Designing Low-Complexity Equalizers for Wireless Systems

Wei Zhang, Xiaoli Ma, Brian Gestner, and David V. Anderson, Georgia Institute of Technology

ABSTRACT

The demand on wireless communications to provide high data rates, high mobility, and high quality of service poses more challenges for designers. To contend with deleterious channel fading effects, both the transmitter and the receiver must be designed appropriately to exploit the diversity embedded in the channels. From the perspective of receiver design, the ultimate goal is to achieve both low complexity and high performance. In this article, we first summarize the complexity and performance of low-complexity receivers, including linear equalizers and decision feedback equalizers, and then we reveal the fundamental condition when LEs and DFEs collect the same diversity as the maximum-likelihood equalizer. Recently, lattice reduction techniques were introduced to enhance the performance of low-complexity equalizers without increasing the complexity significantly. Thus, we also provide a comprehensive review of LR-aided low-complexity equalizers and analyze their performance. Furthermore, we describe the architecture and initial results of a very-large-scale-integration implementation of an LR algorithm.

INTRODUCTION

Wireless communications have become the fastest growing industry and are ubiquitous in almost all areas of our daily life, encompassing radio and television broadcasting, mobile phones, and satellite communications. The increasing demand for wireless services for voice, multimedia, and data transmissions results in a continually expanding market. Clearly, the development of solid-state technology and digital-signal-processing (DSP) devices contributes significantly to this growth because such technology makes low-cost and feature-rich communication devices feasible. More importantly, however, the globalization of wireless transmission standards accelerates the spread of wireless services. For example, driven by widespread acceptance of the IEEE 802.11a/b/g standards, wireless local area networking for computers and other devices is spreading rapidly.

Naturally, as wireless services spread and become integrated into the lives of more people, the expectation of the performance and reliability of wireless devices increases. The evolution of standards and systems is driven by the demand for better quality of service, higher data rates, and higher mobility. As a result, system designers now face more challenges such as limited bandwidth, resource allocation, and particularly, channel-fading effects introduced by variability in the time, frequency, and space domains.

Most wireless transmissions, such as the orthogonal frequency division multiplexing (OFDM) systems of IEEE 802.11a, the multiantenna multi-input multi-output (MIMO) systems of IEEE 802.11n, and the multi-user code division multiple access (CDMA) systems, can be modeled as a linear-block transmission system. Given a linear-block transmission model assumption, maximum-likelihood equalizers (MLEs) or near-ML decoders were adopted at the receiver to collect diversity, which is an important metric for performance; however, these decoders exhibit high complexity. To reduce the decoding complexity, low-complexity equalizers, such as linear equalizers (LEs) and decision feedback equalizers (DFEs) often are adopted. These methods, however, may not utilize the diversity enabled by the transmitter and as a result, have degraded performance compared to MLEs [1]. In this article, we first provide a comprehensive review of low-complexity equalizers based on the linear system model. Then we reveal the fundamental condition when low-complexity equalizers collect the same diversity as that of the near-MLEs.

Lattice reduction (LR) techniques were introduced to improve the performance of low-complexity equalizers without increasing the complexity significantly [2–6]. It has been shown that LR-aided equalizers collect the same diversity as MLEs for vertical-Bell Laboratories layeredspace-time (V-BLAST) systems [3]. After studying low-complexity equalizers, we provide an overview of different LR algorithms and LR-aided equalizers. The performance improvement is analyzed in terms of diversity for two well-adopted LR algorithms, the complex Lenstra-Lenstra-Lovász (CLLL) algorithm (see [3] and references therein) and Seysen's algorithm (SA) [4]. Furthermore, by considering the nature of CLLL operations and exploiting the inherent parallelism in the algorithm, we arrive at a hardware architecture suitable for very-large-scale-integration (VLSI)

This material is based on work supported thorough collaborative participation in the Collaborative Technology Alliance for Communications & Networks sponsored by the U.S. Army Research Laboratory under Cooperative Agreement DAAD19-01-20011. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

Authorized licensed use limited to: Christos Maurokefalidis. Downloaded on December 28, 2009 at 04:02 from IEEE Xplore. Restrictions apply

implementation. We illustrate an efficient fieldprogrammable gate array (FPGA) implementation of the CLLL algorithm for 4×4 systems.

