# Testbed Experiment and Theoretical Analysis of SDL Modeled Hybrid Error Control Scheme for Reliable Multicast

Bo Rong, Alain Servais, Donghui Chen*, Maria Bennani, Ahmed K. Elhakeem*, and Michel Kadoch

Department of Electrical Engineering
Ecole de technologie superieure
Montreal, Quebec, H3C 1K3
CANADA

*Department of Electrical and Computer Engineering
Concordia University
Montreal, Quebec, H3G 1M8
CANADA

*Abstract:* - With the evolution of IP multicast technology, group-based real-time applications, such as on-line games and video conferencing, become more and more popular in the Internet. These real-time multicast applications usually have stringent QoS requirements. As one of the most challenging tasks in QoS guaranteed multicast network, the technology of controlling packet loss has attracted a lot of attention. In this paper, we introduce a new hybrid error control scheme that integrates interleaving, forward error correction (FEC), and automatic repeat request (ARQ) to mitigate the error and loss effects encountered in wire and wireless networks. Moreover, in order to facilitate the description and validation of our scheme, specification and description language (SDL) is used for modeling it. In this paper, after devising the basic algorithms of hybrid error control, the corresponding implementations are developed to build up a testbed and attain the actual performance of our scheme. Then, the theoretical analysis is conducted to support and extend our experimental results.

*Key-Words:* - Reliable multicast; hybrid error control; SDL

## 1 Introduction

Nowadays, multicast is becoming more and more important due to the rapid growth of group-based real-time multimedia applications in the Internet. As one of the most challenging tasks in multicast, the technology of controlling packet loss has attracted a lot of attention. In the past, usually mere packet replication was suggested to combat the best effort delivery and subsequent TPDU (transport layer PDU) loss of the Internet, but now this gives way to the powerful erasure codes. These erasure codes, such as Reed Solomon (RS) Codes, have strong inherent erasure correction capability [1,2].

Although previous work has been done on employing forward error correction (FEC) and automatic repeat request (ARQ) in multicast [3~5], few actual buildups and experiments which integrate all the techniques, such as interleaving, FEC and ARQ, have been reported. This paper concentrates on an actual experiment of interleaved FEC/ARQ error correction. The testbed results show that the combination of interleaving, FEC and ARQ can provide strong multicast error control and excellent QoS guarantee at high user data rate. Moreover, the theoretical analysis is conducted to support and extend our testbed results.

The rest of the paper is organized as follows. Firstly, we introduce our SDL modeling of interleaved FEC/ARQ error correction scheme in Section 2. Then the experimental results are demonstrated and analyzed in Section 3. In Section 4, the theoretical analysis of our scheme is presented. Finally, Section 5 summarizes our results.

## 2 Our SDL Modeled Hybrid Error Control Scheme

### 2.1 SDL Modeling

We use specification and description language (SDL) as a formal technique to model our hybrid error control scheme. SDL is an object-oriented formal language standardized by ITU-T, and it has been developed for modeling telecommunication systems including data communication network [6,7]. Our scheme is based on a Client/Server architecture, and the main entities of the new technique are the server and client algorithms which reside on top of the Internet UDP layer. The server routine typically exists at the sender of video/audio multicast or one of the intermediate routers (called domain receiver or DR in multicast terminology). The client routine exists at the end user or the receiver part of DR. As the SDL outline of our hybrid error control scheme, Fig.1 shows the system model. Furthermore, the algorithms running on the server side and client side are depicted by FSMs in Fig.2 and Fig.3 respectively.

