

Multiuser Channel Estimation and Tracking for Long Code CDMA Systems

Srikrishna Bhashyam and Behnaam Aazhang

Srikrishna Bhashyam (skrishna@rice.edu) and Behnaam Aazhang (aaz@rice.edu) are with the Department of Electrical and Computer Engineering (MS-366), Rice University, 6100, S. Main St., Houston, TX 77005. This work was supported by Nokia Corporation, Texas Advanced Technology Program and National Science Foundation. **Submitted to IEEE Transactions on Communications, May 2000. Revised Jan 2001.**

Abstract

Channel estimation techniques for code-division multiple access (CDMA) systems need to combat multiple access interference (MAI) effectively. Most existing estimation techniques are designed for CDMA systems with short repetitive spreading codes. However, current and next generation wireless systems use long spreading codes whose period is much larger than the symbol duration. In this paper, we derive the maximum likelihood channel estimate for long code CDMA systems over multipath channels using training sequences and approximate it using an iterative algorithm to reduce the computational complexity in each symbol duration. The iterative channel estimate is also shown to be asymptotically unbiased. The effectiveness of the iterative channel estimator is demonstrated in terms of squared error in estimation as well as the bit error rate performance of a multistage detector based on the channel estimates. The effect of error in decision feedback from the multistage detector (used in the absence of training sequences) is also shown to be negligible for reasonable feedback error rates using simulations. The proposed iterative channel estimation technique is also extended to track slowly varying multipath fading channels using decision feedback. Thus, an MAI resistant multiuser channel estimation and tracking scheme with reasonable computational complexity is derived for long code CDMA systems over multipath fading channels.

I. INTRODUCTION

Code-division multiple access (CDMA) systems are inherently interference limited. Receivers can combat multiple access interference (MAI) by using multiuser channel estimation, detection and decoding algorithms. Several multiuser algorithms have been proposed for channel estimation in [1], [2], [3], [4], [5], [6]. These algorithms are developed for CDMA systems with short spreading codes for the various users that repeat every symbol. However, spreading codes used in practical CDMA systems have a period much larger than the symbol duration and are called *long spreading codes*. Therefore, most of the existing algorithms are either inapplicable or need prohibitive computational resources.

Recently, some channel estimation algorithms have been proposed in [7], [8], [9], [10], [11], [12] for long code CDMA systems. The techniques in [7] and [9] are based on the knowledge of the spreading sequences, channel estimates and bits of the interfering users, and they use the interference cancellation and the minimum mean squared error (MMSE) approach, respectively. In [8], an acquisition scheme for a single user entering the system is developed

using the knowledge of the spreading sequence and delays of the interfering users, who have already been acquired, without using their bit decisions. This leads to an estimator similar in complexity to the linear decorrelating detector. Blind estimation of the complex channel amplitudes is studied in [11] and [12] assuming the knowledge of the delays of the various propagation paths and [10] develops channel estimation algorithms for synchronous downlink channels.

In our paper, we develop a multiuser maximum likelihood (ML) channel estimation algorithm given the knowledge of the bits of all the users and approximate the solution directly using an iterative algorithm. The knowledge of the bits is obtained either using training sequences or decision feedback. Since the channel is time-varying, we update the channel estimates of all the users all the time and do not consider users to be acquired. The iterative approach allows the computation of the channel estimate using matrix multiplications during each bit duration and spreads the computation over the length of the preamble. Thus, this algorithm has reasonable complexity and should be implementable in practice. Also, we estimate the effective channel response of all the users simultaneously as a single vector and use it directly in detection instead of estimating the delays and amplitudes of each path separately [13]. We also show, using a simple example, how the channel estimate can be further improved, using the knowledge of the number of paths and reducing the size of the estimated channel response vector. The knowledge of the significant channel coefficients can also be derived statistically from the ML estimate [14], [15], although we do not describe it in our paper.

We show the effectiveness of the proposed channel estimation and tracking algorithms in two ways. We first show the gains achieved in mean squared error of the multiuser ML channel estimate as compared to a single-user channel estimate. Then, we simulate the bit error rate performance of a multiuser multistage detector [16] to show the gains in error rate performance using multiuser channel estimation. The multistage detector is chosen

due to its good performance-complexity trade-off. It requires only matrix multiplication computations in each processing window while other multiuser detection schemes like the linear decorrelating or MMSE detectors require the calculation of a matrix inverse during each processing window due to the time-varying nature of the spreading codes. The results indicate that iterative channel estimation algorithm can perform as well as ML channel estimation while reducing the complexity per bit duration by spreading the computation over the preamble duration or the processing window (during tracking). The proposed estimate is also shown to be asymptotically unbiased and the effect of errors in decision feedback, used in the absence of training sequences, on channel estimation is also studied using simulations.

The rest of the paper is organized as follows. In Section II, the system and received signal model are developed. Then, the maximum likelihood channel estimate is derived in Section III. The iterative algorithm is developed and discussed in Section IV. Section V presents the multiuser channel tracking scheme and extends the channel estimation results to time-varying channels. The simulation results are presented in Section VI to illustrate the effectiveness of the proposed techniques and the conclusions are in Section VII.

II. SYSTEM MODEL

We consider a K user asynchronous direct sequence CDMA system with long spreading codes. The spreading sequence corresponding to $b_{k,i}$, the i^{th} bit of the k^{th} user, is denoted by $c_{k,i}(t)$ and consists of N chips, where N is the spreading gain. The corresponding discrete chip sequence is denoted by $[c_{k,i}[1] \dots c_{k,i}[N]]$. The transmitted signal of the k^{th} user corresponding to an information sequence of length L is given in baseband format by

$$s_k(t) = \sqrt{E_k} \sum_{i=1}^L b_{k,i} c_{k,i}(t - iT), \quad (1)$$

where T is the bit duration and E_k is the transmitted power of the k^{th} user.

Let the channel be a multipath channel with P_k paths for the k^{th} user and let the complex attenuation and the delay with respect to the timing reference at the receiver of the p^{th}

path of the k^{th} user be denoted by $w_{k,p}$ and $\tau_{k,p}$ respectively. The received signal can be represented as

$$r(t) = \sum_{k=1}^K \sum_{p=1}^{P_k} w_{k,p} s_k(t - \tau_{k,p}) + n(t), \quad (2)$$

where $n(t)$ is the additive white Gaussian noise. The channel attenuations and delays are assumed to be constant during the estimation process. The maximum likelihood channel estimation technique provides an estimate of the effective channel impulse response (described later in the discrete received signal model) and not the estimates of the individual attenuations and delays. Therefore, the information about the number of paths of each user is not used in the derivation of the maximum likelihood estimate. This additional knowledge, if available, could be used in order to further improve the maximum likelihood estimate as in Section IV-C.

The received signal is discretized at the receiver by sampling the output of a chip-matched filter at the chip rate [1], [4], [13]. The observation vectors are formed by collecting N successive outputs of the chip-matched filter $r[n]$. The observation vectors correspond to a time interval equal to one symbol period and start at an arbitrary timing reference at the receiver. If we assume that all the paths of all the users are within one symbol period from the arbitrary timing reference, we will have only two symbols of each user in each observation window, and we can develop a representation similar to that in [13]. This model can be easily extended to include more general situations for the delays without affecting the derivation of the channel estimation algorithms [17]. The discrete received vector model is given by

$$\mathbf{r}_i = \mathcal{U}_i \mathbf{Z} \mathbf{b}_i + \mathbf{n}_i, \quad (3)$$

where \mathbf{r}_i is the i^{th} $N \times 1$ observation vector, \mathcal{U}_i is a $N \times 2K(N+1)$ spreading matrix, \mathbf{Z} is a $2K(N+1) \times 2K$ channel response matrix, \mathbf{b}_i is a $2K \times 1$ symbol vector and \mathbf{n}_i is a $N \times 1$ complex Gaussian zero-mean random vector with independent elements each of variance σ^2 . In particular, the spreading matrix, \mathcal{U}_i , is constructed using the shifted versions of the

spreading codes corresponding to the i^{th} and $i + 1^{th}$ symbols of each user in the observation window. Thus, \mathcal{U}_i is of the form $[\mathcal{U}_{1,i}^R \quad \mathcal{U}_{1,i+1}^L \quad \mathcal{U}_{2,i}^R \quad \mathcal{U}_{2,i+1}^L \quad \dots \quad \mathcal{U}_{K,i}^R \quad \mathcal{U}_{K,i+1}^L]$ where

$$\mathcal{U}_{k,i}^R = \begin{bmatrix} c_{k,i}[1] & c_{k,i}[2] & \dots & c_{k,i}[N] & 0 \\ c_{k,i}[2] & c_{k,i}[3] & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ c_{k,i}[N-1] & c_{k,i}[N] & \dots & 0 & 0 \\ c_{k,i}[N] & 0 & \dots & 0 & 0 \end{bmatrix},$$

