

# TECHNICAL RESEARCH REPORT

An Adaptive Hybrid FEC/ARQ Protocol Using Turbo Codes  
for Multi-Media Transmission of ATM over Wireless  
Networks

*by Wai-Chung Chan, Evaggelos Geraniotis*

**CSHCN TR 97-37**  
**(ISR TR 97-95)**



*The Center for Satellite and Hybrid Communication Networks is a NASA-sponsored Commercial Space Center also supported by the Department of Defense (DOD), industry, the State of Maryland, the University of Maryland and the Institute for Systems Research. This document is a technical report in the CSHCN series originating at the University of Maryland.*

**Web site <http://www.isr.umd.edu/CSHCN/>**

# An Adaptive Hybrid FEC/ARQ Protocol Using Turbo Codes for Multi-Media Transmission of ATM over Wireless Networks \*

Wai-Chung Chan and Evaggelos Geraniotis  
Department of Electrical Engineering  
and Institute for Systems Research  
University of Maryland  
College Park, MD 20742

## ABSTRACT

*This paper describes a type-II adaptive hybrid FEC/ARQ protocol using turbo codes for both voice and data services. Recently, a powerful family of error correcting codes using parallel concatenated convolutional codes is introduced. Near-Shannon-limit is achievable by such codes. For example, a bit error rate of  $10^{-5}$  can be achieved at  $E_b/N_o = 0.7$  dB using a rate 1/2 code or at about 2.5 dB if fewer iterations are used in decoding. Given the promising performance of the codes, it is expected that the performance of an adaptive FEC/ARQ protocol employing turbo codes will be better than protocols using other error correcting codes.*

*Rate compatible turbo codes are used in the protocol. Extra parity bits corresponding to a lower rate turbo code will be transmitted if the previous transmission leads to a decoding error. For real-time services, the number of retransmissions is limited to one so as to limit the delay. For data services, the number of retransmissions is unlimited to provide a reliable communication link. Results in terms of throughput are obtained for the adaptive FEC/ARQ protocol using compatible rate 1/2 and rate 1/4 turbo codes.*

## INTRODUCTION

Wireless radio networks differ from wired networks in several aspects. The amount of information that can be transmitted through a radio network is limited by the capacity of the radio spectrum while wired networks can be connected by high speed fiber optics. Error-prone time-varying channel conditions in radio networks also impose another limit. Powerful error correcting codes have to be used to provide the bit error rate required by the applications. The error correcting schemes used should be able to adapt to the instantaneous channel conditions, such as the signal-to-interference ratio, to conserve bandwidth.

---

\*This research is continuing through collaborative participation in the Advanced Telecommunications/Information Distribution Research Program (ATIRP) Consortium sponsored by the U.S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-0002.

The presence of multi-media applications makes the situation more complicated. The error correcting schemes to be used for each application should be based on the required quality of service of the application as well as its delay constraint. Furthermore, it is also important to share the available capacity in the radio networks with the multi-media applications in a fair and efficient manner, which is dealt with by the multiple access protocol. The error correcting scheme used has to be compatible with such access protocol which is expected to operate in a packet switching mode using a combination of reservation and contention to take full advantage of the burstiness of multi-media traffic.

Multi-media applications are generally classified into real-time services and delay-insensitive traffic. The information carried in the first class of traffic is time-sensitive. Examples include packetized voice and video. The maximum allowable delay is about one frame. Bit error rate requirement is less stringent than the delay-insensitive traffic. However, a bit error probability of about  $10^{-3}$  or better is still expected for this type of applications. Example of delay-insensitive traffics include text files, maps and emails. The information has to arrive at the destination error-free. This is usually achieved by combination of powerful error correcting codes and retransmission schemes. The delay requirement is less stringent than the real-time applications.

