Fast Optimal and Sub-optimal Any-Time Algorithms for CWMA Multiuser Detection based on Branch and Bound

J. Luo, K. Pattipati, P. Willett, G. Levchuk

Abstract

A fast optimal algorithm based on the branch and bound method, coupled with an iterative lower bound update, is proposed for the joint detection of binary symbols of K users in a synchronous correlated waveform multiple-access (CWMA) channel with Gaussian noise. Although the group optimal detection problem is generally NP hard, the proposed method can significantly decrease the average computational cost. A fast "any-time" sub-optimal algorithm is also available by simply picking the "Current-Best" solution in the Branch and Bound method. Theoretical results are given on the computational complexity and the performance of the "Current-Best" sub-optimal solution. Although the performance of sub-optimal algorithms are affected by the detection order, an order algorithm is proposed and the optimal detection order is shown to be identical to all the proposed sub-optimal algorithms. Simulation results are presented to verify the theoretical analysis. In the situation when only a small number of users are correlated, sub-optimal algorithm outperforms the decision feedback method significantly and the computational cost can be even less than that of the conventional method.

Keywords

Multiuser Detection, Branch-and-Bound, Optimal Detection Sequence, Correlated Waveform Multiple Access, Any-Time Algorithm.

I. Introduction

DUE to the problem of interuser interference in many multiuser communication systems, multiuser detection for the symbol-synchronous Gaussian correlated waveform multiple-access (CWMA) channel has received considerable attention over the past ten years. When the source signals are binary- or integer-valued, the resulting integer programming problem is generally NP hard [2], unless the signature waveform auto-correlation matrix has a special structure [11] [12]. Consequently, prior research has focused on designing suboptimal receivers with low computational complexity and better performance than a conventional linear detector. Among them are the multistage detection [3] [5], the group detection [7] and the decision feedback detection [6] [8] [13]. Usually, suboptimal methods

The Authors are with the Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT06269, USA. E-mail:krishna@engr.uconn.edu

[1]This work was supported by the Office of Naval Research under contract #N00014-98-1-0465, #N00014-00-1-0101, and by NUWC under contract N66604-1-99-5021

need to perform a projection to satisfy the integrality constraints, which can cause significant detection errors.

Based on the idea of successive cancellation, a systematic Decorrelator-based Decision Feedback Detection (D-DFD) approach was given in [13]. While maintaining the computational complexity of $O(K^2)$, D-DFD methods provide a significant improvement in probability of error when compared with traditional Minimum Mean Square Error (MMSE)-based decorrelation detector. However, computer simulations show that, in most cases, especially when some signature waveforms are correlated, there is still a large gap between the probability of error of D-DFD outputs and that of the optimal solution.

In this paper, we consider the multiuser detection problem as a constrained optimization problem. A fast optimal algorithm based on the branch and bound method is proposed. The Minimum Mean Square Error (MMSE) method is used to find a lower bound; this requires fewest branches to find the best solution in the noise-free case. In addition, when the noise is small, the MMSE lower bound is tight and helps greatly in pruning the number of branches, thereby decreasing the number of sub-problems to be solved. Although the computational cost for the worst case is exponential, computer simulations show that the optimal algorithm maintains the same level of computation on average as the D-DFD method, while being able to provide significant improvement in the probability of error when compared to the D-DFD method [13]. In the noise-free case, the optimal algorithm even requires less computation than the conventional methods. Furthermore, theoretical discussion is given to show that the D-DFD method is in fact an order-one approximation to the proposed optimal solution. When strict computational limits exist, a sub-optimal solution can be obtained by simply picking the "current-best" solution in the branch-bound search. Since the D-DFD method is a first-order approximation to the optimal algorithm, a "current-best" solution of second or third order approximation will generally outperform the D-DFD method with marginal increases in computation. Theoretical analysis of the performance and computational cost on the "current-best" solution is given and verified by the simulation results. For a given CWMA system, these results can be used offline to estimate the performance and computational costs for the proposed optimal and sub-optimal algorithms. In addition, the optimal detection sequences for different computational constraints as well as the D-DFD methods are found to be identical.

The rest of the paper is organized as follows. The synchronous multi-user detection problem formulation and existing solution techniques are discussed in section 2. In section 3, a fast optimal algorithm to reduce the number of sub-problems is presented, and theoretical analysis of the computational complexity is given. By using a depth-first search, analysis is given to show that the D-DFD solution is in fact the first feasible solution in the branch-and-bound approach. In section 4, the performance and the computational complexity for the "current-best" sub-optimal solution are discussed. Theoretical analysis for finding the optimal detection sequence is given at the end of this section. Simulation results and comparative analyses are provided in section 5. The paper concludes with a summary in section 6.

II. Problem Formulation and Existing Methods

A discrete-time equivalent model for the matched-filter outputs at the receiver of a CWMA channel is given by the K-length vector [2]

$$y = Hb + n \tag{1}$$

where $b \in \{-1, +1\}^K$ denotes the K-length vector of bits transmitted by the K active users. Here $H = W^{\frac{1}{2}} R W^{\frac{1}{2}}$ is a nonnegative definite signature waveform correlation matrix, R is the symmetric normalized correlation matrix with unit diagonal elements, W is a diagonal matrix whose k-th diagonal element, w_k, is the received signal energy per bit of the k-th user, and n is a real-valued zero-mean Gaussian random vector with a covariance matrix $\sigma^2 H$. It has been shown that this model holds for both baseband [2] and passband [13] channels with additive Gaussian noise.

When all the user signals are equally probable, the optimal solution of (1) is the output of a Maximum Likelihood (ML) detector [2]

$$\phi_{ML} : \hat{b} = \arg \min_{b \in \{-1,+1\}^K} \left(b^T H b - 2 y^T b \right) \tag{2}$$

The ML detector has the property that it minimizes, among all detectors, the probability that not all users' decisions are correct. Usually, ϕ_{ML} is considered NP-hard and exponentially complex to implement; the focus is then on developing easily implementable and effective multiuser detectors.

The MMSE-based solution of conventional decorrelation detector [2]

$$\phi_D : \hat{b} = \arg \min_{b \in \{-1,+1\}^K} \left\| b - H^{-1} y \right\|_2^2 \tag{3}$$

is found in two steps. First, the unconstrained solution $\tilde{b} = H^{-1} y$ is computed. This is then projected onto the constraint set via: $\hat{b}_i = \text{sign}\left(\tilde{b}_i\right)$.

