UDC

An Efficient Loss Recovery for On-demand Peer-to-Peer Streaming

[1] Mitsuo Hayasaka, [2] Tetsuya Miki
[1,2]Department of Electronics, Faculty of Engineering, The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan, E-mail: mhysk@ice.uec.ac.jp

Abstract
Peer-to-Peer multimedia streaming is expected to grow rapidly in the near future and will consist of variety of media. These media requires a guaranteed end-to-end Quality of Service (QoS). However, packet losses during transmission are a serious problem for them as they result in degradation of the QoS. Forward Error Correction (FEC) is a promising technique to recover the lost packets and improve the QoS of streaming media. In the conventional media streaming, pre-buffering of media stream and FEC techniques are required to cope with delay jitter and packet losses. However, FEC may degrade the QoS of all streaming due to the increased congestion caused by FEC overhead when streaming sessions increase. In this paper, a novel FEC method that avoids the adverse effect of FEC overhead during media streaming is proposed. It accommodates the delay jitter even without pre-buffering of media stream. It also provides the QoS according to requirements and environments of users by using layered coding of FEC. Thus, packet losses and jitter are recovered at each end host and do not affect the next-hop streaming. The results of computer simulations show that our proposed method provides better media quality compared to conventional methods.

***Keywords**: Peer-to-Peer streaming, Packet loss ratio, Forward error correction*

1 Introduction

Real-time multimedia applications over the Internet are increasingly becoming popular with the rapid growth of the Internet and the operational cost reductions. They require a guaranteed end-to-end Quality of Service (QoS). These applications can be categorized into real-time interactive and on-demand streaming communications. The demand for on-demand streaming services such as Video on Demand (VoD), e-learning systems and TV broadcasting over the Internet are expected to grow very rapidly. These applications consist of many media such as text, moving picture, image, audio, etc. According to the user requirements and the equipment at user and network sides, a media-specific QoS is required. Especially audio media require a higher level of quality compared to other media. According to [1,2], the audio stream with good quality can compensate for the degradation of the overall perceptual quality of audio-video applications even if the quality of the video stream is somewhat low. Therefore, it is desirable to provide higher QoS of audio as much as possible.

The peer-to-peer (P2P) multimedia streaming is expected to grow rapidly with the development of broadband networks. This technique can provide a wide-area streaming service and load balancing without changing the existing routers of the network. Advantage of streaming using P2P technique is that packet losses and jitter occurred at links between end hosts can be recovered at each end host. Then the end host can forward streaming media to next-hop end host.

In the on-demand streaming services, main factors responsible for degradation of QoS are jitter and packet losses. The effect of jitter could be overcome by pre-buffering of media stream at the receiver end. Packet losses are considered as a main factor that degrades the QoS of streaming media. They can be solved by implementation of forward error correction (FEC) scheme, which is applied for the end-to-end communications as well as the P2P streaming. Thus, conventional audio streaming methods need both pre-buffering and FEC techniques to cope with jitter and packet losses, respectively. However, it is well-known that conventional FEC scheme may degrade network condition due to the increased congestion caused by FEC overhead when streaming sessions increase. This paper is focused on a P2P audio streaming, which requires a higher quality.

It proposes a novel FEC scheme that can overcome the increased congestion caused by the FEC overhead during media streaming. It also can accommodate delay jitter without pre-buffering of media stream while improving packet loss of streaming audio. Furthermore, it provides the QoS according to requirements and equipments of users by using the layered FEC coding. Thus, packet losses and jitter are recovered at each end host and do not affect the next-hop streaming. As a result, it can provide a higher quality audio even in a wide-area streaming by repeating this loss recovery process.

The rest of this paper is organized as follows. Section 2 describes the conventional media streaming schemes. In Section 3, we discuss the proposed method in detail. Section 4 consists of the numerical analysis of the proposed concept. The performance of the proposed method is evaluated by way of computer simulations, and the results are presented in Section 5. Implementation of proposal is considered in Section 6. Finally this paper is concluded in Section 7.

