Speed and Accuracy Comparison of Techniques for Multiuser Detection in Synchronous CDMA

Fumihiro Hasegawa, *Member, IEEE*, Jie Luo, *Member, IEEE*, Krishna R. Pattipati, *Fellow, IEEE*, Peter Willett, *Fellow, IEEE*, and David Pham, *Member, IEEE*

***Abstract*—In this letter, we compare the complexity and efficiency of several methods used for multiuser detection in a synchronous code-division multiple-access system. Various methods are discussed, including decision-feedback (DF) detection, group decision-feedback (GDF) detection, coordinate descent, quadratic programming with constraints, space-alternating generalized EM (SAGE) detection, Tabu search, a Boltzmann machine detector, semidefinite relaxation, probabilistic data association (PDA), branch and bound (BBD), and the sphere decoding (SD) method. The efficiencies of the algorithms, defined as the probability of group detection error divided by the number of floating point computations, are compared under various situations. Of particular interest is the appearance of an "efficient frontier" of algorithms, primarily composed of DF detector, GDF detector, PDA detector, the BBD optimal algorithm, and the SD method. The efficient frontier is the convex hull of algorithms as plotted on probability of error versus computational demands axes: algorithms not on this efficient frontier can be considered dominated by those that are.**

***Index Terms*—Code-division multiple access (CDMA), multiuser detection.**

I. Introduction

OVER the past decade, synchronous code-division multiple access (CDMA) has been analyzed intensely. For multiuser detection in a synchronous CDMA model, prior research has focused on designing suboptimal receivers with low computational complexity and better performance than a linear detector [15], [19]. Among them are multistage detectors [20], group detectors [21] and decision-feedback (DF) detectors [5], [6], [23]. Detectors based on a Voronoi diagram construction algorithm are proposed in [29]. Usually, suboptimal methods need to perform a projection to satisfy the integrality constraints on the solution domain, and this can significantly increase detection error.

Based on the idea of successive cancellation, a systematic DF detection approach was given in [23]. While maintaining the computational complexity of $\mathcal{O}(K^2)$, DF methods provide a significant improvement in probability of error when compared with the decorrelator [15] and with the traditional minimum mean-square error (MMSE)-based detector [19]. However, computer simulations show that, in most cases, and especially when some signature waveforms are strongly correlated, there is still a large gap between the probability of error of DF detection outputs and that of the optimal solution [23].

In this letter, we compare several methods used for multiuser detection in synchronous CDMA. We consider quadratic programming with constraints, branch and bound (BBD), coordinate descent, group decision making, semidefinite relaxation (SDR), Tabu search, the space-alternating generalized expectation-maximization (SAGE) algorithm, and probabilistic data association (PDA); these algorithms are capable of achieving near-optimal performance with relatively low computational complexity. Simulation results show that the PDA and several other methods can significantly outperform the DF detector or its group decision-feedback (GDF) counterpart, the GDF detector. We also show tradeoffs between computation time and accuracy of each algorithm and display a lower bound in the efficiency created by some of the algorithms mentioned above.

II. Problem Formulation

A discrete-time equivalent model for the matched-filter outputs at the receiver of a CDMA channel is given by the K-length vector [23]

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{b} + \boldsymbol{n} \tag{1}$$

where $\boldsymbol{b} \in \{-1, +1\}^K$ denotes the K-length vector of bits transmitted by the K users. When all the user signals are equally probable, the optimal solution of (1) is the output of a maximum-likelihood (ML) detector [15]

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

The ML detector has the property that it minimizes, among all detectors, the probability that not all users' decisions are correct (i.e., it minimizes the group detection error). Without a specific design of the signature waveforms, ϕ_{ML} is NP-hard and exponentially complex to implement [15]; the focus is then on developing optimal algorithms based on intelligent search and, if that is not feasible, easily implementable near-optimal integer programming algorithms for its solution.

In (1), $\boldsymbol{H} = \boldsymbol{W}\boldsymbol{R}\boldsymbol{W}$ is a positive definite signature waveform correlation matrix, $\boldsymbol{R}$ is the symmetric normalized correlation matrix with unit diagonal elements, $\boldsymbol{W}$ is a diagonal matrix whose kth diagonal element, w_k, is the square root of the received signal energy per bit of the kth user, and $\boldsymbol{n}$ is a real-valued zero-mean Gaussian random vector with a covariance matrix $\sigma^2\boldsymbol{H}$. Thus, the results of this letter apply *directly*

Paper approved by X. Wang, the Editor for Equalization of the IEEE Communications Society. Manuscript received April 30, 2002; revised September 6, 2003. This work was supported in part by the Office of Naval Research under Contract N00014-98-1-0465 and Contract N00014-00-1-0101, and in part by the Naval Undersea Warfare Center under Contract N66604-1-99-5021.

F. Hasegawa is with the Electrical and Computer Engineering Department, University of Toronto, Toronto, ON M5S 3G4, Canada.

