

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/3159141>

Adaptive symbol and parameter estimation in asynchronous multiuser CDMA detectors

Article in IEEE Transactions on Communications · March 1997

DOI: 10.1109/26.554369 · Source: IEEE Xplore

CITATIONS

63

READS

48

2 authors, including:



[Teng Joon Lim](#)

The University of Sydney

268 PUBLICATIONS 14,078 CITATIONS

SEE PROFILE

Adaptive Symbol and Parameter Estimation in Asynchronous Multiuser CDMA Detectors

Teng J. Lim and Lars K. Rasmussen

Paper to appear in IEEE Trans. Communications, Feb 1997.

Abstract

Existing multiuser code division multiple access (CDMA) detectors either have to rely on strict power control or near-perfect parameter estimation for reliable operation. In this paper, a novel adaptive multiuser CDMA detector structure is introduced. Using either an extended Kalman filter (EKF) or a recursive least squares (RLS) formulation, adaptive algorithms which jointly estimate the transmitted bits of each user and individual amplitudes and time delays may be derived. The proposed detectors work in a tracking mode after initial delay acquisition is accomplished using other techniques not discussed here. Through computer simulations, we show that the algorithms perform better than a bank of single-user receivers in terms of near-far resistance. Practical issues such as the selection of adaptation parameters are also discussed in some detail.

I. INTRODUCTION

The area of multiuser CDMA detectors has attracted a lot of attention in the literature in the last few years, ever since the appearance of Verdu's landmark paper [1]. This is due to the severe reduction in system capacity brought about by the near-far problem when a conventional matched filter bank receiver is used [2]. In [1], Verdu showed that the near-far problem can, in theory, be overcome without power control by using a Viterbi decoder at the output of a bank of matched filters. But because the complexity of this optimum receiver increases as 2^K , where K is the number of users, it cannot be implemented in practice. This has driven researchers to look for suboptimal but simpler receivers which provide near-optimal performance without incurring the cost of exponential complexity.

Many suboptimal detectors have been proposed (see eg. [3], [4], [5], [6]) but these all assume knowledge at the receiver of all or some of the following parameters – received signal energies, carrier phase and propagation delays. Recently, attention has started to shift towards near-far resistant methods of parameter estimation and the effect of imperfect estimation on the performance of some multiuser detectors [7], [8], [9]. Of the three system parameters mentioned, time delay appears to be the hardest to estimate accurately because of the nonlinear dependence of the received signal on user delays. This is evident in early work on parameter estimation [10], [11] which tackled amplitude and phase estimation assuming time delays to be known.

Work on joint data detection and parameter estimation was presented in [12] and [13] but the algorithms that resulted are extremely complex. In fact, the one in [13] is exponential in the number of users and so fares no better than Verdu's optimal detector [1] in terms of practicality. Very recently, there has been a spate of activity in the area of subspace-based methods of delay acquisition and tracking using the MUSIC (Multiple Signal Classification) algorithm [14], [15], [16], which result in algorithms of complexity K^3 . These algorithms are significant advances because their delay estimates are near-far resistant, meaning that changes in the multiple access interference (MAI) level do not affect the variance of the estimates. However, the need to estimate the interference subspace with each shift of the observation window may place a limit on their ability to track rapidly time-varying delays.

The authors are with the Centre for Wireless Communications, Department of Electrical Engineering, National University of Singapore, Kent Ridge Crescent, Singapore 119260. Tel: +65 7715158, Fax: +65 7795441, E-mail addresses: cwclimtj@nus.sg, cwclkr@nus.sg.

The work described in this paper complements the MUSIC delay acquisition algorithms of Bensley, Ström and their colleagues by providing a less complex adaptive multiuser detector which performs joint data detection and parameter tracking after initial acquisition. The detector structure resembles that of the synchronous detector in [17] but is more versatile in that it handles the asynchronous, unknown delay case explicitly. We approach the problem from a system identification point of view, and use the prediction error to update the parameter and bit estimates recursively. Two algorithms are derived, one based on recursive least squares (RLS) and the other on the extended Kalman filter (EKF). No claims about the optimality of the algorithms can be made because of the nonlinear relationship between the estimator output and the delay estimates. However, assuming that the linearization introduces negligible errors, then the EKF reduces to the standard Kalman filter which produces maximum *a posteriori* state estimates at each time instant, provided the first-order Gauss-Markov model of parameter time variation is exact and the noise covariance matrices are known exactly. On the other hand, the RLS algorithm is not statistically optimal under any conditions, and thus performs relatively poorly.

The paper is organized as follows. Section II describes the development of the RLS and the EKF algorithms in detail, Section III discusses the important point of selection of filter tuning parameters, Section IV presents pertinent simulation results and Section V concludes the paper with a discussion of future work.

II. ALGORITHM DEVELOPMENT

A. System Model

The following is a list of symbols used in this paper and their meanings:

K	=	Number of users,
$P_k(t)$	=	power of the k th user's received signal at time t ,
$b_k(i)$	=	i th bit transmitted by the k th user,
$s_k(t)$	=	signature waveform of the k th user,
T	=	bit duration,
$\tau_k(t)$	=	propagation delay for the k th user
$\phi_k(t)$	=	random carrier phase, and
$n(t)$	=	additive channel noise.

$\tau_k(t)$ is measured relative to an arbitrary (but fixed) time origin, and lies in the interval $[0, T)$.

Assuming ideal lowpass filtering to remove high-frequency noise components, the baseband received signal is given by

$$r(t) = \sum_{k=1}^K \sqrt{P_k(t)} \cos \phi_k(t) b_k(i_k) s_k[t - \tau_k(t)] + n(t), \quad (1)$$

where $i_k = \lfloor (t - \tau_k)/T \rfloor$, and $b_k(i_k)$ is the bit from the k th user contributing to the received signal at time t and $\lfloor x \rfloor$ denotes the largest integer smaller than x . Suppose that the chip duration is $T_c = T/L$, L being the processing gain. Then sampling at M times the chip rate gives the following discrete-time sequence:

$$r(m) = \sum_{k=1}^K \sqrt{P_k(m)} \cos \phi_k(m) b_k(i_k) s_k[mT_s - \tau_k(m)] + n(m), \quad (2)$$

where $T_s \triangleq T_c/M$. Note that the assumption of short codes has not been made, and is in fact unnecessary in the proposed scheme, in contrast to adaptive minimum mean squared error (MMSE) algorithms [18], [19], which work only when short codes are used.

The generation of this received signal sequence is also depicted graphically in Figure 1, in which $c_k(m) = \sqrt{P_k(m)} \cos \phi_k(m) b_k(i_k)$. From it, we see that it is quite natural to predict both time delays and c_k 's adaptively using the structure shown in the bottom half of that figure. Details of how this can be done adaptively follow in the next two sections.

B. Method I – The Recursive Least Squares Algorithm

Given a parameter estimate vector θ and observations up to time $m - 1$, the minimum mean squared error (MMSE) predictor for $r(m)$ is [20]

$$\begin{aligned} \hat{r}[m|m-1; \theta] &= E[r(m|m-1; \theta)] \\ &= \sum_{k=1}^K c_k(m) s_k[mT_s - \tau_k] \end{aligned} \quad (3)$$

where the amplitude, phase and bit terms have been lumped together into one variable, c_k , and the parameter vector θ is defined as

$$\theta = [c_1, \dots, c_K, \tau_1, \dots, \tau_K]^T, \quad (4)$$

and the conditional expectation is taken with respect to the probability density function of the additive noise.

