Reliability-Based Hybrid ARQ as an Adaptive Response to Jamming

Abhinav Roongta, *Student Member, IEEE*, Jang-Wook Moon, *Student Member, IEEE*, and John M. Shea, *Member, IEEE*

***Abstract*—We consider the use of selective-retransmission automatic repeat request (ARQ) protocols to improve communication performance on partial-time jamming channels. The schemes that we propose are based on reliability-based hybrid ARQ (RB-HARQ), in which the set of bits to be retransmitted is adaptively selected based on the estimated *a posteriori* probabilities at the output of a soft-input, soft-output decoder. For channels with bursty partial-time jamming, the RB-HARQ techniques are particularly appropriate because these techniques can identify which symbols need to be retransmitted, and the bursty nature of the jamming reduces the overhead required to relay this information to the source. We extend our reliability measure to incorporate not only *a posteriori* probability information but also estimates of the probability that a bit was jammed. We compare the performance of the proposed scheme with that of conventional approaches, in which a predetermined set of bits is retransmitted in response to a packet failure. The results show that RB-HARQ schemes can achieve higher throughput than these conventional approaches in several scenarios.**

***Index Terms*—Automatic repeat request (ARQ), hybrid ARQ, jamming mitigation, maximum *a posteriori* (MAP) decoding, soft-input–soft-output decoding.**

I. Introduction

HOSTILE jamming can severely disrupt wireless communications. The typical responses to such disruptions are retransmissions through automatic repeat request (ARQ), ARQ with adaptation of the signaling parameters [1]–[4], and adaptation in the network layer [5]–[9]. Most of the previous work identifies that adaptation is the key to responding to jamming. However, in each of these works, traditional ARQ is assumed. Although traditional ARQ is adaptive in the sense that retransmissions only occur when a packet is in error, it is nonadaptive in the sense that the response to a packet error is fixed: the entire packet should be retransmitted. Even if hybrid-ARQ is used, the response adapts neither to the reliability of the received packet nor to the set of symbols that was jammed. The performance of Type-I hybrid ARQ protocols in a slotted direct-sequence code-division multiple-access network operating in a hostile jamming environment is examined in [10]. The effect of jamming on the throughput of a HARQ protocol is also studied in [11] and [12].

In [13], a reliability-based hybrid ARQ (RB-HARQ) algorithm is proposed that is truly adaptive. In RB-HARQ, soft-input–soft-output decoders are used to identify which bits in a received packet are unreliable, and retransmissions are requested for only those unreliable bits. By requesting information for the unreliable bits, the performance of the decoder can improve more quickly than if a fixed HARQ scheme is used. The performance of RB-HARQ using turbo codes and convolutional codes over additive white Gaussian noise (AWGN) channels is shown in [14] and [15], respectively. Another RB-HARQ scheme that uses received packet reliability to optimize throughput over static and time-varying channels was independently proposed in [16]. All of the previous work on RB-HARQ [13]–[17] uses the magnitude of the log *a posteriori* probability (log-APP) ratio computed by the maximum *a posteriori* (MAP) [18] algorithm to identify the unreliable bits.

In this paper, we propose RB-HARQ schemes for use in hostile jamming environments and evaluate the performance of these schemes. In all of the previous work in which ARQ is evaluated in the presence of jamming [1]–[12], the set of bits that are retransmitted is not adapted to the set of bits that are likely to be jammed or in error. The work presented in this paper is unique in this sense. The proposed schemes use iterative MAP algorithms to estimate the probability that a bit is jammed or in error, and the retransmission is adapted to this set. Iterative estimation of jamming probability and jamming parameters is considered in [19] and [20]. Other related work includes iterative estimation of the fading levels for frequency-hopped spread-spectrum networks using a MAP algorithm [21]. The MAP algorithm was used to solve the problem of jointly demodulating and decoding a serially concatenated system in the absence of channel side information (CSI) in [22].

The main contribution of this paper is to develop RB-HARQ schemes that improve communication performance in hostile jamming environments by taking advantage of decoder estimates of bit reliabilities and jamming probabilities. The proposed schemes do the following:

- use the log-APPs of the information bits and/or the log-APPs of the jammer states to identify bits that are unreliable;
- use iterative MAP algorithms to estimate the probability that each bit is jammed, as well as the reliability of each bit;

Manuscript received April 15, 2004; revised January 24, 2005. This work was supported in part by Department of Defense (DoD) Multidisciplinary University Research Initiative administered by the Office of Naval Research under Grant N00014-00-1-0565 and in part by the National Science Foundation under Grant ANI-0220287. This paper was presented in part at the 2004 IEEE Military Communications Conference, Monterey, CA.

The authors are with the Wireless Information Networking Group, Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611-6130 USA (e-mail: jshea@ece.ufl.edu).

Digital Object Identifier 10.1109/JSAC.2005.845402

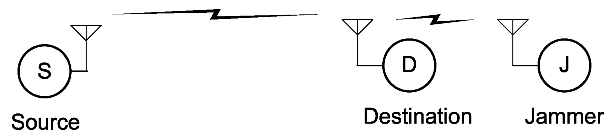


Fig. 1. Communication scenario.

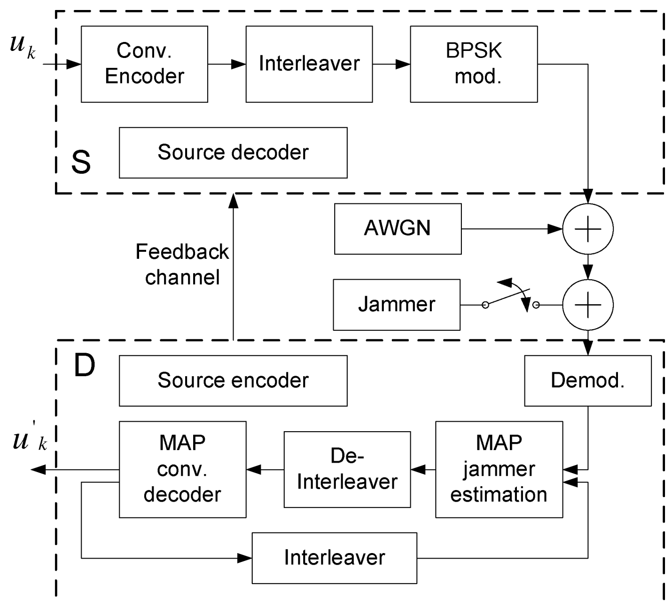


Fig. 2. System model.

- adapt the retransmission based on the output of these MAP algorithms;
- use optimal run-length arithmetic coding or a suboptimal but less-complex source coding to compress the retransmission-request packet.

We provide a performance comparison of the proposed RB-HARQ schemes to conventional HARQ approaches, including Type-I HARQ in which the entire packet is retransmitted and an incremental-redundancy HARQ in which a predetermined set of bits is retransmitted.

II. System Model

Consider the communication scenario shown in Fig. 1 in which a source $\mathbf{S}$ is communicating with a destination $\mathbf{D}$ in the presence of a partial-time jammer $\mathbf{J}$. We consider an asymmetric situation in which $\mathbf{D}$ and $\mathbf{S}$ experience different levels of jamming. In particular, we focus on the scenario illustrated in Fig. 1 in which $\mathbf{D}$ is experiencing high jamming levels compared with $\mathbf{S}$.

The system model for the above communication scenario is shown in Fig. 2. We consider packetized communication in which packets at $\mathbf{S}$ are encoded using a convolutional code for transmission to $\mathbf{D}$. Code symbols are modulated using binary phase-shift keying (BPSK) and received in the presence of white Gaussian thermal noise and time-varying jamming. The jammer is modeled using a discrete-time two-state Markov model as shown in Fig. 3. If at time k the jammer is in state 0, then the code symbol transmitted at time k is not jammed.

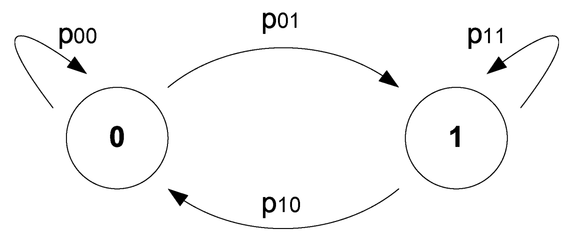


Fig. 3. Two-state Markov model for jammer.

