Parallel Searching-Based Sphere Detector for MIMO Downlink OFDM Systems

Predrag Radosavljevic, *Member, IEEE*, Kyeong Jin Kim, *Senior Member, IEEE*, Hao Shen, *Student Member, IEEE*, and Joseph R. Cavallaro, *Senior Member, IEEE*

***Abstract*—In this paper, implementation of a detector with parallel partial candidate-search algorithm is described. Two fully independent partial candidate search processes are simultaneously employed for two groups of transmit antennas based on QR decomposition (QRD) and QL decomposition (QLD) of a multiple-input multiple-output (MIMO) channel matrix. By using separate simultaneous candidate searching processes, the proposed implementation of QRD-QLD searching-based sphere detector provides a smaller latency and a lower computational complexity than the original QRD-M detector for similar error-rate performance in wireless communications systems employing four transmit and four receive antennas with 16-QAM or 64-QAM constellation size. It is shown that in coded MIMO orthogonal frequency division multiplexing (MIMO OFDM) systems, the detection latency and computational complexity of a receiver can be substantially reduced by using the proposed QRD-QLD detector implementation. The QRD-QLD-based sphere detector is also implemented using Field Programmable Gate Array (FPGA) and application specific integrated circuit (ASIC), and its hardware design complexity is compared with that of other sphere detectors reported in the literature.**

***Index Terms*—ASIC, bounded soft sphere detector, Field Programmable Gate Array (FPGA), implementation complexity, latency analysis, QL decomposition (QLD), QR decomposition (QRD), QRD-M, QRD-QLD-based parallel candidate search.**

I. Introduction

EMERGING wireless receivers are intended to support hundreds of MBits/sec data-rates combined with excellent quality of service. In addition, a high and flexible spectral efficiency is desired, such that multiple transmit antennas are accompanied with multiple receive antennas forming multiple-input multiple-output (MIMO) wireless transceivers [1]. The main challenging problem in the design of a MIMO wireless transceiver is designing high-throughput low-cost MIMO receivers that efficiently mitigate strong inter-stream interference from a plurality of antennas. Current practical receivers employ the minimum mean squared error (MMSE) equalization combined with an outer channel decoder such as the low-density parity-check (LDPC) decoder [2]. Although the quality of *a posteriori* probabilities (APPs) of transmitted coded bits is improved, the error-rate performance is still far from the channel capacity when the spectral efficiency of the system is high.

To improve the error-rate performance while achieving high spectral efficiency, the soft sphere detector (SSD) is proposed in [3] to approximate the maximum *a posteriori* (MAP) data detector. This particular detector can operate close to the channel capacity if it is interfaced with the outer channel decoder. However, this detection scheme (i.e., a search for valid transmission candidates) is still too complex for an efficient hardware implementation. In recent years, the design of suboptimal SSDs have been investigated, such as in [4]–[7]. However, the latency of candidate search becomes large, especially if a number of transmit antennas or a modulation size is high, such as in the case of four transmit antennas and 16-QAM or 64-QAM modulations. It is, therefore, crucial to reduce the computational complexity of a candidate-search algorithm without sacrificing detection performance.

One promising data detector based on the QR decomposition (QRD) and M-algorithm, the QRD-M detector with a simple candidate search algorithm is proposed in [5]. The M-algorithm determines M survivor paths at each level of the search tree. The QRD-M detector has been well studied as a possible solution for downlink OFDM receivers employed in several emerging wireless standards, such as the 3GPP-LTE, IMT-Advanced, WLAN, WiMAX, etc. Using only a significantly reduced computational requirement, the QRD-M can approach near to the maximum likelihood detector (MLD) performance [5], [8]–[13]. However, since the original QRD-M detector still requires high computational complexity in calculating accumulated branch metrics, two categorized approaches have been proposed to reduce this problem. In [5], [8]–[11], the required computations can be reduced by controlling a number of survivor paths, whereas a number of branch metrics being used in one survivor path at each detection stage is selected effectively using reliable symbol information [12], [13]. However, in these works, error-rate performance of the original QRD-M detector has been sacrificed for reduction in computational complexity. In [14], a fixed-complexity sphere decoder (FSD) is proposed. The FSD provides

Manuscript received October 23, 2010; revised March 30, 2011, September 06, 2011, and January 10, 2012; accepted February 16, 2012. Date of publication March 12, 2012; date of current version May 11, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Warren J. Gross. This paper was supported in part by Nokia, Renesas Mobile, Texas Instruments, Xilinx, and by the US National Science Foundation by Grants EECS-0925942 and CNS-0923479.

P. Radosavljevic was with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA. He is now with Patterson & Sheridan LLP, Houston, TX 77056 USA (e-mail: pradosavljevic@gmail.com).

K. J. Kim was with Nokia Inc., Irving, TX 75039 USA. He is now with He is now at 9800 Sunrise Court, Irving, TX 75063 USA (e-mail: kyeong.j.kim@hotmail.com).

H. Shen and J. R. Cavallaro are with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA (e-mail: hao.shen@rice.edu; cavallar@rice.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2012.2190595

reduced computational complexity since a number of search operations is kept fixed to a desired predetermined number. However, error-rate performance of the FSD is worse than that of the K-best detector with $K = 16$ for 4×4 16-QAM (similar scheme to QRD-M, $M = 16$) and than that of the K-best detector with $K = 64$ for 4×4 64-QAM (similar scheme to QRD-M, $M = 64$).

In this work, implementation of a soft sphere detector with simplified candidate-search algorithm is described, where detection performance of the original QRD-M detector [5] is maintained with reduced computational complexity and detection latency. Different simplified candidate-search algorithms are proposed in [7], [15]–[17] as being incorporated into the bounded soft sphere detection (BSSD). In [7], the modified metric algorithm is proposed for data detection. The K-best list sphere detector (LSD) is proposed in [16], while the partial K-best detection for MIMO cooperative relay networks is proposed in [17]. In selecting K-best candidates, the K-best detector is identical to the M-algorithm. It is shown in [15] that the candidate-search algorithm can be divided into two independent parts to find the transmitted symbol candidates corresponding to two separate groups of antennas. These two partial candidate search processes can be simultaneously performed, which can decrease the detection latency.

In this paper, an implementation of the detection scheme based on the candidate search method from [18] is provided. Furthermore, hardware complexity is evaluated based on synthesis results targeting a field programmable gate array (FPGA) and an application specific integrated circuit (ASIC). In order to utilize an individual upper triangular matrix (QR part) and a lower triangular matrix (QL part) obtained by applying, respectively, QR decomposition (QRD) and QL decomposition (QLD) to a MIMO channel matrix, a set of transmit antennas is first divided into two groups (parts), i.e., a QR part and a QL part, respectively. Then, the QRD and QLD are individually applied on the QR part and the QL part, respectively. Following this way, an individual candidate-searching is applied to each of two antenna groups. Combining candidates obtained from both candidate-searching parts, final survivor candidates can be formed in a final detection process. The QRD-QLD-based parallel candidate search method maintains detection performance of the QRD-M type detector, while the proposed implementation of the QRD-QLD-based parallel search provides a reduced detection latency compared to the QRD-M detector. This paper builds upon error-rate performance and detection latency results from [18] to provide an efficient parallel detector architecture, which exploits the fact that the candidate search process comprises two independent parts that can be simultaneously performed.

It should be noted that the proposed detector implementation is also different than the detector from [11], which uses the permutation of the original channel matrix. The method of reducing a number of branches being accessed during the candidate-searching process is also presented in [19]. However, our proposed approach provides a lower computational complexity than the solution from [19], especially for higher order modulations. Detailed analysis for the processing latency and the required hardware complexity of the proposed detector is provided in this work and compared to those of the QRD-M detection algorithm.

The paper is organized as follows. A MIMO downlink system with inner soft sphere detection and outer soft-input soft-output decoding at a mobile terminal is introduced in Section II. A parallel candidate search algorithm based on the QRD and the QLD is described in Section III. Implementation of the QRD-QLD-based parallel detector (QRD-QLD-PD) is proposed in Section IV. Computational complexity, latency, and error-rate performance of the QRD-QLD-PD implementation are given in Section V. In Section VI, the QRD-QLD-PD is implemented using Xilinx FPGA and ASIC design. The paper is concluded in Section VII.

Notation: The superscripts $(\cdot)^T$ and $(\cdot)^H$ denote the transposition and conjugate transposition, respectively. $\boldsymbol{I}_N$ is the $N \times N$ identity matrix; the (i,j)th element of matrix $\boldsymbol{X}$ is denoted by $(\boldsymbol{X})_{i,j}$; the ith element of vector $\boldsymbol{x}$ is denoted by $(\boldsymbol{x})_i$; $\mathcal{CN}(\mu, \sigma^2)$ denotes the complex Gaussian distribution with mean μ and variance σ^2; real and imaginary parts of a complex number are denoted as $\mathrm{Re}(\cdot)$ and $\mathrm{Im}(\cdot)$.

II. MIMO-OFDM Downlink System and Sphere Detection

A coded MIMO downlink system based on orthogonal frequency division multiplexing (OFDM) with K subcarriers is considered in this work. Soft inner detector and outer channel decoder are used at a mobile terminal (receiver side). The transmitter and receiver are equipped with N_t and N_r antennas, respectively, whereas the N_t and N_r can take arbitrary integer values. A binary source bit stream $\{\boldsymbol{x}\}$ is fed into a channel encoder (e.g., an LDPC encoder, Turbo encoder, or a convolutional encoder, etc.). The encoded bit passes through a spatial constellation mapper followed by a serial-to-parallel converter. The resulting coded symbol blocks are denoted by $\{s_k^p(n)\}$ with $s_k^p(n)$ being the coded data symbol modulated over the kth subcarrier in the pth data stream during the nth OFDM block. Collecting $s_k^p(n)$ for $k = 0, \ldots, K-1$ into a vector, we have $\boldsymbol{s}_k(n) = [s_k^1(n), \ldots, s_k^{N_t}(n)]^T$. In addition, a data block $\boldsymbol{s}^p(n) = [s_1^p(n), \ldots, s_N^p(n)]^T$ drives the pth modulator. Applying the inverse discrete Fourier transform (IDFT), the modulated OFDM symbols from the pth transmit antenna becomes:

$$\boldsymbol{d}^p(n) = \boldsymbol{W}^H \boldsymbol{s}^p(n) \tag{1}$$

where $\boldsymbol{W}^H$ is the N-point IDFT matrix. A cyclic prefix (CP) of sufficient length is appended to the front of each modulated symbol to prevent inter-symbol interference (ISI). After the CP has been removed at the receiver side, the received signal is serial-to-parallel converted and then the N-point discrete Fourier transform (DFT) is applied.

