Reliability-Based Incremental Redundancy With Convolutional Codes

Eugene Visotsky, *Member, IEEE*, Yakun Sun, *Member, IEEE*, Vinayak Tripathi, *Member, IEEE*, Michael L. Honig, *Fellow, IEEE*, and Roger Peterson, *Fellow, IEEE*

***Abstract*—Incremental redundancy, or Hybrid type-II ARQ (HARQ), algorithms use a combination of forward error correction and retransmissions to guarantee reliable packet data communications. In this work, we propose a HARQ algorithm that exploits received packet reliability to improve system performance. Specifically, the receiver uses the average magnitude of the log-likelihood ratios of the information bits as the packet reliability metric, which is then used to determine the sizes of subsequent retransmissions. The proposed retransmission strategy attempts to maximize user throughput while satisfying a maximum packet delay constraint. The performance of our reliability-based HARQ algorithm is evaluated in static and time-varying channels through simulations. Furthermore, analytical results on the relationship between the reliability metric, the code rate and the block error rate are presented.**

***Index Terms*—Convolutional codes, Hybrid type-II ARQ (HARQ), incremential redundancy, mobile radio cellular systems.**

I. Introduction

HYBRID ARQ (HARQ) algorithms have now become an integral part of packet communication systems. The enhanced data rates for GSM evolution (EDGE) cellular radio standard, for instance, defines error correction code puncturing patterns and retransmission policies that allow the combining of received information from multiple transmissions to improve the likelihood of successful decoding [2]. When information from multiple retransmissions is combined, the system is said to be using a hybrid type-II ARQ or incremental redundancy [3]. The EDGE system also features link adaptation algorithms that select the initial modulation and coding scheme (MCS) as a function of channel quality. In general, if the link adaptation is well matched to the channel, little additional information should be required to successfully decode packets that fail after the first transmission. Yet, in EDGE, all retransmissions are of the same length as the first transmission, leading to inefficient use of channel resources. In light of this observation, third generation cellular systems, such as High Speed Downlink Packet Access, now permit variable length retransmissions of corrupted packets [1], [4], [5], [10].

In this work, a methodology for designing efficient type-II HARQ schemes for convolutional codes is presented. We consider a single-user link and use throughput and packet delay as performance measures. We do not account for the presence of a multiuser scheduler, but rather, attempt to optimize performance for a given user.

Reducing the size of retransmissions can increase throughput, but at the expense of increasing the number of retransmission attempts, leading to higher packet delays. Our objective, then, is to design a HARQ algorithm that maximizes throughput subject to a maximum packet delay constraint. The design problem involves specifying the size and content of successive retransmissions. In this work, we focus on the retransmission size as the design parameter.

The approach taken in this paper is to design a HARQ algorithm based on the received packet reliability. The reliability of the received codeword is defined as the average magnitude of the *a posteriori* log-likelihood ratios (LLRs) of the information bits. This metric can be computed at the receiver based on the outputs of the classical MAP decoder or other soft output decoders. For the proposed reliability-based HARQ (RBHARQ) algorithm, the reliability information at the receiver is accumulated from the preceding retransmissions and used to determine the optimum size of each subsequent retransmission, which is then relayed to the transmitter. For the size determination, the receiver is assumed to have perfect knowledge of the causal channel state information, but has no knowledge of future channel realizations. For the proposed algorithm, no knowledge of channel state information at the transmitter is necessary. The proposed approach is fundamentally different from standard link adaptation algorithms based on the channel state information, such as the received signal-to-noise ratio (SNR), in two key respects: 1) retransmission sizes are determined based on the state of the channel decoder, as captured by the proposed packet reliability metric and 2) retransmission sizes are determined adaptively in real-time, based on the real-time reliability output from the channel decoder. This is in contrast to SNR-based algorithms that make decisions on retransmission sizes based on the channel state information prior to the decoder. Furthermore, typically these decisions are based on longer-term SNR measurements, precluding real-time link adaptation.

The size determination in the proposed algorithm is based on two mappings: codeword reliability to block error rate (BLER) mapping and coding rate to codeword reliability mapping. In

Paper approved by L. Rasmussen, Editor for Iterative Detection, Decoding, and ARQ of the IEEE Communiations Society. Manuscript received March, 16, 2004; revised December 23, 2004. This work was supported in part by the Northwestern-Motorola Center for Communications. This paper was presented in part at the 2003 International Conference on Communications (ICC2003), Anchorage, AK, May 2003.

E. Visotsky and R. Peterson are with Motorola Laboratories Wireless Access Research Center of Excellence, Schaumburg, IL 60196 USA (e-mail: visotsky@labs.mot.com).

Y. Sun is with Mobile Devices, Motorola, Libertyville, IL 60048 USA.

V. Tripathi is with the Department of Economics, Princeton University, Princeton, NJ 08544 USA.

M. L. Honig is with the Department of Electrical Engineering, Northwestern University, Evanston, IL 60201 USA.

Digital Object Identifier 10.1109/TCOMM.2005.850012

this paper, empirical and, when possible, analytical approaches to arriving at the two mappings are examined. The reliability to code rate mapping is shown to be well approximated by a function which is linear in SNR and related to an average of a minimum of certain hypergeometric random variables. The distribution parameters depend on the weight spectrum of the convolutional code. The analytical approach for arriving at the reliability to BLER mapping is based on the fundamental relationship between the *a posteriori* probability of decoding bit error and the *a posteriori* LLR [17]. The problem is challenging since the sequence of *a posteriori* LLRs at the output of convolutional decoder is correlated. We proceed by deriving upper and lower bound approximations to the BLER as a function of reliability. To derive the bounds, we model the *a posteriori* LLRs as Gaussian-distributed random variables.

The throughput and delay performance of the proposed RBHARQ algorithm is evaluated via numerical simulations against that of the HARQ algorithm standardized for the EDGE cellular system. The simulations are performed in additive white Guassian noise (AWGN) and a number of fading channel environments. Numerical results indicate substantial throughput gains in an AWGN channel, and reduced, but still significant gains, in certain fading channel conditions. In all simulated cases, no degradation in throughput due to the RBHARQ retransmission strategy is observed relative to the EDGE IR algorithm. Furthermore, in all simulated cases, the delays encountered with the RBHARQ and EDGE IR algorithms are essentially equal.

Although our results are specific to convolutional codes, they form a foundation for designing reliability-based link quality control strategies for a broad class of iterative decoding codes with convolutional codes as constituent codes. Extensions to iterative decoding are left for future study.

Independently of this work, RBHARQ schemes based on the received codeword reliability have been also proposed in [8], [9] for turbo and convolutional codes, respectively. In [8], [9], the focus is on finding individual bits in the packet with low reliabilities for retransmissions, whereas this work concentrates on determining the optimum sizes of the retransmissions. Once the size is determined, the retransmitted bits are determined at random. Other adaptive HARQ schemes based on the channel state information at the receiver and soft information at the output of the decoder have been considered in [6], [7]. Our work differs in that the HARQ design methodology proposed here allows for explicit throughput optimization subject to a maximum packet delay constraint. Adaptation of the size, as well as of the modulation format, of the retransmissions was recently proposed in [10]. The adaptation is based on the accumulated mutual information at the receiver. A dynamic programming approach to retransmission size optimization has been proposed in [12]. The approach enables a qualitative characterization of the optimum retransmission policies, as well as an efficient method for computing the policies. The problem of finding optimum retransmission sizes with respect to the average code rate of a type-II hybrid ARQ system is considered in [13]. An information-theoretic view of HARQ algorithms is taken in [14]. The algorithms are analyzed in the setting of the Gaussian collision channel, whereas in this paper a single-user channel is considered.

The rest of this paper is organized as follows. System model and performance metrics are presented in Section II. The notion of codeword reliability is introduced in Section III. The proposed RBHARQ algorithm and its performance are discussed in Section IV. Analytical results concerning the reliability mapping and numerical examples are given in Section V. Conclusions are drawn in Section VI.

II. System Model for Incremental Redundancy

Consider a packet data system where each packet consists of N coded bits obtained by encoding I information bits with a convolutional code of rate R_C. Specifically, let information bits b_1 through b_I be encoded into a codeword $\mathbf{c}$ with bits c_1 through c_N, $c_i \in \{0,1\}$, where $R_C = I/N$. The coded bits are punctured to obtain a desired code rate, possibly interleaved, and mapped to transmission symbols s_1 through s_M. Let Π denote a set of indexes corresponding to transmitted bits and $\bar{\Pi}$ denote its complement. For simplicity, and with little loss of generality, we consider antipodal signaling in this paper, so that $M = N$ and $s_i = 2c_i - 1$.

