Modified Majority-Logic Decoders for Use in Convolutionally Encoded Hybrid-ARQ Systems

STEPHEN B. WICKER

Abstract—**This paper demonstrates a method for modifying majority-logic convolutional decoders for use in type-I hybrid-ARQ protocols. Majority-logic decoders generate reliability information in the form of orthogonal sets of parity check sums. The modified decoder uses this information to identify received packets whose decoded data may be unreliable and to request their retransmission.**

I. Introduction: Hybrid-ARQ Schemes Using Convolutional Codes

CONVOLUTIONAL codes can be used in hybrid-automatic-repeat-request (hybrid-ARQ) protocols through the modification of standard forward-error-correcting (FEC) decoders. The basic idea is to identify a source of reliability information within the decoding algorithm and to use it to estimate the reliability of decoded data packets. If a decoded packet is deemed unreliable, a retransmission is requested; otherwise, the packet is accepted and passed along to the data sink. This approach has been successfully applied to sequential decoding algorithms by Drukarev and Costello [1] and to the Viterbi algorithm by Yamamoto and Itoh [2] and Wicker [3].

In this paper a method is demonstrated for modifying majority-logic decoders for use in hybrid-ARQ protocols. Majority-logic decoders do not provide as much coding gain as Viterbi and sequential decoders for a given convolutional code, but they do have the significant advantage of being implementable at much higher data rates. The paper begins with a description of the decoder modification. This is followed by the derivation of upper bounds for output bit error rate and packet retransmission probability. These bounds show that the hybrid-ARQ modification offers a substantial improvement in reliability performance over the FEC majority-logic decoder at the expense of a negligible reduction in throughput. These results are supported by simulation data.

II. The Modification of Majority-Logic Decoders for Use in Hybrid-ARQ Protocols

The majority-logic decoding of convolutional codes was first demonstrated by Massey in 1963 [4]. Using an (n, k, m) systematic convolutional code he showed that a significant amount of coding gain could be obtained by comparing select syndrome bits computed from a single constraint length ($L = n(m+1)$ bits) of the received data stream. The syndrome bits are used in a voting scheme (hence the term "majority-logic") to resolve the errors in a k-bit information block.

Let J be the number of check sums orthogonal on an information sequence error bit e. The FEC majority-logic decoding rule estimates the value of e in the following manner. Let η be the number of check sums with a value of 1. If $\eta > \lfloor J/2 \rfloor$ then e is assumed to have a value of one; otherwise, e is assumed to be zero [5]. The J check sums used in the decoding process provide a source of reliability information that can be exploited in a hybrid-ARQ protocol. The greater the extent of the majority among the check sums, the more likely that the estimate will be correct. This basic idea leads to the following modified majority-logic decoding rule.

Paper approved by the Editor for Coding Theory and Applications of the IEEE Communications Society. Manuscript received April 7, 1988; revised December 5, 1988.

The author is with the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332.

IEEE Log Number 8933669.

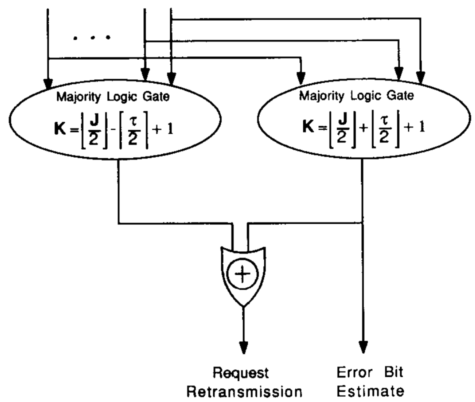

Fig. 1. Modified majority-logic estimation circuit for use in a hybrid-ARQ protocol.

