A FIXED-COMPLEXITY MIMO DETECTOR BASED ON THE COMPLEX SPHERE DECODER

Luis G. Barbero and John S. Thompson

Institute for Digital Communications
University of Edinburgh
Email: {l.barbero, john.thompson}@ed.ac.uk

ABSTRACT

A new detection algorithm for uncoded multiple input-multiple output (MIMO) systems based on the complex version of the sphere decoder (SD) is presented in this paper. The algorithm performs a fixed number of operations to detect the symbols, independent of the noise level. The algorithm achieves this by combining a novel channel matrix preprocessing with a search through a small subset of the complete receive constellation. Simulation results show it has only a very small bit error ratio (BER) degradation compared to the original SD while being suited for a fully-pipelined hardware implementation due to its fixed complexity.

1. INTRODUCTION

The use of multiple input-multiple output (MIMO) technology has become the new frontier of wireless communications. It enables high-rate data transfers and improved link quality through the use of multiple antennas at both transmitter and receiver [1]. The optimum receiver for MIMO systems is the maximum likelihood detector (MLD), but its exponential complexity makes it unrealizable in practical systems when a large number of antennas and higher order constellations are used. The sphere decoder (SD) has been proposed as an alternative, providing maximum likelihood (ML) performance with reduced complexity [2]. Although its average complexity is believed to be polynomial for small array sizes [3], the actual complexity depends on the channel conditions and the noise level, making it difficult to integrate in an actual system where data needs to be processed at a constant rate (i.e. fixed complexity).

Different methods have been proposed to reduce or limit the complexity of the SD although most of them still have a variable complexity depending on the channel conditions. They can be classified in the following categories:

- Modifications of the algorithm to marginally reduce the complexity requiring additional operations or the calculation of limiting thresholds [4]-[6].
- Simplifications of the algorithm for specific constellation types [7].
- Application of the K-Best lattice decoder [8] (equivalent to the sequential M-algorithm [9]).
- A combination of the SD and the K-Best lattice decoder [10].

The K-Best lattice decoder is the only one that provides a fixed complexity although it is considerably higher than the complexity of the SD in order to guarantee a quasi-ML performance. The other alternatives give a reduced complexity that is still variable and makes the algorithm architecture more complex for practical implementation.

In this paper, a new MIMO detector based on the complex SD is proposed that achieves quasi-ML performance in a fixed number of operations. Thus, a parallel implementation of the algorithm can be fully pipelined making it suitable for next-generation wireless communication systems.

2. MIMO SYSTEM MODEL

The system model considered has M transmit and N receive antennas, with $N \geq M$, denoted as $M \times N$. The transmitted symbols are taken independently from a quadrature amplitude modulation (QAM) constellation of P points forming an M-dimensional complex constellation $\mathcal{C}$ of P^M points. The received N-vector, using matrix notation, is given by

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{v} \tag{1}$$

where $\mathbf{s} = (s_1, s_2, \ldots, s_M)^T$ denotes the vector of transmitted symbols with $\mathrm{E}[|s_i|^2] = 1/M$, $\mathbf{v} = (v_1, v_2, \ldots, v_N)^T$ is the vector of independent and identically distributed (i.i.d.) complex Gaussian noise samples with variance $\sigma^2 = N_0$ and $\mathbf{r} = (r_1, r_2, \ldots, r_N)^T$ is the vector of received symbols. $\mathbf{H}$ denotes the $N \times M$ channel matrix where h_{ij} is the complex transfer function from transmitter j to receiver i. The entries of $\mathbf{H}$ are modelled as i.i.d. Rayleigh fading with $\mathrm{E}[|h_{ij}|^2] = 1$ and are perfectly estimated at the receiver.

Since the elements of $\mathbf{H}$ are i.i.d. complex Gaussian, $\mathbf{H}$ has rank M and, therefore, the set $\{\mathbf{Hs}\}$ can be considered as the complex lattice $\Lambda(\mathbf{H})$ generated by $\mathbf{H}$. The detector proposed here is directly applied to the complex lattice so that it can be used for complex constellations different from QAM in a similar way to [11]. In addition, avoiding the more common real decomposition would result in a more efficient hardware implementation as shown for the SD in [12]. This new detector can also be applied to the real decomposition of the system giving a similar performance and complexity trade-off.