The received sample corresponding to the ith coded bit is given by $r_i = A_i s_i + n_i$ if $i \in \Pi$ and $r_i = 0$ if $i \in \bar{\Pi}$, where A_i denotes the signal amplitude and n_i is a sample of AWGN of variance σ^2. The signal amplitude A_i is a function of the channel coefficients, as well as the type of signal processing performed at the receiver for mitigating the effects of channel dispersion. With little loss in generality, a simple matched filter receiver is assumed throughout this paper. In this case, $A_i = \|\mathbf{h}_i\|$, where $\mathbf{h}_i$ denotes a realization of P channel taps $h_{1,i}$ through $h_{P,i}$ at the time instant corresponding to the ith coded bit. In general, taps $h_{1,i}$ through $h_{P,i}$ are samples of an ergodic channel fading process. The receiver is assumed to have perfect knowledge of the current and past channel realizations. The SNR of the received signal is given by $\mathrm{SNR} = E_{\mathbf{h}}\left[\|\mathbf{h}_i\|^2\right]/2\sigma^2$, where the expectation is taken over the realizations of the channel taps. Note that our model disregards the impairments due to the intersymbol interference (ISI) observed at the receiver. Hence, this SNR expression is an upper bound on the signal-to-noise ratio attainable with practical ISI mitigating techniques. In this paper, a MAP decoder at the receiver is assumed. With antipodal signaling, the soft input to the decoder corresponding to the coded bit c_i and denoted as z_i is obtained by scaling the the received sample r_i with the channel gain A_i. That is

$$z_i = A_i r_i. \tag{1}$$

Vector $\mathbf{z}$ containing soft inputs z_1 through z_N then denotes the soft input for the entire codeword.

Consider a hybrid ARQ system with incremental redundancy. The system operates by transmitting a subset of the coded bits, attempting to decode the codeword, and requesting a retransmission if the decoding fails. In general, the subset of transmitted bits varies from retransmission to retransmission. The subsets can be optimized for specific convolutional codes using techniques such as those described in [11]. As stated in the introduction, the problem of designing puncturing patterns (or content of the retransmissions) is not considered in this paper. Instead, retransmissions are constructed with randomly selected bits, and the performance of the RBHARQ algorithm is averaged over the selections.

In a system with incremental redundancy, the decoding is performed jointly by combining soft inputs from previous retransmissions with the soft inputs derived from the current retransmission. It is assumed that a maximum of J transmissions[1] of a packet can be jointly decoded. In the rest of the paper, a collection of transmissions decoded jointly will be referred to as an *attempt* to transmit a packet. If an attempt to transmit a packet fails, the accumulated soft inputs for that packet are discarded and the IR process is restarted on the next transmission. The attempts to transmit a packet are made until successful decoding of the packet.

Let j denote the transmission index, let Π_j denote the set of transmitted bits in the jth transmission, let N_j denote the size of the jth transmission, and let $\mathbf{z}_j$ denote the corresponding soft input vector obtained as in (1). For joint decoding, the soft input vectors are combined across transmissions to obtain $\tilde{\mathbf{z}}_j = \sum_{k=1}^{j} \mathbf{z}_k$, where $\tilde{\mathbf{z}}_j$ is the cumulative soft input vector for the jth transmission. Note that this addition corresponds to the maximal-ratio combining operation. The code rate R_j attained by combining the transmissions is given by

$$R_j = \frac{I}{\sum_{k=1}^{j} N_k}.$$

We define $\tilde{\Pi}_j$, the cumulative set of transmitted bits for the jth transmission, as the union of all transmitted bits in previous and current transmissions. That is, $\tilde{\Pi}_j = \cup_{k=1}^{j} \Pi_k$. In this paper, it is assumed that the bits for the transmissions are chosen uniformly from the subset of coded bits that have not been previously transmitted. That is, $\Pi_{j_1} \cap \Pi_{j_2} = \emptyset$ if $j_1 \neq j_2, R_{j_1} > R_C$ and $R_{j_2} > R_C$. Hence, transmissions are constructed according to the code combining retransmission strategy [3], whereby additional, previously not used, parity bits from the codeword are transmitted to accumulate redundancy at the receiver. Once the subset of bits not previously transmitted is exhausted, or equivalently all the bits from the codeword have been transmitted ($R_j \leq R_C$), the bits are chosen uniformly across the entire codeword. In that case, soft inputs are combined with the previously received copies of soft inputs in the corresponding bit positions, which is a diversity combining retransmission strategy [3].

A. Throughput and Delay Performance Measures

The user throughput η is defined as the average number of information bits per second transmitted successfully *while the channel is allocated to that user*. The user throughput in bits per second is given by

$$\eta = \frac{I}{T_{\text{trans}}} = \frac{I\,(1-p_J)}{T_s \left(N_1 + \sum_{j=2}^{J} N_j p_{j-1} \right)} \tag{2}$$

where T_{trans} is the average cumulative time duration of the packet transmissions until successful decoding, T_s is the symbol period, and p_j is the probability of decoding failure in transmissions 1 through j. That is,

$$\begin{aligned} p_j &= P\left[\bigcap_{k=1}^{j} \text{Decoding failure in the } k\text{th transmission} \right] \\ &\overset{(a)}{=} P\left[\text{Decoding failure in the } j\text{th transmission}\right] = \text{BLER}_j \end{aligned}$$

where BLER_j denotes the block error rate achieved after the jth transmission.[2]

The average packet delay, δ, is the expected time that elapses from the moment the packet is first transmitted over the channel to the moment the packet is successfully decoded. It is given by

$$\delta = T_{\text{trans}} + T_{\text{wait}} = T_s \frac{N_1 + \sum_{j=2}^{J} N_j p_{j-1}}{1-p_J} + \frac{T_r \sum_{j=1}^{J} p_j}{1-p_J} \tag{3}$$

where T_{trans} is defined above, T_{wait} is the average cumulative time between transmissions of a packet until successful decoding, and T_r is the time interval between transmissions. For simplicity, the delay expression is derived assuming that the interval between attempts to transmit a packet is equal to the time interval between transmissions.

III. Codeword Reliability and its Relationship to Code Rate and BLER

In this section, we provide a formal definition of the codeword reliability and characterize, via numerical simulations, the reliability to BLER and code rate to reliability mappings. Design of the RBHARQ algorithm and analytical results which approximate these mappings are presented in the following two sections.

We define the reliability of a codeword, μ, as the expected value of the magnitude of the *a posteriori* LLRs of the information bits, averaged over realizations of noise, sets of transmitted bits and channel taps at a fixed SNR. Following the jth transmission, the *a posteriori* LLR of the ith information bit is defined as

$$L_j(i) = \log \frac{P[b_i = 0 \mid \tilde{\mathbf{z}}_j]}{P[b_i = 1 \mid \tilde{\mathbf{z}}_j]}.$$

Note that the *a posteriori* LLR values for all information bits are available at the output of the MAP decoder. Following the jth transmission, codeword reliability μ_j is then given by

$$\mu_j = E_{\tilde{\Pi},n,\mathbf{h}} \left[\left| \log \frac{P[b_i = 0 \mid \tilde{\mathbf{z}}_j]}{P[b_i = 1 \mid \tilde{\mathbf{z}}_j]} \right| \right] \tag{4}$$

where the expectation is taken over noise realizations, the cumulative set of transmitted bits for the jth transmission and channel

[1]To simplify the terminology, in the rest of this paper, the term transmission encompasses the initial transmission of a packet as well as the subsequent retransmissions.

[2]Since incremental redundancy increases the reliability of the received codeword as the transmissions are combined at the receiver, we assume that a decoding success implies decoding success in all subsequent transmissions, if attempted by the transmitter. Taking the complement of this relationship gives equality (a).

taps.[3] In AWGN and static channel environments, the expectation with respect to channel taps is unnecessary. For a given sequence of information bit LLRs following the jth transmission, an estimate of the codeword reliability $\hat{\mu}_j$ can be obtained at the receiver by computing an empirical average[4]

$$\hat{\mu}_j = \frac{1}{I}\sum_{i=1}^{I}|L_j(i)|.$$

This computation can be performed at the receiver in static as well as fading channels.