A. Modified Majority-Logic Decoding Rule

Let η be the number of check sums that have a value of one. Let τ be an integer less than J. The estimate of the value of e is made in the following manner:

$$\begin{cases} \eta \geq \left\lfloor \frac{J}{2} \right\rfloor + \left\lfloor \frac{\tau}{2} \right\rfloor + 1: \quad e = 1 \\ \left\lfloor \frac{J}{2} \right\rfloor - \left\lceil \frac{\tau}{2} \right\rceil < \eta < \left\lfloor \frac{J}{2} \right\rfloor + \left\lfloor \frac{\tau}{2} \right\rfloor + 1: \\ \qquad \text{Request Retransmission} \\ \eta \leq \left\lfloor \frac{J}{2} \right\rfloor - \left\lceil \frac{\tau}{2} \right\rceil: \quad e = 0 \end{cases}.$$

The above rule clearly defines a type-I hybrid-ARQ protocol [6]. The most frequent error patterns are those that cause almost all of the J check sums to be either one or zero. When these patterns occur, decoding proceeds normally. The less frequent, higher weight error patterns cause the majority among the J check sums to be less pronounced. These are the patterns most likely to cause decoder errors, so these are the patterns that have been selected to trigger retransmission requests. Note that if τ is set to zero the decoding rule is identical to that used in the FEC decoder.

In the modified decoder the FEC majority-logic circuit is replaced by the circuit in Fig. 1. Whenever the value of η equals or exceeds $\lfloor J/2 \rfloor - \lceil \tau/2 \rceil + 1$ but falls short of $\lfloor J/2 \rfloor + \lfloor \tau/2 \rfloor + 1$, the exclusive-OR gate triggers a retransmission request. If η equals or exceeds $\lfloor J/2 \rfloor + \lfloor \tau/2 \rfloor + 1$ or is less than $\lfloor J/2 \rfloor - \lceil \tau/2 \rceil + 1$, both inputs to the exclusive-OR gate are identical and decoding is allowed to

0090-6778/90/0300-0263$01.00 © 1990 IEEE

proceed normally. The additional circuitry is simple in the extreme, preserving the primary motivation for using majority-logic decoders: a high speed implementation. The question remains as to whether the modification provides a substantial increase in reliability and at what cost.

III. Reliability and Throughput for the Modified Majority-Logic Decoder

In this section upper bounds are obtained for the probability of decoder error and retransmission for the hybrid-ARQ version of the majority-logic decoder. These bounds are developed into system level bit error rate expressions to demonstrate the improvement in performance available through the ARQ modification.

Consider an (n, k, m) convolutional code from which a set of J orthogonal parity check sums has been obtained. The J check sums are syndromes or sums of syndromes derived from an effective constraint length l bits of the received sequence. This code is used to transmit packets of length N over a binary symmetric channel with channel bit error probability p.

The FEC decoder generates an error value estimate of one whenever $\lfloor J/2 \rfloor + 1$ or more of the check sums have a value of one; otherwise, an estimate of zero is made. There are two ways in which this estimate can be incorrect. First suppose that the error bit "e" being estimated actually has a value of zero. If one assumes that the other $(l-1)$ error bits being checked are distributed in the worst possible manner, then an incorrect estimate will be made if $\lfloor J/2 \rfloor + 1$ or more of these other bits have a value of one. If e has a value of one, then an incorrect estimate will be made if at least $(J - \lfloor J/2 \rfloor) = \lceil J/2 \rceil$ of the other error bits are one. An upper bound can thus be obtained for the probability of a decoder error in the first effective constraint length of a received packet in the following manner:

$$P_{\mathrm{CL}}(E) \le P\left\{ \ge \left\lfloor \frac{J}{2} \right\rfloor + 1 \text{ errors among } l-1 \text{ bits} \right\} P\{e=0\} + P\left\{ \ge \left\lceil \frac{J}{2} \right\rceil \text{ errors among } l-1 \text{ bits} \right\} P\{e=1\} \quad (1)$$

$$= (1-p) \sum_{e=\lfloor \frac{J}{2} \rfloor + 1}^{l-1} \binom{l-1}{e} p^e (1-p)^{l-1-e} + p \sum_{e=\lceil \frac{J}{2} \rceil}^{l-1} \binom{l-1}{e} p^e (1-p)^{l-1-e}. \quad (2)$$