SYSTEM MODEL

Consider linear-block transmissions depicted in Fig. 1a:

$$y = Hs + w \tag{1}$$

where H is the $M \times N$ channel matrix, s is the $N \times 1$ symbol vector obtained by mapping and block encoding the information bit sequence, y is the $M \times 1$ received vector, and w is independent and identically distributed (i.i.d.), complex, additive white Gaussian noise with variance σ_w^2 . At the receiver, with channel state information and observation y, a detector can be adopted to obtain the estimate of the transmitted symbol vector \hat{s} . Note that the channel matrix H is general enough to represent a number of cases, for example, multi-antenna block transmissions, precoded OFDM systems, single-carrier Toeplitz channels, and multi-user channels [1].

Given the linear-block transmission model in (1), there are various ways to decode the transmitted symbol vector s from the observation y. Here, we generalize the term "equalizer" as the one to equalize the channel effect. Different equalizers lead to different system performance. The bit error rate (BER) describes the reliability of the transmission and therefore is a widely adopted figure of merit to characterize the performance of wireless systems. The BER performance of wireless transmissions over fading channels is usually quantified by two parameters: diversity order and coding gain. Diversity order is defined as the negative asymptotic slope of the BER versus the signal-to-noise ratio (SNR) curve plotted in log-log scale. It describes how fast the error probability decays with SNR. The coding gain further measures the SNR gap among different coding schemes that have the same diversity. The higher the diversity is, the smaller the error probability is when SNR is high. To enjoy the diversity from fading channels, we must design both the transmitter and receiver appropriately. In this article, we focus on the design of the receiver.

An often used and also optimal detector (if there is no prior information about the symbols and/or symbols are treated as deterministic parameters) is the maximum-likelihood equalizer (MLE), which is based on an exhaustive search among all $N \times 1$ symbol vectors. The MLE provides optimal error performance with highdecoding complexity ($\mathcal{O}(|\varsigma|^N)$). Some near-ML equalizers also were proposed to reduce the complexity and achieve near-ML performance. For example, the sphere-decoding (SD) method [7] formulates a tree search and reduces the average complexity to polynomial when N is small and the SNR is high, but the variance of the complexity may remain high. The complexity of near-MLEs is especially high when the size of the channel matrix and/or the constellation size is large. Furthermore, early termination and fixed memory considerations for hardware implementations may degrade the performance of near-MLEs.



Figure 1. Block diagram of a) linear transmission system model; b) linear equalizers; c) decision feedback equalizers.

LOW-COMPLEXITY EQUALIZERS

In addition to near-MLEs, there are other equalizers that usually are characterized and referred to as low-complexity equalizers: the previously mentioned LEs and DFEs. LEs, as depicted in Fig. 1b, are in the form $\hat{s} = Q(Gy)$, where $Q(\cdot)$ corresponds to the Decision block and denotes quantization to the nearest constellation point for a given modulation scheme. Two LEs that often are adopted are the zero-forcing (ZF) equalizer, where G is the Moore-Penrose pseudo-inverse of the channel matrix, and the linear minimum mean-square error (MMSE) equalizer, where G is constructed to minimize the noise effect [3, Eq. (6)]. The ZF equalizer aims to cancel the channel effect by assuming a noiseless environment, whereas the MMSE equalizer further takes into account the noise effect. Thus, the MMSE equalizer achieves better performance in general, but requires an estimate of the noise variance at the receiver. The complexities of both equalizers are dominated by matrix inversion, which requires polynomial complexity $O(N^3)$ through Gaussian elimination. Furthermore, the MMSE equalizer can be expressed in the same form as the ZF equalizer, based on an extended system model as in [3, 5].

The DFEs, also referred to as successive interference cancellation (SIC) equalizers, are depicted in Fig. 1c. The major difference between DFEs and LEs is the feedback of the detected symbols through a feedback matrix **B**. According to the equalization method, DFEs are divided into two categories: ZF-DFE (ZF-SIC) and MMSE-DFE (MMSE-SIC). The specific designs of the feedforward matrix F and the feedback matrix **B** for both DFEs can be found in [8]. Different from LEs, matrix decompositions (e.g., QR-decomposition) constitute the major part of the complexity of DFEs. Algorithms such as these usually are associated with the complexity of $O(MN^2)$. Compared to LEs, the corresponding DFEs achieve better performance. However, the performance of DFEs is greatly affected by the decoding order and the error propagation. To improve the performance of DFEs and to mitigate the complexity over-

The complexity of the ZF equalizer is based on the Gaussian elimination of square matrices, whereas the complexity of SIC equalizers is obtained from the QR-decomposition approach. Furthermore, the complexity of the MMSE equalizer does not include the procedure of estimating the noise variance.