2. Conventional Audio Packet Transmission

Conventional audio streaming methods need pre-buffering of media stream and FEC technique to cope with jitter and packet losses respectively. According to [23], typical pre-buffering sizes of audio stream for accommodating jitter range from 50 to 100 ms. In other words, the jitter up to pre-buffering size at receiver ends can be overcome. The packet loss recovery techniques can be categorized into 2 types. First is a closed loop type, which is called Automatic Repeat reQuest (ARQ) and second is an open loop type, which is called Forward Error Correction (FEC).

2.1. Automatic Repeat reQuest

When packet losses occur, the sender will retransmit lost packets according to the notification of lost packets from the receiver. The retransmission delay is unacceptable for streaming media as it is very sensitive to delays during media streaming. Moreover, the packet losses in the Internet occur due to the network congestion. When retransmission is done, it may cause more congestion leading to more packet losses, resulting in a network collapse. Therefore, it is necessary for the retransmission methods in streaming media to control the source transmission rate according to the network condition [3,4]. This scheme is based on the Additive-Increase Multiplicative-Decrease (AIMD) algorithm [24]. When network is congested, source coding rate is reduced multiplicatively at sender end. Thus, this scheme avoids the network congestion. In other words, this scheme gives a high priority to the retransmitted packets to make sure that they are transmitted to the receiver end at the expense of source rate. The multiplicative decreasing of source rate in streaming media results in degradation of user-level QoS. When the network congestion is reduced, the source rate is increased additively. Therefore, it takes a longer time for the reduced source rate to revert back to the original rate before congestion. When the P2P streaming with this technique is used, the degraded streaming media is forwarded to next-hop end host. Therefore, the QoS of these media is highly degraded as the number of end host forwarding the streaming increases. Even if TCP is used for the streaming, this situation occurs since TCP is performed according to AIMD algorithm as well. From these descriptions, although it is clear that ARQ is an effective technique to solve the network congestion, it is not considered as a suitable scheme for the P2P streaming. Therefore, FEC techniques are preferred.

2.2. Forward Error Correction

There are 2 types of FEC techniques for audio streaming [5], namely an audio specific FEC called Audio Redundancy Coding (ARC) and parity FEC using error correcting code. Both of these techniques require a redundant bandwidth, which is called FEC overhead.

2.2.1. Audio Redundancy Coding

In Audio Redundancy Coding (ARC) [6-8], each packet contains both a primary coding of audio media and a redundant copy of a preceding frame, in a more heavily compressed format (secondary coding). The coding scheme is illustrated in Fig. 1. As this method provides resilience to loss with an acceptable latency, it is suitable for streaming media. If a packet loss occurs, the receiver can use the redundant copies to fill in any gaps in the original data stream. Since redundant copies are more heavily compressed than primary, lost packets that are retrieved using them would not be exactly the same as the original, but it is perceptually better than a gap in the stream. However, it results in a degradation of user-level QoS compared to original primary audio stream.

When P2P audio streaming with ARC is used, the secondary audio is forwarded to the next-hop end host if the packet loss occurs during transmission at current end host. The degradation of QoS at one hop affects the next-hop streaming in this scheme. If packet loss occurs near the original sender end using P2P streaming, secondary audio are provided in the remaining end hosts, leading to a degradation of user level QoS of these hosts. Therefore it is not considered as a suitable scheme for P2P streaming, as well.