After the DFT demodulation, the received vector signal over the subcarrier k becomes:

$$\boldsymbol{y}_k(n) = \boldsymbol{H}_k(n)\boldsymbol{s}_k(n) + \boldsymbol{n}_k(n) \tag{2}$$

where $\boldsymbol{y}_k(n)$ represents a vector of N_r received symbols, $\boldsymbol{H}_k(n)$ is a matrix of size $N_r \times N_t$ of flat-fading channel coefficients

that correspond to the kth subcarrier and $\boldsymbol{n}_k(n)$ is a vector of additive noise at the receiver side, which is denoted by $\boldsymbol{n}_k(n) \sim \mathcal{CN}(\boldsymbol{0}, \sigma_n^2 \boldsymbol{I}_{N_r})$. Since the channel is assumed to be quasi-static over transmission of one OFDM symbol comprising N subcarriers, the symbol index n will be suppressed in the sequel. However, the channel may vary between transmissions of successive OFDM symbols.

To find the maximum-likelihood (ML) solution, the ML detector tests all possible transmit symbol vectors in $|S|^{N_t}$ for obtaining the one with a minimum squared error:

$$\hat{\boldsymbol{s}}_{k,ML} = \arg \min_{\boldsymbol{s}_k \in |S|^{N_t}} \|\boldsymbol{y}_k - \boldsymbol{H}_k \boldsymbol{s}_k\|^2 \tag{3}$$

where $|S|$ denotes the cardinality of the signal constellation S.

The sphere detector simplifies the ML detector where its tested candidates are constrained to only those that are inside a hyper-sphere with a predetermined radius r formed around the received signal vector $\boldsymbol{y}_k$ [20]:

$$d(\boldsymbol{s}_k) = \|\boldsymbol{y}_k - \boldsymbol{H}_k \boldsymbol{s}_k\|^2 \leq r^2. \tag{4}$$

To achieve reduced computation in the sphere detection, (4) can be transformed into the identical problem applying the QR decomposition (QRD) to the channel matrix, that is, $\boldsymbol{H}_k = \boldsymbol{Q}_k \boldsymbol{R}_k$, where matrix $\boldsymbol{R}_k$ is an $N_t \times N_t$ upper triangular matrix, while matrix $\boldsymbol{Q}_k$ is an $N_r \times N_t$ unitary matrix. Replacing $\boldsymbol{H}_k$ by $\boldsymbol{Q}_k \boldsymbol{R}_k$, (4) becomes:

$$d(\boldsymbol{s}_k) = \left\|(\boldsymbol{Q}_k)^H \boldsymbol{y}_k - \boldsymbol{R}_k \boldsymbol{s}_k\right\|^2 \leq r^2. \tag{5}$$

Since the matrix $\boldsymbol{R}_k$ is the upper triangular, the distance $d(\boldsymbol{s}_k)$ can be calculated recursively from one transmit antenna to another:

$$\begin{aligned} P_m(\boldsymbol{s}_k) &= P_{m+1}(\boldsymbol{s}_k) + \left| (\tilde{\boldsymbol{y}}_k)_m - \sum_{j=m}^{N_t} (\boldsymbol{R}_k)_{m,j} (\boldsymbol{s}_k)_j \right|^2 \\ &\leq r^2, m = N_t, \ldots, 1, \end{aligned} \tag{6}$$

where $(\tilde{\boldsymbol{y}}_k)_m$ is the received signal from the mth receive antenna after multiplying by $(\boldsymbol{Q}_k)^H$, $(\boldsymbol{R}_k)_{m,j}$ is an entry of matrix $\boldsymbol{R}_k$ that belongs to the mth row and the jth column, $(\boldsymbol{s}_k)_j$ is the coded symbol candidate for the jth transmit antenna, and $P_m(\boldsymbol{s}_k)$ is a partial Euclidean distance (PED) of the candidate symbol $\boldsymbol{s}_k$ at the search level m. For all possible transmit symbol vectors $\boldsymbol{s}_k \in |S|^{N_t}$, we set $P_{N_t+1}(\boldsymbol{s}_k) = 0$.

By using Jacobian and Max-log approximation as in [21], [22], an extrinsic bit probability (i.e., reliability information of the transmitted coded bit) can be approximated as:

$$\begin{aligned} &L_E(x_i|\boldsymbol{y}_k) \\ &\quad \approx \frac{1}{2} \max_{\boldsymbol{x} \in \mathbb{X}_{i,+1}} \left\{ -\frac{1}{\sigma_n^2} \|\boldsymbol{y}_k - \boldsymbol{H}_k \boldsymbol{s}_k\|^2 + \boldsymbol{x}_{[i]}^T \boldsymbol{L}_{A,[i]} \right\} \\ &\quad - \frac{1}{2} \max_{\boldsymbol{x} \in \mathbb{X}_{i,-1}} \left\{ -\frac{1}{\sigma_n^2} \|\boldsymbol{y}_k - \boldsymbol{H}_k \boldsymbol{s}_k\|^2 + \boldsymbol{x}_{[i]}^T \boldsymbol{L}_{A,[i]} \right\}, \end{aligned} \tag{7}$$

where $\boldsymbol{s}_k = \mathrm{map}(\boldsymbol{x})$ with a mapper $\mathrm{map}(\cdot)$, $x_i \triangleq (\boldsymbol{x})_i$, $\boldsymbol{x}_{[i]}$ is the sub-vector of bits $\boldsymbol{x}$ obtained by excluding the ith bit x_i, and $\boldsymbol{L}_{A,[i]}$ is the vector of all *a priori* bit probabilities associated with the transmitted vector $\boldsymbol{x}$ obtained by excluding the *a priori* probability $L_A(x_i)$ of the ith bit x_i. In addition, $\mathbb{X}_{i,+1}$ is the set of bits representing $\boldsymbol{x}$ with $x_i = +1$, while $\mathbb{X}_{i,-1}$ is similarly defined. The Euclidian distance $d(\boldsymbol{s}_k) = \|\boldsymbol{y}_k - \boldsymbol{H}_k \boldsymbol{s}_k\|^2$ in (7) is computed recursively according to (6).

The computation of $L_E(x_i|\boldsymbol{y}_k)$ by using (7) is exponential function of a length of the transmitted vector $\boldsymbol{x}$ (i.e., function of a size of used constellation) and of the number of transmit antennas N_t. In order to simplify computation in (7), the list (soft) sphere detection is proposed in [3], and it is further modified in [23]. This scheme can approach the error-rate performance of optimal joint detection/decoding while avoiding exponential computational complexity in the number of transmit antennas and constellation size.

The soft sphere detection (SSD) algorithm is based on a simple modification/extension of the original sphere detection algorithm. Instead to find only the ML estimate, the SSD is able to find the list $\mathcal{L}$ of candidate symbols $\boldsymbol{s}_k$ inside the spherical region with a predetermined radius value. This list is large enough to comprise with high probability the symbol-vector $\boldsymbol{s}_k$ that maximizes (7). In order to find the list of candidates, the original sphere detector is modified. Every time when new candidate is found inside the sphere, the radius is not modified, but remains fixed. If the list is not full (i.e., if the predetermined number of candidates $Cand$ is not reached), then the newly found candidate is added to the list. If the list is full, an Euclidian distance (ED) of the newly found candidate is compared with the largest ED in the list; candidate with the largest ED is replaced with the newly found candidate if the ED of new candidate is smaller. In this work, a value of the radius is predetermined based on simulations in which data symbols are transmitted over different Rayleigh fading channels (channels with different fading levels) to ensure that a symbol-vector $\boldsymbol{s}_k$ that maximizes (7) belongs to the list $\mathcal{L}$ with a sufficiently high probability (e.g., probability of 0.99).

The list $\mathcal{L}$ of final candidates contains reliability (soft) information about each transmitted bit x_i. If most candidates inside the list have the bit $x_i = +1$, it can be concluded that it is highly probable that the transmitted value of bit x_i was indeed $+1$. Similarly, if only small number of candidates in the list have bit $x_i = +1$, the likely value for this bit is -1. By considering only candidates in the list $\mathcal{L}$, (7) can be modified as:

$$\begin{aligned} &L_E(x_i|\boldsymbol{y}_k) \\ &\quad \approx \frac{1}{2} \max_{\boldsymbol{x} \in \mathcal{L} \cap \mathbb{X}_{i,+1}} \left\{ -\frac{1}{\sigma_n^2} \|\boldsymbol{y}_k - \boldsymbol{H}_k \boldsymbol{s}_k\|^2 + \boldsymbol{x}_{[i]}^T \boldsymbol{L}_{A,[i]} \right\} \\ &\quad - \frac{1}{2} \max_{\boldsymbol{x} \in \mathcal{L} \cap \mathbb{X}_{i,-1}} \left\{ -\frac{1}{\sigma_n^2} \|\boldsymbol{y}_k - \boldsymbol{H}_k \boldsymbol{s}_k\|^2 + \boldsymbol{x}_{[i]}^T \boldsymbol{L}_{A,[i]} \right\}. \end{aligned} \tag{8}$$

A larger radius of the hyper-sphere corresponds to a larger list which makes the soft information about each transmitted bit more reliable. However, if the predetermined size of the list is smaller (if the sphere radius is smaller), the search process is faster and the detection throughput is increased. Obviously, there is a tradeoff between the reliability of soft information about coded bits and the detection throughput. It is important to note that the final candidates in the list $\mathcal{L}$ utilized in (8) are not equally probable: those candidates that are closer to the received

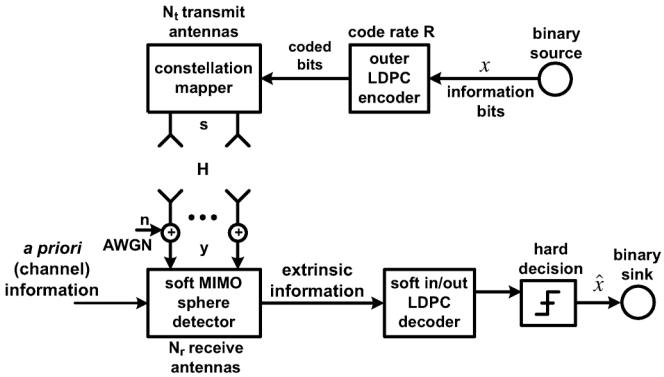


Fig. 1. MIMO transceiver with detection and decoding.