3. FIXED-COMPLEXITY SPHERE DECODER (FSD)

The main idea of the SD is to reduce the computational complexity of the MLD by searching over only those points of the lattice that lie within a hypersphere of radius R around the received signal [2], [13]. The value of the initial radius R limits the number of points of the lattice searched, therefore reducing the complexity compared to the MLD. If the Fincke-Pohst (FP) enumeration is used, the initial radius is selected according to the noise variance per antenna, in order to make sure that, at least, one points is found inside the hypersphere [14]. On the other hand, if the Schnorr-Euchner (SE) enumeration is used, the initial radius can be set to a very large value without affecting the final complexity of the algorithm and removing the need for an estimate of the noise level at the receiver [12], [15]. The SD search can be represented by

$$\hat{\mathbf{s}}_{\mathbf{ml}} = \arg\{\min_{\mathbf{s}} \|\mathbf{r} - \mathbf{H}\mathbf{s}\|^2 \leq R^2\} \quad (2)$$

where the presence of the initial radius has been mantained to indicate the spherical nature of the search.

The sphere constraint in (2) can also be written, after matrix decomposition and removal of constant terms, as

$$\|\mathbf{U}(\mathbf{s} - \hat{\mathbf{s}})\|^2 \leq R^2 \quad (3)$$

where $\mathbf{U}$ is an $M \times M$ upper triangular matrix, with entries denoted u_{ij}, obtained through Cholesky decomposition of the Gram matrix $\mathbf{G} = \mathbf{H}^H\mathbf{H}$ (or, equivalently, QR decomposition of $\mathbf{H}$) and $\hat{\mathbf{s}} = (\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H\mathbf{r}$ is the unconstrained ML estimate of $\mathbf{s}$ [11].

The solution of (3) can be obtained recursively starting from $i = M$ and working backwards until $i = 1$. For each level, the constellation points s_i that satisfy

$$|s_i - z_i|^2 \leq \frac{T_i}{u_{ii}^2} \quad (4)$$

are selected as partial ML candidates, where

$$z_i = \hat{s}_i - \sum_{j=i+1}^{M} \frac{u_{ij}}{u_{ii}}(s_j - \hat{s}_j) \quad (5)$$

and

$$T_i = R^2 - \sum_{j=i+1}^{M} u_{jj}^2 |s_j - z_j|^2. \quad (6)$$

The points s_i on each level that satisfy (4) can be obtained through direct calculation of the P $|s_i - z_i|^2$ values or decomposing the QAM constellation in concentric circles and identifying the valid points in each circle as presented in [11]. When a new point is found inside the hypersphere (at $i = 1$) the radius is updated with the new minimum Euclidean distance and the algorithm continues the search with the new sphere constraint.

3.1. FSD Algorithm

From an implementation point of view, the SD has two main drawbacks. Firstly, the detector complexity depends on the noise level and the channel conditions and, secondly, the sequential nature of the search limits the performance and the level of parallelism of a hardware implementation of the algorithm. A new fixed-complexity sphere decoder (FSD) is proposed to overcome those two problems by searching, independently of the noise level, over only a fixed number of lattice points $\mathbf{Hs}$, generated by a subset $\mathcal{S} \subset \mathcal{C}$, around the received point $\mathbf{r}$.

The algorithm makes use of the fact that the diagonal entries of $\mathbf{U}$, u_{ii}, are such that $2u_{ii}^2$ are real-valued and have a Chi-square (χ^2) distribution with $2(N - i + 1)$ degrees of freedom and $\mathrm{E}[u_{ii}^2] = N - i + 1$, with $i = 1, \ldots, M$, as shown in [16] and references therein. Therefore, the diagonal elements u_{ii} satisfy

$$\mathrm{E}[u_{MM}^2] < \mathrm{E}[u_{M-1M-1}^2] < \cdots < \mathrm{E}[u_{11}^2]\,. \quad (7)$$

If we denote n_i the number of candidates at level i that satisfy (4), with $1 \leq n_i \leq P$, we obtain from (7) that

$$\mathrm{E}[n_M] \geq \mathrm{E}[n_{M-1}] \geq \cdots \geq \mathrm{E}[n_1]. \quad (8)$$

Using the result in (8), the FSD assigns a fixed number of candidates, n_i, to be searched per level independent of the initial radius. This can be explained as follows: whereas in the first level, $i = M$, more candidates need to be considered due to interference from the other levels, the decision-feedback equalization (DFE) performed on z_i and the increase in $\mathrm{E}[u_{ii}^2]$ reduces the number of candidates that need to be considered in the last levels.

