

In this paper, we consider a MIMO system with *spatial multiplexing* signaling (i.e., the signals transmitted from individual antennas are independent of each other). Fig. 1 illustrates a coded MIMO system, where the soft-output MIMO detector and channel decoder work iteratively on the received data to approach the channel capacity. For an uncoded MIMO system, a hard-output MIMO detector is used. Let N_t and N_r represent the number of transmit and receive antennas, respectively. Assume the transmitted symbol is taken from a M-QAM constellation with $M = 2^q$. At once, the transmitter maps one $qN_t \times 1$ binary vector $\mathbf{x}$ to an $N_t \times 1$ symbol vector $\mathbf{s}$. The transmission of each vector $\mathbf{s}$ over MIMO channels can be modelled as $\mathbf{y} = \mathbf{H} \cdot \mathbf{s} + \mathbf{n}$, where $\mathbf{y}$ is an $N_r \times 1$ signal vector received by a MIMO detector, $\mathbf{H}$ is an $N_r \times N_t$ channel matrix, and $\mathbf{n}$ is a noise vector whose entries are independent complex Gaussian random variables with zero mean and variance $N_0/2$.

Following the principle of maximum likelihood (ML) detection, the task of the hard-output MIMO detector is to solve

$$\min_{\mathbf{x}\in\Omega} \|\mathbf{y} - \mathbf{H}\cdot\mathbf{s}\|^2, \tag{1}$$

where Ω contains all the M^{N_t} possible transmitted symbol vectors. The task of the soft-output MIMO detector is to compute the log-likelihood value of each bit, which is defined as

$$L(x_i|\mathbf{y}) = \ln\frac{P(x_i=+1|\mathbf{y})}{P(x_i=-1|\mathbf{y})} = L_A(x_i) + L_E(\mathbf{y}|x_i), \tag{2}$$

where L_A represents the *a priori* L-value provided by the channel decoder and L_E represents the so-called *extrinsic information* that is computed by the MIMO detector and fed to the channel decoder. Through standard simplification, $L(x_i|\mathbf{y})$ can be approximated as [3, 4]:

$$L(x_i|\mathbf{y}) \approx \max_{x_i=+1}\{\Lambda(\mathbf{x},\mathbf{y},L_A)\} - \max_{x_i=-1}\{\Lambda(\mathbf{x},\mathbf{y},L_A)\}, \text{where}$$

$$\Lambda(\mathbf{x},\mathbf{y},L_A) = -\frac{1}{N_0}\|\mathbf{y}-\mathbf{H}\cdot\mathbf{s}\|^2 + \frac{1}{2}\mathbf{x}^T\cdot\mathbf{L}_A, \tag{3}$$

and $\mathbf{L}_A$ denotes the vector whose entries are the L_A values. In a straightforward manner, hard/soft-output MIMO detection can be realized by *exhaustively* examining all the M^{N_t} possible symbol vectors according to (1) or (3), which nevertheless leads to computational complexity prohibitive for practical applications when N_t and/or M is large.

Using standard matrix decompositions such as Cholesky or QR decomposition, we can obtain $\mathbf{H}^*\mathbf{H} = \mathbf{L}^*\mathbf{L}$, where $\mathbf{L} = (l_{i,j})$ is a lower triangular matrix and $(\cdot)^*$ denotes the complex conjugate transpose. Let $\hat{\mathbf{s}} = (\mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*\mathbf{y}$, we have

$$\begin{aligned}\|\mathbf{y}-\mathbf{H}\cdot\mathbf{s}\|^2 &= (\mathbf{s}-\hat{\mathbf{s}})^*\mathbf{L}^*\mathbf{L}(\mathbf{s}-\hat{\mathbf{s}}) \\ &\quad + \mathbf{y}^*(\mathbf{I}-\mathbf{H}(\mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*)\mathbf{y}.\end{aligned} \tag{4}$$