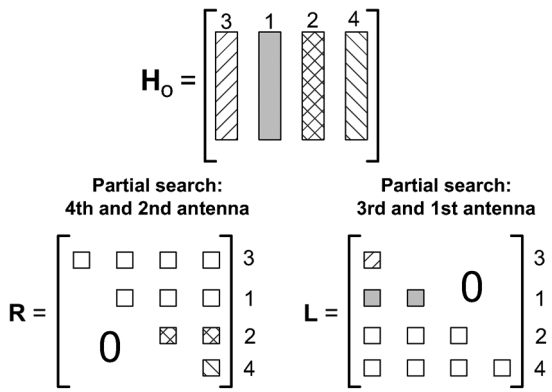


Fig. 2. Channel matrix $\boldsymbol{H}_o$ after applying the detection ordering of transmit antennas. Matrices $\boldsymbol{R}$ and $\boldsymbol{L}$ are obtained based on QRD and QLD, respectively, applied to the channel matrix $\boldsymbol{H}_o$. Two partial candidate search processes are performed simultaneously. Colored and marked elements of $\boldsymbol{R}$ and $\boldsymbol{L}$ are used in the candidate search processes.

point $\boldsymbol{y}_k$ with respect to the channel $\boldsymbol{H}_k$ are more probable and vice-versa.

III. QRD-QLD-Based Parallel Candidate-Search Algorithm

A target wireless OFDM system is assumed to have four transmit and four receive antennas; but the proposed candidate-search algorithm can be applied for any odd or even number of transmit and receive antennas. It is also assumed that the QRD-M detection starts from the most reliable antenna, i.e., an antenna with the largest signal-to-noise ratio (SNR). At each search level, the best M candidates are preserved for the next candidate search. The main idea of the QRD-QLD detection is to simultaneously search for partial candidates from two individual groups of transmit antennas, which represents themain difference from the original QRD-M.

The MIMO channel matrix corresponding to a subcarrier k is illustrated in Fig. 2 for a wireless system with four transmit and four receive antennas, along with an upper-triangular matrix $\boldsymbol{R}_k$ and a lower-triangular matrix $\boldsymbol{L}_k$ obtained after applying the QRD and the QLD, respectively on the MIMO channel matrix. It should be noted that the same triangular structure for matrices $\boldsymbol{R}_k$ and $\boldsymbol{L}_k$ can be obtained for any number of transmit antennas N_t, where the dimension of both matrices $\boldsymbol{R}_k$ and $\boldsymbol{L}_k$ is $N_t \times N_t$.

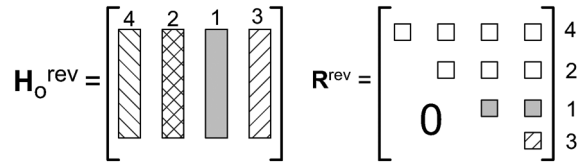


Fig. 3. Channel matrix with a reversed detection order of channel-columns $\boldsymbol{H}_o^{rev}$. The matrix $\boldsymbol{R}^{rev}$ is obtained by applying QRD on the $\boldsymbol{H}_o^{rev}$. Non-zero elements of $\boldsymbol{R}^{rev}$ are identical tonon-zero elements of $\boldsymbol{L}$ from Fig. 2.

In the case of $N_t = 4$, without loss of generality, the 4th transmit antenna can be assumed being the most reliable one. Further, the 3rd transmit antenna can be the next most reliable, whereas the 1st antenna can be the least reliable. To obtain this particular detection ordering, the approach proposed in [24] can be utilized. The partial search for coded symbol-candidates transmitted from the 4th and 2nd antennas is performed simultaneously with the partial candidate search corresponding to the 3rd and 1st antennas. In general, for N_t transmit antennas, the partial search for symbol candidates transmitted from the most reliable antenna, the third most reliable antenna, the fifth most reliable antenna, etc. can be performed based on elements of the upper-triangular matrix $\boldsymbol{R}_k$ starting from the row N_t, continuing with the rows $N_t - 1$ and $N_t - 2$, and so on. This partial search process can be performed simultaneously with the partial search for symbol candidates transmitted from the second most reliable antenna, the fourth most reliable antenna, the sixth most reliable antenna, etc., which is based on elements of the lower-triangular matrix $\boldsymbol{L}_k$ starting from the first row, continuing with the second and third rows, and so on. It should be also noted that in the case of odd number of transmit antennas, N_t antennas are divided into sets of $\lceil \frac{N_t}{2} \rceil$ and $N_t - \lceil \frac{N_t}{2} \rceil$ antennas.

Instead of applying the direct QLD to the equivalent channel matrix, the QRD can be applied to the reversed channel matrix, which imposes a reversed detection order in the original channel matrix. This leads to the identical result as shown in Fig. 2 and Fig. 3, and thus allowing the usage of same hardware blocks for a different antenna group. It should be noted that colored elements of the upper-triangular matrix $\boldsymbol{R}_k^{rev}$ in Fig. 3 are utilized in the partial search for candidates transmitted from the 3rd and 1st antennas. These elements are identical to colored elements of the lower-triangular matrix $\boldsymbol{L}_k$ from Fig. 2.

At the end of the partial candidate search process for the 4th and 2nd transmit antennas, the partial Euclidean distances (PEDs) are calculated for all transmission symbol candidates $(\boldsymbol{s}_k)_2$ for the 2nd transmit antenna with the parent transmission candidate $(\boldsymbol{s}_k)_4$ for the 4th transmit antenna. The partial vector candidate $(\boldsymbol{s}_k)_{4,2} \triangleq [(\boldsymbol{s}_k)_4, (\boldsymbol{s}_k)_2]^T$ is valid if it is inside the hyper-sphere with the radius r_R:

$$P_4\left((\boldsymbol{s}_k)_4\right) = \left|(\tilde{\boldsymbol{y}}_k)_4 - (\boldsymbol{R}_k)_{4,4}(\boldsymbol{s}_k)_4\right|^2 \tag{9}$$

and

$$P_2\left((\boldsymbol{s}_k)_{4,2}\right) = P_4\left((\boldsymbol{s}_k)_4\right) + \left|(\tilde{\boldsymbol{y}}_k)_2 - (\boldsymbol{R}_k)_{3,3}(\boldsymbol{s}_k)_2 - (\boldsymbol{R}_k)_{3,4}(\boldsymbol{s}_k)_4\right|^2 \leq r_R^2. \tag{10}$$

Similarly, for the partial candidate search process for the 3rd and 1st transmit antennas, the partial vector candidate $(\boldsymbol{s}_k)_{3,1} \triangleq [(\boldsymbol{s}_k)_3, (\boldsymbol{s}_k)_1]^T$ is valid if it is inside the hyper-sphere with the radius r_L

$$P_3\left((\boldsymbol{s}_k)_3\right) = \left|(\tilde{\boldsymbol{y}}_k)_3 - (\boldsymbol{R}_k^{rev})_{4,4}\,(\boldsymbol{s}_k)_4\right|^2 \tag{11}$$

and

$$P_1\left((\boldsymbol{s}_k)_{3,1}\right) = P_3\left((\boldsymbol{s}_k)_3\right) + \left|(\tilde{\boldsymbol{y}}_k)_1 - (\boldsymbol{R}_k^{rev})_{3,3}\,(\boldsymbol{s}_k)_2 - (\boldsymbol{R}_k^{rev})_{3,4}\,(\boldsymbol{s}_k)_4\right|^2 \le r_L^2. \tag{12}$$

Depending on the channel quality for an individual antenna group, a different value for r_L and r_R can be used. In this work, for simplicity, we use $r_R = r_L = r$, where the value of radius r is predetermined as described in Section II. Two separate and independent candidate searches for two groups of antennas are represented by (10) and (12). The computational complexity of the presented detection approach is reduced compared to the QRD-M algorithm thanks to a lower complexity of these two partial candidate searches.

In the case of original QRD-M detector [5], sorting of candidates is applied at every search level and up to M most reliable candidates are preserved for further searching. Therefore, up to M final candidates are available at the end of the candidate search process. However, in the described implementation of the QRD-QLD-PD, two simultaneous partial candidate search processes are performed for two groups of transmit antennas. The number of valid partial candidates is upper-bounded by $Cand$ candidates at the end of each partial search processes (i.e., upper-bounded by the size of candidate list defined in Section II). Then, sorting of partial candidates can be applied, and two groups of M most reliable partial candidates out of $Cand$ partial candidates are preserved. Both groups of partial candidates are then fully combined to generate the list of M^2 final candidates for all transmit antennas. It should be noted that the value of $Cand$ can be predetermined such that to provide a sufficiently low probability (e.g., less than 1% on average) that a correct (i.e., transmitted) candidate is not found during a corresponding partial search process.