		M = N = 4		<i>M</i> = <i>N</i> = 6		<i>M</i> = <i>N</i> = 8	
Target BER		10 ⁻³	10-4	10 ⁻³	10 ⁻⁴	10 ⁻³	10 ⁻⁴
SD	SNR	8.25 dB	11.05 dB	5.05 dB	7.35 dB	3.05 dB	5.0 dB
	Complexity	3662	3550	16019	14090	55957	54150
ZF	SNR	27.1 dB	37.05 dB	27.05 dB	37.35 dB	27.00 dB	37.05 dB
	Complexity	298		869		1892	
MMSE	SNR	22.4 dB	32.45 dB	20.1 dB	30.25 dB	18.20 dB	28.25 dB
	Complexity	812		2618		6040	
ZF-SIC	SNR	23.75 dB	33.15 dB	23.15 dB	33.60 dB	22.50 dB	32.80 dB
	Complexity	748		2401		5546	
MMSE-SIC	SNR	19.75 dB	29.65 dB	16.85 dB	27.10 dB	14.60 dB	24.75 dB
	Complexity	1284		4333		10266	

Table 1. Comparison of different equalizers for i.i.d. complex channels with QPSK modulation.

head introduced by the feedback filter, optimum ordering is usually adopted in DFEs. For example, V-BLAST ordering optimizes the BER performance, but the complexity is sub-optimal [8].

To provide a complete comparison of lowcomplexity equalizers with near-MLEs, we first find the SNR of different equalizers to achieve the target BER by searching the SNR with step size 0.05 dB. The corresponding complexity of different equalizers is calculated in terms of average arithmetic operations (including real additions and real multiplications). The results are given in Table 1 for the quadrature phase shift keying (QPSK) constellation and i.i.d. complex channels for different sizes, where the SNR is defined as the symbol energy per transmit dimension versus noise power spectral density. The complexity of the ZF equalizer is based on the Gaussian elimination of square matrices, whereas the complexity of SIC equalizers is obtained from the QR-decomposition approach. Furthermore, the complexity of the MMSE (MMSE-SIC) equalizer does not include the procedure of estimating the noise variance. The SD method is implemented as in [7]. For a higher quadrature amplitude modulation (QAM) constellation, the complexity of SD increases dramatically, whereas the complexity of LEs and DFEs stays the same. From Table 1, it is obvious that the low-complexity equalizers require a higher SNR to achieve a certain BER although their complexity is quite low.

LOW COMPLEXITY OR HIGH DIVERSITY? OR BOTH?

The main drawback of the aforementioned lowcomplexity equalizers is that these equalizers usually cannot collect the same diversity as near-MLEs. For example, the diversity order collected by LEs and DFEs is only M - N + 1 for spatial multiplexing systems with i.i.d. channels, whereas near-MLE exploits diversity M [3]. The impact of the lack of diversity order becomes especially severe when the channel matrix is square, for example, M = N as shown in Table 1. Furthermore, as shown in [8], optimal ordering cannot increase the diversity order collected by DFEs but improves the performance only in terms of coding gain. Because near-MLEs exhibit either high average complexity or high complexity variance, the cubic order complexity results in LEs and DFEs being widely adopted in practical systems. A natural question is whether the complexity reduction is worth the performance sacrifice. Or in other words, is there a way to keep the complexity low while improving the performance in terms of coding gain or even diversity order?

Some interesting observations show that in some channel setups, LEs and DFEs collect the same diversity as that of near-MLEs. For example, for orthogonal space-time block code (OSTBC) and uncoded OFDM systems, low-complexity equalizers have the same diversity as that of near-MLEs [1], where both equivalent channel matrices satisfy the property that $H^{H}H$ is diagonal. This motivates us to determine the fundamental conditions when low-complexity equalizers exploit the same diversity as MLEs. As revealed in [1], the fundamental condition for LEs to collect the same diversity as near-MLEs is that the channels must be constrained within a certain distance from orthogonality, where an orthogonal matrix Hmeans $H^{H}H$ is a diagonal matrix. Or in other words, the channel matrices cannot be arbitrarily close to singularity. Thus, the quality of channels determines the diversity of low-complexity equalizers. Now, a question naturally arises: how do we quantify the distance of channels from orthogonality?

