Adaptive Forward Error Correction Scheme for Real-Time Communication in Satellite IP Networks

Sungrae Cho
School of Computer Science and Engineering, Chung-Ang University
Seoul, Korea
[e-mail: srcho@cau.ac.kr]
*Corresponding author: Sungrae Cho

Received June 23, 2010; revised August 8, 2010; accepted September 20, 2010; published December 23, 2010

Abstract

In this paper, a new forward error correction (FEC) protocol is proposed for point-to-multipoint satellite links. Link-layer error control protocols in point-to-multipoint satellite links impose several problems such as unreliability and receiver-heterogeneity. To resolve the problem of heterogeneous error rates at different receivers, the proposed scheme exploits multiple multicast channels to which each receiver tunes. The more channels a receiver tunes to, the more powerful error correcting capability it achieves. Based on its own channel condition, each receiver tunes to as many channels as it needs, which prevents from receiving unwanted parities. Furthermore, each receiver saves the decoding time, processing overhead, and processing energy. Performance evaluation shows that the proposed scheme guarantees the target PER while saving energy. The proposed technique is highly adaptive to the channel variation with respect to the throughput efficiency, and provides scalable PER and throughput efficiency.

Keywords: Satellite, Adaptive FEC, error control protocol, and reliable multicast

This research was supported by the Chung-Ang University Research Scholarship Grants in 2008.

DOI: 10.3837/tiis.2010.12.008

1. Introduction

In order to deploy and operate global scale multicast services, there are a number of issues that have hindered terrestrial deployment including:

- Difficulties in providing the multicast service across a global geographical area including *ocean and air*,
- Limited geographical coverage since terrestrial deployment is economically infeasible in rough terrains or the areas with insufficient user population,
- Difficulties in upgrading the existing deployed network routers, and
- Lack of multicast protocols for large multicast networks.

For global multicast services, satellite networks would have a greater role than terrestrial networks. Satellite networks are known to be the most efficient communication media providing multicast services due to their large coverage area, broadcasting nature, abundant bandwidth particularly at higher frequency, increased network reliability owing to minimum router hops, and rapid network setup [1][2]. The satellite solution also minimizes the number of multicast-capable routers required in the core network, which will simplify deployment, operation, and maintenance.

However, the major problem using satellite networks is their unreliability since satellite links are characterized by higher error rate and burstier error pattern than terrestrial wireline networks. Sources of channel impairment include fading, shadowing, and atmospheric conditions. Receiver heterogeneity is another significant problem in the satellite multicast services. Different receivers are likely to experience different channel conditions. Suppose a receiver suffers a relatively high error rate. Then, it is questionable to provide the maximum error correcting capability to all receivers in order to salvage the suffering receiver because the throughputs of all other receivers will be limited to the throughput of the suffering receiver – is this a fair policy? Also, different receivers may demand different quality of service (QoS) guarantees, e.g., some of the receivers may require low packet error rate (PER) while the other receivers request marginal PER. Hence, it is a significantly challenging issue how to deal with heterogeneous receivers and to satisfy their QoS fairly.

In this paper, we propose a new forward error correction (FEC) protocol for point-to-multipoint satellite links with the following design objectives:

- **Reliability for Heterogeneous Receivers**: The FEC must provide different channel code rates for different receivers according to their channel conditions in order to achieve receiver fairness. The FEC needs to guarantee its upper layer's PER requirement for each receiver, even when each receiver requires different target PER from others.
- **Throughput Efficiency**: The FEC must maximize receiver's throughput efficiency subject to the PER constraint where throughput efficiency here is defined as the ratio of the number of useful information to the total data received including parities.
- **Low Latency**: It is noted that the retransmission in satellite networks is not desirable for multimedia traffic with time constraint due to satellite's long round-trip time (RTT).

There exist few link-layer forward error correction schemes for reliable multicast services in the literature. Weerackody *et al*. [3] proposed a multicast link-layer static FEC for satellite networks. They use a diversity combining (time diversity) technique from at least two satellites, where multiple identical copies of a packet provide the time diversity, and are more reliable than those of any of individual copies. This scheme, however, introduces a large delay, and it is not bandwidth-efficient. Nikaein *et al*. [4] proposed a multicast adaptive forward error correction (MA-FEC) scheme which has feedback implosion and receiver heterogeneity problems. Noguchi *et al*. [5] proposed a link-layer FEC applied only to the source link (from a source to an immediate router) in the terrestrial wireline multicast networks. Since source link contributes about 5% of total loss in the multicast networks and this source loss is shared by all receivers, the impact of the source loss becomes even worse. Hence, they employ a link-layer FEC scheme to support the reliable multicast services in the terrestrial wire-line networks. Tommasi *et al*. [6] proposed a satellite multicast distribution protocol (SMDP) in which the feedback messages are transmitted through terrestrial links. However, they could not address the feedback implosion problem caused from the feedback messages in their scheme. Si *et al*. [7] proposed a MAC protocol called RMAC that supports reliable multicast for wireless ad hoc networks. They used a tone-based acknowledgement to prevent data collision, which again introduces feedback implosion and may not work in satellite environment due to its long round-trip time. Koutsonikolas *et al*. [8] proposed a reliable multicast protocol for wireless mesh networks. Their scheme, however, does not work with satellite environment because it assumed multi-hop tree-based path while our scheme focuses on one-hop satellite link.

More recently, fountain [9] and its descendant Raptor [10] codes are widely accepted for multicast delivery in multimedia broadcast multicast services in universal mobile telecommunication system (MBMS/UMTS). Smokhina *et al*. [11] applied the Raptor code in FEC for video multicast over IEEE 802.11 WLAN. In their scheme, both Raptor code rate and physical transmission rate are dynamically adapted according to the channel condition. However, their scheme collects SNR feedbacks from the multicast receivers, which may work well in WLAN environment but will adversely behave in satellite environment due to even much more receivers (feedback implosion problem).

