Traffic-Aided Multiuser Detection for Random-Access CDMA Networks

Biao Chen, *Member, IEEE*, and Lang Tong, *Member, IEEE*

***Abstract*—Traffic burstiness has been observed for multimedia packet switching networks. This burstiness results in the predictability of user activity at the individual source level, and the exploitation of such predictability in the receiver design for a random-access CDMA packet switching network is investigated in this paper. Decorrelating detector is considered for performance and complexity tradeoff and it is shown that the conventional approach of assuming all users are active results in substantial performance loss. A two-stage multiuser detection is proposed where the first stage tracks active users and feeds its tracking output to the second stage multiuser detector for symbol detection. It is demonstrated that by using the traffic predictability, accurate estimate of active set of users is possible with a simple matched filterbank at the first stage.**

I. Motivation and Introduction

In packet-switching networks, it has been observed that the conventional Poisson assumption is not adequate to capture the traffic variability and dependence (see, e.g., [5]). In particular, traffic burstiness—the fact that packets usually come in bursts—has been revealed and carefully studied in the past decades. From the macroscopic point of view, traffic burstiness results in unpredictability of the overall network traffic [11]. However, from a microscopic point of view, traffic burstiness implies highly correlated individual user traffic, which results in a more predictable individual user activity; transmission of a packet at the current slot is usually followed by the transmission of packets at the next slots due to the bursty nature. This is in a sharp contrast to the conventional assumption of Bernoulli traffic that is hard to predict due to the independence assumption among different slots. This predictability of the bursty traffic has actually been utilized at the network level to assist in traffic routing [15]. For broadband wireless networks, such as wireless LAN and wireless ATM, the predictability of individual user's activity can also provide valuable information in physical layer designs because of the sharing of a common channel among a number of users.

The system we consider is a random-access packet switching/code division multiple access (CDMA) network. Because of its inherent signal diversity, the direct-sequence CDMA signal has the advantage of combating multipath fading for wireless communications. Moreover, the fact that CDMA systems allow simultaneous transmission of more than one users, which reduces the coordination at the base station, is appealing in data communication systems in view of the bursty and highly variable traffic from a potentially large number of users [1]. For CDMA systems using direct sequence spread spectrum techniques, multiple access interference (MAI) is one of the major performance limiting factors due to the nonorthogonality of spreading vectors among different users. Simple matched filter treats MAI as Gaussian noise, hence, can suffer significant performance loss, especially in the presence of near–far effect. Multiuser detection (MUD), on the other hand, attempts to utilize the structure of interference to mitigate the adverse effect of MAI [21]. Analysis shows that the use of multiuser detection for random-access CDMA networks has significant performance gain over the conventional matched filter receiver [17].

In studying the implementation of MUD for random-access CDMA networks, much research effort is usually concentrated on how to reduce the complexity of the optimal MUD given the potential large number of users [12]. One of the important issues in the implementation of MUD for random-access networks is that the active transmitting users at any particular time slot may not be known to the receiver. For conventional matched filter based receivers, this lack of active user information does not have any effect on the receiver performance. For multiuser detection receivers, however, mitigation of multiple access interference requires the knowledge of interfering users. Presumably, an MUD may be constructed assuming that all users are active. Such viewpoint, however, will incur significant performance loss, which was first observed by Mitra and Poor [13]. Given the performance loss of assuming a "full model" involving all potential users, it is therefore desirable to first infer from the received signal the set of active users that contribute to the current transmission, followed by an MUD that uses only the signature waveforms corresponding to the set of active users.

The problem of detecting active users has been addressed by several authors in a dynamic CDMA system [7], [13], [14], [23]. The treatment in [7], [13], and [14] differentiates the problem of detecting a new user entering the system from that of an old user ceasing communication. Their approaches include two steps: First, those old users are examined for possible drop out. Then, the continuing users' contribution is zeroed out so that the residual signal only contains contribution of one possible new user plus noise. In [13], the projection method is used to cancel out known users' contribution, and the problem of detecting a new user is posed as a multiple hypotheses testing problem,

Manuscript received October 11, 1999; revised March 23, 2001. This work was supported in part by the Multidisciplinary University Research Initiative (MURI) under the Office of Naval Research Contract N00014-00-1-0564. The associate editor coordinating the review of this paper and approving it for publication was Prof. Michail K. Tsatsanis.

B. Chen is with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY 13244 USA (e-mail:bichen@ecs.syr.edu).

L. Tong is with the School of Electrical Engineering, Cornell University, Ithaca, NY 14853 USA (e-mail:ltong@ee.cornell.edu).

Publisher Item Identifier S 1053-587X(01)05195-9.

with each hypothesis corresponding to each potential new user. In [14], equi-correlation spreading vectors are assumed for all different users, and a detection scheme is developed with the assistance of an “idle” spreading vector, which has the same cross-correlation with all other spreading vectors in use. In [7], a difference signal, which has been constructed by subtracting the estimated symbols of those known users, is used to detect the appearance of a possible new user. Further, cyclostationarity of the difference signal in the presence of a new user is utilized to achieve low complexity new user detection. In [23], the detection (identification) of active set of users is dealt with where the authors adopt the MUSIC algorithm for array processing, which requires an accurate estimate of correlation matrix to carry out the eigendecomposition.

A common feature of above approaches is that the detection is focused on a particular data window without accounting for the continuity of the channel traffic. In this paper, by taking advantage of the traffic burstiness for packet networks, we study the possibility of utilizing traffic predictability to improve the detection of active users. First, the problem is shown to be equivalent to model selection in a linear regression problem. A set of indicator functions is incorporated in the model to accommodate all possible set of active user. This treatment also enables us to fully utilize the traffic predictability at source level. By characterizing the dependence of user activity using Markov chain, we are able to convert the user identification problem to state estimation of hidden Markov models (HMMs).

The organization of the paper is as follows. The signal model considered in this paper is given in the next section. In Section III, the effect of including a nonactive user is illustrated using the a decorrelating detector example. A two-stage MUD is proposed in Section IV, whose first stage aims to detect the set of active users. In Section V, we give some simulation results to show the potential improvement on the packet reception probability by using the traffic information. Conclusions are given in Section VI.

II. Signal Model

The system we study is in line of that considered by Raychaudury [18]. We consider the uplink of a slotted packet switching networks with direct sequence/spread spectrum (DS/SS) signaling using transmitter codes, i.e., each user transmits using a unique spreading code. Newly arrived packets are encoded with the user’s own code and transmitted immediately on arrival. Retransmission is dealt with in a similar fashion. For simplicity, the channels are assumed known to the receiver (e.g., can be estimated via training sequence), and therefore, the effective signature vectors of all the users are assumed known to the receiver. For simplicity, the slotted system with synchronous transmission is considered. Assuming BPSK modulation, the received signal within a slot can be written as

$$\mathbf{y}_n(t) = \sum_{k \in \mathcal{I}_n} A^{(k)} b_n^{(k)}(t) \mathbf{c}^{(k)} + \mathbf{z}_n(t), \quad t = 1, 2, \ldots T \quad (1)$$

where

- n packet index;
- t symbol index within a packet;
- T packet size;
- k user index;
- $A^{(k)}$ amplitude of user k;
- $\mathbf{c}^{(k)}$ signature waveform of kth user and is assumed to have unit norm, i.e., $\|\mathbf{c}^{(k)}\| = 1$;
- $b_n^{(k)}(t)$ tth symbol within the nth packet for user k;
- $\mathbf{z}_n(t)$ additive white Gaussian noise with zero mean and covariance matrix $\sigma^2\mathbf{I}$.
- $\mathcal{I}_n$ index set of the active users at slot n.

Thus, $\mathcal{I}_n \subset \{1, 2, \ldots, M\}$. The receiver’s task is to detect the symbols $b_n^{(k)}$ for all $k \in \mathcal{I}_n$ without prior knowledge of $\mathcal{I}_n$.