Consider now a numerical experiment to investigate the reliability to BLER and code rate to reliability mappings. The experiment is performed for the convolutional code standardized for use in the EDGE cellular standard, with generator polynomial [145 171 133], rate $R_C = 1/3$, constraint length $\nu = 7$, and minimum distance $d_{\min} = 14$. A packet of length $I = 462$ information bits is encoded, which is one of the packet lengths specified in the EDGE standard. A sequence of transmissions of fixed size, $N_j = 51$ for all j, is sent over an AWGN channel ($A_i = 1$ for all i) at a specified SNR. The sets of transmitted bits are randomly constructed as described in Section II. As the transmissions are received, they are combined, the packet is decoded, and the estimated packet reliability $\hat{\mu}_j$ is recorded. A total of 40 transmissions per packet are simulated. For each packet, the reliability at which the packet is successfully decoded is also recorded. This experiment is repeated for several packet and noise realizations. The code rate to reliability mapping is obtained by computing average codeword reliability with respect to packet and noise realizations as a function of the code rate. The reliability to BLER mapping is obtained as the percentage of unsuccessfully decoded packets as a function of the packet reliability.

The code rate to reliability and reliability to BLER mappings are displayed in Figs. 1 and 2, respectively, for $\text{SNR} = -3$, 0, and 3 dB. The mappings were obtained by simulating 10 000 packet and noise realizations. For ease of exposition, the code rate to reliability mapping is displayed as a function of $1/R_j$, or the total number of bits contained in transmissions 1 through j normalized by the packet length. From Fig. 1, it is apparent that in the region $R_j > R_C$, where the transmissions consist of bits not previously transmitted, the reliability grows faster than in the region $R_j < R_C$, where the bits are repeated. This behavior is due to the superiority of code combining HARQ relative to diversity combining [3].

The numerical experiment just described was also performed for various fading channel models. For time-varying channels, the results were averaged over channel realizations, in addition to packet and noise realizations. The qualitative features of the reliability mappings displayed in an AWGN channel remain in fading channel conditions. Numerically, the rate to reliability mapping exhibits a slower growth rate in fading channel conditions, whereas the reliability to BLER mapping exhibits a slower decay of the block error rate. As an example, consider the code rate to reliability and reliability to BLER mappings generated for the Rayleigh block fading channel model also displayed in Figs. 1 and 2, respectively. The channel gain A_i is modeled as a Rayleigh distributed random variable, constant over a single transmission and independent across transmissions.

We denote by $\text{BLER} = F(\mu)$ the reliability to BLER mapping and by $\mu = G(R)$ the code rate to reliability mapping. Based on the numerical results, both mappings are one-to-one, so that the inverse mappings $\mu = F^{-1}(\text{BLER})$ and $R = G^{-1}(\mu)$ are well defined. Note that all mappings are functions of SNR, but this dependence, unless explicitly needed, is suppressed for notational simplicity.

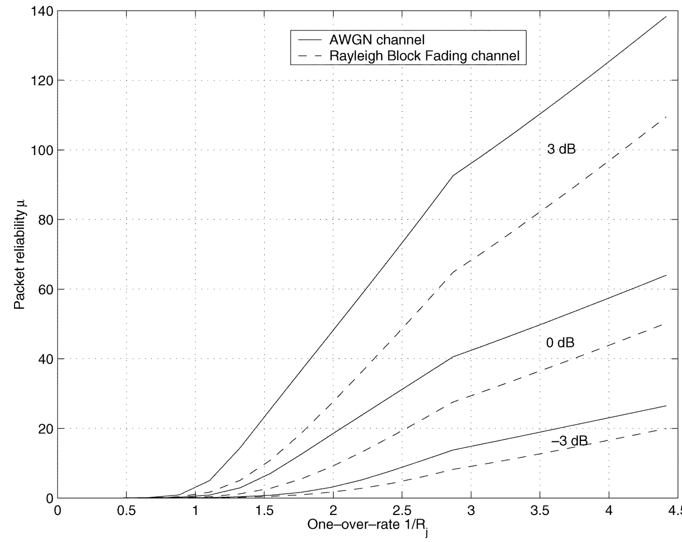

Fig. 1. Code rate to reliability mappings in AWGN and Rayleigh block fading channel models.

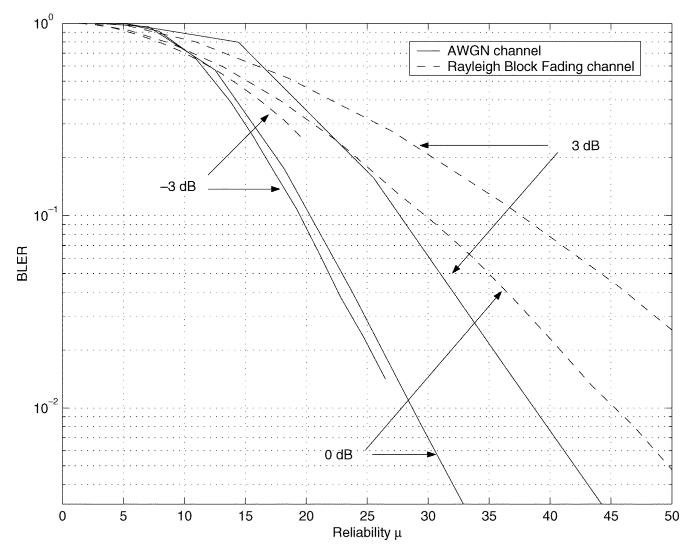

Fig. 2. Reliability to BLER mappings in AWGN and Rayleigh block fading channel models.

[3]It is assumed that the absolute values of the information bit LLRs, $|L_j(i)|$, are identically distibuted random variables, so that the expectation in (4) is not a function of the bit position i.

[4]Other reliability metrics are possible. An alternative reliability metric, suggested to the authors by P. Hoeher, is the LLR of the average of the information bit *a posteriori* probabilities computed over the packet.

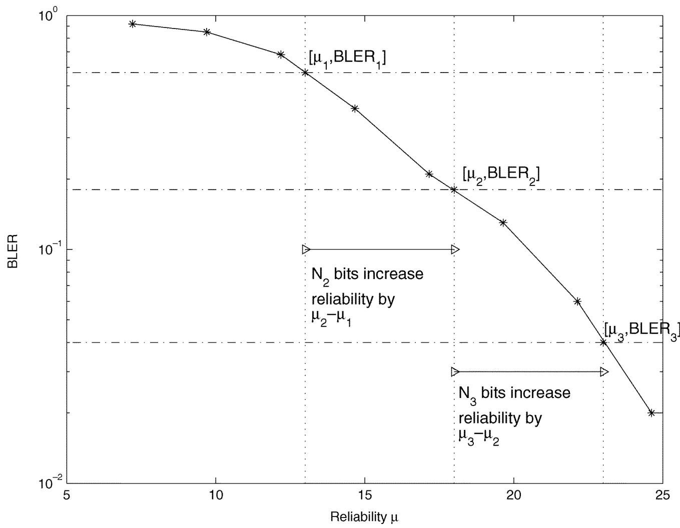

Fig. 3. Example of the the reliability to BLER mapping $F(\cdot)$.

IV. Reliability Based HARQ Schemes

In this section, we demonstrate how the reliability mappings $G(\cdot)$ and $F(\cdot)$ can be used to optimize the size of transmissions. Since the purpose of this work is not to design a link adaptation scheme, it is assumed that the size of the first packet transmission is determined by the MCS selection algorithm. Hence, we seek an RBHARQ algorithm, which adaptively optimizes sizes N_2 through N_J.

To begin, consider Fig. 3, which displays an example of the reliability to BLER mapping $F(\cdot)$. Suppose that after the first transmission, the packet is received with reliability $\hat{\mu}_1$, corresponding to a probability of decoding failure $\mathrm{BLER}_1 = F(\hat{\mu}_1)$. Transmitting additional coded bits and combining them with the initial transmission will raise the reliability to $\hat{\mu}_2$, corresponding to a probability of decoding failure BLER_2. Additional transmissions will continue to increase the reliability and move the probability of failure down the curve, as shown in Fig. 3. Thus, transmitting a block of coded bits of a certain size corresponds to sampling a point on the curve $F(\cdot)$, which represents a certain level of reliability and BLER. The goal of the reliability-based approach is to determine reliability levels for the transmissions, which minimize the use of channel resources.