is constructed with the right part of the spreading code of user k corresponding to symbol i and

$$\mathcal{U}_{k,i+1}^L = \begin{bmatrix} 0 & 0 & 0 & \dots & c_{k,i}[1] \\ 0 & 0 & 0 & \dots & c_{k,i}[2] \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & c_{k,i+1}[1] & \dots & c_{k,i}[N-1] \\ 0 & c_{k,i+1}[1] & c_{k,i+1}[2] & \dots & c_{k,i}[N] \end{bmatrix},$$

is constructed using the left part of the spreading codes of user k corresponding to symbol $i + 1$. Since the spreading codes change from symbol to symbol, the last columns of $\mathcal{U}_{k,i}^R$ and $\mathcal{U}_{k,i+1}^L$ are used additionally as compared to the short code case. The channel response matrix Z is of the form $\text{diag}(\mathbf{z}_1, \mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_2, \dots, \mathbf{z}_K, \mathbf{z}_K)$ where \mathbf{z}_k is the $(N+1) \times 1$ channel response vector for the k^{th} user. When rectangular chip waveforms of duration T_c are used, the $q_{k,p}^{th}$ and $(q_{k,p} + 1)^{th}$ element of \mathbf{z}_k have a contribution of $(1 - \gamma_{k,p})w_{k,p}$ and $(\gamma_{k,p})w_{k,p}$ from the p^{th} path of the k^{th} user, where $\tau_{k,p} = (q_{k,p} + \gamma_{k,p})T_c$. For example, when user k has only one path at delay $\tau_{k,1}$,

$$\mathbf{z}_k = [0 \quad \dots \quad 0 \quad (1 - \gamma_{k,1})w_{k,1} \quad (\gamma_{k,1})w_{k,1} \quad 0 \quad \dots \quad 0]^T \quad (4)$$

where the non-zero elements are at the $q_{k,1}^{th}$ and $(q_{k,1} + 1)^{th}$ positions. The symbol vector $\mathbf{b}_i = [b_{1,i} \quad b_{1,i+1} \quad b_{2,i} \quad b_{2,i+1} \quad \dots \quad b_{K,i} \quad b_{K,i+1}]^T$ has two symbols (chosen to be binary information bits ± 1 in this paper) corresponding to each user.

While equation (3) is used to represent the received vector for detection, we rewrite the received vector for channel estimation as

$$\mathbf{r}_i = \mathcal{U}_i \mathbf{B}_i \mathbf{z} + \mathbf{n}_i, \quad (5)$$

where $\mathbf{z} = [\mathbf{z}_1^\top \ \mathbf{z}_2^\top \ \dots \ \mathbf{z}_K^\top]^\top$ is a $(N+1)K \times 1$ channel response vector and \mathbf{B}_i is a $2K(N+1) \times (N+1)K$ matrix defined as

$$\mathbf{B}_i = \begin{bmatrix} b_{1,i} & 0 & 0 & \cdots & 0 & 0 \\ b_{1,i+1} & 0 & 0 & \cdots & 0 & 0 \\ 0 & b_{2,i} & 0 & \cdots & 0 & 0 \\ 0 & b_{2,i+1} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & & & & \vdots \\ \vdots & \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & 0 & b_{K,i} \\ 0 & 0 & 0 & \cdots & 0 & b_{K,i+1} \end{bmatrix} \otimes \mathbf{I}_{N+1},$$

where \otimes denotes the Kronecker product and \mathbf{I}_{N+1} is the identity matrix of rank $N+1$. Thus, we estimate $N+1$ channel parameters for each user. This effective channel response accounts for all the paths under the assumption that their delays are within one symbol duration. The number of non-zero coefficients in this effective channel response vector is determined by the number of paths and delays as in equation (4). Although the number of zero coefficients that are being estimated affects the overall estimation error, the location of them (which depends purely on the delays) does not affect the error. This is because the total response vector size is still $N+1$ for each user. In the following sections, we will develop MAI resistant methods to estimate \mathbf{z} from several observations of the received vector \mathbf{r}_i using the knowledge of the spreading codes and the transmitted bits.

III. MAXIMUM LIKELIHOOD CHANNEL ESTIMATION

In this section, we obtain the maximum likelihood (ML) estimate of the channel response of all the users (\mathbf{z}) using the knowledge of their spreading codes and transmitted bits. These

known bits could be available either as a preamble before the data or as bits in a separate pilot channel. In the estimation phase, training or pilot sequences are assumed to be used and in the tracking phase, which is discussed in Section V, data decisions from the detector are fed back to the estimator. The joint conditional distribution of L received observation vectors $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_L$, given the knowledge of the spreading sequences, channel and the bits is given by

$$p(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_L | \mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_L, \mathbf{B}_1, \mathbf{B}_2, \dots, \mathbf{B}_L, \mathbf{z}) = \frac{1}{(\pi\sigma^2)^{NL}} \exp \left\{ -\frac{1}{\sigma^2} \sum_{i=1}^L (\mathbf{r}_i - \mathcal{U}_i \mathbf{B}_i \mathbf{z})^H (\mathbf{r}_i - \mathcal{U}_i \mathbf{B}_i \mathbf{z}) \right\}.$$

The estimate $\hat{\mathbf{z}}_{ML}(L)$ that uniquely maximizes this likelihood function is the ML estimate and it satisfies the equation

$$\left\{ \sum_{i=1}^L (\mathcal{U}_i \mathbf{B}_i)^H (\mathcal{U}_i \mathbf{B}_i) \right\} \hat{\mathbf{z}}_{ML}(L) = \sum_{i=1}^L (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{r}_i. \quad (6)$$

For simplicity, we will denote $\frac{1}{L} \sum_{i=1}^L (\mathcal{U}_i \mathbf{B}_i)^H (\mathcal{U}_i \mathbf{B}_i)$ by \mathbf{R}_L , a $(N+1)K \times (N+1)K$ matrix, and $\frac{1}{L} \sum_{i=1}^L (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{r}_i$ by \mathbf{y}_L , a $(N+1)K \times 1$ vector. The rank of \mathbf{R}_L increases by N with each additional term $(\mathcal{U}_i \mathbf{B}_i)^H (\mathcal{U}_i \mathbf{B}_i)$ in the summation. This is based on the assumption that random spreading codes are used and the spreading codes over this duration are linearly independent. Therefore, for \mathbf{R}_L to be full rank (equal to $(N+1)K$), L should be at least equal to $K + \lceil K/N \rceil$. The current and next generation standards provide enough preamble or pilot resources to easily satisfy this condition. Therefore, assuming that \mathbf{R}_L is full rank, we can write

$$\hat{\mathbf{z}}_{ML}(L) = \mathbf{R}_L^{-1} \mathbf{y}_L. \quad (7)$$

Since \mathbf{r}_i is a jointly Gaussian random vector with mean $\mathcal{U}_i \mathbf{B}_i \mathbf{z}$ and covariance matrix $\sigma^2 \mathbf{I}$, any linear transformation $\mathbf{T} \mathbf{r}_i$ of \mathbf{r}_i is also a jointly Gaussian random vector with mean $\mathbf{T} \mathcal{U}_i \mathbf{B}_i \mathbf{z}$ and covariance matrix $\sigma^2 \mathbf{T} \mathbf{T}^H$. Using this property of Gaussian random vectors, we can

show that $\hat{\mathbf{z}}_{ML}(L)$ is also jointly Gaussian with mean \mathbf{z} and covariance matrix $\sigma^2 \mathbf{R}_L^{-1}/L$. This estimate has the following properties (similar to the properties of ML estimates described in [15], [18]):

1. $E[\hat{\mathbf{z}}_{ML}(L)] = \mathbf{z}$ (unbiased)
2. $E[(\hat{\mathbf{z}}_{ML}(L) - \mathbf{z})(\hat{\mathbf{z}}_{ML}(L) - \mathbf{z})^H] = \frac{\sigma^2 \mathbf{R}_L^{-1}}{L} = \text{Cramer-Rao Bound}$ (efficient)
3. $\lim_{L \rightarrow \infty} E[(\hat{\mathbf{z}}_{ML}(L) - \mathbf{z})^H (\hat{\mathbf{z}}_{ML}(L) - \mathbf{z})] = \lim_{L \rightarrow \infty} \frac{\sigma^2 \text{trace}(\mathbf{R}_L^{-1})}{L} = 0$ (consistent)

Later, we will compare this multiuser estimate with the single-user channel estimate given by

$$\hat{\mathbf{z}}_{SU} = \frac{1}{NL} \sum_{i=1}^L (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{r}_i. \quad (8)$$

This comparison will yield a better understanding of the significance of MAI in channel estimation and illustrate the MAI resistance of the multiuser channel estimate in (7). The MAI is significant even though we use long and random spreading codes, i. e., the matrix \mathbf{R}_L is not diagonal for realistic L . This is illustrated by the performance results shown in Section VI and the discussion about the eigenvalues of the correlation matrix \mathbf{R}_L as a function of L at the end of Section IV-A.