IV. Implementation of QRD-QLD-Based Parallel Soft Sphere Detector

High-level architecture design of the QRD-QLD-based parallel soft sphere detector (QRD-QLD-PD) is illustrated in Fig. 4. Four transmit and four receive antennas are assumed to be equipped in a wireless communication system with 16-QAM or 64-QAM subcarrier modulation. The arithmetic logic comprising two parallel search modules is interfaced with random access memory (RAM) modules and register files for storing partial symbol candidates. Two parallel arithmetic blocks are employed for computation of PEDs, as well as for the search of candidates for two groups of transmit antennas. Valid candidates for each transmit antenna (one search level) and its corresponding PEDs are stored in a RAM block used in a subsequent search level. Both groups of partial candidates are first stored in register files and then passed via interface networks into sorting units for finding M best partial candidates

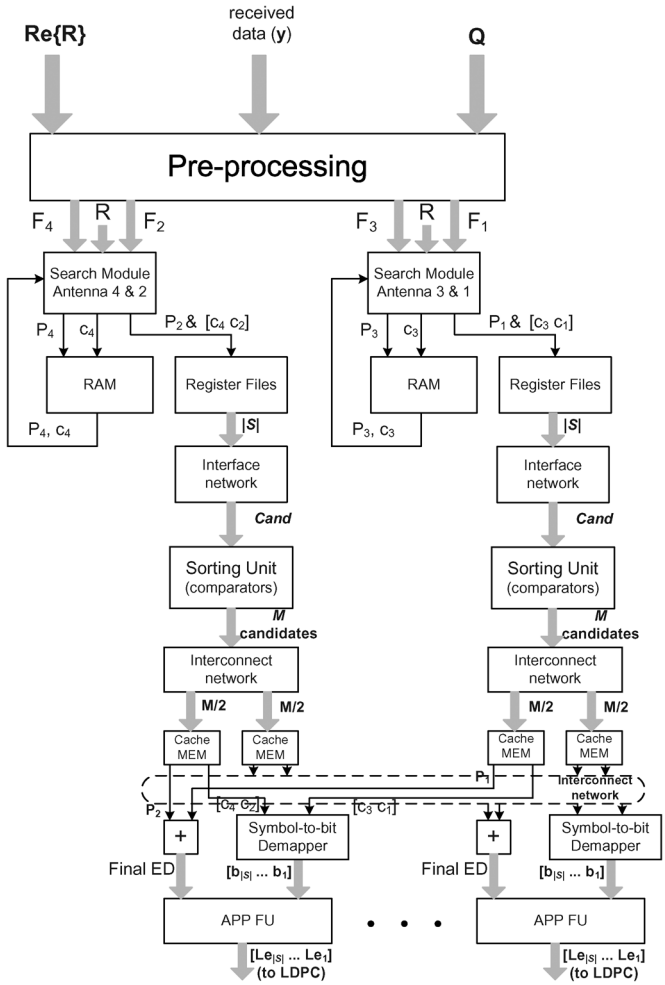


Fig. 4. Block-diagram of the QRD-QLD-based parallel soft sphere detector.

(i.e., candidates with smallest PEDs) for each group of candidates. Best M partial candidates for each group are then passed via interconnect networks to appropriate cache memories, as illustrated in Fig. 4.

It can be observed from Fig. 4 that up to M^2 final symbol candidates can be obtained by combining each pair of partial candidates through a final interconnect network stage and symbol-to-bit de-mappers. A final ED associated with each final symbol candidate can be obtained by summing two PEDs of two partial candidates that correspond to that final symbol candidate. The final EDs and bits corresponding to a set of the final symbol candidates are then utilized at a time by parallel APP function units for obtaining reliability information (APP messages) for the bits associated with these four final symbol candidates. Finally, in the case of 16-QAM, groups of 16 APP messages are available at a time for storage into a memory of an outer channel decoder (e.g., an LDPC decoder) interfaced with the QRD-QLD-PD from Fig. 4. Similarly, in the case of 64-QAM, groups of 64 APP messages are available at a time for storage into the memory of the outer decoder.

A. Preprocessing Unit

The preprocessing unit illustrated in Fig. 4 calculates a center of the hyper-sphere, as well as the common factors defined in

(10) and (12) to test all symbol-candidates for each transmit antenna. The center of hypersphere can be defined as

$$\tilde{\boldsymbol{y}}_k = (\boldsymbol{Q}_k)^H \boldsymbol{y}_k. \tag{13}$$

Since $\boldsymbol{R}_k$ is fully known, certain factors that are not dependent on the symbol-candidates being tested can be precomputed according to

$$F_m \triangleq (\tilde{\boldsymbol{y}}_k)_m - (\boldsymbol{R}_k)_{m,m}(\boldsymbol{s}_k)_m, \quad m = N_t, \ldots, 1. \tag{14}$$

The factors obtained in (14) can be saved in the registers and utilized in the appropriate search level m. It can be observed that computing all $|S|$ products $(\boldsymbol{R}_k)_{m,m}(\boldsymbol{s}_k)_m$ in (14) for each search level is not required. For example, only eight of these products are different in the case of 16-QAM, i.e., $L = 4$ real parts and $L = 4$ imaginary parts, where L corresponds to the number of 16-QAM constellation levels. Similarly, only 16 of these products are different in the case of 64-QAM, i.e., $L = 8$ real parts and $L = 8$ imaginary parts, where L corresponds now to the number of 64-QAM constellation levels. Furthermore, rather than computing products $(\boldsymbol{R}_k)_{m,m}(\boldsymbol{s}_k)_m$, it is more efficient applying shift/add operations on $(\boldsymbol{R}_k)_{m,m}$ thanks to the known levels of constellation points.

B. Search Modules

Each search module illustrated in Fig. 4 associated with a partial search process simultaneously computes PEDs for all $|S|$ constellation points (i.e., PEDs $\{P_4, P_2\}$ or PEDs $\{P_3, P_1\}$). All PEDs that are computed in a single search operation correspond to a common partial candidate found in the previous search level. Once computed, all PEDs are simultaneously tested whether they are inside the hyper sphere. Then, up to $P_C = |S|$ valid candidates along with their PEDs are saved in the memory for later use. It should be noted that the detection order is irrelevant for the architecture design. In this particular case, the detection order is: antenna 4 followed by antenna 2, simultaneously performed for antenna 3 followed by antenna 1.

The search-module for a first detected transmit antenna (i.e., 4th transmit antenna or the most reliable antenna determined after reordering of channel columns) computes PEDs for all $|S|$ constellation points and checks if they are inside the predetermined hypersphere according to

$$P_4^q = |F_4^q|^2 \leq r^2; \quad q = 1, \ldots, |S|. \tag{15}$$

For every valid candidate c_4 from the first search level, accumulated PEDs of the fourth and second transmit antennas are computed as follows:

$$P_2^q = P_4(c_2) + \left| F_3^q - (\boldsymbol{R}_k)_{4,3} c_4 \right|^2 \leq r^2; \quad q = 1, \ldots, |S|. \tag{16}$$

Similarly, the other search module computes $\{P_3^q\}$, and then it accumulates PEDs $\{P_1^q\}$ for every valid candidate $\{c_3\}$. If the maximum predetermined number of candidates for two initial search levels is found, then the search process stops. Partial vector-candidates are combined into M^2 final vector-candidates $[c_4, c_2, c_3, c_1]^T$ after determining the best M candidates for two pairs of transmit antennas.

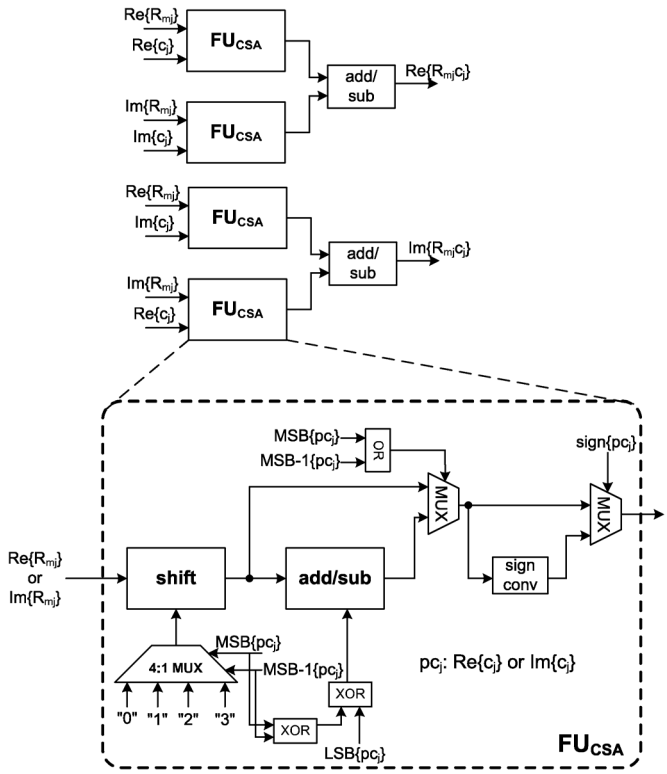


Fig. 5. Block diagram of arithmetic logic for computation of $(\boldsymbol{R}_k)_{m,j} c_j$ for a single value of j in the case of 16-QAM. Block diagram of the check/shift/add FU $\mathrm{FU_{CSA}}$ for computation of partial $(\boldsymbol{R}_k)_{m,j} c_j$ products using bit-checking, shift and add operations.

For fully parallel computation of all products $(\boldsymbol{R}_k)_{m,j} c_j$ inside every search level j ($j = m, \ldots, N_t$), up to twelve $\mathrm{FU_{CSA}}$ function units (check/shift/add function units) and six add/subtract units are required in the case of four transmit antennas. In the $\mathrm{FU_{CSA}}$ illustrated in Fig. 5, because of the known constellation levels of symbol-candidates, multiplication operation can be performed using simpler arithmetic operations, such as: bit-checking, shift and add operations. The $\mathrm{FU_{CSA}}$ first checks the constellation level of a symbol-candidate c_j for the jth transmit antenna using a 4:1 multiplexer and XOR logic, as illustrated in Fig. 5. Then, the $\mathrm{FU_{CSA}}$ performs the appropriate shift, add/subtract operation and sign-conversion on the real and imaginary parts of $(\boldsymbol{R}_k)_{m,j}$. The arithmetic logic utilized for the computation of a single factor $(\boldsymbol{R}_k)_{m,j} c_j$ in the case of 16-QAM is shown in Fig. 5. It should be noted that, in the case of 64-QAM, the 4:1 multiplexer in Fig. 5 is configured as an 8:1 multiplexer with eight binary inputs, where selecting signals originate from three most significant bits of pc_j (where pc_j is either $\mathrm{Re}\{c_j\}$ or $\mathrm{Im}\{c_j\}$, as defined in Fig. 5). Similarly, the OR and XOR circuits in Fig. 5 are configured as three-input circuits with inputs that originate from three (instead of two) most significant bits of pc_j.

