An EM Approach to Multiple-Access Interference Mitigation in Asynchronous Slow FHSS Systems

Xing Tan and John M. Shea

***Abstract*—In this paper, we apply the EM algorithm for mitigation of multi-access interference (MAI) in asynchronous slow frequency-hop spread spectrum (FHSS) systems that employ binary frequency-shift keying (BFSK) modulation. MAI occurs if the hopping patterns of the users are not orthogonal. We show that when FSK signals arrive asynchronously, the time offset exposes portions of the desired and interfering signals in a way that can be exploited to improve the decoder performance. We develop an iterative detection, estimation, and decoding scheme to recover the desired signal in the presence of MAI. We compare the performance of this algorithm with that of a conventional noncoherent BFSK transceiver and show that the EM-based algorithm is particularly effective in the presence of strong interfering signals and allows more users in an FHSS system.**

***Index Terms*—Multiple-access interference mitigation, frequency hop communication, iterative detection and estimation, expectation-maximization algorithm, multiuser detection.**

I. Introduction

FREQUENCY-HOPPING spread spectrum is used to provide robustness to multiple-access interference (MAI) and jamming. In slow frequency-hopping spread-spectrum (FHSS) communication systems, each packet is divided into different segments, which are transmitted over different frequency bands according to a pseudorandom hopping pattern. If the hopping patterns of different transmitters are not orthogonal, then packets from different transmitters may collide (called a "hit") when they occupy the same frequency band. This MAI severely limits the multiple-access capacity of the system.

In contrast with multiple-access mitigation and multiuser detection (MUD) for direct-sequence spread spectrum systems (cf. [1]), MAI mitigation in FHSS is not as well studied. Most previous work focuses on fast FHSS communication, in which each symbol is transmitted at multiple frequencies [2]–[7]. These previous papers focus on suboptimal MUD techniques. For the slow frequency-hopping case, joint detection techniques have been considered for MAI in [8], [9] and for adjacent channel interference in systems with orthogonal hopping patterns in [10]. In all of this previous work, it is assumed that the receiver can simultaneously demodulate the signals at all carrier frequencies, thus requiring a very wideband receiver. Furthermore, it is assumed that the receiver knows the hopping patterns and timing of all of the users in all of these works. In [2]–[5], [7] it is assumed that the symbols are time synchronized at the receiver, and in [6] the powers of the received users are assumed known. These many assumptions make this previous work most appropriate for use at the base station of a cellular system.

Frequency-hopping spread spectrum is also used in ad hoc networks, such as in the military SINCGARS [11] and Have Quick systems. For such systems, the frequency hopping is asynchronous and the complexity of MUD schemes, such as those described in [2]–[10], prohibits their use. Most of the work on improving the performance of frequency hopping systems in ad hoc networks has focused on the mitigation of interference by erasing symbols involved in a hit [12]–[14] or by using an appropriate log-likelihood ratio for symbols involved in a hit [15].

Recently, iterative techniques have been applied to solve several difficult problems in systems with partial-time interference. In [16], a group of receivers uses iterative, collaborative signal processing to detect and excise a jamming signal. In [17], the expectation-maximization (EM) algorithm [18], [19] is applied to the problem of joint data detection and channel estimation in the presence of partial-time interference.

In this paper, we apply an EM-based algorithm for channel estimation and signal detection in the presence of MAI. Our algorithm takes advantage of the way that the desired and interfering signals are partially exposed when frequency-shift keying modulation is used and the symbols from different transmitters arrive asynchronously at a receiver. We note that the ability to discern desired symbols in the presence of asynchronous interference was noted in [20], and in [8] the authors note that the arrival time of each signal is an identifying parameter for asynchronous signals. The EM-based algorithm results in an iterative approach including three processes: channel estimation, soft demodulation, and soft decoding. Information is exchanged among the three processes to refine the decision. The EM algorithm is sensitive to initial conditions, so we consider the design of the initial estimators and the structure of the transmission in a dwell interval to enable better initial estimates.

This paper is organized as follows. The system model is presented in Section II. We propose a channel estimator based on the EM algorithm in Section III. We introduce the soft demodulator and the decoder in Section IV. In Section V, the performance of the EM algorithm is compared with that of the conventional noncoherent demodulator. Conclusions are drawn in Section VI.

Manuscript received DATE; revised DATE; accepted DATE. The associate editor coordinating the review of this letter and approving it for publication was NAME. Portions of this paper were presented at the 2006 IEEE Military Communications Conference. This work was supported in part by the National Science Foundation under Grant CNS-0626863 and by the Air Force Office of Scientific Research under Grant FA9550-07-10456.

The authors are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 (e-mail: {tanxing, jmshea}ufl.edu).

Digital Object Identifier 10.1109/TWC.2008.070084.

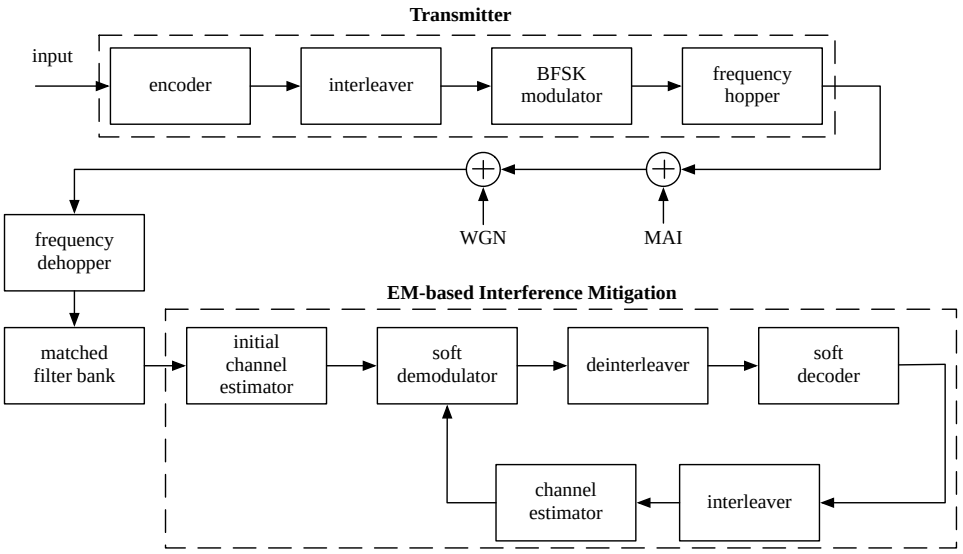


Fig. 1. System model.

II. System Model

The system model considered in this paper is illustrated in Fig. 1. The block labeled "EM-based Interference Mitigation" is designed in this paper and is the topic of Sections III and IV. In this section we describe the remaining blocks, which make up the transmitter, channel, and front-end processing in the receiver.

We consider a FHSS system with multiple users that transmit simultaneously. Each user's transmitter is identical with the exception of the frequency-hopping pattern. The bits at the output of the channel encoder are interleaved and packed into a frame, which we assume is of a fixed length. The symbols in the frame are divided among D dwell intervals of equal length N_D. Each symbol in the frame is modulated using orthogonal BFSK with modulation frequencies (tones) f_0 and f_1. Thus the low-pass equivalent expressions for the signals that are sent for 0 and 1 are $s_0(t) = \sqrt{E_s}e^{2\pi f_0 t+\phi}$, and $s_1(t) = \sqrt{E_s}e^{2\pi f_1 t+\phi}$, respectively. Here we assume that the two modulation tones share the same carrier phase ϕ, but our work can be easily extended to systems in which the carrier phases differ among the two modulation tones.

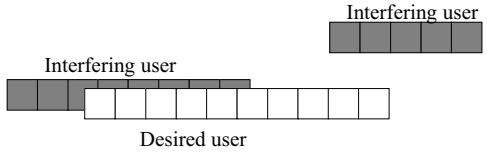
The BFSK symbols are modulated onto a carrier according to the slow frequency-hopping scheme. The channel is divided into F frequency bands. A pseudo-random hopping sequence is used to choose a carrier frequency for each dwell interval, and all of the symbols in a dwell interval are transmitted at the same carrier frequency. The sequences at different users are not necessarily orthogonal, so MAI occurs when two or more users transmit at the same carrier frequency at the same time. The received signals arrive at the receiver with random amplitudes and phases. We consider two different models for the received amplitudes. For the nonfading channel, the amplitude is constant over the entire frame. We consider a frequency-selective Rayleigh fading channel, which we model as a block fading channel with constant amplitude and phase over each dwell interval but independent amplitudes and phases between dwell intervals.