J. Luo is with the Institute for Systems Research, University of Maryland, College Park, MD 20742 USA.

K. R. Pattipati, P. Willett, and D. Pham are with the Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT 06250 USA (e-mail: willett@engr.uconn.edu).

Digital Object Identifier 10.1109/TCOMM.2004.826410

only to the case of additive white Gaussian noise (AWGN) and users' powers precisely known and bits transmitted synchronously. However, synchronous CDMA over *fading* channels can also be modeled by (1), including the situation when the channel status is known to the receiver [23], as well as the situation when the channel status is estimated using Kalman filters [22]. Indeed, quite general situations involving *asynchronous* transmission (e.g., [6], [14]) and both asynchronicity and fading (e.g., [18]) are built around kernel algorithms that solve problems similar to (1); so we submit that our findings may have wider applicability than simply to the known-powers/synchronous case.

III. CDMA Detection by Various Algorithms

A. Matched Filter

The simplest suboptimal algorithm is a single-user matched filter [24]. It makes a decision based on the sign of the observation in (1)

$$\phi_{\text{Match}} : \hat{\boldsymbol{b}}_M = \text{sign}\,(\boldsymbol{y})\,. \tag{3}$$

Due to its poor performance it is not discussed further.

B. Decorrelator

The conventional decorrelation detector improves on the matched-filter output. The decision is found in two steps [15]. First, the unconstrained solution $\tilde{\boldsymbol{b}} = \boldsymbol{H}^{-1}\boldsymbol{y}$ is computed, and then this is projected onto the constraint set via $\hat{b}_i = \text{sign}(\tilde{b}_i)\ \forall\ i$. This method also performs poorly (more errors and higher computational load than DF), and hence, it is not included for comparison here.

C. DF Detector

DF detection based on the decorrelation detector is described in [23]: the method lowers the probability of detection error by applying a successive cancellation technique on users. The choice of user ordering is discussed in [23, Th. 1]. Conceptually, the "easiest," or the most powerful user, is detected first.

D. GDF Detector With Optimal Grouping

The idea of successive cancellation is that a correct decision on the strong users will improve the performance of weak users. In order to avoid an exponentially complex search among all users, it is intuitive to divide users into several groups; this was first introduced by Varanasi in [21], and enhanced in [12]. The grouping algorithm presented in [12] is optimal in the sense that it maximizes the lower bound of the performance measure [12] for every group, given the maximum group size. The details of the optimal grouping algorithm may be found in [12]. In the simulation, the maximum group size was set to three.

E. Quadratic Programming With Various Constraints

In this letter, we formulate the constraints in (2) in three ways: 1) $-1 \leq b_i \leq 1, \forall\ i$; 2) $b_i^2 = 1, \forall\ i$; 3) $\sum_{i=1}^K b_i^2 = K$. The quadratic programming problem with the first constraint was solved by the reflective Newton method [4], motivated by the positive definiteness of $\boldsymbol{H}$. For the minimization problem with the second and third constraint, a primal-dual method [3] was used. A quadratic programming detector with a constraint $\boldsymbol{b}^T\boldsymbol{b} < K$ is proposed in [25]. All of the detectors were followed by a hard-decision ($\hat{b}_i = \text{sign}(\tilde{b}_i)\,\forall\, i$) and were initialized using the output of DF detector. "Dual" values in the second and third approach were initialized with unconstrained decorrelator solution. The first approach is not shown in the results section, since it has often failed to make a correct decision, even when the initial solution was correct.

F. Coordinate Descent Family

The problem in (2) is equivalent to the following [10]:

$$\phi_{\text{cd}} : \hat{\boldsymbol{b}} = \arg \min_{\boldsymbol{x}\in\{0,1\}^K} \left(\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{c}^T\boldsymbol{x}\right) \tag{4}$$

where $\boldsymbol{x} = (\boldsymbol{b} + \boldsymbol{e}/2)$, $\boldsymbol{c} = -(\boldsymbol{y} + \boldsymbol{H}\boldsymbol{e})$, $\boldsymbol{Q} = 2\boldsymbol{H}$, and $\boldsymbol{e} = [11\ldots1]^T$. The problem in (4) is inherently equivalent to

$$\phi_{\text{cd}} : \hat{\boldsymbol{b}} = \arg \min_{\boldsymbol{x}\in\{0,1\}^K} \sum_{j=1}^{K}\sum_{i<j} q_{ij}x_ix_j + \sum_{i=1}^{K}\left(c_i + \frac{1}{2}q_{ii}x_i\right)x_i \tag{5}$$

where q_{ij} denotes the (i,j)th element of $\boldsymbol{Q}$. The *Descent I* and *Descent II* algorithms can be viewed as searching for a discrete "greedy" local minimizer in the neighborhood of the point $\boldsymbol{x} = [x_1,\ldots,x_K]$. The *Descent I* algorithm "flips" one bit per iteration, and *Descent II* flips up to two. Detailed descriptions for both *Descent I* and *Descent II* can be found in [10]. A detailed analysis on generalized local ascent search algorithms is presented in [28].

