Reduced Complexity Closest Point Decoding Algorithms for Random Lattices

Wanlun Zhao, *Student Member, IEEE*, and Georgios B. Giannakis, *Fellow, IEEE*

***Abstract*—Closest point algorithms find wide applications in decoding block transmissions encountered with single- or multiuser communication links relying on a single or multiple antennas. Capitalizing on the random channel and noise models typically encountered in wireless communications, the sphere decoding algorithm (SDA) and related complexity-reducing techniques are approached in this paper from a probabilistic perspective. With both theoretical analysis and simulations, combining SDA with detection ordering is justified. A novel probabilistic search algorithm examining potential candidates in a descending probability order is derived and analyzed. Based on probabilistic search and an error-performance-oriented fast stopping criterion, a computationally efficient layered search is developed. Having comparable decoding complexity to the nulling–canceling (NC) algorithm with detection ordering, simulations confirm that the novel layered search achieves considerable error-performance enhancement.**

***Index Terms*—Closest point algorithm, Lenstra, Lenstra, and Lovasz (LLL) lattice reduction, multiuser detection, random lattice decoding, space–time, sphere decoding.**

I. Introduction

CONSIDER the generic complex model given as

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

where vectors $\mathbf{y}, \mathbf{v} \in \mathbb{C}^M$, and $\mathbf{s} \in \mathbb{Z}^N[j]$, and the matrix $\mathbf{H} \in \mathbb{C}^{M\times N}$ has full column rank with $M \geq N$. Here, $\mathbb{Z}$ and $\mathbb{C}$ denote the sets of integers and complex numbers, respectively, and $\mathbb{Z}[j] := \{a + jb | a, b \in \mathbb{Z}\}$ is the set of Gaussian integers. Operating on $\mathbf{s}$, the matrix $\mathbf{H}$ generates a lattice that we denote as $\Lambda(\mathbf{H}) := \{\mathbf{x} = \mathbf{H}\mathbf{s} | \mathbf{s} \in \mathbb{Z}^N[j]\}$. The closest point problem is stated as follows: Given $\mathbf{y} \in \mathbb{C}^M$ and a lattice Λ with a known generator $\mathbf{H}$, find the lattice vector $\hat{\mathbf{x}} \in \Lambda$ that minimizes the Euclidean distance from $\mathbf{y}$ to $\hat{\mathbf{x}}$; that is, $\hat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\Lambda} \|\mathbf{y} - \mathbf{x}\|^2$, where $\|\cdot\|$ represents the vector norm.

In a wireless communication context, $\mathbf{s}$, $\mathbf{y}$, and $\mathbf{v}$ are the transmitted, received, and the additive white Gaussian noise (AWGN) vectors, whereas $\mathbf{H}$ contains the channel coefficients. The distribution of $\mathbf{v}$ is $\mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$, where $\mathcal{CN}(\cdot,\cdot)$ represents the complex Gaussian distribution; and $\mathbf{H}$ is a random matrix often with known statistical properties. Furthermore, instead of the whole integer lattice $\mathbb{Z}^N[j]$, the vector $\mathbf{s}$ is usually drawn from a finite alphabet (FA; subset) $\mathcal{S}^N \subset \mathbb{Z}^N[j]$. In block decoding, we are interested in determining the maximum likelihood (ML) estimate of $\mathbf{s}$, subject to FA constraints, given as

$$\hat{\mathbf{s}}_{\mathrm{ML}} = \arg\min_{\mathbf{s}\in\mathcal{S}^N} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2.$$

Under FA constraints, closest point algorithms have been employed to find $\hat{\mathbf{s}}_{\mathrm{ML}}$ in various applications including space–time decoding and multiuser detection. The sphere decoding algorithm (SDA) was originally introduced to determine vectors with small norms in an arbitrary lattice [9], but has gained popularity in decoding lattice codes [8], [19], code division multiple access (CDMA) [3], and space–time transmissions [17], [20]. Its popularity stems from the fact that SDA offers near-ML optimality at average polynomial complexity in the medium to high signal-to-noise ratio (SNR) regime—a major reduction of the exponential (in codeword length N and the constellation size $|\mathcal{S}|$) complexity incurred by exhaustive search. A variate of SDA, first utilized by Schnorr and Euchner (SE), appeared recently in both [1] and [5], where an ordering mechanism was introduced to improve search efficiency. Under common random channel and noise models, the average complexity of the SDA was derived in [12] along with an effective method to determine the initial search radius. A soft-decision list SDA was developed in [13] to approach the capacity of multiantenna channels.

In this paper, we improve the computational efficiency of closest point algorithms by exploiting the random noise and channel models. In Section II, we briefly review the SDA and related techniques. In Section III, we provide a statistical justification for combining detection ordering with the SDA. In Section IV, we develop and analyze a probabilistic search algorithm that examines potential candidates in a descending probabilistic order. In Section V, we design an efficient layered search algorithm. We present simulation results in Section VI.

Notation: Upper (lower) bold face letters denote matrices (column vectors); $(\cdot)^T$, $(\cdot)^H$, and $(\cdot)^\dagger$ denote matrix transpose, Hermitian, and pseudoinverse, respectively; $\Re(\cdot)$ and $\Im(\cdot)$ denote the real and imaginary parts; and $E[\cdot]$ stands for expectation. Finally, $\mathcal{G}(n,\lambda)$ denotes the Gamma distribution with density function $f(x) = 1/(\lambda\Gamma(n))(x/\lambda)^{n-1}\exp(-x/\lambda)$,

Manuscript received August 18, 2003; revised July 19, 2004; accepted October 1, 2004. The editor coordinating the review of this paper and approving it for publication is K. Narayanan. This paper was presented in part in the Proceedings of the 41st Allerton Conference, University of Illinois, Monticello, IL, October, 2003. This work was prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

The authors are with the Department of Electrical Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: wlzhao@ece.umn.edu; georgios@ece.umn.edu).

Digital Object Identifier 10.1109/TWC.2005.858034

whereas $\Gamma(n)$ denotes the Gamma function. For brevity, $\mathcal{G}(n)$ denotes the standard Gamma distribution with $\lambda = 1$.

II. SDA and Its Complexity-Reducing Techniques

Even though the closest point problem is, in general, NP-hard [7], there are known relatively efficient algorithms to solve it. These include the SDA of Fincke and Pohst [9] and Kannan's algorithm [14]. A slightly modified version of the SDA will be reviewed here.

The basic idea behind SDA is to search for $\hat{\mathbf{s}}_{\mathrm{ML}}$ in a hypersphere of radius r centered at $\mathbf{y}$. Even though lattice points in this hypersphere are searched exhaustively, calculations are performed recursively so that intermediate computations are efficiently reused. Specifically, it is easy to see that the squared Euclidean distance between $\mathbf{y}$ and an arbitrary lattice point $\mathbf{x} = \mathbf{H}\mathbf{s}$ within radius r is given as

$$\|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 = \|\mathbf{H}(\mathbf{s} - \hat{\mathbf{s}})\|^2 + c = \|\mathbf{R}(\mathbf{s} - \hat{\mathbf{s}})\|^2 + c \leq r^2 \tag{2}$$

where $\hat{\mathbf{s}} := \mathbf{H}^{\dagger}\mathbf{y}$ is the unconstrained least squares solution, $c := \|(\mathbf{I} - \mathbf{H}\mathbf{H}^{\dagger})\mathbf{y}\|^2$ does not depend on $\mathbf{s}$, and $\mathbf{R}$ is the upper triangular matrix in the $\mathbf{QR}$ decomposition of $\mathbf{H}$. The constant c is the noise energy in the orthogonal complement of the column space of $\mathbf{H}$. We will ignore it from now on. Letting $\mathbf{R} = [r_{i,j}]$ with $r_{i,i} > 0$ and $r_{i,j} = 0$ for $i > j$, we obtain

$$\begin{aligned}
&\|\mathbf{R}(\mathbf{s} - \hat{\mathbf{s}})\|^2 \\
&\quad = r_{N,N}^2 |s_N - \hat{s}_N|^2 + r_{N-1,N-1}^2 \\
&\qquad \times \left| s_{N-1} - \hat{s}_{N-1} + \frac{r_{N-1,N}}{r_{N-1,N-1}}(s_N - \hat{s}_N) \right|^2 + \cdots \\
&\quad = r_{N,N}^2 |s_N - \rho_N|^2 \\
&\qquad + r_{N-1,N-1}^2 |s_{N-1} - \rho_{N-1}|^2 + \cdots \qquad (3) \\
&\quad = r_{N,N}^2 \left[\Re(s_N - \rho_N)\right]^2 \\
&\qquad + r_{N,N}^2 \left[\Im(s_N - \rho_N)\right]^2 + \cdots \qquad (4)
\end{aligned}$$

where

$$\rho_k := \hat{s}_k - \sum_{j=k+1}^{N} \frac{r_{k,j}}{r_{k,k}}(s_j - \hat{s}_j), \qquad k = N, \ldots, 1. \tag{5}$$

