Reduced Complexity Sphere Decoding Using Forcing Rules

Tao Cui, Chintha Tellambura and Wen Chen
Department of Electrical and Computer Engineering
University of Alberta
Edmonton, AB, Canada T6G 2V4
Email: {taocui, chintha, wenchen}@ece.ualberta.ca

Abstract—We improve the conventional sphere decoder (SD), which is widely used for lattice codes and multiple antenna space-time communications, by showing that some of the variables can be predetermined by using bounds on partial derivatives of the cost function. This reduces the number of variables to be estimated by the SD. Moreover, we combine the forcing rules with the conventional SD to provide several hybrid algorithms, which are found to be computationally more efficient. Forcing rules reduce complexity significantly when the dimension of the lattice is less than 12.

I. INTRODUCTION

Many communication theory problems can be formulated as
\[ y = Hx + n \] (1)
where \( x \in \mathbb{Z}^m \), \( y, n \in \mathbb{R}^n \) denote the system input, output and additive noise, and \( H \in \mathbb{R}^{n \times m} \) represents the transfer matrix (i.e., the channel matrix in MIMO system), \( \mathbb{Z}^m \) denotes an \( m \)-dimensional integer lattice. Generally, the noise terms \( n_i, i = 1, \ldots, n \) are independent and identically distributed (i.i.d.) zero-mean Gaussian random variables with the same variance. Under such conditions and assuming \( H \) is perfectly known at the receiver, the optimal maximum-likelihood (ML) detector that minimizes the average error probability is given by the integer least-squares problem
\[ \hat{x} = \arg \min_{x \in \mathbb{Z}^m} \| y - Hx \|^2 \] (2)
\( \| \cdot \| \) denotes the Euclidean norm. The lattice generated by \( H \) is [1]
\[ \Lambda(H) = \{ Hx : x \in \mathbb{Z}^m \} \] (3)
he columns of \( H \) are called basis vectors of \( \Lambda \), and the number \( m \) is said to be the dimension of \( \Lambda \). The problem in (2) is known as the closest vector problem (CVP) in the lattice theory. Other applications of CVP include vector quantization and cryptography.

The complexity of lattice problems has been widely investigated since the early 1980s because of the many connections between these problems and other areas of computer science and communication. CVP was proved NP-hard to solve exactly by van Emde Boas in [2]. Micciancio gave a simple proof in [3]. The first inapproximability results for CVP are due to Arora et al. [4] who proved that CVP is NP-hard to approximate within any constant factor, and quasi NP-hard to approximate within any constant factor less than \( \sqrt{2} \) is NP-hard for randomized reductions.

A remarkable algorithm for CVP stems from Fincke and Pohst (FP) [8], which is well-known as sphere decoding (SD) in communication theory [9], [10]. The FP algorithm only tests the lattice points lying inside a hypersphere. In wireless communication, SD first appeared in [11] on lattice code decoding. Since then, SD has gained popularity in CDMA [12], space-time coding [9] and MIMO systems over frequency selective channel [10], SD offers near-ML decoding at lower complexity compared with the exponential complexity incurred by exhaustive search.

Further improving the SD for (2) has recently received considerable interest due to the significant performance gain achieved by SD compared with other suboptimal algorithms such as VBLAST [13]. In section II, we review the basic FP and SE techniques. In section III, we propose two possessing techniques. The first one is by setting a lower bound on the remaining terms which are viewed as zero in original SD by using constrained least squares (CLS). Some components of the vector can be predetermined by checking the sign and the absolute value of the partial derivative at the corresponding components. In section IV, we adopt the partial derivative into the original SD so that some components can be predetermined when going down the search tree. Hence, the computational complexity can be reduced. The complexity comparison between the SD algorithms is given in section V.
Notation: Bold symbols denote matrices or vectors. $(\cdot)^T$ and $(\cdot)^H$ denote transpose and conjugate transpose, $(\cdot)^{-1}$ denotes pseudo-inverse. The set of all complex $K \times 1$ vectors is denoted by $C^K$. For $2^q$-ary phase shift keying (PSK), the signal constellation $Q_{2^q} = \{e^{i 2\pi k/2^q}, k = 0, 1, \ldots, 2^q - 1 \}$ and all PSK $N \times 1$ vectors are denoted by $Q^N_{2^q}$. 4-ary PSK is commonly known as quadrature PSK (QPSK). If $x$ and $y$ are Gaussian with $E[x] = \mu_x$, $E[y] = \mu_y$ and $E[(x - \mu_x)^2] = \sigma^2$, then $z = x + jy$ (where $j = \sqrt{-1}$) is Complex Gaussian. We write $z \sim \mathcal{CN}(\mu_x + j\mu_y, \sigma^2)$ in this case. The $N \times N$ identity matrix is denoted by $I_N$.