G. Tabu Search

Tabu search was first presented in [8], and is considered one of the most effective heuristic optimization techniques known today. This method has many modifications for specific approaches, and accordingly, we have modified it so that it suits the CDMA application. The Tabu search consists of three modes, namely, the ascent step search, the local minimum search, and the penalty mode. During the local minimum search, bit flips are performed to seek a minimum. Once a minimum that improves the current best cost is found, an ascent step will be taken subsequently so that such a step will guide us to a different local minimum. Following ascent or descent steps, a penalty will be imposed on the bit with which an improvement in cost function (2) is made.

H. Boltzmann Machine Algorithm

We consider the synchronous multiuser detection problem in (2) as a problem of state estimation of stochastic neurons in a Hopfield network [9]. A similar method has been proposed in [27]. In particular, we make a decision for each bit according to the following probability:

$$P\{b_i = +1\} = \frac{1}{1 + \exp\left(-\frac{\Delta E_i}{T}\right)} \tag{6}$$

where T is a design parameter and E_i is a change in the cost function (2) by flipping b_i from -1 to $+1$. A decision for each b_i is made as follows:

$$b_i = \begin{cases} +1, & \text{if } P\{b_i = +1\} > 0.5 + \delta \\ -1, & \text{if } P\{b_i = +1\} \leq 0.5 + \delta \end{cases} \tag{7}$$

where $\delta = \pm(T/T_0)\beta$, β is another design parameter such that $0 < \beta < 1$ and T_0 is the initial value of T. The polarity of δ is switched randomly for every bit. At each iteration, T is scaled by a factor α where $0 < \alpha < 1$.

I. SAGE Detector

New iterative multiuser receivers based on the expectation-maximization (EM) algorithm and the SAGE algorithm were proposed in [17]. In the SAGE algorithm, an estimate for each bit is updated sequentially to approximate the ML solution in (2). A detailed description of the algorithm is given in [17].

J. Genetic Algorithm

In [7], a multistage detector embedded into a genetic algorithm was proposed. It was shown that the proposed approach reduces the computational complexity by providing faster convergence. In our approach, bit flips are performed periodically to improve the solution provided by the genetic algorithm.

K. SDR

Semi-definite programs are convex optimization problems that unify several standard problems, e.g., linear and quadratic programming. It is shown in [16] that SDR is an accurate and efficient approximation method for certain NP-hard problems, and that it can approximately solve the ML detection problem in $\mathcal{O}(K^{3.5})$ computations. A detailed SDR algorithm is given in [16].

L. PDA

The PDA is one of the simplest and most effective approaches to target tracking [1]. The PDA idea can be applied to the CDMA model (1) as follows. The decisions on each user can be considered as binomial random variables, with the currently estimated probabilities for the bit from user i to be $+1$ or -1, P_{b_i} and $1-P_{b_i}$, respectively. By using a Gauss–Seidel iteration, the "soft" decisions are updated sequentially on all users. From (1), we have

$$\boldsymbol{H}^{-1}\boldsymbol{y} = \boldsymbol{b} + \boldsymbol{H}^{-1}\boldsymbol{n} = b_i\boldsymbol{e}_i + \left(\sum_{j\neq i} b_j\boldsymbol{e}_j + \boldsymbol{H}^{-1}\boldsymbol{n}\right) \tag{8}$$

where $\boldsymbol{e}_i$ is a column vector whose ith component is one, and whose other components are zero. When updating the probability for user i to be each of its possible values, the combination of interferences from other users are considered (approximately) as a Gaussian random vector, with the mean and variance for user j calculated according to the current decision probabilities as $(2P_{b_j}-1)$ and $4P_{b_j}(1-P_{b_j})$, respectively. We consequently obtain

$$P(y|b_i) \approx N\left(b_i\boldsymbol{e}_i + \left(\sum_{j\neq i}(2P_{b_j}-1)\boldsymbol{e}_j\right), \sum_{j\neq i} 4P_{b_j}(1-P_{b_j})\boldsymbol{e}_j\boldsymbol{e}_j^T + \sigma^2\boldsymbol{H}^{-1}\right) \tag{9}$$

in which $\mathcal{N}$ refers to the standard normal probability density function (pdf). We then update P_{b_i} and iterate. The technique is described in more detail in [13]. The PDA algorithm is shown below, and its computational burden is approximately $\mathcal{O}(K^3)$.