While it is simple enough to use a low rate forward error correcting code to provide the desired bit error rate (e.g.  $10^{-5}$ ), the throughput of the system will be very poor when the channel is good because of the overhead used. In situations where the condition of the channel can vary, an adaptive scheme is desired. Recently, a powerful family of error correcting codes using parallel concatenated convolutional codes (PCCC) is introduced. Near-Shannon-limit is achievable by such codes. For example, a bit error rate of  $10^{-5}$  can be achieved at  $E_b/N_o = 0.7$  dB [1] using a rate 1/2 code or at about 2.5 dB [2] if fewer iterations are used in decoding. Given the promising performance of the codes, it is expected that the performance of the proposed FEC/ARQ protocol employing turbo codes will be better than protocols using other error correcting codes. The multiple access protocol behind is assumed to be any one of the

TDMA protocols proposed in the literature such as PRMA [10] or DQRUMA/NCPRMA [11]. It is assumed that the outcomes of the transmission, in the form of positive acknowledgments ACK or negative acknowledgments NACK, are received by the transmitter before the beginning of the next frame. The FEC/ARQ protocol then adapts to different channel conditions depending on whether an ACK or NACK is received.

Rate compatible turbo codes can be formed easily. In the proposed FEC/ARQ protocol, when the channel is in the good state, only the high rate codeword is sent. In the event of decoding error (which is detected by attaching CRC bits to the information packet), extra parity bits corresponding to the lower rate codeword will be sent. At the receiver, the packet received earlier is combined with the extra parity bits and decoded. If the codeword can be decoded successfully, the next user packet will be sent.

If the channel is expected to be in the bad state, the above method of sending the codewords in two “installments” will introduce unnecessary delay. Hence, the entire low rate codeword is sent to the receiver instead. The user data rate has to be reduced or more bandwidth has to be allocated to the user to send the whole codeword at the same time. For voice services, we assume that the user data rate is reduced. For data services, since the user has to wait for the base station’s polling signal before it can transmit, we assume that the polling signal also informs the user how much bandwidth is allocated to it and thus, the bandwidth allocated to a data user can be variable.

Since the performance of turbo codes is better than other convolutional codes of the same coding rate, it is expected that the throughput of the proposed protocol is better. When the SNR is expected to be good (around 4 or 5 dB), a rate 1/2 turbo code is sufficient. However, if the protocol is expected to operate in extremely low SNR (e.g. around 2 or 3 dB), a rate 1/4 turbo code is required. Results using these two rate compatible turbo codes will be shown in this paper. This adaptive FEC/ARQ scheme suits very well in slow fading environment.

The operation of turbo codes is briefly described in the next section, followed by the description of the proposed adaptive FEC/ARQ protocols and the performance analysis. Results and conclusions will then be given.

## TURBO CODES

The turbo code used in the proposed FEC/ARQ protocol consists of a parallel concatenation of two systematic rate 1/2 convolutional codes [1]. The overall rate is 1/3. (Figure 1.) For an input stream, three output streams are formed: the input stream  $x$  itself (thus the turbo code is systematic), a parity stream produced by the first convolutional code  $y1$  and another parity stream  $y2$  produced by the second convolutional code with an interleaved ver-

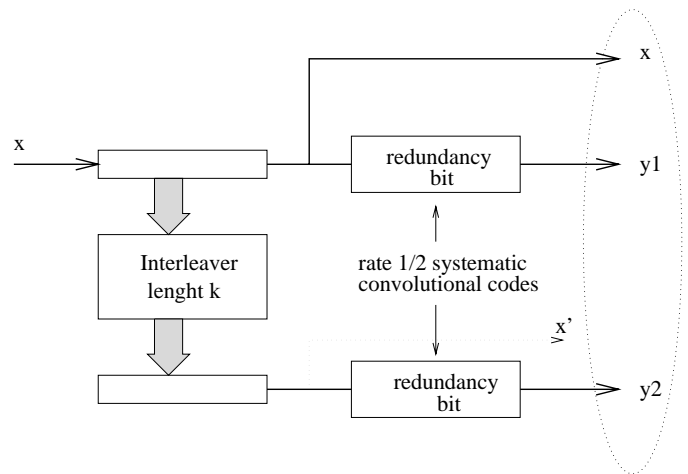


Figure 1: Turbo Encoder.

sion of the input stream. The two parity streams can be punctured to form a rate 1/2 code.