Our proposed FEC technique performs on the satellite backbone link which causes dominant losses in the satellite multicast services. To solve the heterogeneous error rates at different receivers, the proposed scheme uses multiple multicast channels to which each receiver tunes. Based on its channel condition, each receiver tunes to as many channels as it needs. Therefore, each receiver avoids unwanted parities, and saves the decoding time, buffer space, processing overhead and even energy consumption. The scheme maximizes the throughput efficiency subject to target PER constraint. Since the receivers adaptively determine their coding rate (or the number of channels they tune to), the proposed FEC can be categorized as an adaptive FEC (AFEC) protocol. Furthermore, the proposed AFEC does not rely on feedback information from receivers, and thus does not suffer from feedback implosion problem.

The remainder of this paper is organized as follows: the packetization and protocol description for the proposed protocol are described in Section 2. Section 3 presents performance results, and we conclude in Section 4.

2. Protocol Description

2.1 Packetization

To illustrate the heterogeneity problem, consider a single multicast channel from a satellite to *R* direct receivers. Suppose that data packets are transmitted on the single multicast channel. If a sufficiently large number of parities are added to the data packet in order to cope with the worst channel condition, receivers in good channel condition have to receive a large number of unwanted parities. This imposes unnecessary receiver processing overhead, receiving power and decoding time, especially as the number of receivers increases. If a marginal number of parities are added to the data packet, receivers in bad channel condition will suffer from severe PER degradation. Therefore, with a single multicast channel, we are not able to satisfy each individual receiver's need.

In order to avoid the above heterogeneity problem, our proposed scheme employs multiple multicast channels to scope parities to only receivers that need them. For this purpose, the scheme has the following packetization scheme:

- A data packet consists of a header and *L* encoded blocks EB_i's ($i = 0,..., L-1$), as shown in **Fig. 1**. These encoded blocks (EBs) are generated from a channel encoder module at the satellite or at the ground transmitter station. The size of each block EB_i and the number of blocks *L* are fixed value and determined by the channel encoder module. Consequently, the size of the packet is fixed. EB_0 is an encoded block with the highest code rate, thus, EB_0 has the lowest error correcting capability, while EB_i ($0 < i \le L-1$) represents the incremental redundancy (parities). Let us define block concatenation as a process to create a code from several EBs. Block concatenation of EB_0 with the incremental redundancies (EB_i's) results in increased error correcting capability. In other words, a code concatenated from EB_0 to EB_i has a better error correcting capability than a code concatenated from EB_0 to EB_{i-1} for $0 < i < L$, and can be formulated as

$$R_c\left(\bigoplus_{j=0}^{i} \mathrm{EB}_j\right) < R_c\left(\bigoplus_{j=0}^{i-1} \mathrm{EB}_j\right) \tag{1}$$

  where $R_c(x)$ represents the code rate of a code x and $\oplus_{j=0}^{i} \mathrm{EB}_j$ is the block concatenation operation from EB_0 to EB_i. In a Reed-Solomon (RS) code, block concatenation means simply block alignment. An example of block concatenation in a concatenated FEC code is given in **Appendix A**.
- The sender transmits the header and EB_0 on the primary sub-channel (PSC) and EB_i on the *i*th secondary sub-channel (SSC_i) for $1 \le i \le L-1$ as shown in **Fig. 1**. Because of (1), the more channels a receiver tunes to, the more powerful error correcting capability it achieves. Note that, in practice, the sub-channels can be realized by frequency bands, time slots, or codes depending on different types of underlying medium access control (MAC) protocols.

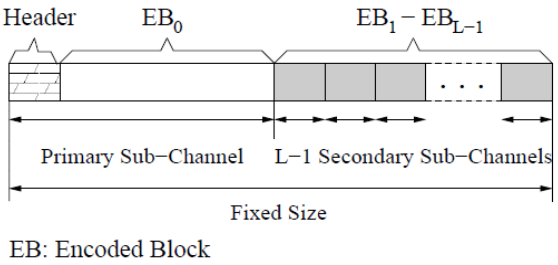


Fig. 1. Data packetization in the proposed protocol (downlink from a sender to *R* receivers)

2.2 The Number of Sub-Channels Tuned by Receiver

Each receiver monitors its channel conditions from physical layer's channel state information (CSI), e.g., bit error rate (BER) [1][12]. Based on its BER, receiver i selects a code $\oplus_{j=0}^{i} EB_j$ where the number of used (or tuned) EBs l_i at receiver i is obtained by

$$l_i = \arg\max_{\substack{k \in A_i \\ k \le L-1}} R_c\left(\bigoplus_{j=0}^{k} \mathrm{EB}_j\right) \tag{2}$$

where A_i is a set of indices k's that satisfy the following condition:

$$P_p\left(\bigoplus_{j=0}^{k} \mathrm{EB}_j \mid P_{b,i}\right) < \tilde{P}_{p,i} \tag{3}$$

where $P_{b,i}$ is the receiver i's bit error rate (BER), $P_p(x \mid y)$ is the PER of code x for given BER y, and $\tilde{P}_{p,i}$ is the target PER of receiver i ($\tilde{A}$ hereafter denotes the target parameter of the random variable A). If $A_i = \phi$, the receiver selects a code $\oplus_{j=0}^{L-1} \mathrm{EB}_j$. *(2)* and *(3) imply that the receiver selects a code with the maximum code rate which satisfies the target PER for its given BER.* This minimizes the number of EBs subject to the PER constraint.

In **Fig. 2**, we illustrate an example of code rate selection where threshold b_k is the BER value satisfies $P_p\left(\oplus_{j=0}^{k} \mathrm{EB}_j \mid b_k\right) = \tilde{P}_{p,i}$. If receiver i's current BER, $P_{b,i}$, is less than threshold b_0 in the figure, all codes satisfy the condition in (3), and the receiver selects EB_0 since code EB_0 has the maximum code rate. If $P_{b,i}$ is in the range of $[b_0, b_1)$, the code $\mathrm{EB}_0 \oplus \mathrm{EB}_1$ will be used with the same reason. Furthermore, if $P_{b,i}$ is within $[b_1, b_2)$, the code $\mathrm{EB}_0 \oplus \mathrm{EB}_1 \oplus \mathrm{EB}_2$ will be selected. If $P_{b,i}$ is in the range of $[b_2, 1]$, $\mathrm{A}_i = \phi$. Therefore, the code $\mathrm{EB}_0 \oplus \mathrm{EB}_1 \oplus \mathrm{EB}_2$ will be used.