To this end, we specify an RBHARQ scheme in terms of a vector of target reliabilities to be achieved in successive transmissions, $[\bar{\mu}_2, \bar{\mu}_3, \ldots, \bar{\mu}_J]$. Note that the target for the first transmission is not set, since it is determined by the MCS selection algorithm. Suppose a packet is in error after the jth transmission and has achieved an estimated reliability level of $\hat{\mu}_j$. The size of the next transmission is chosen to raise the packet's reliability to $\bar{\mu}_{j+1}$. Specifically, based on the inverse of the rate to reliability mapping $G^{-1}(\cdot)$, the size is computed as

$$N_{j+1} = I\left(\frac{1}{G^{-1}(\bar{\mu}_{j+1})} - \frac{1}{G^{-1}(\hat{\mu}_j)}\right). \quad (5)$$

The quantity $G^{-1}(\hat{\mu}_j)$ can be regarded as an estimate of the virtual code rate achieved with the first j transmissions. In a practical system, N_{j+1} would be further quantized to the nearest size from the set of permissible transmission sizes.[5] The index of the quantized size would then be signaled back to the transmitter. Note that the size computation in (5) adapts to the instantaneous channel conditions since it is performed in real time and it explicitly relies on the estimated reliability level $\hat{\mu}_j$. The size computation can also be performed nonadaptively with the recursion

$$N_{j+1} = \frac{I}{G^{-1}(\bar{\mu}_{j+1})} - \sum_{k=1}^{j} N_k. \quad (6)$$

In this case, given a vector of target reliabilities, the transmission sizes can be pre-computed and stored in the transmitter. As one would expect, the performance of the adaptive size computation is superior to the nonadaptive method in fading channel conditions. In static channel conditions, both size computation methods are equivalent. All RBHARQ performance results reported in this paper were generated with adaptive size computation.

The target reliabilities for the transmissions can be chosen to maximize user throughput. The throughput optimization problem can be formulated as

$$[\bar{\mu}_2, \bar{\mu}_3, \ldots, \bar{\mu}_J] = \arg \max_{\mu_2, \mu_3, \ldots, \mu_J} \frac{1 - F(\mu_J)}{N_1 + \sum_{j=2}^{J} N_j F(\mu_{j-1})} \quad (7)$$

where the throughput expression is obtained by rewriting (2) in terms of the reliability to BLER mapping and omitting unnecessary constants. For each candidate vector of target reliabilities, the sizes N_2 through N_J can be recursively computed from (6). Unfortunately, the optimization problem appears to have no closed form solution. For the numerical results reported in this paper, the throughput optimization was performed by quantizing the space of candidate vectors and using an exhaustive search to find the global maximum. To perform the search, only candidate vectors satisfying $0 < \mu_2 < \mu_3 < \cdots < \mu_J$ need to be considered. Note that a maximum packet delay constraint per attempt can be enforced by specifying the maximum number of transmissions per attempt and the maximum allowable transmission size. An alternative approach to the exhaustive search procedure, based on a dynamic programming formulation, is considered in [12].

Although not considered here, the problem formulation in (7) can be easily generalized to include the size of the initial transmission, N_1, as well as its modulation format, as additional optimization parameters. This generalized formulation provides a complete reliability-based solution to the link adaptation problem by optimally selecting the initial MCS, as well as the parameters of the HARQ scheme.

The computational complexity of the multitarget RBHARQ scheme, specified by (7), motivates us to consider a simple RBHARQ scheme with a single target reliability $\bar{\mu}$. The target reliability is chosen to achieve a desired probability of decoding error

[5]It is possible for the estimate of the codeword reliability $\hat{\mu}_j$ to exceed the target reliability for the next transmission $\bar{\mu}_{j+1}$ on a sample-by-sample basis. In this case, the computation in (5) results in a negative number. In a practical system, this number would be mapped by the quantizer to the the smallest permissible transmission size.

$\mathrm{BLER} = F(\bar{\mu})$. For instance, the desired residual block error rate, BLER_J can be used to set the target reliability. The transmission sizes are adaptively determined according to (5), with $\bar{\mu}_2 = \bar{\mu}_3 = \cdots = \bar{\mu}_J = \bar{\mu}$. In the event that the target reliability is achieved, but the packet has not been successfully decoded, the transmitter continues to transmit minimum-length transmissions until the packet is successfully decoded or the maximum number of transmissions in an attempt is reached. The single target scheme can be thought of as a "greedy" algorithm which attempts to achieve the final reliability level in a single step. This is in contrast to the multitarget scheme which reaches the final reliability in a series of optimized steps.

Implementation of an RBHARQ scheme in a practical communication system raises numerous design issues which we can only begin to address in this paper. We comment that both the multitarget and the single target RBHARQ schemes require knowledge of the $G(\cdot)$ and $F(\cdot)$ reliability mappings consistent with the prevalent channel environment observed at the receiver. Once the appropriate $G(\cdot)$ and $F(\cdot)$ mappings are identified, a desired RBHARQ scheme can be readily implemented. As defined, the reliability mappings depend only on the long-term statistics of the channel, such as long-term SNR, power-delay-profile and Doppler frequency. In practice, similar to the implementation of MCS selection tables for link adaptation, the space of long-term channel parameters could be quantized, and the reliability mappings could be pre-computed with the quantized set of channel parameters and stored in the receiver. The receiver would periodically estimate the quantized channel parameters and look-up the corresponding $G(\cdot)$ and $F(\cdot)$ mappings. In principle, it is possible to use different $G(\cdot)$ and $F(\cdot)$ mappings in each transmission attempt. However, in most channel conditions, updating the reliability mappings on a slower time scale should be sufficient. The choice of the channel parameters to be used for indexing the mappings, as well as the specifics of updating the mappings, depend on a particular system implementation. As a final remark on implemenatation, note that the assumption, made in this paper, of perfect knowledge of the causal channel state information by the receiver does not imply that size adaptation based on the transmission's instantaneous channel conditions is possible, as this would require knowledge of future channel realizations by the receiver. Although an estimate of future channel realizations can be obtained using channel prediction techniques, this is outside the scope of the present paper.

A. Numerical Results

In this section, simulation results comparing the performance of the proposed single target and multitarget RBHARQ schemes with the retransmission strategy defined in the EDGE standard are presented. For comparison, the following EDGE parameters are used for all simulated schemes: $J = 3$, $N_1 = 612$. The results are obtained with the EDGE convolutional code with parameters defined in Section III. For the EDGE HARQ scheme, we have $N_2 = N_3 = 612$. For the simulated RBHARQ schemes, the following set of eight allowable sizes for the transmissions is used: N_2 and $N_3 \in [78, 156, 234, 312, 390, 468, 546, 612]$. Note that only 3 bits of feedback are required to specify one of 8 possible sizes. Since the number of transmissions per attempt and the maximum allowable size are the same for the RBHARQ and EDGE HARQ schemes, the maximum packet delays for all schemes are equal. For simplicity, no specific scheduling strategy is adopted. Instead, all simulations are performed with the assumption that the average time interval between transmissions is $T_r = 2500T_s$. That is, the retransmission delay equals 2500 symbol intervals. Recall that it is assumed that the interattempt delays equal the retransmission delays within an attempt. For all schemes, the transmissions are constructed with randomly selected bits, as described in Section II.

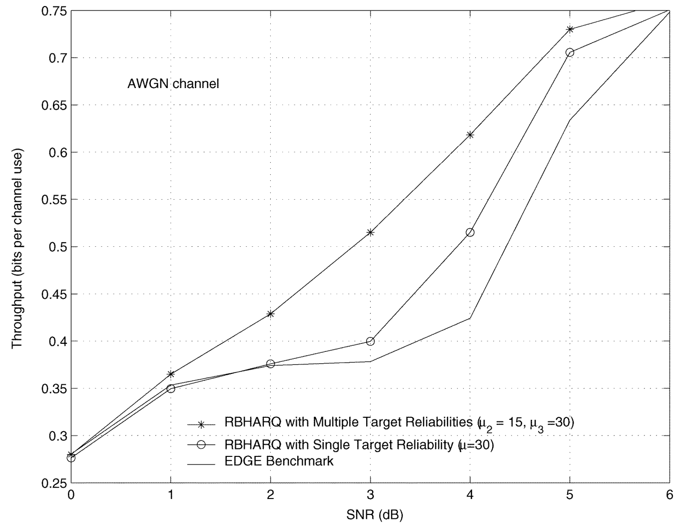

Fig. 4. Performance comparison in AWGN channel.