II. PREDETERMINATION

The original SD algorithm by Fincke and Phrot [8] (FPSD hereafter) is proposed to solve the SVP. FPSD restricts the search space to the lattice points within a sphere instead of searching all the lattice points. Each time a valid lattice point is found, the search space is restricted further by updating the radius. Although its worst case complexity was shown to be exponential, FPSD has been widely used in CVP due to its efficiency on many communication problems.

Before using FPSD or exhaustive search, if some components of $s$ can be determined, then some benefits accrue. The integer least squares problem (1) is equivalent to the following quadratic optimization problem

$$\min_{s \in A^n} f(s) = \frac{1}{2} s^T A s + B^T s$$

where $A = 2H^T H$ and $B = -2H^T y$. If $A = \{0, 1\}$, Eq.(4) is known as Quadratic Bivalent Programming in Operations Research, which belongs to a class of NP-complete combinatorial optimization problems. Hansen [14] gives a survey of the techniques used prior to 1980. Most of the proposed algorithms are of the branch and bound type. In [15]–[17], a preprocessing technique to fix variables by using the gradient of the quadratic function is proposed. This technique takes advantage of the special structure of the quadratic problem. From II C, QPSK modulated complex system can be decoupled to BPSK real system. Hence, we first apply the predetermination technique to BPSK systems and then generalize it to other constellations in this subsection.

Theorem 1 Let $s^*$ be the global minimum of the problem

$$\min_{s \in A^n} f(s) = \frac{1}{2} s^T A s + B^T s, \quad -1 \leq s_i \leq 1$$

Then $s^*$ is also optimal for the linear problem

$$\min_{s \in A^n} (\nabla f(s^*))^T s, \quad -1 \leq s_i \leq 1$$

Proof For any feasible point $y$ and $\lambda \in (0, 1)$, the point $z = (1 - \lambda)s^* + \lambda y$ is also feasible and therefore $f(z) - f(s^*) \geq 0$, or $\lambda (B + A s^*)^T (y - s^*) + \frac{\lambda^2}{2}(y - s^*)^T A (y - s^*) \geq 0$. Since $\lambda \neq 0$, this implies that $(B + A s^*)^T (y - s^*) + \frac{1}{2} \lambda (y - s^*)^T A (y - s^*) \geq 0$. That is $f(s^*)^T y \geq f(s^*)^T y$ for all feasible $y$.

From Theorem 1, we can get the following corollary directly.

Corollary 1 Suppose that $m_i \leq \partial f(s)/\partial s_i \leq M_i$ for all $s \in \{-1, 1\}^M$ and $i = 1, 2, \ldots, M$. If $m_i > 0$, then $s^*_i = 1$. If $M_i < 0$, then $s^*_i = -1$. Corollary 1 can be interpreted as if the partial derivative of $f(s)$ with respect to variable $s_i$ is always larger than zero (or less than zero), $f(s)$ will be monotonously increasing (or decreasing) in the $i$th dimension and then $s_i$ can be fixed as $-1$ (or 1). The bounds $m_i$ and $M_i$ that force variables to be fixed can be improved by exploiting the constant modulus property of set $\{-1, 1\}$. Note that the objective function in the original optimization problem (4) is equivalent to

$$\min_{s \in \{-1, 1\}^M} f(s) = \frac{1}{2} s^T \tilde{A} s + B^T s$$

where $\tilde{a}_{ij} = a_{ij}$ for $i \neq j$ and $\tilde{a}_{ii}, i = 1, 2, \ldots, M$ can be any real number. The partial derivative of $f(s)$ with respect to variable $s_i$ is

$$\frac{\partial f(s)}{\partial s_i} = \tilde{a}_{i}^T s + b_i = \sum_{j=1,j\neq i}^{M} a_{ij} s_j + \tilde{a}_{ii} s_i + b_i$$

where $\tilde{a}_i$ is the $i$th row of $A$. Since $s_j \in \{-1, 1\}, j = 1, 2, \ldots, M$, we have

$$- \sum_{j=1,j\neq i}^{M} |a_{ij}| - |\tilde{a}_{ii}| + b_i \leq \frac{\partial f(s)}{\partial s_i} \leq \sum_{j=1,j\neq i}^{M} |a_{ij}| + |\tilde{a}_{ii}| + b_i$$