Separating the real and imaginary parts as in (4), the N-dimensional complex problem of finding $\arg\min_{\mathbf{s}} \|\mathbf{R}(\mathbf{s} - \hat{\mathbf{s}})\|^2$ is converted into a $2N$-dimensional real one. From (5), it follows that ρ_k depends on $s_{k+1}, \ldots, s_N$. Hence, in the right-hand side (RHS) of (3), the second term depends on s_N, the third term depends on both s_N and s_{N-1}, etc. This recursive dependency is a direct consequence of the upper triangular structure of $\mathbf{R}$. Since only those lattice points inside a hypersphere around $\mathbf{y}$ are checked, a set of necessary conditions for the candidate s_k can be derived based on (4) for $k = N, N-1, \ldots, 1$

$$\begin{aligned}
\lceil \Re\rho_k - \zeta_k \rceil \leq \Re s_k \leq \lfloor \Re\rho_k + \zeta_k \rfloor \\
\lceil \Im\rho_k - \eta_k \rceil \leq \Im s_k \leq \lfloor \Im\rho_k + \eta_k \rfloor
\end{aligned} \tag{6}$$

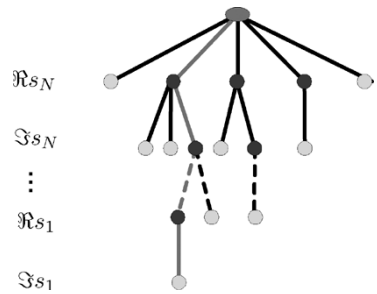

Fig. 1. Search tree of SDA.

where $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote the ceiling and floor operations, $\zeta_k := [r^2 - \sum_{i=k+1}^{N} r_{i,i}^2 |s_i - \rho_i|^2]^{1/2}/r_{k,k}$ and $\eta_k := [r^2 - \sum_{i=k+1}^{N} r_{i,i}^2 |s_i - \rho_i|^2 \quad - r_{k,k}^2 [\Re(s_i - \rho_i)]^2]^{1/2}/r_{k,k}$. These bounds limit the number of candidates in each dimension, thus enabling search complexity reduction.

The SDA proceeds as follows: The candidates for $\Re s_N$ are determined first, say $\Re s_N(1), \ldots, \Re s_N(n)$, which decomposes the original problem into n subproblems. For each candidate $\Re s_N(k)$, there is a corresponding $2N-1$-dimensional search problem. Accordingly, the squared radius for each $(2N-1)$-dimensional sphere can be reduced to

$$r_k^2 = r^2 - r_{N,N}^2 \left[\Re\left(s_N(k) - \rho_N\right)\right]^2, \qquad k = 1, \ldots, n.$$

SDA is recursively applied to these subproblems. In fact, this algorithm creates a search tree as depicted in Fig. 1, where the children of the same parent node share the path from their parent to the root. Hence, intermediate calculations are efficiently used. Furthermore, whenever a candidate of $\mathbf{s}$ is found with a distance less than r, the radius of the search sphere is reduced to the new distance.

A. Nulling–Canceling (NC) Decoding

NC finds an approximately nearest solution in polynomial time [10]. Also known as the Babai nearest plane algorithm [2], the NC yields

$$\mathbf{s}_{\mathrm{NC}} := [\lceil \rho_1 \rfloor, \lceil \rho_2 \rfloor, \ldots, \lceil \rho_{N-1} \rfloor, \lceil \rho_N \rfloor]^T \tag{7}$$

where $\lceil x \rfloor$ denotes the closest Gaussian integer to $x \in \mathbb{C}$. Starting from index N and working downwards to index 1, each s_k is chosen to be $\lceil \rho_k \rfloor$, where ρ_k depends on $s_N, s_{N-1}, \ldots, s_{k+1}$ as in (5). At step k, the distance contribution of the term involving s_k in (3) is minimized without considering its effect on subsequent terms. Therefore, NC is a greedy algorithm in nature.

B. SE-SDA

Recently, a variant of the SDA appeared in both [1] and [5]. The key difference of this algorithm from the conventional SDA lies in a simple ordering of the candidates determined by the necessary conditions in (6). Specifically, if $\Re\rho_k \leq \lceil \Re\rho_k \rfloor$, the candidates for $\Re s_k$ are checked in the order

$$\Re s_k = \lceil \Re\rho_k \rfloor, \lceil \Re\rho_k \rfloor - 1, \lceil \Re\rho_k \rfloor + 1, \lceil \Re\rho_k \rfloor - 2 \ldots$$

until upper and lower bounds are reached. With this ordering mechanism, the first lattice point checked by SE-SDA is the NC solution.

C. Lenstra, Lenstra, and Lovasz (LLL) Lattice Reduction

Without FA constraints, lattice reduction is an effective technique to reduce complexity of SDA (see [1] and the references therein). Since (1) can be easily converted to a real model, we consider the real case here. A lattice Λ can be generated by many matrices: Two matrices $\mathbf{H}$ and $\mathbf{H}_r$ generate the same lattice if and only if $\mathbf{H} = \mathbf{H}_r\mathbf{W}$, where $\mathbf{W}$ has integer entries with $\det(\mathbf{W}) = \pm 1$. The basis of Λ are the column vectors of its generator matrix. Among different bases, those with smaller norms are often preferred in terms of complexity [6, p. 83]. Given a generator matrix $\mathbf{H}$, the process of finding an $\mathbf{H}_r$ with basis vectors having small norms is called lattice reduction (LR). A well-known polynomial-time LR algorithm is the LLL algorithm [15]. In the following, we will briefly explain how the closest point search can benefit from the LLL algorithm.

Following [6] and [9], we apply the LLL algorithm to the row vectors of $\mathbf{R}^{-1}$ instead of $\mathbf{R}$ (the reason will become clear soon). With LR, we obtain $\mathbf{H}_r^{-1} = \mathbf{W}\mathbf{R}^{-1}$, or equivalently, $\mathbf{R} = \mathbf{H}_r\mathbf{W}$. Plugging to (2), we have

$$\begin{aligned}\|\mathbf{R}(\mathbf{s}-\hat{\mathbf{s}})\|^2 &= \|\mathbf{H}_r\mathbf{W}(\mathbf{s}-\hat{\mathbf{s}})\|^2 \\ &= \|\mathbf{H}_r(\mathbf{x}-\hat{\mathbf{x}})\|^2 = \|\mathbf{R}_r(\mathbf{x}-\hat{\mathbf{x}})\|^2 < r^2 \end{aligned} \quad (8)$$

where $\mathbf{x} := \mathbf{W}\mathbf{s}$, $\hat{\mathbf{x}} := \mathbf{W}\hat{\mathbf{s}}$, and $\mathbf{H}_r = \mathbf{Q}_r\mathbf{R}_r$. Clearly, without FA constraints, $\mathbf{s} \in \mathbb{Z}^N$ and $\mathbf{x} \in \mathbb{Z}^N$. Since $\mathbf{x} - \hat{\mathbf{x}} = \mathbf{R}_r^{-1}\mathbf{R}_r(\mathbf{x}-\hat{\mathbf{x}})$, we have $x_i - \hat{x}_i = \mathbf{R}_i^T\mathbf{R}_r(\mathbf{x}-\hat{\mathbf{x}})$, where x_i is the ith element of $\mathbf{x}$ and $\mathbf{r}_i^T$ denotes the ith row vector of $\mathbf{R}_r^{-1}$. Applying the Cauchy–Schwarz inequality, we have

$$(x_i - \hat{x}_i)^2 \le \|\mathbf{r}_i\|^2 \|\mathbf{R}_r(\mathbf{x}-\hat{\mathbf{x}})\|^2 \le \|\mathbf{r}_i\|^2 r^2. \quad (9)$$

Due to the LR on the row vectors of $\mathbf{R}^{-1}$, the rows of $\mathbf{H}_r^{-1}$ have small norm; i.e., the norm of $\mathbf{r}_i$ is small, which implies that the upper bound in (9) is reduced, and, thus, the number of candidates for x_i is reduced accordingly.

Unfortunately, LR does not facilitate incorporating FA constraints. To be specific, let $\mathbf{s} \in \mathcal{S}^N$, where $\mathcal{S}$ is the P-ary pulse amplitude modulation (PAM) constellation $\mathcal{S} := \{-P/2 + 1, \ldots, P/2\} \subset \mathbb{Z}$ and P is an even integer. Although checking the FA constraint on $\mathbf{s}$ is easy, the corresponding $\mathbf{s} = \mathbf{W}^{-1}\mathbf{x}$ for the candidate $\mathbf{x}$ after LR may not be in $\mathcal{S}^N$. Even though $\mathbf{W}^{-1}$ is an integer matrix, checking the FA constraint for $\mathbf{x}$ is much more difficult. The benefit of LR is further limited for certain constellations; for example, in the widely used 4-quadrature amplitude modulation (QAM), binary symbols are transmitted per dimension. Hence, we have a natural constraint from the constellation, which eliminates many nearby lattice points that are not in $\mathcal{S}^N$. On the other hand, any point in the integer lattice is initially considered to be valid by SDA, and many of them are checked unnecessarily. Nonetheless, when the constellation size is large, LR offers an effective means of reducing complexity.