Fig. 4 compares the throughput performance of the RBHARQ schemes with multiple target reliabilities and a single target reliability to that of the EDGE ARQ scheme in an AWGN channel. For this simulation, reliability targets of $\bar{\mu}_2 = 15$ and $\bar{\mu}_3 = 30$ were used for the multitarget scheme, and $\bar{\mu} = 30$ for the single target scheme. For the multitarget scheme, the targets were obtained by numerically solving the optimization problem in (7). The throughput and delay performance of all schemes were obtained by evaluating (2) and (3), respectively. Observe that the multitarget RBHARQ scheme provides up to 45% improvement in throughput at $\mathrm{SNR} = 4$ dB relative the EDGE HARQ scheme. As expected, the multitarget scheme, with targets obtained from throughput optimization, significantly outperforms its single target counterpart, which is based on the "greedy" strategy. At high SNRs, when retransmissions are rare, all schemes have the same performance since the sizes of the initial transmissions for all schemes are equal. At low SNRs, the packet reliability of the initial transmission is so low that both RBHARQ schemes request retransmissions of the largest size, which coincides with the retransmission size of the EDGE scheme, thereby making all schemes equivalent. The average packet delays for all schemes are essentially equal.

A histogram of the retransmission sizes N_2 and N_3 utilized by the multitarget RBHARQ scheme in an AWGN channel at $\mathrm{SNR} = 3$ dB is provided in Fig. 5. As evident in the figure, N_2 tends to be slightly larger than N_3. Furthermore, note that all utilized sizes are less than the maximum retransmission size of

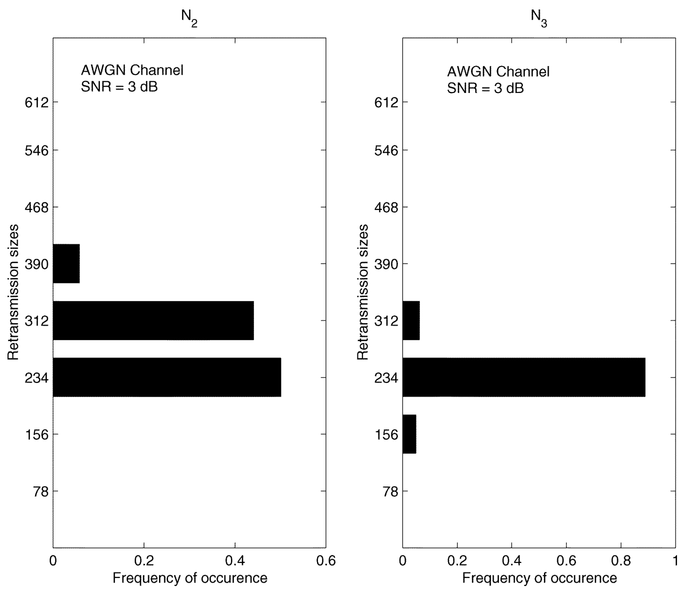

Fig. 5. Example of sizes N_2 and N_3 picked by the multitarget RBHARQ scheme in an AWGN channel, $\mathrm{SNR} = 3$ dB.

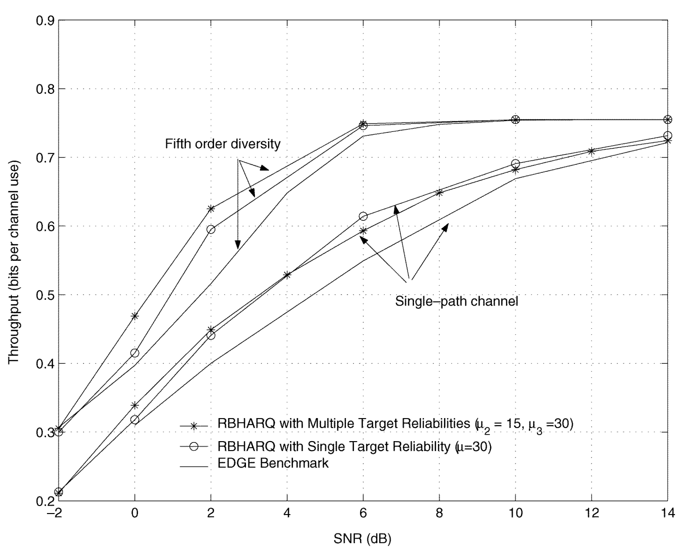

Fig. 6. Performance comparison in block fading channels.

612 bits used by the EDGE HARQ scheme, which results in the significant throughput gain displayed in Fig. 4.

Fig. 6 displays the performance of the proposed RBHARQ schemes and the EDGE RBHARQ scheme in a block fading channel environment. We consider first ($P = 1$) and fifth-order ($P = 5$) diversity channel models in order to demonstrate the effects of SNR variations on the performance of the RBHARQ schemes. For these channel models, each channel tap is an independent circularly-symmetric complex Gaussian random variable, constant across a single transmission and independent across transmissions. For the fifth-order diversity channel model, the variance of each tap is set to 1/5. From Fig. 6 it is evident that the RBHARQ scheme with multiple target reliabilities does not offer much improvement over the performance of the single target scheme in a channel with first-order diversity. This indicates that the multitarget RBHARQ scheme is not robust in conditions of mismatch between the instantaneous and the long-term average SNR, such as the single-tap Rayeigh block fading channel. Both RBHARQ schemes support a throughput that is up to 12% higher than that of the EDGE HARQ scheme. The channel with fifth-order diversity exhibits significantly less SNR variation. In this case, the RBHARQ scheme with multiple target reliabilities performs better than the single target reliability scheme and supports a throughput that is up to 20% greater than that achieved by the EDGE retransmission strategy. In general, the multitarget RBHARQ scheme is more suitable for channels with small SNR deviations from the average, such as AWGN or high-diversity channels, as the reliability targets used in the multitarget scheme are outputs of the optimization procedure based on the average SNR. The user packet delay of the RBHARQ schemes in each scenario considered is no more than 5% higher than that achieved by the EDGE HARQ scheme.

V. Analysis of the Reliability Mappings

Developing precise analytical expressions for the reliability mappings $G(\cdot)$ and $F(\cdot)$ appears to be an intractable problem. Instead, we present an approximation to the $G(\cdot)$ mapping and upper and lower bounds on the $F(\cdot)$ mapping. The accuracy of the approximation and bounds is verified via simulations. The approximation to $G(\cdot)$ is developed for an AWGN channel and then extended to certain fading channel models. The bounds on $F(\cdot)$ are valid for any channel model. The computational complexity of the Monte Carlo simulations required for evaluating the approximation to $G(\cdot)$ and bounds on $F(\cdot)$ is much less than the computational complexity of the purely empirical approach for evaluating the mappings, as described in Section III.

A. Rate to Reliability Mapping

Consider an AWGN channel and define $A \equiv A_i$ for all i as the channel gain. Similarly to $\tilde{\mathbf{z}}_j$, also define $\tilde{\mathbf{r}}_j$ as $\tilde{\mathbf{r}}_j = \sum_{k=1}^{j} \mathbf{r}_k$, where $\tilde{\mathbf{z}}_j = A\tilde{\mathbf{r}}_j$. The codeword reliability μ_j, following the jth transmission in an AWGN channel, is then given by

$$\begin{aligned}\mu_j = E_{\tilde{\Pi},n}\left[|L_j(i)|\right] &= E_{\tilde{\Pi},n}\left[\left|\log \frac{P[b_i = 0 \mid \tilde{\mathbf{z}}_j]}{P[b_i = 1 \mid \tilde{\mathbf{z}}_j]}\right|\right] \\ &= E_{\tilde{\Pi},n}\left[\left|\log \frac{P[b_i = 0 \mid \tilde{\mathbf{r}}_j]}{P[b_i = 1 \mid \tilde{\mathbf{r}}_j]}\right|\right]. \end{aligned} \quad (8)$$

We now proceed to approximate this expression in the region $R_j \geq R_C$. A similar approximation can be developed for $R_j \leq R_C$, and for brevity we omit details. Recall that for $R_j \geq R_C$, the transmissions are constructed from a set of bits not previously transmitted. Hence, the probability of observing a particular cumulative set of transmitted bits for the jth transmission is given by $\tilde{N}_j!\left(N - \tilde{N}_j\right)!/N!$, where $\tilde{N}_j = \sum_{k=1}^{j} N_k$.

We begin by assuming, without loss of generality, that $b_i = 0$. With this assumption, a received information "0" is more likely than a received information "1," so that the value of the LLR in (8) is likely to be positive and the absolute value in (8) can be dropped.