IV. ITERATIVE CHANNEL ESTIMATION

In this section, we will approximate the maximum likelihood estimate obtained in the previous section using iterative algorithms developed using gradient-based adaptation. Iterative algorithms based on the true gradient or an estimated stochastic gradient have been used earlier for various adaptive filtering and detection problems [19], [20]. We apply gradient-based adaptation techniques using the exact gradient in our problem of multiuser channel estimation. After we develop the iterative channel estimation algorithm, we will briefly discuss methods to improve the channel estimate for channels which are known to have very few non-zero elements in the channel response vector \mathbf{z} . Finally, we will discuss the computational complexity of the proposed iterative algorithms.

A direct computation of the exact ML channel estimate involves the computation of the correlation matrix \mathbf{R}_L and then the computation of $\mathbf{R}_L^{-1}\mathbf{y}_L$ at the end of the preamble. The direct computation of the inverse of the correlation matrix at the end of the preamble is computationally intensive and could delay the channel estimation process beyond the preamble duration and limit the information rate. In our iterative algorithms, we use the following ideas to approximate the ML solution. Firstly, we note that the product $\mathbf{R}_L^{-1}\mathbf{y}_L$ can be directly approximated by solving the linear equation $\mathbf{R}\hat{\mathbf{z}} = \mathbf{y}$ using iterative algorithms like the steepest descent algorithm. These iterative algorithms take advantage of the symmetry property of the autocorrelation matrix \mathbf{R}_L and reduce the complexity in the computation of the ML estimate given that we have \mathbf{R}_L and \mathbf{y}_L . Then, we also notice that the iterative algorithms can be modified to update the channel estimate as the preamble is being received instead of waiting till the end of the preamble. This means that the computation is spread over the whole preamble, thereby reducing the computational complexity per bit. We will now describe two iterative algorithms that make use of the ideas mentioned above to approximate the ML solution. First, we will describe a simple gradient descent algorithm with constant step size and then improve the speed of convergence using the steepest descent algorithm which chooses the optimal step size during each iteration [21].

A. Gradient Descent Method

The simple gradient descent algorithm performs the following computations during the l^{th} bit duration.

1. Compute $\mathbf{R}_l = \frac{l-1}{l}\mathbf{R}_{l-1} + \frac{1}{l}(\mathcal{U}_l\mathbf{B}_l)^H(\mathcal{U}_l\mathbf{B}_l)$
2. Compute $\mathbf{y}_l = \frac{l-1}{l}\mathbf{y}_{l-1} + \frac{1}{l}(\mathcal{U}_l\mathbf{B}_l)^H\mathbf{r}_l$
3. Update the estimate $\hat{\mathbf{z}}$ via

$$\hat{\mathbf{z}}^{(l)} = \hat{\mathbf{z}}^{(l-1)} - \mu(\mathbf{R}_l\hat{\mathbf{z}}^{(l-1)} - \mathbf{y}_l), \quad (9)$$

where $\mathbf{R}_l \hat{\mathbf{z}}^{(l-1)} - \mathbf{y}_l$ is the gradient of the squared error surface (corresponding to the exponent in the likelihood function that needs to be minimized) and μ should be chosen to ensure convergence and to control speed of convergence. In each iteration, the estimate of the channel is updated by taking a step along the gradient vector.

In this algorithm, the ML estimate for a preamble of length l is approximated as soon as the l^{th} bit is received. In fact, the updating step (i.e., step 3) can be repeated to improve accuracy. It can be repeated as many times as allowed by the available computational resources. In our paper, we will assume that this updating is done only once per bit. Therefore, the number of iterations is equal to the preamble length.

The mean of the iterative estimate ($\hat{\mathbf{z}}$) converges to the actual channel (\mathbf{z}), i.e., the estimate is asymptotically unbiased. This can be shown under the assumption that the eigen values $\{\lambda_l^{(j)}\}$ ($1 \leq j \leq (N+1)K$ and $l = 1, 2, \dots$) of \mathbf{R}_l can be bounded using positive real numbers α and β such that $\beta \geq \lambda_l^{(j)} \geq \alpha$ for all l and j . The proof in the appendix is done by bounding the sequence $E[\hat{\mathbf{z}}^{(l)}]$ by a converging geometric sequence in terms of α and μ , where μ is chosen to be lesser than $1/\beta$ to ensure convergence. For the case of random codes, the required property on the eigenvalues of \mathbf{R}_l are easily verified in the simulations. One sample set of eigenvalues are shown in Figure 1 to illustrate this. This plot of the maximum and minimum eigenvalues of \mathbf{R}_l as a function of l also illustrates that the MAI is significant for realistic l . Although \mathbf{R}_l is expected to get closer to the identity matrix as l goes to infinity, it cannot be considered diagonal for the values of l that are of interest. The maximum and minimum eigenvalues do not get close enough to approximate the matrix by an identity matrix.

Since the variance of the iterative estimate is very hard to compute analytically, it is estimated using simulations and compared with the Cramer-Rao bound which is satisfied by the ML estimate. Simulation results will be shown in Section VI to illustrate that the iterative estimate performs almost identical to the exact ML estimate for reasonable preamble

lengths.

B. Steepest Descent Method

In the simple gradient descent algorithm for channel estimation described in Section IV-A, the step size is chosen to be constant for all iterations. To speed up convergence, the step size can be chosen optimally for each iteration to minimize the squared error achieved by the updating step (i.e., step 3 which updates the channel estimate along the direction opposite to the gradient). This is achieved by the steepest descent algorithm [21]. The squared error function to be minimized is given by

$$\phi(\mathbf{z}) = \sum_{i=1}^L (\mathbf{r}_i - \mathcal{U}_i \mathbf{B}_i \mathbf{z})^H (\mathbf{r}_i - \mathcal{U}_i \mathbf{B}_i \mathbf{z}).$$

This squared error function is the negative of the log of the likelihood function defined in Section III ignoring the constants independent of the channel \mathbf{z} . Therefore, minimizing this squared error function is the same as maximizing the likelihood function. The squared error after the l^{th} iteration can be written in terms of the squared error after the $(l-1)^{\text{th}}$ iteration as

$$\phi(\hat{\mathbf{z}}^{(l)}) = \phi(\hat{\mathbf{z}}^{(l-1)} + \mu^{(l)} \mathbf{e}^{(l)}) = \phi(\hat{\mathbf{z}}^{(l-1)}) - \mu^{(l)} \mathbf{e}^{(l)H} \mathbf{e}^{(l)} + \frac{1}{2} \mu^{(l)2} \mathbf{e}^{(l)H} \mathbf{R}_l \mathbf{e}^{(l)} \quad (10)$$

where $\mathbf{e}^{(l)} = \mathbf{R}_l \hat{\mathbf{z}}^{(l-1)} - \mathbf{y}_l$. The optimal $\mu^{(l)}$ for the l^{th} iteration in the steepest descent algorithm can be obtained by minimizing the squared error in (10) with respect to $\mu^{(l)}$. The solution for this minimization is given by

$$\mu^{(l)} = \frac{\mathbf{e}^{(l)H} \mathbf{e}^{(l)}}{\mathbf{e}^{(l)H} \mathbf{R}_l \mathbf{e}^{(l)}}.$$

The optimal step size can be easily calculated using the knowledge of \mathbf{R}_l and the gradient. Therefore, the steepest descent algorithm can be implemented with the same information needed for the constant step size algorithm. Further speed up in convergence can be achieved by choosing the search directions in addition to choosing the step size for each iteration. This can be done by the conjugate gradient algorithm [21]. In the conjugate gradient algorithm,

the search direction in any iteration is chosen to be orthogonal to the search directions used in the previous iterations. The steepest descent algorithm does not ensure this since it uses the gradient directly as the search direction. However, the implementation of the conjugate gradient algorithm would require significant additional computation to obtain the search directions. We will show in Section VI that the simple gradient descent algorithm and the steepest descent algorithm perform very well in our case and the extra computation needed for using the conjugate gradient method may not be worth it.