The decoder is made up of a number of modules, each consisting of two decoders in serial concatenation with an interleaver in between. Using the codec in [2], a bit error rate of  $10^{-5}$  can be achieved at about 2.5 dB with an overall global decoding latency of 2318 bits. When the size of the information stream is smaller, the performance will deteriorate.

Now, suppose another interleaver and branch  $y3$  are added. A compatible rate 1/4 codeword can be formed from streams  $x$ ,  $y1$ ,  $y2$ , and  $y3$ .

The bit error probability and the frame error probability for the rate 1/2 and rate 1/4 turbo codes are given in Figures 2 and 3. The size of the information packet for the rate 1/2 code is 400 bits (which is approximately the size of an ATM cell) and that for the rate 1/4 code is 200 bits.

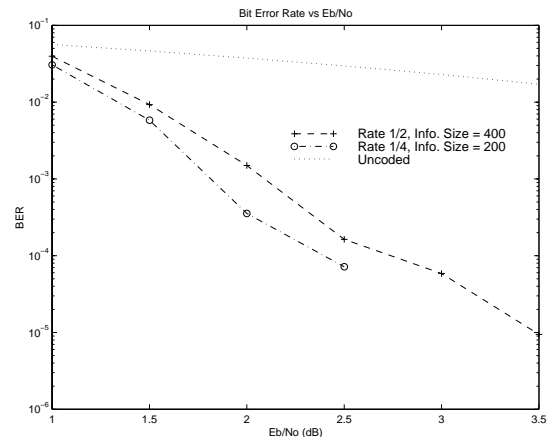


Figure 2: Bit Error Probability.

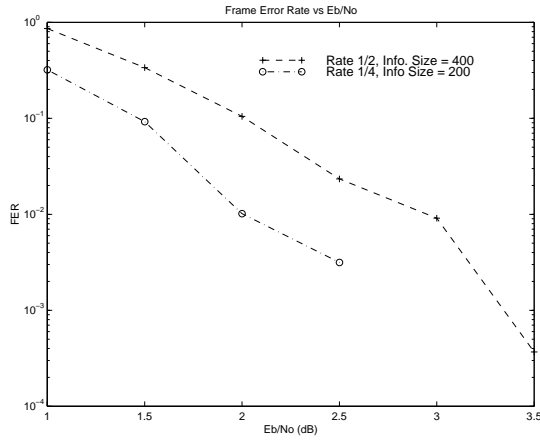


Figure 3: Frame Error Probability.

## ADAPTIVE HYBRID FEC/ARQ PROTOCOL FOR VOICE SERVICES

In this section, we describe the proposed type-II adaptive hybrid FEC/ARQ protocol for real-time/voice services. The proposed protocol consists of three states. The system will be in state 1 if the channel condition is good, while it will be in state 3 if the channel is bad. (Figure 4)

The information packet from the user is encoded using compatible rate 1/2 and rate 1/4 turbo codes. When the protocol is in state 1, only the rate 1/2 codeword will be transmitted. The (extra) parity packet corresponding to the rate 1/4 codeword will be stored at the user's buffer. If there is no decoding error, the protocol remains in state 1 and the next information packet is processed in the same manner. (Decoding error is detected using the CRC attached to the information packet.) In the event of decoding error, state 2 will be entered where the parity packet stored in the buffer will be transmitted. The parity packet and the rate 1/2 codeword received earlier are combined at the receiver and decoded (at rate 1/4). If there is no decoding error, the protocol will go back to state 1 and the next information packet is sent. Otherwise, state 3 will be entered.

Since the proposed protocol is designed for real-time services, upon entering state 3 from state 2, the previous codeword will be discarded. A new information packet is encoded at rate 1/4 and the entire codeword is transmitted. In order to keep the size of the packet constant, the size of the information packet has to be halved in state 3. In other words, the source rate has to be reduced. (An alternative is to assign two times the resource to the user if the service supports a constant bit rate.)