Since the second term in (4) is independent of $\mathbf{s}$ and the matrix $\mathbf{L}$ is lower triangular, we can modify (1) and $\Lambda(\mathbf{x},\mathbf{y},L_A)$ in (3) as

$$\min_{\mathbf{x}\in\Omega}\Big(\sum_{i=1}^{N_t}\Big|\sum_{j=1}^{i} l_{i,j}(s_j-\hat{s}_j)\Big|^2\Big) = \min_{\mathbf{x}\in\Omega}\Big(\sum_{i=1}^{N_t}\Lambda_i^h\Big) \tag{5}$$

and

$$\begin{aligned}\Lambda(\mathbf{x},\mathbf{y},L_A) = \sum_{i=1}^{N_t}\Big(-\frac{1}{N_0}\Big|\sum_{j=1}^{i} l_{i,j}(s_j-\hat{s}_j)\Big|^2 \\ +\frac{1}{2}\sum_{j=(i-1)\cdot q+1}^{i\cdot q}(x_jL_A(x_j))\Big) = \sum_{i=1}^{N_t}\Lambda_i^s.\end{aligned} \tag{6}$$

Hence, we obtain *additive metrics* with the metric increments Λ_i^h and Λ_i^s that depend only on x_j for $j \le i$. This can be leveraged to design nonlinear suboptimal hard/soft-output detectors that *sequentially* and *non-exhaustively* search an N_t-depth M-ary tree as illustrated in Fig. 2(a), where each node has M child nodes labelled with $1, 2, \ldots, M$, respectively, corresponding to the M possible QAM points. The i-th depth of this tree corresponds to the i-th transmit antenna.

The tree can be searched by using either depth-first search algorithms or breadth-first search algorithms. Although they have similar computational complexities, breadth-first search algorithms have better parallelism/regularity than depth-first search algorithms. The breadth-first *M*-algorithm has been used in the design of a hard-output detector [6] and a soft-output detector [5]. We note that all the soft-output MIMO detectors developed so far cannot guarantee to find the two terms in the evaluation of $L(x_i|\mathbf{y})$ according to (3). To solve this problem, a fixed pre-defined value is used as the soft-output when the algorithms fail to find the two terms for certain bits. The performance is sensitive to this pre-defined value and it is not trivial to determine the appropriate value.

3. PROPOSED MIMO DETECTION ALGORITHMS

This section presents the hard-output and soft-output nonlinear suboptimal detectors developed under the theme of parallelism- and regularity-driven algorithm design. Belonging to the family of breadth-first search algorithms, they have two main advantages compared with prior work: (a) They have higher parallelism and better regularity, hence are more favorable to VLSI implementations; (b) The soft-output detector can always find the two terms in the evaluation of $L(x_i|\mathbf{y})$ according to (3), hence obviate the search for an appropriate pre-defined soft-output value as in other soft-output detectors.

3.1. Hard-Output MIMO Detector

Inspired by the excellent parallelism and regularity of the Viterbi algorithm that works on depth-invariant trellises, we propose to *fold* the original tree structure as shown in Fig. 2(a) to an N_t-depth trellis structure as shown in Fig. 2(b), based on which the breadth-first search for hard-output MIMO detection is carried out. There are three important parameters associated with this trellis: (i) u: there are u *states* at each depth; (ii) v: each state contains v sub-nodes corresponding to v distinct QAM points, and we have $u\cdot v = M$; (iii) p: each state-to-state transition channel contains at most p parallel branches, hence each state at most receives $p\cdot u$ incoming paths. To approximate the ML hard-output detection according to (5), we sequentially and non-exhaustively search through the trellis depth-by-depth by extending incoming paths and keeping certain number of survivor paths with best additive metrics at each state. The operation at each depth is outlined as follows:

1. *Path Extension*: Each state extends all the incoming paths (at most $p\cdot u$) with its v sub-nodes, which leads to totally at most $p\cdot u\cdot v = p\cdot M$ extended paths. Each path at the k-th depth has one path metric $\sum_{i=1}^{k}\Lambda_i^h$.

2. *Path Purge*: Given a threshold value T_k at the k-th depth, each state purges all its extended paths whose path metrics are worse than T_k.