The DFD method based on the decorrelation detector is described in [13]. It is characterized by

$$\phi_{D-DF} : \hat{b} = P\tilde{b},\ \tilde{b}_i = \text{sign}\left(\sum_{j=1}^{K} F_{ij} P y_j - \sum_{j=1}^{i-1} B_{ij} \tilde{b}_j \right) \tag{4}$$

where $F = \text{U}\left([PHP]^{-1}\right)$, $B = \text{L}(FPHP)$. Here, $\text{U}(\cdot)$ represents the upper triangular part of a matrix, $\text{L}(\cdot)$ represents the strictly lower triangular part of a matrix, and P is a permutation matrix (symmetric and orthoganal). The choice of P has been discussed in Theorem 1 of [13].

For the group detection of multiuser detectors, Symmetric Energy (SE) is an important performance measure in the low noise regime [13]. It is defined as follows,

Definition: Let $\varepsilon(\phi)$ denote the event that the detector does not detect all users correctly. Let the effective energy $e(\sigma,\phi)$ corresponding to the probability of $\varepsilon(\phi)$ be defined implicitly via the equation

$$Pr(\varepsilon(\phi)) = Q\left(\frac{\sqrt{e(\sigma,\phi)}}{\sigma}\right) \tag{5}$$

where $Q(.)$ is defined as $Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx$. The SE is defined as the limit of the effective energy $e(\sigma,\phi)$ as $\sigma \to 0$ so that

$$E(\phi) = \lim_{\sigma\to 0} e(\sigma,\phi) = sup\left\{e \geq 0; \lim_{\sigma\to 0} \frac{Pr(\varepsilon(\phi))}{Q\left(\sqrt{e}/\sigma\right)} < \infty\right\} \tag{6}$$

If we denote the $(i,j)^{th}$ component of a matrix A by A_{ij}, the SE of the ML detector, Decorrelator and the D-DFD detector can be expressed, respectively, by [13]

$$\begin{aligned} E(\phi_{ML}) &= \min_{e\in\{-1,0,1\}^K-\{0\}} e^T He \\ E(\phi_D) &= \min_{i=1,\dots,K} \frac{1}{[H^{-1}]_{ii}} \\ E(\phi_{D-DF}) &= \min_{i=1,\dots,K} L_{ii}^2 \end{aligned} \tag{7}$$

where L is the Cholesky factor of PHP, i.e., $PHP = L^T L$. Although both the computational costs for the decorrelation and D-DFD algorithms are K^2 multiplications and $K(K-1)$ additions, it has been shown [13] that $E(\phi_{D-DF}) \geq E(\phi_D)$. Usually D-DFD can provide 2 to 3 orders of improvement in the magnitude of probability of error when compared with a conventional linear detector. However, the output of D-DFD is still a sub-optimal solution. Simulation results show that, in most cases, there still exists a substantial gap in performance between the D-DFD and the optimal solutions.

III. Optimal Algorithm Based on Branch and Bound

The idea of using a branch and bound method in solving optimization problems is already well known [4]. However, the tradeoff between a tight lower bound and a lower bound with less computational requirements is common to most of the problems. In multiuser detection, branch-and-bound method with breadth-first search has been used in [9] to find the minimum distance, which is defined by,

$$d_{min} = \sqrt{\min_{e\in\{-1,0,1\}^K-\{0\}} e^T He} \tag{8}$$

Similar to [9], in this paper, the fast optimal algorithm uses a tight (when noise is small) MMSE lower bound. In particular, we update the lower bound in an iterative way such that the computational cost for each estimation is linear in the number of the rest of unassigned users.

Suppose $H = L^T L$ is the Cholesky decomposition of H. Then $H^{-1} = L^{-1}L^{-1^T}$. The objective function in (2) can be equivalently written as

$$\phi_{ML} : \hat{b} \;\; = \arg\min_{b\in\{-1,+1\}^K} (b - H^{-1}y)^T H (b - H^{-1}y)$$

$$= \arg \min_{b \in \{-1,+1\}^K} \left\| L(b - H^{-1}y) \right\|_2^2$$
$$= \arg \min_{b \in \{-1,+1\}^K} \left\| Lb - {L^{-1}}^T y \right\|_2^2 \tag{9}$$

Define $\tilde{y} = {L^{-1}}^T y$, $D = Lb$, and denote the kth component of D and $\tilde{y}$ by D_k and $\tilde{y}_k$, respectively. Consequently, we have

$$\phi_{ML} : \hat{b} = \arg \min_{b \in \{-1,+1\}^K} \| D - \tilde{y} \|_2^2$$
$$= \arg \min_{b \in \{-1,+1\}^K} \sum_{k=1}^{K} (D_k - \tilde{y}_k)^2 \tag{10}$$

Here, since L is a lower triangular matrix, D_k depends only on $(b_1, b_2, ..., b_k)$. When the decisions for the first k users are fixed, the term

$$\xi_k = \sum_{i=1}^{k} (D_i - \tilde{y}_i)^2 \tag{11}$$

can serve as a lower bound of (10). It can be easily seen that the lower bound is in fact an unconstrained MMSE solution and is achievable when the binary constraints on $(b_{k+1}, ..., b_K)$ are disregarded. The branch and bound tree search to find the minimum value of $\|D - \tilde{y}\|_2^2$ is described below.

Similar to a general branch and bound method [10], the algorithm maintains a node list called $OPEN$, and a scalar called $UPPER$, which is equal to the minimum feasible cost found so far, i.e., the "Current-Best" solution. Define k to be the level of a node (virtual root node has level 0). Label the branches with $D_k(b_1, b_2, ..., b_{k+1})$, which connect the two nodes $(b_1, ..., b_k)$ and $(b_1, ..., b_{k+1})$. The node $(b_1, ..., b_k)$ is labeled with the lower bound ξ_k. Also, define $v_k = \sum_{i=1}^{k} [b_i * (\text{the } i\text{th column of } L)] - \tilde{y}$, denote $[v_k]_j$ as the jth component of vector v_k, and L_{ij} as the (i, j)th element of L. The branch and bound algorithm proceeds as follows.