We propose an algorithm to mitigate interference for at most one strong interferer in each dwell interval. Our approach takes advantage of the special structure that is present when there is one asynchronous interfering signal, as described in Sections III and IV-A. If there is more than one strong interferer, detection may be very difficult under any approach except MUD, for which the complexity quickly grows with the number of users involved in the hit [8]. The approach that we consider in no way prohibits the reception of frames that have dwell intervals hit by more than one interferer; we present results in Section V to show that our algorithm still improves performance for frames that have dwell intervals that are hit my more than one interferer. However, the performance gain from using our approach will decrease when dwell intervals are hit by more than one interferer, and the error-control code must then be relied on to provide robustness to the MAI. If we consider a dwell interval that is hit by interference from one other user, we can have one of the three scenarios illustrated in Fig. 2. It could be hit from the left, as in Fig. 2(a); the right, as in Fig. 2(b); or both, as shown in Fig. 2(c).

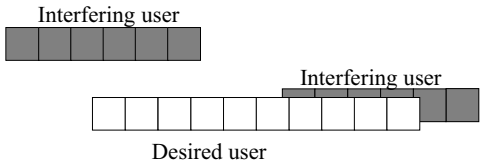
The receive signal is de-hopped and passed into a bank of inphase and quadrature matched filters at frequencies f_0 and f_1. We assume that the receiver can acquire perfect timing for the desired signal, but the timing information for any interfering symbols in a dwell must be estimated. Consider the demodulator outputs for a symbol in the presence of one interfering signal for which the symbol boundaries are τ seconds delayed with respect to the symbol boundaries of the desired signal. Let T denote the symbol duration, and $0 \leq \tau < T$. Again using the complex baseband representation, the decision statistics at the output of the matched filters are

$$\begin{aligned} y_k^0 &= a_1(1-x_k)e^{j\phi_1} + a_2 g_k(\tau)e^{j\phi_2} + n_0, \text{ and} \\ y_k^1 &= a_1 x_k e^{j\phi_1} + a_2 h_k(\tau)e^{j\phi_2} + n_1, \end{aligned}$$

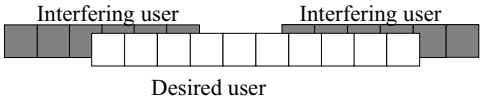
where x_k is the kth transmitted information bit; a_1 and a_2 are the amplitude of the desired symbols and the interfering symbols, respectively; ϕ_1 and ϕ_2 are the random phases of the desired and interfering symbols, respectively; and n_0 and n_1 are zero-mean complex Gaussian random variables



(a) The desired frame is hit from the left.



(b) The desired frame is hit from the right.



(c) The desired frame is hit from both sides.

Fig. 2. Three types of collisions in one dwell interval with a single interferer.

with variance N_0/E_s, where E_s is the symbol energy. The functions $g_k(\tau)$ and $h_k(\tau)$ represent the contributions from any interfering symbols and are defined as

$$\begin{aligned} g_k(\tau) &= \frac{\tau}{T}\delta(I_{k-1}) + \left(1-\frac{\tau}{T}\right)\delta(I_k), \text{ and} \\ h_k(\tau) &= \frac{\tau}{T}\delta(I_{k-1}-1) + \left(1-\frac{\tau}{T}\right)\delta(I_k-1), \end{aligned}$$

where I_{k-1} and I_k are the values of two consecutive interference symbols, and $I_k = -1$ if no interference is present.

In order to simplify the exposition of our joint detection and estimation problem, define the channel parameter $\boldsymbol{\theta} = (a_1, a_2, \phi_1, \phi_2, \tau)$. If we define $\boldsymbol{y} = \{y_1^0, y_1^1, y_2^0, y_2^1, \ldots, y_N^0, y_N^1\}$, $\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$ and $\boldsymbol{I} = \{I_0, I_1, \ldots, I_N\}$, the probability likelihood function of $\boldsymbol{y}$ given $\boldsymbol{x}$, $\boldsymbol{I}$ and $\boldsymbol{\theta}$ is,

$$\begin{aligned} p(y_k^0, y_k^1|\boldsymbol{x}, \boldsymbol{I}, \boldsymbol{\theta}) = \frac{1}{4\pi^2\sigma^4} e^{-\frac{1}{2\sigma^2}\left|y_k^0 - A_1(1-x_k) - A_2 g_k(\tau)\right|^2} \\ \times e^{-\frac{1}{2\sigma^2}\left|y_k^1 - A_1 x_k - A_2 h_k(\tau)\right|^2}. \quad (1) \end{aligned}$$

where $A_1 = a_1 e^{j\phi_1}$ and $A_2 = a_2 e^{j\phi_2}$. Directly estimating $\boldsymbol{x}$ and $\boldsymbol{\theta}$ using (1) is extremely computationally complex, so in Sections III and IV, we apply the EM algorithm to simplify this estimation problem.

III. Estimation Algorithm

A. EM Algorithm

In this section, we apply the EM algorithm to estimate the channel parameter $\boldsymbol{\theta}$. Here, we treat the values of the transmitted symbols for the desired user and the interfering symbols as the missing information in the EM algorithm. The EM algorithm updates the estimate from the equation

$$\boldsymbol{\theta}^{(n)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n-1)}), \quad (2)$$

where $Q(\boldsymbol{\theta}, \boldsymbol{\theta}')$ is Baum's auxiliary function, given by

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}') &= E\left[\log p(\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{I}|\boldsymbol{\theta})|\boldsymbol{y}, \boldsymbol{\theta}'\right] \\ &= E\left[\log p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{I}, \boldsymbol{\theta})|\boldsymbol{y}, \boldsymbol{\theta}'\right] + E\left[\log p(\boldsymbol{x}|\boldsymbol{\theta})|\boldsymbol{y}, \boldsymbol{\theta}'\right] \\ &\quad + E\left[\log p(\boldsymbol{I}|\boldsymbol{\theta})|\boldsymbol{y}, \boldsymbol{\theta}'\right]. \end{aligned}$$

Here $\boldsymbol{y}$ and $\boldsymbol{I}$ are vectors of the received and interfering symbols, respectively; and $\boldsymbol{x}$ is a vector of the transmitted symbols for the desired user. Consider the nth iteration, and for convenience let $\boldsymbol{\theta}' = \boldsymbol{\theta}^{(n-1)}$. Since $\boldsymbol{x}$ and $\boldsymbol{I}$ are assumed independent of $\boldsymbol{\theta}$, the second and third terms will not affect the maximization and thus can be dropped.

Baum's auxiliary function can be further simplified as

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}') &= C_1 + E\left[\log p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{I}, \boldsymbol{\theta})|\boldsymbol{y}, \boldsymbol{\theta}'\right] \\ &= C_2 - \sum_{\boldsymbol{x}, \boldsymbol{I}} \Bigg[\frac{1}{2\sigma^2}\Big(\left|y_k^0 - A_1(1-x_k) - A_2 g_k(\tau)\right|^2 \\ &\qquad + \left|y_k^1 - A_1 x_k - A_2 h_k(\tau)\right|^2 \Big) \\ &\qquad \times p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}') p(\boldsymbol{I}|\boldsymbol{y}, \boldsymbol{\theta}') \Bigg]. \\ &\approx C_2 - \sum_{\substack{k, x_k, \\ I_{k-1}, I_k}} \Bigg[\frac{1}{2\sigma^2}\Big(\left|y_k^0 - A_1(1-x_k) - A_2 g_k(\tau)\right|^2 \\ &\qquad + \left|y_k^1 - A_1 x_k - A_2 h_k(\tau)\right|^2 \Big) \\ &\qquad \times p(x_k|\boldsymbol{y}, \boldsymbol{\theta}') p(I_{k-1}, I_k|\boldsymbol{y}, \boldsymbol{\theta}') \Bigg], \quad (3) \end{aligned}$$

where C_1 and C_2 are constants that are independent of $\boldsymbol{\theta}$. The approximation comes from treating the desired user's symbols as independent at different times and the interfering symbols as independently affecting each symbol from the desired user. These approximations are necessary to reduce the complexity of computing (2).