PDA Algorithm

1) Initialize $P_{b_i} = 0.5\ \forall\, i$. Set $b = [1, 1, \ldots, 1, 1]^T$.
2) Based on the current values of P_{b_j}, update P_{b_i}, $(i \neq j)$ as follows. Formulate a new model
$$\tilde{\boldsymbol{y}} = \boldsymbol{b} + \tilde{\boldsymbol{n}} \tag{10}$$
where $\tilde{\boldsymbol{y}} = \boldsymbol{H}^{-1}\boldsymbol{y}$. Now, the pdf of $\tilde{\boldsymbol{y}}$ is $\mathcal{N}(0, \sigma^2\boldsymbol{H}^{-1})$.
3) Next, repeat Step 3 and Step 4 $\forall\, i$
$$b_i\boldsymbol{e}_i = \tilde{\boldsymbol{y}} - \sum_{j\neq i} b_j\boldsymbol{e}_j. \tag{11}$$
Note that b_i is a scalar number. In addition, $\boldsymbol{e}_k$ denotes $[0, 0, \ldots, 1, \ldots, 0]^T$, where 1 is located at kth position. Next, we denote $[P_{b_1}, P_{b_2}, \ldots, P_{b_K}]^T$ as $\bar{\boldsymbol{P}}_b$, and denote $(\bar{P}_b)_i$ as the ith component of $\bar{\boldsymbol{P}}_b$. Also denote $\mathrm{diag}(\bar{\boldsymbol{P}}_b)_{ii}$ as a matrix with only the ith diagonal element of $\mathrm{diag}(\bar{\boldsymbol{P}}_{b_i})$ as follows:
$$\begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \cdots & \vdots & \vdots \\ 0 & \cdots & 4P_{b_i}(1-P_{b_i}) & \vdots & \vdots \\ \vdots & \cdots & \cdots & \ddots & 0 \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}.$$
Then we can write the pdf of (11) as
$$b_i\boldsymbol{e}_i \sim N\Big(\tilde{\boldsymbol{y}} - (2\bar{\boldsymbol{P}}_b - 1) + (2(\bar{\boldsymbol{P}}_b)_i - 1)\boldsymbol{e}_i, \sigma^2\boldsymbol{H}^{-1} + \mathrm{diag}(\bar{\boldsymbol{P}}_b) - \mathrm{diag}(\bar{\boldsymbol{P}}_b)_{ii}\Big). \tag{12}$$
4) Let $\boldsymbol{\Omega} = \sigma^2\boldsymbol{H}^{-1} + \mathrm{diag}(\bar{\boldsymbol{P}}_b)$. Approximate the inverse of covariance matrix in (12), which we will call $\boldsymbol{\Omega}_i^{-1}$, by using the Sherman–Morrison–Woodbury formula as follows:
$$\boldsymbol{\Omega}_i^{-1} = \boldsymbol{\Omega}^{-1} + 4P_{b_i}(1-P_{b_i})\frac{(\boldsymbol{\Omega}^{-1})_i(\boldsymbol{\Omega}^{-1})_i^T}{1 - 4P_{b_i}(1-P_{b_i})(\boldsymbol{\Omega}^{-1})_{ii}} \tag{13}$$
where $(\boldsymbol{\Omega}^{-1})_i$ and $(\boldsymbol{\Omega}^{-1})_{ii}$ correspond to the ith column and ith diagonal element of $\boldsymbol{\Omega}^{-1}$, respectively. Update P_{b_i} as follows:
$$P_{b_i} = 1 - \frac{1}{1 + e^{-2\boldsymbol{\theta}_i^T(\boldsymbol{\Omega}_i^{-1})_i}} \tag{14}$$
where $\left(\boldsymbol{\Omega}_i^{-1}\right)_i$ denotes the ith column of $\boldsymbol{\Omega}_i^{-1}$. $\boldsymbol{\theta}_i$ denotes the mean vector of the normal distribution in (12). Make a decision on user i using $b_i = +1$ if $P_{b_i} \geq 0.5$ and $b_i = -1$ if $P_{b_i} < 0.5$. If $i < K$, $i = i + 1$, and return to Step 3. If all P_{b_i} are updated, proceed to the next step.
5) Return to Step 3 if none of the P_{b_i} converge. If some of them do, proceed to Step 6 to remove them from the model in (10). If all users converge, stop.
6) Divide users into two groups, $\boldsymbol{b} = [\boldsymbol{b}_G^T \quad \boldsymbol{b}_{\bar{G}}^T]$, where P_{b_i} of the users in group G have converged, and those in $\bar{G}$ have not. Update the model in (1), and update (10) accordingly, as follows:
$$\boldsymbol{y}_{\bar{G}}\boldsymbol{H}_{\bar{G}\bar{G}}\boldsymbol{b}_{\bar{G}} = \boldsymbol{n}_{\bar{G}}$$
where $\boldsymbol{H}_{\bar{G}\bar{G}}$ denotes a portion of $\boldsymbol{H}$ matrix which corresponds to the users in $\bar{G}$. Return to Step 2.

