Collaborative Mitigation of Partial-Time Jamming on Nonfading Channels

Jang-Wook Moon, John M. Shea, and Tan F. Wong

***Abstract*— We propose new collaborative reception techniques for use in the presence of a partial-time Gaussian jammer. Under the proposed techniques, a group of radios acts as a distributed antenna array by exchanging information that is then used to perform jamming mitigation. We propose two such jamming mitigation techniques that offer a tradeoff between performance and complexity. The results show that these techniques can allow communications in much more severe jamming conditions than when collaboration is not employed or when conventional collaboration techniques based on maximal-ratio combining are applied. Example scenarios with strong jamming show that three collaborating radios can reduce the frame error rate by more than two orders of magnitude over single-radio reception. In another scenario it is shown that a jammer must jam at least 75% of the transmitted symbols to produce an unacceptable frame error rate with three collaborating radios, but only 43% of the transmitted symbols if there is no collaboration.**

***Index Terms*— Partial-time jamming, jamming mitigation, collaborative communications, collaborative signal-processing.**

I. Introduction

PARTIAL-TIME jamming or interference can severely affect the performance of communication systems. This is especially true for military communication systems, which may experience hostile jamming. We consider communication in the presence of a partial-time Gaussian jammer. Such a system also approximates the effects of a partial-band jammer in a system that employs frequency-hopping spread spectrum. In systems that do not employ jamming mitigation techniques, the jammer can severely disrupt communications by concentrating its power over just a few symbols of a packet. If the jammer power becomes strong compared to the signal power, then it becomes impossible to communicate successfully. Therefore it is desirable to devise methods to combat or even cancel the detrimental effects. In the absence of multiple receive antennas, most previous research focused on using time diversity via repetition and/or error-control coding [1]–[14] to improve performance in the presence of jamming.

Manuscript received January 20, 2004; revised June 1, 2005 and June 8, 2005; accepted June 11, 2005. The associate editor coordinating the review of this paper and approving it for publication was L. Hanzo. This work was supported by the Office of Naval Research under Grant N00014-02-1-0554, by the National Science Foundation under Grant ANI-0220287, and by the DoD Multidisciplinary University Research Initiative administered by the Office of Naval Research under Grant N00014-00-1-0565.

J.-W. Moon was with the Wireless Networking Group, Department of Electrical Engineering, University of Florida and is now with VIA Telecom, Inc., San Diego, California 92121 (e-mail: jmoon@via-telecom.com).

J. M. Shea and T. F. Wong are with the Wireless Networking Group, Department of Electrical Engineering, University of Florida, Gainesville, Florida 32608 (email: {jshea,twong}@ece.ufl.edu).

Digital Object Identifier 10.1109/TWC.2006.04015.

If multiple receive antennas are available, the jamming signal can be reduced through nulling; see for example, [15]–[17]. However, in many communication systems, multiple antennas are not practical because of size constraints. Recently, collaborative communication techniques [18]–[29] have been proposed as a way to achieve spatial diversity without multiple antennas on any single radio. In these techniques, a group of radios cooperates by exchanging information to improve the performance of one radio or the whole group. In particular, in the collaborative reception techniques proposed in [25]–[27], a cluster of radios act as a distributed antenna array to achieve combining gain or diversity against fading.

Collaborative reception techniques can also be used to provide diversity against a jamming signal. We propose two such collaborative jamming mitigation techniques. In each of the proposed techniques, copies of the demodulator outputs for only the jammed symbols are exchanged among the collaborative radios. Iterative detection, estimation, and decoding algorithms are applied to achieve jamming mitigation. We use an error-correction code with maximum *a posteriori* (MAP) decoding, and the two collaborative decoding techniques differ in how the information from other radios is utilized and in their complexity. We also propose techniques for determining the set of jammed symbols, and we investigate performance for both known and unknown jamming parameters.

The collaborative jamming mitigation schemes that are proposed in this paper build upon previous research on cooperative communication and on jamming mitigation through error-control coding and adaptive antenna beamforming. Our approach utilizes error-control coding and extends the previous work [1]–[14], to take advantage of spatial diversity among radios in a network. The techniques we propose use the radios in a network to form a distributed antenna array. Unlike a conventional antenna array, the relative locations of the radios in a network are not carefully controlled, and thus the phase of the jamming signal with respect to the message signal will be random at each radio. These phases must be estimated, and the performance may be reduced relative to a traditional phased array. In addition, in conventional approaches [15]–[17], the information from multiple receiver front ends is exchanged over wires and there is no cost to exchange all of the information. In our approach, we try to reduce the required communication overhead by exchanging only the jammed symbols. Each radio must individually estimate the set of jammed symbols and the radios must reach a consensus before exchanging the received values for these symbols. Most previous work on cooperative communications [18]–[27] assumes independent noise at the cooperating radios,

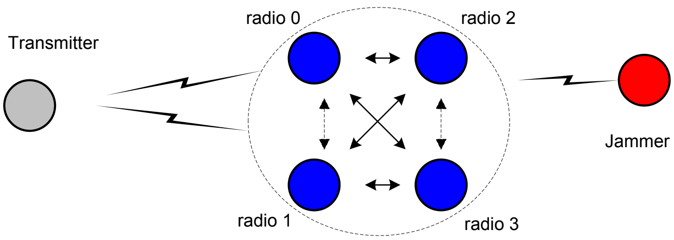


Fig. 1. One transmitter is communicating with multiple radios.

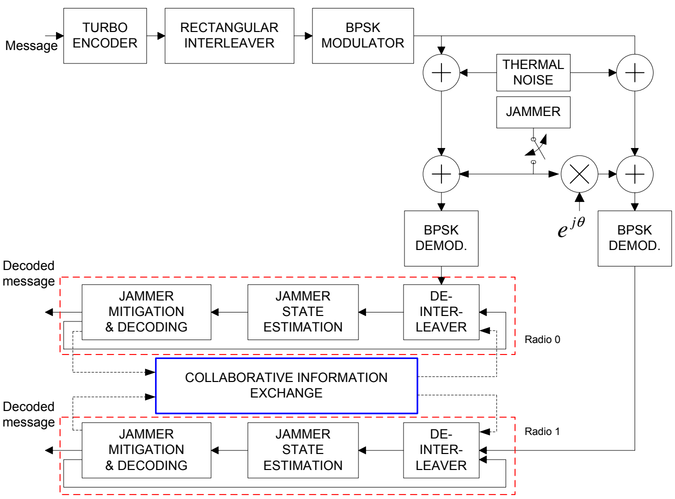


Fig. 2. The overall system model.

and thus uses combining schemes similar to maximal-ratio combining (MRC). In the presence of correlated jamming, we show that MRC does not provide adequate protection from the jamming, and thus we propose more-sophisticated signal processing techniques.

The rest of this paper is organized as follows. In Section II, the system model that is used in this paper is introduced. In Section III, two collaborative jamming mitigation techniques are proposed. In Section IV, the overhead and complexity of the two collaborative mitigation schemes are investigated. In Section V, we first explain how the jammer state can be estimated. Then, we propose a collaborative jamming detection algorithm for the radios to reach consensus on the set of jammed symbols. In Section VI, simulation results are presented, and conclusions are made in Section VII.

II. System Model

We consider a scenario in which a transmitter communicates with N radios in the presence of a partial-time jammer, as illustrated in Fig. 1. The overall system model for $N = 2$ is illustrated in Fig. 2. Binary phase-shift keying (BPSK) is used for modulation. A turbo code is used for channel coding, and log-MAP decoding is performed using the decoding algorithm proposed by Bahl, Cocke, Jelinek and Raviv [30], [31] (the BCJR algorithm). Throughout this paper, we assume that the turbo code has rate 1/3 and employs identical constituent codes. However, our schemes easily generalize to other codes. Let M denote the number of code symbols in a codeword. A rectangular interleaver is used to break up jamming bursts at the input to the decoder.

We model the jammer's transmissions using a two-state hidden Markov model (HMM). If at time k, the jammer is in state 0, then the kth bit is not jammed. If it is in state 1, then the bit is jammed, and the jamming signal is modeled as white Gaussian noise with power spectral density $N_J/(2\rho)$, where ρ is the probability that a bit is jammed. Let $E\{T_J\}$ be the expected amount of time (in terms of number of bits) spent in the jamming state before returning to the unjammed state. Then the transition probability from state i to state j is denoted by P_{ij} and can be determined from ρ and $E\{T_J\}$.