In an AWGN channel, applying Bayes rule and after some algebra, $L_j(i)$ can be expressed as

$$\begin{aligned} L_j(i) &= \log \frac{\sum_{\mathbf{s}:b_i=0} P[\tilde{\mathbf{r}}_j \mid \mathbf{s}]}{\sum_{\mathbf{s}:b_i=1} P[\tilde{\mathbf{r}}_j \mid \mathbf{s}]} = \log \frac{\sum_{\mathbf{s}:b_i=0} e^{-\|\tilde{\mathbf{r}}_j - A\mathbf{s}\|^2/2\sigma^2}}{\sum_{\mathbf{s}:b_i=1} e^{-\|\tilde{\mathbf{r}}_j - A\mathbf{s}\|^2/2\sigma^2}} \\ &= \log \frac{\sum_{\mathbf{s}:b_i=0} e^{2A\langle \tilde{\mathbf{r}}_j, \mathbf{s}\rangle/2\sigma^2}}{\sum_{\mathbf{s}:b_i=1} e^{2A\langle \tilde{\mathbf{r}}_j, \mathbf{s}\rangle/2\sigma^2}} = \log \frac{1 + \sum_{\mathbf{s}:b_i=0,\, \mathbf{s}\neq \mathbf{s}_0} e^{2A\langle \tilde{\mathbf{r}}_j, \mathbf{s}-\mathbf{s}_0\rangle/2\sigma^2}}{\sum_{\mathbf{s}:b_i=1} e^{2A\langle \tilde{\mathbf{r}}_j, \mathbf{s}-\mathbf{s}_0\rangle/2\sigma^2}} \\ &\overset{(a)}{\approx} -\log \sum_{\mathbf{s}:b_i=1} e^{2A\langle \tilde{\mathbf{r}}_j, \mathbf{s}-\mathbf{s}_0\rangle/2\sigma^2} \end{aligned}$$

where $\langle \mathbf{a}, \mathbf{b}\rangle$ denotes inner-product between vectors $\mathbf{a}$ and $\mathbf{b}$, and $\mathbf{s}_0 = 2\mathbf{c}_0 - 1$, where $\mathbf{c}_0$ denotes a codeword being communicated to the receiver, and $\mathbf{s}_0$ denotes the corresponding vector of antipodal symbols. Approximation (a) follows since in the SNR range of interest the second summand in the numerator is much less than unity. As a result of the preceding simplifications, the codeword reliability can be approximated as

$$\mu_j \approx -E_{\tilde{\Pi},n}\left[\log \sum_{\mathbf{s}:b_i=1} e^{2A\langle \tilde{\mathbf{r}}_j, \mathbf{s}-\mathbf{s}_0\rangle/2\sigma^2}\right]. \tag{9}$$

To approximate the expectation in (9), we note that only $\tilde{\mathbf{r}}_j$ depends on a noise realization. We then proceed with the approximation by taking the expectation inside the function and approximating $\tilde{\mathbf{r}}_j$ by its mean.[6] We have $E_n[\tilde{\mathbf{r}}_j] = A\mathbf{s}_0 \times \mathbf{1}_{\tilde{\Pi}_j}$, where $\times$ denotes component-by-component multiplication, and $\mathbf{1}_{\tilde{\Pi}_j}(i) = 1$ if $i \in \tilde{\Pi}_j$ and 0 otherwise. Substituting the mean into (9), we obtain

$$\begin{aligned} \mu_j &\approx -E_{\tilde{\Pi}}\left[\log \sum_{\mathbf{s}:b_i=1} e^{2A^2\langle \mathbf{s}_0 \times \mathbf{1}_{\tilde{\Pi}_j}, \mathbf{s}-\mathbf{s}_0\rangle/2\sigma^2}\right] \\ &= -E_{\tilde{\Pi}}\left[\log \sum_{\mathbf{c}:b_i=1} e^{-4\,\mathrm{SNR}\,\langle \mathbf{1}_{\tilde{\Pi}_j}, \mathbf{c}\rangle}\right] \\ &\approx 4\,\mathrm{SNR}\, E_{\tilde{\Pi}}\left[\min_{\mathbf{c}:b_i=1} \langle \mathbf{1}_{\tilde{\Pi}_j}, \mathbf{c}\rangle\right] \\ &\overset{(a)}{\approx} 4\,\mathrm{SNR}\, E_{\tilde{\Pi}}\left[\min_{\mathbf{c}:b_i=1,\, w(\mathbf{c})\leq \bar{d}} \langle \mathbf{1}_{\tilde{\Pi}_j}, \mathbf{c}\rangle\right] \end{aligned} \tag{10}$$

where $w(\mathbf{c})$ denotes the Hamming weight of the codeword. Approximation (a) in (10) follows since it is likely that only codewords below a certain weight contribute to the minimum. The choice of $\bar{d}$ is a computational parameter. The accuracy of the approximation improves with increasing $\bar{d}$.

Finally, we observe that under our statistical model for $\tilde{\Pi}_j$, the inner-product $\langle \mathbf{1}_{\tilde{\Pi}_j}, \mathbf{c}\rangle$ is a hypergeometric random variable, χ, with parameters $\chi\left(N, \tilde{N}_j, w(\mathbf{c})\right)$ [15]. Note that these parameters can be expressed in terms of R_j. We now proceed by approximating the expectation in (10) as an expectation of a minimum of a set of independent hypergeometric random variables. Furthermore, corresponding to codewords of a certain weight d, precisely I_d hypergeometric random variables need to be included in the set, where I_d is the total information weight of all codewords of Hamming weight d. Let $d_1, \ldots, d_k$ denote all Hamming weights contained in the code that are less than or equal to $\bar{d}$. The final approximation then becomes

$$\mu_j \approx 4\mathrm{SNR} E_\chi\left[\min\left\{\chi_1^{I_{d_1}}(N, \tilde{N}_j, d_1), \ldots, \chi_1^{I_{d_k}}(N, \tilde{N}_j, d_k)\right\}\right] \tag{11}$$

where a_1^n denotes a sequence a_1 through a_n. The final approximation is a function only of the distance spectrum of the code and of the code rate. It can be evaluated numerically via a Monte Carlo simulation. A numerical example is given in Section V-C.

If no puncturing is performed, that is, if decoding is performed at full rate, the approximation in (11) simplifies to $\mu_{\text{full rate}} \approx 4$ SNR $d_{\min}$, where $d_{\min}$ is the minimum distance of the code. A somewhat better approximation for $\mu_{\text{full rate}}$ is obtained by taking into account the fact that the number of the minimum weight sequences in the set $\{\mathbf{c} : b_i = 1,\ w(\mathbf{c}) = d_{\min}\}$ is given by $I_{d_{\min}}$. Then, the approximation becomes $\mu_{\text{full rate}} \approx 4\,\mathrm{SNR}\, d_{\min} - \log I_{d_{\min}}$.

The approximation in (11) can be readily extended to a fast fading channel model. Under this model, the channel amplitudes A_i, $1 \leq i \leq N_j$, are independent, identically distributed random variables. In this case, each hypergeometric term $\chi(N, \tilde{N}_j, d)$ in (11) is replaced with a random sum

$$\Psi(N, \tilde{N}_j, d) = \sum_{k=1}^{\chi(N,\tilde{N}_j,d)} A_k^2.$$

The final approximation then becomes

$$\mu_j \approx 4\,\mathrm{SNR}\, E_{A,\chi} \times \left[\min\left\{\Psi_1^{I_{d_1}}(N, \tilde{N}_j, d_1), \ldots, \Psi_1^{I_{d_k}}(N, \tilde{N}_j, d_k)\right\}\right] \tag{12}$$

where the expectation is computed with respect to both the channel amplitudes and the hypergeometric random variables. The accuracy of (12) is examined in Section V-C. Extension of (11) to block fading channel models does not appear to be tractable.

B. Reliability to BLER Mapping

Consider a sequence of *a posteriori* LLR values $L_j(1)$ through $L_j(I)$ obtained at the output of the MAP decoder after the jth transmission. To obtain estimates of the information bits, the receiver hard-slices the values and declares a block decoding error if at least one of the bit estimates $\hat{b}_i$ is erroneous. We are interested in the μ_j to BLER_j mapping. Deriving an exact expression for the mapping is cumbersome, as the bit estimates are correlated across the packet. Instead, expressions relating μ_j to an upper and lower bound on BLER_j are derived.

[6] Note that by Jensen's inequality $E_x\left[-\log \sum_i e^{x_i}\right] < -\log \sum_i e^{E_x[x_i]}$.