M. BBD

The optimal solution to (2) can be obtained by interrogating each of the 2^K possible $\boldsymbol{b}$'s. There are intelligent ways to compute such combinations. In [11], an optimal algorithm based on the BBD method with an iterative lower bound update was proposed. It was shown that the proposed method can significantly decrease the average computational cost. Suppose $\boldsymbol{H} = \boldsymbol{L}^T\boldsymbol{L}$ is the Cholesky decomposition of $\boldsymbol{H}$. Then $\boldsymbol{H}^{-1} = \boldsymbol{L}^{-1}(\boldsymbol{L}^{-1})^T$. Therefore, as shown in [11], (2) is equivalent to

$$\phi_{\rm ML} : \hat{\boldsymbol{b}} = \arg \min_{\boldsymbol{b}\in\{-1,+1\}^K} \left\|\boldsymbol{Lb} - (\boldsymbol{L}^{-1})^T\boldsymbol{y}\right\|_2^2. \tag{15}$$

Denote $\tilde{\boldsymbol{y}} = (\boldsymbol{L}^{-1})^T\boldsymbol{y}$, $\boldsymbol{D} = \boldsymbol{Lb}$, and denote the kth component of $\boldsymbol{D}$ and $\tilde{\boldsymbol{y}}$ by D_k and $\tilde{y}_k$, respectively. Consequently, (15) becomes [11]

$$\phi_{\rm ML} : \hat{\boldsymbol{b}} = \arg \min_{\boldsymbol{b}\in\{-1,+1\}^K} \sum_{k=1}^{K} (D_k - \tilde{y}_k)^2. \tag{16}$$

Since $\boldsymbol{L}$ is a lower triangular matrix, d_k depends only on $(b_1, \ldots, b_k)$. When the decisions for the first k users are fixed, the term

$$\xi_k = \sum_{i=1}^{k} (D_i - \tilde{y}_i)^2 \tag{17}$$

becomes a lower bound of (16). Besides the use of the lower bound, the general BBD method [2] has several variations in searching the nodes, including the depth-first search, breadth-first search, best-first search, etc. In CDMA multiuser detection, since the observation vector $\boldsymbol{y}$ is generated from a statistical model (2), [11] proposed an efficient BBD-based algorithm that reduces the average computational cost significantly, compared with other optimal algorithms. Define u_k and d_k for user k as

$$\begin{aligned} u_k &= \arg \min_{0<i<k} (\forall\, i \le j < k,\ \ l_{jj} < l_{kk}) \\ d_k &= \arg \max_{k<i\le K} (\forall\, i \le j < k,\ \ l_{jj} \le l_{kk}). \end{aligned} \tag{18}$$

The BBD algorithm is shown below.

"Depth-First" BBD Algorithm

1) Order users according to [23, Th. 1], which is also presented in Proposition 2 of that paper; compute $\boldsymbol{y}$, $\boldsymbol{H}$, and $\boldsymbol{L}$ matrices for the ordered system; precompute the vectors $\boldsymbol{u}$ and $\boldsymbol{d}$, the components of which are defined by (18).
2) Precompute $\tilde{\boldsymbol{y}} = \boldsymbol{L}^{-T}\boldsymbol{y}$.
3) Initialize $k = 0$. $\boldsymbol{z_k} = \tilde{\boldsymbol{y}}$, $\xi_k = 0$, $\mathrm{UPPER} = +\infty$ and initialize K queues by $\forall\, k$, $q_k = \mathrm{NULL}$.
4) Set $k = k + 1$. For both nodes in level k ($b_k = -1, +1$), compute $[z_k]_k = [z_{k-1}]_k - \sum_{i=u_k-1}^{k-1} b_i l_{ki}$.
5) Choose the node in level k such that $b_k = \mathrm{sign}\,([z_k]_k)$. Compute $[z_k]_k = [z_k]_k - b_k l_{kk}$.
6) Compute $\xi_k = \xi_{k-1} + (D_k - \tilde{y}_k)^2 = \xi_{k-1} + [z_k]_k^2$.
7) If $\xi_k \ge \mathrm{UPPER}$ and not all the queues are empty, drop this node. Go to Step 11.
8) If $\xi_k < \mathrm{UPPER}$ and $k < K$, for both nodes in level k, precompute

$$\forall\, k < j \le d_k, \quad [z_k]_j = [z_k]_j - \sum_{i=u_k}^{k-1} b_i l_{ji}. \tag{19}$$

For node $b_k = -\mathrm{sign}\,([z_k]_k)$, do
   - Compute $[z_k]_k = [z_k]_k - b_k l_{kk}$ and $\xi_k = \xi_{k-1} + (D_k - \tilde{y}_k)^2 = \xi_{k-1} + [z_k]_k^2$.
   - If $\xi_k < \mathrm{UPPER}$, precompute

$$\forall\, k < j \le d_k, \quad [z_k]_j = [z_k]_j - b_k l_{jk}. \tag{20}$$

   Append the node to the tail of queue q_{m_k}, and store the associated k, ξ, and z_k together with this node.
   - Go to Step 4.
9) If $\xi_k < \mathrm{UPPER}$, $k = K$ and not all the queues are empty, update the "current-best" solution and $\mathrm{UPPER} = \xi_k$. Go to Step 11.
10) If $\xi_k < \mathrm{UPPER}$, $k = K$ and all the queues are empty, update the "current- best" solution and $\mathrm{UPPER} = \xi_k$. Go to Step 14.
11) Pick one node from the queues (note that we should check queues in the order of $q_{m_1}, \ldots, q_{m_K}$). Set k, ξ, and $\boldsymbol{z_k}$ equal to the stored values associated with this node.
12) If $\xi_k < \mathrm{UPPER}$, go to Step 4.
13) If $\xi_k \ge \mathrm{UPPER}$, and not all the queues are empty, go to Step 11.
14) Stop and report the "current-best" solution.