1) Precompute $\tilde{y} = {L^{-1}}^T y$;

2) Initialize $k = 0$. $v_k = -\tilde{y}$, $\xi_k = 0$, $UPPER = +\infty$ and $OPEN = NULL$;

3) Set $k = k + 1$. Choose the node in level k such that $b_k = -sign\left([v_{k-1}]_k\right)$. If $k < K$, append the node with $b_k = sign\left([v_{k-1}]_k\right)$ to the end of the $OPEN$ list;

4) Compute $v_k = v_{k-1} + b_k * (\text{the } k\text{th column of } L)$;

5) Compute $\xi_k = \xi_{k-1} + (D_k - \tilde{y}_k)^2 = \xi_{k-1} + (v_k)_k^2$;

6) If $\xi_k \geq UPPER$ and the $OPEN$ list is not empty, drop this node. Pick the node from the end of the $OPEN$ list, set k equal to the level of this node and go to step 4;

7) If $\xi_k < UPPER$, $k = K$ and the $OPEN$ list is not empty, update the "Current-Best" solution and $UPPER = \xi_k$. Pick the node from the end of the $OPEN$ list, set k equal to the level of this node and go to step 4;

8) If $\xi_k < UPPER$ and $k \neq K$, go to step 3;

9) If $\xi_k < UPPER$, $k = K$ and the $OPEN$ list is empty, update the "current-best" solution and $UPPER = \xi_k$;

10) For all other cases, stop and report the "current-best" solution.

Example 1: The following 3-user example illustrates the procedure. The system is given by

$$\begin{aligned} y &= Hb + n \\ H &= \begin{bmatrix} 4.25 & 0.85 & 0.57 \\ 0.85 & 3.0 & 1.6 \\ 0.57 & 1.6 & 2.0 \end{bmatrix} \\ &= \begin{bmatrix} 2.0 & 0 & 0 \\ 0.3 & 1.3 & 0 \\ 0.4 & 1.1 & 1.4 \end{bmatrix}^T \begin{bmatrix} 2.0 & 0 & 0 \\ 0.3 & 1.3 & 0 \\ 0.4 & 1.1 & 1.4 \end{bmatrix} \end{aligned} \tag{12}$$

Assume the source signal is $b = [1, -1, 1]^T$, the noise vector is $n = [0.81, 1.93, -0.22]^T$, hence $y = [4.78, 1.38, 0.75]^T$. Figure 1 shows the branch-and-bound tree structure.

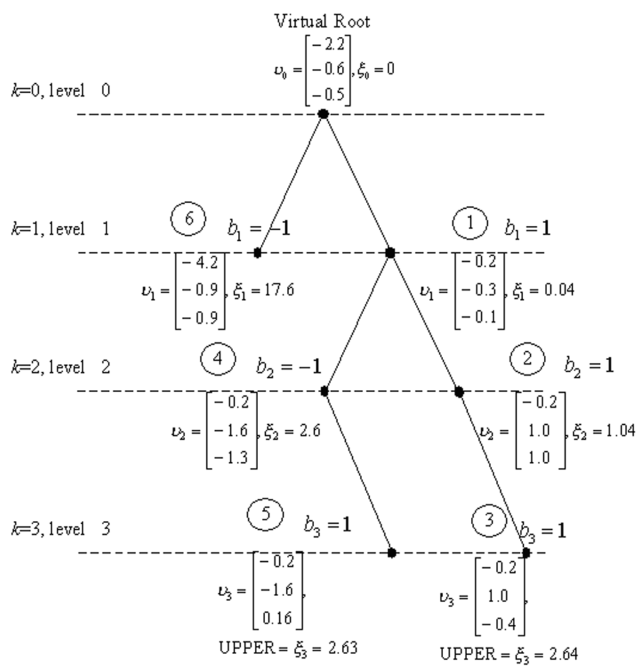

Fig. 1. Example of the depth-first Branch-and-Bound algorithm

In step 1), we precompute $\tilde{y} = (L^{-1})^T y = [2.2, 0.6, 0.5]^T$. Then, initialize $k = 0$, $\upsilon_0 = [-2.2, -0.6, -0.5]^T$, $\xi_0 = 0$, $UPPER = +\infty$, $OPEN = NULL$. In step 3), let $k = 1$, choose the node with $b_1 = -sign(-2.2) = 1$ (node 1 in Figure 1). Add node 5 to the $OPEN$ list. Update $\upsilon_1 = [-0.2, -0.3, -0.1]^T$, $\xi_1 = 0.04$. Since $\xi_1 < UPPER$, goto step 3. This leads us to node 2. Add node 4 to the end of the $OPEN$ list. Then, since the next level is the bottom level, from step 3, we know node 3 gives better result than node $(1,1,-1)$. Therefore, without changing the $OPEN$ list, we go to node 3 (which is the first feasible solution and, as shown later, it also corresponds to the D-DFD solution.) and update $UPPER = \xi_3 = 2.64$. In step 6, we pick node 4 from the end of the $OPEN$ list. Go to node 5, and obtain $\xi_3 = 2.63 < UPPER$, which means that node 5 is a better solution. Update $UPPER = 2.63$ and pick node 6 from the $OPEN$ list. For node 6, since $\xi_1 = 17.6 > UPPER$, we drop this node. Now the $OPEN$ list is empty, the algorithm stops and reports node 5 as the optimal solution.

The above algorithm is a branch and bound method with depth-first search. The computational cost for step 1) is $\frac{K(K+1)}{2}$ multiplications and $\frac{K(K-1)}{2}$ additions. In step 3), since b_k can only take known discrete values, $b_k L$ can be precomputed and stored; hence, only $K-k+1$ additions are needed to obtain v_k. Step 5) needs 1 addition and 1 multiplication. Notice that step 1) is outside the branch and bound search. To update the lower bound for a node on level $K-k+1$ $(k=1,...,K)$, only $k+1$ additions and 1 multiplication is needed. In addition, the computation for finding the first feasible solution (also the optimal solution in the noise-free case) requires $\frac{K(K+3)}{2}$ multiplications and $K(K+1)$ additions.

Proposition 1: The first feasible solution obtained from the above depth-first search is the solution of D-DFD method.

Proof: From step 3), when we branch, we first go to the node with a smaller lower bound value. In the above branch and bound method, suppose $(b_1,...,b_{k-1})$ has already been fixed by the branch, the choice of b_k for the branch and bound method can be described by

$$\begin{aligned} \tilde{b} &= \arg \min_{\substack{b_k \in \{-1,+1\} \\ b_{k+1},...,b_K \in (-\infty,\infty)}} (b - H^{-1}y)^T H (b - H^{-1}y) \\ b_k &= \tilde{b}_k \end{aligned} \tag{13}$$

Notice that in (13), $(b_1,...,b_{k-1})$ is fixed and we only have a binary constraint on b_k. The choice of b_k for D-DFD method, however, is given by

$$\begin{aligned} \tilde{b} &= \arg \min_{b_k,...,b_K \in (-\infty,\infty)} (b - H^{-1}y)^T H (b - H^{-1}y) \\ b_k &= sign(\tilde{b}_k) \end{aligned} \tag{14}$$

Figure 2 shows the difference between the above two choices. The ellipses here represent the level curves of the objective function. For the D-DFD method, the decision on b_k is made by comparing the lengths $|AO|$ and $|BO|$. While for the proposed branch and bound method, the decision on b_k is made by comparing the lengths $|CO|$ and $|DO|$. Since the triangles AOC and BOD are similar, (13) and (14) are equivalent.