Each receiver must tune to the primary sub-channel in order to receive the header and EB_0. However, each receiver tunes to some of the secondary sub-channels (SSCs) according to the selected code. Note that the number of tuned sub-channels is identical to the number of used EBs. l_i also can imply the number of tuned sub-channels at receiver i. Then, l_i will be different among all different receivers.

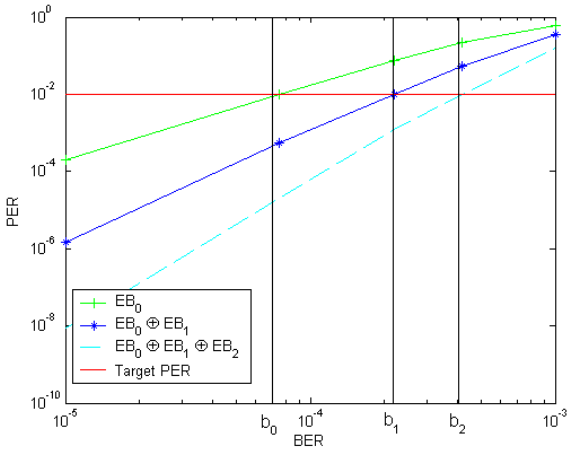


Fig. 2. Example of code rate selection (L=3 and $\tilde{P}_{p,i} = 10^{-2}$)

2.3 AFEC Operation

At the sender (for instance, satellite in on-board satellite systems or ground station in bent-pipe satellite systems), when the proposed AFEC receives a data from upper layer, it creates a packet using the packetization mechanism in Section 2.1, and transmits the packet.

Whenever data packet arrives, the receiver extracts L from the header, and monitors its BER from the physical layer's channel state information (CSI). It then selects its code rate based on L, BER, and target PER as in (2) and (3), and decodes the received packet. If the decoding is successful, each receiver delivers data to the upper layer; otherwise, it ignores the received packet.

2.4 System Requirement and Implementation Issue

Suppose we deploy the same type of receivers using the proposed AFEC. Receivers that have been in good channel conditions, e.g., receivers with rather stationary location and with line of sight (LOS) under-utilize their resources including buffer space and receiving power. Therefore, in this case, it would be preferable to provide receivers with different requirements (e.g., providing limited resources). In other words, in the proposed scheme, receiver i is designed to have the maximum number of tunable channels denoted by C_i where $1 \le C_i \le L$. By introducing C_i, receiver i will be manufactured with limited buffer space depending on C_i, thus reduce its cost of production, power consumption, processing overhead, etc. This will be also a benefit to the satellite operator since cost reduction will attract more users.

Based on C_i, the maximum allowable bit error rate for receiver i (denoted by $\hat{P}_{b,i}$) can be computed such that it satisfies

$$P_p\left(\bigoplus_{j=0}^{C_i} \mathrm{EB}_j \mid \hat{P}_{b,i}\right) < \tilde{P}_{p,i}, \tag{4}$$

or conversely, C_i can be derived from given system requirement of $\hat{P}_{b,i}$. Note that because of C_i, (2) becomes

$$l_i = \arg\max_{\substack{k \in A_i \\ k \le C_i - 1}} R_c\left(\bigoplus_{j=0}^{k} EB_j\right). \tag{5}$$

Remark: The advent of practical rateless codes including Raptor code [10] enables implementation of highly efficient packet-level FEC strategies for reliable data multicasting in wireless networks. Yet, the critical question of accurately quantifying the proper amount of redundancy has remained largely unsolved. To allow the sender to be aware of when it stops sending redundancy, feedback from receivers is inevitable. Our scheme introduces the number of sub-channels l_i tuned at receiver i, which is accurately and automactically adjusted by the receiver itself.

3. Performance Evaluation

3.1 Simulation Model

In the simulation, we consider a point-to-multipoint GEO satellite network. We assume the one-way transmission delay D to be beta-distributed with mean D_{ave} = 250 ms and minimum $D_{\min}$ = 239.6 ms. Its density function $f_D(d)$ is given by

$$f_D(d) = \frac{3(2D_{ave}d - d^2 + D_{\min}^2 - 2D_{ave}D_{\min})}{4(D_{ave} - D_{\min})^3} \times \{u(d - D_{\min}) - u(d - 2D_{ave} + D_{\min})\}. \quad (6)$$

where u(·) is a unit step function.

We assume L = 9 and the data rate is assumed to be 2 Mbps and the packet size is assumed to be 259 bytes including 4-byte header. Furthermore, binary phase shift keying (BPSK) modulation is assumed.

In the simulation, the satellite channel is modeled as an M-state Markov channel which simulates the Rician fading of the received signal envelope. **Fig. 3** depicts the M-state Markov channel model with transition probabilities $P_{i,j}$ from state s_i to state s_j. We assume receivers experience different channel conditions with the Rician factor K = −6, 6, 10, 15, 20 dB. The Doppler frequency shift f_m is assumed to be 0.01 and 100 Hz considering relatively stationary and highly mobile users, respectively. Using K and f_m, we have 10 combinations of channel models which is uniformly assigned to receivers. For instance, each channel model is used for 100 receivers if we have 1000 receiver in simulation. An example of transition probabilities is given in **Table 1** where we assume M = 9, K = 6 dB, and f_m = 0.01 Hz. Derivation of transition probabilities are given in **Appendix B**. With different input parameters, the transition probabilities are easily obtained. Furthermore, each receiver is assumed to experience independent channel errors from its M-state Markov channel.