N. SD Approach

The SD algorithm is well known to provide optimal decoding and was applied to synchronous CDMA multiuser detection in [26]. As shown in [11], the SD algorithm actually falls into the BBD category. Due to the neglect of the statistical information in the model, the complexity of the SD method is significantly higher than the BBD algorithm introduced in the last section [11]. In this letter, the SD method is implemented as described in [11].

IV. Simulation Results

In this section, we compare the performance of algorithms described in Section III. The Tabu search (TABU), Boltzmann machine (BOLZ), Genetic algorithm (GENETIC), and variants of quadratic programming with different constraints:

- $b_i^2 = 1\ \forall_i\ (QPex1)$;
- $\sum_{i=1}^{K} b_i^2 = K(QPsxN)$;

which is initialized by the DF detection output. The SAGE algorithm (SAGE) is initialized with "soft-decorrelator" output, as done in [17].

We compare the efficiencies of the algorithms in terms of the probability of the group detection error versus the number of floating-point operations (FLOPS), with the probability of group detection error defined as the probability that not all user signals are detected correctly. It is important to note that the complexities of many of the algorithms compared above are problem dependent. In practical situations, the detector has to be designed for the worst-case complexity rather than the average complexity. Therefore, we compare the efficiencies of the algorithms in terms of performance versus both the average and worst-case complexity in the next two examples. For the BBD and SD methods, since they provide the best performance while their worst-case complexities grow exponentially in the number of users, it is of particular interest to see what their efficien-

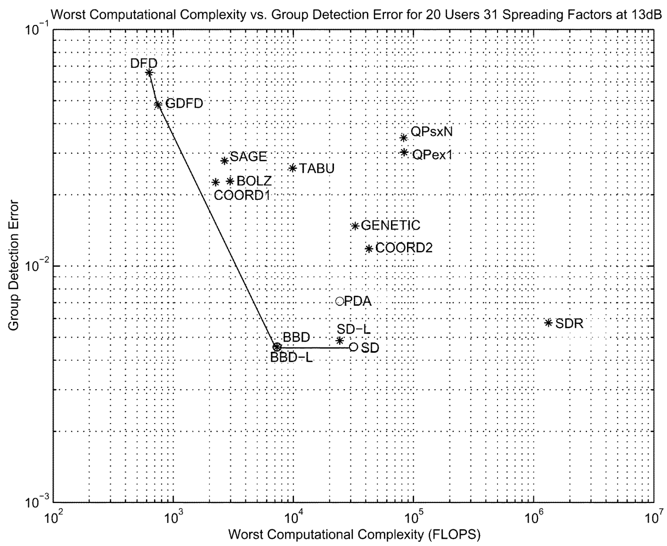

Fig. 1. Worst case: 20-user case.

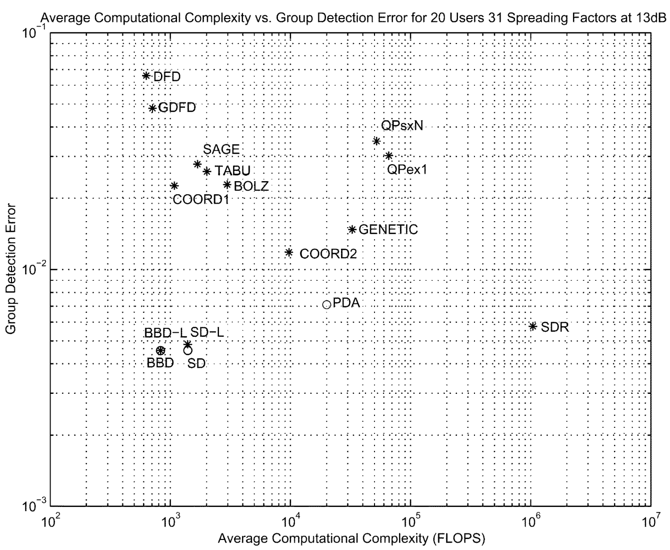

Fig. 2. Average case: 20-user case.