III. SDA From a Probabilistic Perspective

Modeling $\mathbf{H}$ as a random matrix with independent and identically distributed (i.i.d.) zero-mean unit-variance complex Gaussian $\mathcal{CN}(0,1)$ entries is widely adopted in wireless communications [12]. In this section, we will exploit this statistical channel model to improve computational efficiency of the SDA. For the random matrix $\mathbf{H} = \mathbf{Q}\mathbf{R}$, the elements of the upper triangular matrix $\mathbf{R}$ are distributed independently according to $r_{i,i}^2 \sim \mathcal{G}(M+1-i)$ and $|r_{i,j}|^2 \sim \mathcal{G}(1)$ for $i < j$ [18]. Multiplying both sides of (1) with the unitary matrix $\mathbf{Q}^H$, we obtain

$$\tilde{\mathbf{y}} = \mathbf{R}\mathbf{s} + \tilde{\mathbf{n}} \quad (10)$$

where $\tilde{\mathbf{y}} := \mathbf{Q}^H\mathbf{y}$ and $\tilde{\mathbf{n}} \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$. To decode $\mathbf{s}$ from $\tilde{\mathbf{y}}$, a series of scalar decisions has to be made on set of equations [cf. (10)] given as

$$\check{y}_N = r_{N,N}s_N + \tilde{n}_N, \ldots, \check{y}_1 = r_{1,1}s_1 + \tilde{n}_1 \quad (11)$$

where $\check{y}_k := \tilde{y}_k - \sum_{i=k+1}^{N} r_{k,i}s_k$. Since $r_{i,i}$ is real and positive, a set of $2N$ equations can be obtained from (11) as

$$\Re\check{y}_N = r_{N,N}\Re s_N + \Re\tilde{n}_N, \quad \Im\check{y}_N = r_{N,N}\Im s_N + \Im\tilde{n}_N, \ldots. \quad (12)$$

Each decision in (12) corresponds to a branch in the search tree depicted in Fig. 1. Whenever an error occurs, the corresponding subtree is searched in vain. An early error corresponds to a high-dimensional subtree, which leads to a major waste of computations. Since $r_{i,i}^2 \sim \mathcal{G}(M+1-i)$, the probability that $r_{i,i}$ takes small values increases with i. Hence, it is more likely to make an error early in the decision sequence of (12). Unfortunately, with the single goal of efficiently reusing intermediate calculations, SDA adopts a depth-first search strategy, which does not account for this statistical property of $\mathbf{R}$.

With the random model, it follows from (11) that the SE-SDA examines the candidates of s_k in a decreasing probability order. Hence, the ordering mechanism of the SE-SDA improves search efficiency. Nonetheless, the distributions of $r_{N,N}, \ldots, r_{1,1}$ remain unaffected by this ordering, which indicates that further computational savings are possible.

Under our random model, the reason that LR improves the efficiency of SDA is as follows: Since the set of reduced basis vectors is not far from being orthogonal, the off-diagonal elements of $\mathbf{R}_r$ in (8) have relatively small magnitudes. This guarantees that the decision on one equation in (11) has less effect on other equations with LR. On the other hand, $\det(\mathbf{H})$ is the volume of the fundamental region for the lattice, which is invariant under different bases. This implies that the diagonal entries of $\mathbf{R}_r$ are relatively large in magnitude, which enhances SNR (and, thus, lowers the probability of error) for all the decisions in (11).

A. Detection Ordering

In this section, a detection ordering mechanism is examined as a method to improve the statistical properties of $\mathbf{R}$, which will, in turn, reduce the complexity of SDA. For a deterministic

$\mathbf{H}$, this ordering mechanism was mentioned in [9] based on a heuristic argument (see also [16]). Here, we provide a theoretical justification based on our random model of $\mathbf{H}$.

Detection ordering proceeds as follows: Rearranging the columns of $\mathbf{H} := [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_N]$ in an ascending order of their squared norm, i.e., $\|\mathbf{h}_{o(1)}\|^2 \leq \|\mathbf{h}_{o(2)}\|^2 \leq \cdots \leq \|\mathbf{h}_{o(N)}\|^2$, we obtain the ordered generator matrix as $\mathbf{H}_o := [\mathbf{h}_{o(1)}, \mathbf{h}_{o(2)}, \ldots, \mathbf{h}_{o(N)}]$. With the $\mathbf{QR}$ decomposition of $\mathbf{H}_o = \mathbf{Q}_o\mathbf{R}_o$, we are now ready to characterize the distribution of the entries of $\mathbf{R}_o$ denoted by $r_{o,i,j}$. Let $\mathbf{h}_{o(i)} := \sqrt{X_i}\boldsymbol{\theta}_i$, where $\boldsymbol{\theta}_i^H\boldsymbol{\theta}_i = 1$. Since $\{\mathbf{h}_i\}$s are i.i.d. with $\mathbf{h}_i \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$, we deduce that X_i is the ith-order statistic of N independent $\mathcal{G}(M)$ random variables such that $X_1 \leq X_2 \leq \cdots \leq X_N$. Because of this ordering, X_i's are necessarily dependent. However, $\{\boldsymbol{\theta}_i\}$s are i.i.d. uniformly distributed on the unit sphere in $\mathbb{R}^{2M}$ denoted by S^{2M-1}, where we have identified $\mathbb{C}^M$ with $\mathbb{R}^{2M}$ for simplicity. With this geometric interpretation, it is clear that the distribution of $\{\boldsymbol{\theta}_i\}$s remains invariant under orthogonal transformations. That is, for any unitary matrix $\mathbf{U}$, $p(\mathbf{U}^H\boldsymbol{\theta}) = p(\boldsymbol{\theta})$, where the probability density function (pdf) of $\boldsymbol{\theta}$ is $p(\boldsymbol{\theta}) = 1/A_{2M-1}$, $\boldsymbol{\theta}^H\boldsymbol{\theta} = 1$, and A_{2M-1} is the surface area of S^{2M-1} determined by

$$A_{2M-1} = \frac{2\pi^M}{\Gamma(M)}.$$

More generally, $\mathbf{Q}$ in the $\mathbf{QR}$ decomposition of $\mathbf{H}$ is uniformly (also known as isotropically) distributed over the Stiefel manifold $\{\mathbf{Q} : \mathbf{Q}^H\mathbf{Q} = \mathbf{I}\}$ [11]. The pdf of X_i is available in [4] as

$$f_{X_i}(x_i) = \frac{N!}{(i-1)!(N-i)!}\left[F(x_i)\right]^{i-1}\left[1 - F(x_i)\right]^{N-i} f(x_i)$$

where $F(x)$ and $f(x)$ are the cumulative density function (cdf) and pdf of a standard $\mathcal{G}(M)$ random variable, respectively. Letting $\mathbf{Q}_o := [\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_N]$ and using the $\mathbf{QR}$ decomposition of $\mathbf{H}_o$, we obtain

$$r_{o,i,i}^2 = X_i\left[1 - \sum_{k=1}^{i-1}\left(\mathbf{q}_k^H\boldsymbol{\theta}_i\right)^2\right] \equiv X_i\left[1 - \sum_{k=1}^{i-1}\boldsymbol{\theta}_i^2(k)\right]$$

where $\boldsymbol{\theta}_i(k)$ is the kth element of $\boldsymbol{\theta}_i$, and the equivalence follows from the fact that the distribution of $\boldsymbol{\theta}_i$ is invariant under the orthogonal transformation $\mathbf{Q}_o$. To determine the expectation of $r_{o,i,i}^2$, we first calculate $E[\boldsymbol{\theta}_i^2(k)]$. Since $\boldsymbol{\theta}_i^H\boldsymbol{\theta}_i = 1$, it follows that

$$E\left[\boldsymbol{\theta}_i^H\boldsymbol{\theta}_i\right] = E\left[\sum_{k=1}^{M}\boldsymbol{\theta}_i^2(k)\right] = ME\left[\boldsymbol{\theta}_i^2(k)\right] = 1$$

where the second equality follows from the symmetry of the distribution. Hence, $E[\boldsymbol{\theta}_i^2(k)] = 1/M$ for $k = 1, \ldots, M$. Due to the independence between X_i and $\boldsymbol{\theta}_i$, we have $E[r_{o,i,i}^2] = E[X_i][1 - (i-1)/M]$. The moments of the order statistics of i.i.d. Gamma random variables are available in closed form [4, p. 43]. Because the expressions are lengthy, exact formulas are omitted for brevity. Instead, we provide an illustrative example.

TABLE I
COMPARISON OF EXPECTATIONS FOR THE DIAGONAL ELEMENTS OF $\mathbf{R}$ FOR THE CASE $M = N = 16$

i	1	2	3	4	5	6	7	8
$E[r_{i,i}^2]$	16	15	14	13	12	11	10	9
$E[r_{o,i,i}^2]$	9.8	10.5	10.6	10.5	10.2	9.8	9.2	8.7
i	9	10	11	12	13	14	15	16
$E[r_{i,i}^2]$	8	7	6	5	4	3	2	1
$E[r_{o,i,i}^2]$	8.0	7.3	6.5	5.6	4.7	3.7	2.7	1.5

Example 1: Here, we consider $E[r_{i,i}^2]$ and $E[r_{o,i,i}^2]$ for $M = N = 16$. The mean values for the two cases are compared in Table I. Without ordering, it is clear that $E[r_{i,i}^2] = M + 1 - i$, since $r_{i,i}^2 \sim \mathcal{G}(M + 1 - i)$. With ordering, we observe from this table that $E[r_{o,i,i}^2]$ values are considerably larger than the corresponding $E[r_{i,i}^2]$ when i is close to M. Hence, the probability of an early error in the sequence of decisions in (11) is lower, which explains why complexity is reduced with ordering.