The hybrid-ARQ modification increases the number of errors that must occur in an effective constraint length of the received packet before a decoding error can occur. According to the modified majority-logic decoding rule, a retransmission will be requested if η of the check sums have a value of one where

$$\left\lfloor \frac{J}{2} \right\rfloor - \left\lceil \frac{\tau}{2} \right\rceil < \eta < \left\lfloor \frac{J}{2} \right\rfloor + \left\lfloor \frac{\tau}{2} \right\rfloor + 1.$$

The probability of decoder error expression is thus changed to

$$P_{\mathrm{CL}}(E) \le (1-p) \sum_{e=\lfloor \frac{J}{2} \rfloor + \lfloor \frac{\tau}{2} \rfloor + 1}^{l-1} \binom{l-1}{e} p^e (1-p)^{l-1-e} + p \sum_{e=\lceil \frac{J}{2} \rceil + \lceil \frac{\tau}{2} \rceil}^{l-1} \binom{l-1}{e} p^e (1-p)^{l-1-e}. \quad (3)$$

In feedback majority-logic decoders the estimated value of e is subtracted from the syndromes that it affects. If the estimate is correct, successive estimates of error bits in the information portion of the received packet will be equally reliable.[1] The decoded packet bit error probability provided by an FEC feedback decoder may thus be bounded above by (2).

P_e, the probability of a particular information bit being decoded incorrectly by the hybrid-ARQ decoder *on any given transmission*, is bounded above by (3). To obtain an accurate expression for $P(E)$, the probability of bit error within an accepted packet, allowance must be made for the fact that the decoder may have several opportunities to incorrectly decode a given packet. Let P_r be the probability that a particular transmission will cause the generation of a retransmission request. The probability of bit error in an accepted packet [6] is thus

$$P(E) = P_e(1 + P_r + P_r^2 + P_r^3 + \cdots) = \frac{P_e}{1 - P_r}. \quad (4)$$

The probability of packet retransmission can be bounded above in a manner similar to that used with decoder output bit error rate. Consider an N-bit received packet consisting of a message that has been encoded using an (n, k, m) convolutional code. A modified majority-logic decoder generates estimates for each error bit in the information segment of the packet using l-bit portions of the entire received packet. A retransmission is requested if, in making any one of these estimates, the number of ones among the J parity check sums falls within the region

$$\left\lfloor \frac{J}{2} \right\rfloor - \left\lceil \frac{\tau}{2} \right\rceil < \eta < \left\lfloor \frac{J}{2} \right\rfloor + \left\lfloor \frac{\tau}{2} \right\rfloor + 1.$$

Retransmissions are thus generated by patterns of $\lfloor J/2 \rfloor - \lceil \tau/2 \rceil + 1$ or more errors within an l-bit region if those errors are distributed in the "right" manner.

The probability of there being exactly e errors within a group of $(l-1)$ bits is

$$\Psi_e = \binom{l-1}{e} p^e (1-p)^{l-1-e}. \quad (5)$$

The probability of a particular $(l-1)$-bit segment of the received packet causing the generation of a retransmission request is thus obtained by summing Ψ_e over the appropriate values of e. In an N bit packet, $N(k/n)$ error estimates are made. Using the union bound the following upper bound on retransmission probability for the entire packet is obtained:

$$P_r \le N\left(\frac{k}{n}\right) \sum_{e=\lfloor \frac{J}{2} \rfloor - \lceil \frac{\tau}{2} \rceil + 1}^{l-1} \binom{l-1}{e} p^e (1-p)^{l-1-e}. \quad (6)$$

The throughput for the FEC system is equal to the rate of the code. In the hybrid-ARQ system a multiplicative factor must be introduced to account for the retransmissions. It will be assumed that a selective repeat protocol is in use, though it is a simple matter to alter the following to allow for stop-and-wait or go-back-N protocols. The throughput for an SR-ARQ system using an (n, k, m) convolutional

[1] In feedback decoding the decoder errors will propagate, causing further decoder errors to be more likely. There are several methods for ameliorating the effects of error propagation, but perhaps the simplest is to limit the selection of convolutional codes to those with automatic resynchronization properties [5], [7].