State 1 indicates that the jammer is on and the code symbol is jammed. The proportion of time for which the jammer is active is specified as ρ, and $E\{T_J\}$ represents the expected value of the time (in terms of number of code symbols) spent in the jamming state before returning to the unjammed state. The four transition probabilities p_{ij}, where $i, j \in \{0, 1\}$, shown in Fig. 3 can be calculated from ρ and $E\{T_J\}$.

The power spectral density (PSD) of the thermal noise is $N_0/2$. The jamming PSD is $N_j/2$. However, as the jammer is only active for proportion ρ of all time, the effective jamming PSD when the jammer is active is $N_j/(2\rho)$. Thus, the total PSD of the noise (thermal noise and jammer noise) in state 1 is $\Sigma \triangleq N_0/2 + N_j/(2\rho)$. Let E_s denote the energy per modulation symbol. Matched-filter reception is assumed. Then, if the jammer is in state 1 at time k, the received symbol can be modeled by

$$y_k = c_k\sqrt{E_s} + n_k + J_k \tag{1}$$

where c_k is the transmitted code symbol, which takes values from ± 1. Here, n_k represents the contribution from thermal noise and is a zero-mean Gaussian random variable with variance $\sigma_0^2 = N_0/2$. The jamming is modeled as a Gaussian random variable J_k that has zero mean and variance $N_j/(2\rho)$. Thus, the total variance of the noise plus jamming is given by $\sigma_1^2 = (N_0 + \rho^{-1}N_j)/(2)$.

A. MAP Estimation Algorithms

The destination radio $\mathbf{D}$ uses MAP algorithms to estimate the jammer state for each received symbol and to decode the received packet. The use of the channel interleaver prevents the application of a single MAP algorithm to a hyper-trellis containing the states of the convolutional code and the jammer. Therefore, we consider two MAP algorithms connected in a feedback loop, as shown in Fig. 2. The estimation of the jammer state and impact on MAP decoding is considered briefly in Section IV and in detail in [19] and [20]. We provide a brief review of these algorithms and their interaction, and then describe how they are impacted by the ARQ transmissions.

Consider first the MAP algorithm for estimating the jammer state. In the absence of any CSI, the parameters of the jamming signal (ρ, $E\{T_J\}$ and $N_j/2$) can also be estimated using the Baum–Welch algorithm (cf. [19] and [20]). We briefly consider the case where the parameters of the jamming signal are known, as it offers some insight into the processing used in this paper. At each time k, the destination $\mathbf{D}$ computes the log-APP ratio for the jammer state given the received codeword $\mathbf{y}$, which is given by

$$L(j_k) = \log\left(\frac{P(j_k = 1|\mathbf{y})}{P(j_k = 0|\mathbf{y})}\right) \quad (2)$$

where $\mathbf{y}$ is the received codeword in noise and $j_k \in \{0, 1\}$ denotes the jammer state at time instant k. This calculation is performed using the BCJR algorithm [18] operating on the received symbols in the order in which they are transmitted. The branch metric connecting jammer state z' to jammer state z at time k is given by

$$\begin{aligned}\Gamma_k(z', z) &= p(z, y_k|z') = P(z|z') \cdot p(y_k|z', z) \\ &= P(z|z') \cdot [p(y_k|z', z, c_k = +1)P(c_k = +1|\mathbf{y}) \\ &\quad + p(y_k|z', z, c_k = -1)P(c_k = -1|\mathbf{y})] \quad (3)\end{aligned}$$

where y_k and c_k represent the received and transmitted code symbols, respectively, at time k. Note that $P(z|z')$ corresponds to one of the four transition probabilities shown in Fig. 3. The forward- and backward-looking state probabilities are determined in the usual way from the branch metrics. The probabilities $P(c_k = +1|\mathbf{y})$ and $P(c_k = -1|\mathbf{y})$ are set to 0.5 in the first iteration, and are updated in later iterations according to the *a posteriori* estimates produced by the MAP algorithm for decoding the message.

The BCJR algorithm for the message computes the log-APP ratio for each information bit u_k given $\mathbf{y}$

$$L(u_k) = \log\left(\frac{P(u_k = +1|\mathbf{y})}{P(u_k = -1|\mathbf{y})}\right). \quad (4)$$

Note that this BCJR algorithm operates in the order of the original code symbols before interleaving (i.e., in the order of the deinterleaved received symbols). We assume the use of a rate 1/2 convolutional code. Then, the metric for the branch connecting state s' to state s is

$$\gamma_k(s', s) = P(s|s') \cdot p\left(y_k^{(1)}|s', s\right) \cdot p\left(y_k^{(2)}|s', s\right) \quad (5)$$

where $y_k^{(1)}$ and $y_k^{(2)}$ are the matched-filter outputs for the two code symbols corresponding to the kth message bit, u_k. Here, $P(s|s') = P(u_k)$ is the *a priori* probability of information bit u_k, which is taken to be 0.5. Note that $p(y_k^{(1)}|s', s)$ and $p(y_k^{(2)}|s', s)$ depend on whether the symbol is jammed. Let $j_k^{(1)}$ and $j_k^{(2)}$ be the states of the jammer for the received code symbols $y_k^{(1)}$ and $y_k^{(2)}$, respectively, where $j_k^{(i)} = 1$ if the symbol is jammed and $j_k^{(i)} = 0$, otherwise. Then

$$\begin{aligned}p\left(y_k^{(i)}|s', s\right) &\approx p\left(y_k^{(i)}|s', s, j_k^{(i)} = 0\right) P\left(j_k^{(i)} = 0|\mathbf{y}\right) \\ &\quad + p\left(y_k^{(i)}|s', s, j_k^{(i)} = 1\right) P\left(j_k^{(i)} = 1|\mathbf{y}\right) \quad (6)\end{aligned}$$

where we are approximating the probability of jamming as independent from symbol to symbol, although this will not be true for a finite interleaver. As explained previously, $P(j_k^{(i)} = 0|\mathbf{y})$ and $P(j_k^{(i)} = 1|\mathbf{y})$ are estimated using the MAP algorithm for the jammer state.

If the received packet is in error after decoding, $\mathbf{D}$ sends a retransmission-request packet to $\mathbf{S}$ through the feedback channel. In this paper, we assume perfect packet error detection and an error-free feedback channel. The retransmission-request packet contains information about the set of bits that are estimated to be unreliable. The set of unreliable bits is identified using the log-APPs for the jammer states and information bits, which are computed using (2) and (4), respectively. The source encoder at $\mathbf{D}$ is used to compress the retransmission request packet. $\mathbf{S}$ decodes the retransmission-request packet and then retransmits the requested set of code symbols. Noisy (and possibly jammed) versions of the retransmitted code symbols are received at the receiver and combined in an optimal manner with the previously received copies as follows. Let $y_{k,1}$ and $y_{k,2}$ be the two received copies of the symbol c_k after the first and second transmission, respectively. The received log-likelihood ratio (LLR) for symbol c_k is given by

$$L(y_k) = \frac{2}{\sigma_{k,1}^2} y_{k,1} + \frac{2}{\sigma_{k,2}^2} y_{k,2} \quad (7)$$

if the jamming state is known exactly. Here, $\sigma_{k,i}^2$ is the variance of the noise plus jamming for the ith received copy of c_k. If the jamming state is not known exactly, then

$$L(y_k) = \log\left[\frac{p(y_{k,1}|c_k = +1)p(y_{k,2}|c_k = +1)}{p(y_{k,1}|c_k = -1)p(y_{k,2}|c_k = -1)}\right] \quad (8)$$

where each of the four terms is averaged over the two possible jamming states as follows:

$$\begin{aligned}p(y_{k,i}|c_k) &= p(y_{k,i}|c_k, j_k = 0)p(j_k = 0) \\ &\quad + p(y_{k,i}|c_k, j_k = 1)p(j_k = 1) \quad (9)\end{aligned}$$

where $p(j_k)$ is approximated by $p(j_k|\mathbf{y})$.

