An Efficient Code-Timing Estimator for DS-CDMA Signals

Dunmin Zheng, Jian Li, Member, IEEE, Scott L. Miller, Member, IEEE, and Erik G. Ström, Member, IEEE

Abstract—In this paper, we present an efficient algorithm for estimating the code timing of a known training sequence in an asynchronous direct-sequence code division multiple access (DS-CDMA) system. The algorithm is a large sample maximum likelihood (LSML) estimator that is derived by modeling the known training sequence as the desired signal and all other signals including the interfering signals and thermal noise as unknown colored Gaussian noise that is uncorrelated with the desired signal. The LSML estimator is shown to be robust against the near-far problem and is also compared with several other code timing estimators via numerical examples. It is found that the LSML approach can offer noticeable performance improvement, especially when the loading of the system is heavy.

I. INTRODUCTION

DIRECT-SEQUENCE code division multiple access (DS-CDMA) has been shown to be a promising technology for future wireless communication networks. Due to the near-far problem for CDMA systems in a multiuser environment, several near-far resistant multiuser detection schemes have been proposed [1]. All of these schemes depend on the knowledge of one or several parameters, including the propagation delay for each user, received power levels, and carrier phases. Therefore, the performance of the multiuser detectors is usually sensitive to the accuracy of the parameter estimates of the signals.

In this paper, we consider parameter estimation in the context of the DS-CDMA systems. In particular, we present an efficient algorithm to estimate the code timing of a known training sequence in an asynchronous DS-CDMA system. Estimation of carrier amplitude and phase is not considered here since this tends to be a much simpler problem once adequate code acquisition is achieved, but such information could easily be obtained using the technique presented. The timing estimator is a large sample maximum likelihood (LSML) estimator that models the known training sequence as the desired signal and all other signals including the interfering signals and thermal noise as unknown colored Gaussian noise that is uncorrelated with the desired signal. The LSML estimator is asymptotically statistically efficient (under the assumption that the background noise + multiple access interference has a Gaussian distribution) as the length of the training sequence goes to infinity. We shall show that the LSML estimator is robust against the near-far problem and compares favorably against other recently proposed near-far resistant code timing estimation schemes.

The rest of this paper is organized as follows. In Section II, we formulate the problem of interest. The LSML estimator for the problem is then developed in Section III. In Section IV, we provide numerical examples illustrating the performance of the LSML estimator and comparing it with three other estimators. Finally, Section V contains our conclusions.

II. PROBLEM FORMULATION

Consider an asynchronous multiuser DS-CDMA system using binary phase shift keying (BPSK) modulation and operating over an additive Gaussian noise channel. The bit duration is denoted \( T \), and each bit consists of \( N \) chips with duration \( T_c = T/N \), where \( N \) is an integer. The spreading waveform assigned to the user of interest has the form

\[
\alpha(t) = \sum_{n=-N/2}^{N/2-1} a_n \Pi_{T_c}(t - nT_c)
\]

where \( a_n \in \{-1, +1\} \), and \( \Pi_{T_c}(t) \) denotes a unit rectangular pulse over the chip period \( T_c \). We use \( g_0(m) \in \{-1, +1\} \) to denote the \( m \)th data bit of this user. The data bits are assumed to be a randomly generated sequence of 1’s and -1’s (with equal probabilities) known to both the transmitter and receiver. Then, the baseband signal of the desired user can be written as

\[
s_0(t) = \sum_{m=-M/2}^{M/2-1} g_0(m) a(t - mT_c) \tag{1}
\]

The transmitted signal is formed by multiplying \( s_0(t) \) with the carrier \( \sqrt{2P_0} \cos(\omega_c t + \theta_0) \), where \( P_0 \) is the user’s transmitted power, and \( \theta_0 \) is a random carrier phase uniformly distributed between 0 and \( 2\pi \).

The received signal can be written as

\[
r(t) = s_0(t - \tau_0) \sqrt{2P_0} \cos(\omega_c t + \theta_0) + \sum_{k=1}^{K-1} s_k(t - \tau_k) \sqrt{2P_k} \cos(\omega_c t + \theta_k) + n(t) \tag{2}
\]

where \( \tau_0 \in [0, T] \) is the relative propagation delay of the desired signal with respect to the receiver of interest, \( \theta_0 = \theta_0 - \omega_c \tau_0 \), and \( s_k(t), \tau_k, P_k \) and \( \theta_k, k = 1, \ldots, K - 1 \) are...
defined similarly as $s_0(t), \tau_0, p_0$, and $\theta_k$, respectively, for the $K-1$ interfering users, and $n(t)$ is the additive noise.