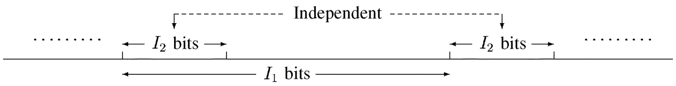

Fig. 7. Partition of the transmitted sequence into successive groups, so that the first I_2 information bits in each group are independent of those in other groups.

To this end, conditioned on the transmitted bit sequence, we model the *a posteriori* LLR values as Gaussian-distributed random variables with mean μ_j and variance σ_j^2 [16].

We proceed by noting the following key relationship from [17]: $P[\hat{b}_i \neq b_i] = 1/(1+e^{|L_j(i)|})$. Then, the BLER can be upper bounded by assuming independent error events across the packet. Namely

$$\begin{aligned} \mathrm{BLER}_j &\leq 1 - \prod_{i=1}^{I}\left(1 - P[\hat{b}_i \neq b_i]\right) \\ &= 1 - \exp\left\{-\sum_{i=1}^{I}\log(1+e^{-|L_j(i)|})\right\}. \end{aligned} \quad (13)$$

By the Strong law of large numbers, for long block lengths, we approximate

$$\begin{aligned} &\frac{1}{I}\sum_{i=1}^{I}\log(1+e^{-|L_j(i)|}) \\ &\qquad \to E_{|L|}\left[\log(1+e^{-|L_j(i)|})\right] \approx E_{|L|}\left[e^{-|L_j(i)|}\right] \end{aligned}$$

where the expectation is taken with respect to the distribution of $|L_j(i)|$.

The upper bound (13) can then be approximated by $\mathrm{BLER}_j \leq 1 - \exp\left\{-I \cdot E_{|L|}\left[e^{-|L_j(i)|}\right]\right\}$. Using the Gaussian assumption for the conditional distribution of $L_j(i)$, after some algebra, we can show that

$$E_{|L|}\left[e^{-|L_j(i)|}\right] \approx \frac{\sqrt{\frac{2}{\pi}}}{\sigma_j} e^{-\mu_j^2/2\sigma_j^2}.$$

To obtain a lower bound on BLER_j, suppose we divide the information block b_1 through b_I into groups of equal length, with each group containing I_1 bits. Consider the first I_2 bits in each group, with $I_2 \ll I_1$. For large enough I_1, it is reasonable to assume that the LLR values corresponding to these I_2 bits are independent of the LLR values corresponding to the first I_2 bits in the neighboring groups. Fig. 7 illustrates the division of the information block. Note that there are a total of I/I_1 groups. Letting index l_k denote the lth bit in the kth group, where $1 \leq l \leq I_2$ and $1 \leq k \leq I/I_1$, we have

$$\begin{aligned} 1 - \mathrm{BLER}_j &= P\left[\bigcap_{i=1}^{I} b_i = \hat{b}_i\right] \\ &\leq P\left[\bigcap_{k=1}^{I/I_1}\bigcap_{l=1}^{I_2} b_{l_k} = \hat{b}_{l_k}\right] \\ &= \prod_{k=1}^{I/I_1} P\left[\bigcap_{l=1}^{I_2} b_{l_k} = \hat{b}_{l_k}\right]. \end{aligned}$$

Furthermore, for any group k

$$\begin{aligned} P\left[\bigcap_{l=1}^{I_2} b_{l_k} = \hat{b}_{l_k}\right] &\leq \frac{1}{I_2}\sum_{l=1}^{I_2} P\left[b_{l_k} = \hat{b}_{l_k}\right] \\ &= \frac{1}{I_2}\sum_{l=1}^{I_2}\frac{1}{1+e^{-|L_j(l_k)|}}. \end{aligned}$$

Combining the preceding two inequalities, we obtain the following lower bound:

$$\mathrm{BLER}_j \geq 1 - \prod_{k=1}^{I/I_1}\frac{1}{I_2}\sum_{l=1}^{I_2}\frac{1}{1+e^{-|L_j(l_k)|}}.$$

Applying the Strong Law of Large Numbers with $I/I_1 \to \infty$, and after some algebra, the lower bound can be simplified to

$$\mathrm{BLER}_j \geq 1 - \exp\left\{-I \cdot \frac{1}{I_1} E_{|L|}\left[e^{-|L|_j(i)}\right]\right\}.$$

Parameter I_1 is code-specific. It should be chosen large enough so that the independence assumption between neighboring groups of bits is satisfied. However, too large a value for I_1 results in an underestimation of BLER_j. Simulations have shown that taking $I_1 = 2\nu$ is appropriate, where ν is the code constraint length.

To summarize, the BLER can be upper and lower bounded as follows:

$$\begin{aligned} &1 - \exp\left\{-I \cdot \frac{1}{I_1} E_{|L|}\left[e^{-|L|_j(i)}\right]\right\} \\ &\qquad \leq \mathrm{BLER}_j \leq 1 - \exp\left\{-I \cdot E_{|L|}\left[e^{-|L_j(i)|}\right]\right\} \end{aligned} \quad (14)$$

where

$$E_{|L|}\left[e^{-|L_j(i)|}\right] \approx \frac{\sqrt{\frac{2}{\pi}}}{\sigma_j} e^{-\mu_j^2/2\sigma_j^2}.$$

The upper and lower bounds relate BLER_j to both the codeword reliability μ_j and the variance of the LLR values σ_j^2. Simulations have shown that the ratio μ_j/σ_j^2 falls within a fairly narrow range of values. It is therefore convenient to express BLER_j in terms of μ_j only by averaging with respect to σ_j^2 for fixed μ_j. The accuracy of the resulting one-dimensional mappings is examined in the following section.

C. Numerical Examples

In this section, the approximations to $G(\cdot)$ in (11) and (12) and the upper and lower bounds on $F(\cdot)$ given by (14) are compared to the corresponding reliability mappings generated according to the numerical procedure described in Section III. Comparison of the approximation to the empirically obtained $G(\cdot)$ for AWGN and Rayleigh fast fading channel models, for $\mathrm{SNR} = -3$, 0, and 3 dB, is displayed in Fig. 8. The approximation was computed with $\bar{d} = 18$. As evident from the figure, its accuracy improves with increasing SNR and decreasing code

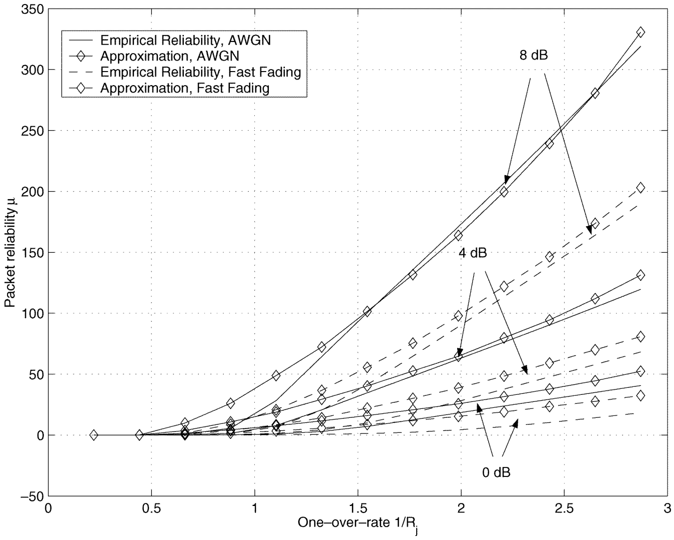

Fig. 8. Comparison of simulated $G(\cdot)$ with the approximations in (11) and (12).

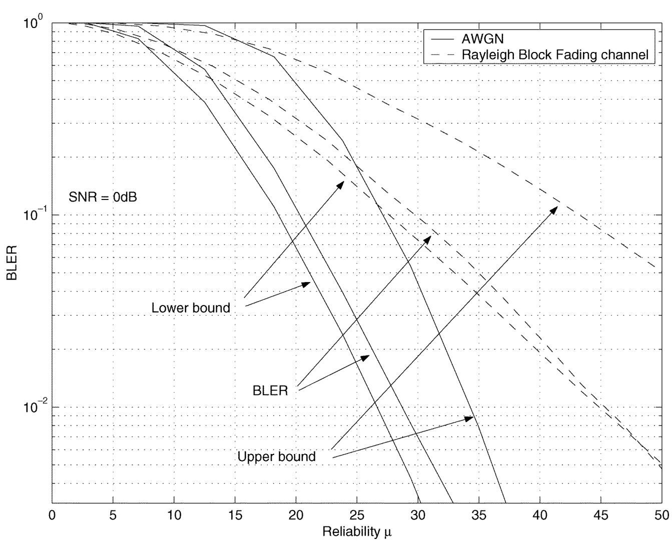

Fig. 9. Comparison of simulated $F(\cdot)$ with the upper and lower bounds in (14).

rate. This trend is to be expected since (11) is based on minimum distance arguments and becomes asymptotically tight with increasing SNR and decreasing code rate. The accuracy in the low SNR and high code rate regions can be improved by increasing the value of $\bar{d}$, thereby including more higher weight codewords into the approximation.