3. *Path Search*: According to the path metrics, each state finds the best p extended paths, which are called *survivor paths* similar to the Viterbi algorithm. If the number of extended paths left after the path purge is less than p, we simply make all the extended paths as survivor paths. Finally, each state copies the survivor paths to all its u output channels towards next depth.

After the N_t-th depth, we obtain at most $p\cdot u$ survivor paths, along each path there are $q\cdot N_t$ bits. We select the one with the best path metric as the final survivor and output the $q\cdot N_t$ bits along this final survivor as the hard output. The above trellis search hard-output detection is similar to the reduced-state trellis search using Viterbi algorithm; the difference is that the additive metric in Viterbi algo-

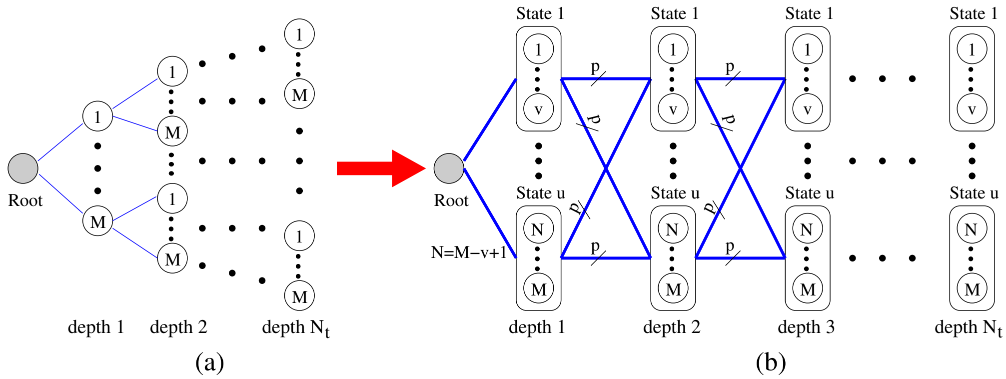

Fig. 2. (a) Original tree structure and (b) the trellis structure for MIMO detection.

rithm is only dependent on the current depth but the additive metric in this context is dependent on the current and all the priori depths along the path. Meanwhile, the operation within each state is similar to the *M*-algorithm, i.e., we keep the p best extended paths as survivors. The choice of $\{u, v, p\}$ plays a key role in the trade-off among detection performance, computational complexity, and detection speed:

(a) *Detection performance*: Intuitively, the more survivors (at most $p \cdot u$) are kept at each depth, the better the detection performance we can achieve. Moreover, even with the same value of $p \cdot u$, different choices of u and p will also lead to certain difference in performance (as shown by the simulation results in Section 4).

(b) *Computational complexity*: The total computational complexity is in the range of $\mathcal{O}(M^3)$. Within this range, we can reduce the complexity by reducing the value of $p \cdot u$.

(c) *Detection speed*: At each depth, the path extension and *search-the-best-p-paths* operations among all the u states can be carried out in fully parallel. Due to its serial essence, the search-the-best-p-paths operation at each state is the real speed limiter, which has a delay proportional to $p \cdot M$. Hence, subject to the same $p \cdot u$, larger value of u (i.e., higher parallelism) can help to reduce p, hence improve the detection speed. If $u = 1$ when the parallelism and hence speed is minimized, the detector will reduce to the one proposed in [6].

The role of threshold value T_k is similar to that of the radius r in sphere decoding. In this work, we simply make all T_k's equal to a fixed value T, where T is selected as $\|\mathbf{L}(\hat{\mathbf{s}}^{(d)} - \hat{\mathbf{s}})\|^2$ with $\hat{\mathbf{s}}^{(d)}$ denoting the closest symbol vector to $\hat{\mathbf{s}}$. Finally, we note that the overall data flow is very regular and static with fixed delay, similar to the Viterbi algorithm, which provides great potential on reducing the power consumption and improving throughput for VLSI implementation.