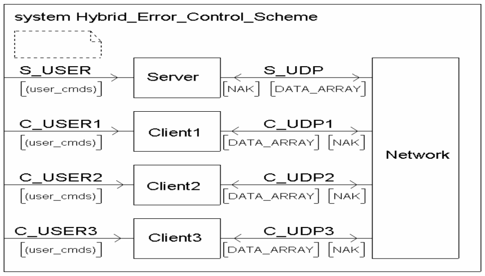


Fig. 1. The system model of our hybrid error control scheme

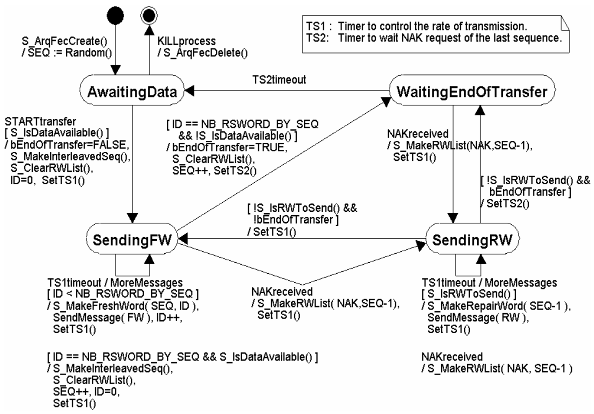


Fig. 2. Server FSM

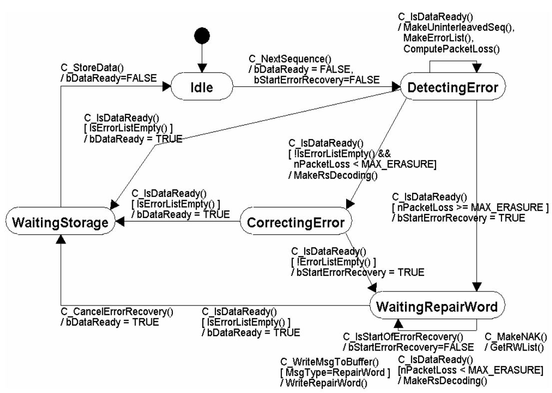


Fig. 3. Client FSM

## 2.2 TPDU Format Used in Our Scheme

The format of frames sent from the server to the clients is shown in Fig.4. From this figure, we can know clearly how the application data are encapsulated within the frame. Each block of data, 218 bytes, is given a sequence number (NO) in the range from 0 to 65535. A one byte data length (DL) field is also added to indicate the end of receiving because the data may be less than 218 bytes. If the data are less than 218 bytes, the remainder of the 218 bytes is filled with padding. A two byte Cyclic Redundancy Check (CRC) field is appended at the end of the word, and this CRC operates over the whole word. The resulting 223 bytes are fed to the (255,223) Reed Solomon encoder routine [1] which outputs a corresponding 255 bytes that we call an encoded word. The RS code is assumed to be systematic, in which case the 32 parity bytes are added by the encoder at the end of the word. A one byte of dummy data is added to obtain a 256 bytes encoded word. Interleaving is next applied to a record of 256 such encoded RS words to yield a corresponding interleaved record.

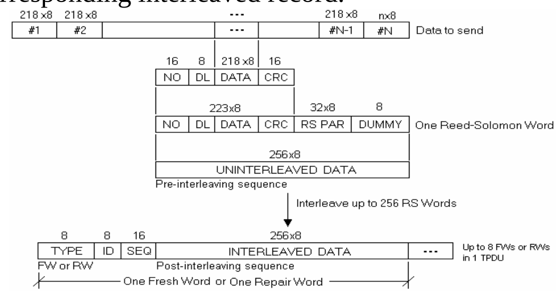


Fig. 4. TPDU format

Due to interleaving, if any 32 words are lost, only 32 out of 255 bytes in each corresponding pre-interleaving word would be lost. Because RS decoder can conceal up to 255-223= 32 lost bytes, such loss is not detrimental and will not result in retransmission requests. For multicast error concealment over the Internet, minimizing the need of retransmissions is an essential asset which subsequently leads to minimizing NAK and repair packet implosions. As shown in Fig.4, the server routine groups up 8 interleaved words into one UDP data unit (TPDU). Upon request of the client, retransmitted words are also included in some TPDUs. A one byte type filed (TYPE) is given to each TPDU to indicate this UDP packet carries fresh words or retransmitted words (11111111 for fresh words and 10101010 for retransmitted words). Furthermore, a one byte ID filed is also added to denote the position of this TPDU inside the interleaved record. Although each original RS encoded word has a sequence number field (NO), this information is scrambled by the interleaving afterward. Therefore, a new sequence number field (SEQ) is added to each TPDU. From the above discussion we know, if an interleaving depth of 256 is used, each TPDU contains 8 interleaved words, then 4 consecutive TPDUs can be lost without any retransmission.