The upper and lower bounds on the empirically measured $F(\cdot)$ are displayed in Fig. 9 for AWGN and Rayleigh block fading channel models at $\mathrm{SNR} = 0$ dB. The bounds were obtained by evaluating the expressions in (14) with the empirically measured reliability and variance values and then, for a fixed value of reliability, averaging over the variance. For both channel models, the resulting upper bound is fairly loose in the performance region of interest ($10^{-1} < \mathrm{BLER}_j < 10^{-2}$). The lower bound provides a reasonably accurate approximation to BLER in that region.

VI. Conclusion

Based on the notion of codeword reliability, and its relationship to code and block error rates, a hybrid ARQ algorithm for communication systems employing convolutional coding has been proposed. The algorithm adaptively computes the sizes of the ARQ transmissions with the goal of optimizing throughput. Several variants of the algorithm have been presented of varying complexity and resiliency against fading. The proposed schemes have the potential of attaining large throughput gains relative to the standard hybrid ARQ schemes with little increase in delay or feedback channel utilization. An approximation to the codeword reliability as a function of the code rate and bounds on the BLER as a function of the reliability were presented. The approximation and bounds are tied to the spectral properties of the convolutional code and first and second order moments of the soft outputs of the decoder. The results are semi-analytic and provide a computationally efficient way of approximating or bounding the respective reliability mappings.

References

[1] V. Tripathi, E. Visotsky, R. Peterson, and M. L. Honig, "Reliability-based type II hybrid ARQ schemes," in *Proc. IEEE Int. Conf. Communications*, vol. 1, Anchorage, AK, May 2003, pp. 2899–2903.

[2] *Digital Cellular Telecommunications System (Phase 2+); Multiplexing and Multiple Access on the Radio Path*, 8.5.0 ed., GSM, 1999.

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

[4] "Technical specification group access radio network; high speed downlink packet access: physical layer aspects,", 3GPP TSG RAN123, 2002.

[5] Y. W. Blankenship and B. K. Classon, "Bounding techniques applied to modulation and coding design," presented at the 3rd Generation Wireless and Beyond, San Francisco, CA, May 2002.

[6] A. Das, F. Khan, and S. Nanda, "A2IR: An asynchronous and adaptive hybrid ARQ scheme for 3G evolution," in *Proc. VTC*, vol. 1, Rhodes, Greece, 2001, pp. 628–632.

[7] F. Khan, H. Olofsson, A. Furuskar, and M. Hook, "Method and system for measurement based automatic retransmission request in a radio communication system," U.S. Patent Application 20 010 056 560.

[8] J. M. Shea, "Reliability-based hybrid ARQ," *Electron. Lett.*, vol. 38, no. 13, pp. 644–645, Jun. 2002.

[9] A. Roongta and J. M. Shea, "Reliability-based hybrid ARQ using convolutional codes," in *IEEE Int. Conf. Communications*, Anchorage, AK, May 2003, pp. 2889–2893 .

[10] J. Cheng, Y. E. Wang, and S. Parkvall, "Adaptive incremental redundancy," in *Proc. 2003 IEEE Vehicular Technology Conf.*, Orlando, FL, Oct. 2003, pp. 737–741.

[11] J. Haguenauer, "Rate compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. 36, no. 4, pp. 389–400, Apr. 1988.

[12] E. Visotsky, V. Tripathi, and M. Honig, "Optimum ARQ design: A dynamic programming approach," in *Proc. IEEE Int. Symp. Information Theory*, Yokohama, Japan, Jun. 2003, p. 451.

[13] E. Uhlemann, L. K. Rasmussen, A. Grant, and P.-A. Wiberg, "Optimal incremental redundancy strategy for type-II hybrid ARQ," in *IEEE Int. Symp. Information Theory*, Yokohama, Japan, Jun. 2003, p. 448.

[14] G. Caire and D. Tuninetti, "The throughput of hybrid-ARQ protocols for the Gaussian collision channel," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1971–1987, Jul. 2001.

[15] G. Casella and R. L. Berger, *Statistical Inference*. Belmont, CA: Duxbury.

[16] H. E. Gamal and A. R. Hammons Jr, "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 671–686, Feb. 2001.

[17] P. Hoeher, I. Land, and U. Sorger, "Log-likelihood values and Monte Carlo simulation – Some fundamental results," in *Proc. Int. Symp. Turbo Codes and Related Topics*, Brest, France, Sep. 2000, pp. 43–46.

Eugene Visotsky (M'00) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Illinois at Urbana-Champaign in 1995, 1997, and 2000, respectively.

He joined the Communication Systems Research Laboratories, Motorola Laboratories, Schaumburg, IL, in 2000 where he has worked on various problems in the area of cellular communication systems. His current areas of interest are dynamic spectrum allocation and *ad hoc* cooperative relay radio systems.

Yakun Sun (S'02–M'05) received the B.S. degree from Tsinghua University, Beijing, China, in 1999. He received the M.S. and Ph.D. degrees in electrical engineering from Northwestern University, Evanston, IL, in 2001 and 2004, respectively.

He is currently with Mobile Devices, Motorola, Libertyville, IL, where he works on physical layer algorithms, and wireless systems analysis and design. His research interests include multiantenna/multicarrier wireless systems, channel estimation and equalization, iterative receivers, and resource allocation.

Vinayak Tripathi (S'99–M'01) received the B.S. degree from Cornell University, Ithaca, NY, in 1997 and the M.S. degree from the University of Michigan at Ann Arbor in 1998, both in electrical engineering. He is currently working toward the Ph.D. degree in financial economics at Princeton University, Princeton, NJ.

He worked as a Research Engineer at the Communication Systems Research Laboratory, Motorola Laboratories, Schaumburg, IL, until August 2003.

Michael L. Honig (S'80–M'81–SM'92–F'97) received the B.S. degree in electrical engineering from Stanford University, Stanford, CA, in 1977, and the M.S. and Ph.D. degrees in electrical engineering from the University of California, Berkeley, in 1978 and 1981, respectively.

He subsequently joined Bell Laboratories in Holmdel, NJ, where he worked on local area networks and voiceband data transmission. In 1983, he joined the Systems Principles Research Division at Bellcore, where he worked on digital subscriber lines and wireless communications. He was a Visiting Lecturer at Princeton University, Princeton, NJ, during the Fall of 1993. Since the Fall of 1994, he has been with Northwestern University, Evanston, IL, where he is a Professor in the Electrical and Computer Engineering Department. He was a Visiting Lecturer at Princeton University, Princeton, NJ, during the Fall of 1993, a visiting Mackay Professor at the University of California, Berkeley, during the Fall of 2000, and a Visiting Scholar at the University of Sydney, Sydney, Australia, during the summer of 2001.

Dr. Honig has served as an editor for the IEEE TRANSACTIONS ON INFORMATION THEORY (1998–2000) and the IEEE TRANSACTIONS ON COMMUNICATIONS (1990–1995), and was a Guest editor for the *European Transactions on Telecommunications* and *Wireless Personal Communications*. He has also served as a Member of the Digital Signal Processing Technical Committee for the IEEE Signal Processing Society, and as a Member of the Board of Governors for the Information Theory Society (1997–2002). He was the co-recipient of the 2002 IEEE Communications Society and Information Theory Society Joint Paper Award.

Roger Peterson (M'68–SM'97–F'98) received the B.S., M.S., and Ph.D. degrees from the Illinois Institute of Technology in Chicago, in 1967, 1968, and 1978, respectively.

From 1968 to 1978, he worked on radar guidance systems at IIT Research Institute, Chicago. From 1978 to 1980, he worked at Bell Laboratories doing studies related to cellular radio systems. He joined Motorola's Government Electronic Group in 1980 where he worked on spread spectrum communications systems. In 1992 he joined Motorola's Corporate Research Laboratories, Schaumburg, IL, where he has worked on cellular systems analysis and simulations. His current area of interest is *ad hoc* cooperative relay radio systems. He is coauthor of three textbooks on digital communications and spread spectrum systems.

Dr. Peterson was Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS from 1993 to 1996.