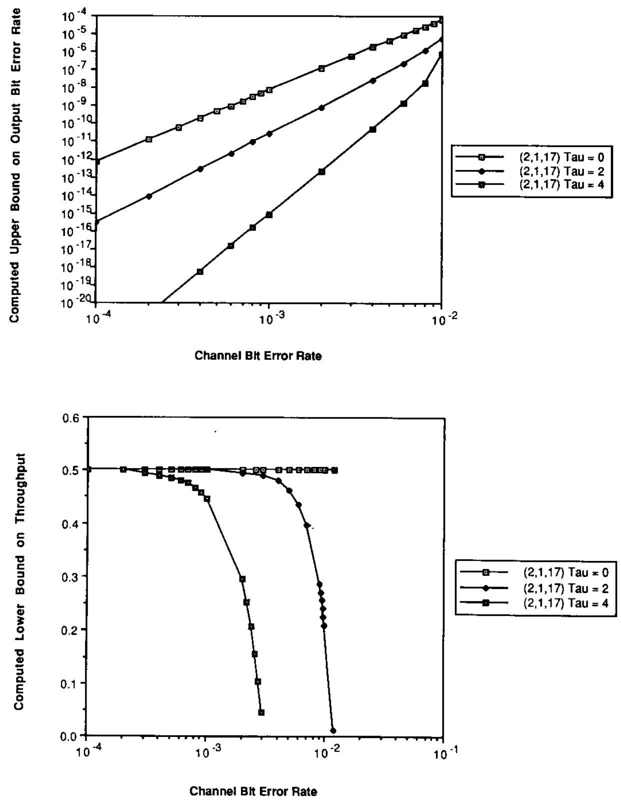

Fig. 2. Reliability and throughput performance for the modified majority-logic decoding of (2, 1, 17) convolutional codes. Packet length = 1000.

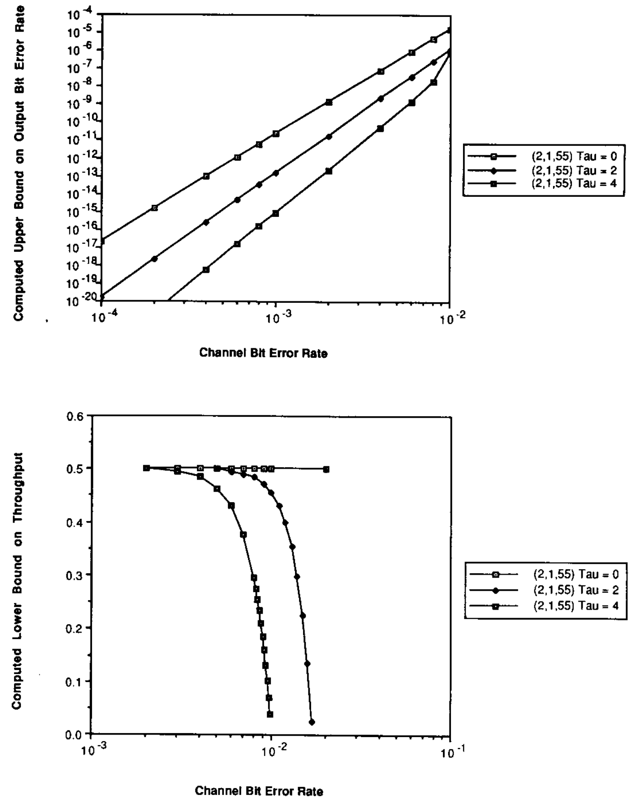

Fig. 3. Reliability and throughput performance for the modified majority-logic decoding of (2, 1, 55) convolutional codes. Packet length = 1000.

code with retransmission probability P_r [6] is

$$v_{SR} = \frac{\left(\frac{k}{n}\right)}{[(1-P_r)+2P_r(1-P_r)+3P_r^2(1-P_r)+\cdots]} = \left(\frac{k}{n}\right)(1-P_r). \tag{7}$$

When (6) is substituted for P_r in (7), a lower bound on throughput it obtained for the hybrid-ARQ system.