Hence the range of the partial derivative is $2 \sum_{j=1,j\neq i}^{M} |a_{ij}| + 2|\tilde{a}_{ii}|$. To improve the bounds $m_i$ and $M_i$, a heuristic consideration is to choose $\tilde{a}_{ii}$ that minimize the range of the partial derivative. The range of the partial derivative is minimized for $\tilde{a}_{ii} = 0, i = 1, 2, \ldots, M$. Therefore, the minimization problem (4) can be replaced by the equivalent optimization problem

$$\min_{s \in \{-1, 1\}^M} f(s) = \frac{1}{2} s^T A_0 s + B^T s$$

where $A_0$ is the matrix $A$ whose diagonal entries are replaced by zero. Hence

$$m_i = - \sum_{j=1,j\neq i}^{M} |a_{ij}| + b_i, \quad M_i = \sum_{j=1,j\neq i}^{M} |a_{ij}| + b_i$$

Note that the bounds $m_i$ and $M_i$ can be improved further by using the predetermined variables prior to the test of $s_i$. If $P_i$ denotes the index set whose components are the indexes corresponding to the predetermined components of $s$ prior to the test of the $i$th components, from (8) we have

$$- \sum_{j=1,j\notin P_i}^{M} |a_{ij}| + \sum_{j \in P_i}^{M} a_{ij} s_j^* + b_i \leq \frac{\partial \tilde{f}(s)}{\partial s_i} \leq \sum_{j=1,j\notin P_i}^{M} |a_{ij}| + \sum_{j \in P_i}^{M} a_{ij} s_j^* + b_i.$$
results, after testing all the components of $s$ for one time all the variables are tested again and again until no further components of $s$ can be predetermined. In this way, all the components of $s$ can use other components’ predetermination information. We call this Predetermination Algorithm 1 or shortly Predetermination 1.

However, if the sign of the partial derivative is not constant, the predetermination fails. To the best of the authors’ knowledge, little attention has been paid to the use of the absolute value of the partial derivative for predetermination in the literature. In the following, we give another predetermination theorem due to the special structure of the quadratic form.

**Theorem 2** Suppose that

$$m_i^{-1} \leq \left| \frac{\partial f(s)}{\partial s_i} \right|_{s_i=1} \leq M_i^{-1}, \quad m_i^2 \leq \left| \frac{\partial f(s)}{\partial s_i} \right|_{s_i=-1} \leq M_i^2$$

for all $s \in \{-1,1\}^M$ and $i = 1, 2, \ldots, M$. If $M_i^{-1} < m_i^2$, then $s_i^* = -1$. If $M_i^2 < m_i^{-1}$, then $s_i^* = 1$.

**Proof** For feasible points $s_1$ and $s_2$, all the components of $s_1$ and $s_2$ are equal except the $i$th component. That is $s_{ij} = s_{2j}$, $j \neq i$ and $s_{ii} = s_{2i}$. The optimization problem (4) is equivalent to

$$\min g(s_i) = \frac{a_{ii}}{2} s_i^2 + \sum_{j=1,j \neq i}^M a_{ij} s_j + b_i s_i$$

Note that since $A = 2H^T \mathbf{H}$, $A = A^T$, $a_{ij} = a_{ji}$ and $a_{ii} > 0$. From (8), the partial derivative of $f(s)$ with respect to variable $s_i$ is

$$\frac{\partial f(s)}{\partial s_i} = a_{ii} s_i + \sum_{j=1,j \neq i}^M a_{ij} s_j + b_i = \frac{dg(s_i)}{ds_i}$$

Thus we have

$$\left( \frac{dg(s_i)}{ds_i} \right)^2 = 2a_{ii} g(s_i) + \left( \sum_{j=1,j \neq i}^M a_{ij} s_j + b_i \right)^2$$

Since $a_{ii} > 0$, minimizing $g(s_i)$ is equivalent to minimize $|dg(s_i)/ds_i|$. For all $s \in \{-1,1\}^M$, if $M_i^{-1} < m_i^2$, $g(-1) < g(1)$ and $f(s_1) < f(s_2)$ where $s_{1j} = -1$, $s_{2j} = 1$. Therefore, $s_i$ is fixed to $-1$ and vice versa.

The theorem is valid due to the special structure of the quadratic form. We follow the iterative testing procedure in the discussion of Theorem 1. We call this Predetermination Algorithm 2 or shortly Predetermination 2.

Theorem 1 and 2 can be readily generalized to the case that the number of elements in $A$ or the cardinal number of $A$ is larger than 2.