C. Reduced Size Channel Estimate

In the estimation methods proposed above, the channel could have any number of paths with delays lying within one symbol period. All of these paths will be captured in the channel response vector \mathbf{z} . We do not assume any parametric model on the number of paths. However, in some practical scenarios where the number of paths is small and rectangular chip waveforms are used, we may not need the whole \mathbf{z} vector. For example, when there are just 2 paths for a user and the chip waveform is rectangular, the number of non-zero elements in \mathbf{z} corresponding to that user is at most 4. For other non-rectangular chip waveforms, more coefficients might be non-zero based on the autocorrelation of the pulse waveform used and the delays of the paths. For the rectangular pulse, the support of the autocorrelation function is only over the interval $[-T_c, T_c]$. If such information about the pulse shape and paths are available at the receiver, the iterative estimate obtained earlier can be further improved by using this knowledge. This information can be used to reduce the size of the estimated channel response vector $\hat{\mathbf{z}}$. One simple ad-hoc method to reduce the size of the estimated channel vector $\hat{\mathbf{z}}$ is to choose a few large coefficients of $\hat{\mathbf{z}}$. In particular, we choose a few large coefficients, say M , for each user. Thus, we are left with a smaller vector of size MK . If the elements that were truly zero were dropped by this procedure, the error in estimation of the zero elements would be made zero and the total squared error in the

estimate will be lower. Once the MK significant elements are chosen, the error in these MK elements can be improved by repeating the ML estimation with a new reduced model of the discrete received signal given by

$$\mathbf{r}_i = (\mathcal{U}_i \mathbf{B}_i)_r \mathbf{z}_r + \mathbf{n}_i,$$

where $(\mathcal{U}_i \mathbf{B}_i)_r$ is $N \times MK$ and \mathbf{z}_r is $MK \times 1$. The new Cramer-Rao bound will be $\sigma^2(\mathbf{R}_L)_r^{-1}/L$ and will be better than the original bound given by $\sigma^2 \mathbf{R}_L^{-1}/L$ since the size of the correlation matrix is smaller. In Section VI, we will show the excellent performance of this simple size reduction method described above without re-estimating the elements that are chosen to be significant. Other complex statistical tests to choose the significant coefficients from the ML estimate can be derived using the ideas in [14], [15] that use the statistical properties of the ML estimate described in Section III. These techniques would require more computation for possibly marginal performance improvement yielding an interesting complexity-performance trade-off.

D. Computational Complexity

Before we conclude the discussion on iterative channel estimation and discuss channel tracking, we will evaluate the computational complexity of the proposed schemes. In both the proposed iterative channel estimation schemes, the computations (additions and multiplications) during the l^{th} processing window for the three step update outlined above are

1. Computation of \mathbf{R}_l : $2N(N+1)^2K^2 + (N+1)^2K^2$
2. Computation of \mathbf{y}_l : $2N(N+1)K + (N+1)K$
3. Update of \mathbf{z} : $2(N+1)^2K^2 + 3(N+1)K$ for the simple gradient descent method and \mathbf{z} : $4(N+1)^2K^2 + 7(N+1)K$ for the steepest descent method (due to the additional computation to obtain μ).

The most complex step is the computation of \mathbf{R}_l which has a complexity of the order of $N(N+1)^2K^2$. However, this matrix multiplication involves only multiplications of ± 1 's

and this can be taken advantage of to speed up practical implementations [22]. The simple single-user estimate, $\hat{\mathbf{z}}_{SU}$, involves computations of the order of $N(N + 1)K$. Therefore, we have an additional factor of NK in complexity because of joint detection of all the $(N + 1)K$ channel parameters.

The simple algorithm for reducing the size of the estimated channel vector proposed in Section IV-C performs K sorts of size $N + 1$ for the K users. This has a complexity of $K(N + 1) \log(N + 1)$. This is lower than the order of complexity of the iterative algorithms and the single user estimation technique.

V. TRACKING TIME-VARYING CHANNELS

The iterative channel estimation scheme can be easily extended to track time-variations in the channel after the preamble. The channel is assumed to be approximately constant over the preamble duration and the tracking is performed by sliding the estimation window and using data decisions instead of training sequences. In particular, a multishot multistage detection [16] scheme is used in our case where bits are detected in blocks and the multiple access interference is canceled using a parallel interference cancellation technique. The MAI for each user is canceled using the estimated interference from the users obtained using their decisions and the knowledge of their spreading codes and channel estimate.

In the tracking scheme, the correlation matrix \mathbf{R}_L and the matched-filter outputs \mathbf{y}_L are averaged over a sliding window of length equal to the preamble length L , as defined in (7). The tracking is done as follows:

1. Detect bits using multishot multistage detection with previous channel estimate
2. Compute new correlation matrix and matched-filter vector : If \mathbf{R}_L^{old} corresponds to the old window over the time indices $T + 1, T + 2, \dots, T + L$ and D bits of each user are detected using multistage detection, then

$$\mathbf{R}_L^{new} = \mathbf{R}_L^{old} + \sum_{i=T+L+1}^{T+L+D} (\mathcal{U}_i \mathbf{B}_i)^H (\mathcal{U}_i \mathbf{B}_i) - \sum_{i=T+1}^{T+D} (\mathcal{U}_i \mathbf{B}_i)^H (\mathcal{U}_i \mathbf{B}_i)$$

$$\mathbf{y}_L^{new} = \mathbf{y}_L^{old} + \sum_{i=T+L+1}^{T+L+D} (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{r}_i - \sum_{i=T+1}^{T+D} (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{r}_i$$

3. Update channel estimate

$$\hat{\mathbf{z}}^{new} = \hat{\mathbf{z}}^{old} - \mu(\mathbf{R}_l^{new} \hat{\mathbf{z}}^{old} - \mathbf{y}_l^{new})$$

As discussed for the estimation scheme, the updating step can be repeated to improve estimation accuracy. Since the channel is assumed to be roughly constant over the window length, the ML channel estimate for the new window should be very close to the previous ML estimate. Therefore, in practice, we notice that one iteration per bit is sufficient, i.e., the channel estimate from the previous window is a very good initialization for both the simple gradient descent with constant step size and the steepest descent algorithm to estimate the new channel estimate.

VI. NUMERICAL RESULTS AND DISCUSSION

In this section, we will show simulation results to illustrate the effectiveness of the channel estimation and tracking techniques developed in the Sections III, IV and V. We will first show the performance gains for a constant multipath channel from multiuser channel estimation in Section VI-A and later show the significant gains for a slowly fading multipath channel with multiuser channel estimation and tracking in Section VI-B. For each simulation, the complex fading coefficients for each path in equation (2) are generated randomly according to a complex Gaussian distribution (corresponding to Rayleigh fading) and the delays are generated according to a uniform distribution from zero to one symbol duration. For the constant multipath channels, the fading coefficients of all the paths are constant throughout the transmission and are of equal magnitude. For the multipath fading channels, the average power of the paths are equal and the instantaneous fading coefficients for each path are generated using the Jakes fading model [23]. In this case, the fading coefficients change even within one frame of transmission.

The significant gains provided by multiuser estimation techniques compared to single user estimation will be shown by comparing the mean square error in the channel estimates as well as the bit error rate of a multistage detector that uses the estimated channel. After demonstrating the excellent performance of the proposed multiuser channel estimation and tracking methods for realistic multipath fading channels, we will finally show using simulations, in Section VI-C, that the effect of decision feedback error on the performance of multiuser channel estimation is negligible for realistic simulation conditions. This is done to demonstrate that decision feedback can be successfully used for channel estimation when training sequences are not available.

A. Performance of Multiuser Channel Estimation

The exact ML, iterative and single user channel estimation methods are simulated for a 16-user asynchronous direct sequence CDMA system with spreading gain 31. All users are assumed to have equal power and have two path channels to the receiver. The channel is estimated using a 100 bit preamble.

Figure 2 shows the improvement in average squared error (over 200 simulations) of the various channel estimates - single user channel estimate from (8), iterative estimate using the gradient descent algorithm with $\mu = 0.002$, iterative estimate using steepest descent and the exact ML estimate - with preamble length. The simulation results show the superior performance of the multiuser estimators compared to the single-user estimator. The simulations also show that the iterative estimate performs almost as well as the ML estimate and can be further improved by performing more iterations after each bit is received or using the steepest descent method. The steepest descent method achieves the performance of the actual ML estimate for fewer iterations (about 40). For realistic preamble lengths that are needed to get a normalized mean squared of at least 0.1 (10%), the iterative algorithms can match the performance of the exact ML estimate while spreading the computation over the

entire length of the preamble. Thus, they reduce the computational resources needed per bit and provide the channel estimates at the end of the preamble without any further processing delay. The Cramer-Rao bound is also shown to illustrate the fact that the ML estimate is efficient.