If the kth bit is jammed, the demodulator output for the kth received symbol at radio i can be modeled as

$$y_k^{(i)} = \sqrt{E_s} u_k + n_k^{(i)} + J_k e^{j\theta^{(i)}}, \quad (1)$$

where E_s is the symbol energy and u_k is the modulated code bit, which takes values from ± 1. Here, $n_k^{(i)}$ and J_k are the contributions from thermal noise and jamming, respectively, and are zero-mean, circular-symmetric Gaussian random variables. The variances of $n_k^{(i)}$ and J_k are given by N_0 and N_J/ρ, respectively. The total variance of the noise and jamming for state 1 is $\Upsilon \triangleq N_0 + N_J/\rho$. $\theta^{(i)}$ is the relative phase of the jamming signal at radio i with respect to the jamming signal at radio 0 and is uniformly distributed on $[0, 2\pi)$. Without loss of generality, we let $\theta^{(0)} = 0$. This does not imply that the jamming signal at radio 0 is in-phase with the information signal, as J_k itself is a complex random variable with phase uniformly distributed on $[0, 2\pi)$. We assume that the value of each $\theta^{(i)}$ is fixed for the duration of each packet. Thus, as shown in Fig. 2, the radios experience independent thermal noise but a phase-shifted version of a common interference from the jammer.

A group of radios acts as a distributed antenna array by exchanging the set of symbols that are estimated to be jammed and then applying collaborative jamming mitigation techniques. These exchanges may occur over the same channel as the original packet reception if the exchange will not interfere with transmission of additional packets, for instance, if the transmitter uses stop-and-wait ARQ. Alternatively, the collaborative exchanges may take place over an orthogonal communication channel, such as a wireless local area network. We note that in the general case, it is possible that the communications among the group of collaborating radios may also be jammed. However, in this paper, we assume that the receiving radios are clustered in a relatively small area and thus have sufficiently high signal-to-noise ratio to communicate in the presence of the jammer.

The radios use iterative estimation, information exchange, and decoding algorithms. First the radios individually estimate which symbols are jammed and then exchange messages to reach a consensus on the set of jammed symbols. Then the radios collaborate by exchanging information about the jammed symbols. Finally, each radio uses all of the received information to mitigate the jamming through one of two collaborative jamming mitigation techniques, which are described in the next section.

III. Collaborative Jamming Mitigation

The collaborative techniques used to mitigate jamming should be designed to take advantage of the highly correlated nature of the interference signal. In this section, we present two techniques of differing complexity for jamming mitigation. We begin by assuming that every radio has perfect channel state information (CSI), which includes knowledge of which bits are jammed as well as the statistics of the received jamming signal. In Section V, we consider the problem of estimating the jammer state and parameters of the jamming signal.

The jamming mitigation techniques that we consider fall in the class of collaborative decoding techniques [24]–[27]. As in previous work, we base our jamming mitigation techniques on the BCJR MAP decoding algorithm [30], [32]. Under the BCJR algorithm, the *a posteriori* likelihood ratios for the messages bits can be calculated as

$$\frac{p(u_k=+1|\mathbf{y})}{p(u_k=-1|\mathbf{y})}=\frac{\prod_{U^+}\alpha_{k-1}(s')\gamma_k(s',s)\beta_k(s)}{\prod_{U^-}\alpha_{k-1}(s')\gamma_k(s',s)\beta_k(s)}, \quad (2)$$

where U^+ and U^- denote the sets of branches that correspond to message bits with value $+1$ and -1, respectively. Here, $\alpha_{k-1}(s')=P(s',\mathbf{y}_1^{k-1})$ is the forward-looking state probability, $\beta_k(s)=P(\mathbf{y}_{k+1}^M|s)$ is the backward-looking state probability, and $\gamma_k(s',s)=P(s,y_k|s')$ is the branch probability. Here, $\mathbf{y}_a^b$ denotes the vector of received symbols between time indices a and b. It can be shown that $\alpha_{k-1}(s')$ and $\beta_k(s)$ can be computed recursively using $\gamma_k(s',s)$ [30], [32]. The rate $1/3$ turbo codes considered in this paper are constructed from two rate $1/2$ constituent convolutional codes that are decoded iteratively. In the absence of collaboration, the branch metric for each of these convolutional codes is given by

$$\gamma_k^{(i)}(s',s)=p(u_k)p(y_{s,k}^{(i)}|s',s)p(y_{p,k}^{(i)}|s',s), \quad (3)$$

where $y_{s,k}^{(i)}$ and $y_{p,k}^{(i)}$ are the matched filter outputs at the ith radio for the kth systematic and parity bit, respectively.

Since every radio receives copies of the same message in the presence of phase-rotated versions of the same jamming signal and independent thermal noise, it is desirable to utilize the received symbols from every radio to discriminate against the jamming signal and simultaneously achieve combining gain. The greatest performance can be achieved if all of the received symbols from every radio are employed in calculating the branch metrics. However, to do so would require exchanging the soft decisions for every received symbol. As the performance is often dominated by the jammed symbols, we assume that only the jammed symbols are exchanged. We propose two techniques to utilize the different copies of the jammed symbols in the computation of the branch metrics. The two techniques offer a tradeoff between complexity and communication overhead, which we investigate in Section IV.

A. Joint Density For Jammed Symbols

In this section, we explain the first jamming mitigation technique, which we call the *joint density* approach. In this approach, the conditional densities for the received symbols in the branch metric $\gamma_k^{(i)}(s',s)$ for the single receiver case, $p(y_{s,k}^{(i)}|s',s)$ and $p(y_{p,k}^{(i)}|s',s)$ are replaced by the joint densities for the symbols from every radio, $p(\mathbf{y_{s,k}}|s',s)$ and $p(\mathbf{y_{p,k}}|s',s)$, for each symbol that is jammed. Here, $\mathbf{y_{s,k}}=\left[y_{s,k}^{(0)},y_{s,k}^{(1)},\ldots,y_{s,k}^{(N-1)}\right]$, and $\mathbf{y_{p,k}}$ is defined similarly.

Consider the conditional joint density function for the set of received symbols representing a jammed information or parity bit given the transmitted symbol u_k. Under the assumption that the jamming parameters N_J and ρ and phases $\theta^{(i)}$ are known, the conditional joint density function is Gaussian with mean $\mathbf{m}_k$ and covariance matrix Σ_k. The mean of $y_k^{(i)}$ is μ_k where $\mu_k=\sqrt{E_s}u_k$. The variance of $y_k^{(i)}$ is N_0+N_J/ρ. Let $\mathbf{y}_k=[y_k^{(0)},\ y_k^{(1)},\ \ldots,\ y_k^{(N-1)}]^T$. Then the mean of $\mathbf{y}_k$ is $\mathbf{m}_k=[\mu_k,\ \mu_k,\ \ldots,\ \mu_k]^T$. The covariance matrix is $\Sigma_k=E[(\mathbf{y}_k-\mathbf{m}_k)(\mathbf{y}_k-\mathbf{m}_k)^H]$, where

$$\begin{aligned}\Sigma_k(l,m)&=\Sigma_k^*(m,l)\\&=E[(n_k^{(l)}+J_ke^{j\theta^{(l)}})(n_k^{(m)}+J_ke^{j\theta^{(m)}})^*]\\&=E[|J_k|^2e^{j(\theta^{(l)}-\theta^{(m)})}]\\&=\frac{N_J}{\rho}e^{j(\theta^{(l)}-\theta^{(m)})},\ l=0,\ldots,N-1\\&\qquad ,\ m=l+1,\ldots,N-1,\end{aligned}$$

where ^{H} denotes complex conjugate transpose and * denotes complex conjugate.

In practice, the radios do not know *a priori* whether a symbol is jammed. Neither do they know Σ_k, nor the parameters needed to compute Σ_k, which are N_J, ρ, and $\theta^{(i)}$. The set of jammed symbols, as well as N_J and ρ can be estimated using the Baum-Welch algorithm, which we discuss in Section V. Then Σ_k can be estimated directly as $\Sigma_k=\frac{1}{|\mathcal{J}|}\sum_{k\in\mathcal{J}}(\mathbf{y}_k-\mathbf{m}_k)(\mathbf{y}_k-\mathbf{m}_k)^H$, where $\mathcal{J}$ is the set of jammed symbols. The mean $\mathbf{m}_k$ is unknown because the correct symbol value u_k is unknown, so we calculate the mean using the *a posteriori* estimate for u_k from the previous iteration of the BCJR algorithm.