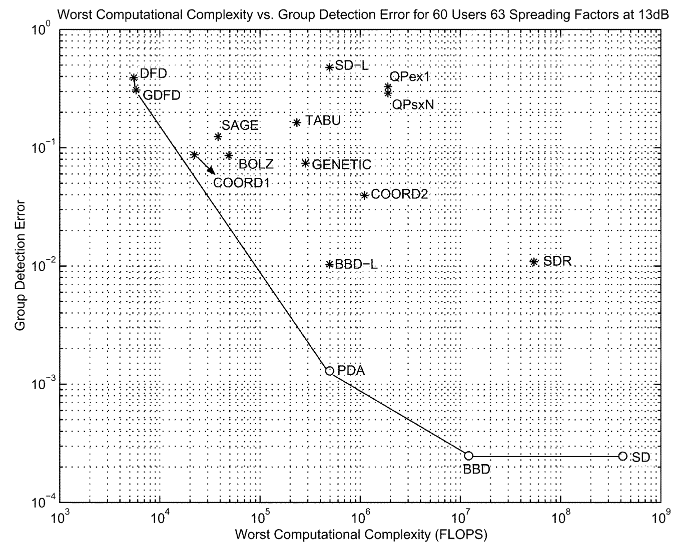

Fig. 3. Worst case: 60-user case.

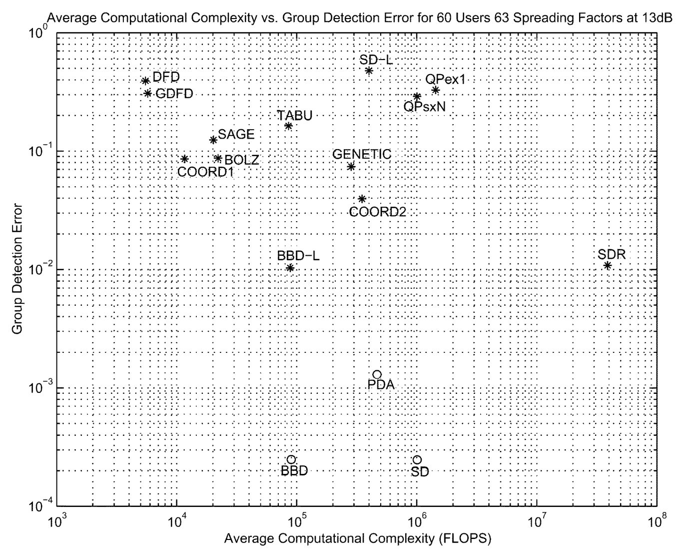

Fig. 4. Average case: 60-user case.

cies will be if we pose a strict upper bound to their worst-case complexity. The BBD and SD algorithms with strict complexity upper bound are labeled as BBD-L and SD-L, respectively. In the computer simulations, we set the worst-case complexity of the PDA algorithm as a strict complexity upper bound to BBD-L and SD-L. Whenever the complexity upper bound is reached, the algorithms stop searching and report the current best solution as the final decision. In every example, the correlation matrix used in the simulations are derived from binary random spreading sequences. The signal power for each user is set to be identical.

In the first example, we have 20 users with 31-length random binary signature sequences. The multiuser detection problem is considered "easy" in the sense that the number of users is relatively small and the spreading factor is much larger than the number of users. The worst and average computational complexity analyses are shown in Figs. 1 and 2, respectively. The "efficient frontier," composed of the DF, the GDF, the BBD, and the SD detectors, is shown in Fig. 1. We note that the BBD is able to provide the optimal performance at moderate average/worst computational complexity. We also note that upper bounding SD's computational complexity causes only slight degradation in its error performance.

In the second example, we have 60 users with 63-length random binary signature sequences. In this case, the multiuser detection problem is considered "hard" in the sense that the number of users is relatively large and is very close to the spreading factor (i.e., a bandwidth-efficient system [23]). The worst and average computational complexity analyses are shown in Figs. 3 and 4, respectively. Due to the NP-hard nature of optimal detection, the worst-case computational complexity of the two optimal algorithms increases significantly. Further, the performance of these algorithms suffers when one lays an upper bound on the number of computations. The PDA algorithm appears in the "efficient frontier" shown in Fig. 3.

One should note that, although the observation vector $\boldsymbol{y}$ does take arbitrary value, it is generated from the statistical model of (1). Algorithms such as the DF, GDF, SAGE, PDA, and BBD that use this statistical information in solving (2) appear close to the "efficient frontier."

V. Conclusions

In this letter, we have presented tradeoff plots between computational complexity and group detection error performance for various algorithms used for synchronous CDMA multiuser detection, and have noted the appearance of an "efficient frontier" of a very few approaches that warrant serious consideration. When the system size is small, or when the system is not bandwidth efficient, the multiuser detection problem can be solved efficiently via smart binary search algorithms. For bandwidth-efficient systems, or when the system size is large, then to get near-optimal performance, advanced algorithms such as the PDA method become preferable. In both cases, multiuser detection algorithms that use the statistical information appear close to the "efficient frontier."

References

[1] Y. Bar-Shalom and X. Li, *Estimation and Tracking: Principles, Techniques and Software*. Norwood, MA: Artech House, 1993.

[2] D. Bertsekas, *Network Optimization, Continuous and Discrete Models*. Belmont, MA: Athena Scientific, 1998, pp. 483–492.