Assuming that θ is time-invariant and that the additive measurement noise $n(m)$ is a zero-mean Gaussian random process, the maximum likelihood (ML) estimate for θ given $m + 1$ samples of the channel output is given by

$$\hat{\theta}_{ML}(m) = \min_{\theta} \sum_{p=0}^m |r(p) - \hat{r}(p)|^2 \quad (5)$$

where $\hat{r}(p)$ is simply the shorthand notation for $\hat{r}(p|p-1; \theta)$ given in (3). This leads to the famous result that minimum least squares (LS) estimation is equivalent to ML estimation for Gaussian random processes. The quantity $r(p) - \hat{r}(p)$ is known as the prediction error at time p and will be denoted by $e(p)$.

However, when θ is expected to be time-varying, (5) cannot be used because it gives equal weighting to prediction errors made over the entire observation window, when more importance should be given to recent errors so as to reflect recent changes in parameter values. The popular but *ad hoc* method of weighted least squares (WLS) minimizes the exponentially weighted cost or loss function

$$V_m(\theta) = \sum_{p=0}^m \lambda^{m-p} e^2(p) \quad (6)$$

at each time instant m and thus is able to forget data collected in the distant past, thereby tracking changes in the parameters. λ is known as the forgetting factor for obvious reasons. Minimization of $V_m(\theta)$ can be carried out by processing a batch of collected data or in an on-line or recursive algorithm, known as an exponentially-weighted recursive least squares (RLS) algorithm. It should be noted that the parameter estimates obtained using the RLS algorithm cannot be said to be optimal in any sense, except that they minimize the loss function $V_m(\theta)$, which provides a reasonable measure of the distance of the true parameters from their estimated values at each sample.

Because $e(p)$ is a nonlinear function of θ , it is impossible to obtain an analytical expression for the weighted least squares estimate $\hat{\theta}_{WLS}(m) = \min_{\theta} V_m(\theta)$. But recursive optimization techniques well-known in the system identification field may be used to minimize this loss function (see [21] or [20]). In particular, the recursive least squares (RLS) algorithm when applied to our problem performs the following steps in each recursion:

$$\hat{r}(m) = \sum_{k=1}^K \hat{c}_k(m-1) s_k[mT_s - \hat{\tau}_k(m-1)] \quad (7)$$

$$e(m) = r(m) - \hat{r}(m) \quad (8)$$

$$K(m) = \mathbf{P}(m-1) \zeta(m) \left[\lambda + \zeta^T(m) \mathbf{P}(m-1) \zeta(m) \right]^{-1} \quad (9)$$

$$\hat{\theta}(m) = \hat{\theta}(m-1) + K(m) e(m) \quad (10)$$

$$\mathbf{P}(m) = \frac{1}{\lambda} \left(\mathbf{I} - K(m) \zeta^T(m) \right) \mathbf{P}(m-1). \quad (11)$$

In these equations, $\zeta(m)$ is defined as

$$\zeta(m) = - \frac{\partial e(m)}{\partial \hat{\theta}(m-1)} \quad (12)$$

and is often known as the gradient vector. The calculation of $\zeta(m)$ will be described in detail in a later section. Also, $\mathbf{P}(m)$ happens to be the inverse of the Hessian matrix $\ddot{V}_m[\hat{\theta}(m-1)]$ where the double dot notation stands for the second derivative with respect to θ .

The RLS algorithm therefore provides estimates of the k th user's transmitted bits as long as $\cos \phi_k(m) > 0$ through $\hat{b}_k(i) = \text{sgn}(\hat{c}_k(m))$ when $mT_s = iT + \hat{\tau}_k(m)$, $i = 0, 1, \dots$. Time delay estimates are explicitly obtained through $\hat{\tau}_k(m)$. The condition placed on the phase of the received signal is unfortunate because it implies that phase recovery must be done accurately. However, differential encoding can be used to minimize the effects of phase offset and so lack of phase synchronism is usually not a very serious problem.

C. Method II – The Extended Kalman Filter

Approaching the problem from a completely different angle, the extended Kalman filter (EKF) equations turn out to be virtually identical to those of the RLS algorithm, except that the EKF is more flexible and is capable of incorporating prior knowledge about the system into the adaptive parameter updates.

The EKF is described in many books on statistical estimation theory or optimal filtering, such as [22], [23]. Briefly, it is an extension of the regular Kalman filter, which gives maximum *a posteriori* (MAP) estimates of the state vector $\mathbf{x}(k)$ in a linear system, to nonlinear systems in which the current state can be a nonlinear function of the previous state, and the measured outputs can be a nonlinear function of the state vector.

The EKF has been used for parameter estimation in spread spectrum and CDMA systems in the past, most notably by Iltis *et. al.* [24], [13]. In the context of multiuser systems, Iltis and Mailaender in [13] proposed a maximum likelihood algorithm that involved detecting all K bits transmitted in a given bit epoch by the K users at the same time. A number of EKF's were used to calculate amplitude and delay estimates for all possible bit combinations in a bit epoch, and these were then used to update the ML metrics. This very complicated algorithm has a complexity that increases exponentially with the number of users, and so cannot be used in practice. In contrast, the algorithm to be presented here has a complexity of $O(K^2)$.

We start by assuming that the true parameter vector varies with time as a first-order Gauss-Markov process:

$$\theta_0(m+1) = \theta_0(m) + \mathbf{w}(m) + \mathbf{u}(m), \quad (13)$$

where $\mathbf{w}(m)$ is a zero-mean noise process with known covariance matrix $\mathbf{Q}(m) = E[\mathbf{w}(m)\mathbf{w}^T(m)]$, and $\mathbf{u}(m)$ is a deterministic vector which models any global trends. $\mathbf{u}(m)$ is ultimately irrelevant to the Kalman filter equations so we will not mention it any further.

The measurement equation relates the system output (in our case the received signal) to the state vector $\theta_0(m)$ and is given by (2), which is re-written here as

$$r(m) = f(s_1(t), \dots, s_K(t), \theta_0(m)) + n(m), \quad (14)$$

where $f(\cdot)$ is a nonlinear function of the τ_k 's and $n(m)$ is zero-mean Gaussian white noise with power $E[n^2(m)] = \sigma^2$. It is the nonlinearity of $f(\cdot)$ which forces us to use the extended Kalman filter instead of the standard Kalman filter.

From this state-space model of the system, we easily derive the following EKF equations [22]:

$$\mathbf{P}(m|m-1) = \mathbf{P}(m-1|m-1) + \mathbf{Q}(m) \quad (15)$$

$$\hat{r}(m) = \sum_{k=1}^K \hat{c}_k(m-1)s_k[mT_s - \hat{\tau}_k(m-1)] \quad (16)$$

$$e(m) = r(m) - \hat{r}(m) \quad (17)$$

$$K(m) = \mathbf{P}(m|m-1)\zeta(m) \left[\sigma^2 + \zeta^T(m)\mathbf{P}(m|m-1)\zeta(m) \right]^{-1} \quad (18)$$

$$\hat{\theta}(m) = \hat{\theta}(m-1) + K(m)e(m) \quad (19)$$

$$\mathbf{P}(m|m) = \left(\mathbf{I} - K(m)\zeta^T(m) \right) \mathbf{P}(m|m-1), \quad (20)$$

where $\zeta(m)$ is defined in (12).