In state 3, the encoded packet will be retransmitted once in the event of decoding error at the receiver end. If the

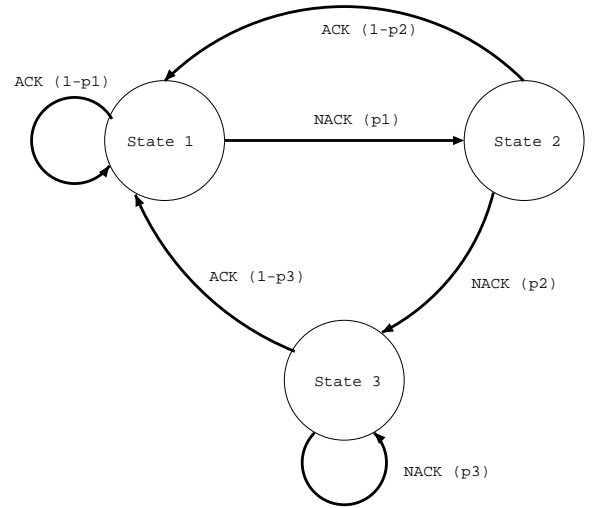


Figure 4: State Diagram for the FEC/ARQ Protocol.

retransmission also results in error, the old packet is discarded and a new encoded packet using a new information packet is transmitted. After the receiver has decoded a number of packets correctly, the protocol will then switch back to state 1 and start sending the rate 1/2 codeword in the first transmission attempt. In the results to be followed, we assume that the protocol will go back to state 1 after one packet has been decoded successfully so as to make the analysis simpler.

### Throughput Analysis (Voice)

Let  $p_i$  denote the probability that a decoding error occurs at the receiver when the protocol is in state  $i$ . (Figure 4.) Let the throughput  $\eta_i$  be defined as the ratio of the number of information packets received correctly at the receiver to the total number of packets transmitted when the protocol begins in state  $i$ . Hence,  $\eta_1$  and  $\eta_3$  are given by

$$\eta_1 = \frac{1(1-p_1) + p_1(1-p_2)}{2(1-p_1) + 2p_1} = \frac{1-p_1p_2}{2(1+p_1)}$$

and

$$\eta_3 = \frac{1(1-p_3) + p_3(1-p_3)}{4(1-p_3) + 2p_3} = \frac{1}{4}(1-p_3)$$

The average throughput of the protocol is then given by  $\bar{\eta} = \pi_3 \times \eta_3 + (1 - \pi_3) \times \eta_1$  where  $\pi_i$  denotes the probability that the protocol begins in state  $i$ . Assuming that steady state can be reached before the channel condition is changed, the values for  $\pi_i$  can be shown to be

$$\pi_1 = \frac{1-p_3}{p_1p_2 + (1+p_1)(1-p_3)}$$

$$\pi_2 = \frac{p_1(1-p_3)}{p_1p_2 + (1+p_1)(1-p_3)}$$

$$\pi_3 = \frac{p_1 p_2}{p_1 p_2 + (1 + p_1)(1 - p_3)}$$

### Packet Dropping Probability (Voice)

Since the number of retransmissions is limited to one in the proposed voice protocol, it is possible that some user packets will never be decoded correctly at the receiver. Such packets are considered dropped. The packet dropping probabilities when the protocol begins in state 1 and state 3 are  $p_1 p_2$  and  $p_3^2$ , respectively. Thus, the average packet dropping probability is  $\pi_3 p_3^2 + (1 - \pi_3) p_1 p_2$ .

## ADAPTIVE HYBRID FEC/ARQ PROTOCOL FOR DATA SERVICES

In the proposed data protocol, the number of retransmissions is unlimited. As a result, no packets will be dropped. However, delay will be introduced in the system.