III. Reliability-Based Hybrid ARQ Schemes

Consider the packet error rate for coded communication in the presence of a partial-time jammer. Let T be the total number of information bits plus tail bits encoded with a rate k/n convolutional code. Then, for soft-decision, maximum-likelihood (ML) decoding, an upper bound on the packet error probability is given by [23], [24]

$$P_e \leq \min\left\{1, T \sum_{d=d_{\text{free}}}^{\left(\frac{n}{k}\right)T} A_d P_d\right\} \quad (10)$$

where A_d is the number of error events of weight d and P_d is the pairwise error probability (PEP) for two codewords separated by Hamming distance d. Consider the performance of a system that does not employ ARQ. Assume that an ideal interleaver is used in which the jamming symbols at the input to the decoder experience independent jamming. Then, the PEP is given by

$$P_d = \sum_{d_1=0}^{d} Q\left(\sqrt{E_s\left(\frac{d_1}{\sigma_1^2} + \frac{d - d_1}{\sigma_0^2}\right)}\right) \cdot \binom{d}{d_1} \rho^{d_1}(1-\rho)^{d-d_1}. \quad (11)$$

Here, d_1 is the number of symbols that are jammed out of the total d symbols that make up the error event. As in Section II, $\sigma_0^2 = N_0/2$, and $\sigma_1^2 = (N_0 + \rho^{-1}N_j)/2$.

An upper bound and good approximation for (11) is [25]

$$P_d \leq Q\left(\sqrt{E_s\frac{d}{\sigma_0^2}}\right)\sum_{d_1=0}^{d}\exp\left[-E_s d_1\left(\frac{1}{2\sigma_1^2}-\frac{1}{2\sigma_0^2}\right)\right] \cdot \binom{d}{d_1}\rho^{d_1}(1-\rho)^{d-d_1} = Q\left(\sqrt{\frac{2E_s d}{N_0}}\right)\sum_{d_1=0}^{d}\cdot\binom{d}{d_1}\left(\rho e^{\xi}\right)^{d_1}(1-\rho)^{d-d_1} \quad (12)$$

where

$$\xi = \frac{E_s}{N_0} - \frac{E_s}{N_0+\rho^{-1}N_j}.$$

Then, it is simple to show that the maximum term in the summation in (12) is for $d_1 = d_{1,\max}$, where

$$d_{1,\max} = \left\lceil \frac{d\left[\rho(1-\rho)e^{\xi}\right]+1}{1+\rho(1-\rho)e^{\xi}} \right\rceil.$$

For fixed E_s/N_j and ρ, as E_s/N_0 increases $d_{1,\max} \to d$, and the performance will be dominated by the event that all d code symbols are jammed. Thus, to ensure maximum asymptotic gain from a HARQ scheme, all jammed symbols should be retransmitted. At high E_s/N_0, retransmitting unjammed symbols will provide a small performance gain and, thus, the number of symbols to be retransmitted can be reduced by not retransmitting symbols that are unjammed.

We use the MAP algorithm for the jammer state to estimate which symbols are jammed and the MAP algorithm for the message to estimate which bit decisions are reliable. We first consider RB-HARQ strategies that use this information to retransmit a fixed number of bits in response to a decoding error. By combining the two reliability measures, we propose several HARQ strategies that request retransmission for some set of bits that is determined to be unreliable.

1) In RB-HARQ(J), the destination uses only the jamming information $L(j_k)$ to decide which code symbols are jammed. In the absence of perfect jammer state information, a symbol is estimated to be jammed if $L(j_k) > 0$ and is estimated to be unjammed, otherwise. The information about the set of code symbols is conveyed to the source, which retransmits the code symbols that the destination estimates to be jammed.
2) In RB-HARQ(R), the destination uses only the reliability $|L(u_k)|$ to decide which information bits are unreliable. The information about the set of such unreliable information bits is conveyed to the source. The source then retransmits the code symbols corresponding to those information bits.
3) In RB-HARQ(R+J), the destination uses both $|L(u_k)|$ and $L(j_k)$ to determine those information bits which have the least value of $|L(u_k)|$ and also have one or both code symbols jammed. Information about the set of such information bits is conveyed to the source, which then retransmits the code symbols corresponding to such bits.

We also consider an RB-HARQ scheme that adapts the number of bits to be retransmitted based on reliability information. In the RB-HARQ(R-A) scheme, the size of the retransmission is adapted based on the bit reliabilities $|L(u_k)|$ of the bits in the packet. We determine whether a bit should be retransmitted by comparing an estimate of the probability of error for the bit to a target bit error probability. For example, packet error probabilities of 10^{-2} result in a negligible degradation in throughput. So, we can choose a target bit error probability that will result in packet error probabilities below 10^{-2}. The probability of bit error for an information bit can be estimated as the minimum of the *a posteriori* probabilities, which is given by

$$P_b \approx \min\{P(u_k=+1|\mathbf{y}), P(u_k=-1|\mathbf{y})\} = \frac{1}{1+e^{|L(u_k|\mathbf{y})|}}. \quad (13)$$

The required probability of bit error, P_b to achieve a specified packet error probability will depend, in general, on a number of different parameters like E_s/N_0, the jammer parameters, and the number of retransmissions allowed. For the work presented in this paper, simulations are used to find the value of P_b which achieves the desired P_e.

We compare these RB-HARQ approaches to conventional approaches in which the set of retransmitted symbols is not adapted based on reliability information. We consider Type-I HARQ schemes in which the entire packet is retransmitted in response to error detection. We consider both the original Type-I HARQ (in which the new packet replaces the previous packet) and Type-I HARQ with packet combining. These schemes retransmit significantly more bits than the RB-HARQ schemes that we propose, so we also consider a HARQ scheme that does not use reliability information and that retransmits the same number of bits as our RB-HARQ scheme. The schemes that we consider transmit either a random set of bits or a set of bits that is uniformly spaced throughout the packet so as to achieve the same overhead as our reliability-based schemes. This approach is analogous to incremental redundancy hybrid ARQ schemes that are used with punctured codes, in which the symbols to be transmitted are selected uniformly from the set of code symbols that were not previously transmitted.

A. Analysis of Probability of Packet Error for HARQ

We provide a brief analysis of several of the HARQ schemes discussed above. In this section, we derive an upper bound on the probability of packet error after a single retransmission for each scheme, but the bounds are easily extended to multiple retransmissions. We make several assumptions that differ from our simulations in order to make the analysis feasible. Our upper bounds are calculated based on codeword ML decoding. However, for the simulation results, we employ bitwise MAP decoding. For sufficiently high signal-to-noise ratio, these will match very closely, as the BCJR algorithm becomes more closely approximated by its max-log-MAP form. The max-log-MAP form has been shown to be equivalent to the soft-output Viterbi algorithm, a codeword ML algorithm [26]. In addition, we calculate the bound under the assumption of ideal interleaving, although for most of our simulation results we use finite, rectangular interleaving.

We first consider conventional approaches to HARQ. For Type-I HARQ without packet combining, the packet error probability after one retransmission is given by $(P_e)^2$, where P_e is given in (10). For Type-I HARQ with packet combining, each symbol is received twice, and the packet error probability can be determined from (11) with PEP given by

$$P_d = \sum_{d_1=0}^{2d} Q\left(\sqrt{E_s\left(\frac{d_1}{\sigma_1^2}+\frac{2d-d_1}{\sigma_0^2}\right)}\right)\cdot\binom{2d}{d_1}\rho^{d_1}(1-\rho)^{d-d_1}. \tag{14}$$

We now consider HARQ schemes that do not retransmit the entire packet. The analysis at the beginning of this section indicates that the asymptotic performance will be dominated by the set of jammed symbols. Let $N_c = (n/k)T$ denote the total number of transmitted bits. Then before retransmission, the expected number of symbols that are jammed is ρN_c, so we constrain the average number of symbols to be retransmitted to also equal ρN_c.