There are several metrics that have been adopted to measure the quality of a matrix and further judge whether the fundamental condition is met. These include the condition number, the orthogonality deficiency, and Seysen's metric. The well-known condition number is defined as the ratio between the maximal and minimal singular values of the matrix. The orthogonality deficiency (od(H)) [3, Eq. (17)] is defined as the ratio between the actual volume of the space spanned by all the columns of the matrix and the volume of space spanned by all the columns, assuming they are orthogonal. Seysen's metric (S(H)) [4, Eq. (3)] further balances the orthogonality between the matrix and the inverse matrix. In addition to these metrics, other metrics exist that help quantify the performance gap between a specific low-complexity equalizer and an MLE. For example, the proximity factor in [9] is a function not only of the channel matrix but also the specific low-complexity equalizer adopted. However, for most practical systems, no matter which metric is adopted, the quality of the channel matrix does not have a lower bound for the worst case, which means the channel matrices can be arbitrarily close to singular. For these transmissions, low-complexity equalizers usually exhibit inferior performance relative to MLEs due to the loss of diversity [1]. For low-complexity equalizers to achieve the same diversity as MLEs, the channel matrix must be modified such that the channel matrix distance from orthogonality is upper-bounded. One approach is to modify the receiver by adopting lattice reduction techniques, which restore the diversity of low-complexity equalizers by modifying channel matrices to meet the fundamental condition.

LATTICE REDUCTION AIDED EQUALIZERS

In the linear-block transmission model in (1), the received signal vector y is the noisy observation of the vector *Hs*, which is in the lattice spanned by the columns of *H* because all the entries of s can be transformed to complex integers by shifting and scaling. In general, a lattice has more than one set of basis vectors. Some bases exist that span the same lattice as H but are closer to orthogonality than H. The process of finding a basis closer to orthogonality is called lattice reduction. The ultimate goal of LR algorithms is to find a "better" channel matrix $\tilde{H} = HT$ where T is a unimodular matrix, which means that all the entries of T and T^{-1} are complex integers, and the determinant of T is ± 1 or $\pm j$. The restrictions on the matrix T ensure that the lattice generated by \tilde{H} is the same as that of *H*. Note that the equivalence of the two lattices spanned by H and \tilde{H} is based on the assumption that all the entries of s belong to the whole complex integer set.

With the new channel matrix \tilde{H} generated by the LR algorithm, the system model in (1) can be written as

$$y = HT (T^{-1}s) + w = \tilde{H}z + w.$$
⁽²⁾

Because all the entries of T^{-1} and the signal constellation belong to a Gaussian integer ring, the entries of z also are Gaussian integers. We first apply low-complexity equalizers onto the system in (2) to obtain \hat{z} , the estimate of z, by taking the constellation of z as the whole Gaussian integer ring. After obtaining \hat{z} , we recover s by mapping $T\hat{z}$ to the appropriate constellation. These two hard-decoding steps consist of the LR-aided lowcomplexity equalizers (LRAEs) for linear-block transmission systems. Details can be found in [2, 3]. Obviously, how good the new basis is depends on the specific LR algorithm and determines whether the diversity of low-complexity equalizers can be restored. Thus, in the following section, we briefly review the existing LR algorithms.

There are several

metrics that have

been adopted to

measure the quality

of a matrix and

further judge

whether the

fundamental

condition is met.

These include the

condition number,

the orthogonality

deficiency, and

Seysen's metric.

LATTICE REDUCTION ALGORITHMS

LR techniques have been studied by mathematicians for decades, and many LR algorithms have been proposed. Gaussian reduction, Minkowski reduction, and Korkine-Zolotareff (KZ) reduction algorithms find the optimal basis for a lattice based on the successive minimal criteria, but these algorithms are highly complex and therefore infeasible for communications systems (see [6] and references therein). The well-known Lenstra-Lenstra-Lovász (LLL) algorithm does not guarantee finding the optimal basis with minimal od, but it guarantees in polynomial time to find a basis within a factor to the optimal one [6]. Seysen's algorithm reduces Seysen's metric to perform LR [4]. A simplified Brun's algorithm is proposed and implemented in [10] to reduce complexity but also sacrifices performance. For the worst cases, these LR algorithms may not terminate, but simulations have shown that this never occurs in practice [6, p. 62; 11].