IV. Computational and Simulated Performance Data

Figs. 2 and 3 show the predicted performance of the modified majority-logic decoder using (2, 1, 17) and (2, 1, 55) convolutional codes and packets of length 1000. In both cases the reduction in output bit error rate is substantial. The throughput graphs show that the consequent reduction in throughput is negligible up to channel bit error rates of approximately 10^{-2}.

Simulations were performed to determine the tightness of the lower bound on throughput. A linear congruential random number generator was used to emulate a binary symmetric channel with arbitrary probabilities of bit error [8]. The results of several of the simulation runs are shown in the graph in Fig. 4. The computed lower bounds have been included for reference. The data indicates that the bounds are fairly tight and provide an accurate representation of the degradation in throughput exhibited by the system as channel bit error rate exceeds 10^{-2}.

The modification of majority-logic decoders for use in convolutionally encoded hybrid-ARQ systems clearly provides a substantial

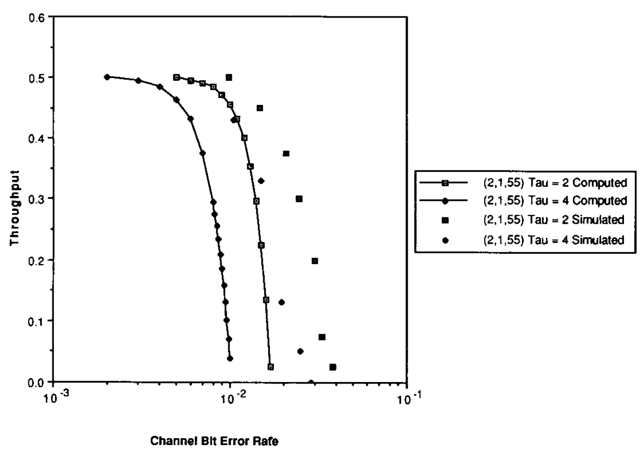

Fig. 4. A comparison of simulation throughput performance to the computed lower bound for (2, 1, 55) convolutional codes. Packet length = 1000.

degree of coding gain at minimal expense. Since the modified decoder retains the implementational simplicity of the forward-error-correcting version, it provides a viable alternative for error control in high data rate applications.

References

[1] A. Drukarev and D. J. Costello, Jr., "Hybrid ARQ error control using sequential decoding," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 521-535, July 1983.

[2] H. Yamamoto and K. Itoh, "Viterbi decoding algorithm for convolutional codes with repeat request," *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 540–547, Sept. 1980.

[3] S. B. Wicker, "An adaptive type-I hybrid-ARQ technique using the Viterbi decoding algorithm," *Proc. 1988 IEEE Military Commun. Conf.*, San Diego, CA.

[4] J. L. Massey, *Threshold Decoding*. Cambridge, MA: M.I.T. Press, 1963.

[5] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. NJ: Prentice-Hall, 1983.

[6] S. Lin, D. J. Costello, Jr., and M. J. Miller, "Automatic-repeat-request error control schemes," *IEEE Commun. Mag.*, vol. 22, pp. 5–17, Dec. 1984.

[7] J. P. Robinson and A. J. Bernstein, "A class of binary recurrent codes with limited error propagation," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 106–113, Jan. 1967.

[8] D. E. Knuth, *The Art of Computer Programming*. Reading, MA: Addison-Wesley, second ed., vol. 2, 1981.