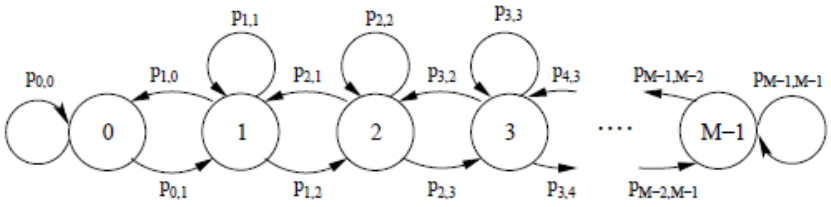


Fig. 3. M-state Markov model which simulates a Rician fading channel with transition probabilities $p_{i,j}$ from state s_i to state s_j

Table 1. Transmission Probabilities for 9-State Markow Model

i	0	1	2	3	4	5	6	7	8
$p_{i,i-1}$	-	3.277e-5	5.071e-5	6.144e-5	6.641e-5	6.615e-5	6.068e-5	4.955e-5	3.141e-5
$p_{i,i}$	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
$p_{i,i+1}$	3.141e-5	4.955e-5	6.068e-5	6.615e-5	6.641e-5	6.144e-5	5.071e-5	3.277e-5	-

Table 2. Channel Codes and their Code Rate used in the Simulation

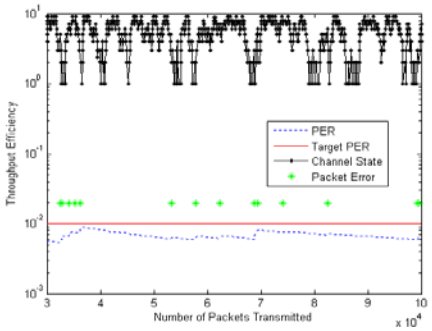
RCPC Rate	8/16	8/15	8/14	8/13	8/12	8/11	8/10	8/9	8/8
Puncturing Index, i	8	7	6	5	4	3	2	1	0
RS(127,119)	0.4685	0.4997	0.5354	0.5766	0.6247	0.6815	0.7496	0.8329	0.9370

In this simulation, 9 concatenated FEC codes are used for channel encoding and decoding where two codes, an *inner code* and an *outer code*, are used in tandem. The inner code corrects most errors and spreads out burst errors, then the outer code corrects the small block errors that remain. In our concatenated FEC code, we employ a rate-compatible punctured convolutional code (RCPC) [13] with a constraint length q = 7 and period p = 8 for the inner code. RCPC codes generate different code rates from an original rate–1/2 convolutional code. Higher and lower code rates can be obtained with puncturing tables by puncturing and repetition, respectively [14]. For the outer code, we exploit a shortened $(n,k,2^8)$ Reed-Solomon (RS) code where n is the block length and k is the information size and we choose n = 127 and k = 119. The RS code is particularly effective at correcting short bursts of errors in a data stream. **Table 2** illustrates the 9 concatenated FEC codes used in our simulation. The resulting code rate is given by

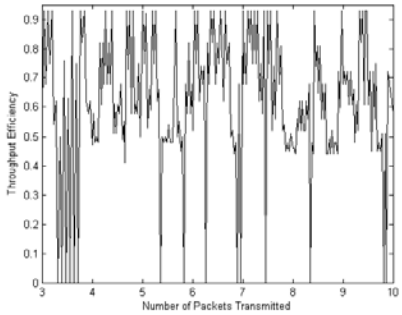
$$R_c = \frac{119}{127} \times \frac{p}{p+i} \tag{7}$$

where i is the RCPC puncturing index.

We also have interleaver/deinterleaver pairs in order to break up burst errors introduced by the channel. The symbol interleaver disperses burst errors out of the inner decoder at the symbol level, while the channel interleaver randomizes channel burst errors at the bit level [14]. Since the symbol interleaver takes input out of the RS encoder, its block length will be n.



(a) Packet error rate vs. the number of packets transmitted



(b) Instantaneous throughput efficiency vs. the number of packet transmitted

Fig. 4. Adaptability performance of the proposed AFEC. (observed from a receiver with channel model parameters SNR=10dB, *K*=6dB, f_m = 0.01 Hz, and target PER=10^{-2})

The choice of symbol interleaver depth depends on several factors including burst error characteristics and delay requirements. Normally, the Viterbi decoder will produce an error burst of length less than four constraint lengths for a convolutional code, when a decoding error is made at the decoder [15]. Therefore, when the symbol interleaver is employed, it can be recommended that its depth be larger than 28 bits ($4 \times q$). In our simulation, we have chosen 64 bits (8 symbols) as the symbol interleaver depth, sufficiently large to handle the burst errors out of the Viterbi decoder. For channel interleaving depth, we choose 300 which is reported to cause an interleaving delay of 400 msec [16]. Note that this amount of delay is reasonable [3] in the satellite multicast services. We further discuss about the performance with various channel interleaving depth in Section 3.3.

In our simulation, BER estimation is not implemented. Instead, we use actual BER for the simulation, and thus we do not have channel estimation error. Also, the performance is measured according to the following metrics:

- *Throughput Efficiency*: the ratio of the number of successfully received data bits to the total number of received bits, and
- *Packet Error Rate* (PER): the ratio of the number of erroneous packets to the total number of received packets.

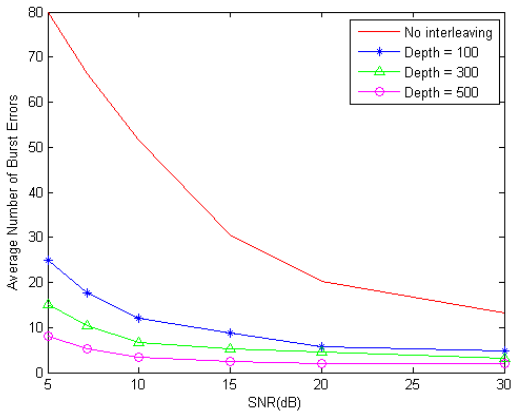
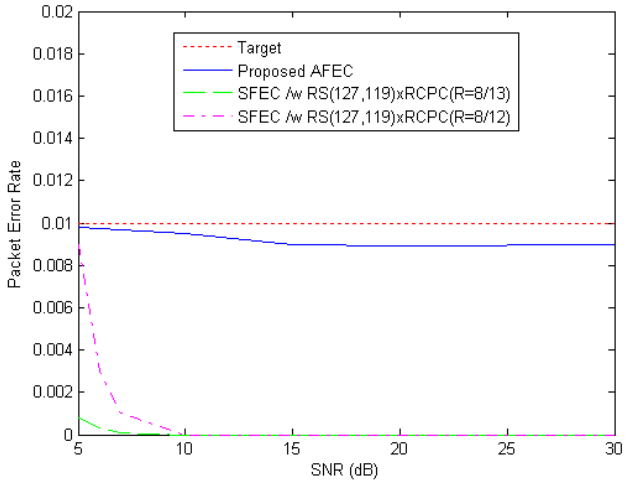