It can be observed that the computation of Euclidean distances from (16) can be rewritten as:

$$P_2 = P_4(c_4) + |\mathrm{Re}\{X\} + i\mathrm{Im}\{X\}|^2 \leq r^2 \tag{17}$$

where $X = F_3 - (\boldsymbol{R}_k)_{4,3} c_4$. There are four different values of $\mathrm{Re}\{X\}$ and four different values of $\mathrm{Im}\{X\}$ in the case of

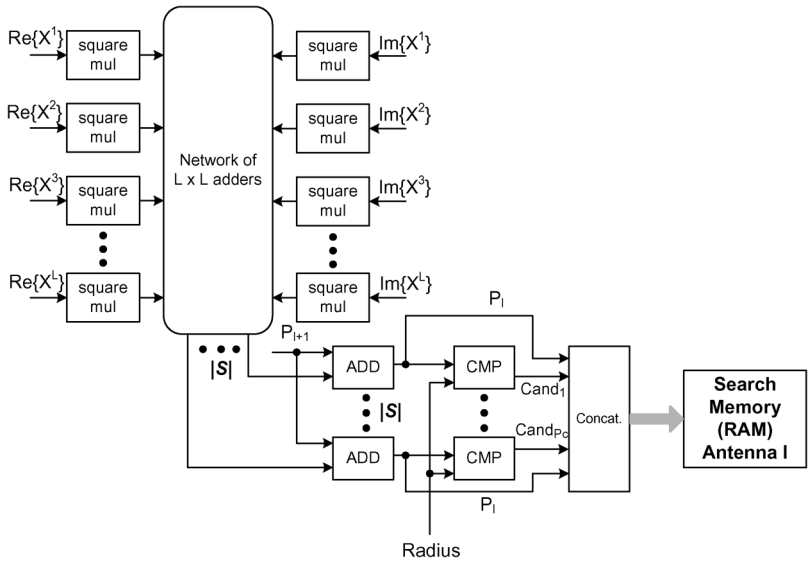


Fig. 6. Final stage of the search module for a transmit antenna I.

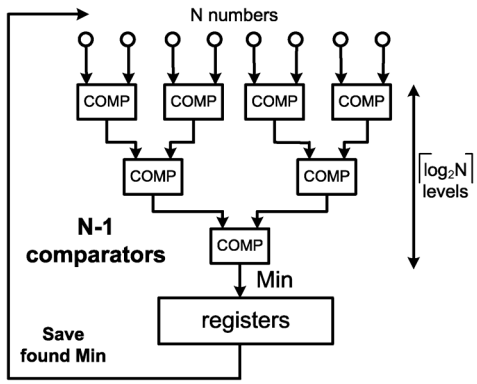


Fig. 7. Sorting function unit composed of binary tree of comparators.

16-QAM. A fully parallel computation of $|X|^2$ requires $2L = 8$ square multipliers, such that $P_C = |S| = 16$ different values of $|X|^2$ are formed by the cross-addition of all individual results of square multiplications, as shown in Fig. 6 for an arbitrary transmit antenna I. Similarly, in the case of 64-QAM, there are eight different values of $\mathrm{Re}\{X\}$ and eight different values of $\mathrm{Im}\{X\}$. In this case, a fully parallel computation of $|X|^2$ requires $2L = 16$ square multipliers, such that $P_C = |S| = 64$ different values of $|X|^2$ are formed by the cross-addition of all individual results of square multiplications, as shown in Fig. 6.

C. Sorting Unit

Each sorting function unit illustrated in Fig. 4 associated with a partial candidate search process can comprise a binary tree of comparators, as illustrated in Fig. 7. For $Cand = N$ PEDs to be sorted for finding M smallest values (i.e., M best partial symbol candidates), $N-1$ comparators can be connected in $\lceil \log_2(N) \rceil$ stages. Once a global minimum PED is found, this particular PED is removed from the list and then the comparison process is repeated to find a second smallest PED. The comparison process is repeated in this way M times until M best partial candidates are obtained for a pair of transmit antennas.

D. APP Function Unit

Each of APP function units illustrated in Fig. 4 simultaneously computes *a posteriori* probabilities for a set of coded bits according to (8). De-mapping of the final vector of the symbol candidate for $\{\boldsymbol{s}_k\}$ into bits $\{\boldsymbol{x}\}$ is required. It can be noticed from (8) that already computed Euclidean distance $ED(\boldsymbol{s}_k)$ corresponding to the final vector candidate $\boldsymbol{s}_k$ can be directly utilized for computing the extrinsic probabilities, i.e.

$$\begin{aligned} L_E(x_i|\boldsymbol{y}_k) &\approx \frac{1}{2}\max_{\boldsymbol{x}\in\mathcal{L}\cap\mathbb{X}_{i,+1}}\left\{-\frac{1}{\sigma_n^2}ED(\boldsymbol{s}_k)+\boldsymbol{x}_{[i]}^T\boldsymbol{L}_{A,[i]}\right\} \\ &\quad -\frac{1}{2}\max_{\boldsymbol{x}\in\mathcal{L}\cap\mathbb{X}_{i,-1}}\left\{-\frac{1}{\sigma_n^2}ED(\boldsymbol{s}_k)+\boldsymbol{x}_{[i]}^T\boldsymbol{L}_{A,[i]}\right\}. \end{aligned} \quad (18)$$

To simplify the computation of (18), an inner product between the *a priori* probabilities and coded bits is calculated using the sign-conversion and summation. Then, the term $\boldsymbol{x}_{[i]}^T\boldsymbol{L}_{A,[i]}$ is computed while excluding the quantity $L_A(x_i)$, that is

$$\boldsymbol{x}_{[i]}^T\boldsymbol{L}_{A,[i]} = \boldsymbol{x}^T\boldsymbol{L}_A - \mathrm{sign}(x_i)L_A(x_i), \quad \forall i. \quad (19)$$

The updated extrinsic probabilities $L_E(x_i|\boldsymbol{y}_k)$ are directly stored in a RAM of the outer LDPC decoder (not shown in Fig. 4) interfaced with the QRD-QLD-PD.

V. Complexity and Latency Analysis of QRD-QLD-Based Parallel Detector, and Detection Performance

A. Computational Complexity

Table I shows a number of arithmetic units for the candidate search and APP computation. The final stage of the search module for each transmit antenna shown in Fig. 6 is considered. In Table I, N_t represents a number of transmit antennas, L is a number of constellation levels (e.g., four levels for 16-QAM,

TABLE I
NUMBER OF ARITHMETIC UNITS FOR CANDIDATE SEARCH AND APP COMPUTATION

Arithmetic Unit	Search Module	APP FU
Full Adders	$2L(N_t+1)+2$	$4N_tM_C+5N_t-7$
Multipliers	-	1
Square Mults.	$2L$	-
Buffers	$L(2N_t+1)+8$	$3N_tM_C+2$
Shift regs.	-	N_tM_C
Comparators	N_tM_C	N_tM_C
b-bit (2:1)MUXs	3	$2N_tM_C+1$
Demapper	-	1
$\mathrm{FU_{CSA}}s$	$4(N_t-1)$	-
2-input XORs	$6bN_t$	$2bN_t$
Fund. Arithmetic Operation	**Add**	**Compare**
search operation	$2^{M_c+1}+2\sqrt{2^{M_C}}$	2^{M_C}
$\mathrm{sorting_{QRD\text{-}M}}$	-	$2^{M_C}(N_t-1)M^2$
$\mathrm{sorting_{QRD\text{-}QLD}}$	-	$\frac{N_t}{2}MCand$

eight levels for 64-QAM), $M_C \triangleq \log_2|S|$ is a number of information bits that represent one constellation symbol, and b represents a fixed-point arithmetic word length. The check-select-add $\mathrm{FU_{CSA}}$ function unit referred in Table I is described in Section IV-B and illustrated in Fig. 5.

Table I also summarizes the fundamental arithmetic complexity of a single search operation, as well as complexity of the sorting algorithm applied to find M smallest Euclidean distances (i.e., M most reliable candidates). The fixed square multiplications require $2\sqrt{2^{M_C}}$ operations in the candidate search process. The number of x^2 multiplications is scaled five times because it is typically five times more complex than an addition operation for the same precision of input operands [25]. The number of comparisons is downscaled by a factor of two since the authors assume complexity of a comparison operation being about 50% simpler than that of an addition operation for the same arithmetic precision.

Fig. 8 illustrates a total number of arithmetic operations as a function of parameter M associated with both the QRD-M-based candidate-search and the QRD-QLD-based parallel candidate-search in the case of 4 × 4 16-QAM. It can be observed from Fig. 8 that the QRD-QLD-based parallel candidate-search algorithm is about half as computationally complex as the QRD-M candidate-search, except for smaller values of parameter M. As the value of M increases, the proposed QRD-QLD-based parallel candidate-search requires significantly less operations compared to the QRD-M candidate-search. Fig. 9 illustrates a total number of arithmetic operations associated with the QRD-M detection and the QRD-QLD-based parallel detection as a function of parameter M for 4 × 4 64-QAM. Similarly as for the case of 4 × 4 16-QAM, it can be observed from Fig. 9 that the QRD-QLD-based parallel candidate-search is substantially less complex than the QRD-M-based candidate-search, except for smaller values of the parameter M. Arithmetic complexity of the QRD-QLD and QRD-M given in Fig. 8 and Fig. 9 is obtained based on a number of fundamental arithmetic operations (add and compare operations) required for a single candidate search operation and based on complexity of sorting provided in Table I. It should be also noted that complexity of a search operation in the QRD-QLD is smaller than complexity of a search operation in the QRD-M, since a depth of candidate searching in the QRD-M is larger than in the QRD-QLD where searching is split into two groups of antennas.

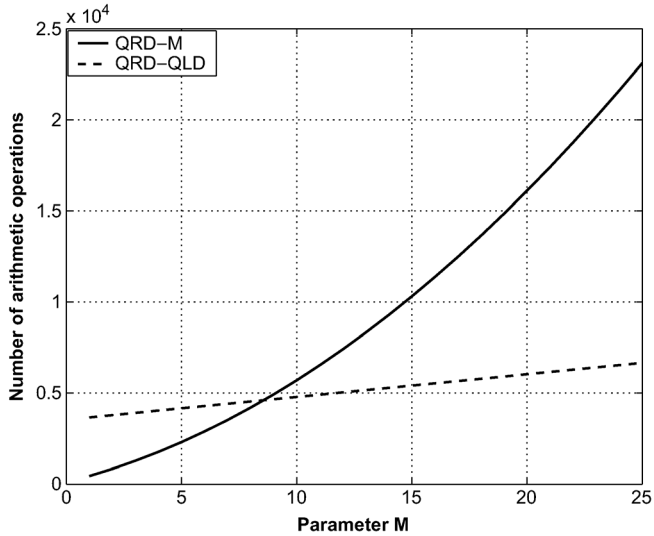


Fig. 8. Number of operations in the candidate search process for the QRD-M detector versus the QRD-QLD-PD as a function of parameter M, 4 × 4 16-QAM Rayleigh fading channels, $Cand = 125$.