This detection ordering can be conveniently combined with SE-SDA. An attractive feature of the resulting algorithm is that the first lattice point checked is the NC solution with the received-symbol energy-based detection ordering, whereas the first vector examined by the original SE-SDA is the NC solution. NC with detection ordering is known to exhibit improved error performance as compared to conventional NC.

IV. PROBABILISTIC SEARCH ALGORITHM

Since SDA follows a depth-first search strategy, the order in which the lattice points are examined is determined by the recursive search tree structure. On the other hand, given the knowledge of a channel realization and the AWGN model, it is possible to predict where to search for the transmitted vector. If we divide the search space into small rectangular parallelepipeds, the probability of each parallelepiped containing the transmitted vector may differ considerably. Can we design an algorithm that examines these cells in a descending probabilistic order? The following algorithm provides such an approach.

A. Algorithm

Let us define $u_i := \lceil \Re(s_i - \rho_i) \rfloor$ and $\mu_i = \Re(s_i - \rho_i) - u_i$, where $u_i \in \mathbb{Z}$ and $\mu_i \in (-1/2, 1/2)$. Similarly, let $v_i := \lceil \Im(s_i - \rho_i) \rfloor$ and $\nu_i = \Im(s_i - \rho_i) - v_i$, where $v_i \in \mathbb{Z}$ and $\nu_i \in (-1/2, 1/2)$. With these definitions, (4) becomes

$$\|\mathbf{R}(\mathbf{s} - \hat{\mathbf{s}})\|^2 = \sum_{i=1}^{N} r_{i,i}^2(u_i + \mu_i)^2 + \sum_{i=1}^{N} r_{i,i}^2(v_i + \nu_i)^2$$

where the vector $\mathbf{c} := [u_1, v_1, u_2, v_2, \ldots, u_N, v_N]^T$ uniquely corresponds to a cell in the $2N$-dimensional space. Even though each cell contains $\mathbf{s}$ with positive probability, it may not yield a valid estimate satisfying the constellation constraint. Next, we determine a sufficient condition on $\mathbf{c}$ vectors so that finding

the desired closest point in the corresponding cells is guaranteed. Since $|\mu_i| < 1/2$ and $u_i \in \mathbb{Z}$, it follows that

$$\sum_{i=1}^{N} r_{i,i}^2 \mu_i^2 \leq \sum_{i=1}^{N} r_{i,i}^2 (u_i + \mu_i)^2. \tag{13}$$

Based on (13) and the triangle inequality

$$\sqrt{\sum_{i=1}^{N} r_{i,i}^2 u_i^2} \leq \sqrt{\sum_{i=1}^{N} r_{i,i}^2 \mu_i^2} + \sqrt{\sum_{i=1}^{N} r_{i,i}^2 (u_i + \mu_i)^2}$$

we conclude that

$$\sum_{i=1}^{N} r_{i,i}^2 u_i^2 \leq 4 \sum_{i=1}^{N} r_{i,i}^2 (u_i + \mu_i)^2.$$

Similar inequalities hold for v_i and ν_i. Hence, if we examine all $\mathbf{c}$ vectors satisfying

$$Q(\mathbf{c}) := \sum_{i=1}^{N} r_{i,i}^2 \left[u_i^2 + v_i^2\right] \leq 4 e_{\min}^2 \tag{14}$$

we are guaranteed to find the closest point and the corresponding minimum Euclidean distance $e_{\min}$. That is, the candidates depend on the true minimum distance albeit through a bound. Furthermore, prior knowledge of $e_{\min}$ is not necessary.

The new algorithm breaks the search tree structure of SDA and follows the following steps:

S1: initialize $e_{\min}^2 = \infty$;
S2: generate a vector $\mathbf{c}$ such that $Q(\mathbf{c})$ is the next smallest;
S3: if $Q(\mathbf{c}) > 4e_{\min}^2$, go to **S6**;
S4: check the FA constraint on the $\mathbf{s}$ corresponding to $\mathbf{c}$. If the $\mathbf{s}$ is invalid, go to **S2**;
S5: calculate $\|\mathbf{R}(\mathbf{s} - \hat{\mathbf{s}})\|^2$, update

$$e_{\min}^2 = \min\left\{e_{\min}^2, \|\mathbf{R}(\mathbf{s} - \hat{\mathbf{s}})\|^2\right\}, \text{ and go to } \mathbf{S2}.$$

S6: return the best candidate for $\mathbf{s}$ and $e_{\min}$.

Several remarks are in order to clarify salient features of this algorithm.

R1: The initial candidate checked in every search is always the integer NC solution, which corresponds to $\mathbf{c} = \mathbf{0}$ and accordingly $Q(\mathbf{0}) = 0$; yet, it may not satisfy the constellation constraint.
R2: We emphasize that prior knowledge of $e_{\min}$ is not required. In S2, candidates with increasing $Q(\mathbf{c})$ are generated; whereas in S5, $e_{\min}$ is decreased. When the inequality in S3 is satisfied, we obtain the true $e_{\min}$. Even though the initial value of $e_{\min}$ is infinity, the number of candidates is bounded only by the final value of $e_{\min}$.
R3: The list of $\mathbf{c}$ vectors is generated in an ascending order of $Q(\mathbf{c})$. This list determines the order in which cells are examined and is shared by all decodings with the same $\mathbf{H}$ realization. An efficient way to generate such a list is described in the Appendix.

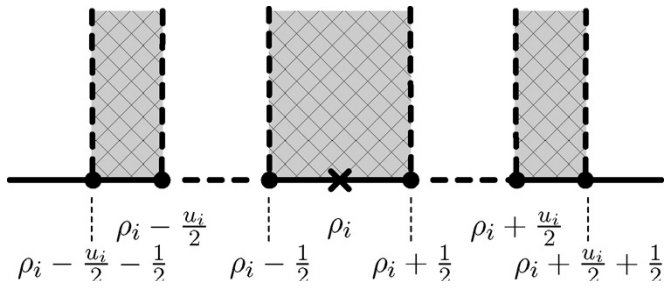

Fig. 2. Approximate probability of u_i given $\hat{s}_i$.

B. Probabilistic Ordering

We offer a probabilistic justification of the novel decoding algorithm. Letting $\Pr(\mathbf{c})$ be the probability that the cell corresponding to $\mathbf{c}$ contains the transmitted vector $\mathbf{s}$, we determine $\Pr(\mathbf{c})$ based on the channel realization $\mathbf{H}$ and the AWGN model $\mathbf{v} \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$.

To calculate $\Pr(\mathbf{c})$, we start with the scalar case. Based on the definition of ρ_i in (5) and the ith equation from (11), we have $\rho_i = \check{y}_i / r_{i,i}$ and $\rho_i = s_i + \check{n}_i$, where $\check{n}_i := \tilde{n}_i / r_{i,i}$ and $\check{n}_i \sim \mathcal{CN}(0, \sigma^2 / r_{i,i}^2)$. Equivalently, $s_i = \rho_i + \check{n}_i$, since $-\check{n}_i$ has the same distribution as $\check{n}_i$. Separating real from imaginary parts, the probability of u_i is the probability that $\Re s_i \in [\Re\rho_i + u_i - 0.5, \Re\rho_i + u_i + 0.5]$ as illustrated in Fig. 2. This probability can be calculated as

$$\begin{aligned}
\Pr(u_i) &= 2\exp\left(-\frac{r_{i,i}^2 u_i^2}{\sigma^2}\right) \cdot \int_0^{\frac{1}{2}} \frac{1}{\sqrt{\frac{\pi\sigma^2}{r_{i,i}^2}}} \\
&\quad \times \exp\left(-\frac{r_{i,i}^2 x^2}{\sigma^2}\right) \cosh\left(\frac{2u_i}{\frac{\sigma^2}{r_{i,i}^2}} x\right) dx \\
&\approx 2\exp\left(-\frac{r_{i,i}^2 u_i^2}{\sigma^2}\right) \int_0^{\frac{r_{i,i}}{\sqrt{2}\sigma}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) dx \\
&:= k_i \exp\left(-\frac{r_{i,i}^2 u_i^2}{\sigma^2}\right)
\end{aligned} \tag{15}$$

where we have applied a first-order approximation for $\cosh(x) = 1 + x^2/2 + x^4/4! + \cdots$. This approximation is reasonable, since if u_i is approximately zero, then $\cosh(2u_i r_{i,i}^2 x/\sigma^2)$ is close to 1; otherwise, $\exp(-r_{i,i}^2 u_i^2/\sigma^2)$ becomes dominant. With this approximation, we observe that $\Pr(u_i)$ depends on $r_{i,i}^2 u_i^2$ only. Since $\check{n}_i$'s are independent, the probability of $\mathbf{c}$ can be approximated by[1]

$$\Pr(\mathbf{c}) \approx \exp\left(-\frac{\sum_{i=1}^{N} r_{i,i}^2 \left[u_i^2 + v_i^2\right]}{\sigma^2}\right) \prod_{i=1}^{N} k_i^2. \tag{16}$$

Hence, $\Pr(\mathbf{c})$ is approximately determined by $Q(\mathbf{c})$. Geometrically, there is a parallelepiped associated with each $\mathbf{c}$, and $\Pr(\mathbf{c})$

[1]Here, we emphasize that $\mathbf{c}$ vectors are enumerated based on channel knowledge. They are not random variables.

is the probability that AWGN carries the transmitted $\mathbf{s}$ into this parallelepiped. The proposed algorithm examines $\mathbf{c}$ with an increasing order of $Q(\mathbf{c})$, which translates to checking cells containing $\mathbf{s}$ in a decreasing probability order. This explains why we term the novel closest point scheme a probabilistic search algorithm.