B. Jamming Signal Cancellation

Now we develop the second technique for jamming mitigation. We use an iterative estimation, cancellation, and decoding process. In this technique, the jamming signals $\mathbf{J}=\{J_k: k\in\mathcal{J}\}$ and the relative phases $\boldsymbol{\theta}=\left[\theta^{(0)},\theta^{(1)},\ldots,\theta^{(N-1)}\right]$ are estimated, and the phase-corrected jamming signal is subtracted from the received symbols in $\mathcal{J}$. Then the symbols from different radios are combined to form a new decision statistic

$$z_k=\sum_{i=0}^{N-1}\left[y_k^{(i)}-\hat{J}_ke^{j\hat{\theta}^{(i)}}\right],$$

where $\hat{J}_k$ and $\hat{\theta}^{(i)}$ are the estimates for J_k and $\theta^{(i)}$. The new decision statistics are used in place of the $y_k^{(i)}$s in the calculation of $\gamma_k(s',s)$ in the BCJR algorithm as

$$\gamma_k^{(i)}(s',s)=p(u_k)p(z_{s,k}^{(i)}|s',s)p(z_{p,k}^{(i)}|s',s). \quad (4)$$

We expect that this jamming cancellation scheme may suffer from error propagation that is common to interference cancellation schemes and thus will generally perform worse than the joint density scheme. However, we show in Section IV that the jamming cancellation scheme has a lower complexity than the joint density scheme.

We consider the joint maximum-likelihood (ML) estimate for the jamming signals $\mathbf{J}=\{J_k: k\in\mathcal{J}\}$ and the relative phases $\theta^{(i)}$. We derive the estimators for $\theta^{(i)}$ and $\mathbf{J}$ under the assumption that $\mathbf{u} = \{u_k^{(i)} : k \in \mathcal{J}, i = 0, 1, \ldots, N-1\}$ is known. As in the previous section, since $\mathbf{u}$ is unknown, we use the estimates from the previous iteration of the BCJR algorithm. Given $\mathbf{u}$, the ML estimator for $\boldsymbol{\theta}$ and $\mathbf{J}$ is

$$[\hat{\mathbf{J}}, \hat{\theta^{(1)}}, \hat{\theta^{(2)}}, \ldots] = \underset{[\mathbf{J}, \theta^{(1)}, \theta^{(2)}, \ldots]}{\operatorname{argmax}} \; p\left(\mathbf{y}^{(0)}, \ldots, \mathbf{y}^{(N-1)} | \mathbf{J}, \boldsymbol{\theta}, \mathbf{u}\right), \tag{5}$$

where $\mathbf{y}^{(i)}$ is a vector of the received symbols in $\mathcal{J}$ at radio i. Given $\boldsymbol{\theta}$, $\mathbf{J}$, and $\mathbf{u}$, the $\mathbf{y}^{(i)}$s are independent Gaussian random variables, and thus the natural logarithm of (5) can be written as

$$\mathrm{f} = C_2 - C_1 \sum_{k \in \mathcal{J}} \sum_{i=0}^{N-1} \left| y_k^{(i)} - \mu_k - J_k e^{j\theta^{(i)}} \right|^2, \tag{6}$$

where C_1 and C_2 are constants that do not have any effect on the maximization. Taking the gradient and setting to zero, it is easy to see that the joint ML estimates for J_k and $\theta^{(i)}$ must satisfy

$$\hat{J}_k = \frac{1}{N} \sum_{i=0}^{N-1} \left(y_k^{(i)} - \mu_k \right) e^{-j\theta^{(i)}}, \text{ and} \tag{7}$$

$$\hat{\theta}^{(i)} = \tan^{-1} \frac{\operatorname{Im} \sum_{k \in \mathcal{J}} \left(y_k^{(i)} - \mu_k \right) J_k^*}{\operatorname{Re} \sum_{k \in \mathcal{J}} \left(y_k^{(i)} - \mu_k \right) J_k^*}. \tag{8}$$

Thus, the ML estimate for $\theta^{(i)}$ must satisfy

$$\frac{\sin \theta^{(i)}}{\cos \theta^{(i)}} = \frac{\operatorname{Im} \sum_{k \in \mathcal{J}} \left(y_k^{(i)} - \mu_k \right) \sum_{n=0}^{N-1} \left(y_k^{(n)} - \mu_k \right)^* e^{j\theta^{(n)}}}{\operatorname{Re} \sum_{k \in \mathcal{J}} \left(y_k^{(i)} - \mu_k \right) \sum_{n=0}^{N-1} \left(y_k^{(n)} - \mu_k \right)^* e^{j\theta^{(n)}}}. \tag{9}$$

We can similarly expand (7).

Consider first (9). It can be shown that an equivalent set of N equations is given by

$$\mathbb{F}_i \triangleq \operatorname{Im} \left\{ \sum_{n=0}^{N-1} A_{i,n} e^{j\left(\theta^{(n)} - \theta^{(i)}\right)} \right\} = 0, \quad i = 0, 1, \ldots, N-1, \tag{10}$$

where $A_{m,n} = \sum_{k \in \mathcal{J}} (y_k^{(m)} - \mu_k)(y_k^{(n)} - \mu_k)^*$. For two radios, the ML solution is given by

$$\hat{\theta}_{ML}^{(1)} = \tan^{-1} \frac{\operatorname{Im} \sum_{k \in \mathcal{J}} \left(y_k^{(0)} - \mu_k \right)^* \left(y_k^{(1)} - \mu_k \right)}{\operatorname{Re} \sum_{k \in \mathcal{J}} \left(y_k^{(0)} - \mu_k \right)^* \left(y_k^{(1)} - \mu_k \right)} \tag{11}$$

and

$$\hat{J}_k = \frac{y_k^{(0)} - \mu_k}{2} + \frac{y_k^{(1)} - \mu_k}{2} \cdot \exp\left\{ j \tan^{-1} \frac{\operatorname{Im} \sum (y_k^{(0)} - \mu_k)(y_k^{(1)} - \mu_k)^*}{\operatorname{Re} \sum (y_k^{(0)} - \mu_k)(y_k^{(1)} - \mu_k)^*} \right\}. \tag{12}$$

Similarly, for any number of radios, a solution to $\mathbb{F}_0$ is given by $\theta^{(i)} = \angle A_{i,0} = \tan^{-1} A_{i,0}, \forall i$. This may not correspond to the ML estimate for $N > 2$. Since the ML estimator of $\theta^{(i)}$ does not admit a simple form for general N, in what follows, we use the estimate given by

$$\hat{\theta}^{(i)} = \angle A_{i,0} = \tan^{-1} \sum_{k \in \mathcal{J}} (y_k^{(i)} - \mu_k)(y_k^{(0)} - \mu_k)^*, \forall i. \tag{13}$$

The value of $\hat{J}_k$ is then estimated using (7) with $\theta^{(i)}$ replaced by its estimate, $\hat{\theta}^{(i)}$.

Note that the dependence of the estimators for each J_k on the values of $\hat{\theta}^{(i)}$, $0 < i \leq N-1$ may make $\hat{J}_k$ sensitive to the quality of $\hat{\theta}^{(i)}$. The estimates for $\hat{\theta}^{(i)}$ are typically much more accurate than for J_k. This is because more samples are typically used for $\hat{\theta}^{(i)}$ than for J_k. Each $\theta^{(i)}$ is constant over the entire block, and the average is over time. The J_ks change from symbol to symbol, and the estimator is averaged over the received values at different radios.

To calculate (7) and (13), every radio needs to broadcast the received values for the jammed symbols. This is the same information required for the joint density approach. This information needs to be exchanged only one time after the collaborators agree on the set of jammed symbols. However, the estimate (7) is very sensitive to errors in $\hat{\mu}_k$ from the output of the BCJR decoder. We consider the following approach to improving the estimate of $\hat{\mu}_k$ for the jamming cancellation technique. Each radio performs independent decoding and generates separate *a posteriori* log-likelihood ratios (LLRs) for each coded bit. For the jamming cancellation technique, these LLRs are exchanged along with the received symbols. The LLRs at different radios for the same bit will be correlated because of the shared jamming samples. However, we use a suboptimal combining technique to avoid the complexity of optimally combining the correlated LLRs. The LLRs from each radio are added together to generate a more-reliable LLR, which improves the probability of correct decoding. As these LLRs may change during the decoding, we investigate the effects of different exchange strategies in Section VI-A.