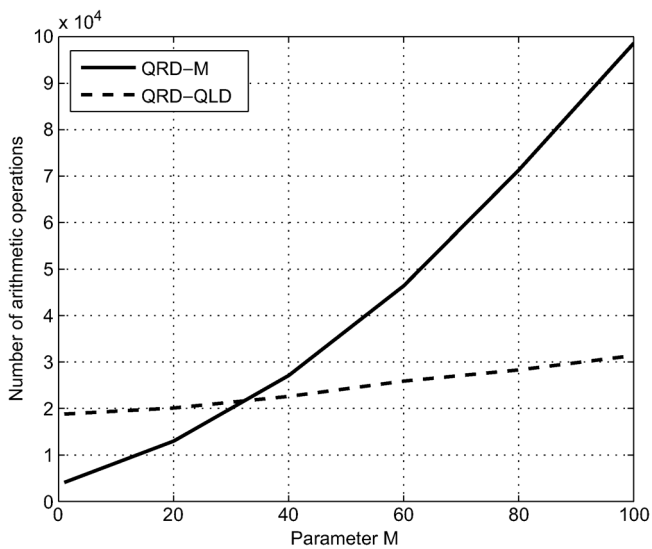


Fig. 9. Number of operations in the candidate search process for the QRD-M detector versus the QRD-QLD-PD as a function of parameter M, 4 × 4 64-QAM Rayleigh fading channels, $Cand = 500$.

B. Latency Analysis

Latency of the QRD-M candidate-search algorithm denoted by T_{QRDM} is given by (20). The scaling factor of one half is applied because two search units operate in parallel. As aforementioned, the QRD-QLD-based parallel detector (QRD-QLD-PD) utilizes two parallel and independent candidate search processes

for two groups of transmit antennas, and therefore two candidate-search units are required to exploit parallelism of the QRD-QLD-PD.

$$T_{\mathrm{QRDM}} = \frac{1}{2}\left[1+(N_t-1)M\right]T_{\mathrm{search-op}} + M \cdot M_C \cdot T_{\mathrm{compare-op}} + M(N_t-1)\left\lceil \log_2(M\cdot 2^{M_C})\right\rceil T_{\mathrm{compare-op}} \quad (20)$$

where $T_{\mathrm{compare-op}}$ is the latency of one comparison operation (approximately a half of clock cycle, as being approximately twice faster than a subtraction operation that typically requires one clock cycle), and $T_{\mathrm{search-op}}$ is the latency of one search operation (approximately two clock cycles, as the search operation comprises a square operation, an addition and a comparison operation, as given by (17)). The latency of candidate search of the QRD-QLD-PD including sorting of all candidates can be approximated as

$$T_{\mathrm{QRD-QLD}} = (1+2^{M_C})T_{\mathrm{search-op}} + M\left\lceil \log_2(Cand)\right\rceil T_{\mathrm{compare-op}} \quad (21)$$

where $Cand$ is a maximum number of partial candidates maintained after last search levels of two parallel candidate search processes (i.e., after $\frac{N_t}{2}$th search levels for N_t transmit antennas).

In the QRD-M candidate-search, sorting of candidates is applied after every search level. After the first search level, the M best candidates out of $P_C = 2^{M_C} = |S|$ candidates are maintained and used later in the next search level. For each of the remaining $N_t - 1$ search levels, the M best out of $M \cdot 2^{M_C}$ partial transmission candidates are saved. On the other hand, in the QRD-QLD-PD, two simultaneous candidate-sorting algorithms are applied at the end of both partial candidate search processes. The best M partial candidates can be determined out of $Cand$ candidates. Instead of sorting the entire list of candidates, only M transmission candidates with the smallest Euclidean distances are found.

The average latency of the conventional quick-sort algorithm for a list of N unsorted elements is $N \log_2 N$ clock cycles [26]. The sorting latency at every stage of the QRD-M candidate-search algorithm can be reduced to $M \log_2 N$ clock cycles since it is sufficient to find only the M smallest Euclidean distances. In this case, full processing parallelism is considered assuming enough comparators for finding the smallest element out of N elements in up to $\lceil \log_2 N \rceil$ stages of comparisons.

Table II shows the candidate-search latency of the QRD-QLD-PD versus theQRD-M detector for the case of four transmit/receive antennas and 16-QAM. For both detection schemes, hardware configurations with identical sorting units are considered having same numbers of comparators (i.e., 31, 63, or 127 comparators for both schemes, as illustrated in Table II). For each hardware configuration with a different number of comparators, a candidate-search latency of the QRD-M detector given in Table II is computed according to (20), while a candidate-search latency of the QRD-QLD-PD is computed according to (21). The numbers of final vector-candidates M in Table II selected for the QRD-M detector and for the QRD-QLD-PD provide approximately same error-rate performance for both detection schemes. It can be observed that the candidate-search latency of the QRD-QLD-PD is approximately twice as small as that of the QRD-M detector, for similar error-rate performance.

TABLE II
CANDIDATE-SEARCH LATENCY FOR THE QRD-M DETECTOR WITH $M = 16$ AND THE QRD-QLD-PD WITH $M = 18$ AS A FUNCTION OF THE NUMBER OF COMPARATORS FOR SORTING OF CANDIDATES, $Cand = 125$ IN THE QRD-QLD-PD, 4 × 4 16-QAM

Algorithm and number of comparators	Search Latency [clks]
QRD-M (M=16, $\frac{1}{2}M \cdot 2^{M_C}$-1=127 comp.)	178
QRD/QLD (M=18, 127 comp.)	77
QRD-M (M=16, 63 comp.)	194
QRD/QLD (M=18, 63 comp.)	83
QRD-M (M=16, 31 comp.)	242
QRD/QLD (M=18, 31 comp.)	89

In the QRD-QLD-PD, the best M partial vector-candidates for two groups of transmit antennas are combined together forming a list of M^2 final vector-candidates. The latency for calculating the soft information for the outer decoder is obviously larger than in the case of the QRD-M detector where only M final vector-candidates are available. Latency in calculating the *a posteriori* probabilities (APPs) of coded bits for a single MIMO channel realization using one final vector-candidate is approximately one clock cycle [27]. Therefore, if P parallel APP function units are utilized and if there are C final vector-candidates, the latency of computing final APPs is approximately $\frac{C}{P}$ clock cycles. For the QRD-M detector with parameter $M = 16$ and with $P = 4$ parallel APP function units, the latency for calculating APPs is approximately 4 clock cycles. In the case of the QRD-QLD-PD with $M = 18$ (similar error-rate performance as QRD-M detector with $M = 16$), the latency for calculating APPs is approximately 81 clock cycles.

A total detection latency of the QRD-QLD-PD is compared with a total detection latency of the QRD-M detector. The total detection latency is equal to a summation of candidate-search latency [see (21) and (20) for the QRD-QLD and the QRD-M, respectively] and the latency of calculating the soft information for the outer decoder. Two parallel search units are considered, four APP units, and 63 comparators for sorting of candidates. Current practical solutions for downlink OFDM receivers, as well as those proposed for emerging wireless standards, typically assume one iteration between inner detector and outer decoder. For the case of four transmit/receive antennas and 16-QAM, total latency of the QRD-QLD-PD is smaller than that of the QRD-M detector if

$$\left(83+\frac{(M_{\mathrm{QRD-QLD}})^2}{4}\right)\mathrm{clks} < \left(194+\frac{M}{4}\right)\mathrm{clks} \quad (22)$$

where $M_{\mathrm{QRD-QLD}}$ specifies a number of final candidates in the QRD-QLD-PD, and the latency of the candidate-search for 63 comparators is available in Table II. If the QRD-M detector with $M = 16$ is considered as a reference, then the inequality (22) is satisfied for $M_{\mathrm{QRD-QLD}} < 21.4$. It should be noted that in computing the total detection latency, one detection is performed for one channel realization including computation of reliability messages for transmitted coded bits.

For the latency analysis in the case of 4 × 4 64-QAM, the QRD-M detector with parameter $M = 67$ and the QRD-QLD-PD with parameters $M = 78$ and $Cand = 500$ are considered providing similar error rate performance, as shown in Fig. 11 in Section V-C. For the QRD-QLD-PD, there are two parallel search units and two parallel sorting units each with 63 comparators, as being also considered for the 4 × 4 16-QAM. The QRD-M detector utilizes one search unit configured to search candidates from all four transmit antennas and a sorting unit with 63 comparators (the same sorting unit as being used in the QRD-QLD-PD). In addition, both the QRD-QLD-PD and the QRD-M detector employ eight parallel APP function units for computation of APPs based on found symbol candidates. According to (20), a candidate search latency of the QRD-M detector with $M = 67$ is approximately 1710 clock cycles. According to (21), a candidate search latency of the QRD-QLD-PD with $M = 78$ and $Cand = 500$ is approximately 481 clock cycles. A total detection latency of the QRD-M detector is based on the candidate search latency of approximately 1710 clock cycles and the latency of calculating APPs with eight parallel APP units (i.e., the latency of $\frac{M}{8}$ clock cycles), which is approximately 1718 clock cycles. A total detection latency of the QRD-QLD-PD is based on the candidate search latency of approximately 481 clock cycles and the latency of calculating APPs with eight parallel APP units (i.e., the latency of $\frac{M^2}{8}$ clock cycles), which is approximately 1242 clock cycles. Therefore, in the case of 4 × 4 64-QAM, the total detection latency of the QRD-QLD-PD is substantially smaller than that of the QRD-M detector for the similar error rate performance.

C. Frame Error Rate Performance

As shown in Fig. 10 for the case of 4 × 4 16-QAM, performance of the QRD-QLD-PD with $M = 18$ is very similar to that of the QRD-M detector with $M = 16$, while the computational complexity is twice as small. The identical outer LDPC decoding is applied for both inner detectors with the codeword size of 1944 bits and the code rate of 1/2. Fifteen inner iterations are employed in the LDPC layered belief-propagation algorithm.

Fig. 11 shows frame error rate performance of the QRD-QLD-PD with $M = 78$ and of the QRD-M-based detector with $M = 67$ for 4 × 4 64-QAM, while the same LDPC decoder as in the aforementioned case of 4 × 4 16-QAM is considered and Rayleigh fading channels between transmit and receive antennas. From Fig. 11, similar error rate performance can be observed, while the computational complexity of QRD-QLD detection is substantially smaller than that of the QRD-M detection, as being illustrated in Fig. 9 for specific values of the parameter M utilized in Fig. 11.