Given the array of LR algorithms in the literature, it is difficult to justify which one is better in terms of both performance and complexity. Therefore, it is hard to choose the appropriate one for VLSI implementation among various LR algorithms. In the following, we try to delineate the performance and complexity of two welladopted LR algorithms: the SA and the LLL algorithm. The reason we do not consider the other algorithms is that the Gaussian reduction method is only for 2×2 systems and is equivalent to SA; the Minkowski and the KZ algorithms do not have polynomial time implementation; and the performance of Brun's algorithm is much worse than that of the LLL algorithm as shown in [10] and the lack of analytical results on diversity.

The real LLL (RLLL) algorithm first is applied to improve the performance of low-complexity equalizers by extending the complex transmission system in (1) to an equivalent real model [5]. Furthermore, the complex LLL (CLLL) algorithm is proposed, based on the Gram-Schmidt orthonormalization in [2], and based on the QR-decomposition in [3]. An $M \times$ N complex matrix \tilde{H} is called an LLL-reduced basis of a lattice if it satisfies two conditions: size reduction and the δ -condition ([2, Eq. (3)] and [3, Eq. (18)] for details). It has been shown that the CLLL algorithm reduces the complexity of the RLLL algorithm without sacrificing performance [2, 3]. As shown in Table 2, the RLLL algorithm requires more basis updates than the CLLL algorithm. One basis update is defined as the process that updates the *n*-th basis vector using the *m*-th basis as $h_n \leftarrow h_n + a_{m,n}h_m$. Furthermore, the sorted QR-decomposition

59

Authorized licensed use limited to: Christos Maurokefalidis. Downloaded on December 28, 2009 at 04:02 from IEEE Xplore. Restrictions apply.

Given the array of LR algorithms in the literature, it is difficult to justify which one is better in terms of both performance and complexity. Therefore, it is hard to choose the appropriate one for VLSI implementation among various LR algorithms.

<i>M</i> = <i>N</i> =	2	4	6	8	10	
Greedy SA	Average	1.0733	5.4579	11.1725	16.9698	22.0766
Greedy SA	Std. deviation	0.6378	2.3032	4.4139	6.9265	9.3423
Peal III	Average	3.5204	19.0711	46.5706	84.7531	132.36
Keal LLL	Std. deviation	2.7614	9.6056	21.5811	39.5851	61.8293
Complex III	Average	1.1151	6.6624	16.2276	29.0076	44.2684
Complex LLL	Std. deviation	0.6963	3.0824	7.0284	12.6450	19.4835
Complex III with SOPD	Average	1.0505	5.7555	13.4083	23.3189	35.2554
Complex LLL with SQRD	Std. deviation	0.6208	2.6426	5.9486	10.3063	15.5275

Table 2. Number of basis updates needed for different LR algorithms for i.i.d. channels.

(SQRD) in [5] is introduced into the LLL process to further reduce the complexity as shown in Table 2. The CLLL algorithm was applied to the dual basis of the channel matrix, and the performance is further improved [9]. The complexity of the LLL algorithm depends on the specific realization of the channel but has an upper bound $O(N^2 \log N)$ on average [11].

As an alternative to the LLL algorithm, SA is an iterative method to reduce the lattice. The ultimate goal of SA is to find a set of bases \tilde{H} , from which the Seysen's metric cannot be reduced further. The lazy method and the greedy method are first proposed to implement SA (see references in [4]), whereas a simplified greedy implementation is proposed in [4] to further reduce the complexity. The lazy implementation of SA guarantees finding the optimal bases that minimize $S(\hat{H})$ but requires high complexity. The greedy implementation requires much fewer operations, but the algorithm may stop at a certain set of bases \tilde{H} with suboptimal $S(\tilde{H})$ (a local minimum). Different from the LLL algorithm, SA requires a fixed number of arithmetic operations in each basis update, although the number of basis updates is still random. As shown in Table 2, the number of basis updates required by simplified, greedy SA in [4] is less than that required by the CLLL algorithm and even CLLL with SQRD, in both average and standard deviation. However, the number of arithmetic operations required by SA in each basis update (16M + 104N - 90) is far more than that of the CLLL algorithm (at most (28M + 46N)+ 6) even if δ condition is violated), which leads to higher algorithm complexity. Another major drawback of SA is that it requires more memory storage during the updating process.