Assume that the receiver front-end consists of an IQ mixer followed by an integrate-and-dump filter with integration time $T_c$. The equivalent received complex sequence $r(t)$ can be written as

$$
r(t) = \sqrt{P_0} e^{j\theta_0} \frac{1}{T_c} \int_{(l-1)T_c}^{lT_c} s_0(t-\tau_0) \, dt + c(l) \tag{3}
$$

where $c(l)$ is the sum of the multiple access interferences (MAI) and the additive noise and is given by

$$
c(l) = \sum_{k=1}^{K-1} \sqrt{P_k} e^{j\theta_k} \frac{1}{T_c} \int_{(l-1)T_c}^{lT_c} s_k(t-\tau_k) \, dt + n(l) \tag{4}
$$

with $n(l)$ denoting zero-mean complex white Gaussian noise with variance $\sigma^2 = E[n(l)^2]$.

Let the received vector $\mathbf{r}(m) \in \mathbb{C}^{N \times 1}$ during the $m$th bit interval be defined as

$$
\mathbf{r}(m) = [r(mN+1), r(mN+2), \ldots, r(mN+N)]^T. \tag{5}
$$

Let the MAI and noise vector $\mathbf{c}(m) \in \mathbb{C}^{N \times 1}$ be formed from $c(l)$ in a similar way. Then,

$$
\mathbf{r}(m) = \mathbf{r}_0(m) + \mathbf{c}(m) \tag{6}
$$

where $\mathbf{r}_0(m)$ is the contribution of the desired user and can be written as

$$
\mathbf{r}_0(m) = \beta_0 [a_1(\tau_0), a_2(\tau_0)] [z_1(m), z_2(m)] \tag{7}
$$

with

$$
z_1(m) = \frac{[g_0(m) + g_0(m-1)]}{2}, \quad z_2(m) = \frac{[g_0(m) - g_0(m-1)]}{2}
$$

and $\beta_0 = \sqrt{T_c e^{j\theta_0}}$. The vectors $a_1(\tau_0) \in \mathbb{R}^{N \times 1}$ and $a_2(\tau_0) \in \mathbb{R}^{N \times 1}$ are given by

$$
a_1(\tau_0) = \left[ \frac{\delta_0}{T_c} \mathbf{J}[p+1] + \left(1 - \frac{\delta_0}{T_c}\right) \mathbf{J}[p+1] \right] \mathbf{a}_0 \tag{8}
$$

and

$$
a_2(\tau_0) = \left[ \frac{\delta_0}{T_c} \mathbf{J}[p+1] + \left(1 - \frac{\delta_0}{T_c}\right) \mathbf{J}[p+1] \right] \mathbf{a}_0 \tag{9}
$$

respectively, where $\tau_0 = p_0 T_c + \delta_0 p_0$ is an integer, and $\delta_0 \in [0, T_c]$. The desired user’s discrete-time code vector $\mathbf{a}_0 \in \mathbb{R}^{N \times 1}$ is defined as

$$
\mathbf{a}_0 = [a_0(N-1), a_0(N-2), \ldots, a_0(0)]^T,
$$

and

$$
c_0(l) = \frac{1}{T_c} \int_{(l-1)T_c}^{lT_c} c_0(t) \, dt. \tag{10}
$$

The matrix $\mathbf{J}_s^{(p)} \in \mathbb{R}^{N \times N}$ is defined in matrix block form as

$$
\mathbf{J}_s^{(p)} = \begin{bmatrix} \mathbf{I}_p & \mathbf{I}_{N-p} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad s = \pm 1 \tag{11}
$$

where $\mathbf{I}_p$ denotes the $p \times p$ identity matrix. Thus, $\mathbf{r}(m)$ can be written as

$$
\mathbf{r}(m) = \beta_0 \mathbf{A}(\tau_0) \mathbf{z}(m) + \mathbf{c}(m) \in \mathbb{C}^{N \times 1} \tag{12}
$$

where

$$
\mathbf{A}(\tau_0) = [a_1(\tau_0), a_2(\tau_0)] \in \mathbb{R}^{N \times 2} \quad \text{and} \quad \mathbf{z}(m) = \begin{bmatrix} z_1(m), z_2(m) \end{bmatrix}^T \in \{-1, 0, +1\}^{2\times 1}.
$$

The vector $\mathbf{e}(m)$ is assumed to be independent of the desired signal and to be a circularly symmetric complex Gaussian random vector with zero-mean and arbitrary covariance matrix $\mathbf{Q}$ that satisfies

$$
E[\mathbf{e}(m)\mathbf{e}^H(m)] = \mathbf{Q} \delta_{ij} \tag{13}
$$

where $(\cdot)^H$ denotes the complex conjugate transpose, and $\delta_{ij}$ is the Kronecker delta. The unknown covariance matrix $\mathbf{Q}$ models both thermal noise and all other interference signals including MAI. Note that when no training sequence is available and the desired data bits are assumed to be either random or unknown deterministic, the problem of propagation delay estimation is ill defined if $\mathbf{Q}$ is unknown. For our case, since $\{g_0(m)\}_{m=0}^{M-1}$ is known, $\{\mathbf{z}(m)\}_{m=0}^{M-1}$ is a known deterministic sequence.