We want to find the channel parameter $\boldsymbol{\theta}$ that maximizes (3). Define an operator $\nabla_x = \frac{\partial}{\partial u} + j\frac{\partial}{\partial v}$, where $x = u + jv$. The optimal value of $\boldsymbol{\theta}$ can be found by solving the following equations,

$$\nabla_{A_1} Q(\boldsymbol{\theta}, \boldsymbol{\theta}') = 0, \quad (4)$$

$$\nabla_{A_2} Q(\boldsymbol{\theta}, \boldsymbol{\theta}') = 0, \quad (5)$$

$$\frac{\partial}{\partial \tau} Q(\boldsymbol{\theta}, \boldsymbol{\theta}') = 0. \quad (6)$$

By solving (4) and (5), we have the following set of linear equations

$$\begin{aligned} &p_1 A_1 + p_2 A_2 = p_3, \text{ and} \\ &q_1 A_1 + q_2 A_2 = q_3, \end{aligned} \quad (7)$$

where

$$\begin{aligned} p_1 &= N_D \\ p_2 &= \sum_k \sum_{x_k} \sum_{I_{k-1}} \sum_{I_k} \Big\{ \left[(1-x_k) g_k(\tau) + x_k h_k(\tau)\right] \\ &\qquad \times p(x_k|\boldsymbol{y}, \boldsymbol{\theta}') p(I_{k-1}, I_k|\boldsymbol{y}, \boldsymbol{\theta}') \Big\} \\ p_3 &= \sum_k \sum_{x_k} \left[(1-x_k) y_k^0 + x_k y_k^1\right] p(x_k|\boldsymbol{y}, \boldsymbol{\theta}'), \end{aligned}$$

and

$$q_1 = \sum_k \sum_{x_k} \sum_{I_{k-1}} \sum_{I_k} \Big\{ [(1-x_k)g_k(\tau) + x_k h_k(\tau)] \times p(x_k|\boldsymbol{y},\boldsymbol{\theta}')p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}') \Big\}$$

$$q_2 = \sum_k \sum_{I_{k-1}} \sum_{I_k} \left[g_k(\tau)^2 + h_k(\tau)^2\right] p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}')$$

$$q_3 = \sum_k \sum_{I_{k-1}} \sum_{I_k} \left[g_k(\tau)y_k^0 + h_k(\tau)y_k^1\right] p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}').$$

Here p_1, p_2, q_1, and q_2 are real numbers, but p_3 and q_3 take complex values.

From (6) we have

$$r_1\left(\frac{\tau}{T}\right) = r_2 - r_3, \tag{8}$$

where r_1, r_2, r_3 are all real numbers and defined as follows,

$$r_1 = \sum_k \sum_{I_{k-1}} \sum_{I_k} |A_2|^2[(\delta(I_{k-1}) - \delta(I_k))^2 + (\delta(I_{k-1}-1) + \delta(I_k-1))^2]p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}')$$

$$r_2 = \sum_k \sum_{I_{k-1}} \sum_{I_k} \sum_{x_k} [\mathrm{Re}\{A_2(\delta(I_{k-1}) - \delta(I_k))(y_k^{0*} - A_1^*(1-x_k))\} + \mathrm{Re}\{A_2(\delta(I_{k-1}-1) - \delta(I_k-1))(y_k^{1*} - A_1^* x_k)\}] \times p(x_k|\boldsymbol{y},\boldsymbol{\theta}')p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}')$$

$$r_3 = \sum_k \sum_{I_{k-1}} \sum_{I_k} |A_2|^2 \Big\{ [(\delta(I_{k-1}) - \delta(I_k))(\delta(I_k)) + (\delta(I_{k-1}-1) + \delta(I_k-1))(\delta(I_k-1))] \times p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}') \Big\}.$$

It is difficult solve (7) and (8) simultaneously; however, we can solve for A_1, A_2, τ by solving (7) and (8) iteratively.

To solve (7) and (8), we need the *a posteriori* probabilities $p(x_k|\boldsymbol{y},\boldsymbol{\theta}')$ and $p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}')$. Given the channel parameter $\boldsymbol{\theta}'$, we show in Section IV that these *a posteriori* probabilities can be determined from two BCJR algorithms, which we call the soft demodulator and soft decoder. However, in the first iteration, the channel estimate from solving (7) and (8) will not be accurate if $p(x_k|\boldsymbol{y},\boldsymbol{\theta}')$ and $p(I_{k-1},I_k|\boldsymbol{y},\boldsymbol{\theta}')$ are not available. Thus, in the first iteration, we apply a separate channel estimate to start the soft demodulation and soft decoding processes.

B. Initial Estimation

To simplify the estimation process, we eliminate the need to estimate the phase ϕ_1 of the desired signal by using noncoherent demodulation, as described in Section IV. However, the values of a_1, a_2, ϕ_2, and τ still have to be estimated for each dwell interval. The timing information can be used to accurately distinguish the desired signal from the interfering signal only if τ is not close to 0 or T. So, for the initial estimate, we let $\tau = 0.5T$. The initial estimate for a_1 for the ith dwell interval is denoted $a_1^{(i)}$ and is found by averaging all the symbols in a dwell interval,

$$|a_1^{(i)}| = \frac{1}{N_D}\left|\sum_{k=1}^{N_D}(y_0^k + y_1^k)\right|.$$

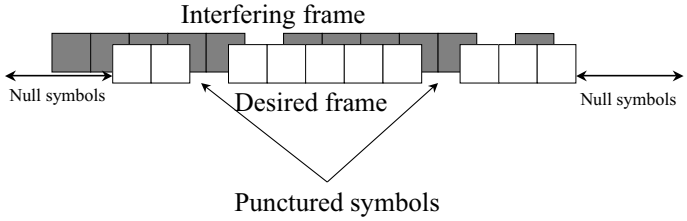


Fig. 3. Symbols are punctured using a pseudo-random pattern in each dwell interval to estimate the channel parameters of any interfering signal.

For the nonfading additive white Gaussian noise (AWGN) channel, we use a single estimate of a_1 that is generated by ordering all the estimates $a_1^{(i)}$ and averaging over the middle 40% to reduce the impact of dwell intervals that are corrupted by jamming.

To assist in the estimation of the amplitude and phase of the interfering symbols, we insert M null symbols at the beginning and the end of each dwell interval. During the null symbols, the transmitter remains silent so that the receiver can evaluate the amplitude and phase of the interference, if there is any. However, the desired and interfering frames are occasionally received almost simultaneously at the receiver, and in this case, the receiver cannot accurately sense any interference during the null symbols at the beginning and the end of the dwell interval. To solve this problem, we puncture R symbols in each dwell interval ($R << N_D$) according to a pseudo-random pattern, and the transmitter sends nothing (nulls) during the punctured symbols, as shown in Fig. 3. This enables a more accurate estimate of the interfering channel, even if the interfering signal's dwell interval aligns with that of the desired signal.

We generate three estimates of a_2 and ϕ_2 using the null symbols at the beginning and end of the dwell interval as well as the punctured symbols in the dwell interval. Let $\hat{a}_2^{(i)}$ and $\hat{\phi}_2^{(i)}$ denote the ith estimates. If a null symbol contains interference, it can be expressed as $z_k^0 = a_2 g_k(\tau)e^{j\phi_2} + n_0$ and $z_k^1 = a_2 h_k(\tau)e^{j\phi_2} + n_1$, where $1 \le k \le M$. We apply maximum-likelihood estimation for a_2 and ϕ_2, requiring maximization of

$$\Lambda(\phi_2) = \log P(z_1^0, z_1^1, z_2^0, z_2^1, ..., z_M^0, z_M^1|\boldsymbol{I},\boldsymbol{\theta}) = C_1 + \sum_{k=1}^{M}\left(\frac{2}{N_0}\mathrm{Re}\{z_k^0 a_2 g_k e^{-j\phi_2}\} + \frac{2}{N_0}\mathrm{Re}\{z_k^1 a_2 h_k e^{-j\phi_2}\}\right).$$

Since we are estimating the channel parameters before estimating x_k and I_k, we let $g_k = h_k = \frac{1}{2}$, yielding

$$\Lambda(\phi_2) = C_1 + \frac{1}{N_0}\mathrm{Re}\left\{\sum_{k=1}^{M}(z_k^0 + z_k^1)a_2 e^{-j\phi_2}\right\},$$

which is maximized by $\hat{\phi}_2^{(i)} = \angle \sum_{k=1}^{M} (z_k^0 + z_k^1)$ (i.e., the angle of $\sum_{k=1}^{M} (z_k^0 + z_k^1)$).