3.2. Soft-Output MIMO Detector

To realize soft-output MIMO detection, we need to find a set of candidate paths to obtain the two terms in the evaluation of $L(x_i|\mathbf{y})$ according to (3) for each bit x_i. Using the above hard-output detection, we obtain at most $p \cdot u$ survivor paths after the last depth. If we simply use those survivor paths as the candidate paths for soft-output detection, we cannot guarantee that we can always obtain the two terms in the evaluation of $L(x_i|\mathbf{y})$ for each x_i, since these paths may all agree on one bit position (i.e., all these paths contain a +1 (or -1) at the same position). Clearly, to solve this problem, we need more candidate paths. To this end, we propose to modify the above hard-output detector as follows to support soft-output detection:

(a) At the last depth (i.e., N_t-th depth) of the trellis search, instead of searching the best p paths among all the extended paths left by path purge in each state, we search for the best path among those paths extended by the same sub-node. If, after path purge, there is no extended paths that are extended by certain sub-node, then we recover those purged paths extended by this node and search the best one among them. In this way, after the last trellis depth, we will get v survivor paths at each state and totally $u \cdot v = M$ survivor paths, each one ends with one distinct sub-node. For each bit in the symbol transmitted by the N_t-th antenna, $M/2$ survivor paths will have the decision of +1 and the other $M/2$ will have -1. Hence, we can directly evaluate the $L(x_i|\mathbf{y})$ for all the q bits in the symbol transmitted by the N_t-th antenna.

(b) We perform the above modified trellis search process on N_t *differently ordered* trellises, where the last depth of each trellis corresponds to one distinct transmit antenna. We may consider this as performing the same trellis search on N_t re-ordered copies of the same received data. Meanwhile, we should permute the original channel matrix $\mathbf{H}$ corresponding to the N_t different orders. After applying matrix decomposition, we will have N_t different lower triangular matrices $\mathbf{L}$.

Moreover, the threshold value T in the trellis search is calculated as $T = -\frac{1}{N_0}\|\mathbf{L}(\hat{\mathbf{s}}^{(d)} - \hat{\mathbf{s}})\|^2 + \frac{1}{2}\mathbf{x}^T \cdot \mathbf{L}_A$. Applying the above re-ordered modified trellis search, we obtain totally $M \cdot N_t$ candidate paths, among which, for each bit x_i, at least $M/2$ paths have the decision of +1 and at least $M/2$ paths have the decision of -1. Therefore, we can always find the two terms in the evaluation of $L(x_i|\mathbf{y})$ for each x_i. We note that such soft-output MIMO detector has the following two features:

(i) It can guarantee the direct calculation of the soft-output $L(x_i|\mathbf{y})$ for all the bits, hence obviate the issue of determining an appropriate pre-defined fixed soft-output value as in all the previously proposed nonlinear suboptimal soft-output detectors;

(ii) At the first glance, it has the computational complexity N_t times higher than the hard-output detector. A closer observation suggests that, if two differently ordered data sequences are identical at the first t positions, the computation in the first t depth