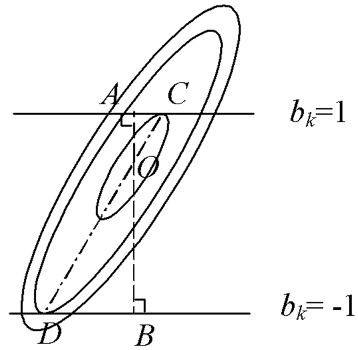

Fig. 2. Comparison of D-DFD and Branch-Bound decisions on b_k

Example 1 - continued : In the above example, on node 1, the user expurgated channel for the D-DFD method is represented by

$$\begin{bmatrix} y_2 \\ y_3 \end{bmatrix} - \begin{bmatrix} 0.85 \\ 0.57 \end{bmatrix} = \begin{bmatrix} 3.0 & 1.6 \\ 1.6 & 2.0 \end{bmatrix} \begin{bmatrix} b_2 \\ b_3 \end{bmatrix} + \begin{bmatrix} n_2 \\ n_3 \end{bmatrix} \tag{15}$$

According to (14), the decision on b_2 for D-DFD is made by

$$\begin{aligned} \tilde{b} = & \begin{bmatrix} 3.0 & 1.6 \\ 1.6 & 2.0 \end{bmatrix}^{-1} \left\{ \begin{bmatrix} 1.38 \\ 0.75 \end{bmatrix} - \begin{bmatrix} 0.85 \\ 0.57 \end{bmatrix} \right\} \\ = & \begin{bmatrix} 0.22 \\ -0.09 \end{bmatrix} \\ b_2 = & \, sign(\tilde{b}_2) = 1 \end{aligned} \tag{16}$$

which is consistent with the depth-first direction of branch-and-bound algorithm. However, as shown in the example, D-DFD method failed to find the optimal solution.

Recall that in the branch and bound algorithm, the computational cost required to obtain the first feasible solution (also the solution of D-DFD) is much less than the computational cost of a conventional linear detector. Evidently, any further computations will result in better accuracy than the D-DFD (unless the D-DFD is already optimal).

IV. "Any-Time" Suboptimal Algorithm

Although the average computational cost may not be very high, the computation for the worst case is still exponential in the number of users since the ML solution is generally NP hard. Hence the optimal algorithm is not implementable when the number of users is large. When a strict limitation on computational cost exists, the "current-best" solution in the above branch and bound method can serve as a sub-optimal alternative to the NP hard optimal solution.

Define the sub-opimal detector that explores the sub-tree under and including level $K-k+1$ to be ϕ_{BB-k} $(k=1,...,K)$. From the above analysis of the computational cost, the worst-case computation for ϕ_{BB-i} is given by

$$\begin{aligned} \text{Multiplications} \quad & \leq \frac{K(K+3)}{2} + 3 * 2^{k-1} - k - 2 \\ \text{Additions} \quad & \leq K(K+1) + 5 * 2^{k} \\ & - \frac{(k+3)(k+4)}{2} \end{aligned} \tag{17}$$

To derive the Symmetric Energy (SE) measure for ϕ_{BB-k}, define $P(i|1,...,i-1)$ to be the event that the decision on user i is correct $(i=1,...,K-k)$, given all the decisions on users $j<i$ are correct. Consider the ML solution (9). Substitute (1) into (9), denote the true source signal by b_0, to obtain

$$\begin{aligned} \phi_{ML}: \hat{b} \quad & = \arg \min_{b \in \{-1,+1\}^K} \left\| Lb - {L^{-1}}^T y \right\| \\ & = \arg \min_{b \in \{-1,+1\}^K} \left\| L(b-b_0) - {L^{-1}}^T n \right\| \\ & = \arg \min_{b \in \{-1,+1\}^K} \left\| L(b-b_0) - \tilde{n} \right\| \end{aligned} \tag{18}$$

Since $E[nn^T] = \sigma^2 H$, we have $E[\tilde{n}\tilde{n}^T] = \sigma^2 I$. Thus, $\tilde{n}$ can be viewed as K independent zero mean Gaussian noise variables with covariance matrix $\sigma^2 I$. Assuming that the decisions on $(b_1,...,b_{i-1})$ are correct, the lower bound ξ_i can be expressed as

$$\xi_i = \sum_{j=1}^{i} (D_j - \tilde{y}_j)^2 = \sum_{j=1}^{i-1} \tilde{n}_j^2 + [L_{ii}(b_i - b_{0i}) - \tilde{n}_i]^2 \tag{19}$$

Since $\tilde{n}$ is Gaussian with a covariance matrix $\sigma^2 I$, $P(i|1,...,i-1)$ is given by

$$P(i|1,...,i-1) = Q(\frac{|L_{ii}|}{\sigma}) \tag{20}$$

Also, similar to (8), define the minimum distance among users $K-k+1,...,K$ by

$$d_{min-k} = \sqrt{\min_{\substack{e \in \{-1,0,1\}^K - \{0\} \\ e_1,...,e_{K-k} = 0}} e^T H e} \tag{21}$$

Given that the decisions on users $1,...,K-k$ are correct, the group detection error of ϕ_{BB-k} can be approximated by

$$P(K-k+1,...K|1,...,K-k) \approx Q(\frac{d_{min-k}}{\sigma}) \tag{22}$$

Therefore, the overall group decision error of ϕ_{BB-k} can be expressed as

$$P_e^{\phi_{BB-k}} \approx 1 - \{\prod_{j=1}^{K-k} [1 - Q(\frac{|L_{jj}|}{\sigma})]\}[1 - Q(\frac{d_{min-k}}{\sigma})] \tag{23}$$

The SE is then given by

$$E(\phi_{BB-k}) = \min_{i=1,...,K-k}(L_{ii}^2, d_{min-k}^2) \tag{24}$$

Furthermore, from the definitions of (21) and (8), we have

$$d_{min-k}^2 \geq d_{min}^2 = E(\phi_{ML}) \geq E(\phi_{BB-k}) \tag{25}$$

$E(\phi_{BB-k})$ can then be denoted by

$$E(\phi_{BB-k}) = min(d_{min}^2, \min_{i=1,...,K-k} L_{ii}^2) \tag{26}$$

Evidently, the performance and even the average computational cost of the above sub-optimal method are also affected by the detection order of the users. For the D-DFD method, a user ordering algorithm is proposed in [13] as follows,

Order Algorithm: Order users as follows: select the first user in the new order (denote this user's index as i_1) as

$$i_1 = arg \min_{j=1,...,K}[H^{-1}]_{jj} \tag{27}$$

For $k = 2, ..., K$, form a new matrix $\hat{H}$ to be part of H that only contains the components $\{H_{ij}\}$ $(i, j \in \{\{1, ..., K\} - \{i_1, ..., i_{k-1}\}\})$. Find

$$\hat{i}_k = arg \min_{j=1,...,K-k+1}[\hat{H}^{-1}]_{jj} \tag{28}$$

and let i_k equal to the user corresponding to $\hat{i}_k$.