The problem of interest herein is to estimate $\tau_0$ from the $\mathbf{r}(m)$.

III. LARGE-SAMPLE MAXIMUM LIKELIHOOD (LSML) ESTIMATOR

The log-likelihood function of the receiver output vector $\mathbf{r}(m), m = 1, 2, \ldots, M$, is proportional to

$$
-\ln|\mathbf{Q}| - \text{tr} \left\{ \mathbf{Q}^{-1} \frac{1}{M} \sum_{m=1}^{M} [\mathbf{r}(m) - \mathbf{D}\mathbf{z}(m)][\mathbf{r}(m) - \mathbf{D}\mathbf{z}(m)]^H \right\} \tag{14}
$$

where $\text{tr}$ denotes the determinant of a matrix, and

$$
\mathbf{D} = \beta_0 \mathbf{A}(\tau_0). \tag{15}
$$

It can be shown that (14) is maximized (for a fixed $\mathbf{D}$) by

$$
\mathbf{Q} = \frac{1}{M} \sum_{m=1}^{M} [\mathbf{r}(m) - \mathbf{D}\mathbf{z}(m)][\mathbf{r}(m) - \mathbf{D}\mathbf{z}(m)]^H \tag{16}
$$

and $\mathbf{D}$ may be obtained by minimizing the following cost function

$$
F_1 = \left| \frac{1}{M} \sum_{m=1}^{M} [\mathbf{r}(m) - \mathbf{D}\mathbf{z}(m)][\mathbf{r}(m) - \mathbf{D}\mathbf{z}(m)]^H \right|. \tag{17}
$$

Minimizing the cost function gives an unstructured estimate $\hat{\mathbf{D}}$ of $\mathbf{D}$ [2]:

$$
\hat{\mathbf{D}} = \mathbf{R}_{2\mathbf{r}} \hat{\mathbf{R}}_{zz}^{-1} \tag{18}
$$

where

$$
\mathbf{R}_{2\mathbf{r}} = \frac{1}{M} \sum_{m=1}^{M} \mathbf{z}(m)\mathbf{z}^H(m) \tag{19}
$$

and

$$
\hat{\mathbf{R}}_{zz} = \frac{1}{M} \sum_{m=1}^{M} \mathbf{z}(m)\mathbf{z}^H(m). \tag{20}
$$
By using (18) with (16), \( \hat{Q} \) may be rewritten as
\[
\hat{Q} = \hat{R}_{TT} - \hat{R}_{ZZ}^H \hat{R}_{ZZ}^{-1} \hat{R}_{TT}
\]  
where
\[
\hat{R}_{TT} = \frac{1}{M} \sum_{m=1}^{M} r(m)r^H(m).
\]

Consider now the structure of \( \hat{D} \). It has been shown in [2] that minimizing \( \Sigma(F_3) \) is asymptotically (for large \( M \)) equivalent to minimizing
\[
F_2 = \text{tr}[\hat{R}_{ZZ}(\hat{D} - \hat{D})^H \hat{Q}^{-1}(\hat{D} - \hat{D})].
\]

Note that
\[
E[z_2(m)z_2(m)] = \{E[z_2^2(m)] - E[z_2(m-1)]\}/4 = 0.
\]

Since \( \theta_0(m) \) is an independently and identically distributed binary sequence, \( \hat{R}_{ZZ} \) is a diagonal matrix with equal diagonal elements. Then, minimizing (23) is equivalent to minimizing the following cost function:
\[
F_3 = \text{tr}[\hat{d}_1 - \alpha_0 \hat{a}_1(\theta_0)]^H \hat{Q}^{-1}[\hat{d}_1 - \alpha_0 \hat{a}_1(\theta_0)]
+ \text{tr}[\hat{d}_2 - \alpha_0 \hat{a}_2(\theta_0)]^H \hat{Q}^{-1}[\hat{d}_2 - \alpha_0 \hat{a}_2(\theta_0)]
\]

where \( \hat{d}_1 \) and \( \hat{d}_2 \) denote the first and the second columns of \( \hat{D} \), respectively.