At first glance, these equations do not seem to differ very much from the RLS equations. However, there is now a matrix $\mathbf{Q}(m)$ which models the nonstationarity of the true parameters, and a user-selectable variable σ^2 which models the amount of noise in the output measurement. In addition, $\mathbf{Q}(m)$ may be time-varying and is unconstrained with respect to the values of its elements except that it has to be symmetric and non-negative definite. These two conditions arise naturally from the definition of $\mathbf{Q}(m)$, given below equation (13).

The availability of $\mathbf{Q}(m)$ and σ^2 translates into improved flexibility compared to the RLS algorithm, which only has one adjustable adaptation constant λ . To illustrate, consider the state transition equation (13), where

$$\theta_0(m) = [c_1(m), \dots, c_K(m), \tau_1(m), \dots, \tau_K(m)]^T \quad (21)$$

and $c_k(m) = \sqrt{P_k(m)}b_k(i_k)^1$. We expect the propagation delays to change smoothly with time, but at the bit boundaries of the k th user, which occur when $mT_s = iT + \tau_k(m)$ for integer values of i ,

$$c_k(m+1) = \begin{cases} 0 & \text{with probability } \frac{1}{2} \\ -2c_k(m) & \text{with probability } \frac{1}{2} \end{cases} \quad (22)$$

The first scenario occurs when $b_k(i_k + 1) = b_k(i_k)$ while the second corresponds to $b_k(i_k + 1) = -b_k(i_k)$. The perturbation sequence $\mathbf{w}(m)$ will thus be nonstationary and have a covariance matrix

$$\mathbf{Q}(m+1) = \begin{cases} \mathbf{Q}_0 & \text{if } mT_s \neq iT + \tau_k(m) \\ \mathbf{Q}_0 + \text{diag}(\overbrace{0, \dots, 0}^{k-1}, 2c_k^2(m), \overbrace{0, \dots, 0}^{2K-k}) & \text{if } mT_s = iT + \tau_k(m) \end{cases} \quad (23)$$

where $\mathbf{Q}_0$ is a diagonal matrix with elements representing our belief in the degree of time variation in the parameters. The $2c_k^2(m)$ term on the right is the expected value of $c_k^2(m+1)$ given the probability density function (22).

¹Assuming that perfect phase lock or $\phi_k(m) = 0$ is achieved.

In addition, σ^2 is the power of the measurement noise which the user assumes is in the observed signal. $\mathbf{Q}(m)$ and σ^2 may be considered “tuning” parameters which affect the performance of the EKF in the following way.

By lowering the value of σ^2 , the algorithm puts more faith in the measurement and causes the parameter estimates to change more wildly from one sample to the next; increasing σ^2 results in a more skeptical algorithm which tends to ignore rapid fluctuations and produces smoother parameter trajectories. Conversely, making $\mathbf{Q}(m)$ smaller (in a matrix sense, for instance in the trace) causes parameter estimates to change more smoothly. Thus, the variance of the parameter estimates obtained using the EKF varies inversely with σ^2 but directly with $\mathbf{Q}(m)$. In a later section, it is shown that the performance of the EKF is in fact largely determined by the ratio between $\mathbf{Q}(m)$ and σ^2 .

D. The Gradient Vector

Using (3), the definition of $\zeta(m)$ given in (12) and the fact that $r(m)$ is independent of $\hat{\theta}(m)$, we have

$$\zeta_c^k(m) \triangleq -\frac{\partial e(m)}{\partial \hat{c}_k} = \frac{\partial \hat{r}(m)}{\partial \hat{c}_k} \quad (24)$$

$$= s_k(mT_s - \hat{\tau}_k) \quad (25)$$

and

$$\zeta_\tau^k(m) \triangleq -\frac{\partial e(m)}{\partial \hat{\tau}_k} = \frac{\partial \hat{r}(m)}{\partial \hat{\tau}_k} \quad (26)$$

$$= \hat{c}_k \frac{\partial s_k(mT_s - \hat{\tau}_k)}{\partial \hat{\tau}_k}, \quad (27)$$

for $k = 1, \dots, K$. It is to be understood that the $\hat{c}_k$'s and $\hat{\tau}_k$'s take on their most recently updated values in these equations. In addition, it is necessary to assume that $\hat{\tau}_j(m)$ is not a function of $\hat{c}_k(m)$ for $j, k = 1, \dots, K$ in order to arrive at these expressions².

The evaluation of the gradient vector when $\hat{\tau}_k$ is not a multiple of the sampling interval T_s may proceed in two ways. The simpler method is to treat $s_k(t)$ as a square waveform, the more complicated one considers it a smooth waveform.

D.1 Square Signature Waveforms

In this section, $\hat{\tau}_k(m)$ is normalized to the sampling interval T_s . This results in no loss of generality and makes notation less cluttered. Assuming a square and binary-valued signature waveform, and that chip boundaries are located in the middle of sampling intervals, the value of $s_k(t)$ in between samples is equal to that of the nearest sample, or

$$s_k(m - \hat{\tau}_k) = s_k(m - [\hat{\tau}_k]), \quad (28)$$

where $[x]$ denotes x rounded off to the nearest integer.

The evaluation of the derivative of $s_k(\cdot)$ w.r.t. $\hat{\tau}_k(m)$ which appears in (27) is also quite trivial because of our assumption on the shape of $s_k(t)$. A little thought reveals that this derivative can only take on one of three values: 0, $+\infty$ and $-\infty$. Since it is undesirable for any adaptive

²This assumption holds exactly if the $\mathbf{P}$ matrix (in both RLS and EKF algorithms) is diagonal. In practice, because $\hat{c}_k$ is an estimate of the received signal power for user k , which partly depends on the distance between the k th transmitter and the base station, which in turn affects the k th propagation delay, it probably does not hold. However, to keep things simple, we have chosen to go with this assumption for now and leave the coupled power/delay case for future work.

algorithm to have extreme parameter corrections from one sample to the next, we choose to use the more moderate estimate

$$\frac{\partial s_k(m - \hat{\tau}_k)}{\partial \hat{\tau}_k} \approx \text{sgn}[s_k(m - \lceil \hat{\tau}_k \rceil) - s_k(m - \lfloor \hat{\tau}_k \rfloor)] \quad (29)$$

where the signum function is defined as

$$\text{sgn}(x) = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \\ -1 & \text{if } x < 0 \end{cases} . \quad (30)$$

Finally, putting (28) into (25) and (29) into (27) allows us to evaluate the gradient vector under the square signature waveform assumption.

D.2 Smooth Signature Waveforms

Practical waveform generators cannot generate perfect square wave functions, and in any case, such functions have an infinite bandwidth even though a spread spectrum signal need only occupy $1/T_c$ Hz to realize its full processing gain, and so are not desirable for use as signature waveforms. As such, the assumption of square signature waveforms is not very realistic. Instead, $s_k(t)$ should be considered a smooth function obtained by analog lowpass filtering of the k th pseudo-noise (PN) code sequence.