Figure 3 shows the bit error rate performance of the multistage detector with signal to noise ratio (SNR) for the following channel estimation methods - single-user estimation, iterative estimation using constant step size μ , exact ML estimation and actual channel knowledge. We show only one of the iterative channel estimation methods since their performance is almost the same for the preamble length of 100 considered here. The preamble length is chosen to achieve reasonably low channel estimation error. Trends similar to those observed in the mean squared error comparison of the various channel estimates in Figure 2 can be seen in the bit error rate comparison in Figure 3 too. As expected, there is a significant gain in performance achieved by using the iterative multiuser channel estimator over the single-user estimator. Also, the performance of the multistage detector with the iterative estimate is virtually the same as the performance with the ML estimate. It is also worth noting that this result shows significant gains even in the equal power case. When users have different powers, i.e., in near-far situations, MAI can further degrade the single-user estimate. Figure 4 shows the normalized mean squared error of the channel estimate for the weak user (user 1) and strong users (user 2-15) for both the single-user channel estimate and the iterative multiuser estimate. The strong users have 6 dB more power than the weak user. It can be seen that the normalized mean squared error for the single-user channel estimate has a floor for both the weak and the strong users, with a lower floor for the strong user. This is because multiuser interference is not mitigated. However, for the iterative multiuser estimate, the normalized mean squared error decreases with increasing SNR and the performance is not interference-limited. In fact, the mean squared error follows the linear relationship described in property 3 of the ML estimate (in Section III) between the mean squared error and the

noise variance and reduces to zero as SNR increases. The bit error rate performance of the weak user is shown in Figure 5. The performance of the multistage detector is significantly affected by using single-user channel estimates that are not near-far resistant. However, the proposed iterative multiuser channel estimates can perform well even in near-far situations.

Finally, we illustrate by a simple example how the reduction in size of the channel response vector described in Section IV-C improves estimation performance. Figure 6 shows the bit error rate performance of a multistage detector with the following channel estimates - reduced size single user estimate, reduced size iterative estimate and actual channel parameters. The reduction in size is done for both the single user and iterative estimates by just choosing the four largest channel coefficients for each user. The preamble length is reduced to 50 to take advantage of the lower channel estimation error. The simulations show that the performance of the multistage detector with reduced size iterative channel estimation is closer to the performance with the actual channel than the original iterative estimate in Figure 3. The single-user estimate also performs better when its size is reduced to use the knowledge of the number of paths. However, it still performs significantly worse than the iterative multiuser method.

B. Performance of Multiuser Channel Tracking

We will now present the simulation results for the single user and iterative multiuser tracking techniques for a 8-user CDMA system with a spreading gain of 16 employing a multiuser multistage detector. All users have two paths and are assumed to travel at 10 kilometers per hour corresponding to a slowly fading channel with Doppler spread of 17.67 Hz. Their data is transmitted in 10 ms frames consisting of 2400 bits. Each frame has a preamble of length 100 and the channel for the rest of the frame is tracked using decision feedback from the multistage detector. Figure 7 shows the tracking performance in terms of the bit error rate performance of the multistage detector with the various channel estimation

and tracking methods. The iterative multiuser estimator is able to track the fading channel much better than a conventional single-user estimator (about 3 dB gain in performance). Performance gains similar to those observed in the constant multipath case in Figure 3 are seen for the multipath fading scenario in Figure 7 as well. To give an idea of the variations in the channel and in the estimates we show in Figure 8 the tracking performance of the algorithm at 8 dB SNR for a single element of the channel response vector. The tracking is done in this case for a duration of 4000 bits. We can see that the iterative channel estimate is able to track the channel variations much better than the single-user channel estimate, thereby giving the gains in bit error rate performance. Similar performance can be observed for the other coefficients as well.

C. Effect of Decision Feedback error on Channel Estimation

In the simulation results for channel estimation, the bits of all the users are assumed to be known. In practice, this knowledge is perfect for the users entering the system (estimation phase) since they can be assumed to use training sequences that are known to the receiver. For the other users in the tracking phase, the bit estimates have to be obtained from the detector or decoder. This decision feedback could be erroneous. However, the basis of our approach is that this feedback is “reasonably accurate”. Here, we will study the effect of possible errors in the feedback on estimation.

The multiuser channel estimate is obtained by solving the following linear equation,

$$\mathbf{R}\hat{\mathbf{z}} = \mathbf{y}. \quad (11)$$

Here, we have dropped the subscript L denoting the length of the preamble for convenience. When an incorrect decision is fed back we end up solving a different equation

$$(\mathbf{R} + \Delta\mathbf{R})(\hat{\mathbf{z}} + \Delta\hat{\mathbf{z}}) = \mathbf{y} + \Delta\mathbf{y} \quad (12)$$

obtained by perturbing the matrix \mathbf{R} and vector \mathbf{y} . The relative error in the channel estimate

due to erroneous feedback can be bounded by the relative error in \mathbf{R} and \mathbf{y} using the condition number \mathcal{K} of the matrix \mathbf{R} . Suppose $\Delta\mathbf{R} = \epsilon\mathbf{E}$ and $\Delta\mathbf{y} = \epsilon\mathbf{e}$, where \mathbf{E} and \mathbf{e} are arbitrary matrices (of appropriate dimension), then from [21], [24] we have,

$$\frac{\|\Delta\hat{\mathbf{z}}\|}{\|\hat{\mathbf{z}}\|} \leq \mathcal{K} \left(\frac{\|\Delta\mathbf{R}\|}{\|\mathbf{R}\|} + \frac{\|\Delta\mathbf{y}\|}{\|\mathbf{y}\|} \right) + \mathcal{O}(\epsilon^2). \quad (13)$$

From (13), we can see that as long as the condition number of \mathbf{R} is low the relative error in $\hat{\mathbf{z}}$, denoted by $\Delta\hat{\mathbf{z}}$, is bounded by a reasonably small multiple of the relative error in \mathbf{R} and \mathbf{y} . In our simulation example shown in Figure 1, the condition number of \mathbf{R} for a preamble length of 100 is 5.75. The error in \mathbf{R} and \mathbf{y} will be small if the number of feedback errors is small compared to the number of preamble bits.

Figures 9 and 10 show the effect of feedback error on the performance of the iterative channel estimator in terms of the mean squared error and bit error rate. The results are shown for two values of feedback error rates (i.e., bit error rate of the multistage detector used in decision feedback) - 0.02 and 0.04. As can be seen, the effect of erroneous feedback is not very significant, especially in terms of the bit error rate. The feedback error rate is chosen to be constant for simplicity in simulations. Ideally, the feedback error rate should be dependent on the SNR. However, we choose a feedback error rate that is approximately the error rate at $6dB$. This gives a conservative performance result for higher SNR's. Feedback error rates that vary with SNR are used in the channel tracking results in Section VI-B.

VII. CONCLUSIONS

In this paper, we derive the maximum likelihood channel estimate for multiple users in a CDMA system with long spreading codes using training sequences or decision feedback. Then, we approximate the maximum likelihood estimate using an iterative algorithm to reduce the computational complexity. The channel estimate is near-far resistant and is determined as the effective channel impulse response which can be directly used in multiuser detection and decoding. We show that the iterative channel estimate is asymptotically

unbiased, i.e., the mean of the iterative estimate asymptotically converges to the actual channel as the training sequence length increases. Simulations are used to illustrate the significant performance gains achievable using multiuser channel estimation as opposed to single user channel estimation techniques used in current cellular systems. The results are shown in terms of mean squared error in channel estimates as well as the bit error rates of a multistage detector using the various channel estimates. The simulations also show that the iterative scheme can perform as well as the maximum likelihood estimation method with reasonable computational complexity per bit comprising mainly of a matrix multiplication. This matrix multiplication involves only multiplications of ± 1 's and this can be taken advantage of to speed up practical implementations [22]. The proposed iterative scheme is also extended to track fading channel variations using decision feedback. A simple method to reduce the size of this estimated channel response vector was also demonstrated to take advantage of the knowledge of the number of paths. In fact, methods to statistically determine the size of the channel response vector required for detection from the ML estimate can be derived [14], [15]. Thus, we see that the combined channel estimation and tracking scheme can effectively estimate and track the channels of multiple users and provide significant performance gains over currently used single-user techniques for realistic multipath fading channels.

APPENDIX

Proposition: Let \mathbf{R}_l be the correlation matrix corresponding to a preamble length l in the CDMA system described above and let $\lambda_l^{(1)}, \lambda_l^{(2)}, \dots, \lambda_l^{((N+1)K)}$ be its eigenvalues. If there exist positive real numbers α and β such that $\beta \geq \lambda_l^{(j)} \geq \alpha$ for all l and j , then there exists a real number μ such that

$$\lim_{l \rightarrow \infty} E[\hat{\mathbf{z}}^{(l)}] = \mathbf{z}$$

when $\hat{\mathbf{z}}^{(l)}$ is updated according to equation (9).