We consider a conventional approach to incremental redundancy HARQ (IR-HARQ) in which the ρN_c bits are uniformly spaced throughout the entire packet. For the purposes of analysis, we model this as a system in which a random set of symbols is retransmitted such that the average number of symbols retransmitted is ρN_c. Any given symbol is independently selected to be retransmitted with probability ρ. The PEP after retransmission for this HARQ scheme is given by

$$P_d = \sum_{d_1=0}^{d}\sum_{j=0}^{d+d_1} Q\left(\sqrt{E_s\left(\frac{j}{\sigma_1^2}+\frac{d+d_1-j}{\sigma_0^2}\right)}\right) \cdot\binom{d+d_1}{j}\rho^{j}(1-\rho)^{d+d_1-j}\cdot\binom{d}{d_1}\rho^{d_1}(1-\rho)^{d-d_1} \tag{15}$$

where d_1 of the d symbols in the error event are randomly selected for retransmission. Then, of the total $d+d_1$ symbols that are transmitted in either the original transmission or the retransmission, j denotes the number of symbols that are jammed.

In the RB-HARQ(J) approach, the set of symbols to be retransmitted depends on the set of symbols that is estimated to be jammed. Assuming perfect knowledge of the jamming state, the PEP after one retransmission is

$$P_d = \sum_{d_1=0}^{d}\sum_{d_2=0}^{d_1} Q\left(\sqrt{E_s\left(\frac{d_1+d_2}{\sigma_1^2}+\frac{d-d_2}{\sigma_0^2}\right)}\right) \cdot\binom{d_1}{d_2}\rho^{d_2}(1-\rho)^{d_1-d_2}\cdot\binom{d}{d_1}\rho^{d_1}(1-\rho)^{d-d_1} \tag{16}$$

where d_1 is the number of symbols out of d that are jammed before the retransmission. All the d_1 jammed symbols are retransmitted, and d_2 of them are jammed during the retransmission.

With the RB-HARQ(J) scheme, there is a question of what to do if we allow further retransmissions. If we request that only the jammed symbols from the previous transmission are resent, then after k transmissions, only ρ^k symbols will be requested for retransmission. This number may be very small (for example the third retransmission with $\rho = 0.4$ will contain only 6.4% of the symbols in the packet). So, we consider an alternative approach that can provide a higher throughput at low E_s/N_0. For RB-HARQ(J) with multiple retransmissions, the source alternates between retransmitting the symbols that are estimated to be jammed and retransmitting the entire packet (as in Type-I HARQ). In each case, soft combining is employed. For example with three retransmissions, the first retransmission will consist of those bits that are estimated to be jammed in the original transmission. If the packet can still not be decoded successfully, the entire packet is retransmitted. In the third retransmission, only those bits that were jammed during the previous transmission will be retransmitted. The PEP for this scheme is

$$P_d^m = \sum_{d_1=0}^{d}\sum_{d_2=0}^{d_1}\sum_{d_3=0}^{d}\sum_{d_4=0}^{d_3} Q\left(\sqrt{E_s\left(\frac{d_1+d_2+d_3+d_4}{\sigma_1^2}+\frac{2d-d_2-d_4}{\sigma_0^2}\right)}\right) \cdot\binom{d}{d_1}\binom{d_1}{d_2}\binom{d}{d_3}\binom{d_3}{d_4} \times\rho^{d_1+d_2+d_3+d_4}(1-\rho)^{2d-d_2-d_4}. \tag{17}$$

As in (16), d_1 denotes the number of symbols that are jammed in the first retransmission. All of these d_1 symbols are retransmitted, and d_2 denotes the number of those symbols that are jammed. Similarly, the entire packet is retransmitted in the second retransmission, and d_3 denotes the number of symbols that are jammed. All of these d_3 symbols are retransmitted in the third iteration, and d_4 denotes the number of those symbols that are jammed.

B. Size of Retransmission-Request Packet

In conventional HARQ, it is theoretically possible for a single feedback bit to be sent from the receiver to the transmitter to indicate an acknowledgment (ACK) or negative acknowledgment (NACK). In practice, unless this bit is piggybacked on a data packet, the ACK or NACK typically uses much more resources including a synchronization preamble and medium access control (MAC) address information for the sender and receiver. In our results, we consider the best-case scenario of single feedback bit for the conventional HARQ schemes so that our results are not tied to a particular system.

In the RB-HARQ schemes considered in this paper, the feedback packet is larger, as it contains information about the set of unreliable bits. In order to evaluate the throughput, we first evaluate the size of the retransmission-request packet. We calculate the expected value of the size of the retransmission-request packet for RB-HARQ schemes under different approaches to compress the retransmission-request packet.

Size of Retransmission-Request Packet With Transmission of Uncompressed Bit Indices: Recall that for the RB-HARQ schemes with fixed retransmission size, the number of bits to be retransmitted is equal to the expected number of jammed code symbols per packet, which is given by $\rho(n/k)T$. The simplest (and one of the least efficient) ways to design the retransmission-request packet is to provide the source $\mathbf{S}$ with a list of the bits to be retransmitted. As an alternative, the retransmission-request packet can be equal to the size of the transmitted packet, $N_c = (n/k)T$, with a 1 in the position of each symbol to be retransmitted and 0 in the other positions. The number of bits required to represent the position of a code symbol is equal to $\lceil \log_2((n/k)\cdot T)\rceil$, where $\lceil\cdot\rceil$ denotes the ceiling operator.

Then, the average size of the retransmission-request packet, denoted by N_f, is given by

$$N_f = \min\left[\left(\rho\frac{n}{k}T\right)\cdot\left\lceil\log_2\left(\frac{n}{k}\cdot T\right)\right\rceil, \frac{n}{k}\cdot T\right]. \quad (18)$$

Size of Retransmission-Request Packet With Run-Length Arithmetic Coding: Both the bit reliabilities and jamming states are correlated over time and, thus, are amenable to compression. The jamming states are Markovian and, thus, can be optimally compressed using arithmetic run-length coding [27], [28] for a Markov source. Bit reliabilities can also be treated as approximately Markovian and thus can also be compressed in a similar way. However, modeling and compression of bit reliabilities is beyond the scope of this paper. Consider the RB-HARQ(J) scheme. Once the receiver has identified the jammed symbols, it can use the estimates of the jamming parameters in the arithmetic run-length source coding process. The compression rate achievable using the arithmetic run-length coding is equal to the entropy rate of the Markov source, which is given by

$$H(S) = P_0 \cdot H(S_0) + P_1 \cdot H(S_1) \quad (19)$$

where $H(S_0)$ and $H(S_1)$ are the entropy of state 0 and 1, respectively, of the jammer. The entropy $H(S_i)$ of state i is given by the standard entropy of a binary source with output probabilities p_{ii} and $1-p_{ii}$, corresponding to the transition probabilities from state i in Fig. 3. The expected size of the retransmission-request packet is then given by

$$N_f = \left(\frac{n}{k}\cdot T\right)H(S). \quad (20)$$

Size of Retransmission-Request Packet With Simple Compression: Because of the complexity of arithmetic coding and decoding, as well as the need to accurately estimate the transition probabilities of the hidden Markov model for the jammer in order to achieve optimal compression, we propose the following suboptimal scheme for use with RB-HARQ(J). In this simple compression scheme, the retransmission-request packet consists of the start and end positions of all the bursts of jammed symbols. Here, a *burst* of jammed symbols is a consecutive sequence of jammed symbols such that the symbols immediately before and after the burst are unjammed. To calculate the size of the retransmission-request packet with simple compression, we first calculate the average number of bursts of jammed symbols in the received packet.