Similarly, $\hat{a}_2^{(i)}$ can be found from

$$\begin{aligned}\Lambda(a_2) &= \log P(z_1^0, z_1^1, z_2^0, z_2^1, ..., z_M^0, z_M^1 | \boldsymbol{I}, \boldsymbol{\theta}) \\ &= C_1 - \frac{1}{N_0} \sum_{k=1}^{M} (|z_k^0 - a_2 g_k e^{-j\phi_2}|^2 \\ &\qquad + |z_k^1 - a_2 h_k e^{-j\phi_2}|^2) \\ &= C_1 - \frac{1}{N_0} \sum_{k=1}^{M} \big(|z_k^0|^2 + |z_k^1|^2 - 2\,\mathrm{Re}\{z_k^0 a_2 g_k e^{-j\phi_2}\} \\ &\qquad - 2\,\mathrm{Re}\{z_k^1 a_2 h_k e^{-j\phi_2}\} + a_2^2 g_k^2 + a_2^2 h_k^2 \big).\end{aligned}$$

Taking $g_k = h_k = \frac{1}{2}$ and $\phi_2 = \angle \sum_{k=1}^{M} (z_k^0 + z_k^1)$ yields

$$\Lambda(a_2) = C_2 - \frac{M}{2N_0} a_2^2 + \frac{1}{N_0} \left| \sum_{k=1}^{M} (z_k^0 + z_k^1) \right| a_2.$$

Thus, the maximizing value is $\hat{a}_2^{(i)} = \frac{1}{M} \left| \sum_{k=1}^{M} (z_k^0 + z_k^1) \right|$. The overall estimate of a_2 is found as $\hat{a}_2 = \max \left\{ \hat{a}_2^{(1)}, \hat{a}_2^{(2)}, \hat{a}_2^{(3)} \right\}$. Then $\hat{\phi}_2$ is set equal to the value of $\hat{\phi}_2^{(i)}$ that corresponds to the maximum value $\hat{a}_2^{(i)}$.

For the nonfading channel, the EM algorithm is used to update $\boldsymbol{\theta}$ in every iteration. However, for the Rayleigh fading channel, we found that the EM algorithm often had a problem distinguishing between the desired and interfering signals over several iterations. We found that for the values of signal-to-noise ratio that give reasonable block error rates, the initial estimate $\hat{a_2}$ is sufficiently accurate that it can be fixed for all iterations to avoid this problem. Thus, we apply this approach for the Rayleigh fading channel results in this paper.

IV. Soft Demodulator and Decoder

As mentioned in Section III-A, the EM update equations (7) and (8) require knowledge of the *a posteriori* probabilities $p(x_k | \boldsymbol{y}, \boldsymbol{\theta}')$ and $p(I_{k-1}, I_k | \boldsymbol{y}, \boldsymbol{\theta}')$. These probabilities can be found by utilizing the estimate for $\boldsymbol{\theta}$ that is described in Section III.

A. Soft Demodulator

In this section, we develop a trellis-based algorithm to calculate the probabilities of the desired and interfering symbols. At this point in the estimation process, we do not utilize the fact that the symbols are encoded by a forward error-correction code. As explained in Section II, we design the estimator for the case of at most one strong interferer in each dwell interval. We also assume the number of punctured symbols satisfies $R << N_D$, so that it has little effect on the algorithm described in this section.

As described in Section II and illustrated in Fig. 2, there are three types of interference that can occur in a dwell interval when a single strong interferer is present. The dwell interval can be hit from the left, the right, or from both sides. Because the interferer is asynchronous with the desired user, each desired symbol may be hit with interference from two consecutive interfering symbols. We refer to the earlier and later of the interfering symbols as the left and right interfering symbols, respectively.

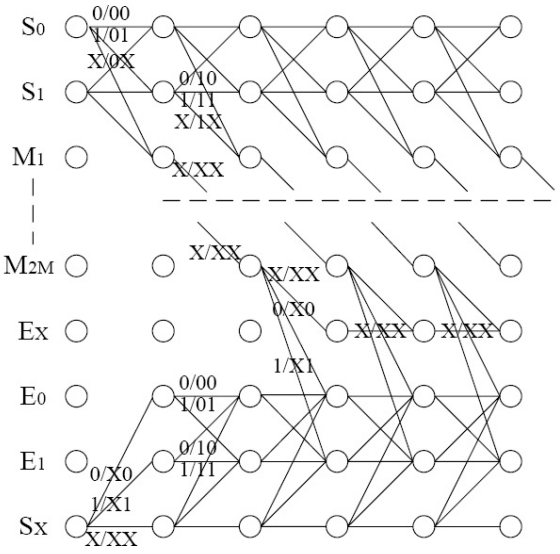


Fig. 4. Trellis used in the BCJR algorithm for the soft demodulator.

We use the trellis illustrated in Fig. 4 to estimate the desired and interfering signals by taking advantage of the asynchronous nature of the two signals. The receiver also uses knowledge about the types of hits (left and right) to estimate when the interference starts and stops. The trellis starts in one of three states: S_0, S_1, or S_X. When the desired frame is hit from the left, the trellis will start at state S_0 or S_1, corresponding to the value of the first left interfering symbol. State S_X corresponds to the event that the desired symbol is not hit from the left. If the trellis is in state S_0 or S_1, then the dwell interval has been hit from the left, and the trellis can transition to either state S_0 or S_1 if the next desired symbol is hit by a left interfering symbol, or can transition to M_1 when the interference from the left terminates. The $2M$ states M_1 to M_{2M} are states used to enforce the $2M$ null symbols that exist between consecutive dwells. During these states, there is no interference. Each state $M_i, i = 1, 2, \ldots, 2M-1$ transitions to state M_{i+1}. State M_{2M} transitions to either state E_0 or E_1 if the interferer stays at the same frequency as the desired user, or state M_{2M} transitions to state E_X if the desired frame is not hit from the right. State S_X transitions to state E_0 or E_1 if the desired frame is hit from the right, where E_0 and E_1 represent the events that the desired signal is hit by a right interfering symbol at frequency f_0 or f_1, respectively.

The branch input to the trellis is the interference symbol, taking values from the set $\{X, 0, 1\}$, which stand for no interference, interference symbol 0, and interference symbol 1, respectively. The trellis transitions to states corresponding to its branch input. For example, a branch input of 0 brings the trellis to one of the two states: S_0, E_0; an input of 1 brings the trellis to one the two states: S_1, E_1; and an input of X brings the trellis to one of the other $2M + 2$ states.