III. Effect of Overmodeling

A linear decorrelating detector can be viewed as a least square (LS) estimator followed by a sign detector. The MAI is totally compensated via premultiplying the matched filter output by the inverse of correlation matrix of the signature waveform at the cost of increased noise variance. Thus, if a decorrelator intends to mitigate the interference from those users that are not actually transmitting, its lone effect is to enhance noise, therefore degrading the receiver performance. This effect was documented in [13] and later studied in detail in [23]. In particular, in [23], it was shown that the covariance matrix of the LS estimator using only the active users is always smaller than the covariance matrix of LS estimator if inactive users are also included, which is termed as the conventional decorrelating detector. This is obtained using the result for the inverse of positive-definite block matrix (the correlation matrix of the spreading vectors). Along a similar line, we obtain a result in this section that quantitatively characterizes the performance difference of the two different approaches. The result also leads to an easy geometrical interpretation that is consistent with the intuitive augment of the effect of the correlation on the receiver performance.

A. Increased Covariance for the Overmodeling LS Solution

Suppressing the packet and time indices for clarity, we rewrite (1) in a matrix form as

$$\mathbf{y} = \sum_{k \in \mathcal{I}} s^{(k)} \mathbf{c}^{(k)} + \mathbf{z} = \mathbf{CAb} + \mathbf{z} = \mathbf{Cs} + \mathbf{z} \quad (2)$$

where $\mathbf{s} = \mathbf{Ab}$, and $\mathcal{I} = \{k_1, \ldots, k_m\}$ is the index set of active users. Given $\mathcal{I}$, this is a standard linear regression problem where the regressors are the signature waveforms, and the response variable is the observation $\mathbf{y}$. The optimal solution in the maximum likelihood sense is the least squares solution, which is a property dictated by the independent Gaussian assumption of the noise.

By using nonactive users’ channels, the decorrelating receiver aims to fit the data $\mathbf{y}$ with the following model:

$$\mathbf{y} = [\mathbf{C} \vdots \mathbf{D}] \begin{bmatrix} \mathbf{s} \\ \mathbf{r} \end{bmatrix} + \mathbf{z} \quad (3)$$

with $\mathbf{D}$ representing the spreading vectors of those nonactive users, and $\mathbf{r}$ being the “pseudo” message of these users. We refer to this as the “full model” as opposed to the true model of (2) when only the active users are involved. Comparing the

LS estimate for $\mathbf{s}$ using models (2) and (3), we obtained the following Lemma.

Lemma 1: Consider the least square estimate of $\mathbf{s}$ of the two models. The error covariance of the full model satisfies

$$\text{cov}(\bar{\mathbf{s}}) = [\mathbf{C}'\mathbf{C} - \hat{\mathbf{C}}'\hat{\mathbf{C}}]^{-1}\sigma^2 \geq \text{cov}(\hat{\mathbf{s}}) \tag{4}$$

where $\hat{\mathbf{C}}$ is the projection of $\mathbf{C}$ onto the space spanned by $\mathbf{D}$, i.e.,

$$\hat{\mathbf{C}} = \mathbf{D}(\mathbf{D}'\mathbf{D})^{-1}\mathbf{D}'\mathbf{C}. \tag{5}$$

The equality holds iff the signal space spanned by active users is orthogonal to the signal space spanned by inactive users, i.e., $\mathbf{C}'\mathbf{D} = \mathbf{0}$

A proof is sketched in Appendix A. Note that the covariance matrix of (4) can be rewritten as

$$\text{cov}(\hat{\mathbf{s}}) = [\mathbf{C}'(\mathbf{I} - \mathbf{D}(\mathbf{D}'\mathbf{D})^{-1}\mathbf{D})\mathbf{C}]^{-1}\sigma^2$$

i.e., it is the inverse of the projection error matrix from $\mathbf{C}$ onto the space spanned by the columns of $\mathbf{D}$. This allows a very simple geometrical explanation of the overmodeling effect; the increase of the error covariance using a full model is propotional to the correlation between the signature vectors of active users and inactive users. For example, in the extreme case of orthogonal spreading vectors among all users, there is no performance loss by using a full model.

B. Effects on Packet Success Probability

The packet success probability, which is defined as the average portion of successfully received packets among all packets transmitted, is an alternative figure of merit to the network throughput. To compute packet success probability, we first need to compute the bit error rate (BER) for the decorrelating detector. Note that the BER not only is a function of channel noise variance but also depends on the number of users assumed in the decorrelating detector, which is denoted by N. To allow exact expression of BER in terms of N, we adopt the model proposed in [14], where different pairs of users' spreading vectors are assumed to have equal correlation coeficient ρ. Given this assumption and that the number of users involved in a decorrelating detector is N, the diagonal element of the inverse of the correlation matrix of spreading codes is, as in [14]

$$a(N) = \frac{1 + (N-2)\rho}{1 + (N-2)\rho - (N-1)\rho^2}. \tag{6}$$

The resulting BER is therefore

$$P_e(N) = Q\left(\frac{A}{\sigma\sqrt{a(N)}}\right)$$

where A is the received signal's amplitude, and σ^2 is the channel noise variance. Assume that the rror correction code can correct up to b bit errors out of a total of T bits (packet length); then, the packet success probability given L users are active, and the decorrelator with $N(\geq L)$ users is

$$P_N(\text{success}|L) = \sum_{j=0}^{b} \binom{T}{j} P_e(N)^j (1 - P_e(N))^{T-j}.$$

Assuming a homogeneous channel traffic where each user at any slot transmitting a packet with probability q, we can obtain the the overall packet success probability by averaging over L, which is *Binomial* (N, q). The result is plotted in Fig. 1, where the packet error rate is plotted against the channel traffic load q. Clearly, the gain in terms of packet error rate using true user information is of several orders of magnitude over the full model, whose performance is independent of the traffic load. The packet error rate using a matched filter bank for all users is also plotted as a dashed line. Clearly, as channel load approaches zero, the matched filter receiver should be arbitrarily close to the decorrelating detector using the true model, as the probability of having more than two active users vanishes.

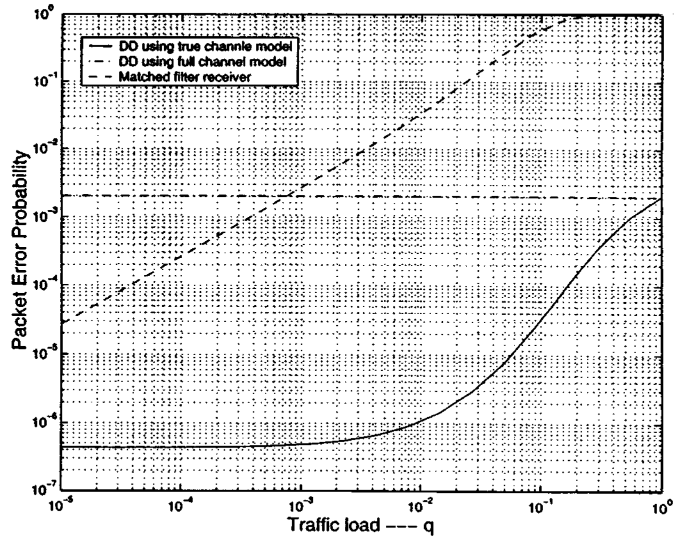

Fig. 1. Packet error probability as a function of traffic load parameter q. While the performance of the decorrelating assuming all users are active is essentially a constant (dash-dotted line), the decorrelator using the true model (solid line) performs differently according to different traffic load. The performance of conventional matched filter-based receiver is also plotted against the traffic load parameter. The total number of potential users is 30, and the correlation coefficient is assumed to be $\rho = 0.2$ for each pair of signature vectors. Packet length is 8 bits with error correction capability of up to $b = 8$ bit errors.