PERFORMANCE OF LRAES

In this section, we review performance results of different LRAEs, using either the LLL algorithm or SA, and see why the fundamental condition is met and how much diversity is restored.

As shown in [3], the CLLL algorithm upper bounds $od(\hat{H})$ by a constant strictly less than 1, indicating that the output matrix \hat{H} is constrained within a certain distance from orthogonality. Therefore, LEs based on (2) collect the same diversity as that exploited by near-MLEs based on \tilde{H} because the fundamental condition is met. However, as explained earlier, the constellation of z is extended to the whole Gaussian integer ring, which is infinite. Therefore, in the first quantization step to obtain \hat{z} with low-complexity equalizers, under a finite-bit number representation (e.g., in practical systems and simulation tools), LRAEs achieve only the asymptotic diversity that may be less than the diversity of MLE [12]. It has been proved in [3] for i.i.d. channels that LLL-aided low-complexity equalizers collect diversity order M, which is the same as MLE. However, the infinite constellation of z and finite bit representation lead to the failure of LLL-aided LEs to collect the same diversity as MLE on some systems in simulations [12].

Similar to the LLL algorithm, if Seysen's metric $S(\tilde{H})$ is upper bounded by a finite constant, it can be shown that SA-aided low-complexity equalizers collect the same diversity as MLEs. With a finite bit representation, the asymptotic diversity for infinite constellations is guaranteed. It was proved that for 2×2 systems, SA upper bounds the Seysen's metric of the output matrix \tilde{H} by a finite number. For an N-D lattice, the simulation results show the existence of a finite upper bound, but it is not proved theoretically yet. However, according to the simulation results in [4] and our own simulation experience, SA-aided equalizers collect the same diversity as LLL-aided equalizers for many transmission systems.

We compare the performance of different LRAEs in Fig. 2 for 6×6 systems with i.i.d. channels and QPSK modulation. Nine different equalizers were applied on the system. From the figure, we can see that LRAEs collect diversity 6, the same as the SD method, although there still exist performance gaps. Among different LRAEs, detectors employing the CLLL algorithm and SQRD lead to the best performance.

VLSI ARCHITECTURE

Although LR-aided equalizers achieve high diversity and thus attract much attention, currently, there is only one LR VLSI implementation reported in the literature — the Brun's algorithm-based channel pre-coder described in [10]. However, Brun's algorithm requires lower average complexity than the CLLL algorithm but achieves inferior performance, and no analytical result has been reported to prove the diversity of LR-aided equalizer with Brun's algorithm. Thus, there is a need for a hardware implementation of the CLLL algorithm, which includes ORdecomposition as the pre-processing step and the main CLLL process. The QR-decomposition was implemented efficiently using Given's Rotations (see references in [10] and [13]), but the main CLLL process remains unexplored. Examination of the main CLLL process in [3] reveals that this algorithm may not be applied on hardware directly. However, by considering the nature of the CLLL algorithm and exploiting the inherent parallelism, we arrive at a hardware architecture suitable for VLSI implementation.

Because SORD results in the main CLLL process requiring fewer basis updates, both in average and variation, we adopt the SQRD as the preprocessing step to make the complexity of the CLLL algorithm "less random." To satisfy the size reduction condition of the CLLL algorithm, we adopt Newton-Raphson iterations with a small look-up table for initial values to implement the integerrounded division required by the CLLL algorithm. The δ -condition checking of the CLLL algorithm is implemented through a well-understood and numerically stable algorithm, Householder COordinated Rotation DIgital Computer (CORDIC), which is adopted to rotate a 3-D real vector by "vectoring" to a principal axis using iterative, lowhardware, complexity operations. Then, if the δ condition is not satisfied, we use the inverse of the vectoring Householder CORDIC operations to find the unitary matrix to rotate \tilde{R} as shown in [3]. After further considerations on scheduling design, we propose a superscalar CLLL processor in Fig. 3 with a data path consisting of a Householder CORDIC module, reduced-precision division pipeline, and shared complex multiplication pipeline. Multiple control loops, as opposed to a single central controller coupled with a simple arbitration scheme, enable the CLLL processor to take advantage of the parallelism in scheduling.