The format of NAK packet in our scheme is shown in Fig.5, and it is the only packet type that the client can send. In Fig.5, we use 01010101 as the value of TYPE field to denote NAK messages. A one byte field DL denoting the length of NAK message is also added. Moreover, each packet is given a sequence number (SEQ) in the range from 0 to 65535. After this, the subsequent field ID_LIST gives ID information of the lost words that need to be retransmitted from the server to the client. At last, one byte CRC field is added for error checking at the server side. Constrained by the delivery deadline, in our scheme, one lost packet cannot be retransmitted more than twice.

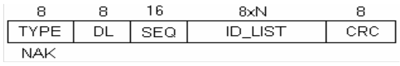


Fig. 5. NAK packet format

## 3 Experimental Results

Fig.6 shows a simple outline of our experimental environment. Although there are 7 PCs (one serves as server and others serve as clients) involved in the multicast experiment, for the sake of simplicity, only the server and one client are shown in Fig.6. All 7 PCs are connected by 100Base-T Ethernet virtual LANs, while Linux real-time platform is running on them. Moreover, the IP Wave Network Impairment Emulator (IP Wave) on Windows NT platform is configured as the default router between different virtual LANs to resemble the error and loss effects between server and clients.

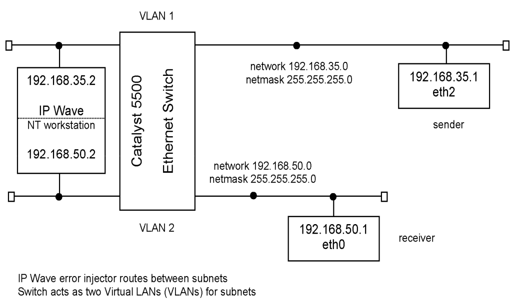


Fig. 6. Block diagram of our experimental testbed

Many tests have been conducted for different packet loss and byte errors. The main results obtained in our experiment are shown from Fig.7 to Fig.9. Fig. 7 shows that the NAK generating does not increase quickly when the data rate and packet loss are growing, due to the fact that many damaged words are corrected by the erasure correction capability of RS codes. Fig.8 demonstrates the percentage of those words that are delivered correctly to the receiver buffer before the final time out. From this figure, we can know that the overall efficiency of our scheme is satisfying. By subtracting the values in Fig. 8 from 100%, we get the residual errors in Fig.9. The residual errors reflect the percentage of those words that our scheme cannot save. In addition, bandwidth expansion due to FEC/ARQ utilization may bring occasional congestion, packet loss and errors. However, testbed results indicate that the performance of our scheme is not compromised as long as the speed of CPUs at server and clients is capable of handling the real time interleaved FEC/ARQ coding/decoding algorithms herein.

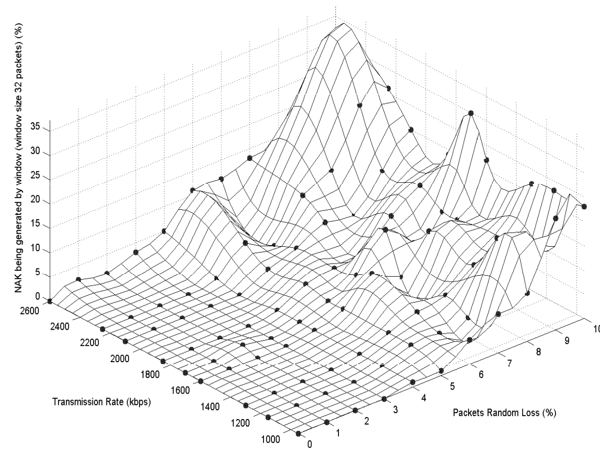


Fig. 7. Percentage of NAK generating vs. data rate and packet loss (our interleaved FEC/ARQ scheme)