This trellis representation can be used to calculate probabilities of the interference symbols as well as the desired symbols

using the BCJR algorithm [21]. The branch metric $\gamma_k(s', s)$ for the branch connecting state s' at time $k-1$ to state s at time k is

$$\begin{aligned}\gamma_k(s', s) &= p(y_k^0, y_k^1, s_k = s | s_{k-1} = s') \\ &= p(y_k^0, y_k^1 | s_{k-1} = s', s_k = s) p(s_k = s | s_{k-1} = s') \\ &= \left\{ \sum_{x_k} p\left[y_k^0, y_k^1 | I_{k-1}(s_{k-1}), I_k(s_k), x_k\right] p(x_k) \right\} \\ &\quad \times p(s_k = s | s_{k-1} = s'). \end{aligned} \quad (9)$$

Here $I_\ell(s_\ell) = 0$ if $s_\ell \in \{S_0, E_0\}$, $I_\ell(s_\ell) = 1$ if $s_\ell \in \{S_1, E_1\}$, and $I_\ell(s_\ell) = -1$ otherwise. The probability $p(y_k^0, y_k^1 | I_{k-1}(s_{k-1}), I_k(s_k), x_k)$ is derived from (1) by integrating over ϕ_1, since we adopt noncoherent detection, which yields

$$\begin{aligned} &p(y_k^0, y_k^1 | I_{k-1}(s_{k-1}), I_k(s_k), x_k = \ell) \\ &= \exp\left[-\frac{1}{2\sigma^2} (|y_k^0 - A_2 g_k(\tau)|^2 + |y_k^1 - A_2 h_k(\tau)|^2 + a_1^2) \right] \\ &\quad \times I_0\left(\frac{a_1}{\sigma^2} |y_\ell - A_2 H_k(\tau, \ell)| \right), \end{aligned}$$

where $H_k(\tau, \ell) = g_k(\tau)$ if $\ell = 0$ and $H_K(\tau, \ell) = h_k(\tau)$ if $\ell = 1$. Here $p(x_k)$ is the *a priori* probability for x_k. In the first iteration, we let $p(x_k = 0) = p(x_k = 1) = 0.5$. In future iterations, we replace $p(x_k)$ with the estimate for $p(x_k)$ from the output of the MAP decoder for the error-correcting code.

Unlike decoding with the BCJR algorithm, in which the state transition probabilities $p(s_{k+1} = s | s_k = s')$ represent *a priori* probabilities for the information bits, are generally taken to be equally likely, and are not a function of time, the probabilities $p(s_{k+1} = s | s_k = s')$ in our algorithm represent probabilities of a symbol being part of a particular type of hit, are not equally likely, and are a function of the time k. For example, the symbols at the beginning of a dwell interval (i.e., for small k) are more likely to be hit from the left by MAI. On the other hand, the symbols at the end of a dwell interval (i.e. for large k) are more likely to be hit from the right by MAI. We provide details of the calculation of these probabilities in the Appendix.

With the branch metric $\gamma(s', s)$ known, the forward- and backward-looking state probabilities, $\alpha(s)$ and $\beta(s')$ [21], can be calculated as

$$\begin{aligned} \alpha_{k+1}(s) &= \sum_{s'} \alpha_k(s') \gamma_{k+1}(s', s), \text{ and} \\ \beta_{k-1}(s') &= \sum_{s} \beta_k(s) \gamma_k(s', s). \end{aligned}$$

The initial values of α and β are the probabilities that the trellis starts (ends) at each initial (terminal) state:

$$\begin{aligned} \alpha_0(s = S_i) &= \frac{1}{2F} \left(\frac{N_D}{N_D + 2M} \right), && i = 0, 1 \\ \alpha_0(s = S_X) &= 1 - \frac{1}{F} \left(\frac{N_D}{N_D + 2M} \right), \\ \beta_{N_D}(s = S_i) &= 0, && i = 0, 1 \\ \beta_{N_D}(s = M_i) &= \frac{1}{F(N_D + 2M)}, && 1 \le i \le 2M \\ \beta_{N_D}(s = E_X) &= \frac{1}{F} \left(1 - \frac{1}{F} \right) \frac{N_D - 2M}{N_D + 2M}, \\ \beta_{N_D}(s = E_i) &= \frac{1}{2F} \left(\frac{N_D}{N_D + 2M} \right), && i = 0, 1 \\ \beta_{N_D}(s = S_X) &= \frac{1}{F} \left(1 - \frac{1}{F} \right) \\ &\quad \times \left(\frac{F(N_D + 2M) - (N_D - 2M)}{(N_D + 2M)} \right). \end{aligned}$$

The joint *a posteriori* probabilities for consecutive interference symbols are then easily calculated as

$$p(I_{k-1}, I_k, \boldsymbol{y}) = C \sum_{U\{I_{k-1}, I_k\}} \alpha_{k-1}(s') \gamma_k(s', s) \beta_k(s), \quad (10)$$

where $I_k \in \{X, 0, 1\}$. $U\{I_{k-1}, I_k\}$ represents all pairs of (s', s) that have I_{k-1}, I_k as its branch output. C is a normalization constant to ensure that the probabilities sum to one.

The primary output of the soft demodulator is extrinsic information about the transmitted symbol x_k. The likelihood function for the transmitted symbol x_k is

$$p(y_k^0, y_k^1 | x_k) = \sum_{I_{k-1}} \sum_{I_k} p(y_k^0, y_k^1 | x_k, I_{k-1}, I_k) p(I_{k-1}, I_k),$$

where we approximate the probabilities $p(I_{k-1}, I_k)$ with $p(I_{k-1}, I_k | \boldsymbol{y})$.

The output of the soft demodulator is

$$LLR(x_k) = \log \frac{p(y_k^0, y_k^1 | x_k = 0)}{p(y_k^0, y_k^1 | x_k = 1)} + \log \frac{p(x_k = 0)}{p(x_k = 1)},$$

where $\log[p(x_k = 0)/p(x_k = 1)]$ is the extrinsic information coming from the decoder. Thus, the extrinsic information produced by the soft demodulator is

$$L_{demod}^e(x_k) = \log \frac{p(y_k^0, y_k^1 | x_k = 0)}{p(y_k^0, y_k^1 | x_k = 1)}.$$

B. Soft Decoder

The soft demodulator produces symbol-wise estimates of the transmitted symbol x_k. This information can be further improved by decoding the entire frame, which is coded and interleaved so that symbols from dwell intervals with interference are distributed throughout the coded frame. Here we assume that the information is encoded with a rate 1/2 convolutional code. However, our technique easily extends to other codes. In particular, this technique is particularly appropriate for use with turbo and low-density parity-check codes.

The decoder uses the extrinsic information generated in the soft-demodulator as the branch metric in a BCJR algorithm for the convolutional code,

$$\overline{\gamma}_k(s', s) = p_k^{(1)} L_{demod}^e(p_k^{(1)})/2 + p_k^{(2)} L_{demod}^e(p_k^{(2)})/2,$$

where $p_k^{(1)}$ and $p_k^{(2)}$ are the first and the second parity bit, respectively, at time k. The soft decoder generates extrinsic information for the transmitted symbols $L_{decode}^e(x_k)$, which is

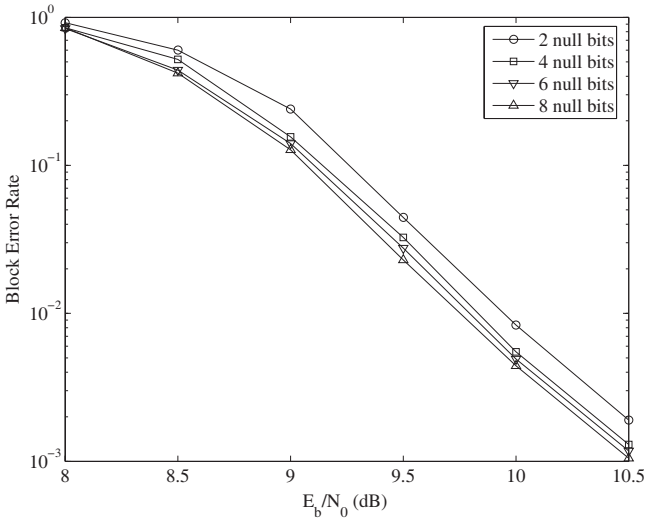


Fig. 5. Block error rate vs. E_b/N_0 for different numbers of null bits at each end of a dwell interval. Two users occupying ten frequency bands with SIR =-6 dB.