Let B be the number of bursts in the received packet and $j_i \in \{0,1\}$ represent the state of the jammer at time i. Let B_i be the number of bursts starting at time i, where a burst is defined to start at time i if either $i = 0$ and $j_0 = 1$ or if $j_i = 0$ and $j_{i+1} = 1$. Then

$$\begin{aligned} E[B] &= E\left[\sum_{i=0}^{N_c-2} B_i\right] \\ &= E[B_0] + \sum_{i=1}^{N_c-2} E[B_i] \\ &= 1\cdot P_1 + P_0p_{01} + (N_c-2)P_0p_{01} \\ &= \frac{(N_c-1)\rho}{E\{T_J\}} + \rho \end{aligned} \quad (21)$$

where P_0 and P_1 are the steady-state probabilities of the jammer being in state 0 and 1, respectively. Thus, the average size of the retransmission-request packet is the expected number of bursts multiplied by the number of bits required to represent the start and end positions of the bursts, which is given by

$$N_f = E[B]\cdot 2\left\lceil\log_2\left(\frac{n}{k}\cdot T\right)\right\rceil. \quad (22)$$

IV. Performance of Estimation Algorithm

We assume that $\mathbf{D}$ knows the statistics of the thermal noise, but in general does not have any CSI about the jamming state, the transition probabilities, and the PSD in the jamming state. This information about the jamming needs to be accurately estimated for best performance in decoding the packet. In this section, we show the performance of the iterative MAP algorithm for jamming estimation and decoding. For all of the results presented in this paper, the code used for transmission from $\mathbf{S}$ to $\mathbf{D}$ is a rate 1/2, constraint length $K = 7$ convolutional code with generator polynomials 554 and 744 (in octal). In all of the results, the total block size is 1000 information bits, including the tail bits. Except where noted, the coded bits are interleaved using a rectangular interleaver of size 45 × 45. The retransmission process effectively reduces the code rate and, hence, increases the received energy per bit, E_b, at the receiver. We present results in terms of the channel symbol energy-to-noise density ratio E_s/N_0 and the average symbol-energy to jammer-noise density ratio E_s/N_j. These ratios remain constant during the ARQ process.

The parameters of the Markov chain for the jammer are $\rho = 0.4$ and $E\{T_J\} = 40$. For the case of no CSI, all of the jamming parameters are estimated using the Baum–Welch/BCJR algorithm. The Baum–Welch algorithm requires some initial estimate to distinguish the densities emitted by the two states. We use the initial estimate that the variance in the jamming state is twice the variance in the unjammed state.

We can measure the performance of the estimation and detection algorithm directly in terms of the probability of miss and probability of false alarm. The probability of miss is calculated as the ratio of the number of symbols that are jammed and not detected to be jammed to the number of symbols that are jammed. The probability of false alarm is calculated as the ratio of the number of symbols that are unjammed and detected to be jammed to the total number of unjammed symbols. These performance metrics are illustrated in Fig. 4 for the estimation algorithm after ten iterations at $E_s/N_j = -3$ dB. The performance of the ML estimation algorithm is compared with the performance of jamming detection with perfect knowledge of all the jammer parameters including the transition probabilities, the average jammer PSD $N_j/2$, and ρ.

The performance of the decoding algorithm has been shown to be most sensitive to misses in jamming detection, in which a jammed symbol is identified as unjammed [20]. The results in Fig. 4 show that for $E_s/N_j = -3$ dB, the iterative MAP algorithm with ML estimation achieves probability of miss less than 0.05 for all values of E_s/N_0 greater than 0 dB. Detection with estimation of all parameters performs as well as detection when all parameters are known except at very low values of E_s/N_0.

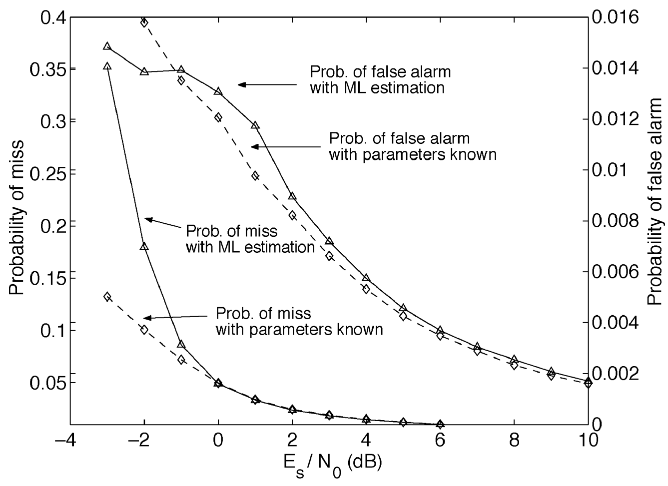


Fig. 4. Probability of miss and false alarm of jammed symbols when all jamming parameters must be estimated in comparison to when all jamming parameters are known at $E_s/N_j = -3$ dB.

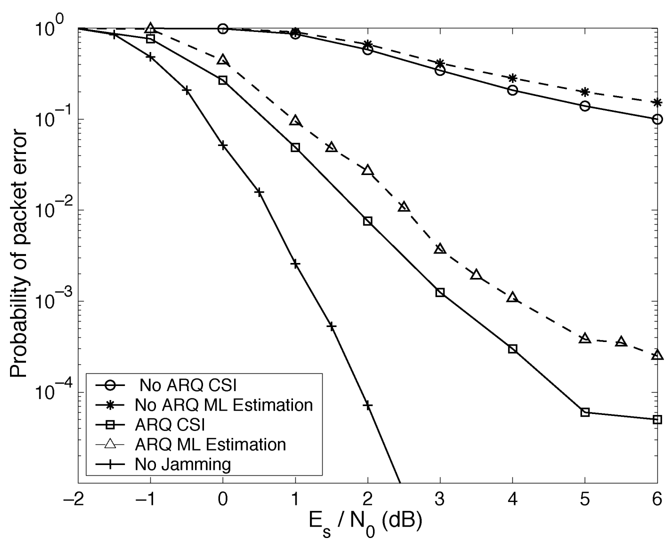


Fig. 5. Probability of packet error for RB-HARQ(J) with estimation of jamming parameters or perfect CSI, $\rho = 0.4$ and $E_s/N_j = -3$ dB.

This is because at low values of E_s/N_0, the variance of the thermal noise, $N_0/2$ is comparable to the variance, $N_j/2\rho$ of the jammer signal. Thus, it is difficult to detect whether a symbol is jammed. However, at such low E_s/N_0, the packet error probability will be very high with even perfect knowledge of the jammer state. The results in Fig. 4 show that the ML estimation algorithm achieves a probability of false alarm of less than 1% for all values of E_s/N_0 greater than 1 dB. The performance of ML estimation, in terms of false alarm probability, is close to the performance when the jammer parameters are known.

The results in Fig. 5 show the probability of packet error at $E_s/N_j = -3$ dB for RB-HARQ(J), which requests retransmission for all symbols that are identified as jammed. The performance of RB-HARQ(J) scheme with estimation of all parameters is compared with the case when the decoder has perfect CSI. Perfect CSI means that the decoder knows all the jammer parameters and which symbols are jammed. The results show that the performance of the estimation algorithm is within 0.25 to 0.5 dB of the CSI case.

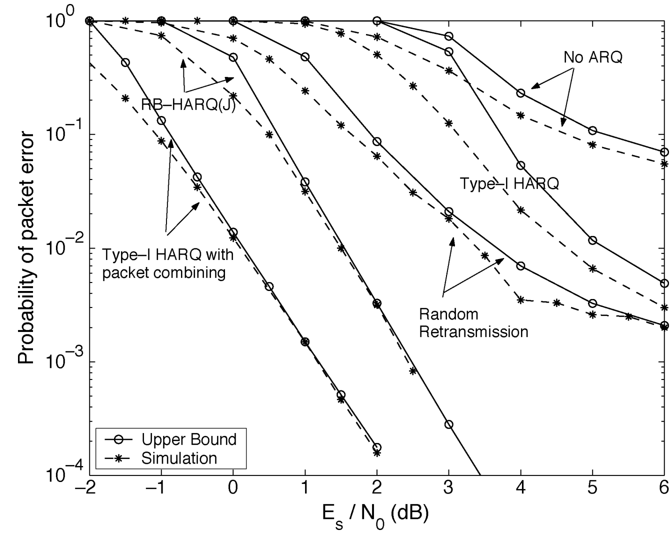


Fig. 6. Probability of packet error for RB-HARQ(J), Type-I HARQ and retransmission of a random set of symbols, $\rho = 0.4$ and $E_s/N_j = -3$ dB.

The results in this section show that iterative parameter estimation achieves very good performance and that having to estimate the jamming parameters does not significantly degrade the performance of RB-HARQ. In the next section, we compare the performance of the different proposed RB-HARQ schemes with conventional HARQ schemes. We show the results in terms of probability of packet-error and assume perfect CSI for these results.

V. Performance Results

In this section, we compare the performance of the proposed RB-HARQ schemes to conventional HARQ schemes. The convolutional code is the same as in the previous section. Except where noted, the parameters of the jammer are given by $E_s/N_j = -3$ dB, $\rho = 0.4$, and $E\{T_J\} = 40$. We evaluate the performance in terms of packet error probabilities and throughput.