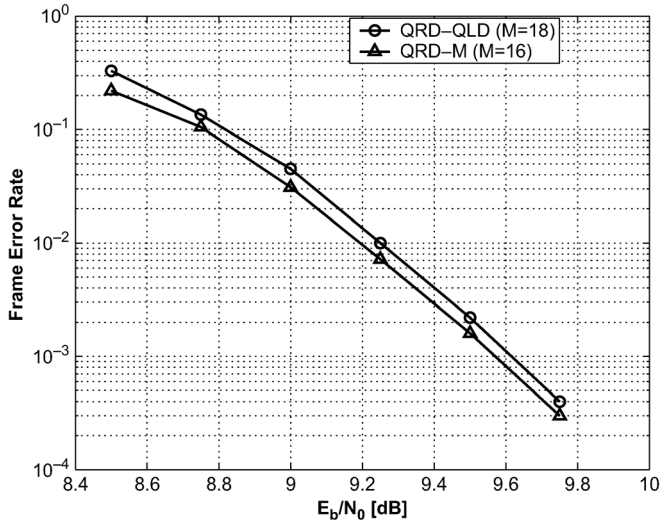


Fig. 10. Frame error rate performance for the QRD-QLD detector with $M = 18$ versus the QRD-M detector with $M = 16$ in Rayleigh fading channels, 4 × 4 16-QAM; outer LDPC, code size of 1944 bits and code rate of 1/2.

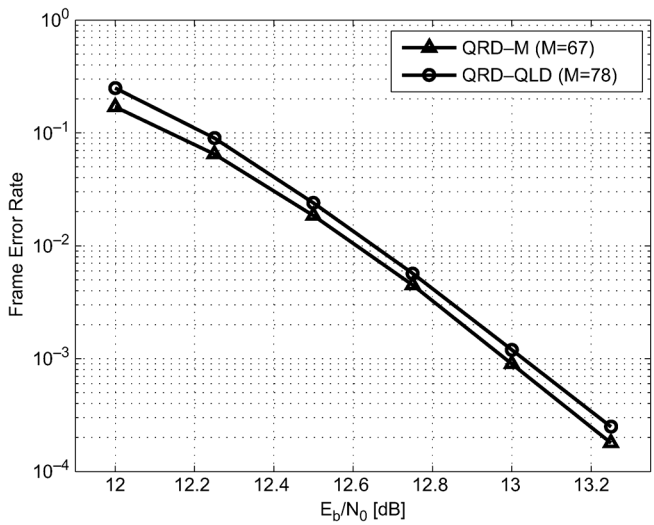


Fig. 11. Frame error rate performance for the QRD-QLD detector with $M = 78$ versus the QRD-M detector with $M = 67$ in Rayleigh fading channels, 4 × 4 64-QAM; outer LDPC, code size of 1944 bits and code rate of 1/2.

VI. FPGA and ASIC Implementation

The hardware implementation of the QRD-QLD-PD (as well as of the QRD-M-based detector) is performed in two steps. First, for fast prototype and comparison of the QRD-QLD-PD versus QRD-M-based detector, these two detectors are implemented using Xilinx Field Programmable Gate Array (FPGA). Second, ASIC design flow including synthesis, placement and routing is carried out using Taiwan Semiconductor Manufacturing Company (TSMC) 65 nm process technology [28] and Synopsys low-power design flow [29]. The ASIC synthesis results for the QRD-QLD-PD are also compared with that of other sphere detectors reported in the literature. The implemented QRD-QLD-PD and QRD-M-based detector support 4 × 4 16-QAM wireless communications system, while the parameter $M = 18$ for the QRD-QLD-PD and $M = 16$ for the QRD-M, as being used in Fig. 10 illustrating similar error-rate performance.

For FPGA prototyping, fixed-point C codes emulating the QRD-QLD-PD and QRD-M-based detector are synthesized by AutoESL [30] targeting Virtex-5 xc5vfx100tff1738-2 FPGA [31], and corresponding register transfer level (RTL) codes are generated. Using Xilinx ISE [32], the RTL codes are mapped

TABLE III
FPGA RESOURCE UTILIZATION FOR THE QRD-QLD-PD AND QRD-M: VIRTEX-5 XC5VFX100TFF1738-2

	QRD-QLD-PD	QRD-M
Slices	307 (2%)	920 (6%)
BRAMs	10 (2%)	15 (3%)
DSP48E	38 (14%)	104 (41%)
FFs	558 (1%)	1837 (3%)
LUTs	699 (1%)	3065 (5%)
Max. Freq. [MHz]	140.92	135.94

TABLE IV
ASIC DESIGN PARAMETERS OF QRD-QLD-PD AND QRD-M IN 65 nm TSMC TECHNOLOGY

	Logic Area (mm^2)	GEs	Core Area (mm^2)	Max. Freq. (MHz)	Power (mW)
QRD-QLD	0.052	36.1K	0.64	358	57
QRD-M	0.084	58.4K	0.79	333	93
SESD [34]	1.9 [1]	56.8K	Not Reported	273 [3]	Not Reported
MKSE [6]	0.56 [2]	97K	Not Reported	400 [4]	Not Reported
FSD [35]	0.18 [2]	29.8K	Not Reported	800 [4]	Not Reported
FSD [36]	0.17 [2]	27.5K	Not Reported	800 [4]	Not Reported

[1] 0.25 μm CMOS technology.
[2] 0.13 μm CMOS technology.
[3] Scaled from 0.25 μm to 65 nm by multiplying by 0.25/0.065.
[4] Scaled from 0.13 μm to 65 nm by multiplying by 0.13/0.065.

to bit streams, and FPGA timing information and resource utilization are then measured. Table III provides FPGA resource utilization and maximum achievable clock frequency for the QRD-QLD-PD and the QRD-M-based detector. The resource utilization is illustrated in terms of number of FPGA slices, 18 Kb block RAMs (BRAMs), DSP48E slices (optimized for multiply-accumulate operations), flip-flops (FFs), and 4-input look-up tables (LUTs). The utilization percentage for each FPGA resource is also given in Table III. It can be noticed from Table III that the QRD-QLD-PD has smaller FPGA implementation complexity than the QRD-M-based detector for similar error-rate performance.

For the ASIC implementation, the fixed-point C codes emulating the QRD-QLD-PD and QRD-M-based detector are synthesized by the AutoESL [30] to generate the corresponding RTL codes. Then, the Synopsys Design Compiler tool [29] is employed to get netlists from the RTL codes, while the TSMC 65 nm technology library is used. Finally, the Cadence system-on-chip (SoC) encounter [33] is utilized to place and route the netlists, and to measure timing and core area information.

Table IV provides ASIC synthesis results for the QRD-QLD-PD and the QRD-M-based detector (with parameter M of 18 and 16, respectively). These design results are also compared with that of soft-output Schnorr-Euchner sphere decoder (SESD) where every candidate node in the search-tree is visited at most once [34], with that of the soft-output modified K-best Schnorr-Euchner (MKSE) sphere detector [6], and with that of the fixed-complexity sphere decoders (FSDs) [35], [36]. The ASIC synthesis results in Table IV are presented in terms of logic cell area before place and route, number of gate equivalents (GEs) for technology-independent area characterization (one GE corresponds to the area of a two-input drive-one NAND gate), core area after place and route, maximum achievable clock frequency (obtained from a post-layout static timing analysis), and power dissipated at the maximum clock frequency (dynamic power and cell leakage power). The logic cell area is converted into the number of GEs based on the area of one logic gate being equal to $1.44 \cdot 10^{-6}$ mm^2 for the TSMC 65 nm technology [37]. Table IV shows that the proposed QRD-QLD-PD has smaller logic area size (represented as the number of GEs) compared to the QRD-M-based detector, the SESD [34], and the MKSE detector [6]. Further, the logic area of the QRD-QLD-PD is comparable with that of the FSDs [35], [36]. Fig. 12 illustrates the layout of QRD-QLD-PD after place and route, with the area of $0.64\mathrm{mm}^2$ given in Table IV.

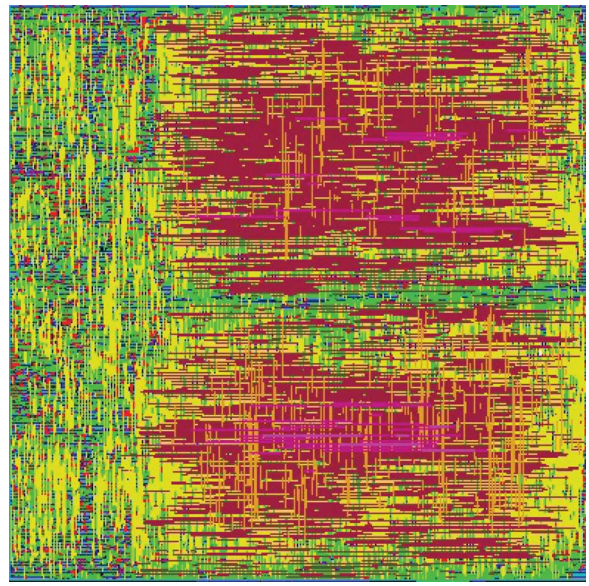


Fig. 12. ASIC layout of the QRD-QLD-PD after place and route.

VII. CONCLUSION

We present implementation of a partial candidate search algorithm for soft sphere detector based on QR and QL decompositions of a channel matrix. Two independent and parallel search algorithms are applied for two separate groups of transmit antennas. A final list of candidates is obtained after combining two groups of independent partial candidates. The QRD-QLD-PD is compared in terms of computational complexity, detection latency and error-rate performance with the QRD-M detector proposed for several emerging wireless technologies. It is shown in this work that a search latency of the QRD-QLD-PD is smaller than that of the QRD-M detector for same error-rate performance. Further, implementation complexity (i.e., a number of arithmetic operations) associated with the QRD-QLD-PD is shown to be substantially smaller than that of the QRD-M detector for the same error rate performance. The FPGA and ASIC synthesis results for the QRD-QLD-PD also show smaller design complexity than that of the QRD-M detector. Finally, ASIC design results of the QRD-QLD-PD

are compared with that of other sphere decoders reported in the literature. It is shown that the QRD-QLD-PD has comparative advantages over related solutions from the literature, or comparable design results.

REFERENCES

[1] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. J.*, vol. 1, pp. 41–59, Oct. 1996.

[2] B. Lu, G. Yue, and X. Wang, "Performance analysis and design optimization of LDPC-coded MIMO OFDM systems," *IEEE Trans. Signal Process.*, vol. 52, pp. 348–361, Feb. 2004.