IV. Traffic-Aided MUD

A. Traffic Modeling for Packet Switching Networks

Given the performance degradation of an MUD when inactive users' signature waveforms are erroneously included in the MUD, it is therefore desirable to first estimate the set of active users and then form an MUD using only those signature waveforms corresponding to the active users. This results in a two-stage receiver as in Fig. 2. In the first stage, the set of active users is detected, while in the second stage, an MUD is implemented by adapting to the change of user profile.

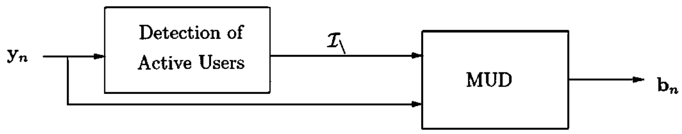

Fig. 2. Two-stage multiuser detection.

The implementation of an MUD at the second stage is straightforward, given the set of active users' signature waveforms. It is worth mentioning that fast algorithms for adapting the decorrelating detector to the change of active users exist in the literature [8], [9]. Our focus will be on the first stage, i.e., the detection of the active set of users. To this end, we first introduce the following model as an alternative to (2):

$$\mathbf{y}_n(t) = \sum_{k=1}^{M} \mathbf{c}^{(k)} s_n^{(k)}(t) \gamma_n^{(k)} + \mathbf{z}_n(t) \tag{7}$$

where, in the nth packet slot, $\gamma_n^{(k)}$ is the indicator function that takes value 1 if the kth user is active and 0 otherwise. Incorporating the indicator functions in the signal model converts the user identification problem into an estimation problem for the binary random sequence $\gamma_n^{(k)}$. More important, it also provides a natural framework to include the traffic information that is interpreted as the probability law that dictates the binary process of $\gamma_n^{(k)}$.

As mentioned before, traffic burstiness implies that for each individual user, its transmission occurs in clusters. Reflected in the indicator function, sequences of zeros (idle) and ones (active) occur alternately. A first-order approximation of the bursty traffic for an individual source is a two-state Markov chain, which was originally proposed by Viterbi for the performance analysis of packet switching networks [22]. Specifically, $\gamma_n^{(k)}$ is assumed to be a two-state Markov chain with state transition matrix

$$a^{(k)} = \begin{bmatrix} 1-p^{(k)} & p^{(k)} \\ q^{(k)} & 1-q^{(k)} \end{bmatrix}$$

where $p^{(k)} = P\left(\gamma_n^{(k)} = 1/\gamma_{n-1}^{(k)} = 0\right)$, and $q^{(k)} = P\left(\gamma_n^{(k)} = 0/\gamma_{n-1}^{(k)} = 1\right)$. As a digression, this two-state Markov chain is equivalent to stating that the lengths of consecutive zeros and ones of the binary sequence are independently and geometrically distributed, which is an assumption used in [6] to obtain explicit expression of protocol information in packet networks. We will hereafter adopt this Markov model in the user tracking problem both because of its analytical tractability and the existence of rich inference tools associated with the problem.

B. *Optimal HMM Traffic Tracking*

Combining all the indicator functions $\gamma_n^{(k)}$, we can define $\Gamma_n = \left(\gamma_n^{(1)}, \ldots, \gamma_n^{(M)}\right)$ as the state variable of active user profile, i.e., those coordinates that equal to "1" in Γ_n indicate that the corresponding users are transmitting at slot n. Note the state space has a cardinality that grows exponentially with the total number of users. Given that each individual indicator function is a two-state Markov chain, the combined state variable Γ_n is now also a Markov sequence with state dimensionality 2^M, where M is the total number of potential users. Its transition matrix $\mathbf{A}$ is simply the Kronecker product of each individual transition matrix for each $\gamma_n^{(k)}$. In turn, our observation is now an HMM.

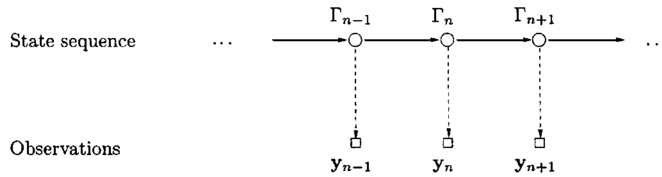

Fig. 3. Hidden markov modeling of traffic.

$$\mathbf{y}_n \triangleq \begin{bmatrix} \mathbf{y}_n(1) \\ \vdots \\ \mathbf{y}_n(T) \end{bmatrix} = \sum_{k \in \mathcal{I}(\Gamma_n)} \begin{bmatrix} \mathbf{c}^{(k)} s_n^{(k)}(1) \\ \vdots \\ \mathbf{c}^{(k)} s_n^{(k)}(T) \end{bmatrix} + \begin{bmatrix} \mathbf{z}_n(1) \\ \vdots \\ \mathbf{z}_n(T) \end{bmatrix} \tag{8}$$

where T is the packet size. Specifically, this relationship is shown in Fig. 3, where the underlying state Γ_n evolves according to a Markov transition matrix, whereas the actual observation $\mathbf{y}_n$ depends (probabilistically) only on the current state Γ_n. With this formulation, the user identification problem is now converted to a state estimation problem for an HMM. One of the attractive features of HMMs is the existence of efficient and often recursive estimation algorithms. An excellent tutorial about HMMs can be found in [16]. Notice, however, that the optimal HMM formulation is based on the state combining, which results in a state dimensionality of 2^M, i.e., it is exponential in the number of potential users. As seen in (8), each different state corresponds to a different regression model, and hence, the overall computational complexity of the optimal HMM tracking is usually formidable.

C. *Suboptimal Traffic Tracking*

Because of the prohibitive complexity of the optimal HMM traffic tracking, a suboptimal approach whose complexity grows linearly in the number of users is therefore highly desirable. We note that the complexity of the optimal HMM tracking is due to the state combining, which leads to the exponential explosion of the state dimensionality. The key to complexity reduction is therefore the "decoupling" of individual user's state tracking. Notice that such decoupling inevitably leads to suboptimality as the "interaction" between different user's signature vectors in the regression model is not fully accounted. However, it should be realized that suboptimality does not necessarily imply poor performance. This is because the SNR requirement for the detection of the *presence* of one particular user is much less demanding than the required SNR for the correct symbol detection. To further elaborate this, consider the model described by (7). While $s_n^{(k)}(t)$ is different from bit to bit (i.e., for different t), the indicator function $\gamma^{(k)}$ remains the same throughout the whole packet. User identification amounts to the estimation of $\gamma_n^{(k)}$ and the fact that $\gamma_n^{(k)}$ remaining constant for all symbols provides an important diversity for its inference. Given this diversity and the fact that traffic predictability also helps to improve the estimate of $\gamma_n^{(k)}$, it is therefore possible to achieve reasonably good performance, even with a simple front end user detector.

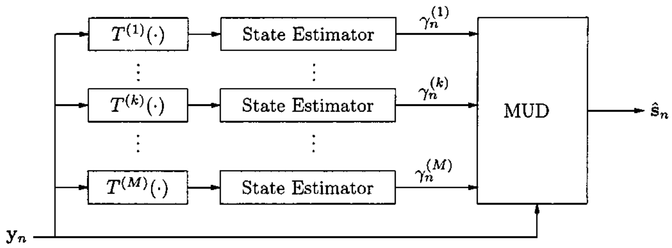

Fig. 4. Suboptimal two-stage MUD implementation.

1) Receiver Structure: Fig. 4 illustrates a suboptimal traffic tracking algorithm. Instead of expanding all users' states into a single state space, we track individual user activity separately. The receiver consists of two stages: the state tracking and the multiuser detection. The state tracking is achieved independently for each user, hence avoiding the prohibitive complexity due to the state expansion. Specifically, the received signal is first mapped to a new statistic that serves as the "likelihood"[1] of a particular user being active. Then, the status of the user in the previous slot is used to jointly decide whether the user is transmitting at the current slot. In other words, instead of a combined HMM where the underlying Markov state has the dimensionality of 2^M, we have now an M individual two-state HMM tracker, and hence, the complexity is now linear in M.