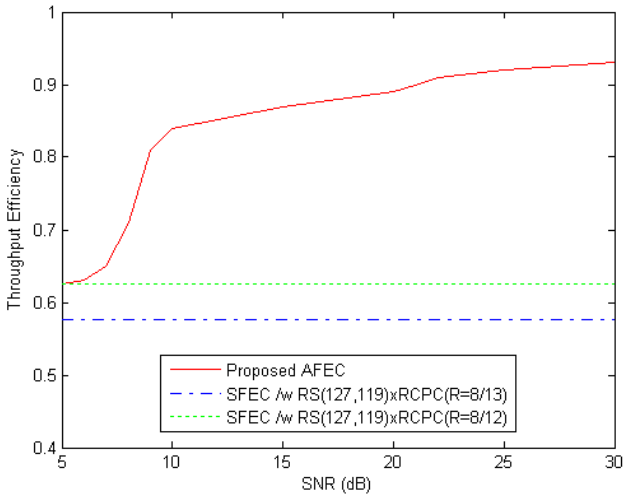
Fig. 5. Average number of burst errors vs. Channel interleaving depth; (observed at a receiver with channel parameters: target PER=10^{-2} ; K= 6dB ; f_m = 0.01 Hz)

3.2 Adaptability

In this experiment, we investigate how the receiver adaptively changes its coding rate and how the packet error rate fluctuates. In **Fig. 4-(a)**, we show the variations of PER and channel condition at a certain receiver with the number of packets transmitted. Here, the receiver has a channel model with parameters SNR = 10 dB, K = 6 dB, f_m = 0.01 Hz. We observe that packet errors occur in the bad channel states. When the error occurs (star), the PER (dotted line) increases but never reaches to the target PER (solid line). **Fig. 4-(b)** shows how the instantaneous throughput efficiency (or code rate) adapts to the varying-channel condition for each packet transmitted. As can be seen, the throughput efficiency drops to zero whenever error occurs, and it achieves high value when the channel becomes good. With respect to throughput efficiency, hence, the proposed AFEC is highly adaptive to the channel variations.



(a) Packet error rate vs. SNR



(b) Throughput efficiency vs. SNR

Fig. 6. Performance of the proposed AFEC. (observed from a receiver with channel parameters of K=15 dB; f_m = 0.01 Hz; and target PER=10^{-2})

3.3 Burst Errors

In **Fig. 5**, we show the average number of burst errors over the SNR at a certain receiver. Without interleaving, we observe unacceptable number of burst errors at low SNRs. For example, at SNR = 5 dB, the number of burst errors is about 80, which results in 20 Kbyte burst error block in 163 msec. However, introducing interleaving, we could effectively reduce the average number of burst errors. With depth = 300, the average number of burst errors achieves less than 15 at 5 dB, which results in 3.8 Kbyte burst error block in 30 msec. Interleaving depth = 500 shows even better performance on the average number of burst errors. However, the delay due to interleaving is reported to be approximately 1.25 sec [16], which may not be good choice for real-time traffic. Therefore, throughout the simulation, we assume the channel interleaving depth=300.

3.4 Throughput Efficiency and Packet Error Rate Comparison

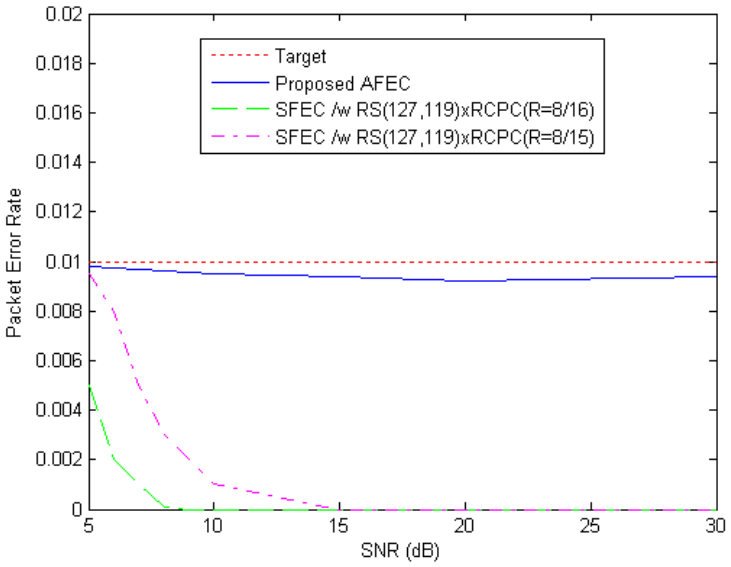
In this section, our objective is to show how the proposed protocol performs in comparison with static FEC (SFEC) schemes with respect to throughput efficiency and packet error rate. Static FEC scheme is a FEC using a single code rate throughout the communications. We ran an experiment with other SFECs from the code set space in **Table 2**. We simulate 10 channel models for 1000 receivers as described in section 3.1. We collect the performance results from receivers in two different channel models. The first one characterizes rather good channel conditions with parameters (target PER = 10^{-2}; Rician factor K = 15 dB; and f_m = 0.01 Hz) simulating receivers with the better line of sight (LOS) component and with rather stationary mobility. For the second model, we use K = 6 dB and f_m = 100 Hz simulating receivers with the worse LOS component and with higher mobility.

Fig. 6-(a) depicts the packet error rate versus SNR with the first model. In **Fig. 6-(a)**, we only show two SFECs which provided acceptable PER ($=10^{-2}$). SFECs with lower coding rate are not shown since we have not seen any errors during simulation. Other SFECs with higher coding rate did not guarantee the target PER ($=10^{-2}$) in all ranges of SNR. We observe that the two SFEC schemes achieve very low packet error rate (PER). In the range SNR $\geq$10 dB, the two SFEC schemes even show no errors. The results also confirm that our proposed AFEC can guarantee the QoS requirement (with target PER = 10^{-2}) for all SNR values.