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

[4] D. Garrett, L. Davis, S. ten Brink, B. Hochwald, and G. Knagge, "Silicon complexity for maximum likelihood MIMO detection using spherical decoding," *IEEE J. Solid-State Circuits*, vol. 39, pp. 1544–1552, Sep. 2004.

[5] K. J. Kim, J. Yue, R. A. Iltis, and J. D. Gibson, "A QRD-M/Kalman filter-based detection and channel estimation algorithm for MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 710–721, Mar. 2005.

[6] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE J. Sel. Areas Commun.*, vol. 24, pp. 491–503, Mar. 2006.

[7] M. Myllyla, J. R. Cavallaro, and M. Juntti, "Architecture design and implementation of the metric first list sphere detector algorithm," *IEEE Trans. VLSI Syst.*, vol. 19, no. 5, pp. 895–899, May 2011.

[8] K. J. Kim, T. Reid, and R. A. Iltis, "Iterative soft-QRD-M for turbo coded MIMO-OFDM systems," *IEEE Trans. Commun.*, vol. 56, no. 7, pp. 1043–1046, Jul. 2008.

[9] J. Cha, J. Ha, and J. Kang, "Low-complexity iterative QRD-M detection algorithm for V-BLAST systems," *Electron. Lett.*, vol. 43, pp. 1374–1376, Nov. 2007.

[10] M. Baek, Y. You, and H. Song, "Combined QRD-M and DFE detection technique for simple and efficient signal detection in MIMO-OFDM system," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 1632–1638, April 2009.

[11] C. Ahn, "Parallel detection algorithm using multiple QR decompositions with permuted channel matrix for SDM/OFDM," *IEEE Trans. Veh. Technol.*, vol. 57, pp. 2578–2582, Jul. 2008.

[12] H. Kawai, K. Higuchi, N. Maeda, and M. Sawahashi, "Adaptive control of surviving symbol replica candidates in QRM-MLD for OFDM multiplexing," *IEEE J. Sel. Areas Commun.*, vol. 24, pp. 1130–1140, May 2006.

[13] T. Detert, "An efficient fixed complexity QRD-M algorithm for MIMO-OFDM using per-survivor slicing," in *Proc. IEEE Int. Symp. Wireless Commun. Systems*, Trondheim, Norway, Oct. 2007, pp. 572–576.

[14] L. G. Barbero and J. S. Thompson, "Performance analysis of a fixed-complexity sphere decoder in high-dimensional MIMO systems," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Toulouse, France, May 2006, pp. IV–IV.

[15] P. Radosavljevic and J. R. Cavallaro, "Soft sphere detection with bounded search for high-throughput MIMO receivers," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Oct. 2006, pp. 1175–1179.

[16] J. Ketonen, M. Juntti, and J. R. Cavallaro, "Performance-complexity comparison of receivers for a LTE MIMO-OFDM system," *IEEE Trans. Signal Process.*, vol. 58, pp. 3360–3372, June 2010.

[17] K. Amiri and J. R. Cavallaro, "Partial detection for multiple antenna cooperation," in *Proc. IEEE 43rd Ann. Conf. on Inf. Sci. Syst.*, Baltimore, MD, Apr. 2009, pp. 669–674.

[18] P. Radosavljevic, K. J. Kim, and J. R. Cavallaro, "QRD-QLD searching based sphere detection for emerging MIMO downlink OFDM receivers," in *Proc. IEEE Global Commun. Conf.*, New Orleans, LA, Dec. 2008, pp. 1–5.

[19] S. Chen and T. Zhang, "Low power soft-output signal detector design for wireless MIMO communication systems," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 2007, pp. 232–237.

[20] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Math. Comput.*, vol. 44, pp. 463–471, Apr. 1985.

[21] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. Int. Conf. Commun.*, Seattle, WA, Jun. 1995, pp. 1009–1013.

[22] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory*, vol. 42, pp. 429–445, Mar. 1996.

[23] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *IEEE Trans. Wireless Commun.*, vol. 3, pp. 2299–2311, Nov. 2004.

[24] Y. Dai, S. Sun, and Z. Lei, "A comparative study of QRD-M detection and sphere decoding for MIMO-OFDM systems," in *Proc. IEEE Int. Symp. Pers., Indoor and Mobile Radio Commun. Syst.*, Berlin, Germany, Sep. 2005, pp. 186–190.

[25] M. M. Mansour and N. R. Shanbhag, "High-throughput LDPC decoders," *IEEE Trans. VLSI Syst.*, vol. 11, pp. 976–996, Dec. 2003.

[26] B. C. Dean, "A simple expected running time analysis for randomized divide and conquer algorithms," *Discr. Appl. Math.*, vol. 154, no. 1, pp. 1–5, Jan. 2006.

[27] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bolcskei, "VLSI implementation of MIMO detection using the sphere decoding algorithm," *IEEE J. Solid-State Circuits*, vol. 40, pp. 1566–1577, Jul. 2005.

[28] Taiwan Semiconductor Manufacturing Company (TSMC) Limited, 65 nm TSMC Process Technology [Online]. Available: http://tsmc.com/english/dedicatedFoundry/technology/65nm.htm

[29] *"Synopsys Design Compiler User Guide,"* ver. D-2010.03-SP2 [Online]. Available: http://synopsys.com/Tools/Implementation/RTLSynthesis/DCGraphical/Pages/default.aspx

[30] *"An Independent Evaluation of the AutoESL AutoPilot High-Level Synthesis Tool,"* Berkeley Design Technol. Inc., 2010 [Online]. Available: http://bdti.com/MyBDTI/pubs/AutoPilot.pdf

[31] *"Virtex-5 FPGA User Guide, UG190,"* ver. 5.3, Xilinx, May 2010 [Online]. Available: http://xilinx.com/support/documentation/user_guides/ug190.pdf

[32] *"Xilinx Integrated Synthesis Environment (ISE) Design Suite 13.2,"* Jul. 2011 [Online]. Available: http://xilinx.com/support/documentation/dt_ise13-2_userguides.htm

[33] E. Brunvand, *Digital VLSI Chip Design With Cadence and Synopsys CAD Tools*, 1st ed. Reading, MA: Addison-Wesley, 2009.

[34] C. Studer, A. Burg, and H. Bolcskei, "Soft-output sphere decoding: Algorithms and VLSI implementation," *IEEE J. Sel. Areas Commun.*, vol. 26, pp. 290–300, Feb. 2008.

[35] B. Wu and A. Masera, "A novel VLSI architecture of fixed-complexity sphere decoder," in *Proc. Euromicro Conf. on Digit. Syst. Design*, Lille, France, Sep. 2010, pp. 737–744.

[36] B. Wu and A. Masera, "An eight-nodes-per-cycle fixed-complexity sphere decoder in complex signal model," in *Proc. Int. Conf. on Future Inf. Technol.*, Singapore, Feb. 2011, pp. 62–67.

[37] Synopsys Insight: Predictable Success From Concept to Chip [Online]. Available: http://synopsys.com/news/pubs/insight/2008/art2_dftimplem_v3s4.html?cmp=NLC-insight&Link=Dec08_V3I4_Art2

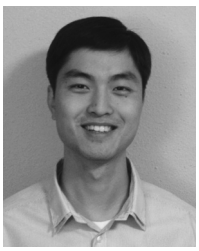
Predrag Radosavljevic (S'03–M'08) received the B.S. degree from the University of Belgrade, Belgrade, Serbia, in 2000, and the M.S. and Ph.D. degrees from Rice University, Houston, TX, in 2004 and 2008, respectively, all in electrical engineering.

From 2007 to 2008, he was with CGGVeritas, Houston. In 2008, he joined Patterson & Sheridan LLP, Houston, where he is currently a Senior Technical Advisor in relation to intellectual property law practicing preparation and prosecution of patent applications with emphasis on wireless communications, electronics, and signal processing. His research interests include signal processing for wireless communications, and DSP and VLSI architectures for applications in wireless communications.

Kyeong Jin Kim (SM'11) received the B.S. degree from the Korea Advanced Institute of Science and Technology (KAIST) in 1991, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California, Santa Barbara, in 2000.

During 1991–1995, he was a Research Engineer at the Video Research Center of Daewoo Electronics, Ltd., Korea. In 1997, he joined the Data Transmission and Networking Laboratory, University of California, Santa Barbara. After receiving his degrees, he joined the Nokia Research Center, Dallas, TX, as a Senior Research Engineer, where he was, from 2005 to 2009, an L1 specialist. His research has been focused on the transceiver design, resource management, and scheduling in the cooperative wireless communication systems. During 2010–2011, he was an invited professor at Inha University, Korea.

Dr. Kim currently serves as an Associate Editor for the IEEE COMMUNICATIONS LETTERS.



Hao Shen (S'10) received the B.S. degree in microelectronics from Shanghai Jiao Tong University, Shanghai, China, in 2009, and the M.Elect.Eng. degree from Rice University, Houston, TX, in 2011.

He is currently pursuing the Ph.D. degree at the Department of Electrical and Computer Engineering, Rice University, Houston. His research interests include parallel algorithms and VLSI architectures for wireless communication systems, and high level C-to-RTL synthesis for high speed DSP algorithms.

Joseph. R. Cavallaro (S'78–M'82–SM'05) received the B.S. degree from the University of Pennsylvania, Philadelphia, in 1981, the M.S. degree from Princeton University, Princeton, NJ, in 1982, and the Ph.D. degree from Cornell University, Ithaca, NY, in 1988, all in electrical engineering.

From 1981 to 1983, he was with AT&T Bell Laboratories, Holmdel, NJ. In 1988, he joined the faculty of Rice University, Houston, TX, where he is currently a Professor of electrical and computer engineering. His research interests include computer arithmetic, VLSI design and microlithography, and DSP and VLSI architectures for applications in wireless communications. During the 1996–1997 academic year, he served at the National Science Foundation as Director of the Prototyping Tools and Methodology Program. He was a Nokia Foundation Fellow and a Visiting Professor at the University of Oulu, Finland, in 2005 and continues his affiliation there as an Adjunct Professor. He is currently the Director of the Center for Multimedia Communication at Rice University.

Dr. Cavallaro was Co-Chair of the 2004 Signal Processing for Communications Symposium, IEEE Global Communications Conference, and General/Program Co-Chair of the 2003, 2004, and 2011 IEEE International Conference on Application-Specific Systems, Architectures and Processors (ASAP), and Program Co-Chair for the 2012 ACM/IEEE GLSVLSI.