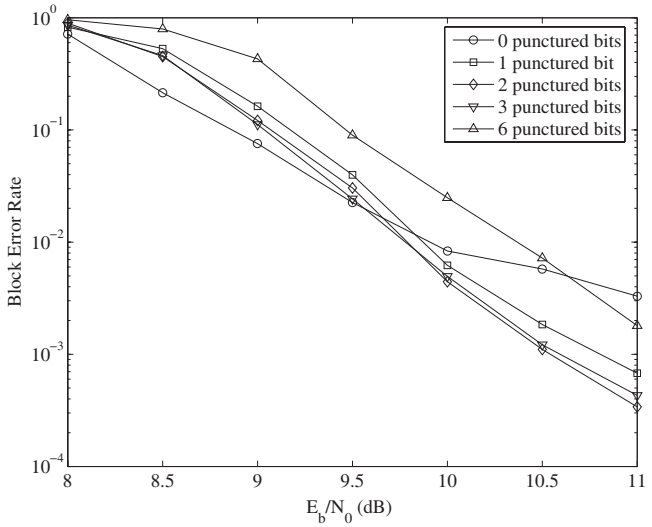


Fig. 6. Block error rate vs. E_b/N_0 for different numbers of punctured bits. Two users occupying ten frequency bands with SIR =-6 dB.

used in the next iteration of the channel estimation algorithm and the soft demodulator. The soft decoder also generates the *a posteriori* estimate for the information bits. We assume that an error-detecting code has been applied by the higher layers, and at each iteration of the algorithm we check the error-detecting code to determine whether correct decoding has occurred. If the packet has decoded correctly, then the iterations are terminated.

V. Performance Results

We simulated the performance of the proposed algorithm for a fixed frame size of 1000 information bits encoded with a rate 1/2 convolutional code with constraint length 7 of maximum free distance. The encoded bits are interleaved using a pseudo-random channel interleaver. Except where noted, we assume that the frame is transmitted over 10 dwell intervals. The channel is either a nonfading AWGN channel or a frequency selective, slow Rayleigh fading channel. For the AWGN channel, a_1, a_2 are fixed for each signal-to-interference ratio (SIR) and ϕ_1, ϕ_2 are random variables that are uniformly distributed on $[0, 2\pi)$. For the Rayleigh fading channel, we model A_1 and A_2 as complex Gaussian random variables that are constant over each dwell interval and independent among dwell intervals. For both channels, τ is uniformly distributed on $[0, T)$. We assume that all of these parameters are unknown at the receiver. We compare the performance of our algorithm with a conventional BFSK system that uses the same system parameters but does not perform interference mitigation at the receiver. Except where noted, six iterations of the EM algorithm are used.

Consider first the performance for a system in which each dwell interval has interference from at most one user. For this purpose, we use a simple system with two users in ten frequency bands. We investigate the effect of the choice of the number of null bits M that are used at the beginning and end of each dwell interval and the number of bits that are punctured from each interval at a signal-to-interference ratio (SIR) of -6 dB. The results in Fig. 5 show the block error rate as a function of E_b/N_0 for different numbers of null bits at each end of a dwell interval. Increasing the number of null bits improves the block error rate but reduces the overall spectral efficiency. The results indicate that the performance begins to saturate for more than four to six null bits. In all future results, we use five null bits. The block error rate for different number of punctured bits is shown in Fig. 6 at a SIR of -6 dB. Puncturing more bits improves knowledge of the interferer's parameters but degrades the performance of the convolutional code. The results show that for this scenario, two punctured bits offers the best performance. We use two punctured bits for all of the remaining results in this paper.

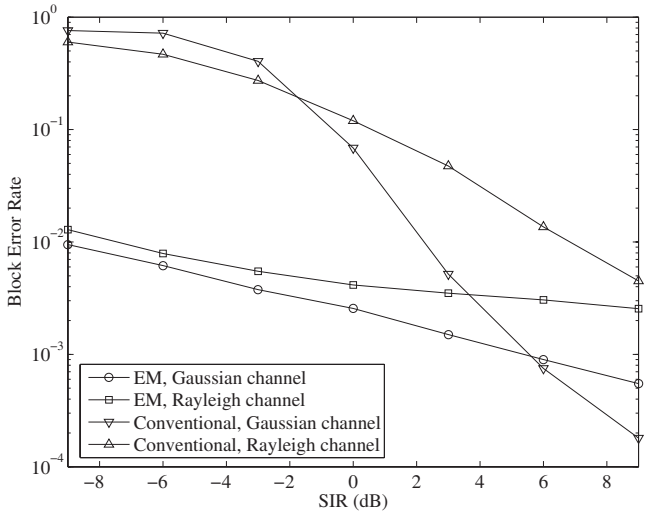


Fig. 7. Block error rate vs. SIR for two users occupying ten frequency bands. For the AWGN channel, $E_b/N_0 = 10$ dB, and for the Rayleigh fading channel, $\overline{E_b}/N_0 = 18$ dB.

The block error rate in both AWGN and Rayleigh fading as a function of SIR is shown in Fig. 7. For the AWGN channel, $E_b/N_0 = 10$ dB, and for the Rayleigh channel the average bit energy-to-noise density ratio $\overline{E_b}/N_0$ is 18 dB. The results show that the EM algorithm provides better performance than the conventional receiver for all considered SIRs on the

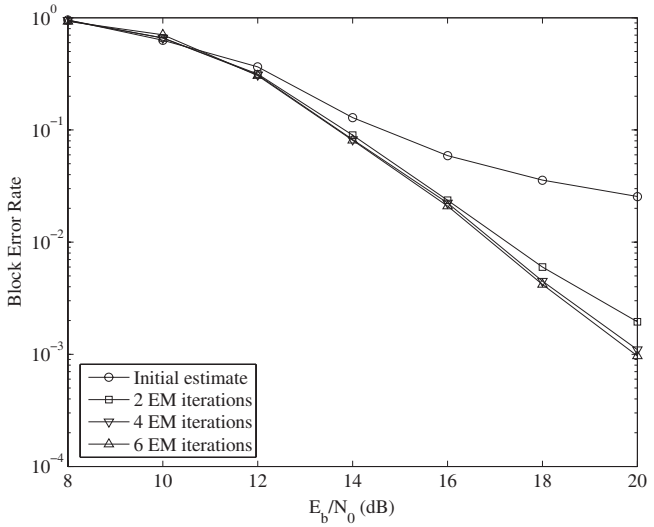


Fig. 8. Block error rate vs. $\overline{E_b}/N_0$ for two users occupying ten frequency bands on a block-fading Rayleigh channel with various number of iterations for the EM algorithm.

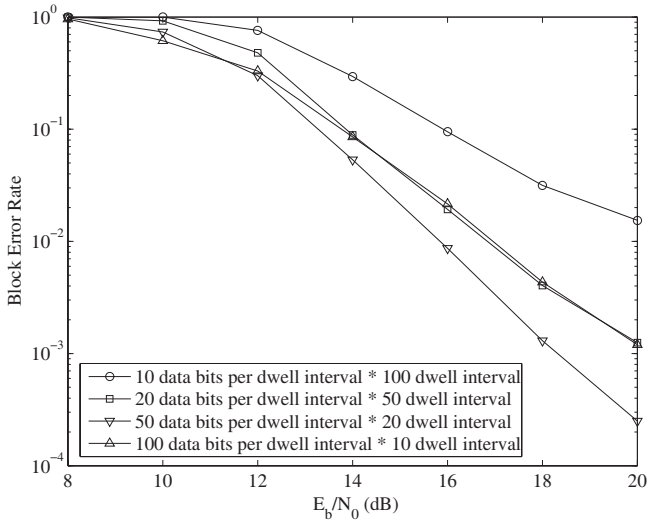


Fig. 9. Block error rate vs. $\overline{E_b}/N_0$ for two users occupying ten frequency bands on a block-fading Rayleigh channel with different numbers of bits per dwell interval. The average SIR is 0 dB.

Rayleigh channel and for SIRs smaller than 7 dB on the AWGN channel. As the interference becomes stronger, the performance gain of our algorithm becomes larger. For the AWGN channel with an SIR larger than 7 dB, the performance of our algorithm is a little bit worse than the conventional algorithm. In this range, the interference is so weak that the EM algorithm cannot accurately detect it and estimate its parameters and thus actually makes the performance worse. This performance difference can be removed by running the conventional algorithm and checking the error-detecting code before running our EM algorithm.