Proposition 2: When ordering users by the order algorithm, the Symmetric Energy $E(\phi_{BB-k})$ of all $k = 1, ..., K$ are maximized simultaneously.

Proposition 2 states that the order defined by the order algorithm is optimal for all the proposed sub-optimal detectors as well as the D-DFD. See appendix for the proof.

V. Simulation Results

According to (17), the computational complexity for the sub-optimal detector ϕ_{BB-k} is exponential in k. However, since we assume H to be known, the SE of the proposed "current-best" sub-optimal solutions can be found offline by (26). The following simulation results show that, in some cases, a small amount of extra computation can significantly improve the performance of the system, when compared with the ϕ_{D-DF} (which is the same as ϕ_{BB-1}).

Example 1 - continued: In the previous example, since users 2 and 3 are strongly correlated, we expect that $E(\phi_{BB-2})$ will be a significant improvement over $E(\phi_{D-DF})$. The SE for different detectors can be obtained via (7) (26), $E(\phi_{ML}) = 1.8$, $E(\phi_{BB-2}) = 1.8$, $E(\phi_{D-DF}) = 1.69$. The simulation result is given in Figure 3, which is consistent with the theoretical analysis.

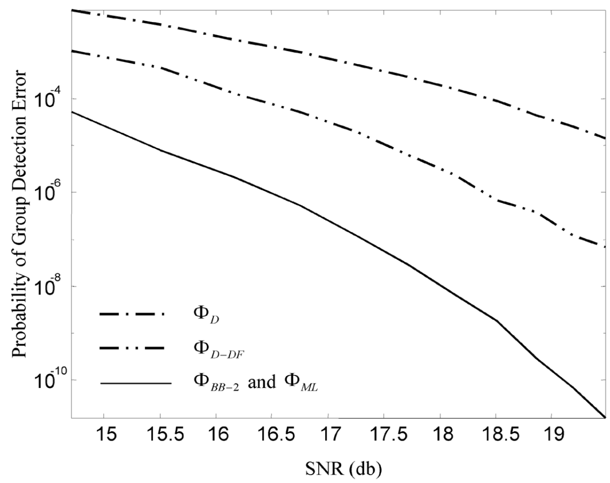

Fig. 3. Performance of various methods (3 users, 10000 Monte-Carlo runs)

Example 2: Now suppose we have 50 users. We use binary signature sequences of length 55. The signature sequences are generated such that $5 - 10$ users are correlated with each other (The maximum correlation among users is set to be around 0.85). The energy of each user is generated randomly between $[1, 4.5]$. In these cases, the proposed sub-optimal algorithm outperforms the D-DFD method significantly. In the situations when only a small number of users are correlated, the sub-optimal algorithms can even reach the performance bound of the optimal detector with marginal increases in computational cost over the D-DFD method. Figure 4 shows the simulation results of one of these examples. The comparison of the computational cost for the group detection of different algorithms is given in Table I.

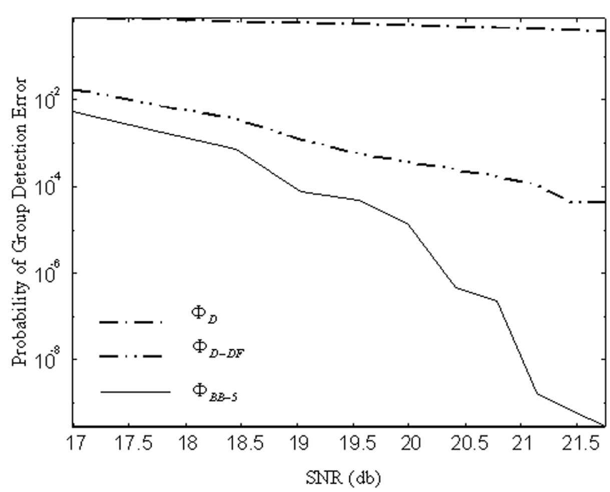

Fig. 4. Performance of various methods (50 users, spreading factor 55, 10000 Monte-Carlo runs)

Example 3: In another 8 user example, the H matrix is randomly generated as

	ϕ_D		ϕ_{D-DF}		ϕ_{BB-5}			
SNR (db)	$\times$	$+$	$\times$	$+$	$\times$ Average	$+$	$\times$ Maximum	$+$
17	2500	2450	1325	2550	1329.1	2568.5	1366	2674
18.5	2500	2450	1325	2550	1329	2568.1	1366	2674
19.5	2500	2450	1325	2550	1329	2568	1366	2674
21.1	2500	2450	1325	2550	1329	2568	1366	2674
21.8	2500	2450	1325	2550	1329	2568	1366	2674

TABLE I

Comparison of Computational Cost (50 users, spreading factor 55, 10000 Monte-Carlo runs, $\times$ = number of multiplications, $+$ = number of additions)

$$\begin{bmatrix} 3.0 & -0.4 & 1.4 & -0.5 & 0.4 & -0.3 & 0.3 & -0.6 \\ -0.4 & 1.9 & -0.8 & 0.0 & 0.7 & 0.6 & -0.5 & 0.2 \\ 1.4 & -0.8 & 2.8 & -1.8 & 0.8 & -0.0 & 0.0 & -0.3 \\ -0.5 & 0.0 & -1.8 & 2.6 & -1.6 & -0.6 & -0.6 & -0.3 \\ 0.4 & 0.7 & 0.8 & -1.6 & 2.2 & 1.2 & -0.0 & 0.2 \\ -0.3 & 0.6 & -0.0 & -0.6 & 1.2 & 1.4 & -0.0 & 0.2 \\ 0.3 & -0.5 & 0.0 & -0.6 & -0.0 & -0.0 & 1.2 & -0.2 \\ -0.6 & 0.2 & -0.3 & -0.3 & 0.2 & 0.2 & -0.2 & 1.0 \end{bmatrix}$$

The users have already been ordered by the order algorithm. The Symmetric Energy for the ML detector is

$$E(\phi_{ML}) = d_{min}^2 = 1.0 \tag{29}$$

The SE for various sub-optimal detectors are (for $E(\phi_{BB-1})$ through $E(\phi_{BB-7})$)

$$\{0.87, 0.87, 0.87, 0.87, 0.87, 0.89, 1.0\} \tag{30}$$

In this example, the computational cost for improving the performance from D-DFD is high. In addition, even the SE of the ML detector does not differ much from that of D-DFD. Hence, D-DFD is an efficient detector in this case. Figure 5 shows the probability of error for group detection.