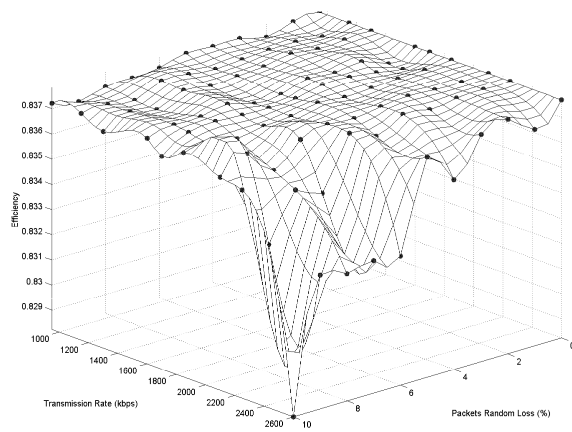


Fig. 8. Efficiency vs. data rate and packet loss (our interleaved FEC/ARQ scheme)

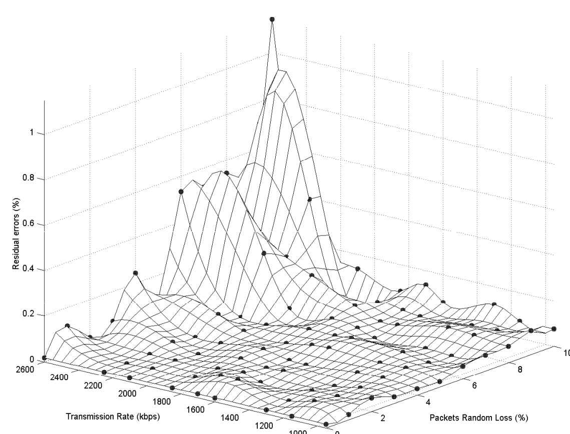


Fig. 9. Percentage of residual errors vs. data rate and packet loss (our interleaved FEC/ARQ scheme)

# 4 Theoretical Analysis

In this section, we intend to conduct the theoretical analysis of our interleaved FEC/ARQ scheme step by step. At first, we analyze the basic characteristics of our scheme in unicast. Then, its performance in multicast environment is investigated. We present a performance comparison between our interleaved FEC/ARQ scheme and the ARQ only scheme. We evaluate the performance enhancement of our scheme in terms of the number of transmissions per packet and the number of NAKs. In addition, only the sending data rate, group size, and packet loss rate, but no random byte error, are considered in this section.

Firstly, we assess the performance of interleaved FEC/ARQ scheme in Unicast with the following notations.

$P$: Probability of packet loss.

$I$: Number of lost packets.

$NN$: Number of interleaved RS words per packet (8).

$N$: Total number of packets in an interleaved RS words block (32).

$K$: Number of packets containing original data in an interleaved RS words block (28).

$P'$: Probability that a packet is RS decoded incorrectly.

$P_{no\ NAK}$: Probability that no NAK is generated for a certain block at receiver in our interleaved FEC/ARQ scheme.

$P_{NAK}$: Probability that a block generates NAK at receiver in our interleaved FEC/ARQ scheme.

$P_{error\ RS/ARQ}$: Expected residual packet error of our interleaved FEC/ARQ scheme.

$P_{error\ ARQ}$: Expected residual packet error of the ARQ only scheme.

Each packet contains 8 interleaved RS words, so one packet loss implies only 8 symbols lost in each of 256 RS encoded words. Since (255, 223) Reed-Solomon codes can correct 32 erasures per RS word, the loss of 4 packets is not detrimental and would not result in retransmission requests. When the number of lost packets is more than 4, NAK will be generated. Constrained by the delivery deadline, NAK cannot be generated more than twice for one interleaved block. For the same reason, retransmission only contains original data. So, after receiving retransmitted packets, no RS decoder is called. Formula (1) gives the probability that a packet cannot be RS decoded correctly and has to be retransmitted.

$$P' = \frac{1}{N} \sum_{i=N-K+1}^{N} i \binom{N}{i} (1-P)^{N-i} P^{i} \qquad (1)$$