Figure 4 shows the results of the simulations using fixed-point arithmetic with [G,F]-bit, where G and F are the numbers of integer and fractional bits, respectively. We adopt the 4 × 4 V-BLAST transmission as an example. The signal constellation is 4-QAM, and LR-aided ZF-DFE is employed at the receiver. We plot three curves for the same system but with floatingpoint, [13, 13], [10, 13], [13, 10], and [10, 10]-bit fixed-point representations for the \tilde{R} matrix, respectively. The figure shows that the performance of the system with [13, 13]-bit fixed-point arithmetic is nearly the same as that of a floating-point implementation, which verifies our hardware implementation.

IMPLEMENTATION RESULTS

We implemented the proposed architecture in hardware using the verilog hardware description language, verifying the implementation using a co-simulation tool we developed. We then used *Synplify Pro* with the retiming and pipelining



Figure 2. *Performance of LRAEs with* M = N = 6.



Figure 3. Simplified block diagram of superscalar CLLL processor. Solid lines indicate data flow paths, while dashed lines indicate control flow paths.

options enabled for synthesis and Xilinx ISE 9.1 for place and route. Implementation on a Virtex4 FPGA results in a design that has a maximum clock frequency of 140 MHz, an average processing latency of 928.6 ns, and requires 88,308 gate equivalents (10 real multipliers and 3617 Virtex4 slices). Implementation on a Virtex5 FPGA results in a design that has a maximum clock frequency of 163 MHz, an average processing latency of 797.5 ns, and requires 78,683 gate equivalents (10 real multipliers and 1712 Virtex5 slices). For both implementations, fixed-point representations of [1, 12] for the Qmatrix, [13, 13] for the R matrix, and [13, 0] for



■ Figure 4. Hardware fixed-point simulations of an LR-aided ZF-DFE for a 4 × 4 V-BLAST transmission using a variety of bit precisions.

the T matrix are assumed. Comparison to other architectures is difficult because no VLSI implementation of an LR-aided low-complexity equalizer that collects the same diversity as MLE has been reported in the literature. Because the proposed architecture requires few multipliers and modest FPGA resources, multiple CLLL processors could be realized on a high-end FPGA to achieve a desired throughput for a particular LR-aided hard detector. For a given channel matrix, the CLLL processor can begin operation after the first column of the QR-decomposition is computed, partially hiding the latency of the CLLL processor. Furthermore, additional area reductions and performance gains can be achieved by optimizing the VLSI implementation for a particular channel model and lowcomplexity equalizer as done in [10]. Currently, we have found that further algorithm and hardware optimizations lead to improved results, including lower average cycle count, lower fraction and integer bits, and lower hardware resource requirements.

CONCLUSIONS

Among different detectors for linear block transmissions, traditional low-complexity equalizers are favored for their cubic order polynomial complexity, but they often suffer from diversity loss. When the channel is constrained within a certain distance from orthogonality, low-complexity equalizers achieve the same diversity as MLE. LR techniques are one approach to impose a constraint on the orthogonality of the channel matrix while maintaining the low-complexity property. After a thorough investigation of LR algorithms, the CLLL algorithm with SQRD is selected to be implemented in VLSI. The proposed architecture and implementation results are presented. The encouraging results summarized in this article demonstrate that LRAEs are an attractive solution for future wireless receiver designs.1