The estimated codeword may contain errors, but we can use the LLRs to estimate the reliability of the estimated code bits and adapt the jamming cancellation accordingly, thereby reducing problems from error propagation. The branch metric is computed by conditioning on whether the bit decision is correct and averaging over the two cases. To illustrate this, consider the case of $N = 2$. In what follows, we assume that $\theta^{(1)}$ is known or accurately estimated. We then have the following two cases:

1) The decision is correct. The probability that this happens can be approximated as a maximum of *a posteriori* probabilities (APPs), i.e. [33]

$$\text{Prob(correct decision for } k\text{th bit at radio } i | \, \mathbf{y}^{(i)}) \approx \max\left(\frac{e^{L^{(i)}(k)}}{e^{L^{(i)}(k)} + 1}, \frac{1}{e^{L^{(i)}(k)} + 1} \right), \tag{14}$$

where $\mathbf{y}^{(i)}$ is the total received vector at radio i and $L^{(i)}(k)$ is the LLR for the kth message bit at radio i. The reliability of $L^{(i)}(k)$ can be improved by adding all $L^{(i)}(k),\ 0 \leq i < N$ as previously explained. From (1) and (12), the estimated J_k is a random variable given by

$$\hat{J}_k = \frac{n_k^{(0)} + J_k + n_k^{(1)} e^{-j\theta^{(1)}} + J_k}{2} = J_k + \frac{n_k^{(0)} + n_k^{(1)} e^{-j\theta^{(1)}}}{2}.$$

After cancellation using these estimates, we get the new random variable $z_k^{(i)} = y_k^{(i)} - \hat{J}_k e^{j\theta^{(i)}}$, which is Gaussian distributed with mean $E\{z_k^{(i)}\} = \mu_k$ and variance $\mathrm{Var}\{z_k^{(i)}\} = N_0(N-1)/N$.

2) The decision for u_k is incorrect. For this case, $\hat{J}_k$ is given by

$$\hat{J}_k = \frac{2\mu_k + n_k^{(0)} + J_k + (2\mu_k + n_k^{(1)} + J_k e^{j\theta^{(1)}})e^{-j\theta^{(1)}}}{2}.$$

The random variable $z_k^{(i)} = y_k^{(i)} - \hat{J}_k e^{j\theta^{(i)}}$ is Gaussian with

$$E\{z_k^{(i)}\} = \mu_k - \frac{2\mu_k}{N}\sum_{\ell=0}^{N-1} e^{j(\theta^{(i)}-\theta^{(\ell)})},$$

and $\mathrm{Var}\{z_k^{(i)}\} = N_0(N-1)/N$.

After jamming cancellation, the $z_k^{(i)}$ are combined to improve the decoder performance. The $z_k^{(i)}$ are not independent, but in the interest of constraining the complexity of the algorithm, we use $z_k = \sum_{i=0}^{N-1} z_k^{(i)}$ as our new decision statistic. Then z_k and its reliability and moments are used to calculate the branch metrics in the BCJR algorithm. The terms of the form $p(z_k|s', s)$ in (4) are calculated as

$$p(z_k|s', s) = p(z_k|s', s, \text{correct decoding})P(k) + p(z_k|s', s, \text{incorrect decoding})[1 - P(k)], \quad (15)$$

where $P(k)$ is the probability of correctly decoding the kth bit, which we approximate by (14).

IV. Overhead and Complexity of Jamming Mitigation

The two proposed schemes offer a trade-off between overhead and complexity, which we briefly investigate in this section. We define the *average overhead* as the average number of bits of collaborative information that is transmitted by each radio for each symbol in the original packet. In Fig. 3 we provide a flowchart that illustrates the operation of the two jamming mitigation algorithms along with the information exchanges required. As the flowchart illustrates, the overhead comes from several sources. First, every radio broadcasts the indices for the set of symbols that are estimated to be jammed in order to reach consensus on the set of jammed bits. If L is the number of information bits in the block and R is the rate of the code, then $\lceil \log_2(L/R) \rceil$ is the number of bits required to index an arbitrary jammed symbol within the block. Let $\mathcal{J}_i$ denote the set of symbols that are estimated to be jammed at radio i, and the expected number of such symbols is $E[|\mathcal{J}_i|]$. The set of jammed symbols is highly compressible because of the Markovian nature of the jammer. In what follows we present analytical values for optimum compression and for compression based on sending the indices of the starting and stopping times of jamming bursts, both of which are considered in detail in [34]. The analytical results are based on exact knowledge of which symbols are jammed. For the simulation results, the radios do not know the set of jammed symbols, but select the set using the majority voting scheme described in Section V-C.

The second set of information that is exchanged consists of the complex received values for the consensus set of jammed symbols, $\mathcal{J}$. Let Q denote the number of bits needed to represent a real number. Then the number of bits required to exchange the received values for the jammed symbols is $N\,(2Q)\,E[|\mathcal{J}|]$. For the jamming cancellation technique, the *a posteriori* LLRs are also exchanged for the jammed symbols. In Fig. 3, we show only one exchange of these LLRs (which has the best performance), but we also considered schemes in which the LLRs are updated in several iterations. These are real values, and thus each time the LLRs are exchanged, a total of $N\,(Q)\,E[|\mathcal{J}|]$ bits are required.

The results in Table I show the average overhead required for the two schemes with two or three collaborating radios and four-bit quantization. For comparison, the overhead required for MRC is also shown. The results are based on blocks of 1000 information bits that are encoded with rate 1/3 symmetric turbo codes based on constituent convolutional codes with feedforward polynomial $1 + D^2$ and feedback polynomial $1 + D + D^2$. For all of the results, $\rho = 0.6$. For the simulation results, $E_b/N_J = -6$ dB and $E_b/N_0 = 6$ dB. The joint density scheme offers a significant overhead savings over MRC, especially for $N = 3$. The jamming signal cancellation technique requires more overhead, approximately equal to that of MRC. These comparisons will depend on the parameters of the jamming. For smaller ρ, the proposed schemes will require much less overhead than MRC, while for larger ρ, it may be more efficient to exchange all of the received symbols (as in MRC).

The complexity of the two approaches can be estimated by considering the number of multiplications required for each jammed symbol in one iteration of the BCJR decoder. Except where noted, all operations are assumed to be on complex numbers. For the joint density approach, the complexity is dominated by the matrix multiplications in the Gaussian density. The total number of complex multiplications is easily seen to be $N(N+1)$. For the jamming signal cancellation approach, estimating $\theta^{(i)}$ and J_k requires 1 and N multiplications per jammed symbol, respectively. Computing (15) requires 2 complex and 2 real multiplications per symbol. Thus, the equivalent number of complex multiplications required is approximately $N + 4$. Thus, although the jamming signal cancellation approach requires higher overhead than the joint density approach, it is equally computationally efficient for $N = 2$ and more computationally efficient than the joint density approach for $N > 2$.

Since the jamming signal estimates can be updated over multiple iterations, the jamming signal cancellation technique will require the number of computations estimated above times the number of iterations in which the jamming signal estimates are updated. By comparison, the joint density approach does not require the density to be updated in later iterations. Thus, the actual trade-off in complexity depends not only on N but also on the average number of iterations in which the jamming estimates are updated for the jamming signal cancellation technique. However, for large N, the $O(N^2)$ complexity of the joint density scheme will generally be much larger than the $O(N)$ complexity of the jamming cancellation scheme.