[3] ——, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1999.

[4] T. Coleman and Y. Li, "A reflective Newton method for minimizing a quadratic function subject to bounds on some of the variables," *SIAM J. Optim.*, vol. 6, no. 4, pp. 1040–1058, Nov. 1996.

[5] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous code-division multiple-access channel," *IEEE Trans. Commun.*, vol. 41, pp. 285–290, Feb. 1993.

[6] ——, "A family of multiuser decision-feedback detectors for asynchronous code-division multiple-access channels," *IEEE Trans. Commun.*, vol. 43, pp. 421–432, Feb.-Apr. 1995.

[7] C. Ergun and K. Hacioglu, "Multiuser detection using a genetic algorithm in CDMA communication systems," *IEEE Trans. Commun.*, vol. 48, pp. 1374–1383, Aug. 2000.

[8] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Comput. Oper. Res.*, vol. 13, pp. 533–549, 1986.

[9] S. Haykin, *Neural Networks: A Comprehensive Foundation*. New York: Macmillan, 1994.

[10] J. Luo, K. Pattipati, P. Willett, and G. Levchuk, "A class of coordinate descent algorithms for multiuser detection," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, vol. 5, Istanbul, Turkey, June 2000, pp. 2853–2856.

[11] J. Luo, K. Pattipati, P. Willett, and L. Brunel, "Branch-and-bound-based fast optimal algorithm for multiuser detection in synchronous CDMA," in *Proc. IEEE Int. Conf. Communications*, vol. 5, Anchorage, AK, May 2003, pp. 3336–3340.

[12] J. Luo, K. Pattipati, P. Willett, and G. Levchuk, "Optimal grouping algorithm for a group decision-feedback detector in synchronous code-division multiaccess communications," *IEEE Trans. Commun.*, vol. 51, pp. 341–346, Mar. 2003.

[13] J. Luo, K. Pattipati, P. Willett, and F. Hasegawa, "Near-optimal multiuser detection in synchronous CDMA using probabilistic data association," *IEEE Commun. Lett.*, vol. 5, pp. 361–363, Sept. 2001.

[14] ——, "Optimal user ordering and time labeling for decision-feedback detection in asynchronous CDMA," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, vol. 3, Orlando, FL, May 2002, pp. 2317–2320.

[15] 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.

[16] W. Ma, T. Davidson, K. Wong, Z. Luo, and P. Ching, "Quasi-maximum-likelihood multiuser detection using semi-definite relaxation with application to synchronous CDMA," *IEEE Trans. Signal Processing*, vol. 50, pp. 912–922, Apr. 2002.

[17] L. Nelson and H. Poor, "Iterative multiuser receivers for CDMA channels: An EM-based approach," *IEEE Trans. Commun.*, vol. 44, pp. 1700–1710, Dec. 1996.

[18] D. Pham, J. Luo, K. Pattipati, and P. Willett, "A PDA-Kalman approach to multiuser detection in asynchronous CDMA," *IEEE Commun. Lett.*, vol. 6, pp. 475–477, Nov. 2002.

[19] C. Rushforth, R. Short, and Z. Xie, "A family of suboptimum detectors for coherent multiuser communications," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 683–690, May 1990.

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

[21] M. Varanasi, "Group detection for synchronous Gaussian code-division multiple-access channels," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1083–1096, July 1995.

[22] S. Vasudevan and M. Varanasi, "Achieving near-optimal asymptotic efficiency and fading resistance over the time-varying Rayleigh-faded CDMA channel," *IEEE Trans. Commun.*, vol. 44, pp. 1130–1143, Sept. 1996.

[23] M. Varanasi, "Decision-feedback multiuser detection: A systematic approach," *IEEE Trans. Inform. Theory*, vol. 45, pp. 219–240, Jan. 1999.

[24] S. Verdu, *Multi-User Detection*. New York: Cambridge Univ. Press, 1998.

[25] A. Yener, R. D. Yates, and S. Ulukus, "CDMA multiuser detection: A nonlinear programming approach," *IEEE Trans. Commun.*, vol. 50, pp. 1016–1024, June 2002.

[26] L. Brunel and J. Boutros, "Lattice decoding for joint detection in direct-sequence CDMA systems," *IEEE Trans. Inform. Theory*, vol. 49, pp. 1030–1037, Apr. 2003.

[27] Y. Sun, "A family of linear complexity likelihood ascent search detectors for CDMA multiuser detection," in *Proc. ISSSTA*, Princeton, NJ, Sept. 2000, pp. 713–717.

[28] ——, "Local maximum-likelihood multiuser detection for CDMA communications," in *Proc. ITCC*, Las Vegas, NV, Apr. 2001, pp. 307–311.

[29] Q. Li and C. N. Georghiades, "On a geometric view of multiuser detection for synchronous DS-CDMA channels," *IEEE Trans. Inform. Theory*, vol. 46, pp. 2723–2731, Nov. 2000.