The total number of candidates whose Euclidean distance is calculated is, therefore, $N_{\mathcal{S}} = \prod_{i=1}^{M} n_i$, where simulations show that quasi-ML performance is achieved with $N_{\mathcal{S}} \ll P^M$, i.e. $\mathcal{S}$ is a very small subset of $\mathcal{C}$. The n_i candidates on each level i are selected according to increasing distance to z_i, following the SE enumeration [15].

Fig. 1 shows a hypothetical subset $\mathcal{S}$ in a 4×4 system with 4-QAM modulation where the number of points per level $\mathbf{n}_{\mathcal{S}} = (n_1, n_2, n_3, n_4)^T = (1, 1, 2, 3)^T$. In each level i, the n_i closest points to z_i are considered as components of the subset $\mathcal{S}$.

A trade-off exists between the complexity and the performance of the FSD. If more candidates are searched, the performance will be closer to that of the original SD but the required computational power will increase. That makes the

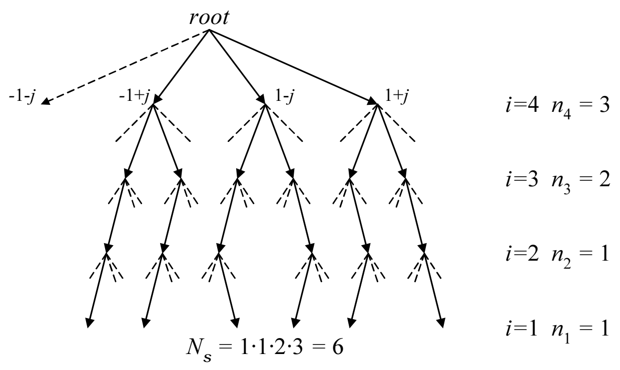

Fig. 1. Example of points $\mathbf{s} \in \mathcal{S}$ in a 4×4 system with 4-QAM modulation

FSD suitable for reconfigurable architectures where the number of candidates can be made adaptive depending on the MIMO channel conditions.

3.2. FSD Preprocessing of the Channel Matrix

A novel method is proposed for the preprocessing of the channel matrix in the FSD. It determines the detection ordering of the signals $\hat{s}_i$ according to the distribution of candidates, $\mathbf{n}_{\mathcal{S}}$, that is used in the receiver.

The FSD preprocessing iteratively orders the M columns of the channel matrix. On the i-th iteration, considering only the signals still to be detected, the signal $\hat{s}_k$ (the index k is used to indicate that it does not necessarily coincide with the index i) with the smallest post-detection noise amplification, as calculated in [17], is selected if $n_i < P$. If $n_i = P$, the signal with the largest noise amplification is selected instead.

The steps performed in every iteration are the following (for $i = M, \ldots, 1$):

1. The matrix $\mathbf{H}_i^{\dagger} = (\mathbf{H}_i^H \mathbf{H}_i)^{-1} \mathbf{H}_i^H$ is calculated, where $\mathbf{H}_i = \mathbf{H}_{\mathbf{k}_{i+1}}$ is the channel matrix with the columns selected in previous iterations zeroed (represented by the index vector $\mathbf{k}_{i+1}$).

2. The signal $\hat{s}_k$ to be detected is selected according to

$$k = \begin{cases} \arg\{\max_j \|(\mathbf{H}_i^{\dagger})_j\|^2\}, & \text{if } n_i = P \\ \arg\{\min_j \|(\mathbf{H}_i^{\dagger})_j\|^2\}, & \text{if } n_i \neq P \end{cases} \quad (9)$$

   where $(\mathbf{H}_i^{\dagger})_j$ represents the jth row of $\mathbf{H}_i^{\dagger}$ with $j \in [1, M] - \{\mathbf{k}_{i+1}\}$.

The following heuristic supports this ordering approach: if the maximum possible number of candidates, P, is searched in one level, the *robustness* of the signal is not relevant to the final performance, therefore, the signals that suffer the largest noise amplification can be be detected in the levels where $n_i = P$.