Depending on the summary statistic $T^{(k)}(\cdot)$, various structures of user tracker can be formed. One of the desirable features of the statistic $T^{(k)}(\cdot)$ is that it should reflect only the likelihood of the kth user being transmitting; it should be interference free. One obvious choice is to use the projection method, which follows the line of [13] in detecting the possible dropout of an old user. Specifically, the interference from other users can be eliminated by projecting the received signal onto the space orthogonal to the space spanned by interfering users spreading vectors [3]. This, in fact, amounts to a decorrelating detector for the particular user of interest. However, since the detector has no explicit knowledge of the active users, the only reasonable choice is to decorrelate all potential interfering users. This puts an undesirable clamp on the number of users to be accommodated in the network; to guarantee such projection operator to work appropriately, the number of potential users should be smaller than the spreading gain. Note that in packet switching network, the traffic burstiness and variability implies that at any particular time, the actual number of active users may be drastically smaller than the total number of potential users in the system. Therefore, even though the correlation matrix of the active users' spreading vectors is often well conditioned, it is likely that the overall correlation matrix of all potential users' spreading vectors may be singular.

2) Matched Filter Front End: We propose, in this paper, a simple matched filter front end for user detection. The drawback, obviously, is that the matched filter user detector is still subject to multiple access interference. The advantage is that it does not impose any restriction on the rank of the correlation matrix of all spreading vectors. The rational, again, is that the user identification is relatively easier to carry out compared with symbol detection due to the diversity induced by the fact that all symbols within a packet are modulated by the same set of spreading vectors. More importantly, the use of the traffic predictability can compensate largely the insufficiency of the matched filter front end. It is worth emphasizing that the receiver structure is now a matched filter-based user detector plus a second-stage multiuser detector, as opposed to a simple matched filter receiver for symbol detection. We will show in our simulation that this two-stage receiver significantly outperforms the simple matched filter receiver, whose performance is severely limited by the MAI, especially in the high SNR region.

Consider a matched filter output for the kth user with input signal described in (7)

$$\begin{aligned} w^{(k)}(t) =& s_n^{(k)}(t)\gamma_n^{(k)} + \sum_{i\in\mathcal{I}_n, i\neq k} \rho_{ki} s_n^{(i)}(t) + z^{(k)}(t) \\ t =& 1, 2, \ldots, T \end{aligned}$$

where ρ_{ki} is the correlation coeficient between user k and is spreading vectors, and $z^{(k)}(t)$ is zero mean Gaussian random variable whose variance remains σ^2 due to the unit norm assumption of the spreading vector. Our purpose at this stage is, given $w^{(k)}(t)$, $t = 1, 2, \ldots, T$, to infer about the value of the indicator function $\gamma_n^{(k)}$. Apparently, Gaussian approximation of the second term in the matched filter output is necessary to make the approach tractable. We therefore define

$$\begin{aligned} \hat{z}^{(k)}(t) &= \sum_{i\in\mathcal{I}_n, i\neq k} \rho_{ki} s_n^{(i)} + z^{(k)}(t) \\ &= \sum_{i\in\mathcal{I}_n, i\neq k} \rho_{ki} A_n^{(i)} b_n^{(i)}(t) + z^{(k)}(t). \end{aligned}$$

Invoking the Gaussian approximation, the variance of $\hat{z}^{(k)}(t)$ should be

$$\sigma_k^2 = \sigma^2 + \sum_{i\in\mathcal{I}_n, i\neq k} \rho_{ki}^2 {A_n^{(i)}}^2.$$

Because of the unknown active user set $\mathcal{I}_n$, the above variance approximation cannot be directly computed in practice. However, one can get around this problem by resorting to the Markov model of the user activity with the assistance of the state estimate from the previous slot. Specifically, we propose the so-called predicted variance

$$\begin{aligned} \hat{\sigma}_k^2 = \sigma^2 + & \sum_{\substack{\gamma_{n-1}^{(j)}=1 \\ j\neq k}} (1 - q^{(j)}) \rho_{kj}^2 {A^{(j)}}^2 \\ & + \sum_{\substack{\gamma_{n-1}^{(i)}=0 \\ i\neq k}} p^{(j)} \rho_{kj}^2 {A^{(i)}}^2 \end{aligned} \tag{9}$$

where

- σ^2 channel noise variance;
- $q^{(j)}$ state transition probability of $\gamma^{(j)}$ from state 1 to state 0;
- $p^{(i)}$ state transition probability of $\gamma^{(i)}$ from state 0 to 1, as defined in Section IV-A.

[1]Likelihood is an abuse of the term since without jointly accounting for all possible set of signature vectors, the obtained "likelihood" is not strict in its original sense.

This predicted variance is in essence similar to the state prediction covariance in the Kalman filter update [2], except that the probabilistic state transition is used in place of the dynamic model assumed in the Kalman filter.

Under the above Gaussian approximation, one can combine the observations within a packet in a vector form, which leads to the following hypotheses test regarding $\gamma_n^{(k)}$:

$$\begin{array}{ll}\mathbf{H}: & \mathbf{w}^{(k)} = \hat{\mathbf{z}}^{(k)} \\ \mathbf{K}: & \mathbf{w}^{(k)} = \mathbf{s}_n^{(k)} + \hat{\mathbf{z}}^{(k)} = A^{(k)}\mathbf{b}^{(k)} + \hat{\mathbf{z}}^{(k)}\end{array} \tag{10}$$

where $\mathbf{b}^{(k)}$ is the symbol vector of the packet for user k (if it is actually transmitting), and $\hat{\mathbf{z}}_i$ is Gaussian random vector with zero mean and covariance matrix $\hat{\sigma}_k^2 I$. Note that the model implicitly assumes that the received amplitude $A^{(k)}$ is constant for all the symbols within a packet, which is a reasonable assumption in a slow fading environment with short packet length. Further, under this assumption, $A^{(k)}$ can be trivially estimated using $(1/T)\sum_{n=1}^{T}|w_n^{(k)}|$.

To deal with the unknown symbol vector $\mathbf{b}_n^{(k)}$, several different approaches can be adopted. A generalized likelihood ratio (GLR) approach simply estimates the symbol first and then plug them into the likelihood ratio. The summary statistic, therefore, is in the form of GLR between the two hypotheses. The problem is that the symbol estimate at this stage is prone to error; hence, the obtained GLR can be misleading. The second approach [the optimal ML (or Bayesian) approach] marginalizes out $\mathbf{b}^{(k)}$, assuming they are equally probable among all possible value. The marginalization, however, is with respect to a discrete probability measure spread over 2^T grid points; hence, its complexity is prohibitive. A third approach, whose resulting statistic is the simplest among the three methods, is to adopt the invariance principle [10] to develop a uniformly most powerful invariant (UMPI) test statistic.

3) UMPI Statistic: For composite hypotheses testing such as the one we are dealing with, a uniformly most powerful test often does not exist. It is possible, however, to take advantage of the particular structure of the hypotheses testing problem, e.g., symmetries with respect to the parameters, to restrict the test to a meaningful subset of all possible tests; hence, the most powerful test among the subset may exist. When the invariance principle is applied to reduce the test to the set of invariant test, the resulting optimal test is the UMPI test. The first step to apply the invariance principle is to identify if the test problem is invariant under a group of transformations. Once the invariant transformation group is obtained, the next step is to find a maximal invariant (MI) statistic, which is so named because every other invariant statistic must be a function of the MI statistic. The UMPI test should therefore based only on the MI statistic as such tests comprise the totality of invariant tests. For a detailed description of invariance principles and UMPI test, see [4] and [10].