### Throughput Analysis (Data)

When the protocol begins in state 1, the expected number of packets transmitted in order to successfully decode one information packet is

$$\begin{aligned} \bar{N}_1 &= 1(1 - p_1) + 2p_1(1 - p_2) \\ &\quad + p_1 p_2 (1 - p_3)^2 \sum_{k=2}^{\infty} (2 + k) p_3^{k-2} \\ &= 1 + p_1 + 2p_1 p_2 - 3p_1 p_2 p_3 \end{aligned}$$

When the protocol begins in state 3, the expected number of packets transmitted in order to successfully decode one information packet is  $\bar{N}_3 = 1/(1 - p_3)$ .

Therefore, the average throughput is

$$\bar{\eta} = \frac{\pi_3}{4\bar{N}_3} + \frac{1 - \pi_3}{2\bar{N}_1}$$

### Delay of the Data Protocol

The expected delay (in terms of the number of TDMA frames) of the proposed data protocol when the protocol begins in state 1 and state 3 are, respectively,

$$\begin{aligned} \bar{D}_1 &= 0(1 - p_1) + 1p_1(1 - p_2) \\ &\quad + p_1 p_2 (1 - p_3)^2 \sum_{k=2}^{\infty} (1 + k) p_3^{k-2} \\ &= p_1 + 2p_1 p_2 - 2p_1 p_2 p_3 \\ \bar{D}_3 &= p_3 / (1 - p_3) \end{aligned}$$

Therefore, the average delay is  $\bar{D} = \pi_3 \times 2\bar{D}_3 + (1 - \pi_3) \times \bar{D}_1$ .

## RESULTS

The values for  $p_1$  and  $p_3$  are readily available through simulations as a function of SNR. They are the frame error probabilities for the rate 1/2 and rate 1/4 systems, respectively. The exact value for  $p_2$ , the packet error probability of the rate 1/4 system given that the compatible rate 1/2 codeword leads to decoding error, is not available. In the results in Figures 5 and 7, we replace  $p_2$  by the corresponding  $p_3$  at the same SNR. Hence, the results give an upper bound for the throughput.

Since the number of retransmissions is limited to one in the voice protocol, packets can be dropped. The packet dropping probabilities are plotted in Figure 6 for the voice case. The delay for the data case is plotted in Figure 8.

## CONCLUSIONS

Adaptive hybrid FEC/ARQ protocols using rate compatible turbo codes are described in this paper for voice and data services. The rate 1/4 turbo code employed gives a bit error rate of  $10^{-4}$  at a SNR of about 2.5 dB. A more reliable transmission is made possible by using retransmissions if necessary. The number of retransmissions is restricted to one to limit the delay introduced by the protocol in the case of voice services. By using different FEC schemes under different channel conditions, good throughput can be achieved in a slow fading channel<sup>1</sup>.

### References

- [1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo-codes," in *Proc. ICC'93*, pp. 1064-1070.
- [2] *CAS 5093 Turbo-Code Codec*, rev. 4.1, data sheet, Chateaubourg, France: Comatlas, May 1995.
- [3] S. Benedetto, G. Montorsi, D. Divsalar, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design, and iterative decoding," *TDA Progress Report 42-126*, August 1996, Jet Propulsion Laboratory.
- [4] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice Hall, 1995.
- [5] L. R. Lugand, D. J. Costello, and R. Deng, "Parity retransmission hybrid ARQ using rate 1/2 convolutional codes on a nonstationary channel," *IEEE Trans. Commun.*, vol. COM-37, no. 7, pp. 755-765, July, 1989.

<sup>1</sup>The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government.

- [6] Z. Sayeed, and V. Weerackody, "Adaptive hybrid ARQ schemes for slow fading channels," in *Proc. CISS'96*.
- [7] Q. Zhang, and S. A. Kassam, "Adaptive hybrid ARQ/FEC protocols for low-bit-rate video over wireless channels."
- [8] Y.-M. Wang, and S. Lin, "A modified selective-repeat type-II hybrid ARQ system and its performance analysis," *IEEE Trans. Commun.*, vol. COM-31, no. 5, pp. 593-608, May, 1983.
- [9] W. C. Chan, E. Geraniotis, and V. D. Nguyen, "An Adaptive Hybrid FEC/ARQ Protocol Using Turbo Codes," in *Proc. ICUPC'97*.
- [10] D. J. Goodman, R. A. Valenzuela, K. T. Gayliard, and B. Ramamurthi, "Packet reservation multiple access for local wireless communications," *IEEE Trans. Commun.*, vol. COM-37, no. 8, pp. 885-890, Aug. 1989.
- [11] W. C. Chan, C. Derrien, and E. Geraniotis, "A medium access protocol for interconnecting ATM and wireless networks," in *Proc. ICC'97*, pp. 1448-1453.