The filtering operation ought to be repeated at the receiver to reproduce the signature waveforms. But it is difficult (if not impossible) to create continuous-time versions of digital signal processing algorithms, and so working in the continuous-time domain is highly undesirable. We can approximate the smooth signature waveforms by oversampling the known pulse-shaping transmit filter and passing the signature sequence through it; or, if better delay resolution is required and computational power is available, interpolation of the chip-rate signature sequence using various types of on-line interpolators, such as the cubic spline interpolators described in [25], can be carried out.

Looking at equations (7), (16), (25) and (27), $s_k(mT_s - \hat{\tau}_k)$ and its derivative with respect to $\hat{\tau}_k$ are the only two quantities that need to be computed for the k th user per sample. It should not be difficult to see how interpolation or oversampling can be used to extract these values from the signature sequence of the k th user.

III. IMPLEMENTATION

In this section, some implementation aspects of the two algorithms described in the previous section are discussed.

A. The RLS Algorithm

The exponential form of the cost function (6) used in the RLS algorithm implies that the system parameters of interest are expected to change slowly with time. If one or more of the parameters change abruptly at time t , the RLS algorithm based on the minimization of $V_m(\theta)$ may fail completely after that time because information unrelated to the present values of the affected parameters remain in the system. In our multiuser system model, $c_k(m)$ may change at the bit boundaries, depending on whether adjacent transmitted bits are equal or not, and these potentially fatal abrupt changes need to be taken into account.

One obvious step in the right direction is to reset $\hat{c}_k(m)$ to zero at the (estimated) bit boundaries, ie. when $m = iT + \hat{\tau}_k(m)$. This gets rid of the first-level dependency of $\hat{c}_k(m+1)$ on

$\hat{c}_k(m)$. But the link with the past is still not totally broken because $K(m)$ in (10) is a function of $\mathbf{P}(m-1)$, which is the estimate of the Hessian matrix of the cost function formed from past information using an exponential window. It is thus also necessary to re-initialize $\mathbf{P}(m)$ at the bit boundaries.

If it is assumed that changes in c_k do not affect the other system parameters in θ , then it suffices to set the k th column and k th row of $\mathbf{P}(m)$ equal to the corresponding column and row in the initial $\mathbf{P}(0)$ matrix. In this way, the adaptation of $\hat{c}$ and $\hat{\tau}$ for other users is not disrupted every time we reach one user's bit boundary. Since the time delays are not expected to change suddenly, their continuous adaptation is desirable.

As for the initial value of the $\mathbf{P}$ matrix, we take the conventional approach [20] of making it equal to $\mathbf{I}/\delta$, where δ is some small number reflecting our lack of knowledge about the system initially. The exponential forgetting factor λ should be chosen so that convergence within one bit interval is probable. This generally means that, if there are L chips per bit and the oversampling factor is M , $\lambda = 1 - (1/ML)$ should be a reasonable choice – larger values will lead to slower convergence, while smaller values leads to excessive noise sensitivity.

B. The EKF Algorithm

To use the EKF, we need to assume knowledge of the modelling noise covariance matrix $\mathbf{Q}(m)$ and the measurement noise power σ^2 . While σ^2 can perhaps be measured during quiet periods (enforced or natural) when no data is transmitted, it is hard to see how $\mathbf{Q}(m)$, the covariance matrix of the noise driving the state transition equation, can be found. It is thus of interest to find out how sensitive the estimation algorithm is to changes in the value of $\mathbf{Q}(m)$ about its correct setting.

Sensitivity studies are not new to general Kalman filtering. Mendel [23] presents a table of sensitivity coefficients obtained through computer simulations. They measure the percentage changes in the Kalman gain $K(k)$ to one percent changes in various model parameter values. However, these results are not generally applicable, and thus we need to examine the sensitivity of our particular system to variations in the assumed $\mathbf{Q}(m)$ matrix.

To this end, the following computer simulation is set up. Two users are in the system, each transmitting with length-25 randomly-generated binary short codes generated using $s_k(m) = \text{sgn}(x(m))$ for $m = 1, \dots, L$, where $x(m)$ is an independent zero-mean Gaussian random process. The matrix $\mathbf{Q}_0$ in equation (23) is given by

$$\mathbf{Q}_0 = \text{diag}(\mathbf{q}_c^T, \mathbf{q}_\tau^T) \quad (31)$$

where $\mathbf{q}_c^T = q_c[1, \dots, 1]$ and $\mathbf{q}_\tau^T = q_\tau[1, \dots, 1]$ represent the time variation of the c 's and τ 's respectively.

The system parameters evolved according to the first-order Gauss-Markov equation (13) with

$$E[\mathbf{w}(m)\mathbf{w}^T(m)] = \text{diag}(\overbrace{0, \dots, 0}^K, \overbrace{1 \times 10^{-4}, \dots, 1 \times 10^{-4}}^K), \quad (32)$$

which means that the true values of c are unchanging. It is also convenient to define $q_{\tau,0} = 1 \times 10^{-4}$ as the true value of q_τ . Finally, the multiple access interference is set to 10 dB, with the weaker user received with unit power (ie. $P_1 = 1$).

In Figure 2, the root mean square error between the delay estimate and the true delay measured using one thousand samples is plotted against $q_\tau/q_{\tau,0}$ for two cases – when the signal to noise ratio for the weaker user is 20 dB, and when it is 40 dB. The true value of σ^2 is used in both experiments. It appears that a dynamic range of about 20 dB is allowed for q_τ for the algorithm to produce approximately the same performance in terms of RMSE in both cases.

Unfortunately, the point of peak performance, and more importantly, the permissible range of q_τ , depends on the SNR. This makes the maximum blind allowable range of q_τ (ie. assuming no knowledge of SNR), closer to about 5 to 10 dB around $q_{\tau,0}$.

However, extensive simulation studies show that the RMSE of the delay estimates does not depend on absolute values of $\mathbf{Q}(m)$ and σ^2 so much as the ratio between the two. Furthermore, the variation of the RMSE with $\mathbf{Q}(m)/\sigma^2$ is approximately independent of the SNR. This is demonstrated in Figure 3, which is a superposition of ten curves of RMSE against q_τ/σ^2 obtained using MAI's ranging from 0 to 20 dB and SNR's of 10, 20 and 40 dB. Although they are not identical, the lines converge in the region of $q_\tau/\sigma^2 = 0.007$ to $q_\tau/\sigma^2 = 0.07$ and hence this may be considered the permissible range for q_τ/σ^2 under all conditions. In simulation results to follow, q_τ/σ^2 is set to 0.01 while q_c is kept constant at 0.001.

IV. SIMULATION RESULTS

A. Comparison of the Two Proposed Algorithms

It was mentioned briefly in Section I that the RLS algorithm is not optimal under any conditions whereas the EKF is near-optimal under certain circumstances. The EKF was also shown to be more flexible and allows prior information about the system to be incorporated. Finally, the resetting of $\mathbf{P}(m)$ and $c_k(m)$ at every bit boundary in order to remove noninformative data can be expected to affect the performance of the detector adversely because bit decisions may have to be made before full convergence of other users' c parameters is achieved. For these reasons, the RLS detector is not expected to perform as well as the EKF algorithm.