C. Complexity Analysis

In this subsection, we evaluate the average complexity of our probabilistic search algorithm. The decoding complexity is averaged over both the channel and AWGN realizations. Only the case without detection ordering will be considered.

We adopt the average number of $\mathbf{c}$ vectors examined by the probabilistic search algorithm as the indicator of average complexity. Since the number of $\mathbf{c}$ vectors is limited by the minimum squared Euclidean distance $e_{\min}^2$ and $e_{\min}^2 \leq \|\mathbf{v}\|^2$, we relax the constraint in S3 of this algorithm from $Q(\mathbf{c}) < e_{\min}^2$ to $Q(\mathbf{c}) < \|\mathbf{v}\|^2$. Since $e_{\min}$ equals $\|\mathbf{v}\|$ most of the time, the effect of this relaxation is negligible. We determine the average probability that $\mathbf{c} = [u_1, v_1, u_2, v_2, \ldots, u_N, v_N]^T$ needs to be checked next and denote this probability by $P_{\mathbf{c}}$. To find $P_{\mathbf{c}}$, we first calculate the conditional probability given as

$$P_{\mathbf{c}|V} := \Pr\left(\sum_{i=1}^{N} r_{i,i}^2 \left(u_i^2 + v_i^2\right) \leq 4v\right)$$

where $V = \|\mathbf{v}\|^2$ and $r_{i,i}^2 \sim \mathcal{G}(M+1-i)$ are independent of each other. To derive a closed-form expression for $P_{\mathbf{c}|V}$, we introduce the following two lemmas. The first one is a well-known property of Gamma variables.

Lemma 1: If $G_1 \sim \mathcal{G}(n_1)$ is independent of $G_2 \sim \mathcal{G}(n_2)$, then $G = G_1 + G_2 \sim \mathcal{G}(n_1 + n_2)$.

Lemma 2: Let $\{G_i\}_{i=1}^K$ be independently distributed according to $\mathcal{G}(n_i)$ with $n_i \in \mathbb{N}$. If $l_i > 0$ and $l_i \neq l_j$, then the pdf of $Y = \sum_{i=1}^K l_i G_i$ is given by

$$f_Y(y) = \sum_{k=1}^{K} \sum_{\sum_{q=1}^K m_q = n_k - 1} C(k, \mathbf{m}) \frac{l_k}{\Gamma(m_k+1)} \left(\frac{y}{l_k}\right)^{m_k} e^{-\frac{y}{l_k}}$$

where $\mathbf{m} := [m_1, \ldots, m_K]^T$ and

$$C(k, \mathbf{m}) := (-1)^{n_k - 1 - m_k} \cdot \prod_{\substack{q=1 \\ q \neq k}}^{K} \binom{n_q + m_q - 1}{m_q} \times \left(\frac{l_q}{l_k}\right)^{m_q} \left(1 - \frac{l_q}{l_k}\right)^{-(n_q + m_q)}.$$

Proof: Using the Laplace transform of Y

$$\mathcal{L}_r\left(f(y)\right) = \frac{1}{(1 + l_1 r)^{n_1}} \frac{1}{(1 + l_2 r)^{n_2}} \cdots \frac{1}{(1 + l_K r)^{n_K}}$$

$f_Y(y)$ can be uniquely determined by the inverse Laplace transform as

$$f(y) = \frac{1}{2\pi i} \int_{c - i\infty}^{c + i\infty} \mathcal{L}_r\left(f(y)\right) e^{ry}\, dr = \sum_{k=1}^{K} R_k$$

where R_k's are the K residues of $\mathcal{L}_r(f(y))$ that are calculated as

$$R_k = \frac{1}{\Gamma(n_k)} \frac{d^{n_k - 1}}{dr^{n_k - 1}} \left[(1 + l_k r)^{n_k} \mathcal{L}_r\left(f(y)\right) e^{ry}\right]\Big|_{r = -\frac{1}{l_k}}.$$

The lemma follows by applying the multinomial formula to the derivatives. ■

Based on Lemma 1, we combine those $r_{i,i}^2$'s with equal coefficients into one Gamma variable. Let the resulting K variables and their coefficients be G_i and l_i, where $G_i \sim \mathcal{G}(n_i)$ are all independent. Then, the closed-form expression for $P_{\mathbf{c}|V}$ is

$$\begin{aligned} P_{\mathbf{c}|V} &= \Pr\left(\sum_{i=1}^{K} l_i G_i \leq 4v\right) \\ &= \sum_{k=1}^{K} \sum_{\mathbf{m}} C(k, \mathbf{m}) \left[1 - e^{\frac{-4v}{l_k}} \sum_{p=0}^{m_k} \frac{\left(\frac{4v}{l_k}\right)^{m_k - p}}{\Gamma(m_k + 1 - p)}\right] \end{aligned}$$

where we have applied Lemma 2 and the identity

$$\frac{1}{\Gamma(n+1)} \int_0^x t^n e^{-t}\, dt = 1 - e^{-x} \sum_{k=0}^{n} \frac{x^{n-k}}{\Gamma(n+1-k)}.$$

The next step is to evaluate $P_{\mathbf{c}}$. Since $V = \|\mathbf{v}\|^2$, the pdf of V is

$$p_V(v) = \frac{1}{\sigma^2 \Gamma(M)} \left(\frac{v}{\sigma^2}\right)^{M-1} e^{-\frac{v}{\sigma^2}}, \quad v \geq 0.$$

Therefore, $P_{\mathbf{c}}$ can be calculated as

$$P_{\mathbf{c}} = c_1 - \sum_{k=1}^{K} \sum_{\mathbf{m}} \sum_{p=0}^{m_k} D(k, \mathbf{m}, p) \left[\frac{4}{l_k} + \frac{1}{\sigma^2}\right]^{-(M + m_k - p)}$$

where

$$c_1 := \sum_{k=1}^{K} \sum_{\mathbf{m}} C(k, \mathbf{m}) \quad \text{and}$$

$$D(k, \mathbf{m}, p) := C(k, \mathbf{m}) \binom{M + m_k - p - 1}{m_k - p} \frac{4^{m_k - p}}{l_k^{m_k - p} \sigma^{2M}}.$$

Finally, the average number of $\mathbf{c}$ vectors examined by the algorithm is $N_{\text{av}} = \sum_{\mathbf{c}} P_{\mathbf{c}}$, where the summation involves enumerating all possible $\mathbf{c}$ vectors. Nonetheless, it can be inferred

from (16) that $P_{\mathbf{c}}$ decreases exponentially with $Q(\mathbf{c})$. Hence, a finite number of $\mathbf{c}$ is sufficient to provide a good approximation to $N_{\rm av}$.

V. Layered Search Algorithm

Even though NC with detection ordering requires low decoding complexity, its error performance is rather poor. On the other hand, SDA offers ML or near-ML error performance, yet, the decoding complexity is often too high to be practical. There are sizable gaps both in terms of error performance and complexity between these two extremes, which motivates us to develop suboptimal algorithms. Specifically, our goal is to design approximate algorithms with considerably better error performance and average decoding complexity comparable to the NC algorithm with detection ordering.

In this section, we will classify dimensions based on channel realizations, develop an error-performance-oriented fast stopping criterion, and describe our layered search algorithm. The main idea behind the layered search decoding algorithm is as follows. With the probabilistic search, we examine potential candidates in a descending likelihood order. Whenever a reasonable candidate satisfying the fast stopping criterion is found, we terminate the search.