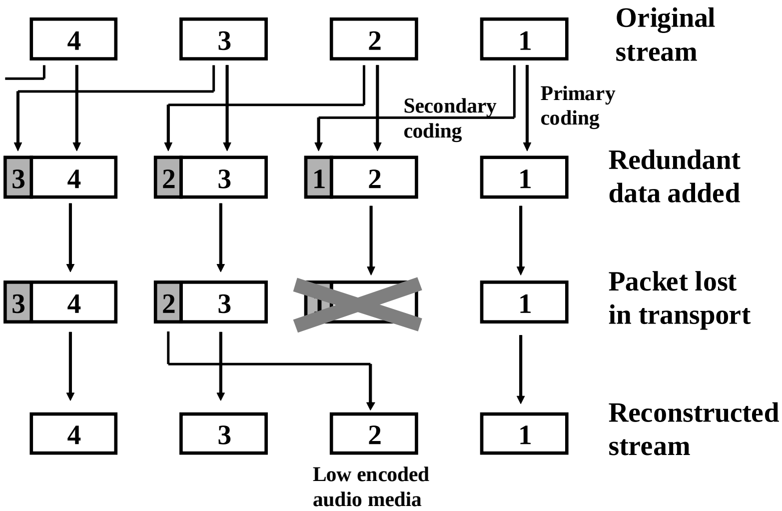


Figure 1. Audio redundancy coding

2.2.2. Parity FEC

The redundant packets, which are generated from original media packets by using an error correction code, are transmitted along with the media packets so that the lost original packets can be recovered using them [9]. This is illustrated in Fig. 2. When FEC with (n,k) block code is applied, where n is the total number of packets and k is the number of media packets, it adds (n-k) redundant FEC packets for every k media packets. Notation n and k are called the block length and the data length respectively. When there are packet losses, if any k packets of n block length are received at receiver end, all original media packets within n block length can be recovered using FEC. When k is equal to (n-1), it is called parity FEC that are used for audio streaming. In contrast to ARC, the quality of the lost packets retrieved using FEC codes are the same as the original media packets. When P2P audio streaming with this parity FEC is used, packet losses can be recovered using FEC if they are within the FEC recovery ability. Then, lost packets recovered by FEC are forwarded to next hop end host.

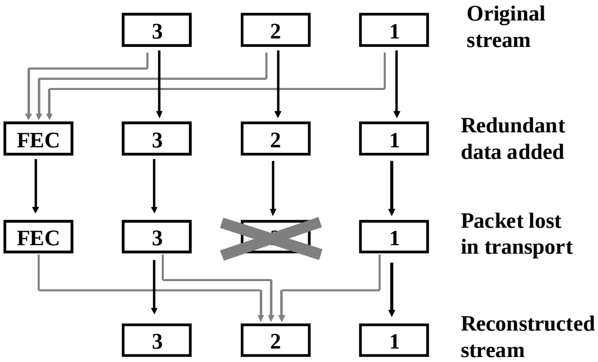


Figure 2. Repair using parity FEC

2.2.3. Issues in FEC techniques

Generally FEC overheads increase the network congestion with the increase of streaming sessions using FEC technique, leading to more packet losses. This may degrade the QoS of all communications sent to the networks due to the increased congestion caused by FEC overhead. This is a well-known problem in a FEC technique. In ARC, since the provision of secondary audio increases in this situation, the QoS of each end host is highly degraded. Similarly, in parity FEC, these packet losses are beyond the FEC recovery ability and they cannot be recovered using FEC. One popular technique to avoid this effect is to limit the bandwidth on which FEC is applied. When FEC using an error correcting code is applied to just a part of the whole bandwidth, the QoS of that part is considerably improved, but the QoS of the remaining part is degraded slightly [10-12]. If this limitation is applied, it is valid to assume the FEC overhead is negligible. This assumption is widely used in all conventional FEC methods and is valid for the past and current networks, where the bandwidth of access networks is small and the amount of real-time traffic is little. However, in the near future the real-time multimedia applications will increase drastically due to the implementation of broadband access networks. Therefore, future networks will have more real-time traffic. The assumption to neglect the FEC overheads might be invalid for the future networks.

In conventional streaming schemes, the same FEC techniques are applied for both real-time interactive and on-demand streaming applications without distinguishing them. As session durations of these real-time applications tend to increase, redundant packets are continuously added to media stream for longer duration. Since FEC is created instantaneously for real-time interactive communications and is sent along with media packet, it is very difficult to avoid the FEC overhead being added to media packets through out the duration of the session. However, if FEC overheads of on-demand streaming media can be avoided, this is an attractive solution because FEC overheads are caused by only the real-time interactive applications. It is likely to accommodate the adverse effect of FEC overhead within the limitation of bandwidth that FEC can be applied without degradation. Therefore, a new FEC method for on-demand streaming media that does not increase the congestion caused by the FEC overhead during the media streaming is required. As a result, QoS of both real-time interactive and on-demand streaming applications can be improved because the FEC overheads are negligible as like in the past and current networks.