A. Packet Error Probabilities

We first compare analytical and simulation results for the probability of packet error P_e after one retransmission for the HARQ schemes analyzed in Section III. In the RB-HARQ(J) scheme, all jammed symbols are retransmitted. We compare the performance of this approach with three conventional HARQ schemes. We consider Type-I HARQ with and without packet combining. These schemes retransmit significantly more bits than RB-HARQ(J), so we also consider an IR-HARQ scheme that retransmits a random set of bits such that the average overhead is the same as for RB-HARQ(J). The results of this comparison are illustrated in Fig. 6. The analytical upper bounds are shown using solid lines, and the simulation results are shown using dashed lines.

The results show that to achieve $P_e = 10^{-1}$, Type-I HARQ without packet combining provides approximately 1.5 dB gain over no ARQ. If Type-I HARQ is used with packet combining, the gain is approximately 6 dB. For the other HARQ results, the overhead is only 40% of that of the Type-I HARQ schemes.

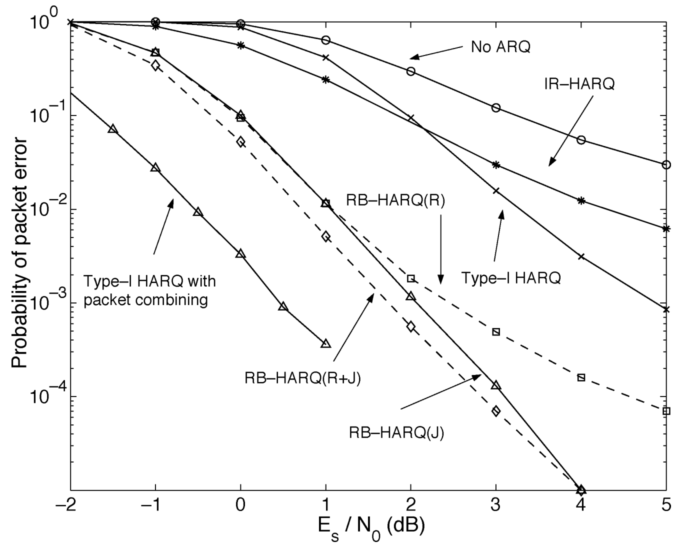


Fig. 7. Probability of packet error for different RB-HARQ schemes compared with Type-I HARQ and conventional HARQ, $\rho = 0.4$ and $E_s/N_j = 0$ dB.

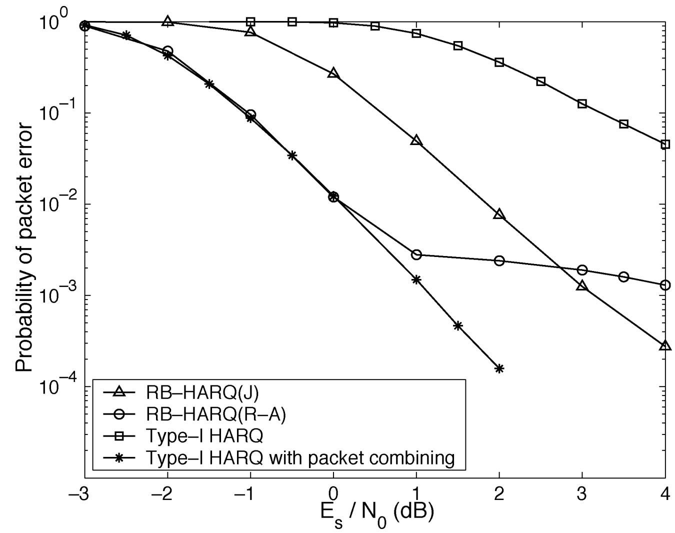


Fig. 8. Probability of packet error for adaptive and fixed RB-HARQ, $\rho = 0.4$, and $E_s/N_j = -3$ dB.

The IR-HARQ scheme can achieve $P_e = 10^{-1}$ with 1.5 dB lower E_s/N_0 than Type-I HARQ without packet combining. RB-HARQ(J) requires 1.25 dB lower E_s/N_0 than incremental redundancy with random retransmissions at $P_e = 10^{-1}$ and the performance difference increases drastically for lower target values of P_e. Although RB-HARQ(J) requires approximately 1.2–1.6 dB higher E_s/N_0 than Type-I HARQ with packet combining, it only retransmits 40% of the bits of Type-I HARQ. The results also show that the upper bounds computed using (10) and (14)–(16) provide very tight bounds on the packet error probabilities.

For the remainder of the results, the channel symbols are interleaved using a 45 × 45 rectangular bit interleaver. For these results, the conventional IR-HARQ scheme transmits a uniformly spaced set of bits. The results in Fig. 7 illustrate the packet error rate for different RB-HARQ and conventional HARQ schemes. For these results $E_s/N_j = 0$ dB. The average number of symbols retransmitted in response to NACK is equal to $\rho(n/k)T = 800$ (the expected number of jammed symbols) for all of the HARQ schemes except for Type-I HARQ in which the entire packet is retransmitted.

The results in Fig. 7 show that all three RB-HARQ schemes achieve better performance than the conventional HARQ approaches that do not employ reliability, except for Type-I HARQ with packet combining, which retransmits significantly more symbols. To achieve a packet error rate of less than 10^{-2}, the required E_s/N_0 for RB-HARQ is at least 3 dB less than for IR-HARQ, which retransmits the same number of symbols but does not use reliability information. RB-HARQ(R+J) achieves the best performance because it uses both the log-APPs to decide which symbols are to be retransmitted. This scheme performs about 0.25 dB better than RB-HARQ(J) in which all jammed symbols are retransmitted. Among all the proposed RB-HARQ schemes, RB-HARQ(R) that selects the bits to be retransmitted based only on $|L(u_k)|$ performs the worst. This because at high values of E_s/N_0, the performance is limited by the jamming, as shown in the analysis in Section III. Since RB-HARQ(R) retransmits both code symbols for information bits that have low values of $|L(u_k)|$ and one or both of these code symbols are unjammed, some jammed symbols will not be retransmitted because the total number of retransmitted symbols is equal to the average number of jammed symbols. The residual set of jammed symbols results in an error floor for high E_s/N_0.

In Fig. 8, we consider the RB-HARQ(R-A) scheme, in which the number of bits to be retransmitted is adaptively selected based on a specified target packet error probability P_e. For these results only one retransmission is considered. As discussed in Section III, offline simulations were used to determine that a target bit error probability $P_b = 10^{-5}$ achieves $P_e \approx 10^{-2}$. Using (13), this translates to requiring retransmission for all bits with $|L(u_k)| < 11.5$. The results in Fig. 8 illustrate the packet error probability achieved at $E_s/N_j = -3$ dB for RB-HARQ(R-A), RB-HARQ(J), and Type-I HARQ with and without packet combining. The results show that for $E_s/N_0 < 0$ dB, none of the schemes are able to achieve the target packet error probability of 10^{-2}. For $E_s/N_0 \geq 0$ dB, the adaptive retransmission scheme does achieve packet error probabilities below 10^{-2}. The real effect of adaptive RB-HARQ is that the average number of symbols to be retransmitted decreases as the E_s/N_0 increases. We study this in terms of its effect on throughput in the next subsection.

B. Throughput Results

We now compare the throughput performance of RB-HARQ and conventional HARQ schemes. We first consider the performance of RB-HARQ(J) that retransmits all the jammed code symbols. Throughput is defined as the ratio of the number of bits per packet to the expected number of coded bits that must be transmitted to achieve correct decoding of the packet. Thus, the throughput S is given by

$$S = \frac{T}{X} P_S$$

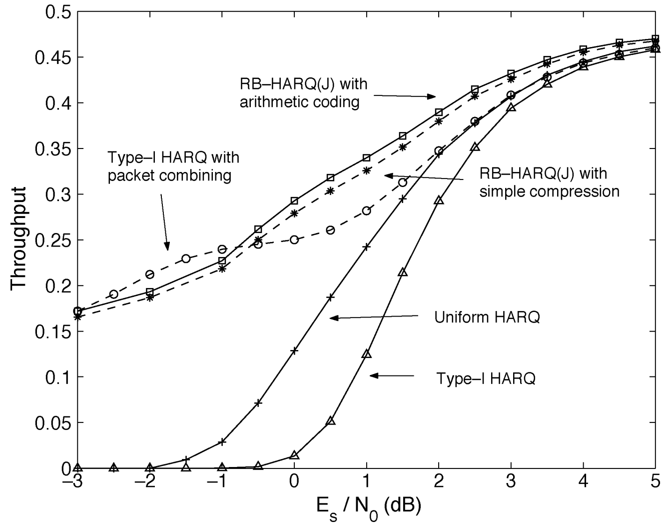


Fig. 9. Throughput for RB-HARQ, Type-I HARQ and conventional (uniform) HARQ, after three retransmissions at $\rho = 0.4$ and $E_s/N_j = -3$ dB.