VI. Conclusion

The proposed branch-and-bound algorithm shows that, in addition to the D-DFD method, there exists a class of sub-optimal methods that provides "any-time" sub-optimal solutions to the user. Given a CWMA system, the performance (measured by the Symmetric Energy), the computational bound and even the distribution of computational cost for the proposed sub-optimal algorithms can be estimated offline via (17) and (26). In addition, the detection sequence provided by the order algorithm is proved to be optimal for all the sub-optimal algorithms. The proposed algorithm can be

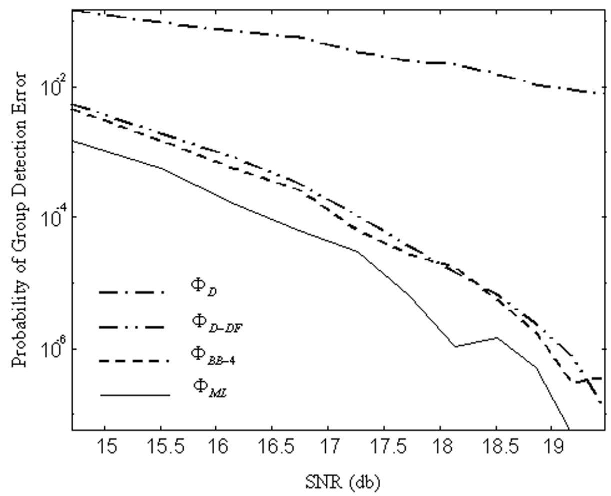

Fig. 5. Performance of various methods (8 users, 10000 Monte-Carlo runs)

easily extended to finite-alphabet signals instead of binary ones. The proposed computational method can also be easily extended and applied to group detection [7], and applied to the two-path tree search algorithm [14], which are the focus of future study.

Appendix

I. Proof of The Optimal Detection Sequence

To simplify the proof, we will first derive 2 useful lemmas.

Lemma 1: Suppose $H = L^T L$ is partitioned on two arbitrary diagonal elements as

$$\begin{bmatrix} H_{11} & H_{21}^T & H_{31}^T \\ H_{21} & H_{22} & H_{32}^T \\ H_{31} & H_{32} & H_{33} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix}^T \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \tag{31}$$

For any permutation matrix P of the same size as H_{22}, if

$$\begin{bmatrix} I & 0 & 0 \\ 0 & P & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} H_{11} & H_{21}^T & H_{31}^T \\ H_{21} & H_{22} & H_{32}^T \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ 0 & P & 0 \\ 0 & 0 & I \end{bmatrix} = \begin{bmatrix} \tilde{L}_{11} & 0 & 0 \\ \tilde{L}_{21} & \tilde{L}_{22} & 0 \\ \tilde{L}_{31} & \tilde{L}_{32} & \tilde{L}_{33} \end{bmatrix}^T \begin{bmatrix} \tilde{L}_{11} & 0 & 0 \\ \tilde{L}_{21} & \tilde{L}_{22} & 0 \\ \tilde{L}_{31} & \tilde{L}_{32} & \tilde{L}_{33} \end{bmatrix} \tag{32}$$

then the following results hold.

$$\begin{aligned}
&\tilde{L}_{11} = L_{11} \\
&\tilde{L}_{33} = L_{33} \\
&\tilde{L}_{22}^T \tilde{L}_{22} = P L_{22}^T L_{22} P
\end{aligned} \tag{33}$$

Proof:

From (32), we obtain

$$\begin{aligned}
&\tilde{L}_{11}^T \tilde{L}_{11} + \tilde{L}_{21}^T \tilde{L}_{21} + \tilde{L}_{31}^T \tilde{L}_{31} \\
&= L_{11}^T L_{11} + L_{21}^T L_{21} + L_{31}^T L_{31} \\
&\tilde{L}_{22}^T \tilde{L}_{21} + \tilde{L}_{32}^T \tilde{L}_{31} = P L_{22}^T L_{21} + P L_{32}^T L_{31} \\
&\tilde{L}_{22}^T \tilde{L}_{22} + \tilde{L}_{32}^T \tilde{L}_{32} = P L_{22}^T L_{22} P + P L_{32}^T L_{32} P \\
&\tilde{L}_{33}^T \tilde{L}_{31} = L_{33}^T L_{31} \\
&\tilde{L}_{33}^T \tilde{L}_{32} = L_{33}^T L_{32} P \\
&\tilde{L}_{33}^T \tilde{L}_{33} = L_{33}^T L_{33}
\end{aligned} \tag{34}$$

Substitute the last 3 equations into the other ones to obtain

$$\begin{aligned}
&\tilde{L}_{11}^T \tilde{L}_{11} + \tilde{L}_{21}^T \tilde{L}_{21} = L_{11}^T L_{11} + L_{21}^T L_{21} \\
&\tilde{L}_{22}^T \tilde{L}_{21} = P L_{22}^T L_{21} \\
&\tilde{L}_{22}^T \tilde{L}_{22} = P L_{22}^T L_{22} P \\
&\tilde{L}_{33} = L_{33}
\end{aligned} \tag{35}$$

Substitute the second and third equations into the first one to obtain

$$\tilde{L}_{11}^T \tilde{L}_{11} = L_{11}^T L_{11} \tag{36}$$

Hence Lemma 1 holds.

Lemma 2: Suppose H, $\tilde{H}$ and $\hat{H}$ are $K \times K$ symmetric and positive definite matrices, and

$$H = \tilde{H} + \hat{H} \tag{37}$$

Assume the Cholesky decompositions for H and $\tilde{H}$ are, respectively,

$$\begin{aligned}
&H = L^T L \\
&\tilde{H} = \tilde{L}^T \tilde{L}
\end{aligned} \tag{38}$$

Then, the following result holds for the diagonal elements of L and $\tilde{L}$

$$L_{ii} \geq \tilde{L}_{ii} \quad (\text{for } i = 1, ..., K) \tag{39}$$

Proof:

- It is obvious that Lemma 2 holds for 1×1 matrices.
- Suppose Lemma 2 is true for $(n-1) \times (n-1)$ matrices.