From these descriptions, it is desirable to have the following features in such a FEC scheme.

- Provide primary audio without heavily compressed secondary audio even when a packet loss occurs.
- Recover packet losses at each end host using P2P streaming.
- Avoid the increased congestion caused by FEC overheads during media streaming.

3. Referential Loss Recovery

In parity FEC method, the repaired stream will be the same as the original stream. We focus on this advantage and a novel FEC method is proposed.

3.1. Concept of Referential Loss Recovery (RLR)

Streaming media is very sensitive to transmission delay. However, regarding the on-demand media streaming, there is a permissible delay time before its communication starts. It is called the start-up delay [13]. We propose the referential loss recovery (RLR) method, which separates FEC packets from original media packets and send them to the receiver end before media streaming starts. The FEC content that consists of FEC packets is created when audio content is created. The FEC content is transmitted using TCP in order to make sure they are received correctly at the receiver end. Therefore, the adverse effects of FEC overheads can be ignored because of the congestion control based on AIMD in TCP, described in Section 2.1. TCP provides a reliable transmission at the expense of transmission time when it is applied for data communications. Also it can adjust its transmission rate according to the network congestion. As usual media packets are transmitted using RTP and UDP. The transmission sequence of our proposed method is shown in

Fig 3. If media packets are lost, they are recovered by referring to the FEC content already received at the receiver end. Thus, the proposed method can provide original primary audio without heavily compressed secondary audio even when a packet loss occurs. It also can avoid the increased congestion caused by FEC overhead during media streaming.

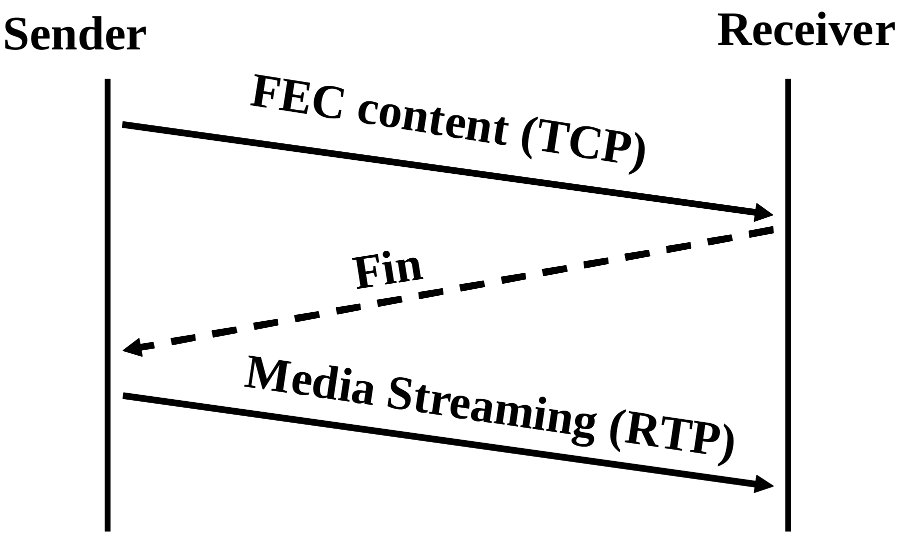


Figure 3. Transmission sequence in proposed method

In addition, it can use already received FEC packets to generate some of the original media packets before all of media packets arrive at the receiver, and this can be used as a solution for delay jitter [14]. For example, when FEC with (3,2) Single Parity Check (SPC) code is applied, basically any 2 packets are enough to generate original media packets in 3 block length. In our proposed FEC method, one media packet is enough to recover original media packets in a block length. This media packet and the already received FEC packet can be used to generate the other media packet. If the arrival of the other packet excesses the requirement of delay jitter, this packet is created by FEC content. Thus it can accommodate delay jitter. In the conventional media streaming, pre-buffering of media stream and FEC technique are required to overcome jitter and packet loss respectively. On the other hand, the RLR scheme only downloads FEC content before media streaming starts but this may cause some start-up delay. Although this scheme does not need the pre-buffering of media stream, it is expected that the start-up delay of RLR will be longer than conventional methods. In other words, this scheme avoids the adverse effect of FEC overheads at the expense of the start-up delay, and improves the packet loss ratio that is a main factor degrading the QoS of media streaming.