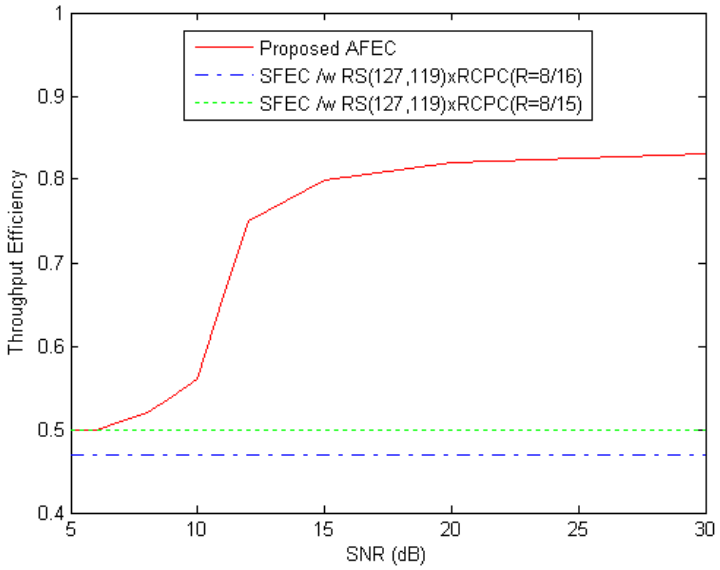
On the other hand, with respect to throughput efficiency, significant gain is achieved for the proposed scheme over the SFECs. **Fig. 6-(b)** shows the average throughput efficiency among 1000 receivers versus SNR. Due to static nature of the SFEC, the SFEC schemes provide almost constant throughput efficiency over the SNR. However, the proposed AFEC practically doubles the throughput efficiency from the low-end to the high-end values of SNR. The tradeoff between throughput and PER in our AFEC scheme becomes clearer. For instance, at low SNRs, the PER is held below the target PER at the expense of high bandwidth consumption. Then, the throughput of the proposed AFEC has a considerable increase at high SNRs.

A similar observation can be found with the second channel model which is shown in **Fig. 7**. In this case, only two SFECs guarantee the target PER while our AFEC still guarantees the target. Overall, slightly reduced throughput efficiency for our AFEC is found in **Fig. 7-(b)**.

As we have observed from the throughput results in **Fig. 6-(b)** and **Fig. 7-(b)**, receivers tend to tune to smaller number of channels at higher SNR. As described in section 2.4, by using this SNR (or bit error rate) as a system requirement, we can significantly reduce the cost of receivers.



(a) Packet error rate vs. SNR



(b) Througput efficiency vs. SNR

Fig. 7. Performance of the proposed AFEC. (observed from a receiver with channel parameters of K=6 dB; f_m = 100 Hz; and target PER=10^{-2})

4. Conclusions

In this paper, we have proposed a new adaptive forward error correction scheme for point-to-multipoint satellite links. Link-layer error control protocols for point-to-multipoint satellite IP networks impose several problems including unreliability and receiver-heterogeneity.

To solve the heterogeneous error rates at different receivers, the proposed scheme exploits multiple multicast channels. We have proposed a packetization scheme for the multiple multicast channels. Based on its own channel condition, the receiver tunes to as many channels as it needs, which prevents from receiving unwanted parities. Therefore, the receiver saves the buffer space, receiving energy, and decoding time, and thus the heterogeneity problem is effectively resolved. Using system requirement for receivers, we can deploy receivers with limited buffer space in relative good channel environments.

Performance evaluation shows that the proposed technique guarantees the target PER by tuning more channels. The technique is shown to be highly adaptive to the channel variation with respect to the throughput efficiency, and provide scalable PER and throughput efficiency. The proposed protocol is expected to support any type of satellite and receivers.

References

[1] S. Cho, "Resource Allocation and Error Control Proto- cols for Real-Time Communications in Satellite Networks," Ph.D Dissertation, Georgia Institute of Technology, Dec. 2002.

[2] S. Cho, A. Goulart, I. F. Akyildiz, and N. Jayant, "An Adaptive FEC with QoS Provisioning for Real-Time Traffic in LEO Satellite Networks," in *Proc. of IEEE ICC, Helsinki, Finland*, pp. 2938-2942, June 2001. Article (CrossRef Link)

[3] V. Weerackody, H. Lou, and Z. Sayeed, "A Code Division Multiplexing Scheme for Satellite Digital Audio Broadcasting," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 11. pp. 1985–1998, Nov. 1999. Article (CrossRef Link)

[4] N. Nikaein, H. Labiod, and C. Bonnet, "MA-FEC: A QoS-Based Adaptive FEC for Multicast Communication in Wireless Networks," in *Proc. of IEEE ICC 2000, New Orleans, USA*, pp. 954–958, June 2000. Article (CrossRef Link)

[5] T. Noguchi, M. Yamamoto, and H. Ikeda, "Reliable Multicast Protocol applying Local FEC," *IEICE Transactions on Communications*, vol. E86-B, no. 2, pp. 690–698, Feb. 2003. Article (CrossRef Link)

[6] F. Tommasi, S. Molendini, and A. Vilei, "The Satellite Multicast Distribution Protocol (SMDP)," in *Proc. of IEEE Softcom, Croatis*, Oct. 2002. Article (CrossRef Link)

[7] W. Si and C. Li, "RMAC: A Reliable Multicast MAC Protocol for Wireless Ad hoc Networks," in *Proc. of IEEE ICPP*, 2004. Article (CrossRef Link)

[8] D. Koutsonikolas, Y. Hu, and C.-C. Wang, "High-Throughput, Reliable Multicast without Crying Babies in Wireless Mesh Networks," in *Proc. of ACM CoNEXT*, Dec. 2008. Article (CrossRef Link)

[9] J. W. Byers, M. Luby, and M. Mitzenmacher, "A Digital Fountain Approach to Asynchronous Reliable Multicast," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1528–1540, Oct. 2002. Article (CrossRef Link)