  Suppose matrices H, $\tilde{H}$ are of size $n \times n$. Partition H, $\tilde{H}$ and their Cholesky decomposisions on the last diagonal element as

$$\begin{bmatrix} H_{11} & H_{21}^T \\ H_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & l_{22} \end{bmatrix}^T \begin{bmatrix} L_{11} & 0 \\ L_{21} & l_{22} \end{bmatrix}$$
$$\begin{bmatrix} \tilde{H}_{11} & \tilde{H}_{21}^T \\ \tilde{H}_{21} & \tilde{h}_{22} \end{bmatrix} = \begin{bmatrix} \tilde{L}_{11} & 0 \\ \tilde{L}_{21} & \tilde{l}_{22} \end{bmatrix}^T \begin{bmatrix} \tilde{L}_{11} & 0 \\ \tilde{L}_{21} & \tilde{l}_{22} \end{bmatrix} \tag{40}$$

  Here h_{22}, $\tilde{h}_{22}$, l_{22} and $\tilde{l}_{22}$ are scalars. Since $\hat{H}$ is positive definite, we can find a positive definite matrix $\bar{H}$ with the Cholesky factor $\bar{L}$ (also partitioned on the last diagonal element) that satisfies

$$\begin{bmatrix} \bar{H}_{11} & \bar{H}_{21}^T \\ \bar{H}_{21} & \bar{h}_{22} \end{bmatrix} = \begin{bmatrix} \bar{L}_{11} & 0 \\ \bar{L}_{21} & \bar{l}_{22} \end{bmatrix}^T \begin{bmatrix} \bar{L}_{11} & 0 \\ \bar{L}_{21} & \bar{l}_{22} \end{bmatrix}$$
$$\begin{bmatrix} L_{11} & 0 \\ L_{21} & l_{22} \end{bmatrix}^T \begin{bmatrix} L_{11} & 0 \\ L_{21} & l_{22} \end{bmatrix} =$$
$$\begin{bmatrix} \tilde{L}_{11} & 0 \\ \tilde{L}_{21} & \tilde{l}_{22} \end{bmatrix}^T (I + \bar{H}) \begin{bmatrix} \tilde{L}_{11} & 0 \\ \tilde{L}_{21} & \tilde{l}_{22} \end{bmatrix} \tag{41}$$

  Using the fact that $l_{22}^2 = \tilde{l}_{22}^2(1 + \bar{l}_{22}^2)$, we obtain

$$L_{11}^T L_{11} = \tilde{L}_{11}^T \left[I + \bar{L}_{11}^T \bar{L}_{11} + \bar{L}_{21}^T \bar{L}_{21} \left(\frac{1}{1 + \bar{l}_{22}^2} \right) \right] \tilde{L}_{11} \tag{42}$$

  Therefore, according to the above assumptions and since $l_{22}^2 >= \tilde{l}_{22}^2$, Lemma 2 also holds for $n \times n$ matrices.

The proof is complete.

With the help of lemmas 1 and 2, the order algorithm can be equivalently stated as follows.

Equivalent Order Algorithm: Order users as follows: Among all the permutation matrices P, find the matrix P_1 and the concomitant Cholesky decomposition matrix $L^T L = P_1 H P_1$ such that L_{11} is maximized. Select the first user in $P_1 H P_1$ to be the first user of the new order (denote this user's index as i_1).

For $k = 2, ..., K$, form a new matrix $\hat{H}$ to be part of H that only contains the components $\{H_{ij}\}$ $(i, j \in \{\{1, ..., K\} - \{i_1, ..., i_{k-1}\}\})$. Among all the permutation matrices $\hat{P}$ of compatible size, find the matrix $\hat{P}_k$ and the Cholesky decomposition matrix $\hat{L}^T\hat{L} = \hat{P}_k\hat{H}\hat{P}_k$ such that $\hat{L}_{11}$ is maximized. Set the kth user of the new order to be the first user in $\hat{P}_k\hat{H}\hat{P}_k$ (denote this user's index as i_k).

Proof of equivalence: Firstly, for any permutation matrix $\hat{P}$, we have

$$(\hat{P}\hat{H}\hat{P})^{-1} = \hat{P}\hat{H}^{-1}\hat{P} \tag{43}$$

Hence

$$\min_{j=1,...,K-k+1}[\hat{H}^{-1}]_{jj} = \min_{j=1,...,K-k+1}\{[\hat{P}\hat{H}\hat{P}]^{-1}\}_{jj} \tag{44}$$

Secondly, since $[\hat{L}^{-1}]_{11} = \frac{1}{\hat{L}_{11}}$ and

$$\begin{aligned} \tfrac{1}{\hat{L}_{11}^2} &= [\hat{L}^{-1}]_{11}^2 \\ &= \{[\hat{P}\hat{H}\hat{P}]^{-1}\}_{11} \\ &\leq \min_{j=1,...,K-k+1}[\hat{H}^{-1}]_{jj} \end{aligned} \tag{45}$$

The equality holds when $\hat{P} = \hat{P}_1$, which means that the above order algorithm is equivalent to the one in (27) and (28).

We are now ready to prove proposition 2. Denote the detection sequence determined by the order algorithm as S. For an arbitrary detection sequence $\tilde{S}$, define $M_j^{\tilde{S}} = \min_{\tilde{S}}(|L_{11}|, ..., |L_{jj}|)$ and define $\tilde{S}_j$ to be the jth user in the sequence. The following proposition gives a stronger result than proposition 2.

Proposition 3: For any detection sequence $\tilde{S}$, suppose $\tilde{S}$ differs from S and their first difference begins at user i, that is,

$$\begin{aligned} &\tilde{S}_1 = S_1, ..., \tilde{S}_{i-1} = S_{i-1} \\ &\tilde{S}_i \neq S_i \end{aligned} \tag{46}$$

There exists a sequence $\hat{S}$ that is identical to S on at least the first i users, $(\hat{S}_1 = S_1, ..., \hat{S}_i = S_i)$ and satisfies,

$$M_j^{\hat{S}} \geq M_j^{\tilde{S}} \quad \forall j \in [1, K] \tag{47}$$

Proof: Accroding to (46), we can find $j > i$, that $\tilde{S}_j = S_i$. Construct $\hat{S}$ as,

1) Let $\hat{S} = \tilde{S}$

2) "Move" $\hat{S}_j$ to $\hat{S}_i$, which gives $\hat{S}_i = \tilde{S}_j = S_i$ and $\{\hat{S}_{i+1}, ..., \hat{S}_j\} = \{\tilde{S}_i, ..., \tilde{S}_{j-1}\}$.