When P2P audio streaming with RLR is used, packet losses can be recovered at each end host before the each end host forwards it to next-hop end host. Since packet loss and jitter are recovered before the next-hop streaming, each end host can receive an original quality of streaming media without degradation. In other words, the degradation of QoS at one hop does not affect the next-hop streaming. Therefore, this technique is suitable for on-demand P2P media streaming.

3.2. Layered FEC coding for delivery of FEC content

In RLR, the delivery way of FEC content for each end host is an important factor to be considered. When the FEC content is generated, a layered coding is applied to provide different FEC recovery ability according to the user requirement of QoS and equipment. The concept of layered FEC is already proposed in [15]. In this layered FEC method, multicast groups of media streaming and FEC provisions are created separately. Some FEC multicast groups are created to provide different loss recovery ability. The packet loss of media stream is recovered by combining appropriate FEC multicast group. However, the increased congestion caused by FEC overhead cannot be solved in this scheme. Generally, packet losses due to FEC overheads are beyond the FEC recovery ability and they cannot be recovered using FEC. Here, we propose a layered coding method of FEC suitable for P2P streaming using RLR. There are two layers, namely base layer and extended (ext) layer in layered coding. It is assumed that FEC with (4,3) SPC code is applied. As shown in Fig. 4, the encoding process of layered coding at sender end is as follows. FEC packets are processed and created by (4,3) SPC code from primary encoded media packets. The base layer will have only one FEC packet ('a') for every two FEC packets ('b' and 'c') generated by (4,3) SPC code. This results in one FEC packet in base layer for every 6 media packets. In other words, base layer FEC ('a') is generated by (7,6) SPC code and this can be easily obtained by sending the two FEC packets ('b' and 'c') through an exclusive-or gate ('b' XOR 'c' = 'a'). The ext layer consists of all odd FEC packets generated by (4,3) SPC code. The decoding process of layered coding at receiver end is as follows. When the receiver (end host) receives only the base layer of FEC content, lost packets are recovered using (7,6) SPC code. If both base and ext layers are received, it is possible to obtain the original (4,3) SPC generated FEC packets, as shown in Fig. 5. In other words, FEC packet ('b') can be directly obtained from the ext layer and FEC packet ('c') can be obtained by sending FEC packets ('a' and 'b') through exclusive-or gate. These generations of original (4,3) SPC FEC packets are done before receiving the media packets because FEC content is sent in advance. Therefore lost packets are recovered using (4,3) SPC code.

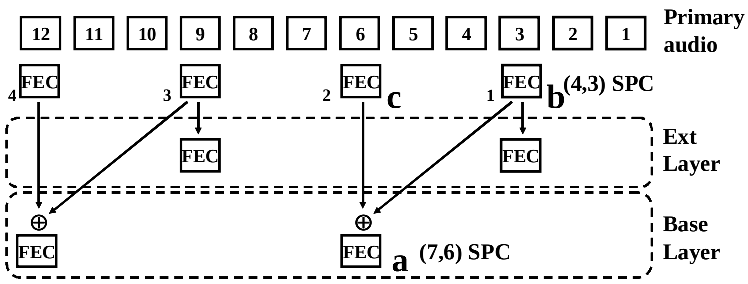


Figure 4. Encoding process of layered FEC coding

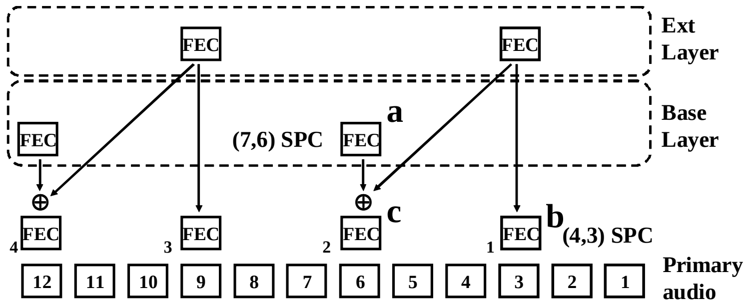


Figure 5. Decoding process of layered FEC coding

Basically the base layer of FEC content is sent to each end host. After the delivery of base layer the ext layer of FEC content is sent to the end hosts that require a guaranteed QoS and all hosts that have the bandwidth capacity to accommodate both layers. Sending the ext layer content after the base layer makes the delivery of base layer fast.