	Preprocessing of $\mathbf{H}$			
	No ordering		FSD ordering	
	mean	std deviation	mean	std deviation
n_1	1.0	0.0	1.0	0.0
n_2	1.0069	9.2513×10^{-2}	1.0005	2.3771×10^{-2}
n_3	1.0466	2.7643×10^{-1}	1.0006	2.4598×10^{-2}
n_4	1.3896	9.9812×10^{-1}	1.7326	1.3916×10^{0}

Table 1. Mean and standard deviation of n_i for the SE-SD in a 4×4 system with 16-QAM for different preprocessings of $\mathbf{H}$ at $\frac{E_b}{N_0}$ = 15dB

4. RESULTS

The performance and complexity of the FSD has been obtained via Monte Carlo simulations for different constellations and MIMO configurations. The main aim is to evaluate its suitability for quasi-ML detection in a fixed number of operations in systems where the MLD is unfeasible due to its complexity. The results have been obtained using 50,000 channel realizations with 200 uncoded symbols transmitted in every channel realization.

A key aspect in the performance and complexity of the FSD is the choice of the distribution of points $\mathbf{n}_{\mathcal{S}}$. However, the correlation between the values n_i, due to the DFE performed on z_i, and the FSD ordering of the channel matrix make it difficult to obtain a close analytical expression for the distribution of points. Simulations results have been used to initially identify optimum distributions and infer the evolution for different number of antennas and constellation orders.

Table 1 shows the mean and the standard deviation of the number of points n_i that need to be considered per level to find the ML solution in the SE version of the SD for a 4×4 system with 16-QAM. The results have been obtained for a signal to noise ratio (SNR) per bit of 15 dB. The SD without channel matrix ordering has been compared with the FSD ordering applied to the SD. In the latter, the signal with the largest noise amplification is detected in the first level, $i = M$.

It can be seen that, in the FSD ordering, the mean and the standard deviation of the number of points in the first level, n_4, is higher than in the no ordering case. This is consistent with the fact that the signal with the lowest *quality* is detected in the first level. On the other hand, for the subsequent levels, the standard deviation is significantly reduced, while the mean is slightly reduced. From an implementation point of view, the standard deviation results in Table 1 for the FSD ordering indicate that, in the first level, more points should be checked in order to find the ML solution. In addition, the ordering presented in section 3.2 requires that all the constellation points should be considered ($n_M = P$), given that the signal that suffers the largest noise amplification is detected

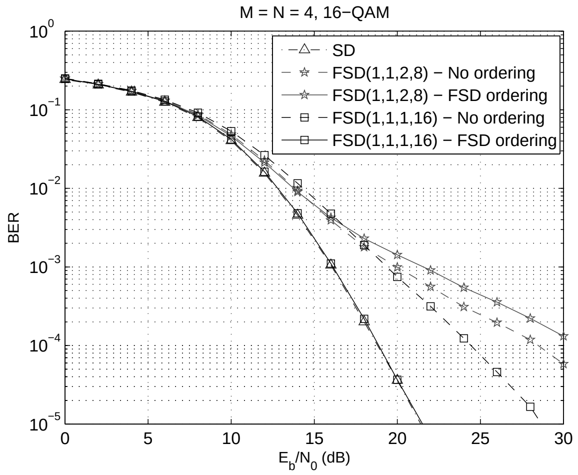

Fig. 2. BER performance of the FSD as a function of the SNR per bit for different distributions of points.

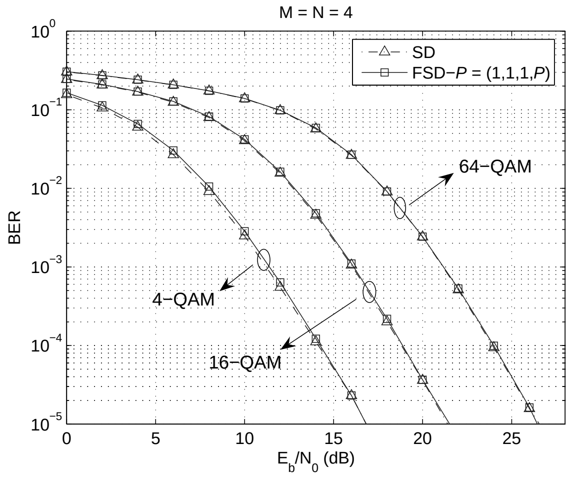

Fig. 3. BER performance of the FSD and the SD as a function of the SNR per bit in a 4×4 system.

in that level. For the remaining levels ($i \neq M$), the reduction in the standard deviation indicates that considering only one point ($n_i = 1$ for $i \neq M$) would give the ML solution with higher probability than for the no ordering case.