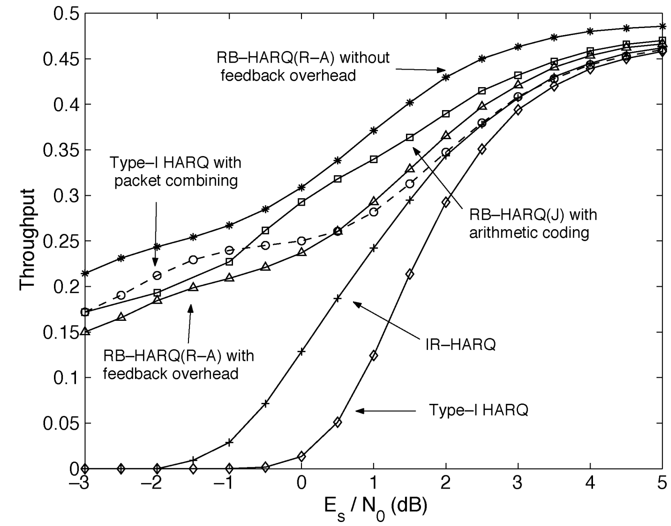


Fig. 10. Throughput of adaptive RB-HARQ(R), RB-HARQ(J), Type-I HARQ, and conventional (uniform) HARQ, $\rho = 0.4$ and $E_s/N_j = -3$ dB.

where T is the number of information bits in a packet, X is the expected number of coded symbols that are transmitted in both directions during the HARQ process, and P_S is the probability of packet success by the end of the HARQ process. For these results, we consider up to three retransmissions. If the packet is still in error after three retransmissions, then the whole packet is retransmitted and the HARQ process starts over.

The results in Fig. 9 show the throughput of RB-HARQ(J) and the conventional HARQ schemes for $E_s/N_j = -3$ dB. As previously discussed, RB-HARQ(J) alternates between retransmitting the set of symbols that are estimated to be jammed and complete packet retransmission (as in Type-I HARQ). The size of the retransmission-request packet for the conventional HARQ scheme is taken to be 1 bit. The size of the retransmission-request packet for RB-HARQ is calculated using the formulas in Section III-B. The size of the retransmission-request packet composed of bit indexes (no compression) is 2000 bits. The average size of the retransmission-request packet compressed using arithmetic run-length coding, calculated using (20) is 282 bits. Thus, using arithmetic coding helps in reducing the size of retransmission-request packet by almost 85%. For RB-HARQ(J) with the simple compression scheme, the average size of the retransmission-request packet is 462 bits. The results in Fig. 9 show that despite larger retransmission-request packets, RB-HARQ(J) technique that uses compression achieves better throughput than all the other HARQ techniques for all values of $E_s/N_0 \geq -0.5$ dB. At very low values of $E_s/N_0 (\leq -1$ dB), Type-I HARQ with packet combining performs around 0.5 dB better than RB-HARQ(J) because in this regime, the performance gain from retransmitting more symbols outweighs the additional overhead of retransmitting the entire packet. The results also show that using the simple compression scheme for the retransmission-request packet achieves throughput close to that achieved by using optimal arithmetic run-length coding.

The results in Fig. 10 show the throughput for RB-HARQ(R-A) in comparison with RB-HARQ(J) and the conventional schemes. Recall that in RB-HARQ(R-A), the number of retransmitted bits is adaptively selected to achieve some target error probability. For these results, up to three retransmissions are allowed, and simulations were carried out to find the target P_b that achieves the maximum throughput. It was observed that a target bit error probability of 5×10^{-3} offered the best throughput for the range of E_s/N_0 and jammer parameters considered in our work. We show two curves for RB-HARQ(R-A), one in which the overhead is determined based on the retransmission-request packet consisting of the bit indexes of all symbols to be retransmitted (no compression) and one in which that overhead is ignored. Our reason for including the results without the overhead is twofold. First, in many cases, it might not be desired to treat the overhead of the retransmission-request packet the same as the forward transmission because that link is assumed to not be jammed. Second, although outside of the scope of this paper, compression can be applied to this retransmission-request packet, and the results we present represent upper and lower bounds on the performance with compression. The results show that if the overhead in the retransmission-request packet can be ignored or significantly reduced through compression, then RB-HARQ(R-A) achieves the best throughput at all values of E_s/N_0. This is because adaptive RB-HARQ(R) adapts the size of retransmission to the reliability of the received bits. It retransmits more bits at low E_s/N_0 to achieve correct decoding of the packet and retransmits fewer bits at high E_s/N_0 while still achieving a sufficiently low P_s. Even when we account for the size of uncompressed retransmission-request packet in throughput calculations, adaptive RB-HARQ(R) performs better than conventional HARQ and Type-I HARQ that does not use packet combining for the entire range of E_s/N_0.

Finally, we investigate the effect of different values of ρ on the performance of the various HARQ techniques. The results in Fig. 11 compare the throughput of RB-HARQ(J) and RB-HARQ(R-A) with the conventional HARQ schemes as a function of ρ. For these results $E_s/N_0 = 0$ dB, $E_s/N_j = -3$ dB, and $E\{T_J\} = 40$. The results show that if we neglect the overhead in the retransmission-request packet,

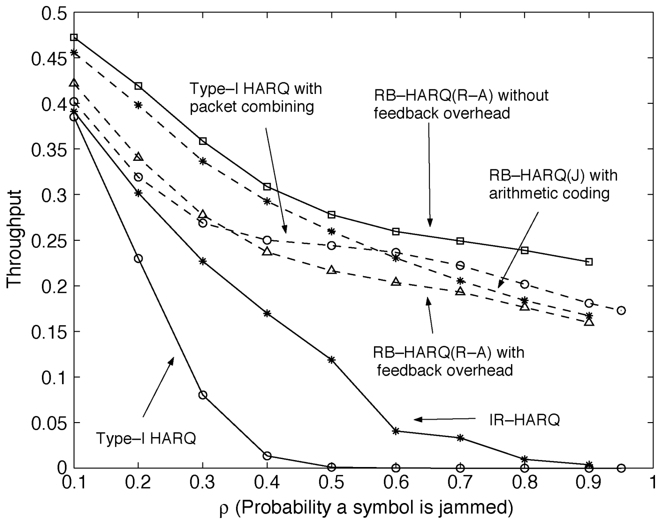


Fig. 11. Throughput for RB-HARQ, Type-I HARQ, and conventional (uniform) HARQ as a function of ρ at $E_s/N_0 = 0$ dB and $E_s/N_j = -3$ dB.

RB-HARQ(R-A) achieves the best performance over all ρ. Thus, the RB-HARQ(R-A) scheme can effectively adapt the set of retransmitted bits to different values of ρ. The RB-HARQ(J) scheme achieves the same performance as RB-HARQ(R-A) for $\rho < 0.5$ For higher values of ρ, the overhead in the retransmission-request packet reduces the performance of RB-HARQ(J). Above $\rho = 0.58$, Type-I HARQ with packet combining outperforms RB-HARQ(J). This suggests that if ρ is estimated to be very high, then the RB-HARQ(J) scheme can be simply modified to request retransmission of the whole packet. Note that neither Type-I HARQ without packet combining nor IR-HARQ are competitive techniques for dealing with jamming except at very low values of ρ.