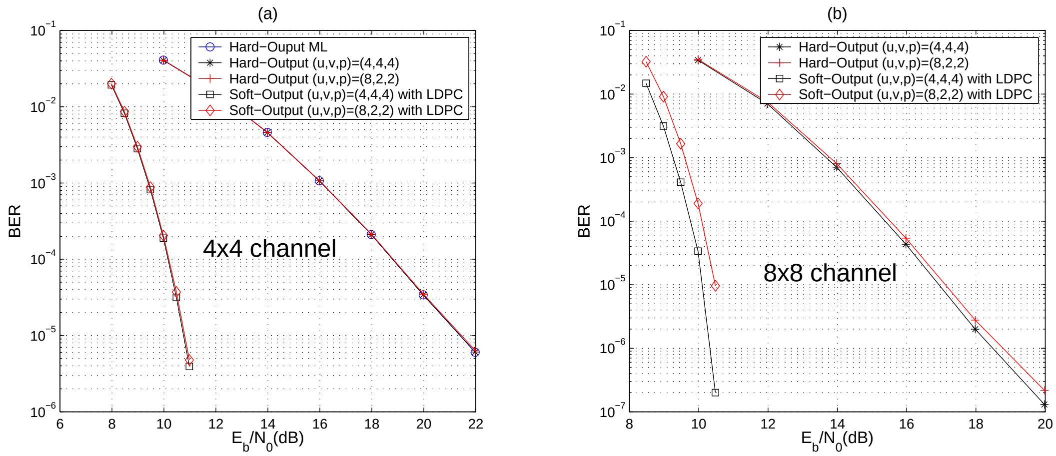

Fig. 3. Simulation results for (a) 4×4 16QAM, and (b) 8×8 16QAM.

trellis search can be shared. This provides a potential to further reduce the computational complexity. Clearly, different re-ordering scheme will lead to different computational complexity reduction, meanwhile the detection performance may be also different. Optimum choice of re-ordering scheme for maximized complexity reduction while realizing good detection performance is still under investigation.

4. SIMULATION RESULTS

In this work, we use the uncoded and coded MIMO orthogonal frequency division multiplexing (OFDM) system as a test vehicle to demonstrate the performance of the proposed detection algorithms. For high data rate wireless communication, OFDM can effectively mitigate the effects of intersymbol interference. We assume that the OFDM modulation employs 64-point FFT as in the IEEE 802.11a standard. For coded systems, we use a rate-1/2 LDPC code with the block size of 9216 bits, and we perform four iterations over the detection/decoding loop and three iterations within the LDPC decoder. Fig. 3 shows the simulated performance for $N_t = N_r = 4$ (i.e., 4×4 channel) and $N_t = N_r = 8$ (i.e., 8×8 channel). Let R denote the code rate ($R = 1$ for uncoded system), the definition of SNR follows the one presented in [3]:

$$\frac{E_b}{N_0}\Big|_{dB} = \frac{E_s}{N_0}\Big|_{dB} + 10\log_{10}\frac{N_r}{R \cdot N_t \cdot q},$$

where E_s denotes the average symbol energy of the M-QAM constellation.

5. CONCLUSIONS

This paper presents nonlinear suboptimal hard/soft-output MIMO detection algorithms with high operational parallelism and regularity that are of great importance from VLSI implementation perspective. The basic idea is to re-formulate the tree-search problem for nonlinear MIMO detection to a breath-first trellis-search problem leading to significant improvement on parallelism and regularity. For soft-output MIMO detection, a re-ordered trellis search scheme is proposed to guarantee the direct calculation of the soft output for all the bits in the received data vector. Besides the high parallelism and regularity, the good detection performance has been demonstrated in the simulations. Future research is directed to the development of the parallel VLSI architecture design and circuit implementation.

6. REFERENCES

[1] Semiconductor Industry Association, *The International Technology Roadmap for Semiconductors (ITRS)*, http://public.itrs.net/Files/2003ITRS/Home2003.htm, 2003.

[2] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Transactions on Information Theory*, vol. 45, pp. 1639–1642, July 1999.

[3] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Transactions on Communications*, vol. 51, pp. 389–399, March 2003.

[4] S. Baro, J. Hagenauer, and M. Witzke, "Iterative detection of MIMO transmission using a list-sequential (LISS) detector," in *Proc. of IEEE International Conference on Communications*, May 2003, pp. 2653–2657.

[5] Y. L. C. de Jong and T. J. Willink, "Iterative tree search detection for MIMO wireless systems," in *Proc. of Vehicular Technology Conference*, Sept. 2002, pp. II–24–II–28.

[6] K.-W. Wong, C.-Y. Tsui, R. S. Cheng, and W.-H. Mow, "A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels," in *IEEE International Symposium on Circuits and Systems*, May 2002, pp. III–273–III–276.

[7] J. B. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Transactions on Communications*, vol. 32, pp. 169–176, Feb. 1984.

[8] Z. Guo and P. Nilsson, "VLSI implementation issues of lattice decoders for MIMO systems," in *International Symposium on Circuits and Systems*, May 2004, pp. IV–477–IV–480.