In this method, the length of the start-up delay time to download all FEC content is one of the key factors to be considered. The proposed method does not need pre-buffering of media stream. Furthermore, both broadband access network environment and layered coding of FEC content will make sure the delay time is well within the tolerable limits.

4. Numerical Analysis

In this analysis, the methods of ARC, conventional parity FEC and RLR are compared. Let ρ_0 denote the initial network load without considering the FEC overheads. When FEC technique is applied to all real-time communications, it requires extra bandwidth, resulting in an increase of the network load. The realistic network load, ρ_{real} is given by;

$$\rho_{real} = (1-\alpha)\rho_0 + \frac{\alpha\rho_0}{R_{cr}} \tag{1}$$

where α and R_{cr} are the percentage of real-time traffic and the coding rate respectively. Let T be the threshold of real-time traffic that FEC can be applied without degradation. In the past and current networks, the percentage of real-time traffic is small and α is usually less than T. In this scenario, FEC overheads are negligible. Therefore,

$$\rho_{real} \approx \rho_0 \quad . \tag{2}$$

The lost packets could be recovered efficiently by using FEC technique.

However, the percentage of real-time traffic increases rapidly with the development of broadband networks. The α is expected to excess the threshold T in the future networks. Adding FEC packets to real-time communications in order to guarantee their QoS causes more network congestion, leading to more packet losses. These packet losses are beyond the FEC recovery ability and they cannot be recovered using FEC. Furthermore, session durations of recent multimedia applications tend to increase and redundant FEC information are added to these longer durations. FEC overheads cannot be ignored anymore. Then, the proposed RLR method is effective as a solution to this problem.

Real-time multimedia applications can be categorized into real-time interactive and on-demand streaming communications. Therefore,

$$\alpha = \alpha_{Interactive} + \alpha_{Ondemand} \tag{3}$$

where $\alpha_{Interactive}$ and $\alpha_{Ondemand}$ are the percentages of real-time interactive and on-demand streaming traffic respectively. The realistic network loads of each ARC, ρ_{arc} and conventional parity FEC methods, ρ_{fec} are calculated as follows.

$$\rho_{arc} = \rho_0 + (\alpha_{Interactive} + \alpha_{Ondemand})\rho_0\left(\frac{1}{R_{arc}} - 1\right) \tag{4}$$

$$\rho_{fec} = \rho_0 + (\alpha_{Interactive} + \alpha_{Ondemand})\rho_0\left(\frac{1}{R_{fec}} - 1\right) \tag{5}$$

where R_{arc} and R_{fec} are the coding efficiencies of each method. They are given by;

$$R_{arc} = \frac{P_{rate}}{P_{rate} + S_{rate}} \tag{6}$$

$$R_{fec} = \frac{k}{n} \tag{7}$$

where S_{rate} and P_{rate} are coding rates of secondary and primary audios in ARC respectively. The k and n are data and block lengths in conventional parity FEC, respectively. From (4) and (5), it is clear that each FEC overhead increases network load. However, in RLR scheme, FEC overhead of on-demand streaming media can be ignored. Therefore, the realistic FEC overhead is caused by only the real-time interactive applications and the percentage of these applications is unlikely to exceed the threshold T. Therefore, the FEC overheads can be neglected as in the current networks. The realistic network load of RLR scheme, ρ_{rlr} is given by;

$$\rho_{rlr} = \rho_0 + (\alpha_{Interactive})\rho_0\left(\frac{1}{R_{fec}} - 1\right) \approx \rho_0 \tag{8}$$

As this network load is applied for both real-time interactive and on-demand streaming applications, QoS of both applications are improved. In other words, RLR scheme can make the best use of FEC recovery ability for these appli cations of the future networks.