To illustrate, Figures 4 and 5 show the trajectories of $\hat{c}$ and $\hat{\tau}$ respectively for the weaker user in a time-invariant two-user system with an MAI of 20 dB for the EKF and RLS algorithms. There are 25 chips per bit, the sampling interval is $T_c/2$, the power of the weaker signal is $P_1 = 1$ and the SNR for the weaker user is 20 dB. In the RLS detector, λ was set to 0.98 so as to achieve convergence in about 50 iterations while in the EKF detector, $q_c = 0.001$, $q_\tau = 0.01\sigma^2$ and $\sigma^2 = 0.01$, as explained previously.

It is seen that the RLS algorithm, while converging to the true delay estimate in about 3 bit intervals, exhibits large spurious jumps in parameter estimate values whereas the convergence of the EKF parameters is a lot smoother. Also, note that bits 4 to 7 are all '1's, and $\hat{c}$ in the EKF detector stays close to unity at all times while in the RLS, it has to be reset to zero at each bit boundary.

B. The Single-User EKF Delay Tracker

In [24], an extended Kalman filter was used to estimate both PN code delay and multipath coefficients for a single-user system. Analogous to the extension of the single-user matched filter or correlating detector to multiuser systems using a bank of filters, a simple extension of the delay-tracking single-user EKF would be a bank of EKF's, each using its own prediction error in the EKF equations. This is illustrated in Figure 6. The parameter update equations are identical to (15) to (20), except that they apply to only one user at a time, meaning that K cycles through these equations must be carried out per sample. This also implies that different $\mathbf{P}$'s, $\mathbf{Q}$'s, $\hat{r}$'s, e 's and K 's exist for different users, and that $\hat{\theta}_k$ has the two elements $\hat{\tau}_k$ and $\hat{c}_k$ only.

In the following section, the EKF algorithm is compared to this bank of single-user EKF delay estimators.

C. Near-Far Resistance

Near-far resistance in the context of multiuser delay estimation refers to the sensitivity of the mean squared estimation error to changes in the level of multiple access interference – the more the MSE changes for a given change in MAI, the poorer the detector's near-far resistance. It would be nice to be able to quantify the near-far resistance of delay estimates but there does not appear to be a reliable way to do this at this time and therefore, we need to rely on the qualitative definition given here.

Table I compares the RMSE or root mean square estimation error of the delay estimates obtained over 5,000 samples using both the multiuser (MU) and the single-user (SU) EKF estimators. The results are for a two-user $2 \times$ oversampled system using length-25 random binary codes, with a 20 dB SNR for the weaker user and the standard tuning parameter values. The system delays were made to vary with time in the same way as in Section III. The last two columns represent the number of errors in the bit estimates detected over the 100 bits used in the simulation.

It is quite evident from both the RMSE and the bit error results that the proposed multiuser EKF detector is approximately near-far resistant, whereas the single-user detector is severely affected by MAI.

To provide a visual picture of the tracking of the delay estimates using the multiuser and single-user EKF's, Figure 7 shows the actual and estimated delay trajectories for the weaker user in the system for an MAI of 15 dB, from which we see that the single-user detector fails to keep track of the actual delay, whereas the multiuser detector performs this task adequately.

It is reasonable at this point to ask if a rate of change of 0.005 chips per chip interval, which was the rate used in these simulations, is realistic. That question can be answered using some simple physics. Suppose a mobile and a base station are x metres apart, and that the velocity of the mobile in the line-of-sight direction is v metres per second. Then $x = c\tau$, where c is the speed of electromagnetic propagation, and hence

$$v = \frac{dx}{dt} = c \frac{d\tau}{dt} \quad (33)$$

where t represents time. In our simulations, we have $d\tau/dt \approx 0.005$ and the EKF detector has no problems tracking changes in the delay. If we now took 0.005 as the maximum rate of change of τ allowable for reliable detector performance, then the maximum mobile speed is $v_{max} = 0.005c = 1.5 \times 10^6$ m/s! In practice, we may expect a far smaller mobile velocity and thus the EKF detector should be well capable of handling the situation.

V. CONCLUSIONS AND DISCUSSION

Two adaptive multiuser asynchronous CDMA detectors implemented using the extended Kalman filter (EKF) and weighted recursive least squares (RLS) have been introduced. It was reasoned that the EKF detector should perform better than the RLS one because it is capable of incorporating prior knowledge about the system, and is therefore more flexible. Computer simulations demonstrate that this is true.

The choice of tuning parameters for both adaptive detectors was discussed, and it was found that the performance of the EKF detector depended more on the ratio between the two parameters than on their absolute values, which eased the selection of parameter values considerably, while some *ad hoc* resetting procedure had to be used in the RLS detector at the bit boundaries, which resulted in poorer performance. The multiuser EKF detector was shown to easily outperform a bank of single-user EKF's in terms of near-far resistance.

The proposed detector operates in a tracking mode, and relies on the availability of accurate initial delay estimates, which may perhaps be obtained using one of the recently introduced

subspace-based approaches [14], [15], [16]. It is significantly less complex to implement than these methods, which require the estimation of noise subspaces recursively. However, analytical error probability results have not been derived, and it is envisaged that this will be a difficult task since the probability of error may be expected to depend on the degree and type of time variation in the parameters, as well as the more traditional E_b/N_0 . This is an area of current research.

The complexity of the algorithm is still of concern however. The updating of the parameter estimates need to be carried out at the sampling rate, which must be a multiple of the chip rate to satisfy the Nyquist sampling theorem. One way of reducing complexity is to train the receiver in an initial startup period using unmodulated spreading waveforms, and then keep the delay estimates unchanged thereafter. An unmodulated spreading waveform does not cause abrupt jumps in c_k at the bit boundaries, and thus convergence within one bit interval becomes unnecessary. We can then implement an adaptive algorithm at a slower updating rate, thereby reducing complexity. Keeping the time delays fixed after initial acquisition also makes sense since the mobile station is unlikely to move fast enough to cause significant changes in the propagation delay. Periodic re-training is an option if this assertion proves to be false.

In conclusion, we see the adaptive EKF detector as a viable multiuser CDMA detector for the base station when paired with near-far resistant delay acquisition and phase-lock devices. Design of the latter will allow for coherent modulation and is being investigated at present. In practice, multipath fading is also a major source of performance degradation, and needs to be taken care of. One obvious extension of the method described here is to have more weights in each branch to span the channel response for each user, but this adds to complexity and may not be feasible. Work is ongoing in this area.