A. Classifying Dimensions

We consider the following set of N complex scalar models

$$s_N = \rho_N + \check{n}_N, \ldots, s_1 = \rho_1 + \check{n}_1$$

where ρ_i is defined in (5) and $\check{n}_i \sim \mathcal{CN}(0, \sigma^2/r_{i,i}^2)$. Based on channel realizations $r_{i,i}$, we classify the N dimensions into three groups. If $r_{i,i}^2$ is greater than a certain threshold, $r_{i,i}^2 \geq T_U$, then we consider the dimension associated with $r_{i,i}^2$ reliable. For a reliable dimension and the model $s_i = \rho_i + \check{n}_i$, the most likely estimate $\lceil \rho_i \rfloor$ for s_i is correct with high probability. In the suboptimal algorithm design, we assume that this probability equals 1. That is, if all previous decisions on $s_N, \ldots, s_{i+1}$ are correct, the decision on s_i will introduce no error. When applying this idea to the probabilistic search, we can set those $\mathbf{c}$ entries corresponding to reliable dimensions to zero. The remaining question is how we determine T_U. We calculate T_U based on the real model $\Re s_i = \Re\rho_i + \Re\check{n}_i$, where $\Re\check{n}_i \sim \mathcal{N}(0, \sigma^2/(2r_{i,i}^2))$. The threshold T_U is the smallest $r_{i,i}^2$ such that $\lceil \Re\rho_i \rfloor$ equals $\Re s_i$ with probability $1 - P_U$. Explicitly, $T_U = 2\sigma^2[\mathrm{Erfcinv}(P_U)]^2$, where $\mathrm{Erfcinv}(x)$ is the inverse function $\mathrm{Erfc}(x) := (2/\sqrt{\pi}) \int_x^\infty \exp(-t^2)\, dt$ and is available in Matlab. The probability P_U is often close to 0 and should be determined based on the required error performance.

The second threshold T_L is used to determine when to exploit the FA property. For a small alphabet size per dimension, the FA property of a constellation can be very helpful in reducing search complexity. Without ordering the columns of $\mathbf{H}$, we have $r_{i,i}^2 \sim \mathcal{G}(M + 1 - i)$, $1 \leq i \leq N$, from which we can infer that some realizations of $r_{i,i}^2$ are likely to be small. Even though ordering can bring improvements, it is still possible that some $r_{o,i,i}^2$'s are small. For small $r_{i,i}^2$, the variance of $\check{n}_k$ in $s_k = \rho_k + \check{n}_k$ is considerably magnified, since $\check{n}_k \sim \mathcal{CN}(0, \sigma^2/r_{i,i}^2)$. As a result, the number of possible choices for u_i and v_i could be quite large [cf. (14)]. However, many of them are invalid constellation points. On the other hand, it is straightforward to determine the probability order for all s_k candidates per dimension, which has been employed in SE-SDA. The threshold T_L decides when to exploit the FA property, and when to pursue the probabilistic search. That is, when $r_{k,k}^2 \geq T_L$, we employ the probabilistic search for the kth dimension, i.e., the kth dimension is included in the summation $\sum_i r_{i,i}^2(u_i^2 + v_i^2)$; otherwise, we omit that dimension and exploit the FA property. Similar to T_U, the threshold T_L corresponds to the smallest $r_{k,k}^2$ such that $\Pr(\Re s_k \in [\Re\rho_k - P/2, \Re\rho_k + P/2]) = 1 - P_L$, where the even integer P is the constellation size of $\Re s_k$ and a suitable value of P_L is 0.01.

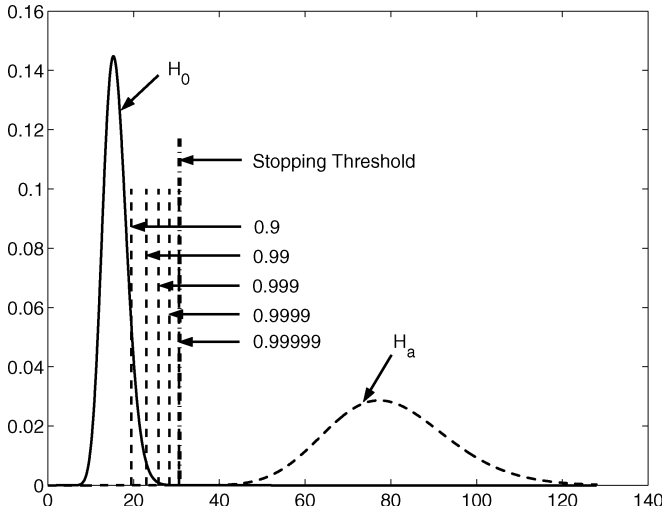

Fig. 3. Hypothesis testing for $M = N = 32$ at SNR $= 24$ dB.

B. Error-Performance-Oriented Fast Stopping Criterion

In the exact probabilistic search, the $\mathbf{c}$ vectors are examined in a descending probability order, which maximizes the chance to find the closest point early. However, even when the solution has been found, the algorithm continues until the bound in (14) is achieved, thus performing unnecessary computations. Certainly, an efficient algorithm should find the closest point as quickly as possible, and stop as fast as possible. Hence, a fast stopping criterion with controllable performance degradation is indispensable for an efficient approximate algorithm.

We develop such a criterion based on binary hypothesis testing next. Whenever a candidate $\mathbf{s}'$ for the transmitted vector $\mathbf{s}$ is found, we test the corresponding distance V. Let the null hypothesis H_0 be $V = \|\mathbf{v}\|^2$, where $\mathbf{v}$ is the true noise vector in (1). This is the case when $\mathbf{s}' = \mathbf{s}$. Let the alternative hypothesis H_a be $V = \|\mathbf{v} + \mathbf{h}\|^2$, where $\mathbf{h} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$. The alternative occurs when $\mathbf{s}'$ differs from $\mathbf{s}$ in only one position, and the difference between the two corresponding elements may be either ± 1 or $\pm j$. Only such an error pattern is considered, because it is the dominant one at high SNR. Let the testing threshold be $V_{\rm th}$. Whenever $V < V_{\rm th}$, we decide that H_0 is true and stop searching; otherwise, we continue to search. Since $\mathbf{v}$ and $\mathbf{h}$ are independent, it follows that $V \sim \mathcal{G}(M, \sigma^2)$ under

H_0, while $V \sim \mathcal{G}(M, \sigma^2+1)$ under H_a. We illustrate how we determine $V_{\rm th}$ through the following example.

Example 2: Let us consider dimensions $M = N = 32$ corresponding to either a large BLAST-type space–time system, or to a symbol synchronous CDMA system. The modulation employed is 16-QAM, and the system operates at receive SNR $= 24$ dB, which corresponds to 5.9382 dB in terms of transmit SNR. Receive SNR seems to be preferred in space–time settings; whereas transmit SNR is often used in multiuser scenario. In a space–time setting, 24 dB belongs to the high-SNR regime, whereas in multiuser detection, 5.9382 dB is at best in the medium-SNR range. The density functions for V under H_0 and H_a are depicted in Fig. 3. Since $V \sim \mathcal{G}(N, \sigma^2+1)$ under H_a, the probability of missing with the threshold $V_{\rm th}$ can be calculated in closed form as

$$P_M(V_{\rm th}) = 1 - e^{-\frac{V_{\rm th}}{(1+\sigma^2)}} \sum_{k=0}^{N-1} \frac{1}{k!} \left(\frac{V_{\rm th}}{1+\sigma^2} \right)^k$$

which equals the probability that a pairwise error happens, but remains undetected. We can consider P_M as an upperbound of the error performance penalty payed for fast stopping. Continuing with our example, let the target P_M be 10^{-6}. P_M can be determined either from the target symbol error rate (SER) or the actual SER. For a certain P_M, the corresponding $V_{\rm th}$ can be calculated by standard toolboxes, e.g., Mathematica. In this case, $V_{\rm th} = 29.6099$ for $P_M = 10^{-6}$. One way to evaluate the effectiveness of the stopping threshold $V_{\rm th}$ is by calculating $P_{in} := \Pr(\|\mathbf{v}\|^2 < V_{\rm th})$. For this example, $P_{in} = 0.999971$. Hence, the fast stopping criterion is effective for almost all noise realizations at this SNR level.

C. Algorithm Description

An apparent drawback of the probabilistic search algorithm is the overhead of generating an ordered list of $\mathbf{c}$ vectors, which requires additional computations and/or memory. To eliminate this overhead, we introduce a suboptimal layered search algorithm. In this algorithm, $\mathbf{c}$ vectors are examined in an approximate decreasing likelihood order.

As in the probabilistic search, the first vector examined is $\mathbf{c} = \mathbf{0}$. After that, the first tier of $\mathbf{c}$ is searched in the following order $[0, \ldots, 0, 0, 1]$, $[0, \ldots, 0, 1, 0]$, …, and $[1, 0, \ldots, 0, 0]$. The second tier includes the following vectors:

$$[0, \ldots, 0, 0, 1, 1], [0, \ldots, 0, 1, 0, 1], \ldots, [1, 0, \ldots, 0, 1];$$
$$[0, \ldots, 0, 0, 1, 1, 0], [0, \ldots, 0, 1, 0, 1, 0], \ldots, [1, 0, \ldots, 0, 1, 0];$$
$$\vdots$$
$$[1, 1, 0, \ldots, 0].$$

More tiers of $\mathbf{c}$ vectors can be further identified in a similar fashion. The advantage of generating $\mathbf{c}$ vectors in a layered structure is that iterations can be conveniently used to enumerate $\mathbf{c}$ vectors. Hence, precalculating the $\mathbf{c}$ list is circumvented. Furthermore, the classification of dimensions discussed in Section V-A can be straightforwardly combined with the enumeration.

The description of our layered search algorithm is as follows.