VI. Conclusion

In this paper, we proposed RB-HARQ schemes for use in hostile jamming environments and evaluated the performance of these schemes. The proposed schemes use the MAP algorithm to estimate the *a posteriori* probabilities for the information bits and the jammer state. Results show that the performance of the estimation algorithm is within 0.25–0.5 dB of the perfect CSI case. In terms of packet error rate, all of the proposed RB-HARQ schemes are shown to offer significantly better performance than an incremental redundancy HARQ (IR-HARQ) scheme that has the same overhead but does not utilize reliability information. We also presented throughput results that take into account the overhead of the retransmission-request packet. An optimal arithmetic run-length coding technique and a suboptimal but much simpler run-length coding technique are proposed to compress the retransmission-request packet for the RB-HARQ(J) scheme, which retransmits the symbols that are estimated to be jammed. The results show that RB-HARQ(J) offers a higher throughput than Type-I HARQ with packet combining except at very low E_s/N_0 or very high ρ. We also presented performance results for a scheme that adapts the size of the retransmission-request packet based on the bit reliabilities. This RB-HARQ(R-A) scheme offers the highest throughput if the overhead of the retransmission-request packet can be neglected. Thus, adaptation in the HARQ scheme based on reliability is shown to be an effective means for dealing with hostile jamming.

References

[1] M. B. Pursley and C. S. Wilkins, “Adaptive transmission for frequency-hop radio networks,” in *Proc. 35th Allerton Conf. Commun., Control, Comput.*, Monticelli, IL, Sep. 1997, pp. 713–718.

[2] ——, “Adaptive transmission for frequency-hop communication over time-varying channels,” in *Proc. IEEE Int. Symp. Inf. Theory*, Cambridge, MA, Aug. 1998, p. 454.

[3] C. S. Wilkins and M. B. Pursley, “Adaptive-rate coding for frequency-hop communications over Rayleigh-fading channels,” *IEEE J. Sel. Areas Commun.*, vol. 17, no. 7, pp. 1224–1232, Jul. 1999.

[4] M. B. Pursley and J. S. Skinner, “Adaptive-rate turbo product coding for frequency-hop transmission and time-varying partial band interference,” in *Proc. IEEE Military Commun. Conf.*, Boston, MA, Oct. 2003, pp. 446–451.

[5] M. B. Pursley and H. B. Russell, “Adaptive forwarding in frequency-hop spread-spectrum packet radio networks with partial band jamming,” *IEEE Trans. Commun.*, vol. 41, pp. 613–620, Apr. 1993.

[6] ——, “Routing in frequency-hop packet radio networks with partial band jamming,” *IEEE Trans. Commun.*, vol. 41, no. 7, pp. 1117–1124, Jul. 1993.

[7] ——, “Network protocols for frequency-hop packet radios with decoder side information,” *IEEE J. Sel. Areas Commun.*, vol. 12, pp. 612–621, May 1994.

[8] J. H. Gass, M. B. Pursley, H. B. Russell, R. J. Saulitis, C. S. Wilkins, and J. S. Wysocarski, “Adaptive transmission protocols for frequency-hop radio networks,” presented at the IEEE Military Commun. Conf., Boston, MA, Oct. 1998.

[9] M. B. Pursley, H. B. Russell, and J. S. Wysocarski, “Energy-efficient routing in frequency-hop radio networks with partial-band interference,” in *Proc. IEEE Wireless Commun. Netw. Conf.*, Chicago, IL, Sep. 2000, pp. 79–83.

[10] J. M. Hanratty and G. L. Stuber, “Performance analysis of hybrid ARQ protocols in a slotted direct-sequence code-division multiple-access network: Jamming analysis,” *IEEE J. Sel. Areas Commun.*, vol. 8, pp. 562–579, May 1990.

[11] P. M. Feldman, “Effect of interference on throughput of adaptive hybrid ARQ protocol,” in *Proc. IEEE Military Commun. Conf.*, San Diego, CA, Oct. 1988, pp. 691–696.

[12] G. Levannier, M. Testard, and P. Bataille, “Performance analysis of RS-coded PSK with ARQ protocol for slow-frequency-hopping spread spectrum EHF military satellite communication,” in *Proc. IEEE Military Commun. Conf.*, San Diego, CA, Nov. 1995, pp. 394–398.

[13] J. M. Shea, “Reliability-based hybrid ARQ,” *Inst. Elect. Eng. Electron. Lett.*, vol. 38, no. 13, pp. 644–645, 2002.

[14] H. Kim and J. M. Shea, “New turbo-ARQ techniques based on estimated reliabilities,” in *Proc. Wireless Commun. Netw. Conf.*, vol. 1, New Orleans, LA, Mar. 2003, pp. 843–848.

[15] A. Roongta and J. M. Shea, “Reliability-based hybrid ARQ using convolutional codes,” in *Proc. Int. Conf. Commun.*, vol. 4, Anchorage, AK, May 2003, pp. 2889–2893.

[16] V. Tripathi, E. Visotsky, R. Peterson, and M. Honig, “Reliability-based type II hybrid ARQ schemes,” in *Proc. IEEE Intl. Conf. Commun.*, vol. 4, Anchorage, AK, May 2003, pp. 2899–2903.

[17] A. Roongta and J. M. Shea, “Reliability-based hybrid ARQ and rate-compatible punctured convolutional codes,” presented at the IEEE Wireless Commun. Netw. Conf., Atlanta, GA, Mar. 2004.

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

[19] J. Kang, W. E. Stark, and A. Hero, “Turbo codes for fading and bursty channels,” in *Proc. IEEE Commun. Theory Mini Conf.*, Nov. 1998, pp. 40–45.

[20] J.-W. Moon, J. M. Shea, and T. F. Wong, “Mitigation of partial-time jamming through collaboration,” *IEEE Trans. Wireless Commun.*, 2004. [Online]. Available: http://www.wireless.ece.ufl.edu/jshea/pubs/twireless_jangwook04.pdf, submitted for publication.

[21] 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.

[22] A. O. Yilmaz and W. E. Stark, "Performance of turbo coded continuous phase modulation in fading channels," in *Proc. IEEE Military Commun. Conf.*, Oct. 2002, pp. 697–702.

[23] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Englewood Cliffs, NJ: Prentice-Hall, 1995, ch. 12.

[24] S. Lin and D. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1983, ch. 15.

[25] A. J. Viterbi, "Convolutional codes and their performance in communication systems," *IEEE Trans. Commun.*, vol. COM-19, pp. 751–772, Oct. 1971.

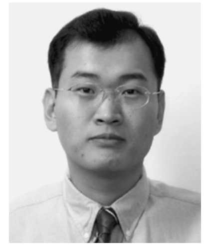
[26] M. P. C. Fossorier, F. Burkert, S. Lin, and J. Hagenauer, "On the equivalence between SOVA and max-log-MAP decodings," *IEEE Commun. Lett.*, vol. 2, pp. 137–139, May 1998.

[27] K. Sayood, *Introduction to Data Compression*. San Mateo, CA: Morgan Kaufmann, 1996, ch. 2.

[28] S. Golomb, "Run-length encodings," *IEEE Trans. Inf. Theory*, vol. 12, pp. 399–401, 1966.

Abhinav Roongta (S'03) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Delhi (IITD), India, in 2001 and the M.S. degree in electrical engineering from the University of Florida, Gainesville, in 2003. Since August 2001, he has been working towards the Ph.D. degree in electrical and computer engineering at the University of Florida.

His research interests include error control coding, signal processing, and system design for wireless communications.



Jang-Wook Moon (S'03) received the B.S. degree in electrical engineering from Sung-Kyun-Kwan University, Seoul, Korea, in 1999 and the M.S. degree in electrical and computer engineering from Yonsei University, Seoul, in 2001. He is currently working towards the Ph.D. degree in electrical and computer engineering at the University of Florida, Gainesville.

His research interest is wireless communications, with emphasis on 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. degree (highest honors) in computer engineering, and the M.S. and Ph.D. degrees in electrical engineering from Clemson University, Clemson, SC, in 1993, 1995, and 1998, respectively.

He is currently an Assistant Professor of electrical and computer engineering at the University of Florida, Gainesville. From January 1999 to August 1999, he was a Postdoctoral Research Fellow at Clemson University. He was a Research Assistant in the Wireless Communications Program, 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 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.