For a systematic RS words block, there is no NAK being generated when the number of packets lost is less than $N$-$K$. Therefore, the probability that there is no NAK being generated for a block is:

$$P_{no\ NAK} = \sum_{i=0}^{N-K} \binom{N}{i} (1-P)^{N-i} P^{i} \qquad (2.a)$$

The probability that a block generates NAKs for one or more packets at one receiver is:

$$P_{NAK} = 1 - P_{no\ NAK} \qquad (2.b)$$

Limiting the number of NAKs for one interleaved block to 2, the expected residual error of one packet in the interleaved FEC/ARQ scheme is:

$$P_{error\ RS/ARQ} = P'P^2 \qquad (3)$$

Formula (3) describes the case that one RS decoding and two retransmission trials cannot result in final packet reception. Without FEC, the residual error rate is:

$$P_{error\ ARQ} = P^3 \qquad (4)$$

Formula (4) describes the case that one transmission and two retransmissions cannot attain final packet reception in the ARQ only scheme.

After discussing the unicast performance, group communication performance of our interleaved FEC/ARQ scheme is evaluated as follows. In order to simplify the performance evaluation, we make several assumptions on the network model. We assume that the packet loss rate of a link is not affected by the sender data rate. This is reasonable when the congested links used by our protocol also carry many other sessions. We also suppose that the repair words are always sent from sender to receivers by multicast, and all the receivers have the same packet loss rate. Moreover, NAKs from the various users are assumed to be aggregated and sent by using unicast. If some sub-trees share the same loss, we treat them as one member when calculating the equivalent group size of an independent loss. The above assumptions make the network model much easier for performance evaluation. The notations used in this part are described as follows:

U: Equivalent group size of independent loss.

$P'_{NAK}$: Probability that a block generates NAK the second time at one receiver in our interleaved FEC/ARQ scheme.

$\alpha$: Probability that sender has to retransmit a packet in our interleaved FEC/ARQ scheme.

$\beta$: Probability that sender has to retransmit a packet in the ARQ only scheme.

$\alpha'$: Probability that the second trial of packet retransmission is needed in our interleaved FEC/ARQ scheme.

$\beta'$: Probability that the second trial of packet retransmission is needed in the ARQ only scheme.

$E(TRAN_{RS/ARQ\_group})$ : Average number of transmissions per packet among the multicast group in our interleaved FEC/ARQ scheme.

$E(TRAN_{ARQ\_group})$ : Average number of transmissions per packet among the multicast group in the ARQ only scheme.

$E(NAK_{RS/ARQ\_group})$ : The average number of NAKs per block among the multicast group in our interleaved FEC/ARQ scheme.

$E(NAK_{ARQ\_group})$ : The average number of NAKs per block among the multicast group in the ARQ only scheme.

In our interleaved FEC/ARQ scheme, $(1-P')^U$ is the probability that no member of the multicast group needs packet retransmission. So, the probability that the sender needs the first trial of packet retransmission is:

$$\alpha = 1-(1-P')^U \tag{5}$$

Obviously, the average number of receivers who need retransmission is $P' \cdot U$ , and the probability of packet loss in retransmission is *P*. Therefore, as to our interleaved FEC/ARQ scheme, an approximation of the probability that the second trial of packet retransmission has to be employed is shown below.

$$\alpha' \approx 1-(1-P)^{P' \cdot U} \tag{6}$$

Similarly, the probability that sender needs to retransmit a packet in the ARQ only scheme is:

$$\beta = 1-(1-P)^U \tag{7}$$

An approximation of the probability that the second trial of packet retransmission is needed in the ARQ only scheme is:

$$\beta' \approx 1-(1-P)^{P \cdot U} \tag{8}$$

As for the FEC/ARQ scheme, the first transmission of an interleaved RS block will take N packets. The retransmissions will only have the first K original data packets. In the second retransmission, the packet has to be accepted as is no matter it is right or wrong. For the above reason, the number of transmissions per packet in our interleaved FEC/ARQ scheme is:

$$E(TRAN_{RS/ARQ\_group}) = \frac{1}{K}\left((1-\alpha)N + \alpha(1-\alpha')\cdot(N+K) + \alpha\cdot\alpha'(N+2K)\right)$$

$$\therefore \quad E(TRAN_{RS/ARQ\_group}) = \frac{N}{K} + \alpha + \alpha\cdot\alpha' \tag{9}$$