Specific to our problem, given that the symbol vector $\mathbf{s}_n^{(k)}$ is unknown at this stage, the testing may be abstracted as testing the mean of a Gaussian vector $\mathbf{w}^{(k)} \sim \mathcal{N}\left(\mathbf{m}, \sigma_k^2\mathbf{I}\right)$:

$$\begin{array}{ll}\mathbf{H}: & \mathbf{m} = \mathbf{0} \\ \mathbf{K}: & \mathbf{m} \neq \mathbf{0}\end{array}. \tag{11}$$

Note that the generality of the test problem allows the relaxation of the assumption that the received amplitude being constant throughout the packet. The test problem is easily seen to be invariant under any orthogonal transformation of the observation vector, i.e., multiplying the observation with any orthonormal matrix will not alter the inference problem and its associated testing structure. The MI under such a transformation can be obtained as [4]

$$S_{\mathrm{MI}} = \mathbf{w}^{(k)\prime}\mathbf{w}^{(k)}.$$

Clearly, S_{MI} remains constant when premultiplying $\mathbf{w}^{(k)}$ with any orthonormal matrix. It turns out that such MI is either chi-square or noncentral chi-square distributed, depending on which hypothesis is true. Specifically, $S_{\mathrm{MI}} \sim \hat{\sigma}_k^2\chi_T^2$ under $\mathbf{H}$, whereas $S_{\mathrm{MI}} \sim \hat{\sigma}_k^2\chi^2_{T,A^{(k)2}T/2\sigma_k^2}$ under $\mathbf{K}$, where the second subscript of chi-square, if applicable, is the noncentrality parameter. Note the difference between the two hypotheses is the noncentrality of the summary statistic. For the testing of noncentrality of a chi-squared distributed random variable, it is easy to show that it has the monotone likelihood ratio [10] in the observation itself, and hence, a UMPI test statistic is simply the maximum invariant itself, i.e., $T^{(k)}(\mathbf{y}_n) = \mathbf{w}^{(k)\prime}\mathbf{w}^{(k)}$.

4) User Identification Algorithm: Once we obtain the UMPI statistic, we can now incorporate the traffic predictability to infer about $\gamma_n^{(k)}$. Given that $\gamma_n^{(k)}$ is a two-state Markov chain, $T^{(k)}(\mathbf{y}_n)$ is now an HMM whose distribution is central or noncentral chi-square, depending on the value of $\gamma_n^{(k)}$. We can therefore form an HMM state tracker to dynamically estimate $\gamma_n^{(k)}$ using $T^{(k)}(\mathbf{y}_n)$ and the state estimate from the previous slot. In the absence of traffic information, a one-shot UMPI test can be used, which amounts to the simple thresholding of the UMPI statistic S_{MI}. It is therefore of particular interest to study the performance improvement by incorporating the traffic predictability.

To summarize, the activity detection algorithm for the kth user is described in the following.

1) Pass the observation vectors through matched filter $\mathbf{c}^{(k)}$

$$w^{(k)}(t) = \mathbf{c}^{(k)\prime}\mathbf{y}_n(t), \quad t = 1, \ldots, T.$$

2) Compute the MI statistic

$$S_{\mathrm{MI}} = \mathbf{w}^{(k)\prime}\mathbf{w}^{(k)} = \sum_{t=1}^{T}|w^{(k)}(t)|^2.$$

3) Estimate $\gamma_n^{(k)}$ by

$$\frac{P(\gamma_n^{(k)} = 1|T^{(k)}(\mathbf{y}_n)\cdots T^{(k)}(\mathbf{y}_1))}{P(\gamma_n^{(k)} = 0|T^{(k)}(\mathbf{y}_n)\cdots T^{(k)}(\mathbf{y}_1))} \overset{\mathbf{H}}{\underset{\mathbf{K}}{\lesseqgtr}} \tau \tag{12}$$

where $\mathbf{K}$ and $\mathbf{H}$ represent the hypotheses that the kth user is active or not. The above posterior probability is obtained via the forward variables of the two-state HMM S_{MI}. Details about the update the posterior probability is briefly summarized in Appendix B.

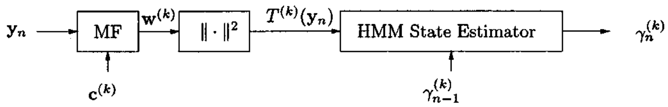

Fig. 5. Suboptimal state tracker using MF front end.

In the event that no traffic modeling information is available, the last step is replaced with a simple thresholding of the MI statistic to detect the active user, i.e., for the kth user, we have

$$T^{(k)}(\mathbf{y}_n) \underset{\mathbf{K}}{\overset{\mathbf{H}}{\lesseqgtr}} t. \tag{13}$$

The block diagram of a individual user tracker with matched filter front end is illustrated in Fig. 5. For the one-shot approach, the HMM state estimator is replaced with a thresholding device.

We should mention that for the HMM state estimate, the minimum error probability state estimate corresponds to setting $\tau = 1$. That is, the minimum error state estimator always uses the state estimate with the maximum posterior probability. However, in the current context where the HMM state estimate serves as the front end for a multiuser detector, minimum error probability state estimate does not necessarily translate into the best packet reception performance. The reason is that the minimum error probability criterion equally penalizes the two types of error: a miss detection, which is referred to an active user being declared inactive, and a false alarm, when an inactive user is erroneously claimed active. In the following, we show that the effect of a false alarm and a miss detection is different on the effective SNR at the packet receiver when a decorrelating detector is used.

5) Approximate Analysis of the Effect of a Miss Detection and a False Alarm on the Effective SNR: A miss detection will incur a sure packet loss itself. In addition, the interference of the user undetected in the first stage will impair the reception of other user at the MUD. Here, we quantify the effect of a false alarm and miss detection on the effective SNR when a decorrelating detector is used under some simple assumptions. Suppose the actual user number is N and the correlation coeficient is ρ, which is equal for all pairs of users, as we have seen in Section III, and the channel noise variance is σ^2. Again, for simplicity, we consider equal power case with A being the received amplitude of all users. We will also use Gaussian approximation for the interference incurred by the undetected user.

Assuming correct user detection, we have seen that the effective SNR at the receiver is $A^2/\sigma^2 a(N)$ with $a(N)$ as in (6). In the case of a false alarm, $N+1$ users are declared active. Thus, the SNR at receiver is now $A^2/\sigma^2 a(N+1)$. The equivalent noise variance is now

$$\sigma_{fa}^2 = \sigma^2 a(N+1). \tag{14}$$

Next, we quantify the equivalent noise variance when a miss detection happens. Assume for the N active users that $N-1$ is correctly detected, whereas the Nth user is missed; then, the decorrelating detector output, assuming only $N-1$ users are active, is now

$$\mathbf{r} = \left(\mathbf{C}_{N-1}'\mathbf{C}_{N-1}\right)^{-1}\mathbf{C}_{N-1}'\mathbf{y}$$

where without confusion, $\mathbf{C}_{N-1}$ represent the spreading vector matrix of the $N-1$ users correctly detected, and $\mathbf{y}$ is the received signal that can be conveniently written as

$$\mathbf{y} = \mathbf{C}_{N-1}\mathbf{A}\mathbf{b} + \mathbf{c}_N A b_N + \mathbf{n}.$$

Plugging it into the decorrelating output, we have

$$\begin{aligned}\mathbf{r} = \mathbf{A}\mathbf{b} &+ \left(\mathbf{C}_{N-1}'\mathbf{C}_{N-1}\right)^{-1}\mathbf{C}_{N-1}'\mathbf{c}_N A b_N \\ &+ \left(\mathbf{C}_{N-1}'\mathbf{C}_{N-1}\right)^{-1}\mathbf{C}_{N-1}'\mathbf{n}.\end{aligned} \tag{15}$$