1) Order the columns of $\mathbf{H}$ and the entries of $\hat{\mathbf{s}}$ as described in Section III-A.
2) Calculate T_L and T_U, and determine reliable and FA dimensions as in Section V-A.
3) Compute the stopping threshold $V_{\rm th}$ as in Section V-B.
4) Determine a search radius upperbound r_U and set $e_{\min} = r_U$.
5) If there is no more $\mathbf{c}$ vector in tiers to enumerate, go to 8); else, generate the next $\mathbf{c}$.
6) Examine the current $\mathbf{c}$ within distance $e_{\min}$ under FA constraint.
7) If a valid candidate for $\mathbf{s}$ is within $V_{\rm th}$, go to 9); else, update $e_{\min}$ and go to 5).
8) If no valid $\mathbf{s}$ candidate is found, use $\mathbf{s}_{\rm NC}$ as the estimate.
9) Invert the entry ordering in 1) and return the best candidate.

The purpose of the search radius upperbound r_U in 4) is to limit the search within some reasonable distance. Without r_U, it is possible to find some valid, yet, unlikely candidates of $\mathbf{s}$ before finding a nearby candidate, which incurs unnecessary computations. The upperbound r_U is determined from the AWGN model as

$$\int_0^{r_U} f_V(v)\, dv = 1 - P_B$$

where $f_V(v)$ is the pdf of $V = \|\mathbf{v}\|^2 \sim \mathcal{G}(N, \sigma^2)$ and P_B is negligible when compared to the desired symbol error probability.

VI. Simulations

The complexity reduction brought by detection ordering and the suboptimal layered search algorithm are tested via Monte Carlo simulations in this section. The complexity is measured in terms of average number of floating point operations (flops). We treat real additions, multiplications, and comparisons equally as flops.

Test Case 1: To illustrate the average decoding complexity reduction brought by detection ordering, we consider a system with dimension $M = N = 16$ and 16-QAM constellation. We have modified the SE-SDA available from [1] and incorporated both the increasing radius search from [12] and the FA constraints. In the simulation, we generate 100 noise realizations per channel realization, and at least 10 000 channel realizations for each SNR value. With or without ordering, SE-SDA achieves the same SER performance depicted in Fig. 4. However, the average complexity differs considerably. Let the average complexity exponent be $\log_{2N}(N_{\rm flop})$, where $N_{\rm flop}$ is the average number of flops per decoding. The complexity exponents for both cases are plotted

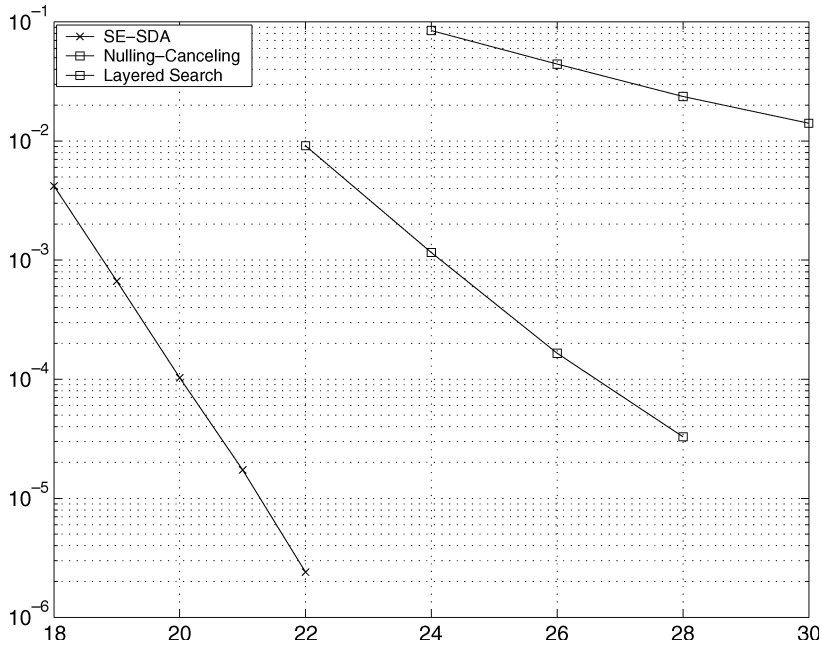

Fig. 4. SER comparison for SE-SDA, layered search, and NC for $M = N = 16$.

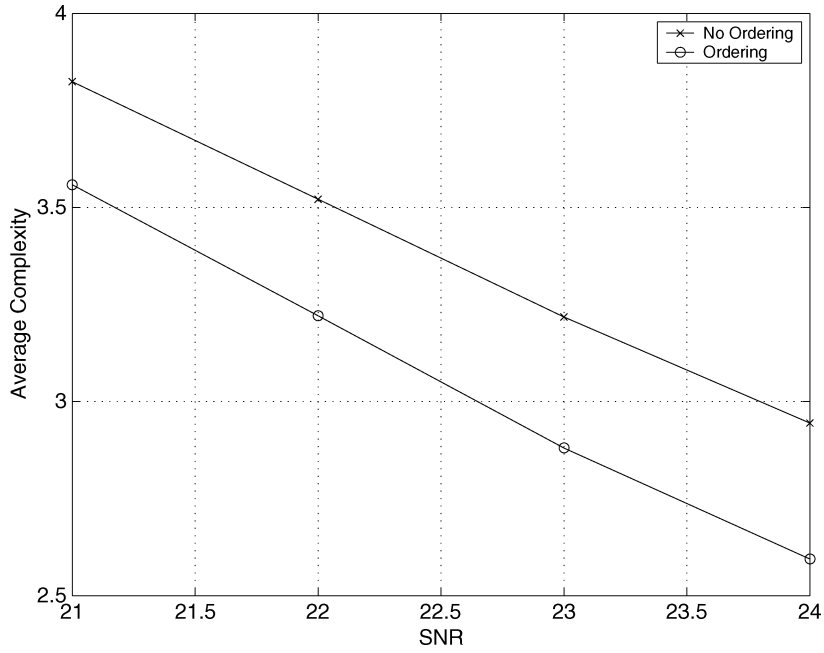

Fig. 5. Average complexity exponent comparison for $M = N = 16$.

in Fig. 5. It is evident that detection ordering is a simple, yet, effective technique to improve SDA. Furthermore, detection ordering needs to be performed only once per channel realization.

Test Case 2: We evaluate the error performance and average decoding complexity of the layered search algorithm. We consider a system with $M = N = 32$ and 4-QAM modulation. Hence, the equivalent real model has 64 dimensions and 2-PAM constellation, which is often encountered in symbol synchronous CDMA detection. As described in Section V, parameters T_L, T_U, r_U, and V_{th} of the layered search are specified through the corresponding probabilities P_L, P_U, P_B, and P_M, respectively. Here, we have fixed $P_L = 10^{-2}$. The remaining parameters are reported in Table II. In the simulation, the decoding complexity is averaged over at least 10 000

TABLE II
PARAMETERS FOR THE LAYERED SEARCH AND AVERAGE COMPLEXITY COMPARISON FOR $M = N = 32$ AND 4-QAM

SNR	18	21	24	27
P_U	$10^{-3.5}$	10^{-4}	10^{-5}	10^{-6}
P_M	10^{-4}	10^{-5}	10^{-6}	10^{-7}
P_B	10^{-5}	10^{-6}	10^{-7}	10^{-8}
LS	5177	4532	4488	4483
NC	4495	4495	4495	4495

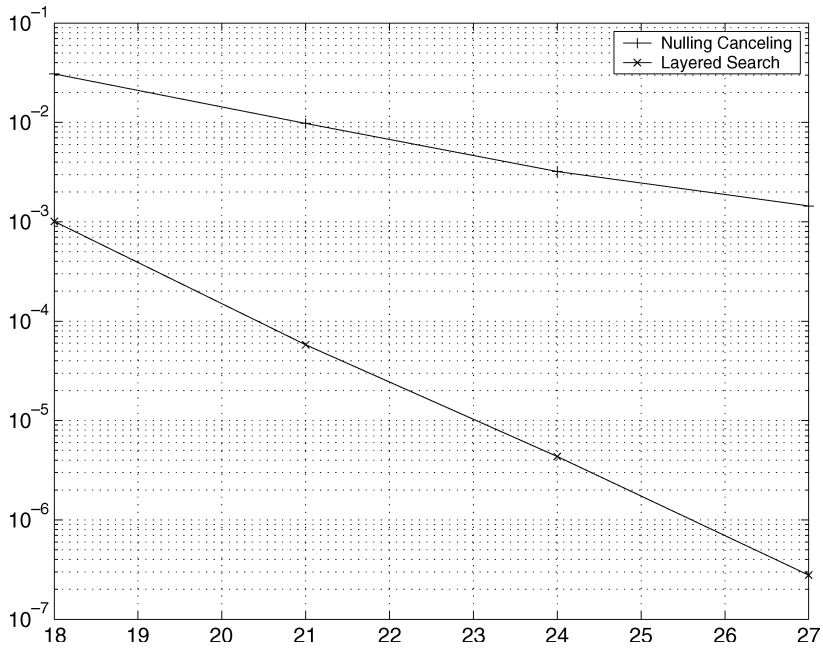

Fig. 6. Symbol error performance comparison between layered search and NC with detection ordering.

channel realizations and 100 AWGN realizations per channel. The average number of flops per decoding for NC and layered search are provided in the last two rows of Table II. It can be observed that similar complexities are achieved by both algorithms. The error performance results of these algorithms are depicted in Fig. 6. Remarkably, with similar complexity, the layered search achieves considerably better error performance than the NC algorithm.