References

- [1] X. Ma and W. Zhang, "Fundamental Limits of Linear Equalizers: Diversity, Capacity, and Complexity," *IEEE Trans. Info. Theory*, vol. 54, no. 8, Aug. 2008, pp. 3442–56.
- [2] Y. H. Gan and W. H. Mow, "Complex Lattice Reduction Algorithms for Low-Complexity MIMO Detection," Proc. IEEE Global Telecom. Conf., St. Louis, MO, vol. 5, Nov. 28–Dec. 2, 2005, pp. 2953–57.
- [3] X. Ma and W. Zhang, "Performance Analysis for MIMO Systems with Linear Equalization," *IEEE Trans. Commun.*, vol. 56, no. 2, Feb. 2008, pp. 309–18.
- [4] D. Seethaler, G. Matz, and F. Hlawatsch, "Low-Complexity MIMO Data Detection Using Seysen's Lattice Reduction Algorithm," *Proc. IEEE Int'l. Conf. Acoustics, Speech, and Signal Processing*, Honolulu, HI, vol. 3, Apr. 15–20, 2007, pp. 53–56.
 [5] D. Wübben *et al.*, "Near-Maximum-Likelihood Detection
- [5] D. Wübben et al., "Near-Maximum-Likelihood Detection of MIMO Systems Using MMSE-Based Lattice Reduction," Proc. IEEE ICC, Paris, France, vol. 2, June 20–24, 2004, pp. 798–802.
- [6] H. Yao, Efficient Signal, Code, and Receiver Designs for MIMO Communication Systems, Ph.D. diss., Dept. of Elec. Eng. and Comp. Sci., MIT, 2003.
 [7] B. Hassibi and H. Vikalo, "On the Sphere-Decoding
- [7] B. Hassibi and H. Vikalo, "On the Sphere-Decoding Algorithm: Part 1: Expected Complexity," *IEEE Trans. Sig. Processing*, vol. 53, no. 8, Aug. 2005, pp. 2806–18.
- Sig. Processing, vol. 53, no. 8, Aug. 2005, pp. 2806–18.
 [8] Y. Jiang, X. Zheng, and J. Li, "Asymptotic Performance Analysis of V-BLAST," Proc. IEEE GLOBECOM, St. Louis, MO, vol. 6, Nov. 28–Dec. 2, 2005, pp. 3882–86.
- [9] C. Ling, "Approximate Lattice Decoding: Primal versus Dual Basis Reduction," Proc. IEEE Int'l. Symp. Info. Theory, Seattle, WA, July 9–14, 2006.
- [10] A. Burg, D. Seethaler, and G. Matz, "VLSI Implementation of a Lattice-Reduction Algorithm for Multi-Antenna Broadcast Precoding," *Proc. IEEE Int'l. Symp. Circuits and Sys.*, New Orleans, LA, May 27–30, 2007, pp. 673–76.
 [11] J. Jaldén, D. Seethaler, and G. Matz, "Worst- and Aver-
- [11] J. Jaldén, D. Seethaler, and G. Matz, "Worst- and Average-Case Complexity of LLL Lattice Reduction in MIMO Wireless Systems," Proc. IEEE Int'l. Conf. Acoustics, Speech, and Sig. Processing, Las Vegas, NV, Mar. 30–Apr. 4, 2008.
- [12] W. Zhang and X. Ma, "Quantifying Diversity for Wireless Systems with Finite-Bit Representation," Proc. IEEE Int'l. Conf. Acoustics, Speech, and Sig. Processing, Las Vegas, NV, Mar. 30–Apr. 4, 2008.
- [13] B. Gestner et al., "VLSI Implementation of a Lattice Reduction-Aided Low-Complexity Equalizer," Proc. IEEE Int'I. Conf. Circuits and Sys. Commun., Shanghai, China, May 26–28, 2008, pp. 643–47.

BIOGRAPHIES

WEI ZHANG [S'05] (zhangw1@ece.gatech.edu) received a B.S. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2004 and an M.Sc. degree in electrical engineering from Auburn University, Alabama, in 2006. He is now working toward a Ph.D. degree in the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta.

XIAOLI MA [M'03] (xiaoli@ece.gatech.edu) received a B.S. degree in automatic control from Tsinghua University, Beijing, China, in 1998, an M.S. degree in electrical engineering from the University of Virginia in 2000, and a Ph.D. degree in electrical engineering from the University of Minnesota in 2003. She is an assistant professor in the School of Electrical and Computer Engineering at Georgia Institute of Technology. Her research interests include transceiver designs and diversity techniques for wireless channels.

BRIAN GESTNER [S'04] (bgestner@ece.gatech.edu) received his B.S. and M.Sc. degrees in electrical engineering from Carnegie Mellon University, Pittsburgh, Pennsylvania, in 2004. He is now working toward his Ph.D. degree in the School of Electrical and Computer Engineering, Georgia Institute of Technology.

DAVID V. ANDERSON [SM'05] (dva@ece.gatech.edu) received his B.S. and M.S. degrees from Brigham Young University and his Ph.D. degree from Georgia Institute of Technology in 1993, 1994, and 1999, respectively. His research interests include audio and psychoacoustics, signal processing in the context of human auditory characteristics, and the real-time application of such techniques using both analog and digital hardware.

¹ The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government.