Similarly, the number of transmissions per packet in the ARQ only scheme is:

$$E(TRAN_{ARQ\_group}) = (1-\beta)\cdot 1 + \beta(1-\beta')\cdot 2 + \beta\cdot\beta'\cdot 3$$

$$\therefore \quad E(TRAN_{ARQ\_group}) = 1 + \beta + \beta\cdot\beta' \tag{10}$$

An approximation of the probability that a block generates NAK the 2nd time at one receiver in our interleaved FEC/ARQ scheme is:

$$P'_{NAK} \approx 1-(1-P)^{K \cdot P'} \tag{11}$$

By using $P'_{NAK}$ and $P_{NAK}$, we can get the following formula:

$$E(NAK_{RS/ARQ\_group}) = \left(P_{NAK}\cdot(1-P'_{NAK})\cdot 1 + P_{NAK}\cdot P'_{NAK}\cdot 2\right)\cdot U$$

$$\therefore \quad E(NAK_{RS/ARQ\_group}) = U\cdot P_{NAK}(1+P'_{NAK}) \tag{12}$$

Similarly, we get the following result for the ARQ only scheme:

$$E(NAK_{ARQ\_group}) \approx U\cdot\left(1-(1-P)^K\right)\cdot\left(2-(1-P)^{PK}\right) \tag{13}$$

To achieve numerical results, we assume that the multicast group size ranges from 1 to 10000 and the packet loss rate changes from 0 to 10 percent. Moreover, we choose 28 packets as the window size of the ARQ only scheme. Therefore, the window size of the ARQ only scheme is equal to the size of original data in one interleaved RS block.

Fig.10 shows the number of transmissions per packet of both interleaved FEC/ARQ scheme and ARQ only scheme (based on formula 9 and 10). Firstly, we would like to compare the result in Fig.10 with the achievement of other literatures. In [8], the number of necessary transmissions for correctly transferring a single packet without any protecting measure is shown in Fig.1. As we expected, the numerical result of ARQ only scheme in Fig.10 can match the result in Fig.1 of [8] very well. To some extent, the above comparison can justify our assumptions and analysis. Secondly, we compare the numerical result of our interleaved FEC/ARQ scheme in Fig.10 with the experimental result in Fig.8 under the condition that the group size is 6. For the sake of comparison, *the number of transmissions per packet* in Fig.10 has to be transformed to the *Efficiency* in Fig.8 by calculating its reciprocal. After the transformation, we find the result in Fig.10 is a little lower than the experimental result in Fig.8 in terms of *Efficiency*. That is reasonable, because the result in Fig.10 is for the worst case. Thirdly, let us compare the performances of interleaved FEC/ARQ scheme and ARQ only scheme in Fig.10. Due to the coding gain of FEC, there is an obvious difference between two schemes, especially when the packet loss rate is less than 4% and the group size is large. When the packet loss rate is high and group size is large, our interleaved FEC/ARQ scheme needs more transmissions because of the parities in the first transmission. But this does not necessarily mean the ARQ only scheme has better performance, because this small amount of parities can contribute to less number of NAKs and less residual errors. From

Fig.11, we can see the strong capability of RS coding on minimizing NAKs while group size changes from 1 to 10000 (based on formula 12 and 13). When the group size is 6, under the transformation that the result in Fig.11 is divided by the *group size*, the performance of our interleaved FEC/ARQ scheme in Fig.11 can match the experimental result in Fig.7 very well. From the above discussions, we can conclude that our theoretical analysis has supported and extended our experimental results successfully.

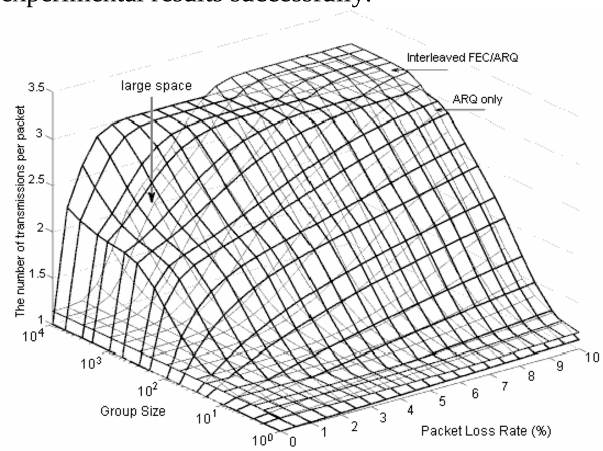


Fig. 10. The number of transmissions per packet (our scheme and ARQ only scheme)