Proof: From equation (9), we can write

$$\hat{\mathbf{z}}^{(l)} - \mathbf{z} = \hat{\mathbf{z}}^{(l-1)} - \mathbf{z} - \mu(\mathbf{R}_l \hat{\mathbf{z}}^{(l-1)} - \mathbf{y}_l)$$

Since $\mathbf{y}_l = \frac{1}{l} \sum_{i=1}^l (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{r}_i = \mathbf{R}_l \mathbf{z} + \frac{1}{l} \sum_{i=1}^l (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{n}_i$ we get

$$\hat{\mathbf{z}}^{(l)} - \mathbf{z} = (\mathbf{I} - \mu \mathbf{R}_l)(\hat{\mathbf{z}}^{(l-1)} - \mathbf{z}) + \frac{\mu}{l} \sum_{i=1}^l (\mathcal{U}_i \mathbf{B}_i)^H \mathbf{n}_i$$

Because the noise is zero-mean, we have

$$E[\hat{\mathbf{z}}^{(l)}] - \mathbf{z} = (\mathbf{I} - \mu \mathbf{R}_l)(E[\hat{\mathbf{z}}^{(l-1)}] - \mathbf{z}) \quad (14)$$

Now, \mathbf{R}_l is a symmetric matrix and can be expressed using the eigenvalue decomposition as $\mathbf{Q}_l \mathbf{\Lambda}_l \mathbf{Q}_l^\top$, where \mathbf{Q}_l is a unitary matrix and $\mathbf{\Lambda}_l$ is a diagonal matrix of the eigenvalues of \mathbf{R}_l .

Therefore,

$$\mathbf{Q}_l^\top (E[\hat{\mathbf{z}}^{(l)}] - \mathbf{z}) = (\mathbf{I} - \mu \mathbf{\Lambda}_l) \mathbf{Q}_l^\top (E[\hat{\mathbf{z}}^{(l-1)}] - \mathbf{z}).$$

We can choose μ such that $\mu < \frac{1}{\beta}$. Equivalently, $1 > 1 - \mu\alpha > 0$. Since \mathbf{Q}_l is a unitary matrix and $\lambda_l^{(j)} \geq \alpha$, we have

$$\|E[\hat{\mathbf{z}}^{(l)}] - \mathbf{z}\| \leq (1 - \mu\alpha) \|E[\hat{\mathbf{z}}^{(l-1)}] - \mathbf{z}\|$$

where $\|\cdot\|$ denotes the \mathcal{L}_2 norm of a vector. Since $1 > 1 - \mu\alpha > 0$, $\|E[\hat{\mathbf{z}}^{(l)}] - \mathbf{z}\|$ converges to 0 as $l \rightarrow \infty$. \square

REFERENCES

- [1] S. E. Bensley and B. Aazhang, "Subspace-based channel estimation for code division multiple access communication systems," *IEEE Transactions on Communications*, vol. 44, no. 8, pp. 1009–1020, Aug 1996.
- [2] S. E. Bensley and B. Aazhang, "Maximum-likelihood synchronization of a single user for code-division multiple-access communication systems," *IEEE Transactions on Communications*, vol. 46, no. 3, pp. 392–399, Mar 1998.
- [3] E. G. Strom, S. Parkvall, S. L. Miller, and B. E. Ottersten, "Propagation delay estimation in asynchronous direct-sequence code-division multiple access systems," *IEEE Transactions on Communications*, vol. 44, no. 1, pp. 84–93, Jan 1996.

- [4] R. Madyastha and B. Aazhang, "Antenna arrays for joint maximum likelihood parameter estimation in CDMA systems," in *Proceedings CISS, Baltimore, MD*, March 1997, vol. 2, pp. 984–988.
- [5] C. Sengupta, J. R. Cavallaro, and B. Aazhang, "Subspace-based tracking of multipath channel parameters for cdma systems," *European Transactions on Telecommunications, Special issue on CDMA techniques for wireless communication systems*, vol. 9, no. 5, pp. 439–447, Sep-Oct 1998.
- [6] C. Sengupta, A. Hottinen, J. R. Cavallaro, and B. Aazhang, "Maximum likelihood multipath channel parameter estimation in CDMA systems," in *Proceedings CISS, Princeton, NJ*, March 1998.
- [7] R. Cameron and B. Woerner, "Synchronization of CDMA systems employing interference cancellation," in *Proceedings VTC, Atlanta, GA*, April 1996, pp. 178–182.
- [8] A. Mantravadi and V. V. Veeravalli, "Multi-access interference resistant acquisition for CDMA systems with long spreading sequences," in *Proceedings CISS, Princeton, NJ*, March 1998.
- [9] J. Thomas and E. Geraniotis, "Iterative MMSE multiuser interference cancellation for trellis coded CDMA systems in multipath fading environments," in *Proceedings CISS, Baltimore, MD*, March 1999.
- [10] A. J. Weiss and B. Friedlander, "Channel estimation for DS-CDMA downlink with aperiodic spreading codes," *IEEE Transactions on Communications*, vol. 47, no. 10, pp. 1561–1569, Oct 1999.
- [11] Z. Xu and M. K. Tsatsanis, "Blind channel estimation for long code multiuser CDMA systems," *IEEE Transactions on Signal Processing*, vol. 48, no. 4, pp. 988–1001, Apr 2000.
- [12] M. Torlak, B. L. Evans, and G. Xu, "Blind estimation of FIR channels in CDMA systems with aperiodic spreading sequences," in *Conference Record of the Thirty-First Asilomar Conference on Signals, Systems, and Computers*, Nov 1997, vol. 1, pp. 495–499.
- [13] C. Sengupta, S. Das, J. R. Cavallaro, and B. Aazhang, "Efficient Multiuser Receivers for CDMA Systems," in *IEEE Wireless Communication and Networking Conference*, Sep 1999, pp. 1459–1463.
- [14] A. Wald, "Tests of statistical hypotheses concerning several parameters when the number of observations is large," *Transactions of the American Mathematical Society*, vol. 54, no. 3, pp. 426–482, Nov 1943.
- [15] C. R. Rao, *Linear Statistical Inference and its Applications*, Wiley Publications, 1966.
- [16] M. K. Varanasi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," *IEEE Transactions on Communications*, vol. 38, no. 4, pp. 509–519, Apr 1990.
- [17] E. Ertin, U. Mitra, and S. Siwamogsatham, "Iterative techniques for DS/CDMA multipath channel estimation," in *Allerton Conference, Monticello, IL*, Sep 1998, pp. 772–781.
- [18] H. L. Van Trees, *Detection, Estimation, and Modulation Theory - Part 1: Detection, Estimation, and Linear Modulation Theory*, John Wiley, 1st edition, 1968, New York, NY.
- [19] S. Haykin, *Adaptive Filter Theory*, Information and System Sciences Series. Prentice Hall, 3rd edition, 1996.
- [20] M. Honig, U. Madhow, and S. Verdu, "Blind adaptive multiuser detection," *IEEE Transactions on Information Theory*, vol. 41, no. 4, pp. 944–960, Jul 1995.
- [21] G. H. Golub and C. F. Van Loan, *Matrix computations*, The Johns Hopkins University Press, 3rd edition, 1996.
- [22] S. Rajagopal, S. Bhashyam, J. R. Cavallaro, and B. Aazhang, "Efficient VLSI architectures for baseband signal processing in wireless base-station receivers," in *To appear in Proceedings ASAP, Boston, MA*, July 2000.
- [23] W. C. Jakes, Ed., *Microwave Mobile Communications*, Wiley, New York, 1974.
- [24] N. J. Higham, *Accuracy and stability of numerical algorithms*, Society of Industrial and Applied Mathematics, 1996.

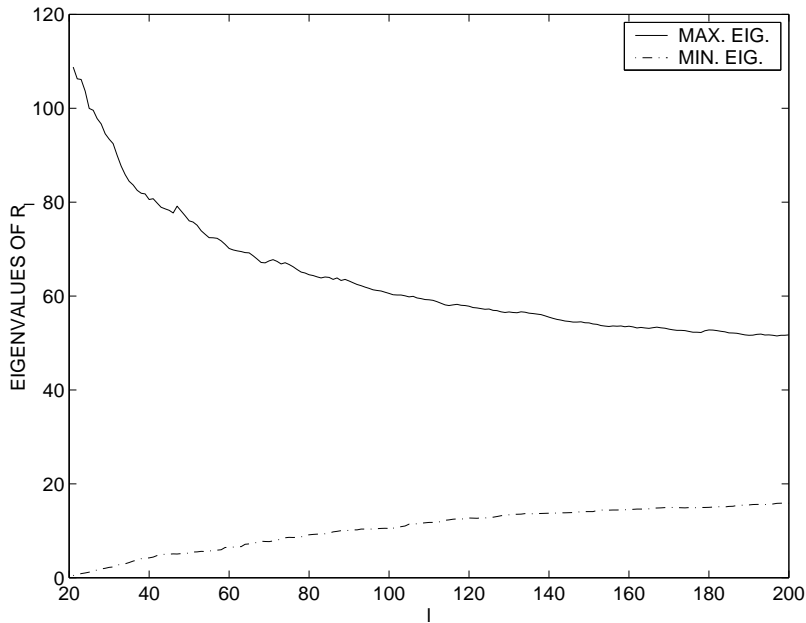


Fig. 1. Maximum and minimum eigen values of R_l – number of users = 16, spreading gain = 31, preamble length = l .