From (24), the LSML estimate of \( \beta_0 \) may be written as a function of \( \theta_0 \)
\[
\hat{\beta}_0(\theta_0) = \frac{\hat{a}_1^H(\theta_0) \hat{Q}^{-1/2} \hat{d}_1 + \hat{a}_2^H(\theta_0) \hat{Q}^{-1/2} \hat{d}_2}{\hat{a}_1^H(\theta_0) \hat{Q} \hat{a}_1(\theta_0) + \hat{a}_2^H(\theta_0) \hat{Q} \hat{a}_2(\theta_0)}.
\]

Thus, the LSML estimate of \( \theta_0 \) can be determined by
\[
\hat{\theta}_0 = \arg \max_{\theta_0} \left\{ \frac{\hat{a}_1^H(\theta_0) \hat{Q}^{-1/2} \hat{d}_1 + \hat{a}_2^H(\theta_0) \hat{Q}^{-1/2} \hat{d}_2}{\hat{a}_1^H(\theta_0) \hat{Q} \hat{a}_1(\theta_0) + \hat{a}_2^H(\theta_0) \hat{Q} \hat{a}_2(\theta_0)} \right\}.
\]

Let
\[
\hat{d}_1 = \hat{Q}^{-1/2} \hat{d}_1, \quad \hat{d}_2 = \hat{Q}^{-1/2} \hat{d}_2
\]

and
\[
\hat{a}_1(\theta_0) = \hat{Q}^{-1/2} \hat{a}_1(\theta_0), \quad \hat{a}_2(\theta_0) = \hat{Q}^{-1/2} \hat{a}_2(\theta_0).
\]

Then, (25) and (26) can be rewritten, respectively, as
\[
\hat{\beta}_0(\theta_0) = \frac{\hat{a}_1^H(\theta_0) \hat{d}_1 + \hat{a}_2^H(\theta_0) \hat{d}_2}{\|\hat{a}_1(\theta_0)\|^2 + \|\hat{a}_2(\theta_0)\|^2}
\]

and
\[
\hat{\theta}_0 = \arg \min_{\theta_0} \|\hat{d}_1 - \beta_0(\theta_0) \hat{a}_1(\theta_0)\|^2
+ \|\hat{d}_2 - \beta_0(\theta_0) \hat{a}_2(\theta_0)\|^2.
\]

Further, (30) can also be simplified to
\[
\hat{\theta}_0 = \arg \max_{\theta_0} \left\{ \frac{\|\hat{a}_1^H(\theta_0) \hat{d}_1 + \hat{a}_2^H(\theta_0) \hat{d}_2\|^2}{\|\hat{a}_1(\theta_0)\|^2 + \|\hat{a}_2(\theta_0)\|^2} \right\}.
\]

We have shown in Appendix A that the maximization in (31) is equivalent to solving a second-order polynomial for each chip interval (i.e., \( N \) quadratic equations must be solved). This is similar to the optimization required for other timing estimators such as those that will be presented in the next section for comparison.

The LSML algorithm may be summarized by the following steps:

Step 1: Compute \( \hat{D} \) with (18).  
Step 2: Compute \( \hat{Q} \) with (21).  
Step 3: Determine \( \hat{\theta}_0 \), as described in Appendix A.

We remark that according to the general theory of maximum likelihood estimation, this LSML estimator is asymptotically (for large \( M \)) statistically efficient.

IV. NUMERICAL EXAMPLES

In this section, we will present several numerical examples showing the performance of the LSML estimator. In order to put the performance of the LSML estimator in a proper context, we compare the simulation results with that of three other timing estimation techniques that are briefly described in the following.

A. The Correlator [3]

This is the conventional approach to timing estimation whereby the received signal is correlated with time delayed versions of the known code sequence, and the timing estimate is simply the value of the time delay that maximizes the correlation. Mathematically, this estimate is given by
\[
\hat{\theta}_0 = \arg \max_{\theta_0} \left\{ \frac{\|\hat{a}_1^H(\theta_0) \hat{r}\|^2}{\|\hat{a}_1(\theta_0)\|^2} \right\}, \quad \hat{r} = \frac{1}{M} \sum_{m=1}^{M} r(m).
\]

This approach requires that the training sequence consists of all ones (i.e., no data on the spread carrier). The correlator is computationally simple, and it is well known to be optimal for a single user in the presence of white Gaussian noise only but can be highly suboptimal in the presence of MAI, especially if a significant near-far problem exists.

B. The MMSE-Based Timing Estimator [4], [5]

This approach was developed by Smith and Miller [4], [5] and is based on a near-far resistant single-user detector commonly referred to as the MMSE detector, which has been extensively studied by many authors (e.g., [6]–[10]). In short, the MMSE receiver computes a receiver vector \( \hat{w} \in \mathbb{C}^{N \times 1} \), which is chosen to minimize the mean squared error
\[
J = E[\|g(\theta_0) - \hat{w}^H \hat{r}(m)\|^2].
\]

In practice, the receiver vector is computed using conventional adaptive filtering techniques such as the least mean square (LMS) or the recursive least square (RLS) algorithms. Once the receiver vector is computed, the timing can be estimated from this vector using a correlator-type approach. The MMSE timing estimator is given by
\[
\hat{\theta}_0 = \arg \max_{\theta_0} \left\{ \frac{\|\hat{a}_1^H(\theta_0) \hat{w}\|^2}{\|\hat{a}_1(\theta_0)\|^2} \right\}.
\]
This technique also requires that the training sequence be all
ones, although this restriction can be relaxed with a simple
modification [11]. The additional complexity of this approach
beyond that of the correlator is only that needed to drive the
adaptive algorithm that computes the receiver vector \( \mathbf{w} \). If
LMS is used, this requires \( O(N) \) operations per bit, whereas
if RLS is used, \( O(N^2) \) operations per bit are needed. In the
simulation results to be presented, only RLS is considered
since we found the MMSE timing estimator when driven by
the LMS algorithm to perform only slightly better than the
correlator.