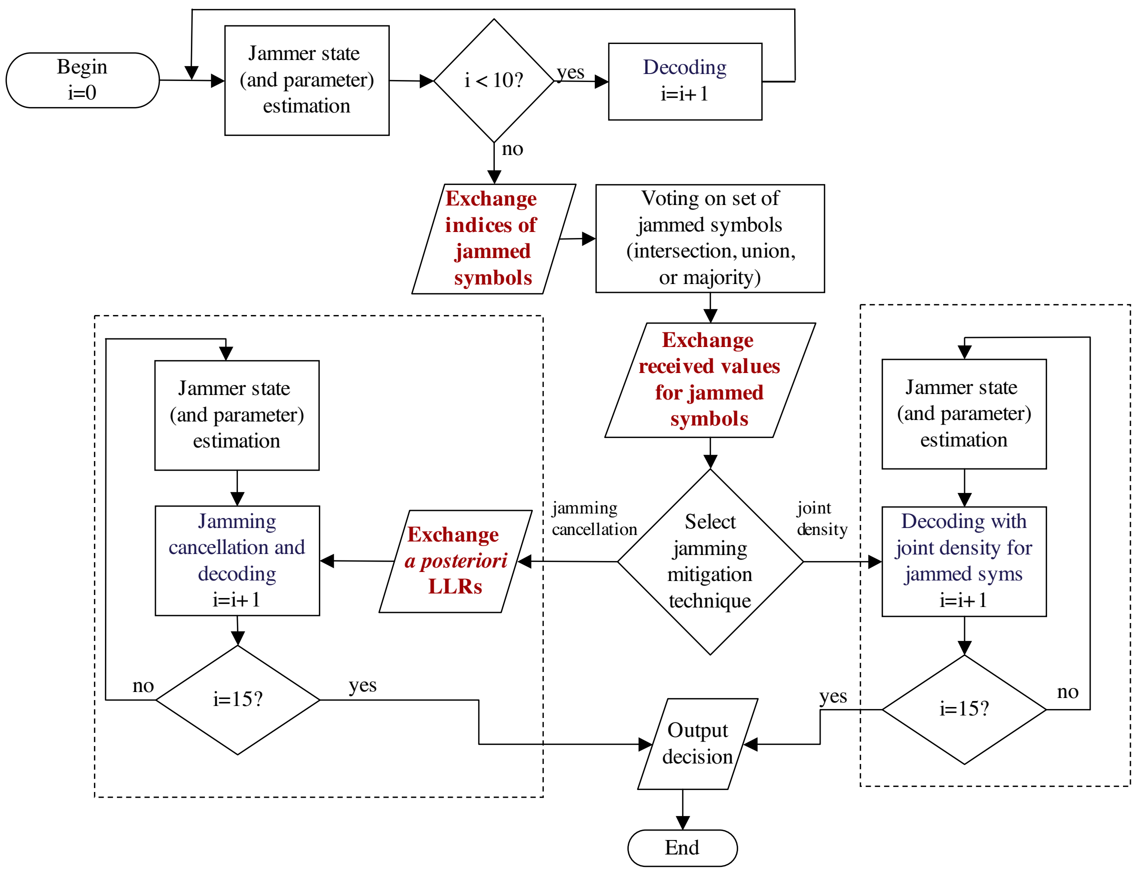


Fig. 3. Flowchart for collaborative jamming mitigation techniques.

TABLE I
COMPARISON OF AVERAGE OVERHEAD FOR FOUR-BIT QUANTIZATION.

N	Joint Density			Jamming Signal Cancellation			MRC
	Optimal-Analysis	Suboptimal-Analysis	Suboptimal-Simul.	Optimal-Analysis	Suboptimal-Analysis	Suboptimal-Simul.	Analysis
2	4.96	5.09	5.31	7.36	7.49	7.84	8.00
3	4.96	5.09	5.30	7.36	7.49	7.81	8.00

V. JAMMER STATE ESTIMATION

The two jamming mitigation techniques introduced in Section III require accurate knowledge of the set of jammed bits, as well as the variance of the jamming signal. We follow the general approach of iterative estimation and decoding that is described in [10], [11]. In this approach, the probability of jamming is determined for each received symbol by using another BCJR-type forward-backward algorithm. Information from the turbo decoder is used to improve the performance of the detection scheme, and vice versa. A significant difference between our work and previous work is that we require that the collaborating radios reach consensus on the set of jammed symbols. In addition, after information is exchanged, the radios have more information that can be used to refine the estimates for the probability that a symbol is jammed. We note that alternative structures are possible for combining jamming estimation and decoding. In [12], a unified approach is presented that can perform both procedures by passing messages on a graph. However, because of the channel interleaver, the structure of the graph is extremely complicated, so this technique is not practical except for very small block sizes.

A. MAP Algorithm for Jammer State Estimation

Consider first the case that the state transition probabilities of the HMM and the variance of the jammer are known. This may be a reasonable assumption if the jammer does not rapidly change its parameters because these parameters may then be accurately estimated over many packets. As the partial-time jammer is modeled using a two-state HMM, the APP that a symbol is jammed can be estimated using the BCJR algorithm. Let $\Gamma_k(z', z)$ be the branch metric for the transition from jammer state z' to state z at time k. Then

$$\Gamma_k(z', z) = P(z|z') \cdot \left[p\left(y_k|z', z, c_k = 0\right) P\left(c_k = 0\right) + p\left(y_k|z', z, c_k = 1\right) P\left(c_k = 1\right)\right], \quad (16)$$

where y_k and c_k represent the received symbol and coded bit corresponding to J_k, respectively. The probabilities $P(z|z')$ are the state-transition probabilities, which we have assumed are known, and $p(y_k|z', z, c_k)$ are Gaussian densities that depend on the transmitted symbol and the jamming state. Note that we assume that we do not have the *a priori* probabilities $P(c_k = 0)$ and $P(c_k = 1)$, and thus these probabilities are approximated by the APPs from the output of the turbo decoder, $P(c_k = 0|\mathbf{y})$ and $P(c_k = 1|\mathbf{y})$. This process results in an overall iterative approximation for the APP for the jamming state and message.

B. ML Estimation of Jamming Parameters

If the parameters of the HMM for the jammer are not known accurately, then they, too, must be estimated. The algorithm to do so is a generalization of the BCJR algorithm that is usually called the Baum-Welch algorithm [35], [31]. The first term in equation (16) is a transition probability of the HMM, which can be estimated by

$$\hat{P}(z|z') = \frac{\text{expected number of transitions from } z' \text{ to } z}{\text{expected number of transitions from } z'} = \frac{\sum_{k=1}^{L/R} \Lambda_{k-1}(z')\Gamma_k(z',z)\Delta_k(z)}{\sum_{k=1}^{L/R}\sum_{s=0}^{1} \Lambda_{k-1}(z')\Gamma_k(z',s)\Delta_{k-1}(z)}, \quad (17)$$

where L is the block length and R is the code rate. Here, $\Lambda_{k-1}(z')$ and $\Delta_k(z)$ are the forward and backward-looking state probabilities, which are defined as in Section III. Similarly, the estimator for the variance of the noise plus jamming in state 1 is

$$\hat{\Upsilon}_{ML} = \frac{\sum_{k=1}^{L/R} \Lambda_k(1)\Delta_k(1)\,|y_k - \mu_k|^2}{\sum_{k=1}^{L/R} \Lambda_k(1)\Delta_k(1)}. \quad (18)$$

Note that in the first iteration of the decoder, the parameters of the jammer may be unknown. Therefore, we initially set the transition probabilities for the jammer state to 0.5, and we set the jammer variance to twice that of the thermal noise. The results in Section VI-B indicate that these assumptions do not severely affect the performance because the estimates are updated after each iteration.

C. Collaborative Jamming Detection

Each radio initially estimates the jammer state using the BCJR algorithm independent of other radios. Thus the set of symbols which are estimated to be jammed may vary from radio to radio. To perform the jamming mitigation techniques introduced in Section III, we require that every radio share information about the same set of bits. To achieve this objective, we propose a scheme that allows the collaborating radios to come to consensus on the set of jammed symbols. Each radio independently estimates the jammer state for each received symbol. A hard decision on whether a symbol is jammed is made by comparing the *a posteriori* probability that the symbol is jammed to a threshold (0.5 for the results in this paper).

Each radio broadcasts the symbol indices for the set of jammed symbols. In practice, these indices can be highly compressed by taking advantage of the bursty nature of the jammer. Finally, consensus on the set of jammed symbols is reached by voting. A group of N collaborating radios will decide that a symbol is jammed if the number of radios that estimate that bit is jammed is greater than or equal to a threshold T. We consider three voting rules, which can be described in simple terms: 1) union of each radio's set ($T = 1$), 2) intersection of each radio's set ($T = N$), and 3) majority voting ($T = N/2$). The union and intersection rules represent the extreme (non-trivial) cases that minimize the probability of miss and probability of false alarm, respectively. The majority voting rule is a compromise between the two extremes. The performance of these algorithms is investigated in Section VI.

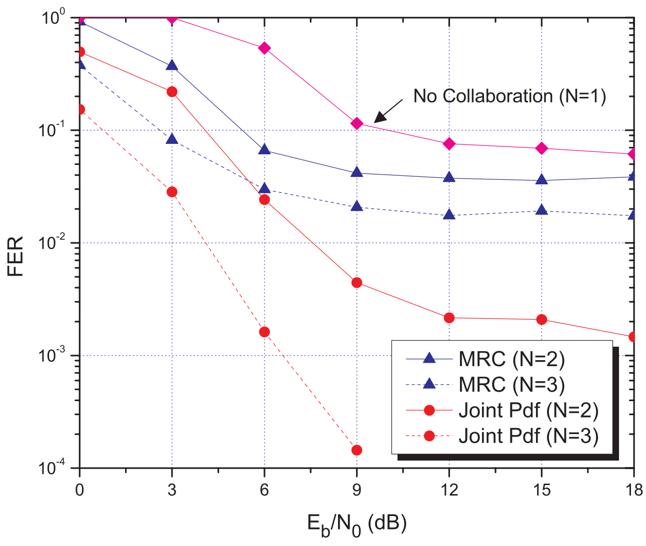


Fig. 4. Performance of the joint density scheme and MRC for perfect CSI ($E_b/N_J = -6$dB, $\rho = 0.6$, $E\{T_J\} = 50$).