In order to validate the previous reasoning, different simulations have been run to compare the performance of the FSD using different distributions of points with and without FSD ordering. Fig. 2 shows the bit error ratio (BER) performance of the FSD as function of the SNR per bit in a 4×4 system using 16-QAM compared to the ML performance provided by the SD. The FSD checks a total of 16 points using the distributions $\mathbf{n}_{\mathcal{S}_1} = (1, 1, 1, 16)^T$ and $\mathbf{n}_{\mathcal{S}_2} = (1, 1, 2, 8)^T$. With no ordering of the channel matrix, the distribution $\mathbf{n}_{\mathcal{S}_2}$ yields a better performance at low SNR, given that the noise level requires more points to be checked in the levels where $i \neq M$. At high SNR, the distribution $\mathbf{n}_{\mathcal{S}_1}$ gives a better performance (with a cross over at E_b/N_0 = 18 dB). In this case, due to the low level of noise, it is more relevant to check all the points in the first level (to capture the cases with high power noise samples) than to check additional points in the following levels.

The performance has also been measured when the FSD ordering is applied to the channel matrix. In both cases, the signal with the largest noise amplification is detected in the first level independently of the number of points checked. It can be observed how the distribution $\mathbf{n}_{\mathcal{S}_2}$ has a worse performance compared to the no ordering case. In that case, checking only 8 points in the first level is not sufficient due to the noise amplification in that level. On the other hand, the FSD ordering considerably improves the performance of the $\mathbf{n}_{\mathcal{S}_1}$ distribution, achieving quasi-ML performance. The FSD ordering yields a gain of 3.35 dB at a BER = 10^{-3} when using the distribution $\mathbf{n}_{\mathcal{S}_1}$ and provides the FSD with a diversity order (i.e. slope of the BER curve) equal to that of the MLD.

Fig. 3 shows the bit error ratio (BER) performance of the FSD in a 4×4 system using 4-,16- and 64-QAM modulation. Using the results presented above, the total number of points searched in the FSD is $N_{\mathcal{S}} = P$ for a P-QAM constellation following the distribution $\mathbf{n}_{\mathcal{S}} = (1, 1, 1, P)^T$. This distribution has the additional advantage that the SE enumeration is not necessary, further simplifying the receiver. The channel matrix has been ordered using the FSD preprocessing, minimizing the BER for the selected distribution of candidates $\mathbf{n}_{\mathcal{S}}$. It can be observed that the FSD gives practically ML performance independent of the SNR, especially for larger constellations, by calculating only P Euclidean distances. The performance curves for the K-Best lattice decoder have not been included for clarity purposes. However, we have observed that, for 16-QAM and at a BER=10^{-3}, the performance degradation of the FSD compared to the SD is of 0.06 dB while the K-Best decoder (with $K = 16$) has a degradation of 0.015 dB.

The number of real floating point operations of the FSD is shown in Fig. 4 where its fixed nature can be observed. The FSD is compared to the SE-SD with and without channel matrix ordering in a 4×4 system using 16-QAM modulation (vertical Bell Labs layered space time-zero forcing (VBLAST-ZF) ordering used as in [13]). The 90-percentile is plotted to indicate the number of operations required to perform the detection process in 90% of the cases. It can be seen how only at high SNR is the number of operations of the FSD slightly higher than of the SD. However, the fixed structure of the FSD would allow a fully-pipelined parallel implementation of the algorithm achieving a higher throughput (i.e. number of bits detected per second) compared to the SD. The number of operations of the complex version of the K-Best lattice decoder is also plotted where it can bee seen that it suffers from a considerably higher fixed complexity.

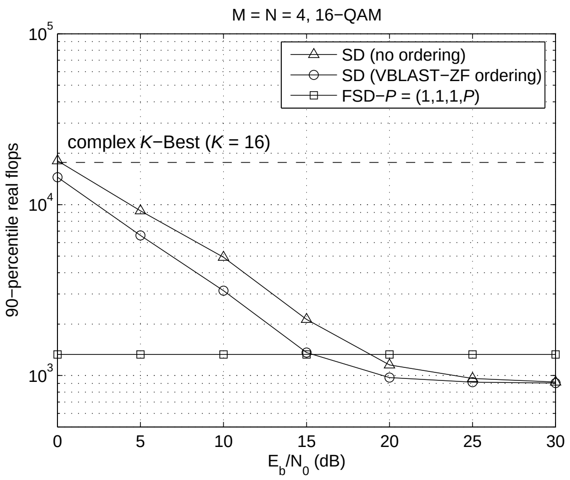

Fig. 4. Complexity of the search stage of the FSD and the SE-SD as a function of the SNR per bit in a 4×4 system.