The noise variance of the last term is easily seen to be $\sigma^2 a(N-1)$. To derive the equivalent variance of interference term using Gaussian approximation, we invoke the equi-correlation assumption and the fact that $(\mathbf{C}(N)'\mathbf{C}(N))^{-1}$ is a T-matrix[2] with an off-diagonal element being

$$b(N) = -\frac{\rho}{1+(N-2)\rho-(N-1)\rho^2}.$$

We can then expand the middle term on the right side of (15)

$$\begin{aligned}&\left(\mathbf{C}_{N-1}'\mathbf{C}_{N-1}\right)^{-1}\mathbf{C}_{N-1}'\mathbf{c}_N A b_N \\ &\quad= \left(\mathbf{C}_{N-1}'\mathbf{C}_{N-1}\right)^{-1}\mathbf{1}_{N-1}\rho A b_N \\ &\quad= (a(N-1)+(N-2)b(N-1))\rho A b_N \mathbf{1}_{N-1} \\ &\quad= \left(\frac{1}{1+(N-2)\rho}\right)\rho A b_N \mathbf{1}_{N-1}\end{aligned}$$

where $\mathbf{1}_{N-1}$ is $(N-1)\times 1$ vector with all 1s. Therefore, the total noise variance due to a miss detection is now

$$\sigma_{\mathrm{md}}^2 = \left(\frac{1}{1+(N-2)\rho}\right)^2 \rho^2 A^2 + \sigma^2 a(N-1). \tag{16}$$

From (14) and (16), it is easy to see that if

$$\frac{A^2}{\sigma^2} > \frac{2(1+(N-2)\rho)}{1-\rho} \tag{17}$$

then the noise variance with a miss detection is always greater than a false alarm. First, notice that (17) is always true if $\rho < -1/(N-2)$. Otherwise, if

$$N < \frac{(4-s)\rho + s - 2}{2\rho}$$

with $s = A^2/\sigma^2$, the equivalent variance of a miss detection is also greater than a false alarm. For example, with channel SNR being > 10 dB and $\rho \le 0.2$, then as long as the total number of active users is smaller than 7, the SNR loss due to a miss detection is always more severe than a false alarm. Therefore, under the usual channel condistion, the penalty incurred by a miss detection on the equivalent SNR is always more severe than a false alarm when a decorrelating detector is used. Note further that this SNR loss does not take into account the packet loss of the undetected user itself due to the miss detection. This justifies that a threshold other than 1 should be used in (12) to put more stringent control on the probability of miss detection. Since the ultimate performance measure is the packet error probability,

[2]A T-matrix is a square matrix with identical diagonal matrix and identical off-diagonal element.

ideally, we would want to quantify the effect of a miss detection or false alarm on packet error probability; hence, a cost matrix may be applied to determine the optimal threshold for the user detection schemes. The complexity of the problem, however, prohibits such analysis to be carried out, even under the above-simplified assumptions. Notice that the obtained variance of the above analysis is a nonlinear function of the number of active users in the system. Further, multiple false alarms and/or miss detections, whose effect on the noise variance is even more complicated, can happen together. These, coupled with the fact that the packet error probability is a nonlinear function of bit error rate, which is, in turn, a nonlinear function of the effective SNR, render the analysis impractical. Numerical simulations should be carried out to determine the "best" threshold for the packet detection performance.

V. Simulation

In this section, we present some numerical examples to demonstrate the advantage of utilizing the traffic information in the receiver design. The simulation setup is as follows. For each potential user, the state variable $\gamma_n^{(k)}$ is generated according to the following state transition matrix:

$$a^{(k)} = \begin{bmatrix} 0.99 & 0.01 \\ 0.10 & 0.90 \end{bmatrix}.$$

If $\gamma_n^{(k)} = 1$, the kth user will randomly generate a data packet with BPSK modulation at the nth slot. Further, we consider the nonpower-control case, but for simplicity, we assume that each user's amplitude is fixed and equally spaced in the log scale in the range of 20 dB. That is, assuming total number of users is N and the users are ordered according to their power, then $\log_{10} A_{i+1}/A_i$ remains constant, and $20\log_{10} A_N/A_1 = 20$ dB. The SNR shown in the plots is the SNR for the user with minimum power. Our emphasis will be on the suboptimal scheme, but we start with a simple example with the optimal HMM tracking front end.

A. Optimal HMM Tracking Approach

In this example, the total number of users is assumed to be 7. Packet length is 4 where the error correction code is able to correct up to 2 bit errors. The spreading vector length is 5. The spreading vectors for different users are obtained using different phases of an m-sequence with $m = 20$, i.e., the period is of the m-sequence is $2^m - 1$. Notice that the particular form of the spreading vectors are not as important, but their correlation matrix is more critical on the receiver performance. For this reason, we write down explicitly the correlation matrix of the spreading vectors used in our simulation, shown in the equation at the bottom of the page.

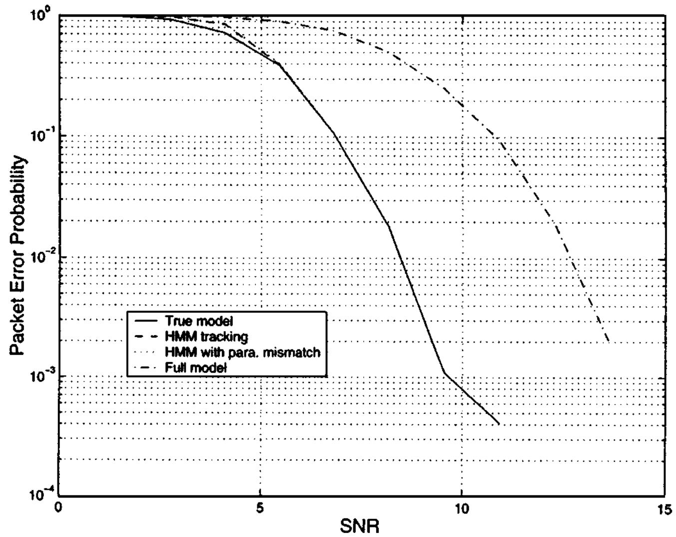

Fig. 6. Decorrelator using true model (solid), HMM tracker with true parameter (dashed), HMM tracker with mismatched parameter (dotted), and the full model (dash-dotted). The channel traffic is generated assuming the mean active time ten slots and mean idle time 100 slots. The total number of users is seven, and the spreading vector's length is 15.

From Fig. 6, the optimal HMM tracker performs essentially as well as that uses the true model, and both substantially outperform the receiver assuming a full model. Further, we have considered the case where the HMM tracker uses mismatched parameters—the actual state transition probabilities between the two states are 0.1 and 0.01, respectively, but the mismatched tracker assumes both equal to 0.05. It turns out that the performance with parameter mismatch is identical to that using the true parameter (the dotted and dashed line overlap in Fig. 6), which is an advantage thanks to the robust property of HMM inference with regard to the mismatch of state transition matrix [20]. In particular, it was found that the state estimate of an HMM is relatively insensitive to the deviation of the state transition matrix to its true value, as long as the overall "shape" of the transition matrix is preserved. That is, the state transitions that are more likely in the true state transition matrix are still the more likely transitions in the mismatched transition matrix.

B. Suboptimal Traffic Tracking

We investigate the performance of the suboptimal user detector with a matched filter front end. In particular, we are interested in the performance divergence between the detector using

$$R = \begin{bmatrix} 1.0000 & -0.2000 & -0.3333 & 0.6000 & 0.3333 & -0.3333 & 0.3333 \\ -0.2000 & 1.0000 & 0.0667 & -0.6000 & 0.2000 & -0.2000 & -0.0667 \\ -0.3333 & 0.0667 & 1.0000 & 0.0667 & 0.0667 & 0.4667 & -0.2000 \\ 0.6000 & -0.6000 & 0.0667 & 1.0000 & -0.0667 & 0.0667 & 0.2000 \\ 0.3333 & 0.2000 & 0.0667 & -0.0667 & 1.0000 & -0.4667 & 0.2000 \\ -0.3333 & -0.2000 & 0.4667 & 0.0667 & -0.4667 & 1.0000 & -0.2000 \\ 0.3333 & -0.0667 & -0.2000 & 0.2000 & 0.2000 & -0.2000 & 1.0000 \end{bmatrix}$$

one-shot UMPI test and the HMM tracker that also utilizes the state estimate from the previous slot. Throughout this section, the packet length is assumed to be 128 bits with error correction capability of correcting up to 8 bit errors. The total number of users is assumed to be 20, whereas the spreading vector's length is 31. Again, the spreading vectors are taken from different phases of an m-sequence with $m = 20$. The pairwise correlation coefficients are spaced between -0.4839 to 0.3548 with a bell-shaped histogram.