For the remaining results, we focus on the performance of our algorithm on the Rayleigh fading channel. We further investigate the choice of system parameters in Figs. 8 and 9. In Fig. 8, we show the effects of the number of iterations of the EM algorithm. There is little gain from using more than four iterations in this scenario. We also investigate the effect

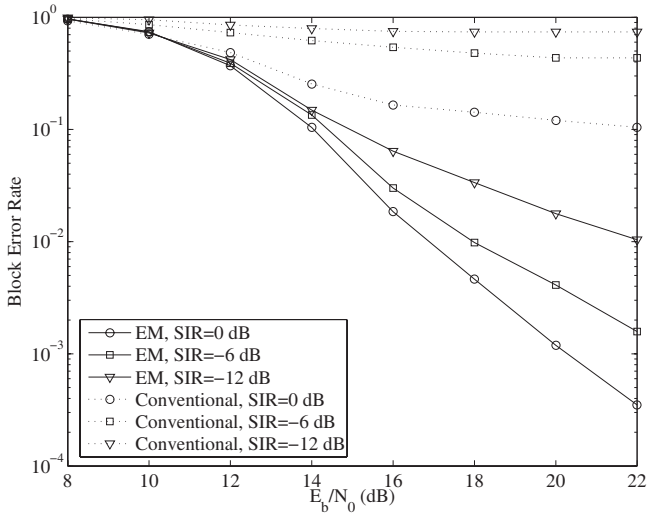


Fig. 10. Block error rate vs. $\overline{E_b}/N_0$ for two users occupying ten frequency bands on a Rayleigh fading channel with average SIR of 0 dB,-6 dB, or -12 dB.

of varying the number of bits per dwell interval in Fig. 9. We keep the frame length fixed at 1000 data bits, so the number of dwell intervals per frame varies with the number of data bits per dwell interval. It is expected that the performance will be poor with too few bits per dwell interval because the EM algorithm will not be able to accurately estimate the parameters of an interferer without sufficient data. It is also expected that the performance will be poor with too many bits per dwell interval, as the system loses diversity against the fading and interference. The results confirm this, showing that the best performance is obtained with 20 dwell intervals of 50 data bits per dwell interval. The block error rate is shown in Fig. 10 as a function of E_b/N_0 for average SIRs of -12 dB, -6 dB, and 0 dB with $\overline{E_b}/N_0 = 18$ dB. The EM-based algorithm performs much better than the conventional algorithm, reducing the block error rate by more than two orders of magnitude in some cases and removing the error floor that limits the performance of the conventional receiver.

We next consider a more practical system with multiple users occupying 100 frequency bands. We vary the number of users to evaluate the impact of our algorithm on the multiple-access capability of the system. The results in Fig. 11 illustrate the block error rate as a function of the number of users in the system when the signals are received at an average E_b/N_0 of 18 dB. We assume that all users are received with the same average signal strength. The figure shows that our algorithm allows far more users than the conventional algorithm, given the same block error rate requirement. For instance, at a target block error rate of 2×10^{-2}, the conventional system can support only 2 users, whereas our EM interference mitigation algorithm can support 19 users.

Finally, we investigate the performance of our system when a frame is simultaneously hit by more than one interferer. To generate the results in Fig. 12, we simulated 20 users in 100 frequency bands with Rayleigh fading and equal average powers. We categorized the frames of one user based on the maximum number of interferers that were present in any one dwell interval of the frame. We then plotted the block error

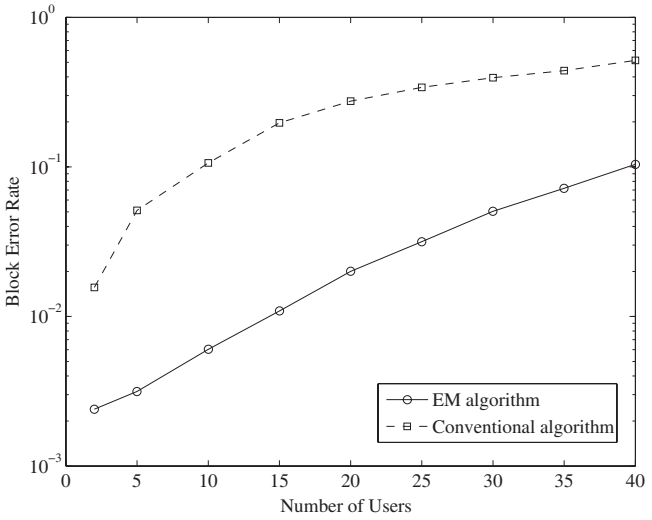


Fig. 11. Block error rate vs. number of users occupying 100 frequency bands and Rayleigh fading channels for $\overline{E_b}/N_0 = 18$ dB, equal average received powers for all users.

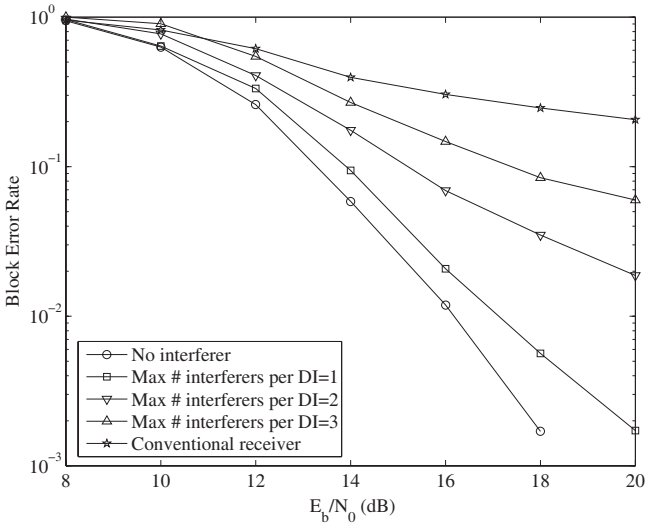


Fig. 12. Block error rate vs. $\overline{E_b}/N_0$ for different numbers of maximum interferers in any dwell interval of a frame. Results are for 20 users at equal average received power occupying 100 frequency bands.

rate as a function of $\overline{E_b/N_0}$ for each category in Fig. 12. The results show that the performance of our algorithm is close to that of a system with no interference when there is at most one interferer in any dwell interval of a frame. The performance degrades significantly for frames that experience simultaneous hits by two or three interferers, but the performance is still much better than the conventional receiver.

VI. Conclusion

In this paper, we proposed an interference-mitigation algorithm for use with asynchronous frequency-hopping spread spectrum systems that employ BFSK modulation. Our receiver structure is based on the EM algorithm and decomposes the detection problem into an iterative algorithm consisting of channel estimation, soft demodulation, and soft decoding processes. The soft demodulator exploits the structure that comes from the interference being asynchronous with the desired signal. The results show that on nonfading and Rayleigh fading channels, the algorithm that we propose significantly improves performance in the presence of strong interferers. For instance, for a block Rayleigh fading channel with average E_b/N_0 of 18 dB, average SIR of 0 dB, and target block error rate of 2×10^{-2}, our receiver structure can support 19 users in 100 frequency bands, whereas the conventional receiver can only support 2 users.

Appendix

In this Appendix, we calculate the state transition probabilities $p(s_k|s_{k-1})$ for the BCJR algorithm in the soft demodulator. The probability that the $(k-1)$th dwell interval of the interfering user occupies the same frequency band as the desired user is $1/F$. Since the $2M$ null symbols do not cause interference, the probability that the first symbol of the desired user is hit when they both occupy the same frequency band is $N_D/(N_D+2M)$. We assume that the interfering symbols are equally likely to be 0 or 1. Thus, we have,

$$p(s_0 = S_i) = \frac{1}{2F}\left(\frac{N_D}{N_D+2M}\right), \qquad i = 0,1$$
$$p(s_0 = S_i, s_1 = M_1) = \frac{1}{2F(N_D+2M)}, \qquad i = 0,1$$
$$p(s_1 = M_1|s_0 = S_0) = \frac{1}{N_D},$$
$$p(s_1 = S_j|s_0 = S_0) = \frac{1}{2}\left(1 - \frac{1}{N_D}\right), \qquad i = 0,1; j = 0,1$$

Then

$$\begin{aligned} p(s_1 = S_j) &= p(s_1 = S_j|s_0 = S_0)P(s_0 = S_0) \\ &\quad + p(s_1 = S_j|s_0 = S_0)P(s_0 = S_0) \\ &= \frac{1}{2F}\left(\frac{N_D - 1}{N_D + 2M}\right) \end{aligned}$$

for $j = 0, 1$ and by repeatedly applying this approach,

$$p(s_{k+1} = S_j|s_k = S_i) = \frac{1}{2}\left(1 - \frac{1}{N_D - k}\right), \quad i,j \in \{0,1\}$$

Similarly,

$$p(s_{k+1} = M_1|s_k = S_i) = \frac{1}{N_D - k}, \qquad i = 0,1.$$

These probabilities match the intuitive notion that a hit from the left is most likely to affect symbols at the beginning of the dwell interval. As time moves further into the trellis, the probability that the dwell interval continues to be hit from the left decreases.