REFERENCES

- [1] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 32, no. 1, pp. 85–96, Jan 1986.
- [2] R. Lupas and S. Verdú, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 35, no. 1, pp. 123–136, Jan 1989.
- [3] R. Lupas and S. Verdú, "Near-far resistance of multiuser detectors in asynchronous channels," *IEEE Trans. Communications*, vol. 38, no. 4, pp. 496–508, Apr 1990.
- [4] A. Duel-Hallen, "A family of multiuser decision-feedback detectors for asynchronous code-division multiple-access channels," *IEEE Trans. Communications*, vol. 43, no. 2/3/4, pp. 421–434, Feb/Mar/Apr 1995.
- [5] U. Fawer and B. Aazhang, "A multiuser receiver for code division multiple access communications over multipath channels," *IEEE Trans. Communications*, vol. 43, no. 2/3/4, pp. 1556–1565, Feb/Mar/Apr 1995.
- [6] M. K. Varanasi and S. Vasudevan, "Multiuser detectors for synchronous CDMA communication over non-selective Rician fading channels," *IEEE Trans. Communications*, vol. 42, no. 2/3/4, pp. 711–722, Feb/Mar/Apr 1994.
- [7] S. D. Gray, M. Kocic, and D. Brady, "Multiuser detection in mismatched multiple-access channels," *IEEE Trans. Communications*, vol. 43, no. 12, pp. 3080–3089, Dec 1995.
- [8] F. C. Zheng and S. K. Barton, "On the performance of near-far resistant CDMA detectors in the presence of synchronization errors," *IEEE Trans. Communications*, vol. 43, no. 12, pp. 3037–3045, Dec 1995.
- [9] S. Parkvall, E. Ström, and B. Ottersten, "The impact of timing errors on the performance of linear DS-CDMA receivers," *To appear in IEEE J. Sel. Areas in Comms.*, 1996.

- [10] T. K. Moon, Z. Xie, C. K. Rushforth, and R. T. Short, "Parameter estimation in a multi-user communication system," *IEEE Trans. Communications*, vol. 42, no. 8, pp. 2553–2559, Aug 1994.
- [11] Z. Xie, C. K. Rushforth, R. T. Short, and T. K. Moon, "Joint signal detection and parameter estimation in multiuser communications," *IEEE Trans. Communications*, vol. 41, no. 7, pp. 1208–1215, Aug 1993.
- [12] A. Radovic, "An iterative near-far resistant algorithm for joint parameter estimation in asynchronous CDMA systems," in *Proceedings of 5th Int'l Sym. Personal, Indoor and Mobile Radio Comms.*, The Hague, The Netherlands, Sep 1994, pp. 199–203.
- [13] R. A. Iltis and L. Mailaender, "An adaptive multiuser detector with joint amplitude and delay estimation," *IEEE J. Sel. Areas in Comms.*, vol. 12, no. 5, pp. 774–784, June 1994.
- [14] S. E. Bensley and B. Aazhang, "Subspace-based channel estimation for code division multiple access communication systems," *To appear in IEEE Trans. Communications*, 1996.
- [15] E. G. Ström, S. Parkvall, S. L. Miller, and B. E. Ottersten, "Propagation delay estimation in asynchronous direct-sequence code-division multiple access systems," *IEEE Trans. Communications*, vol. 44, no. 1, pp. 84–93, Jan 1996.
- [16] E. G. Ström, S. Parkvall, S. L. Miller, and B. E. Ottersten, "DS-CDMA synchronization in time-varying fading channels," *To appear in IEEE J. Sel. Areas in Comms.*, 1996.
- [17] D. S. Chen and S. Roy, "An adaptive multiuser receiver for CDMA systems," *IEEE J. Sel. Areas Comms.*, vol. 12, no. 5, pp. 808–816, June 1994.
- [18] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. Communications*, vol. 42, no. 12, pp. 3178–3188, Dec 1994.
- [19] S. L. Miller, "An adaptive direct-sequence code-division multiple-access receiver for multiuser interference rejection," *IEEE Trans. Communications*, vol. 43, no. 2/3/4, pp. 1746–1755, Feb/Mar/Apr 1995.
- [20] T. Söderström and P. Stoica, *System Identification*, Prentice Hall, Hertfordshire, UK, 2nd edition, 1994.
- [21] L. Ljung and T. Soderstrom, *Theory and Practice of Recursive Identification*, MIT Press, Cambridge, MA, 1983.
- [22] S. M. Kay, *Fundamentals of statistical signal processing – Estimation theory*, Prentice Hall International, Englewood Cliffs, NJ, 1993.
- [23] J. M. Mendel, *Lessons in Estimation Theory for Signal Processing, Communications, and Control*, Prentice Hall, Englewood Cliffs, NJ, 1995.
- [24] R. A. Iltis, "Joint estimation of PN code delay and multipath using the extended Kalman filter," *IEEE Trans. Communications*, vol. 38, no. 10, pp. 1677–1685, Oct 1990.
- [25] T. J. Lim and M. D. Macleod, "On-line interpolation using spline functions," *IEEE Signal Proc. Lett.*, vol. 3, no. 5, pp. 144–146, May 1996.

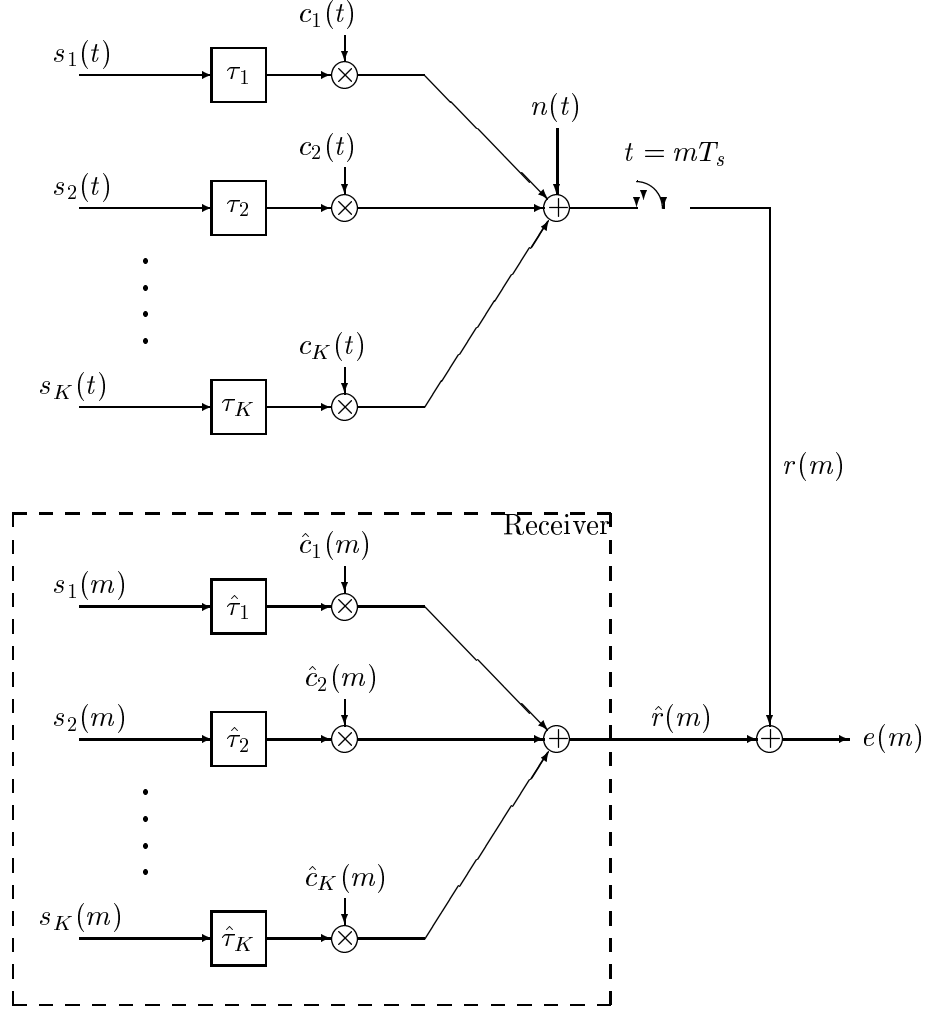


Fig. 1. The asynchronous CDMA signal model and the proposed adaptive multiuser receiver.