C. The MUSIC Timing Estimator [12], [13]

Multiple signal classification (MUSIC) is a subspace-based
approach to parameter estimation, whereby the vector space
of the received vector \( \mathbf{r}(m) \) is decomposed into a signal space
(consisting of the subspace spanned by all the CDMA signals)
and a noise subspace (which is the orthogonal complement
to the signal subspace). Since the known desired signal is
part of the signal subspace, it must be orthogonal to the noise
subspace, and hence, the timing estimator is taken to be the
value of the timing delay for which the known code sequence
is nearest to being orthogonal to an estimate of the noise
subspace. The MUSIC timing estimator is given by

\[
\hat{\tau}_0 = \arg \min_{\tau_0} \left\{ \frac{|a^H(\tau_0) \mathbf{E}_n|}{|a(\tau_0)|^2} + \frac{|a^H(\tau_0) \mathbf{E}_y|}{|a(\tau_0)|^2} \right\}
\]  

(35)

where \( \mathbf{E}_n \) is an estimate of the noise subspace whose columns
are the eigenvectors corresponding to the \( N - 2K \) smallest
eigenvalues of the sample autocorrelation matrix \( \mathbf{R}_{\mathbf{y}\mathbf{y}} \). The
MUSIC cost function in (35) is slightly different from the
one proposed in the original work by Ström et al. [12], but we
found it to work just as well and results in a computa-
tionally simpler algorithm. The MUSIC timing estimator,
like the LSML estimator, does not require an all-ones training
sequence. However, unlike the LSML approach, the MU-
SIC approach does not even need training and can work
when unknown data is being sent. The operations required
in minimizing the cost function of (35) is similar to the
other approaches. However, the MUSIC algorithm is more
complex than either the correlator or MMSE-based approaches
since the eigendecomposition of \( \mathbf{R}_{\mathbf{y}\mathbf{y}} \) requires roughly \( O(N^3) \)
operations. If the eigendecomposition is done recursively, the
complexity could be \( O(N^2) \) per bit, which would make it
similar to the MMSE approach when the RLS algorithm is
used. The MUSIC approach has the disadvantage that it needs
to know the number of users and that it will not function if
\( K \geq N/2 \) since, in that case, the noise subspace has zero rank.
This problem could be overcome in practice by identifying the
most dominant users and including them in the signal subspace
and lumping the remaining users in with the noise. The noise
would no longer be white, but we suspect MUSIC would still
work reasonably well in such a scenario. We have made no
attempt to incorporate such a modification into our simulations.

The above timing estimation strategies are not exhaustive
in the sense that other schemes have been presented in the
literature. However, they represent a good cross-section of
the various different approaches that have been presented and
provide a good context with which to compare the performance
of the proposed LSML estimator. The reader is referred to [14]
and [15] for some examples of other possible approaches not
considered here.

In the timing estimation approaches that involve a training
sequence, it is assumed that the transmitter and receiver have
aligned their clocks to roughly within a bit interval. This could
be done, for example, on a side “signaling channel,” where a
call is initially set up. However, regardless of how this is done,
it is assumed in this work that the job of the timing estimator
is to estimate the timing of the received signal modulo one
bit interval.

In the simulation results to follow, all users were assigned
Gold sequences of chip length \( N = 31 \). The performance
of the estimators in each of the examples was obtained from
250 Monte Carlo simulations. The received signal is scaled
so that the carrier power for the desired user is one, i.e.,
\( \mathbf{P}_1 = 1 \). In the first two experiments (Figs. 1–4), all interfering
users were given a random received power with a log-normal
distribution with a mean 10 dB above the desired signal and
a standard deviation of 10 dB. That is \( \mathbf{P}_k = 10 \mathbf{E}_{\mathbf{p}} / \mathbf{E}_{\mathbf{p}} \), where
\( \mathbf{E}_{\mathbf{p}} \sim N(10, 10) \), and this set of interfering powers is changed
for each Monte Carlo run. The additive noise \( \mathbf{n}(t) \) is zero-mean
white Gaussian noise with power spectral density of \( N_0 / 2 \).
Hence, the noise samples, \( \mathbf{n}(\mathbf{t}) \) in (4), at the output of the chip
matched filter (after being normalized as indicated above) are
complex zero mean Gaussian random variables with variance
of \( \sigma_n^2 = E[|\mathbf{n}(\mathbf{t})|^2] = N_0 / E_{\mathbf{p}} \), where \( E_{\mathbf{p}} \) is the received
energy per bit for the desired user. The carrier phases \( \theta_k \),
timing offsets \( \tau_k \), and data bits of all users are independent of
each other. The carrier phases and timing offsets are uniformly
distributed over \( [0, 2\pi) \) and \( [0, T] \), respectively, whereas all
data bits are equally likely to be \(+1\) or \(-1\).