VI. Simulation Results

For the simulation results presented in this paper, the turbo codes are constructed from identical, memory-two, recursive, systematic convolutional codes with feedforward polynomial $1 + D^2$ and feedback polynomial $1 + D + D^2$. The code rate is 1/3, the information block size is 1000, and a random interleaver is used. A total of 15 decoder iterations are performed. Except where otherwise stated, $E_b/N_J = -6$ dB, $\rho = 0.6$, and $E\{T_J\} = 50$. For the following graphs, we use the labels "Joint PDF" and "Cancel" to refer to the joint density and jamming signal cancellation schemes, respectively.

A. Collaborative Jamming Mitigation with Perfect CSI

We first consider the performance of the two proposed jamming mitigation algorithms for the case of perfect CSI in the sense that the radios know exactly which symbols are jammed and the statistics of the jamming signal. The results in Fig. 4 show the frame error rates (FERs) for the joint density scheme and for MRC with two or three collaborating radios. The results show that MRC is not an effective approach for mitigating the jamming signal because it does not take into account the correlated nature of the jamming signal among the radios. Even for $N = 3$, the FER has an error floor above 10^{-2}. The joint density scheme is better able to mitigate the jamming signal, achieving error rates close to 10^{-3} and 10^{-4} for $N = 2$ and $N = 3$, respectively.

We next consider the cancellation scheme because we wish to show the performance effects of exchanging the *a posteriori* LLR information from other radios. We present results for two different approaches. In the first approach, no LLRs

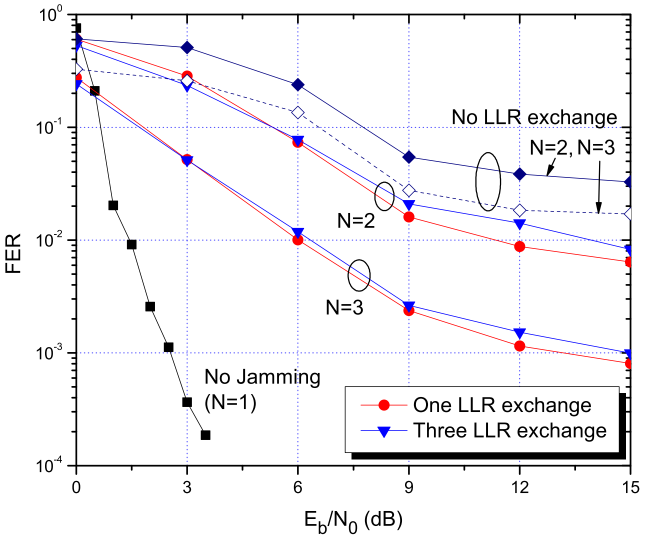


Fig. 5. Performance of the variations of the cancellation schemes for perfect CSI ($E_b/N_J = -6\text{dB}, \rho = 0.6, E\{T_J\} = 50$).

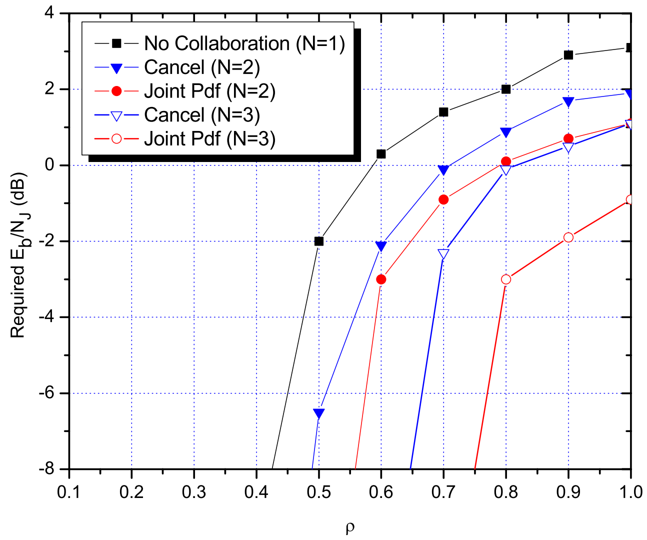


Fig. 7. Required E_b/N_J in dB vs. ρ ($E_b/N_0 = 6\text{dB}, E\{T_J\} = 50$).

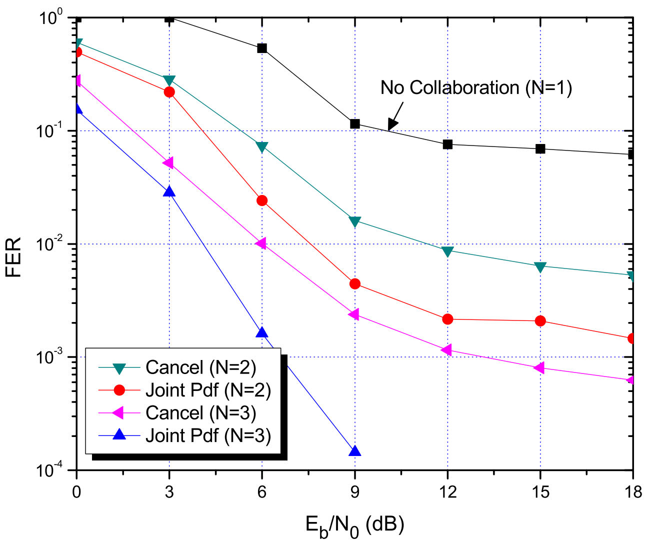


Fig. 6. Performance of the jamming mitigation schemes for perfect CSI ($E_b/N_J = -6\text{dB}, \rho = 0.6, E\{T_J\} = 50$).

are exchanged; that is, each radio uses its own *a posteriori* estimate of the jammed symbols to do the cancellation. In the second approach, the updated LLRs are exchanged after 4, 8, and 12 iterations. The performance for these variations of the jamming cancellation scheme is illustrated in Fig. 5. The results show that the performance degrades significantly if no *a posteriori* LLR information is exchanged. However, the results are relatively insensitive to whether the LLRs are updated in future iterations with a slight degradation in performance when the LLRs are exchanged three times. Thus, for all future results, we use only one LLR exchange.

A performance comparison between the two jamming mitigation schemes proposed in this paper is illustrated in Fig. 6. Consider the performance at $E_b/N_0 = 9$ dB. Without collaboration ($N = 1$), the FER is 10^{-1}. With the joint density scheme, the FER is less than 5×10^{-3} and 2×10^{-4} for $N = 2$ and $N = 3$, respectively. While the performance of the jamming cancellation scheme is significantly better than no collaboration, it is worse than the joint density scheme. For example, the performance of the jamming cancellation scheme for $N = 3$ is only slightly better than the joint density scheme with $N = 2$. As previously mentioned, this can be attributed to error propagation from incorrect bit decisions used in jamming cancellation.

As suggested in [3], the antijam capability of the jamming mitigation schemes can be measured by determining ρ^*, which is the value of ρ required to prevent the radio(s) from achieving an acceptable error probability. The higher the value of ρ^*, the more symbols that must be jammed in order to significantly degrade communications. In Fig. 7, the value of E_b/N_J that is required to achieve a frame error rate of 10^{-2} is shown as a function of ρ, the probability that a symbol is jammed. For these results, $E_b/N_0 = 6$ dB and perfect CSI is assumed. The results show that the jamming mitigation schemes are effective at increasing the value of ρ^*. For no collaboration $\rho^* \approx 0.43$. For $N = 2$, the jamming signal cancellation and joint density schemes increase ρ^* to 0.48 and 0.56, respectively. For $N = 3$, the value of ρ^* further increases to 0.64 for the jamming cancellation scheme and 0.75 for the joint density scheme. Thus, these schemes offer a significant advantage in partial-time jamming, with the joint density approach providing the best performance.