1) User Detection Performance: In the optimal HMM state tracker, different users' state variables are combined together and user detection is achieved by a single state estimate of a high-dimension HMM. For the suboptimal approach, each individual user is detected separately via a two-state HMM state estimate or, in the absence of traffic information, through a simple thresholding of the UMPI statistic. It is therefore possible to compare user detection performance of the two suboptimal approaches (HMM and one-shot) through their detection probability and false alarm rate by comparing their receiver operating characteristic (ROC) curves. Notice, however, that ROC curves can be only used for binary hypotheses testing, and hence, we will have to concentrate on the detection performance on one of the users in the system, say, the kth user, where the user is ordered by their received power. The decision rule for the kth user using the two schemes are specified, respectively, in (12) and (13).

To actually obtain the ROC curve for the kth user, a very subtle issue is the simulation setting. Traditional ROC curve is based on the repeated sampling paradigm, where repeated samples of test statistic, whose distribution (or histogram) is dictated by the underlying true hypothesis, are collected. Sticking to this repeated sampling idea, we will have to simulate the result for a particular state (in terms of the set of active users). Therefore, it would not really reflect the true performance of the HMM state tracker, which intends to utilize the channel dynamics, i.e., our prior knowledge of the transitions between different states, to help infer about the current state. Therefore, we modify the traditional ROC idea by adopting an "ergodic" sampling idea. That is, a data sequence is generated according to the specified channel dynamics. The probability of detection and false alarm rate for one particular user is computed by averaging over time, instead of the statistical averaging based on repeated sampling. If the sequence generated is long enough such that all state transitions with nonzero probability will have been visited repeatedly, then the obtained receiver performance will be representative of the true performance in an "ergodic" average (as opposed to statistic average) sense.

We choose, in particular, the first user in our simulation, i.e., the user with the minimum power. The SNR is set at 10 dB. It should be noted that the choice of this particular user is rather arbitrary, and we have observed quite similar performance for all the users in our simulation. Packet sequences are generated for all the users according to the specified state transition matrix. Active user detection is implemented for all the users, yet only the result for the first user is used in plotting the ROC curve. The total number of time duration is 50 000 slots. The false alarm rate and detection probability for the first user is directly computed by counting false alarms among the idle slots and correct detection among busy slots with different threshold. The results are plotted in Fig. 7. Clearly, the performance of the HMM state estimator is superior than the one-shot UMPI test. This performance gain is the result of using traffic predictability that is inherently integrated in the hidden Markov modeling. Further, since the impairment on packet detection of a miss detection is much severe than a false alarm, as discussed in Section IV-C, it is desirable to use the operating region with detection probability close to 1. It is easily seen from the ROC curve that in this operating region (where P_D is close to 1), the divergence in terms of false alarm rates between the two detection schemes differ quite dramatically, which explains the significant difference in packet reception performance, as we shall see next.

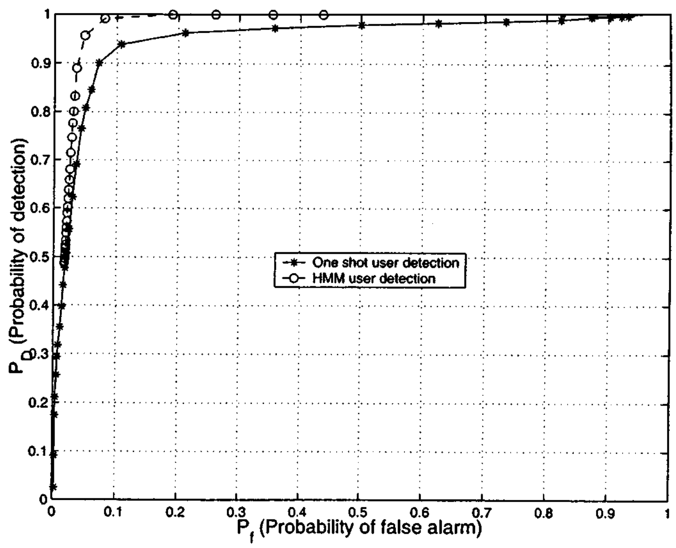

Fig. 7. Receiver operating characteristic curves of the one shot UMPI test and the suboptimal HMM tracker.

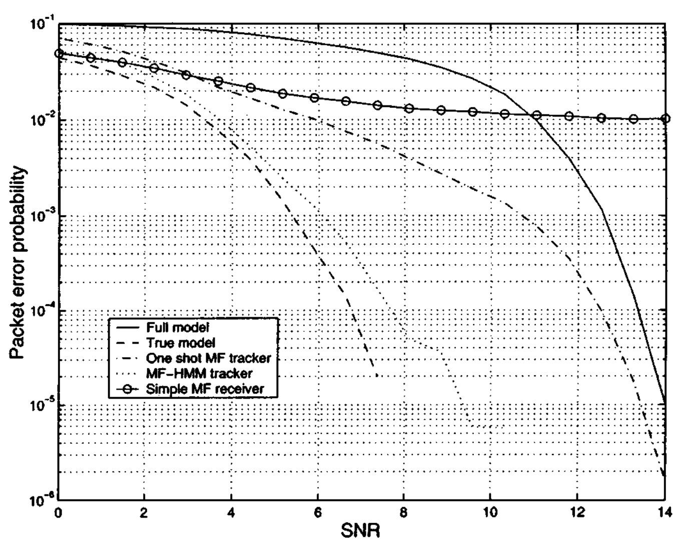

Fig. 8. Performance of suboptimal HMM tracker using MF front end.

2) Packets Reception Performance: From the previous section, we know that one should apply a threshold in the user detection that has more stringent control on the miss detections due to its more severe effect on packet reception performance compared with false alarms. Further, we have argued that an explicit cost matrix that aims to minimize the packet error probability for the two types of errors is impossible to obtain, given the complexity of the problem. In the simulation, the threshold is determined by trial-and-error method for both schemes, and the best thresholds found are used to obtain the packet error probability for both approaches.

It is seen from Fig. 8 that the performance difference between one-shot and HMM approaches is quite dramatic. The reason is that the decoupling of the state tracker, as well as the inaccuracy of Gaussian approximation, results in a poor performance of the one-shot approach. This inaccuracy, fortunately, is well compensated in the HMM tracker by the utilization of the state continuity imposed by the channel burstiness. For comparison, we also plotted out the receiver performance using simple matched filter for symbol detection. Clearly, the packet error probability is saturated as SNR increases. The exhibited error floor effect is due to the presence of the multiple access interference, which dominates the performance when SNR is large.

VI. Conclusions

This paper studies the possibility of utilizing source level traffic predictability in a packet-switching network for the purpose of receiver design. The system considered is the random-access CDMA packet switching networks, and we investigate the implementation of a decorrelating detector. The conventional decorrelating detector that assumes all users are active is shown to perform poorly under typical network traffic conditions. A two-stage MUD is proposed, where the first stage of detecting the set of active users in a packet/CDMA system is investigated in detail.

To take advantage of the traffic predictability in user identification, a first-order Markovian assumption is adopted to describe the traffic burstiness. This reduces the user detection problem to the state estimation problem of a hidden Markov model. Computational concern, however, motivates the development of the suboptimal user tracker with a simple matched filter front end. The idea is to decompose the state tracking into individual user tracking. With the help of traffic predictability, we show that the matched filter-based user tracker can accomplish near-optimal performance compared with a decorrelating detector with perfect user activity information.