As previously discussed, the transitions between the null states are deterministic, so $p(s_{k+1} = M_{i+1}|s_k = M_i) = 1$, for $i = 1, 2, \ldots, 2M-1$. The probability that the dwell interval is hit from the right is $1/F$, so

$$p(s_{k+1} = E_X|s_k = M_{2M}) = 1 - \frac{1}{F},$$
$$p(s_{k+1} = E_i|s_k = M_{2M}) = \frac{1}{2F}, \qquad i = 0,1$$
$$p(s_{k+1} = E_j|s_k = E_i) = \frac{1}{2}, \qquad i = 0,1; j = 0,1.$$

The calculation of $p(s_{k+1} = S_X|s_k = S_X)$, $p(s_{k+1} = E_0|s_k = S_X)$ and $p(s_{k+1} = E_1|s_k = S_X)$ are more complicated and deserve further explanation. The joint probability function $p(s_k = S_x, s_{k+1} = E_i)$ for $i = 0, 1$ have different expressions on two separate intervals. When $k < 2M$, it amounts to the probability of being hit from the right at the kth symbol; when $k \geq 2M$, it amounts to the probability of avoiding hit from the left but being hit from the right at the kth symbol,

$$p(s_k = S_x, s_{k+1} = E_i) = \begin{cases} \frac{1}{F(N_D+2M)}, & k < 2M \\ (1-\frac{1}{F})\frac{1}{F(N_D+2M)}, & k \geq 2M \end{cases}$$

for $i = 0, 1$. Once we have the joint probability, we can calculate the conditional probability $p(s_{k+1} = E_i)$ by dividing the joint probability by the marginal probability $p(s_k = S_x)$,

$$p(s_{k+1} = E_i|s_k = S_X) = \begin{cases} \frac{1}{2}\frac{1}{F(N_D+2M)-N_D-k}, & 0 \leq k < 2M \\ \frac{1}{2}\frac{1}{F(N_D+2M)-(k-2M)}, & 2M \leq k \leq N_D \end{cases}$$

for $i = 0, 1$. Then

$$p(s_{k+1} = S_X|s_k = S_X) = \begin{cases} 1 - \frac{1}{F(N_D+2M)-N_D-k}, & 0 \leq k < 2M \\ 1 - \frac{1}{F(N_D+2M)-(k-2M)}, & 2M \leq k \leq N_D. \end{cases}$$

REFERENCES

[1] S. Verdú, *Multiuser Detection*. Cambridge Univ. Press, 1998.

[2] U. Timor, "Improved decoding scheme for frequency-hopped multilevel FSK system," *Bell Syst. Tech. J.*, vol. 59, pp. 1839–1855, 1980.

[3] ——, "Multistage decoding of frequency-hopped FSK system," *Bell Syst. Tech. J.*, vol. 60, pp. 471–483, 1981.

[4] T. Mabuchi, R. Kohno, and H. Imai, "Multiuser detection scheme based on canceling cochannel interference for MFSK/FH-SSMA system," *IEEE J. Select. Areas Commun.*, vol. 12, no. 4, pp. 593–604, May 1994.

[5] U.-C. Fiebig, "Iterative interference cancellation for FFH/MFSK MA systems," *IEE Proc. Commun.*, vol. 143, no. 6, pp. 380–388, Dec. 1996.

[6] K. Halford and M. Brandt-Pearce, "Multistage multiuser detection for FHMA," *IEEE Trans. Commun.*, vol. 48, no. 9, pp. 1550–1562, Sept. 2000.

[7] K. Hamaguchi, L.-L. Yang, and L. Hanzo, "Multi-stage multi-user detection assisted fast-FH/MFSK," *IEE Electron. Lett.*, vol. 39, no. 4, pp. 399–400, 20 Feb 2003.

[8] N. Sharma, H. El Gamal, and E. Geraniotis, "Multiuser demodulation and iterative decoding for frequency-hopped networks," *IEEE Trans. Commun.*, vol. 49, no. 8, pp. 1437–1446, Aug. 2001.

[9] N. Sharma and E. Geraniotis, "Soft multiuser demodulation and iterative decoding for FH/SSMA with a block turbo code," *IEEE Trans. Commun.*, vol. 51, no. 9, pp. 1561–1570, Sept. 2003.

[10] P.-P. Liang and W. Stark, "Iterative multiuser detection for turbo-coded FHMA communications," *IEEE J. Select. Areas Commun.*, vol. 19, no. 9, pp. 1810–1819, Sept. 2001.

[11] J. Gass, J.H., P. Curry, and C. Langford, "An application of turbo trellis-coded modulation to tactical communications," in *Proc. 1999 IEEE Military Commun. Conf.*, vol. 1, Oct. 1999, pp. 530–533.

[12] M. B. Pursley and W. E. Stark, "Performance of Reed-Solomon coded frequency-hop spread-spectrum communications in partial-band interference," *IEEE Trans. Commun.*, vol. 33, pp. 767–774, Aug. 1985.

[13] C. W. Baum and M. B. Pursley, "Erasure insertion in frequency-hop communications with fading and partial-band interference," *IEEE Trans. Veh. Technol.*, vol. 46, pp. 949–956, Nov. 1997.

[14] T. G. Macdonald and M. B. Pursley, "Staggered interleaving and iterative errors-and-erasures decoding for frequency-hop packet radio," *IEEE Trans. Wireless Commun.*, vol. 1, pp. 92–98, Jan. 2003.

[15] J. H. Kang and W. E. Stark, "Iterative estimation and decoding for FH-SS with slow Rayleigh fading," *IEEE Trans. Commun.*, vol. 48, pp. 2014–2023, Dec. 2000.

[16] J.-W. Moon, J. M. Shea, and T. F. Wong, "Collaborative mitigation of partial-time jamming on nonfading channels," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 1371–1381, June 2006.

[17] J.-W. Moon, T. F. Wong, and J. M. Shea, "Pilot-assisted and blind joint data detection and channel estimation in partial-time jamming," *IEEE Trans. Commun.*, vol. 54, pp. 2092–2102, Nov. 2006.

[18] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.*, vol. 41, pp. 164–171, 1970.

[19] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257 –286, 1989.

[20] J. Wieselthier and A. Ephremides, "Discrimination against partially overlapping interference–its effect on throughput in frequency-hopped multiple access channels," *IEEE Trans. Commun.*, vol. 34, no. 2, pp. 136–142, Feb. 1986.

[21] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rates," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.

Xing Tan received the BS degree and MS degree from Shanghai Jiaotong University, China in 2003 and 2005. He is currently a PhD candidate in the Department of Electrical and Computer Engineering, University of Florida. His research interests include detection and estimation theory, signal processing for communications, and channel coding.

John M. Shea (S'92-M'99) received the B.S. (with highest honors) in Computer Engineering from Clemson University in 1993 and the M.S. and Ph. D. degrees in electrical engineering from Clemson University in 1995 and 1998, respectively.

Dr. Shea is currently an Associate Professor of electrical and computer engineering at the University of Florida. Prior to that, he was an Assistant Professor at the University of Florida from July 1999 to August 2005 and a post-doctoral research fellow at Clemson University from January 1999 to August 1999. He was a research assistant in the Wireless Communications Program at Clemson University from 1993 to 1998. He is currently engaged in research on wireless communications with emphasis on error-control coding, cross-layer protocol design, cooperative diversity techniques, and hybrid ARQ.

Dr. Shea was selected as a Finalist for the 2004 Eta Kappa Nu Outstanding Young Electrical Engineer Award. Dr. Shea was a National Science Foundation Fellow from 1994 to 1998. He received the Ellersick Award from the IEEE Communications Society for the Best Paper in the Unclassified Program of the IEEE Military Communications Conference in 1996. He was an Associate Editor for the *IEEE Transactions on Vehicular Technology* from 2002 to 2007.