B. Collaborative Jamming Estimation and Mitigation

For the results presented up to this point, we have assumed that the radios know not only exactly which symbols are jammed but also the signal-to-noise plus interference ratio in the jamming state, $E_b/(N_0 + \rho^{-1}N_J)$. We first consider the performance of the voting schemes that allow the collaborating radios to determine a consensus set of jammed bits. A *miss* is defined to be the event in which jammed bits are incorrectly identified as being unjammed. Similarly, a *false alarm* is

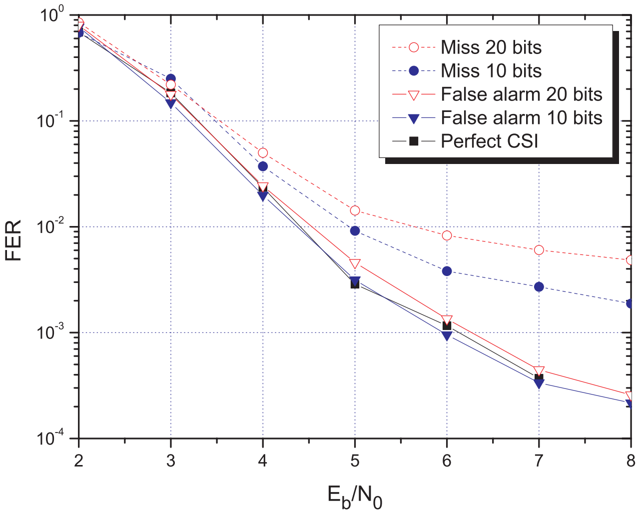


Fig. 8. Performance when there are misses for some jammed bits or false alarms for some unjammed bits ($N = 1$, block size=1000, $E_b/N_J = 0$ dB, $\rho = 0.4$, $E\{T_J\} = 50$).

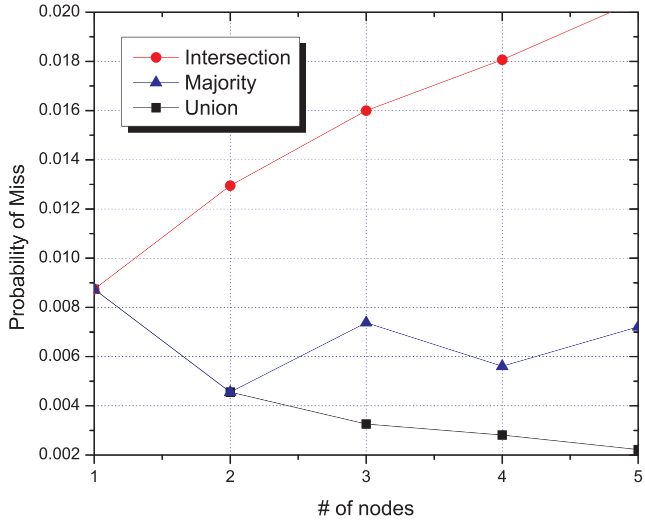


Fig. 9. Probability of miss vs. number of radios (block size 1000, $E_b/N_0 =$ 5 dB, $E_b/N_J = 0$ dB, $\rho = 0.4$, $E\{T_J\} = 50$).

defined to be the event in which unjammed bits are identified as jammed. The effects of false alarm and miss can be most easily investigated by evaluating the performance of these two phenomena separately. To do so, we degrade the performance of the system with perfect CSI by providing incorrect information about a random set of 10 or 20 bits. For example, to investigate the effects of missed symbols, 10 or 20 bits that were jammed were incorrectly indicated as unjammed at the input to the decoder. The results of this investigation are illustrated in Fig. 8. As seen, the effects of false alarm in terms of FER are negligible, but miss events cause a significant degradation in performance. However, the false alarm probability should be small because it results in additional overhead when information about the set of jammed symbols are exchanged among multiple radios. Thus, when error rate performance is most important, the probability of miss should be small, and if overhead is most important, the false alarm probability should be small. Typically, there is a trade-off between these two probabilities such that reducing one increases the other.

As the probability of miss has the greatest effect on performance, this performance measure is plotted in Fig. 9 for the three collaborative detection schemes described in Section V-C. The union scheme provides the lowest probability of miss, while the intersection scheme provides the highest probability of miss. The majority scheme provides a compromise that can achieve lower overhead than the union scheme while achieving performance close to that of the union scheme. Thus, the majority technique is used for the rest of the results in this paper.

Now consider the scenario in which the collaborating radios must estimate the set of jammed symbols and the jamming parameters in conjunction with the collaborative jamming mitigation techniques. The system model for this scenario is illustrated in Fig. 2. The operation and information exchange for this scenario are illustrated in Fig. 3. After ten iterations of jammer state and parameter estimation, the radios transmit

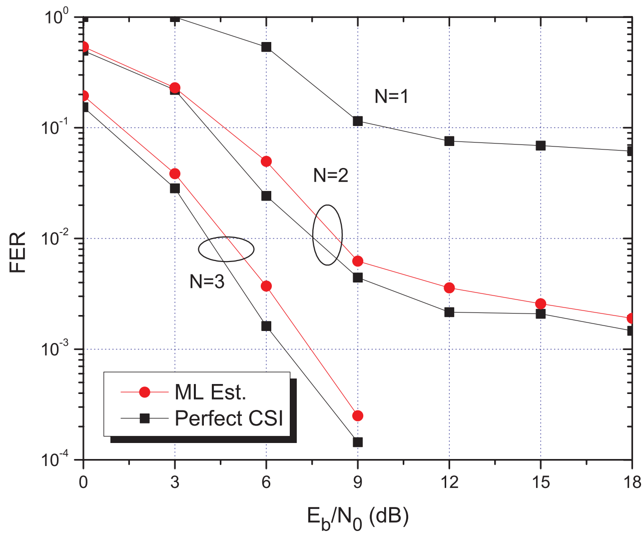


Fig. 10. Performance of the joint density scheme using estimated jammer state ($E_b/N_J = -6$dB, $\rho = 0.6$, $E\{T_J\} = 50$).

the indices of their jammed symbols and use the majority voting technique to determine the consensus set of jammed symbols. Following this, the received values and (for jamming cancellation only) the *a posteriori* LLRs for these symbols are exchanged. Then each radio iterates between two algorithms: 1) jamming mitigation and decoding, and 2) jammer state and parameter estimation.

The performance of the joint density scheme is shown in Fig. 10 for perfect CSI and for majority detection of jammed symbols. For perfect CSI, the radios know which symbols are jammed and the parameters of the jamming signal. For majority detection, the jamming parameters are estimated using the Baum-Welch algorithm. The performance for both two and three collaborators degrades from imperfect knowledge of which symbols were jammed and the jamming parameters. The degradation results in an increase in the FER by a factor

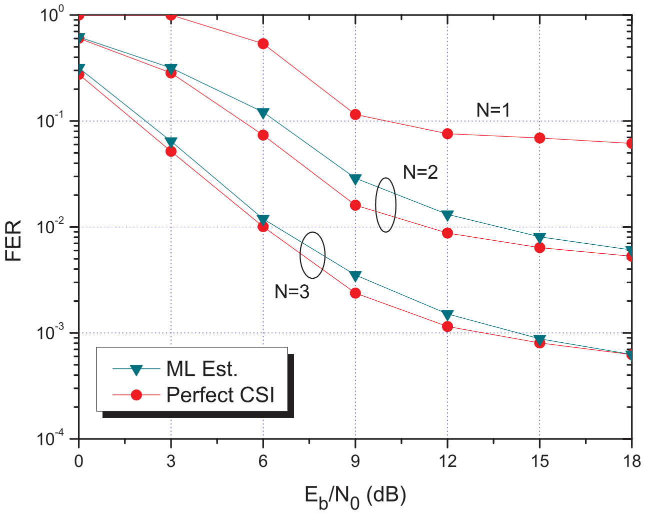


Fig. 11. Performance of the cancellation scheme using estimated jammer state. ($E_b/N_J = -6$dB$, \rho = 0.6, E\{T_J\} = 50$)

of approximately 2. However, the collaborative system still achieves most of the performance gain over reception without collaboration. The performance of the cancellation scheme is illustrated in Fig. 11 for perfect CSI and for majority detection of jammed symbols and ML parameter estimation. Again, the performance degrades from imperfect knowledge of which symbols are jammed and the jamming parameters.

VII. Conclusions

In this paper, two collaborative jamming mitigation techniques are investigated. In each technique, the branch metrics of the BCJR algorithm are modified to utilize information from other collaborating nodes. In the first technique, the joint density for the jammed symbols at different radios is applied directly to the calculation of the branch metrics in the BCJR algorithm. In the second technique, the jamming signals are estimated and subtracted from the received symbols. The results show that both techniques are effective at improving performance in the presence of a partial-time Gaussian jammer. The joint density technique provides better performance than the jamming cancellation technique, but the jamming cancellation technique requires fewer computations for more than two collaborating radios. We also presented approaches to estimate the probability of jamming for each symbol as well as the statistics of the jamming signal. We showed that such techniques can provide performance close to that of perfect CSI. Moreover, we presented techniques that allow a group of collaborating radios to achieve consensus on the set of jammed symbols. The symbols in this set are those that are exchanged and operated on by the jamming mitigation algorithms. Overall, the results show that collaborative jamming mitigation techniques can significantly improve performance in the presence of a partial-time jammer.