[10] A. Shokrollahi, "Raptor Codes," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551-2567, June 2006. Article (CrossRef Link)

[11] M. Samokhina, K. Moklyuk, S. Choi, and J. Heo, "Raptor Code-based Video Multicast over IEEE 802.11 WLAN," in *Proc. of IEEE APWCS*, 2008. Article (CrossRef Link)

[12] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel Estimation Techniques Based on Pilot Arrangement in OFDM Systems," *IEEE Transactions on Broadcasting*, vol. 48, no. 3, pp. 223–229, Sept. 2002. Article (CrossRef Link)

[13] J. Hagenauer, "Rate-Compatible Punctured Convolutional Codes (RCPC Codes) and their Applications," *IEEE Transactions on Communications*, vol. 36, no. 4, pp. 389– 400, Apr. 1988. Article (CrossRef Link)

[14] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*, Prentice-Hall Inc., New Jersey, 1995. Article (CrossRef Link)

[15] J. B. Cain and D. N. McGregor, "A Recommended Error Control Architecture for ATM Networks with Wireless Links," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 1, pp.

16–28, Jan. 1997. Article (CrossRef Link)
[16] I. Joe, *Error Control in Wireless ATM Networks*, Ph.D Dissertation, Georgia Institute of Technology, Jul. 1998. Article (CrossRef Link)
[17] H. S. Wang and N. Moayeri, "Finite-State Markov Channel–A Useful Model for Radio Communication Channels," *IEEE Transactions on Vehicular Technology*, vol. 44, no. 1, pp. 163–171, Feb. 1995. Article (CrossRef Link)
[18] S. Rice, "Statistical Properties of a Sine Wave Plus Noise," *Bell System Technical Journal*, vol. 27, pp. 109–157, Jan. 1958.
[19] P. Lettieri, C. Fragouli, and M. B. Srivastava, "Low Power Error Control for Wireless Links," *in Proc. of ACM/IEEE MOBICOM '97, Budapest, Hungary*, pp. 139–150, Sep. 1997. Article (CrossRef Link)
[20] J. K. Tugnait and U. Gummadavelli, "Blind Equalization and Channel Estimation with Partial Response Input Signals" *IEEE Transactions on Communications*, vol. 45, no. 9, pp. 1025-1031, Sept. 1997. Article (CrossRef Link)

Appendix A: A Channel Encoding/Decoding Example for Concatenated FEC Code

This appendix illustrates a snapshot of the channel encoding/ decoding procedure for the concatenated FEC code in Section 3.1. **Fig. (A· 1)** depicts an example of encoding process for a concatenated FEC code RS(170,162)×RCPC($R_c = 8/12$) and decoding process with RS(170,162)×RCPC($R_c = 8/10$). 162-byte information is placed in a parent RS(255,247) code with zero padding. RS(255,247) encoder then creates a 8- byte parity blocks at the end. The 162-byte information block and 8-byte parity block are concatenated into 170-byte shortened RS(170,162) code. By using rate-1/2 convolutional encoder, two 170-byte blocks are created. Then, since period p = 8, two 8 bits are extracted from the two 170-byte blocks, which forms a 16-bit group (8 bits in the upper group and 8 bits in the lower group). With the puncturing table *A* of rate- 8/12 given by

$$A = \begin{bmatrix} 1\,1 & 1\,1\,1\,1 & 1\,1 \\ 1\,0\,1\,0\,1\,0\,1\,0 \end{bmatrix}, \qquad \textbf{(A· 1)}$$

8 bits from the upper group are placed in EB_0, and remaining four bits from the lower group are placed from EB_1 to EB_4. The puncturing table indicates which of the bits are to be punctured prior to transmission for each code rate [14]. Each entry of a rate table is zero or a positive integer. A zero denotes puncturing and a positive integer denotes the number of repetitions of the corresponding code bit. These tables operate on each group periodically with a period of p to generate a family of RCPC codes with code rates: $R_c = p/(p+i)$ where i is the RCPC puncturing index.

The remaining (170-1) blocks are placed as the same method. This results in RS (170,162) × RCPC ($R_c = 8/12$) code with L = 5. This packet is transmitted to all receivers. If a receiver tunes to 3 EBs, RS(170,162) ×RCPC ($R_c = 8/10$) code is used, and corresponding decoding process is performed in the opposite direction of **Fig. (A· 1)**.

Appendix B: Derivation of Parameters for M-State Markov Channel

This appendix describes how to derive the state transition probabilities for *M*-state Markov

channel discussed in Section 3.1. It is noted that any partition of the received signal into a finite number of intervals form a finite-state channel model [17]. Let $0 = r_0 < r_1 < \cdots < r_M = \infty$ denote the thresholds of the received signal. Then, the channel is in state s_i if the received signal is in the interval $[r_i, r_{i+1})$ where $i = 0,1,...,M-1$. From [18], we obtain the crossing rate N_{r_i} for signal level r_i, defined as the expected rate at which the Rice fading envelope crosses a threshold level r_i in a positive-going direction, given by

$$N_{r_i} = \sqrt{2\pi(K+1)}\, f_m \rho_i e^{-K-(K+1)\rho_i^2} I_0(2\rho_i\sqrt{K(K+1)}) \quad \textbf{(A· 2)}$$

where $1 \le i \le M-1$; K is the Rician factor; f_m is the Doppler frequency shift; $\rho_i = r_i / r_{rms}$; r_{rms} is the signal rms value; and $I_0(\cdot)$ is the zero-order modified Bessel function of the first kind which is defined as

$$I_0(x) = \sum_{k=0}^{\infty} \left(\frac{x^k}{2^k k!} \right)^2 . \quad \textbf{(A· 3)}$$