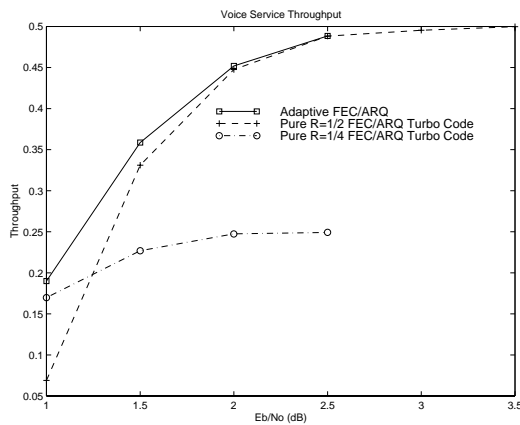


Figure 5: Throughput of the Voice Adaptive FEC/ARQ Protocol

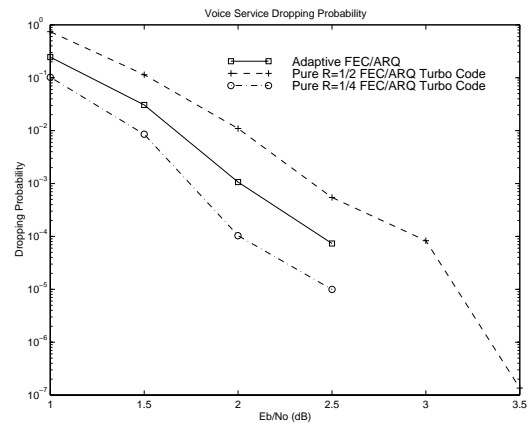


Figure 6: Packet Dropping Probability for the Voice Adaptive FEC/ARQ Protocol

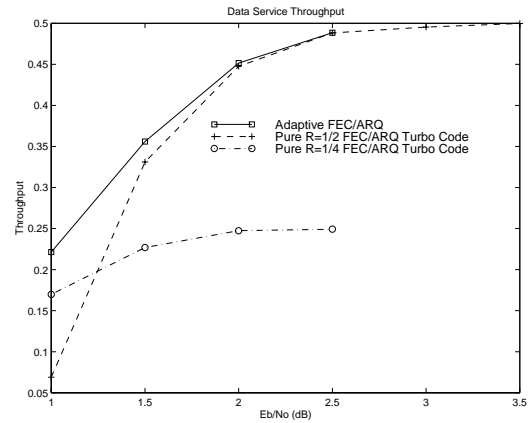


Figure 7: Throughput of the Data Adaptive FEC/ARQ Protocol

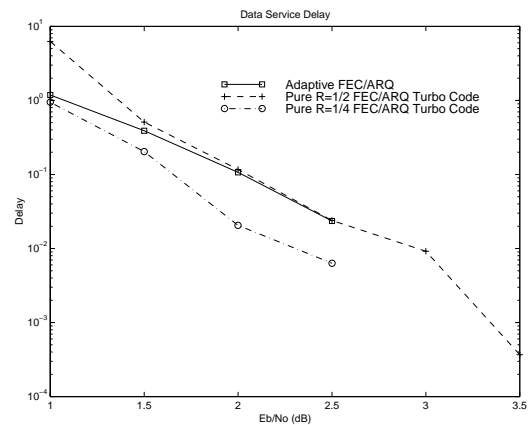


Figure 8: Expected Delay for the Data Adaptive FEC/ARQ Protocol