MAI (dB)	RMSE (chips)		# of errors	
	MU	SU	MU	SU
30	0.1060	—	3	—
25	0.0962	142.4	0	53
20	0.1114	16.41	0	56
15	0.0869	3.020	0	58
10	0.0860	0.1937	0	1
5	0.0787	0.0430	0	0
0	0.0826	0.0638	0	0

TABLE I

RMSE and error rates measured over 100 bits or 5,000 samples for both multiuser (MU) and single-user (SU) EKF detectors for different MAI levels. The near-far resistance of the proposed MU detector relative to the SU detector is clear.

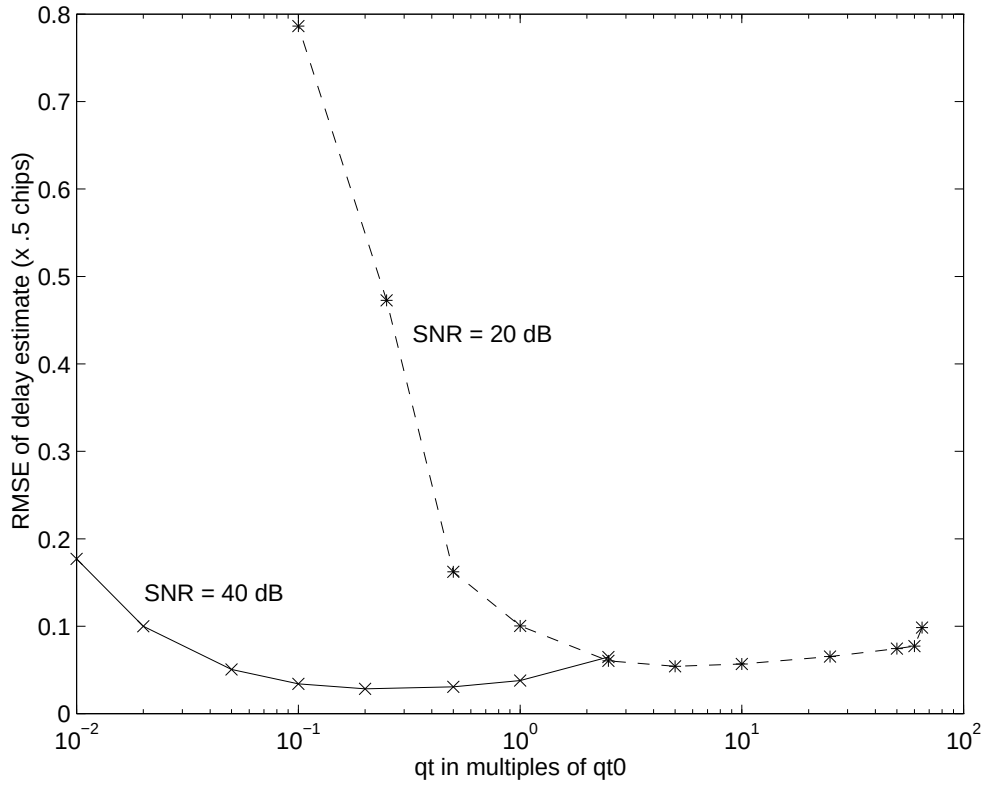


Fig. 2. The root mean square delay estimation error as a function of q_T , defined in the text, parameterized by the signal-to-noise ratio for the weaker user in a two-user system, with an MAI of 10 dB.

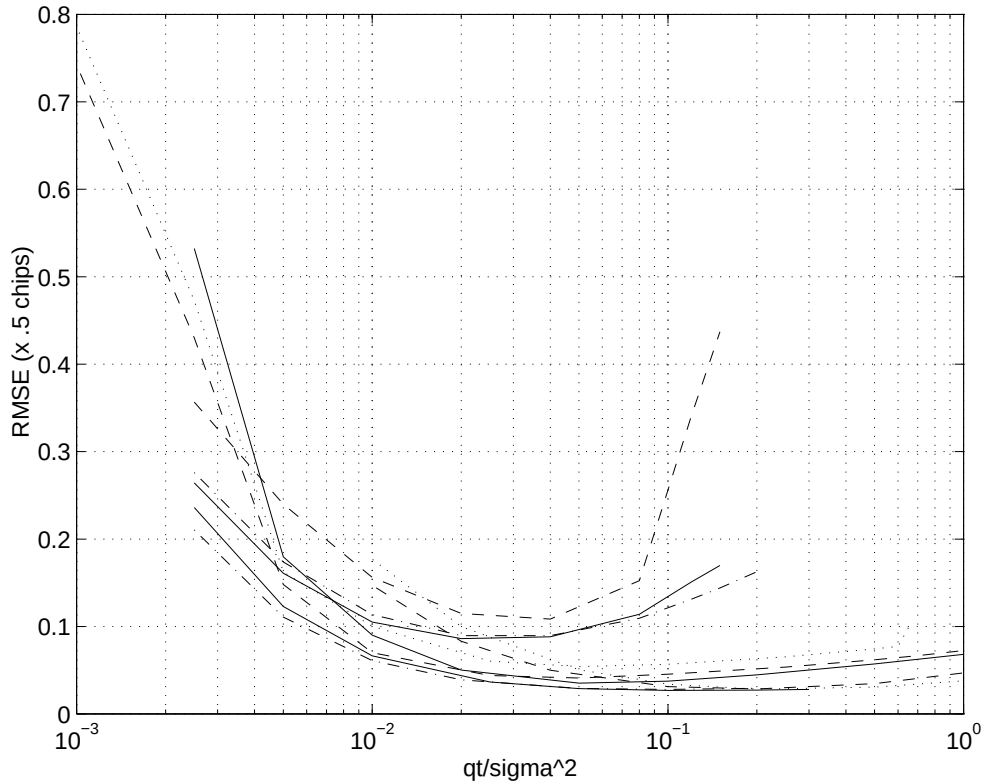


Fig. 3. The RMSE as a function of the ratio q_T/σ^2 for different MAI and SNR levels.