Let π_i be the steady-state probability for state s_i given by

$$\pi_i = P(r \le r_{i+1}) - P(r \le r_i) . \quad \textbf{(A· 4)}$$

Since the cumulative distribution $P(r \le r_i)$ of the Rician distribution is given by $1 - Q_0(\sqrt{2K}, \sqrt{2(K+1)\rho_i^2})$, **(A· 4)** is rewritten as

$$\pi_i = Q_0(\sqrt{2K}, \sqrt{2(K+1)\rho_i^2}) - Q_0(\sqrt{2K}, \sqrt{2(K+1)\rho_{i+1}^2}) . \quad \textbf{(A· 5)}$$

where $Q_0(a,b)$ is the zero-order Marcum Q function defined as

$$Q_0(a,b) = \int_b^{\infty} x e^{-\frac{x^2+a^2}{2}} I_0(ax)dx . \quad \textbf{(A· 6)}$$

The transition probabilities are approximated by [19]

$$\begin{aligned}
&p_{i,i+1} \approx \frac{N_{r_{i+1}}}{\pi_i R_t}, && i = 0,1,...,M-2 \\
&p_{i,i-1} \approx \frac{N_{r_i}}{\pi_i R_t}, && i = 0,2,...,M-1 \\
&p_{i,i} \approx 1 - t_{i,i-1} - t_{i,i+1}, && i = 0,2,...,M-2 \\
&p_{0,0} \approx 1 - p_{0,1} \\
&pM, M-1 \approx 1 - pM-1, M-2 .
\end{aligned} \quad \textbf{(A· 7)}$$

where R_t is a transmission rate in symbols/second.

When binary phase shift keying (BPSK) modulation is assumed, the bit error probability given that the state is s_i is given by

$$P_{b|s_i} = \int_{r_i}^{r_{i+1}} Q\left(\sqrt{\frac{2E_b}{N_0}}r\right) f(r)dr \qquad \textbf{(A· 8)}$$

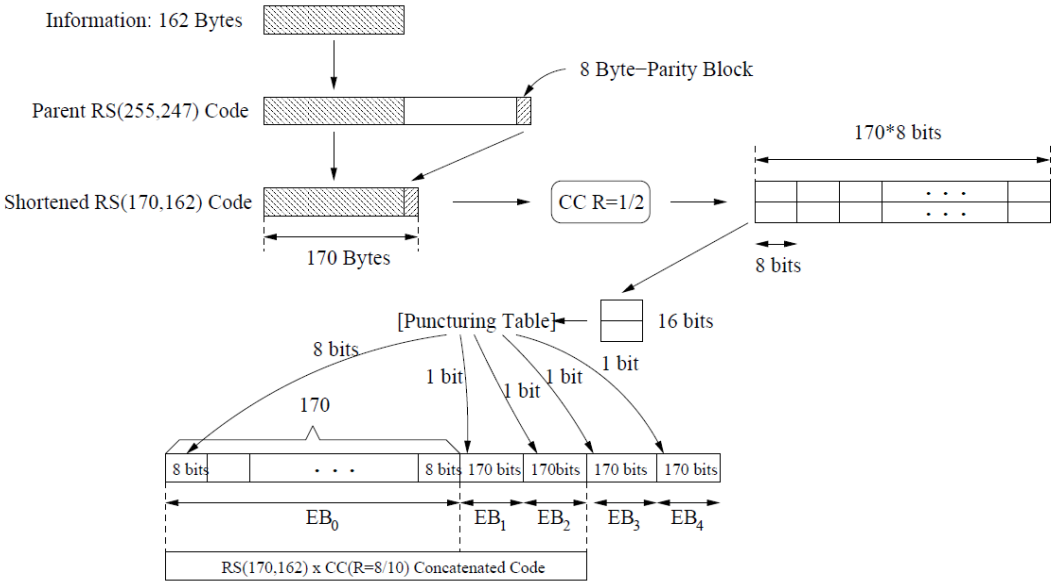


Fig. A· 1. Parity creation using a code RS(170,162) ×RCPC($R_c = 8/12$) and decoding with RS(170,162) ×RCPC($R_c = 8/10$)

Table A· 1. ρ_i and N_{ri} for M = 9 (K = 6 dB and f_m = 0.01 Hz)

i	1	2	3	4	5	6	7	8
p_i	0.5791	0.7162	0.8177	0.9067	0.9926	1.0823	1.1857	1.3272
N_{ri}	0.0036	0.0055	0.0067	0.0072	0.0072	0.0066	0.0054	0.0034

where $Q(x)$ is defined as $Q(x) = 1/\sqrt{2\pi}\int_x^\infty e^{-y^2/2}dy$; $f(r)$ is the Rician density function; E_b is the energy per bit; and N_0 is the one-sided Gaussian noise power spectral density.

As usual approach [17], if we assume $\pi_i = \pi_j, \forall i, j$, we have

$$1 - Q_0\left(\sqrt{2K}, \sqrt{2(K+1)\rho_i^2}\right) = P(r \le r_i) = \frac{i}{M}, \; i = 0,1,...,M \qquad \textbf{(A· 9)}$$

and

$$\pi_i = \frac{1}{M}, \qquad i = 0,1,...,M-1. \tag{A· 10}$$

From **(A· 9)**, we are able to compute ρ_i, and from **(A· 10)** and **(A· 2)**, we can compute the transition probabilities. **Table A· 1** shows the results of ρ_i and N_{ri} from **(A· 9)** and **(A· 2)**, respectively with given parameters (M = 9, K = 6 dB, and f_m = 0.01 Hz).

Sungrae Cho received his B.S. and M.S. degrees in Electronics Engineering from Korea University, Seoul, Korea, and his Ph.D. degree in Electrical and Computer Engineering from the Georgia Institute of Technology, Atlanta, GA in 1992, 1994, and 2002, respectively. He is currently an associate professor in Computer Science and Engineering at Chung-Ang University, Seoul, Korea. Prior to joining Chung-Ang University, he was an assistant professor in Computer Science at Georgia Southern University, Statesboro, GA from 2003 to 2006. In 2003, he was a principal research engineer at Samsung Advance Institute of Technology, Keung, Korea. From 1994 to 1996, He was a member of research staff at Electronics and Telecommunications Research Institute, Daejon, Korea. He is a journal editor of Korea Information and Community Society as well as Korean Institute of Information Scientists and Engineers. He also served a TPC member of ICOIN 2010, 2011, IEEE ICUFN 2010, 2011, IEEE MASS2009, CSIE 2009, IEEE Tridentcom 2007, and IEEE MADWH 2005.