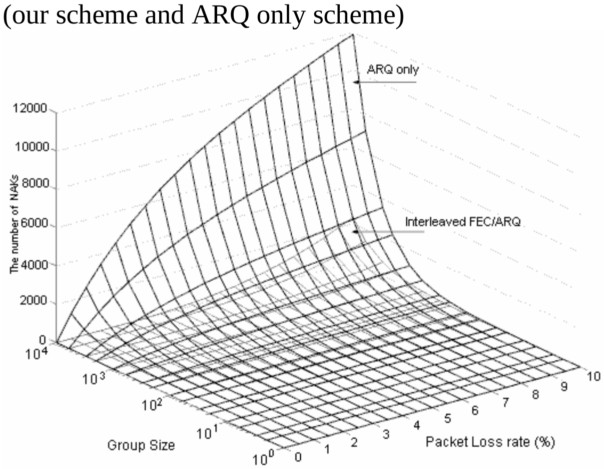


Fig. 11. The number of NAKs generated per window (our scheme and ARQ only scheme)

In summary, compared with the ARQ only scheme, our interleaved FEC/ARQ scheme has less transmissions and NAKs. Furthermore, there is a large range of packet loss and group size where our interleaved FEC/ARQ scheme can be applicable.

## 5 Conclusion

In this paper, a hybrid error control scheme is presented to provide good QoS experimental results in case of multicast transmissions. While some theoretical results have been recently documented for miscellaneous techniques of reliable multicast, our laboratory work shows the practical feasibility of integrating all these techniques. According to our testbed results, the receivers can correct most of the packet loss and byte errors while operating at a reasonable data rate, such as 1Mbps video transmission. Moreover, the theoretical analysis reveals that our interleaved FEC/ARQ scheme can work satisfyingly when the group size is under 100 and packet loss rate is less than 10%. Currently, the experiments are being extended to teleconferencing applications.

*References:*


[1] S. B. Vicker, *ERROR CONTROL SYSTEMS FOR DIGITAL COMMUNICATIONS AND STORAGE*. Prentice-Hall, 1995.

[2] B. Fong, P. B. Rapajic, G. Y. Hong, and A. C. M. Fong, Forward Error Correction with Reed-Solomon Codes for Wireless ATM Networks, *The 2002 WSEAS Int. Conf. on Electronics, Control & Signal Processing*, Paper Number: 451-210, Singapore, December 9-12, 2002.

[3] Christos Pappadopoulos, Guru Parulkar, and George Varghese, An Error Control Scheme For Large-Scale Multicast Applications, *INFOCOM'98*, Volume: 3, pp. 1188 – 1196, 29 March-2 April 1998.

[4] S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang, A reliable multicast framework for light-weight sessions and application level framing, *IEEE/ACM Trans. on Networking*, Volume: 5, Issue: 6, pp. 784–803, Dec. 1997.

[5] Abdullah Alwehaibi, Anjali Agarwal, Michel Kadoch, and Ahmed Elhakeem, Hybrid FEC/ARQ Delay for DiffServ Over IP and MPLS Multicast Networks, *WSEAS AIC-ISTASC-ISCGAV 2003*, Paper Number: 458-146, Rhodes, Greece, Nov.15-17, 2003.

[6] Sandhu K. K., Specification and description language (SDL), *IEE Tutorial Colloquium on Formal Methods and Notations Applicable to Telecommunications*, pp. 3/1-3/4, Mar 1992.

[7] Chie Dou, A timed-SDL for performance modeling of communication protocols, *IEEE GLOBECOM '95*, Volume: 3, pp. 1585-1589, Nov. 1995.

[8] Villela D. A. M. and Duarte O. C. M. B., Improving Scalability on Reliable Multicast Communications, *Computer Communications*, Vol. 24 (5-6), pp. 548-562, March 2001.