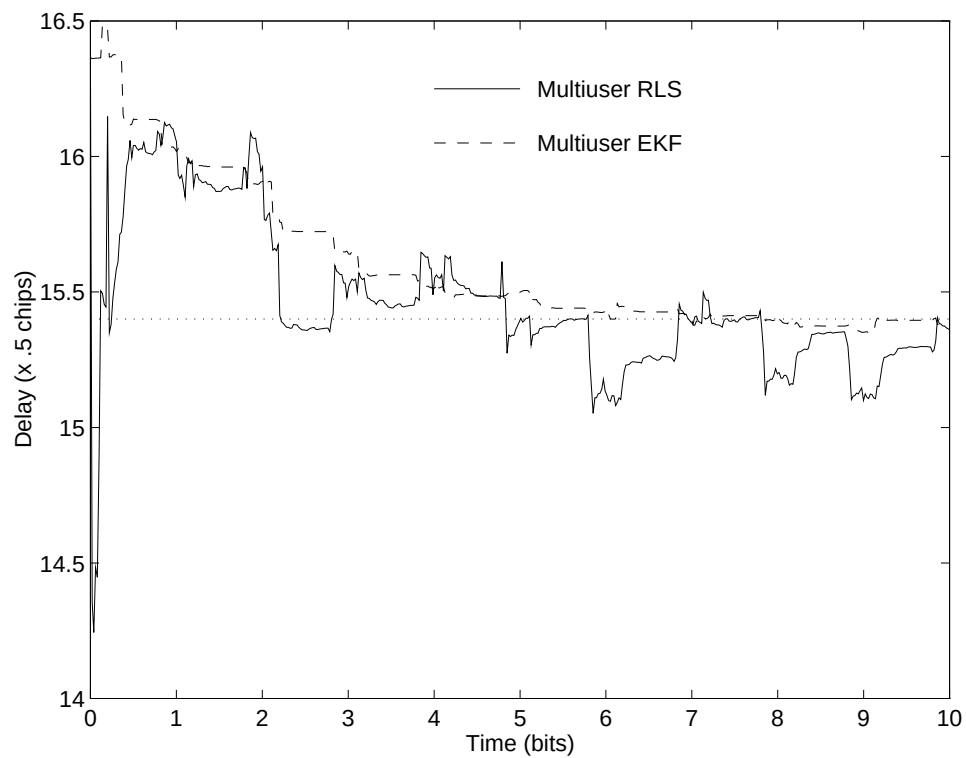


Fig. 4. Convergence of RLS and EKF delay estimates in a time-invariant system.

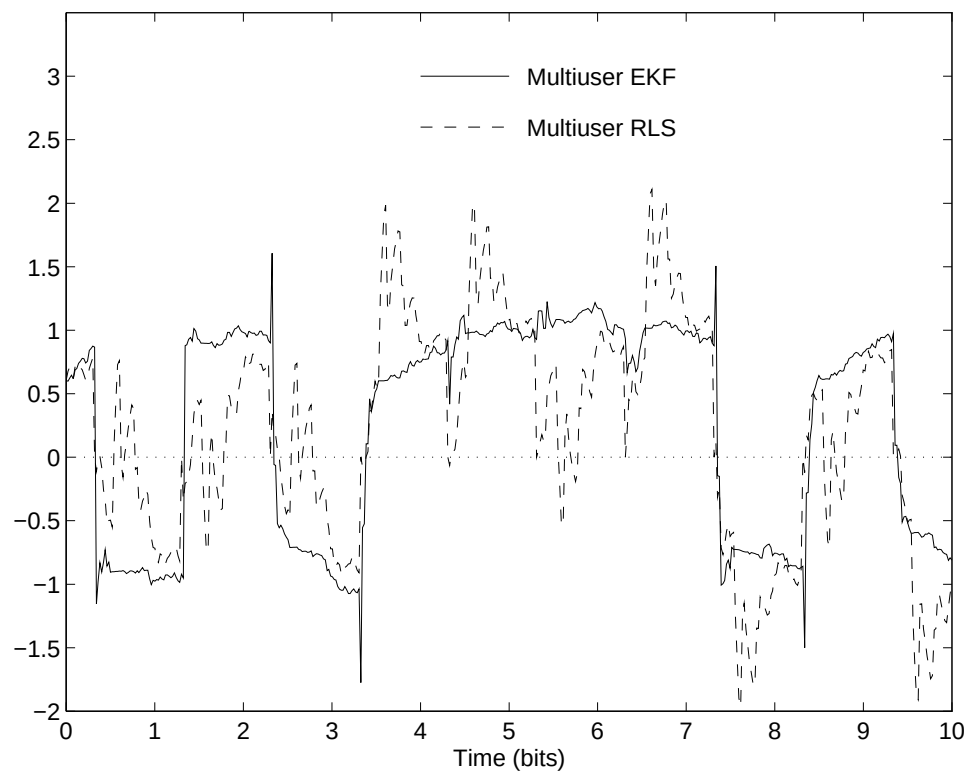


Fig. 5. Convergence of RLS and EKF bit estimates.

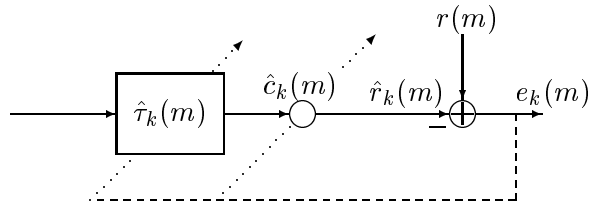


Fig. 6. k th branch of multiuser extension of single-user EKF detector.

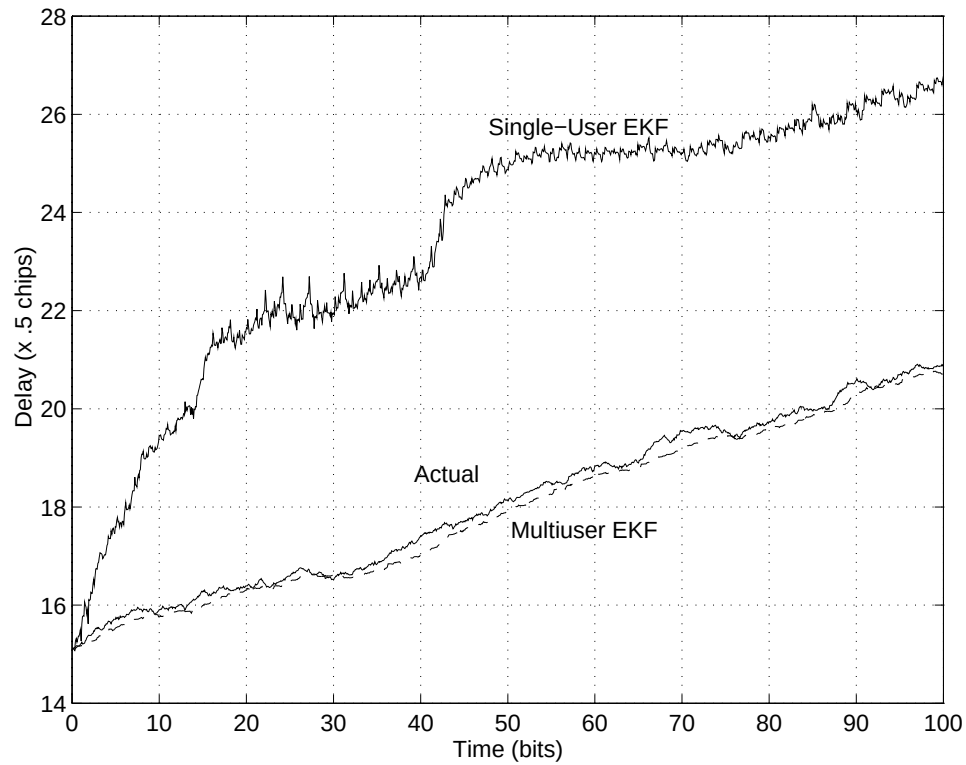


Fig. 7. Estimated and actual delay trajectories for two-user system with an MAI of 15 dB. The single-user system is unable to track the true time delay while the multiuser detector does it adequately.