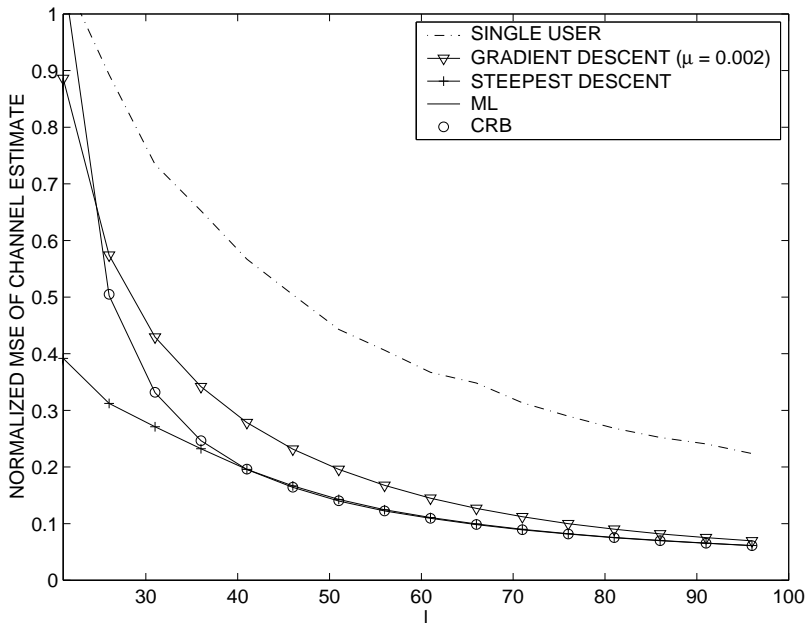


Fig. 2. Mean squared error of channel estimate ($E[||\hat{\mathbf{z}} - \mathbf{z}||^2]$) – number of users = 16, spreading gain = 31, number of paths per user = 2, no Doppler spread, SNR = 10 dB, preamble length = l , all users have equal power.

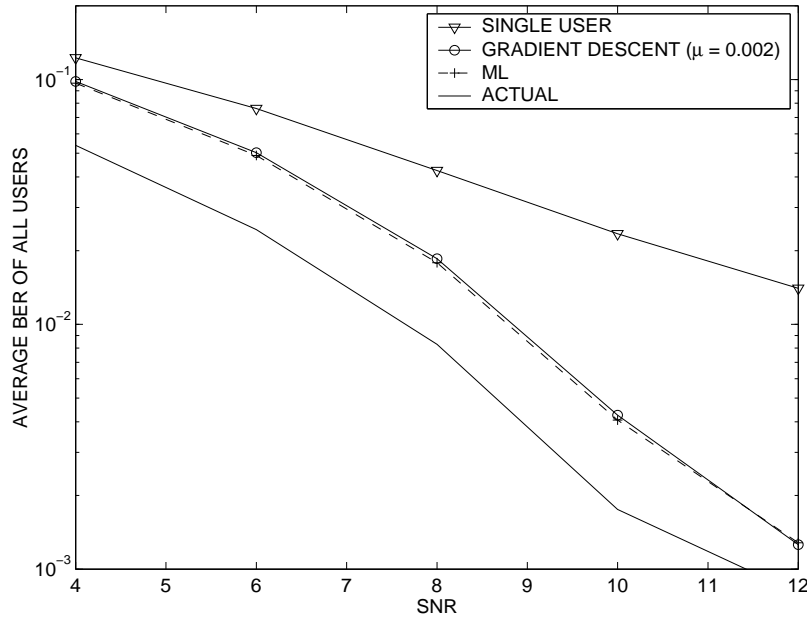


Fig. 3. Performance of multistage detector with different channel estimation methods – number of users = 16, spreading gain = 31, number of paths per user = 2, no Doppler spread, all users have equal power, preamble length = 100.

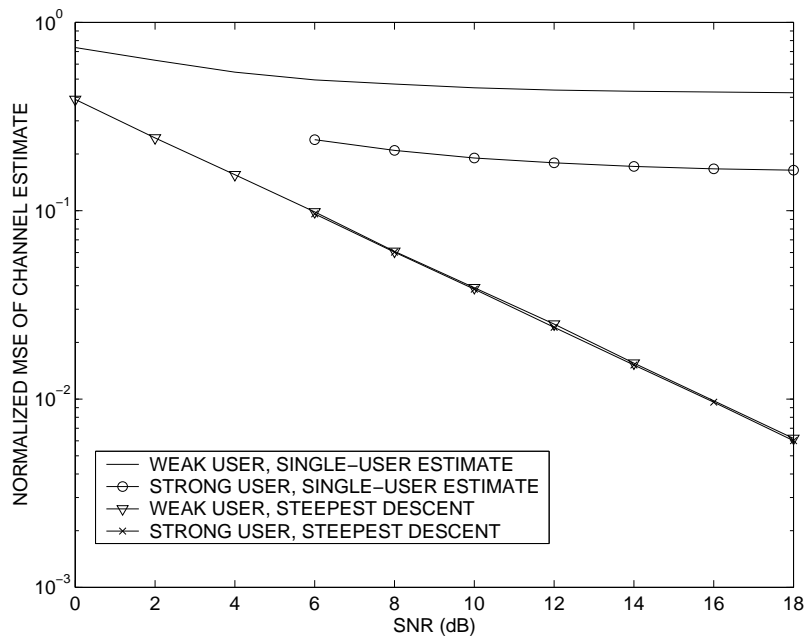


Fig. 4. Mean squared error of channel estimate – number of users = 16, spreading gain = 31, number of paths per user = 2, no Doppler spread, user 1 (weak user) has 6 dB lower power than other users, preamble length = 100.

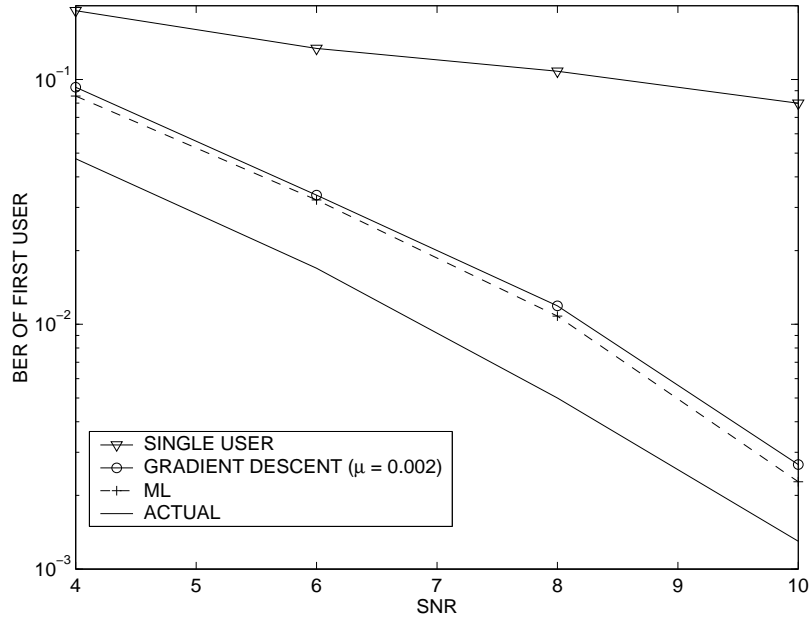


Fig. 5. Performance of multistage detector with different channel estimation methods – number of users = 16, spreading gain = 31, number of paths per user = 2, no Doppler spread, Interfering users have 6 dB higher power, preamble length = 100.