References

[1] A. J. Viterbi and I. M. Jacobs, *Advances in Communication Systems*, vol. 4, ch. Advances in coding and modulation for noncoherent channels affected by fading, partial band, and multiple access interference, pp. 279–308. New York: Academic, 1975.

[2] J. K. Omura and B. K. Levitt, "Coded error probability evaluation for antijam communication systems," *IEEE Trans. Commun.*, vol. 30, pp. 896–903, May 1982.

[3] 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.

[4] W. E. Stark, "Coding for frequency-hopped spread-spectrum communication with partial-band interference–part I: capacity and cutoff rate," *IEEE Trans. Commun.*, vol. 33, pp. 1036–1044, Oct. 1985.

[5] W. E. Stark, "Coding for frequency-hopped spread-spectrum communication with partial-band interference–part II: coded performance," *IEEE Trans. Commun.*, vol. 33, pp. 1045–1057, Oct. 1985.

[6] C. W. Baum and M. B. Pursley, "Bayesian methods for erasure insertion in frequency-hop communication systems with partial-band interference," *IEEE Trans. Commun.*, vol. 40, pp. 1231–1238, July 1992.

[7] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread Spectrum Communications Handbook*. McGraw-Hill, 2001.

[8] K. Cheun and W. E. Stark, "Performance of robust metrics with convolutional coding and diversity in FHSS systems under partial-band noise jamming," *IEEE Trans. Commun.*, vol. 41, pp. 200–209, Jan. 1993.

[9] C. D. Frank and M. B. Pursley, "Concatenated coding for frequency-hop spread-spectrum with partial-band interference," *IEEE Trans. Commun.*, vol. 44, pp. 377–387, Mar. 1996.

[10] J. H. Kang and W. E. Stark, "Turbo codes for coherent FH-SS with partial band interference," in *Proc. 1997 IEEE Military Commun. Conf.*, pp. 5–9.

[11] J. H. Kang and W. E. Stark, "Turbo codes for noncoherent FH-SS with partial band interference," *IEEE Trans. Commun.*, vol. 46, pp. 1451–1458, Nov. 1998.

[12] A. P. Worthen and W. E. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. Inform. Theory*, vol. 47, pp. 843–849, Feb. 2001.

[13] S. H. Schremmer, "Low density parity check codes in a frequency hopped communication system with partial-band interference," in *Proc. 2002 IEEE Military Commun. Conf.*, vol. 1, pp. 894–898.

[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] D. Torrieri and K. Bakhru, "An anticipative adaptive array for frequency-hopping communications," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 24, pp. 449–456, July 1988.

[16] O. Besson, P. Stoica, and Y. Kamiya, "Direction finding in the presence of an intermittent interference," *IEEE Trans. Signal Processing*, vol. 50, pp. 1554–1564, Jul. 2002.

[17] Y. Kamiya and O. Besson, "Interference rejection for frequency-hopping communication systems using a constant power algorithm," *IEEE Trans. Commun.*, vol. 51, pp. 627–633, Apr. 2003.

[18] A. Sendonaris, E. Erkip, and B. Aazhang, "Increasing uplink capacity via user cooperation diversity," in *Proc. 1998 IEEE Int. Symp. Inform. Theory*, p. 156.

[19] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity–part I: System description," *IEEE Trans. Commun*, vol. 51, pp. 1927–1938, Nov. 2003.

[20] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity–part II: Implementation aspects and performance analysis," *IEEE Trans. Commun*, vol. 51, pp. 1939–1948, Nov. 2003.

[21] J. N. Laneman, G. W. Wornell, and D. N. C. Tse, "An efficient protocol for realizing cooperative diversity in wireless networks," in *Proc. 2001 IEEE Int. Symp. Information Theory*, p. 294.

[22] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inform. Theory*, vol. 50, pp. 3062–3080, Dec. 2004.

[23] J. N. Laneman and G. W. Wornell, "Distributed space-time coded protocols for exploiting cooperative diversity in wireless network," *IEEE. Trans. Inform. Theory*, vol. 49, pp. 2415–2425, Oct. 2003.

[24] T. F. Wong, X. Li, and J. M. Shea, "Iterative decoding in a two-node distributed array," in *Proc. 2002 IEEE Military Communications Conference (MILCOM)*, vol. 2, pp. 1320–1324.

[25] T. F. Wong, X. Li, and J. M. Shea, "Distributed decoding of rectangular parity-check code," *IEE Electron. Lett.*, vol. 38, pp. 1364–1365, Oct. 2002.

[26] A. Avudainayagam, J. M. Shea, T. F. Wong, and X. Li, "Reliability exchange schemes for iterative packet combining in distributed arrays," in *Proc. 2003 Wireless Communications and Networking Conference*, vol. 1, pp. 832–837.

[27] A. Avudainayagam, J. M. Shea, and T. F. Wong, "Cooperative diversity through reliability filling," in *Proc 2003 Allerton Conf. on Commun., Control, and Computing*, pp. 1586–1587.

[28] R. Pabst, B. H. Walke, P. Herhold, H. Yanikomeroglu, S. Mukherjee, H. Viswanathan, M. Lot, W. Zirwas, D. D. Falconer, and G. Fettweis, “Relay-based deployment concepts for wireless and mobile broadband radio,” *IEEE Commun. Mag.*, vol. 42, pp. 80–89, Sept. 2004.
[29] M. Dohler and H. Aghvami, “Information outage probability of distributed STBCs over Nakagami fading channels,” *IEEE Commun. Lett.*, vol. 8, pp. 437–439, July 2004.
[30] 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. 20, pp. 284–287, Mar. 1974.
[31] P. L. McAdam, L. R. Welch, and C. L. Weber, “MAP bit decoding of convolutional codes,” in *Proc. 1972 Int. Symp. Inform. Theory*, p. 91
[32] W. E. Ryan, “Concatenated convolutional codes and iterative decoding,” in *Encyclopedia of Telecommunications* (J. G. Proakis, ed.), John Wiley & Sons, 2003.
[33] S. Benedetto and G. Montorsi, “Unveiling turbo codes: Some results on parallel concatenated coding schemes,” *IEEE Trans. Inform. Theory*, vol. 42, pp. 409–428, Mar. 1996.
[34] A. Roongta, J.-W. Moon, and J. M. Shea, “Reliability-based hybrid ARQ as an adaptive response to jamming,” *IEEE J. Select. Areas. Commun.*, vol. 23, pp. 1045–1055, May 2005.
[35] L. E. Baum and T. Petrie, “Statistical inference for probabilistic functions of finite state Markov chains,” *Ann. Math. Stat.*, vol. 37, pp. 1554–1563, 1966.

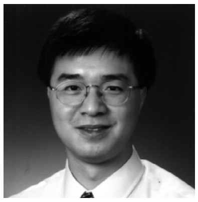
Jang-Wook Moon (S’03) received the B.S in electrical engineering from Sung-Kyun-Kwan University in Korea in 1999 and the M.S. in electrical and computer engineering from Yonsei University in Korea in 2001. He received the Ph. D. in electrical and computer engineering from the University of Florida in August 2005. He is currently employed as a Senior Systems Engineer at VIA Telecom, Inc.

His research interest is in wireless communications with emphasis on CDMA, turbo coding and iterative decoding, anti-jamming communication techniques, collaborative communications, and ad hoc networks.

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 in 1996. He is an Associate Editor for the *IEEE Transactions on Vehicular Technology*.



Tan F. Wong received the B.Sc. degree (1st class honors) in electronic engineering from the Chinese University of Hong Kong in 1991, and the M.S.E.E. and Ph. D. degrees in electrical engineering from Purdue University in 1992 and 1997, respectively. He was a research engineer working on the high speed wireless networks project in the Department of Electronics at Macquarie University, Sydney, Australia. He also served as a post-doctoral research associate in the School of Electrical and Computer Engineering at Purdue University. Since August 1998 he has been with the University of Florida, where he is currently an associate professor of electrical and computer engineering. He serves as Editor for Wideband and Multiple Access Wireless Systems for the *IEEE Transactions on Communications* and as the Editor for the *IEEE Transactions on Vehicular Technology*.