Appendix A
Proof of Lemma 1

Denoting $\mathbf{W} = [\mathbf{C}|\mathbf{D}]$, the LS solution of model (3) has a covariance matrix of $\sigma^2(\mathbf{W}'\mathbf{W})^{-1}$. Now

$$\mathbf{W}'\mathbf{W} = \begin{bmatrix} \mathbf{C}'\mathbf{C} & \mathbf{C}'\mathbf{D} \\ \mathbf{D}'\mathbf{C} & \mathbf{D}'\mathbf{D} \end{bmatrix}$$

and invoking the inversion lemma for block matrix [19], we get that the covariance matrix for $\hat{\mathbf{s}}$ is the upper left block of $(\mathbf{W}'\mathbf{W})^{-1}$, which is precisely in the form of (4).

Appendix B
Recursive Update of the Posterior Probability for an HMM Using Forward Variables

We quickly summarize the procedure to recursively update the *a posteriori* probability of a state given past and present data, i.e.,

$$P\left(\gamma_n^{(k)} = i|\mathbf{T}^{(k)}(\mathbf{y}_n), \cdots, \mathbf{T}^{(k)}(\mathbf{y}_1)\right)$$

where $i = 1, 0$. Clearly, $P\left(\gamma_n^{(k)} = i|\mathbf{T}^{(k)}(\mathbf{y}_n), \cdots, \mathbf{T}^{(k)}(\mathbf{y}_1)\right)$ can be done by using the expression

$$P\left(\gamma_n^{(k)} = i|\mathbf{T}^{(k)}(\mathbf{y}_n), \cdots, \mathbf{T}^{(k)}(\mathbf{y}_1)\right) = \frac{\alpha_n(i)}{\sum_j \alpha_n(j)}$$

where $\alpha_n(i)$ is the so-called forward variable, which is defined as

$$\alpha_n(i) = P\left(\mathbf{T}^{(k)}(\mathbf{y}_n), \cdots, \mathbf{T}^{(k)}(\mathbf{y}_1), \gamma_n^{(k)} = i\right).$$

The forward variable can be computed by the following recursion [16]:

$$\alpha_n(i) = \left[\sum_j \alpha_{n-1}(j)A_{ji}\right] f\left(\mathbf{T}^{(k)}(\mathbf{y}_n)|\gamma_n^{(k)} = i\right)$$

where $\mathbf{A} = [A_{ij}]$ is the state transition matrix for $\gamma_n^{(k)}$, and $f\left(\mathbf{T}^{(k)}(\mathbf{y}_n)|\gamma_n^{(k)} = i\right)$ is the likelihood function, which is either chi-square distributed—$\mathbf{T}^{(k)}(\mathbf{y}_n) \sim \hat{\sigma}_k^2\chi_T^2$ or noncentral chi-square distributed—$\mathbf{T}^{(k)}(\mathbf{y}_n) \sim \hat{\sigma}_k^2\chi^2_{T,A^{(k)2}T/2\sigma_k^2}$, depending on the current state of $\gamma_n^{(k)}$. $\hat{\sigma^2}$ is computed using (9).

References

[1] N. Abramson, “Multiple access in wireless digital networks,” *Proc. IEEE*, vol. 82, pp. 1360–1370, Sept. 1994.

[2] Y. Bar-shalom and X. R. Li, *Estimation and Tracking—Principles, Techniques, and Software*. Boston, MA: Artech House, 1993.

[3] B. Chen and L. Tong, “Traffic modeling and tracking for multiuser detection for random access networks,” in *Proc. ICASSP Conf.*, Istanbul, Turkey, June 2000.

[4] D. R. Cox and D. V. Hinkley, *Theoretical Statistics*. New York: Chapman & Hall, 1974.

[5] V. Frost and B. Melamed, “Traffic modeling for telecommunications networks,” *IEEE Commun. Mag.*, Mar. 1994.

[6] R. G. Gallager, “Basic limits on protocol information in data communication networks,” *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 385–399, 1976.

[7] K. W. Halford and M. Brandt-Pearce, “New-user identification in a CDMA system,” *IEEE Trans. Commun.*, vol. 46, pp. 144–155, Jan. 1998.

[8] M. Junti, “Linear multiuser detector update in synchronous dynamic CDMA systems,” in *Proc PIMRC*, Toronto, ON, Canada, 1995, pp. 980–984.

[9] M. Junti, B. Aazhang, and J. Lilleberg, “Iterative implementations of linear multiuser detction for asynchronous CDMA systems,” *IEEE Trans. Commun.*, vol. 46, pp. 503–508, Apr. 1998.

[10] E. L. Lehmann, *Testing Statistical Hypotheses*, 2nd ed. Wiley, New York, 1986.

[11] W. E. Leland, M. S. Taqqu, W. Willinger, and D. Wilson, “On the self-similar nature of ethernet traffic,” *IEEE/ACM Trans. Networking*, vol. 2, pp. 1–15, Feb 1994.

[12] Q. Liu, A. Scholtz, and Z. Zhang, "Channel utilization efficiency and throughput of random access multichannel CDMA systems with multiuser detection," in *Proc. 15th IEEE MILCOM*, Oct. 1996, pp. 918–922.
[13] U. Mitra and H. V. Poor, "Adaptive decorrelating detectors for CDMA systems," *Wireless Pers. Commun. J.*, vol. 2, pp. 256–290, Dec. 1995.
[14] ——, "Analysis of an adaptive decorrelating detector for synchronous CDMA channels," *IEEE Trans. Commun.*, vol. 44, pp. 257–268, 1996.
[15] P. Newman, G. Minshall, and T. L. Lyon, "IP switching—ATM under IP," *IEEE/ACM Trans. Networking*, vol. 6, Apr. 1998.
[16] L. Rabiner, "A tutorial on hidden Markov Models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–285, Feb 1989.
[17] P. B. Rapajic, "Performance analysis of slotted ALOHA/CDMA system with adaptive MMSE receivers," *IEICE Trans. Fundamentals*, vol. E80-A, no. 12, pp. 2485–2492, Dec. 1997.
[18] D. Raychaudhuri, "Performance analysis of random access packet-switched code division multiple access systems," *IEEE Trans. Commun.*, vol. COMM-29, pp. 895–901, June 1981.
[19] J. Schott, *Matrix Analysis for Statistics* Wiley, New York, 1996.
[20] R. Streit and R. Barrett, "Frequency line tracking using hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 586–598, Apr. 1990.
[21] S. Verdu, *Multiuser Detection*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
[22] A. M. Viterbi, "Approximate analysis of time-synchronous packet networks," *IEEE J. Select. Areas Commun.*, vol. SAC-4, pp. 879–890, Sept. 1986.
[23] W. Wu and K. Chen, "Identification of active users in synchronous CDMA multiuser detection," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1723–1735, Dec. 1998.

Biao Chen (M'99) received the B.S. and M.A. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1992 and 1994, respectively. After a short stint with AT&T (China) in 1995, he joined the University of Connecticut, Storrs, where he received the M.S. degree in statistics and the Ph.D. degree in electrical engineering in 1998 and 1999, respectively.

From 1999 to 2000, he was with Cornell University, Ithaca, NY, as a Post-Doctoral Research Associate. He has been with Syracuse University, Syracuse, NY, since 2000 as an Assistant Professor with the Department of Electrical Engineering and Computer Science. His area of interest mainly focuses on statistical signal processing with application to communication, radar, and sonar systems.

Lang Tong (S'87–M'91) received the B.E. degree from Tsinghua University, Beijing, China, in 1985 and the M.S. and Ph.D. degrees in electrical engineering in 1987 and 1990, respectively, from the University of Notre Dame, Notre Dame, IN.

He was a Postdoctoral Research Affiliate with the Information Systems Laboratory, Stanford University, Stanford, CA, in 1991. Currently, he is an Associate Professor with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY. His areas of interest include statistical signal processing, adaptive receiver design for communication systems, signal processing for communication networks, and information theory.

Dr. Tong received the Young Investigator Award from the Office of Naval Research in 1996 and the Outstanding Young Author Award from the IEEE Circuits and Systems Society.