Test Case 3: As another example, we demonstrate the effectiveness of the layered search algorithm for $M = N = 16$ and 16-QAM constellation. Parameters are determined from the probabilities in Table III. Average numbers of flops per decoding for the layered search and the NC algorithm are reported in the last two rows of this table. Symbol error performance is depicted in Fig. 4. With complexity similar to NC, the layered search fills in the large error performance gap between these two extremes. However, the layered search becomes less effective with decreasing SNR values, which is indicated by the number of flops at 22 dB in Table III. For low and medium SNR, it becomes difficult to find a good estimate fast, and reusing intermediate computations as in SDA is important.

TABLE III
PARAMETERS FOR THE LAYERED SEARCH AND AVERAGE COMPLEXITY COMPARISON FOR $M = N = 16$ AND 16-QAM

SNR	22	24	26	28
P_U	10^{-2}	10^{-3}	10^{-4}	10^{-5}
P_M	10^{-2}	10^{-3}	10^{-4}	10^{-5}
P_B	10^{-4}	10^{-5}	10^{-6}	10^{-7}
LS	5298	1832	1344	1251
NC	1235	1235	1235	1235

VII. CONCLUSION

The conventional SDA was studied under a random channel model commonly encountered in wireless communications. It was observed that the depth-first search strategy of SDA dominates the order in which candidates of $\mathbf{s}$ are examined, which is opposite to what the distributions of $r_{i,i}^2$ suggest. With both theoretical analysis and simulations, we justified that detection ordering offers a simple but effective means to improve the search efficiency.

A novel ML probabilistic search algorithm was derived and justified on probability grounds. A theoretical average complexity analysis was also provided. This algorithm enjoys two main attractive features, namely: 1) potential candidates of the transmitted vector are examined in a descending probability order, which breaks the search tree structure of SDA; and 2) the number of possible candidates is bounded by the actual minimum Euclidean distance.

Even though the probabilistic search initiates a new direction in decoding algorithm design, it is only efficient in the high–SNR regime. Nonetheless, it can be conveniently adopted in suboptimal decoding algorithms. We also developed an efficient layered search algorithm equipped with an error-performance-oriented fast stopping criterion. Our design intuition is rather simple: trying to find a good candidate early and stop fast. Furthermore, by generating search patterns in a layered fashion from iterations, we eliminated the overhead in the probabilistic search. With decoding complexity comparable to the NC algorithm with detection ordering, simulations confirmed that our layered search improves error performance considerably.

APPENDIX

We describe an algorithm to generate a $\mathbf{c}$ vector list in an ascending order of $Q(\mathbf{c})$. Letting $B = \{\mathbf{c}, Q(\mathbf{c})\}$ be a data structure, our algorithm will return a list of B objects with ascending $Q(\mathbf{c})$. We denote the head and tail of this list by h

and t, respectively. This algorithm recursively updates the list as follows:

List($r_{1,1}^2, \ldots, r_{n,n}^2, C$)
S1: $h \leftarrow \{\mathbf{0}, 0\}$;
S2: for $i = n : 1$
S3: find $u_{\max}$, the largest u such that $r_{i,i}^2 u^2 < C$;
S4: for $j = 1 : u_{\max}$
S5: for $p = h : t$
S6: $\mathbf{c} = p \rightarrow \mathbf{c}$;
S7: set $c_i = j$;
S8: $Q(\mathbf{c}) = p \rightarrow Q(\mathbf{c}) + j^2 r_{i,i}^2$;
S9: if $Q(\mathbf{c}) < C$
SA: insert $\{\mathbf{c}, Q(\mathbf{c})\}$ in the ordered list;
SB: end;
SC: end;
SD: end;
SE: end;
SF: return h.

REFERENCES

[1] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inf. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[2] L. Babai, "On Lovasz' lattice reduction and the nearest lattice point problem," *Combinatorica*, vol. 6, no. 1, pp. 1–13, 1986.

[3] L. Brunel and J. Boutros, "Euclidean space lattice decoding for joint detection in CDMA systems," in *Proc. Information Theory Workshop*, Kruger National Park, South Africa, Jun. 1999, p. 129.

[4] N. Balakrishnan and A. C. Cohen, *Order Statistics and Inference Estimation Methods*. New York: Academic, 1991.

[5] A. Chan and I. Lee, "A new reduced-complexity sphere decoder for multiple antenna systems," in *Proc. Int. Conf. Communications*, New York, Apr. 28–May 2, 2002, vol. 1, pp. 460–464.

[6] H. Cohen, *A Course in Computational Algebraic Number Theory*. Berlin, Germany: Springer-Verlag, 1993.

[7] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices, and Groups*, 3rd ed. New York: Springer-Verlag, 1999.

[8] M. O. Damen, H. El Gamal, and G. Caire, "On maximum likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.

[9] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Math. Comput.*, vol. 44, no. 170, pp. 463–471, Apr. 1985.

[10] G. J. Foschini, "Layered space–time architecture for wireless communication in a fading environment when using multielement antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–49, Autumn 1996.

[11] A. K. Gupta and D. K. Nagar, *Matrix Variate Distributions*. Boca Raton, FL: CRC, 2000.

[12] B. Hassibi and H. Vikalo, "On the expected complexity of sphere decoding," in *Proc. 35th Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2001, pp. 1051–1055.

[13] 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.

[14] R. Kannan, "Improved algorithms for integer programming and related lattice problems," in *Proc. 15th Annu. ACM Symp. Theory Computing*, Boston, MA, Apr. 1983, pp. 193–206.

[15] A. K. Lenstra, H. W. Lenstra, and L. Lovasz, "Factoring polynomials with rational coefficients," *Math. Ann.*, vol. 261, no. 4, pp. 513–534, 1982.

[16] J. Luo, K. Pattipati, P. Willett, and L. Brunel, "Branch-and-bound-based fast optimal algorithm for multiuser detection in synchronous CDMA," in *Proc. Int. Conf. Communications*, Anchorage, AK, May 11–15, 2003, pp. 3336–3340.

[17] X. Ma and G. B. Giannakis, "Full-diversity full-rate complex-field space–time coding," *IEEE Trans. Signal Process.*, vol. 51, no. 11, pp. 2917–2930, Nov. 2003.

[18] R. J. Muirhead, *Aspects of Multivariate Statistical Theory*. New York: Wiley, 1982.

[19] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1639–1642, Jul. 1999.

[20] Y. Xin, Z. Wang, and G. B. Giannakis, "Space–time diversity systems based on linear constellation precoding," *IEEE Trans. Wireless Commun.*, vol. 2, no. 2, pp. 294–309, Mar. 2003.

Wanlun Zhao (S'02) received the B.S. degree in electrical engineering and information science from Huazhong University of Science and Technology, Wuhan, China, in 1996 and the M.Sc. degree in applied and computational mathematics from the University of Minnesota, Minneapolis, in 2000. He is currently working toward the Ph.D. degree in electrical engineering at the University of Minnesota.

At the University of Minnesota, he works with Prof. G. Giannakis in the areas of wireless communications, coding techniques, and cross-layer designs for wireless networks.

Georgios B. Giannakis (S'84–M'86–SM'91–F'97) received the Diploma in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1981, and the M.Sc. degree in electrical engineering, the M.Sc. degree in mathematics, and the Ph.D. degree in electrical engineering from the University of Southern California (USC), Los Angeles, in 1983, 1986, and 1986, respectively.

After lecturing for one year at USC, he joined the University of Virginia, Charlottesville, in 1987, where he became a Professor of Electrical Engineering in 1997. Since 1999, he has been a Professor at the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, where he now holds an ADC Chair in Wireless Telecommunications. His general interests span the areas of communications and signal processing, estimation and detection theory, time-series analysis, and system identification—subjects on which he has published more than 220 journal papers, 380 conference papers, and two edited books. Current research focuses on transmitter and receiver diversity techniques for single- and multiuser fading communication channels, complex-field and space-time coding, multicarrier ultrawide-band wireless communication systems, cross-layer designs, and sensor networks.

Dr. Giannakis is the (co)recipient of six paper awards from the IEEE Signal Processing (SP) and Communications Societies (1992, 1998, 2000, 2001, 2003, and 2004). He also received the IEEE-Signal Processing Society's Technical Achievement Award in 2000 and European Association for Signal, Speech and Image Processing (EURASIP) Technical Achievement Award in 2005. He served as Editor-in-Chief for the IEEE SIGNAL PROCESSING LETTERS, as Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING and the IEEE SIGNAL PROCESSING LETTERS, as Secretary of the Signal Processing Conference Board, as Member of the Signal Processing Publications Board, as Member and Vice-Chair of the Statistical Signal and Array Processing Technical Committee, as Chair of the Signal Processing for Communications Technical Committee, and as a Member of the IEEE Fellows Election Committee. He has also served as a Member of the IEEE-Signal Processing Society's Board of Governors, the Editorial Board for the PROCEEDINGS OF THE IEEE, and the Steering Committee of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.