5. CONCLUSION AND FUTURE WORK

A new fixed complexity MIMO detector has been proposed that provides quasi-ML performance independent of the noise level. The algorithm calculates the Euclidean distances of a very small subset of points of the complete receive constellation and uses a novel preprocessing method of the channel matrix tailored to that subset. Its fixed complexity makes it a very suitable algorithm for hardware implementation and integration in a complete wireless system where a minimum throughput needs to be guaranteed.

The analysis of the FSD and the required distribution of points for larger systems and a real-time hardware implementation of this algorithm are the main subjects of ongoing work.

6. ACKNOWLEDGEMENT

The authors would like to thank Alpha Data Ltd., company that partially sponsors this research.

7. REFERENCES

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

[2] E. Viterbo and J. Boutros, “A universal lattice code decoder for fading channels,” *IEEE Trans. Inform. Theory*, vol. 45, no. 5, pp. 1639–1642, July 1999.

[3] B. Hassibi and H. Vikalo, “On the expected complexity of sphere decoding,” in *Proc. 35th Asilomar Conference on Signals, Systems and Computers*, vol. 2, Monterey, CA, Nov. 2001, pp. 1051–1055.

[4] L. Brunel, “Multiuser detection techniques using maximum likelihood sphere decoding in multicarrier cdma systems,” *IEEE Trans. Wireless Commun.*, vol. 3, no. 3, pp. 949–957, May 2004.

[5] Z. Guo and P. Nilsson, “Reduced complexity Schnorr-Euchner decoding algorithms for MIMO systems,” *IEEE Commun. Lett.*, vol. 8, no. 5, pp. 286–288, May 2004.

[6] W. Zhao and G. B. Giannakis, “Sphere decoding algorithms with improved radius search,” *IEEE Trans. Commun.*, vol. 53, no. 7, pp. 1104–1109, July 2005.

[7] Z. Safar, W. Su, and K. R. Liu, “Fast sphere decoding of space-frequency block codes via nearest neighbor signal point search,” in *Proc. 5th European Wireless Conference (EW ’04)*, vol. 1, Barcelona, Spain, Feb. 2004.

[8] K. wai Wong, C. ying Tsiu, R. S. kwan Cheng, and W. ho Mow, “A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels,” in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS ’02)*, vol. 3, Scottsdale, AZ, May 2002, pp. 273–276.

[9] J. B. Anderson and S. Mohan, “Sequential coding algorithms: A survey and cost analysis,” *IEEE Trans. Commun.*, vol. 32, no. 2, pp. 169–176, Feb. 1984.

[10] J. Tang, A. H. Tewfik, and K. K. Parhi, “Reduced complexity sphere decoding and application to interfering IEEE 802.15.3a piconets,” in *Proc. IEEE International Conference on Communications (ICC ’04)*, vol. 5, Paris, France, June 2004, pp. 2864–2868.

[11] B. M. Hochwald and S. ten Brink, “Achieving near-capacity on a multiple-antenna channel,” *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[12] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bölcskei, “VLSI implementation of MIMO detection using the sphere decoding algorithm,” *IEEE J. Solid-State Circuits*, vol. 40, no. 7, pp. 1566–1577, July 2005.

[13] M. O. Damen, H. E. Gamal, and G. Caire, “On maximum-likelihood detection and the search for the closest lattice point,” *IEEE Trans. Inform. Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.

[14] U. Fincke and M. Pohst, “Improved methods for calculating vectors of short length in a lattice, including a complexity analysis,” *Mathematics of Computation*, vol. 44, no. 170, pp. 463–471, Apr. 1985.

[15] C. P. Schnorr and M. Euchner, “Lattice basis reduction: Improved practical algorithms and solving subset sum problems,” *Mathematical Programming*, vol. 66, pp. 181–199, 1994.

[16] A. M. Tulino and S. Verdú, *Random Matrix Theory and Wireless Communications*. Hanover, MA: Now Publishers, 2004, p. 22.

[17] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, “V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel,” in *Proc. URSI International Symposium on Signals, Systems and Electronics (ISSSE ’98)*, Atlanta, GA, Sept. 1998, pp. 295–300.