**Remarks:**

- If $P$ variables can be predetermined, the search space will reduce to $A^{M-P}$, which means the complexity of the search can be significantly reduced. From simulation, we find that when $N$ is small, most of the variables can be predetermined.

- The predetermination technique can be combined with the lower bound preprocessing to achieve lower complexity.

- If the channel is constant over several blocks, $\sum |a_{ij}|$ does not need to be calculated again. The predetermination stage computational cost is negligible.

### III. Sphere Decoding with Predetermination

We next incorporate predetermination into the FP decoder. The FP decoder [8] solves the equivalent problem

$$\hat{s} = \arg\min_{s \in \mathbb{A}^M} ||y' - Rs||^2$$

where $y' = Q^T y$, $R$ an $M \times M$ upper triangular matrix and $Q = [Q_1, Q_2]$ an $N \times N$ orthogonal matrix are QR factorization of the matrix $H$. The matrices $Q_1$ and $Q_2$ represent the first $M$ and last $N - M$ orthonormal columns of $Q$.

**Hybrid Algorithm 1:** When the search proceeds to $s_i$ and the size of candidate set for $s_i$ is larger than one, we check the bounds for $\partial f(s)/\partial s_i$, $m_i = -\sum_{j=1}^{-M} |a_{ij}| + \sum_{j=1}^{-M} |a_{ij}|^2 + b_i$, $M_i = \sum_{j=1}^{M} |a_{ij}| + \sum_{j=1}^{M} a_{ij}s_j + b_i$ $(j = i + 1, \ldots, M)$ have already been assigned values. Using Theorem 1, if $m_i > 0$, $s_i^* = -1$ and if $M_i < 0$, $s_i^* = 1$. Since $\sum_{j=1}^{-M} |a_{ij}|$ can be pre-computed before search, it needs 2$(M - i)$ flops to obtain the bounds. Theorem 2 can be applied in the same way.

**Hybrid Algorithm 2:** In (17), when $s_{i+1}$ has been assigned a value, $r_{i+1}s_{i+1}$ can be subtracted from $y'$ and it forms a new equivalent problem

$$s^i = \arg\min_{s^i \in \mathbb{A}^i} ||y'' - R_i s^i||^2$$

where $s^i = [s(1), s(2), \ldots, s(i)]^T$, $y'' = y'^{i+1} - r_{i+1}s_{i+1}$, $R_i$ is the $i \times i$ submatrix of $R$ and $r_{i+1}$ is the $(i+1)$-th column of $R$. Problem (18) is equivalent to

$$\min f(s^i) = \frac{1}{2} s^i^T A_i s^i + (B^i)^T s^i$$

where $A^i = (R^i)^T R^i$ and $B^i = -R^i y'^i$. The bounds for the partial derivative of $f(s^i)$ with respect to $s_i$ are $m_i = -\sum_{j=1}^{i-1} |a_{ij}| + b_i$, $M_i = \sum_{j=1}^{i-1} |a_{ij}| + b_i$. Since $a_{ij} = a_{ji}$ where $a_{ij}$ are elements of $A = R^T R$, $M_{ij} = M_{ji}$ can be pre-computed. Calculating $b_i$ needs 2$i$ flops.

### IV. Simulation Results and Conclusion

We now compare the proposed algorithms with the FP decoder. We consider an $M$ transmit and $N$ receive antennas, BPSK modulated, uncoded MIMO system over a multiple antenna flat Rayleigh fading channel. We simulate this system using MATLAB V5.3 on a PC with an Intel Pentium-4 processor at 1.8GHz. The matlab command "flops" is used to count the number of flops. Only the flops of the search algorithm are counted without accounting for the preprocessing stage. The initial radius is chosen according to the noise variance as in [18]. Fig. 1 compares the average complexity of the two predetermination techniques with that of the FP decoder for SNR=15dB. Predetermination is especially effective when
Surprisingly, some cases can sometimes be totally predetermined without performing any search when \( m \leq 12 \), which is often the case in practical systems. When \( m \) increases, the bounds for partial derivative becomes loose. Fig. 2 show the average complexity of different algorithms: Hybrid 1, Hybrid 2 and FP, when SNR=15dB. The two hybrid algorithms are effective when \( m \) is small. If \( m > 12 \), the forcing rules can be applied in the last 12 dimensions.

In this paper, we developed reduced complexity variants of the SD algorithm by using forcing rules that utilizes bounds on the partial derivatives. Simulation results demonstrate that such hybrid algorithms are effective when the dimension of the lattice is less than 12.

REFERENCES