We consider two performance measures relevant to acqui-
sition and tracking of code timing of DS-CDMA signals. The
first emphasizes the acquisition aspect whereby the estimator
tries to get a rough estimate of the code timing close enough
that some code tracking loop could then work on driving the

![Fig. 1. Probability of correct acquisition; \( K = 10 \) users, \( N = 31 \) chips/bit, \( \mathbf{E}_{\mathbf{p}} / N_0 = 10 \) dB, log-normally distributed interfering powers.](image)
Fig. 2: Root mean squared error given correct acquisition; $K = 10$ users, $N = 31$ chips/bit, $E_b/N_0 = 10$ dB, log-normally distributed interfering powers. The RMSE is normalized with respect to $T_r$.

Fig. 3: Probability of correct acquisition; $M = 100$ bits, $N = 31$ chips/bit, $E_b/N_0 = 10$ dB, log-normally distributed interfering powers.

Timing error to zero. In this work, we define correct acquisition to be the event $|\hat{\tau}_0 - \tau_0| < T_r/2$. That is, correct acquisition has occurred when the estimate is within a half chip of the true value. The next measure would be relevant if we wanted to use these estimators to replace the code tracking loop. In this case, it is generally assumed that correct code acquisition has already been achieved; therefore, we measure the root mean squared estimation error (RMSE) given correct acquisition. Therefore, when we refer to RMSE, we mean

$$\text{RMSE} = \sqrt{E[(\hat{\tau}_0 - \tau_0)^2 | (|\hat{\tau}_0 - \tau_0| \leq T_r/2)].} \quad (36)$$

The results of the first experiment are shown in Figs. 1 and 2. The number of users was fixed to be $K = 10$, and $E_b/N_0 = 10$ dB was used. All other parameters are as described above. The length of the observation time $M$ (in bits) was varied from 5 to 100 in increments of 5 bits. Fig. 1 shows the code acquisition probability for the LSML estimator along with the three other schemes presented at the beginning of this section. The first thing that stands out from this figure is the poor performance of the correlator. This is due to the fact that the interfering users are generally much stronger than the desired user (10 dB on average). While the other three techniques are robust to the near-far problem, the conventional correlator degrades horribly in the presence of a significant near-far effect. Note that the LSML estimator cannot form a timing estimate until the number of bits received is $\geq N$ ($\approx 31$ in this case) since the sample autocorrelation matrix must be full rank in order to form an estimate of $Q^{-1}$. It may be possible to get around this problem using some sort of pseudoinverse, but we made no attempt to do so. Once the observation window is increased beyond $N$, the LSML works quite well. In situations where correct acquisition has to be achieved with a reasonable probability with $M \leq N$, it would seem that the MMSE timing estimator (with RLS) is the best option. Fig. 2 shows the RMSE for the four timing estimators in the same environment. It is seen that the LSML and MUSIC estimators offer similar performance in terms of RMSE and both perform substantially better than either the correlator or the MMSE as long as $M \geq N$.

In the next experiment, we investigate the performance of the estimators as the number of users is varied. The observation time was fixed at $M = 100$ bits, and the number of users was varied from 3 to 30. All other parameters are as before. The correct acquisition probability is shown in Fig. 3 with the RMSE shown in Fig. 4. As mentioned earlier, the MUSIC estimator cannot function when $K \geq N/2$ ($K > 15$ in this case). Once again, the correlator is seen to be severely near-far limited and give relatively poor performance even when the loading is light. The LSML and MMSE estimators seem to be the only ones that can give reliable timing estimates when the number of users becomes large with a definite advantage to the LSML technique.
Fig. 5. Comparison of simulation results and the Cramer–Rao bound; \(N = 31\) chips/bit, \(K = 15\) users, \(E_b/N_0 = 22\) dB, \(P_k = P_0\). The RMSE is normalized with respect to \(T_c\).

Fig. 6. Comparison of simulation results and the Cramer–Rao bound; \(N = 31\) chips/bit, \(K = 15\) users, \(M = 100\) bits, \(P_k = P_0\). The RMSE is normalized with respect to \(T_c\).

Our last set of results involve comparing the performance of the LSML algorithm with the Cramer–Rao bound (CRB), which is derived in Appendix B. It should be pointed out that the CRB presented here is based on a model consisting of a single user in the presence of colored Gaussian noise with unknown covariance matrix \(Q\). In our simulations, while the background white noise was Gaussian, the contribution from the interfering users is not necessarily Gaussian. However, as can be seen in Figs. 5 and 6, using this Gaussian approximation for the interference plus noise seems to give a pretty good indication of how the LSML algorithm will perform.