Suppose the Cholesky decompositions corresponding to detection sequences $\tilde{S}$ and $\hat{S}$ are

$$\begin{aligned} \tilde{P}H\tilde{P} &= \tilde{L}^T\tilde{L} \\ \hat{P}H\hat{P} &= \hat{L}^T\hat{L} \end{aligned} \tag{48}$$

respectively. Partition $\tilde{L}$ and $\hat{L}$ on the $i-1$ and the jth diagonal elements as

$$\tilde{L} = \begin{bmatrix} \tilde{L}_{11} & 0 & 0 \\ \tilde{L}_{21} & \tilde{L}_{22} & 0 \\ \tilde{L}_{31} & \tilde{L}_{32} & \tilde{L}_{33} \end{bmatrix}$$

$$\hat{L} = \begin{bmatrix} \hat{L}_{11} & 0 & 0 \\ \hat{L}_{21} & \hat{L}_{22} & 0 \\ \hat{L}_{31} & \hat{L}_{32} & \hat{L}_{33} \end{bmatrix} \tag{49}$$

According to (46), Lemma 1, and the construction of $\hat{S}$, it is easy to see $\tilde{L}_{11} = \hat{L}_{11}$ and $\tilde{L}_{33} = \hat{L}_{33}$. Define

$$\tilde{C} = \tilde{L}_{22}$$

$$\hat{C} = \hat{L}_{22} \tag{50}$$

Partition $\hat{C}$ on the first diagonal element, and $\tilde{C}$ on the last diagonal element as

$$\tilde{C} = \begin{bmatrix} \tilde{C}_{11} & 0 \\ \tilde{C}_{21} & \tilde{c}_{22} \end{bmatrix}$$

$$\hat{C} = \begin{bmatrix} \hat{c}_{11} & 0 \\ \hat{C}_{21} & \hat{C}_{22} \end{bmatrix} \tag{51}$$

Notice that $\tilde{c}_{22}$ and $\hat{c}_{11}$ are scalars. According to Lemma 1, we have

$$\begin{bmatrix} \hat{c}_{11} & 0 \\ \hat{C}_{21} & \hat{C}_{22} \end{bmatrix}^T \begin{bmatrix} \hat{c}_{11} & 0 \\ \hat{C}_{21} & \hat{C}_{22} \end{bmatrix} =$$

$$\begin{bmatrix} 0 & 1 \\ I & 0 \end{bmatrix} \begin{bmatrix} \tilde{C}_{11} & 0 \\ \tilde{C}_{21} & \tilde{c}_{22} \end{bmatrix}^T \begin{bmatrix} \tilde{C}_{11} & 0 \\ \tilde{C}_{21} & \tilde{c}_{22} \end{bmatrix} \begin{bmatrix} 0 & I \\ 1 & 0 \end{bmatrix} \tag{52}$$

which gives,

$$\hat{C}_{22}^T \hat{C}_{22} = \tilde{C}_{11}^T \tilde{C}_{11} + \tilde{C}_{21}^T \tilde{C}_{21} \tag{53}$$

From Lemma 2, we know that the diagonal elements of $\hat{C}_{22}$ is larger than or equal to the corresponding diagonal elements of $\tilde{C}_{11}$. In addition, according to the equivalent order algorithm and the construction of $\hat{S}$, $\hat{c}_{11} \geq$ (The upper left component of $\tilde{C}_{11}$). Recall $\tilde{L}_{11} = \hat{L}_{11}$ and $\tilde{L}_{33} = \hat{L}_{33}$ in (49), it is evident that (47) is satisfied.

The proof is complete.

References

[1] S. Verdu, *Minimum Probability of Error for asynchronous Gaussian Multiple-Access Channel*, IEEE Trans. Inform. Theory, vol. IT-32, pp. 85–96, Jan. 1986.

[2] R. Lupas and S. Verdu, *Linear multiuser detectors for synchronous code-devision multiple-access channels*, IEEE Trans. Inform. Theory, vol. 35, pp. 123–136, Jan. 1989.

[3] M. K. Varanasi, *Multistage detection for asynchronous code-division multiple-access communications*, IEEE Trans. Comun., vol. 38, pp. 509–519, Apr. 1990.

[4] P.M. Pardalos, G.P. Rodgers, *Computational aspects of a branch and bound algorithm for quadratic zero-one programming*, Computing, vol. 45, pp. 131–144, 1990.

[5] M. K. Varanasi and B. Aazhang, *Near-optimum detection in synchronous code-division multiple access systems*, IEEE Trans. Comun., vol. 39, pp. 725–736, May 1991.

[6] Alexandra Duel-Hallen, *Decorrelating Decision-Feedback Multiuser Detector for Synchronous Code-Divison Multiple-Access Channel*, IEEE Trans. Comun., vol. 41, pp. 285–290, Feb. 1993.

[7] M. K. Varanasi, *Group Detection for Synchronous Gaussian Code-Division Multiple-Access Channels*, IEEE Trans. Inform. Theory, vol. 41, pp. 1083–1096, July 1995.

[8] Alexandra Duel-Hallen, *A Family of Multiuser Decision-Feedback Detectors for Asynchronous Code-Division Multiple-Access Channels*, IEEE Trans. Comun., vol. 43, pp. 421–432, Feb./Mar./April 1995.

[9] C. Schlegel and L. Wei, *A simple way to compute the minimum distance in multiuser CDMA systems*, IEEE Trans. Communications, vol. 45, pp. 532–535, May 1997.

[10] Dimitri P. Bertsekas, *Network Optimization, continuous and discrete models*, Athena Scientific, Belmont, Massachusetts, Chap. 10, pp. 483-492, 1998.

[11] Chandrasekar SanKaran, Anthony Ephremides, *Solving a Class of Optimum Multiuser Detection Problems with Polynomial Complexity*, IEEE Trans. Inform. Theory, vol. 44, pp. 1958-1961, Sep. 1998.

[12] S. Ulukus, R. Yates, *Optimum multiuser detection is tractable for synchronous CDMA systems using M-sequences*, IEEE Commun. Lett., vol. 2, pp. 89-91, 1998.

[13] M. K. Varanasi, *Decision feedback multiuser detection: a systematic approach*, IEEE Trans. Inform. Theory, vol. 45, pp. 219–240, Jan. 1999.

[14] Xiaodong Wang and Richard R. Spencer, *A Two-Path Tree Search Algorithm for Use with DFE*, IEEE Trans. Commun., vol. 48, pp. 913–916, June 2000.