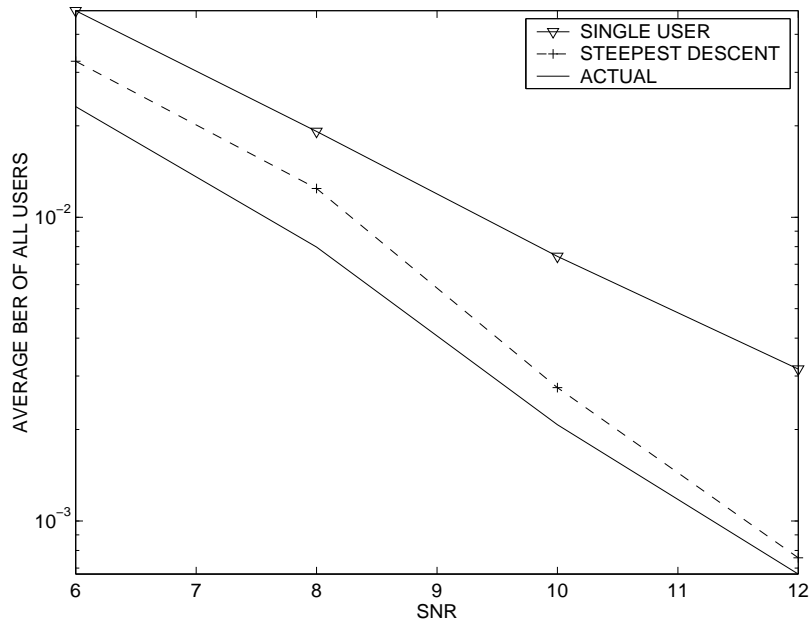


Fig. 6. Performance of multistage detector with different channel estimation methods with reduction in size – number of users = 16, spreading gain = 31, number of paths per user = 2, no Doppler spread, all users have equal power, preamble length = 50.

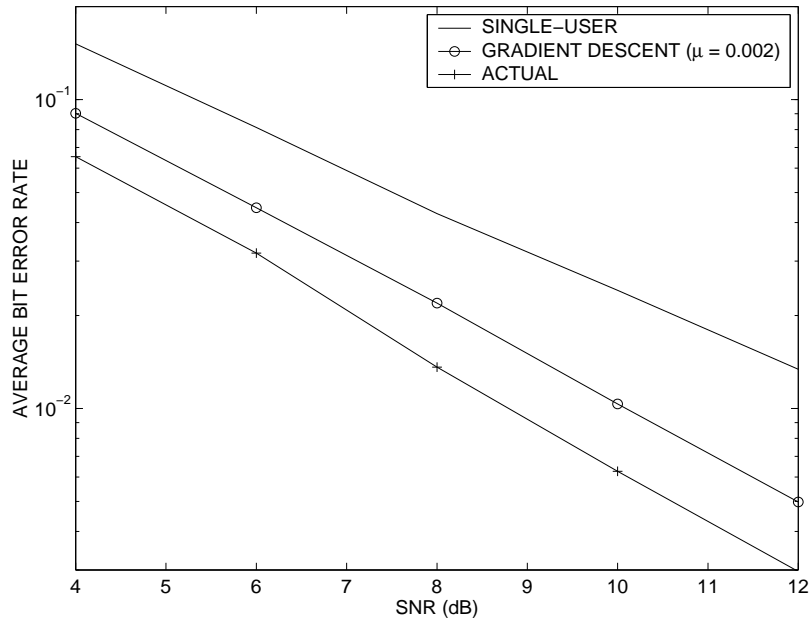


Fig. 7. Performance of multistage detector with different channel estimation and tracking methods – number of users = 8, spreading gain = 16, number of paths per user = 2, Doppler spread = 17 Hz, all users have equal average power, preamble length = 100.

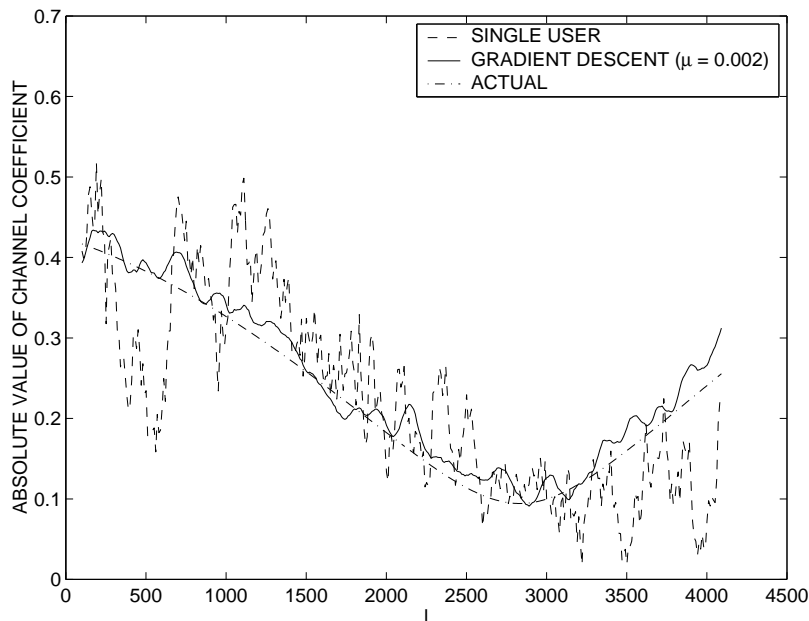


Fig. 8. Tracking performance – number of users = 8, spreading gain = 16, number of paths per user = 2, Doppler spread = 17 Hz, all users have equal average power, preamble length = 100.

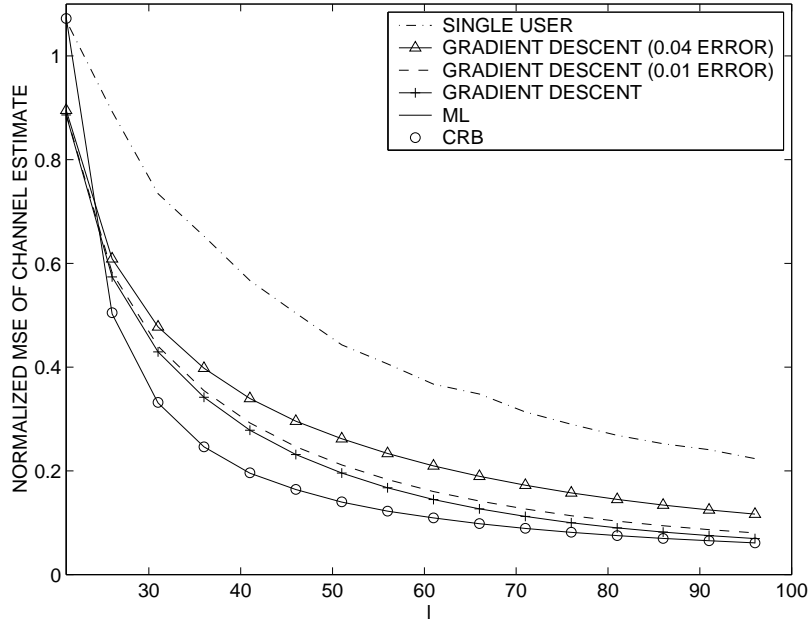


Fig. 9. Effect of decision feedback error: Mean squared error of channel estimate ($E[||\hat{\mathbf{z}} - \mathbf{z}||^2]$) – number of users = 16, spreading gain = 31, number of paths per user = 2, no Doppler spread, SNR = 10 dB, preamble length = l , all users have equal power.

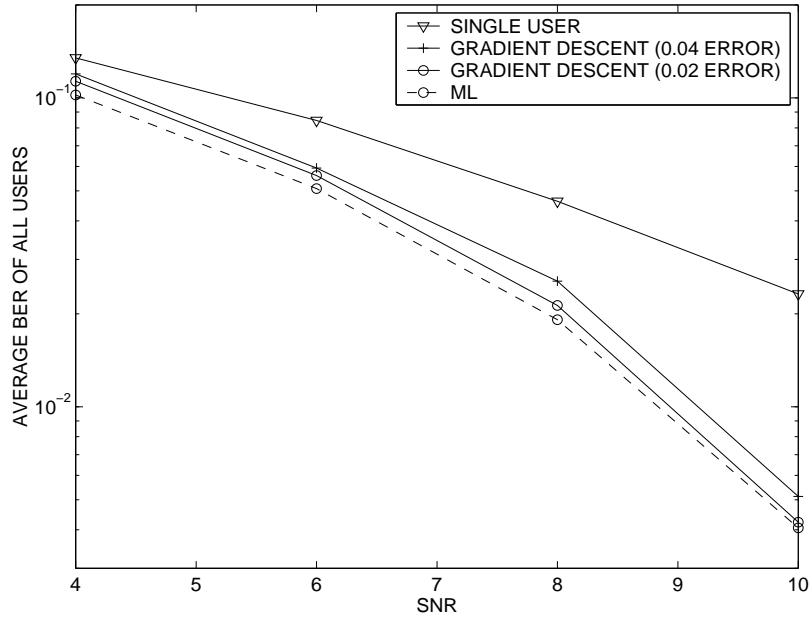


Fig. 10. Effect of decision feedback error: Performance of multistage detector with iterative channel estimation – number of users = 16, spreading gain = 31, number of paths per user = 2, no Doppler spread, all users have equal power, preamble length = 100.