In Fig. 5, the performance of the LSML estimator is compared with the CBR as a function of the observation time \(M\). It is seen that as \(M\) gets large, the performance of the LSML estimator approaches the CRB. Fig. 6 shows the same comparison for fixed \(M (=100)\) and varying values of \(E_b/N_0\). It is seen that even when \(E_b/N_0\) is quite small, the CRB does a pretty good job of predicting the performance of the LSML estimator. Hence, the CRB presented in Appendix B offers a relatively simple way to approximately evaluate the performance of the LSML estimator without having to run long simulations. It should be pointed out that in generating these last two figures, the system contained \(K = 15\) users, and perfect power control was assumed (i.e., \(P_k/P_0 = 1\)).

V. CONCLUSIONS

We have presented a new technique for code timing estimation in DS-CDMA systems. The technique is near-far resistant and, hence, would be applicable for use in a system employing multiuser detection where power control requirements could be relaxed. The technique provides a timing estimate for a single user’s signal when that desired signal is sending a known pseudorandom training sequence. The technique could easily be extended to estimate the timing of multiple signals each transmitting an uncorrelated training sequence through the use of several estimators in parallel, which could share some of the computations and thus reduce the overall complexity per user. The LSML estimator was compared with a conventional correlator-type timing estimator as well as two other recently proposed near-far resistant timing estimation strategies. Based on these comparisons, it seems that the LSML estimator would be most useful in situations where the system is heavily loaded. Since it is often the process of code acquisition that limits the capacity of a DS-CDMA system, use of the LSML estimator could enhance the capacity of a DS-CDMA system.

One major limitation of the LSML estimator is that it is not clear to us how one would extend the technique to work in a fading channel. While both the MUSIC- and MMSE-based timing estimators can be modified to work in a fading channel, the LSML cannot. Hence, the LSML approach would only be applicable to systems where the channel remains reasonably static over the duration of time in which code acquisition is performed. This would be the case in an indoor wireless type of environment where fading is relatively slow, and data rates tend to be quite high. In that case, the duration of an observation window on the order of 100 bits may be quite short compared with the correlation time of the channel. In addition, we believe the technique will also work in a situation where the fading rate may be high, but the variations in the amplitude and phase are not severe. That may be the case in some satellite systems where the fading tends to be Rician rather than Rayleigh. In any event, we leave the topic of evaluating the performance of the LSML estimator in fading channels as a topic for future research.

APPENDIX A

MAXIMIZATION OF (31)

Denote the objective function in (31) by \(J(\tau_0)\),

\[
J(\tau_0) = \frac{|\hat{a}_1^R(\tau_0)\tilde{a}_1 + \hat{a}_2^R(\tau_0)\tilde{a}_2|^2}{|\tilde{a}_1(\tau_0)|^2 + |\tilde{a}_2(\tau_0)|^2}.
\]
Consider $J(\tau_0)$ for $\tau_0 \in [0, T_c + 1, \ldots, T_c]$ for $p = 0, 1, \ldots, N - 1$. From the definition of $a_1(\tau_0)$ and $a_2(\tau_0)$, we have

$$\hat{a}_1(\tau_0) = Q^{-1/2}[a_1(pT_c) \ a_1((p+1)T_c)] \begin{bmatrix} 1 - \mu \\ \mu \end{bmatrix}$$
$$= \bar{A}_1(pT_c)\mu$$

$$\hat{a}_2(\tau_0) = Q^{-1/2}[a_2(pT_c) \ a_2((p+1)T_c)] \begin{bmatrix} 1 - \mu \\ \mu \end{bmatrix}$$
$$= \bar{A}_2(pT_c)\mu$$

where $\mu = [1 - \mu \ \mu]^T$, $\mu = \delta_0/\tau_c \in [0, 1]$, and

$$\bar{A}_1(pT_c) = Q^{-1/2}[a_1(pT_c) \ a_1((p+1)T_c)]$$
$$\bar{A}_2(pT_c) = Q^{-1/2}[a_2(pT_c) \ a_2((p+1)T_c)]$$

The explicit dependence of $\bar{A}_1$ and $\bar{A}_2$ on $pT_c$ will be dropped for notational convenience.

The numerator of $J(\tau_0)$ can now be expressed as $N(\mu) = \mu^T G \mu$, where

$$G = (\bar{A}_1^H d_1 + \bar{A}_2^H d_2)(\bar{A}_1^H d_1 + \bar{A}_2^H d_2)^H.$$ Similarly, the denominator of $J(\tau_0)$ may be written as $D(\mu) = \mu^T H \mu$, where

$$H = \bar{A}_1^H \bar{A}_1 + \bar{A}_2^H \bar{A}_2.$$ Now, $J(\tau_0) = N(\mu)/D(\mu)$ is seen to be a rational function of two second-order polynomials. Any extreme point $\hat{\mu}$ in the differentiable region of $J(\tau_0)$ must therefore satisfy the equation $S(\mu) = N'(\mu)/D(\mu) - N(\mu)/D'(\mu) = 0$.

Let $G_{ij}$ and $H_{ij}$ denote the $ij$th element of $G$ and $H$, respectively. Then, $N(\mu)$ and $D(\mu)$ can be written as

$$N(\mu) = [G_{11} + G_{22} - G_{12} - G_{21}]\mu^2$$
$$+ [G_{11} + G_{22} - 2G_{12}]\mu + G_{11}$$
$$\triangleq n_2\mu^2 + n_1\mu + n_0$$

$$D(\mu) = [H_{11} + H_{22} - H_{12} - H_{21}]\mu^2$$
$$+ [H_{11} + H_{22} - 2H_{12}]\mu + H_{11}$$
$$\triangleq d_2\mu^2 + d_1\mu + d_0.$$ Differentiating with respect to $\mu$ yields

$$N'(\mu) = \frac{dN(\mu)}{d\mu} = 2n_2\mu + n_1$$
$$D'(\mu) = \frac{dD(\mu)}{d\mu} = 2d_2\mu + d_1$$

and hence

$$S(\mu) = (n_2d_1 - n_1d_0)\mu^2 + 2(n_2d_0 - n_0d_2)\mu$$
$$+ n_1d_0 - n_0d_1$$
$$\triangleq s_2\mu^2 + s_1\mu + s_0.$$ (37)

The roots to $S(\mu)$ are

$$\hat{\mu} = \frac{1}{2s_2} \left[-s_1 \pm \sqrt{s_1^2 - 4s_0s_2}\right].$$ (38)

Let $T$ denote the set of costs corresponding to the candidate estimates of $\tau_0$. We can find the maximizer $\hat{\tau}_0$ of $J(\tau_0)$ as follows:

1. Let $T = \{J(0), J(T_c), \ldots, J([N - 1]T_c)\}$.
2. For $p = 0, 1, \ldots, N - 1$, do the following:
   a) Compute the coefficients to $S(\mu)$ according to (37).
   b) Find the roots $\hat{\mu}_1$ and $\hat{\mu}_2$ according to (38).
   c) If a root $\hat{\mu}_i \in [0, 1]$, then add $J(p + \hat{\mu}_iT_c)$ to $T$.
3. Choose $\hat{\tau}_0$ to be the $\tau$ corresponding to the largest element of $T$.

This procedure, which only contains noniterative computations, guarantees that the global maximum of $J(\tau)$ is found. Note that $\{J(0), J(T_c), \ldots, J([N - 1]T_c)\}$ must be added to $T$ since $J(\tau_0)$ is not always differentiable for $\tau_0 = pT_c$.

APPENDIX B

CRAMER-RAO BOUND

We briefly derive below the CRB for the propagation delay, the carrier amplitude, and phase of the desired signal when the unknown covariance matrix $Q$ models both the thermal noise caused by the receiver and all other interference signals including MAI.

Consider the data model given by (12). The unknowns of the likelihood function for $\mathbf{r}(m)$ include the unknown elements of $Q$, the propagation delay $\tau_0$, the carrier amplitude $\gamma_0 \triangleq \sqrt{\tau_0}$, and phase $\theta_0$ of the desired signal. The extended Bangs’ formula for the $ij$th element of the Fisher information matrix has the form [16]

$$\{\text{FIM}\}_{ij} = \text{tr}(Q^{-1}Q^{-1}Q^{-1}Q^{-1}Q^{-1}(Dz)^HJ_z) + 2\text{Re}[\{z^H D^H\}J_z (Dz)]$$

where

$$X_i^\prime = \text{gradient of } X \text{ with respect to the } i \text{th unknown of the likelihood function},$$

$$\text{tr}(X) = \text{trace of } X,$$

$$\text{Re}(X) = \text{real part of } X.$$ Note that FIM is block diagonal since $Q$ does not depend on the parameters in $(Dz)$, and $(Dz)$ does not depend on the elements of $Q$.

If we only consider the parameter vector

$$\eta = [\gamma_0 \ \theta_0 \ \tau_0]^T$$

then the corresponding submatrix of the Fisher information matrix can be written as

$$\text{FIM}_\eta = \begin{bmatrix} F_{\gamma_0\gamma_0} & 0 & F_{\gamma_0\tau_0} \\
0 & F_{\theta_0\theta_0} & 0 \\
F_{\gamma_0\tau_0} & 0 & F_{\tau_0\tau_0} \end{bmatrix}$$

where

$$F_{\gamma_0\gamma_0} = 2 \sum_{m=1}^{M} \{z^H(m)A^HQ^{-1}Az(m)\}$$

(42)