The packet loss ratios of ARC, conventional FEC and RLR methods according to a network load that each method provides are calculated using simple M/M/1/K queue. Simple M/M/1/K queue can easily model the loss process in the network [25]. Their packet loss ratios are given by;

$$P_{loss\_arc} = \frac{1-\rho_{arc}}{1-\rho_{arc}^{K+1}}\rho_{arc}^{K} \tag{9}$$

$$P_{loss\_fec} = \frac{1-\rho_{fec}}{1-\rho_{fec}^{K+1}}\rho_{fec}^{K} \tag{10}$$

$$P_{loss\_rlr} = \frac{1-\rho_{0}}{1-\rho_{0}^{K+1}}\rho_{0}^{K} \tag{11}$$

where K is buffer size. From (9) and (10), it is clear that ARC and conventional FEC methods increase each packet loss ratios due to each FEC overhead.

Finally, the play out ratios of primary audio of ARC, conventional parity FEC and proposed RLR methods in P2P streaming are evaluated. The play out ratio of primary audio is defined as the ratio of number of primary audio packets that are played out to the total number of audio packets at receiver ends. An advantage of proposed RLR method using P2P streaming is that lost packets can be recovered completely at each end host if they are within the FEC recovery ability and the degradation due to FEC overhead is negligible. Then, the recovered stream is forwarded to next-hop end host. Therefore, the packet loss does not affect the next-hop streaming. Assuming that the number of end hosts forwarding the streaming media is M, the play out ratios of primary audio in each method are given by;

$$P_{playout\_arc} = 1 - MP_{loss\_arc} \tag{12}$$

$$P_{playout\_fec} = \sum_{j=1}^{M}\sum_{i=0}^{n_j-k_j} {}_{n_j}C_i P_{loss\_fec}^{i}\left(1-P_{loss\_fec}\right)^{n_j-i} \tag{13}$$

$$P_{playout\_rlr} = \sum_{j=1}^{M}\sum_{i=0}^{n_j-k_j} {}_{k_j}C_i P_{loss\_rlr}^{i}\left(1-P_{loss\_rlr}\right)^{k_j-i} \tag{14}$$

where n_j and k_j are the block and data lengths used at each hop. According to (12) and (13), the audio quality of ARC and conventional parity FEC methods is degraded with the increase of M. Especially, the audio quality of ARC scheme is drastically degraded. In contrast, proposed RLR scheme does not degrade audio quality with the increase of level M.

5. Performance Evaluations

The performance of RLR scheme is evaluated by way of computer simulations. It is assumed that audio redundancy coding uses ITU G.726 ADPCM (Adaptive differential pulse code modulation) with 32kbps as a primary coding and ITU G.729 CS-ACELP (Conjugate Structure-Algebraic Code Excited Linear Predictive) with 8kbps as the redundant copy of secondary coding [16,17]. The conventional FEC and RLR methods use G.726 as audio coding and SPC codes are applied as a FEC. The on-off model is used as the voice source traffic since human speech consists of talk-spurts and silence gaps that are known as on-off patterns. The holding time in on and off periods is assumed to be exponentially distributed with mean of 1.004 s and 1.587 s, respectively [18]. In all simulations an audio packet is produced for every 20ms in on-periods and no packet is produced in off-periods. It is called Silence Suppression [24].

5.1. Evaluation between 2 end hosts

Suitable FEC codes were determined by evaluating characteristics of play out ratio of primary audio and pre-buffering time between two end hosts as a design example of RLR.

A loss pattern of audio packets is considered to be random because bursty losses in audio packets are very small and can be neglected. It is well-known that packet losses in IP networks tend to be bursty for aggregated traffic in the network. If a generation of audio packets is bursty, the QoS of them is very much affected by the bursty losses. However, when we focus on an end-to-end audio streaming, its loss pattern will be random as the generation of audio packets is periodic with a period of 20ms. From this point of view, the SPC code has enough ability to recover lost audio packets and therefore has been used for conventional audio streaming. In the proposed method, since all FEC content are transmitted before the streaming starts, any lost media packets can be recovered if they are within the FEC recovery ability. The play out ratio of primary audio is evaluated when random packet loss ratio varies from 0.01 to 0.1 between two end hosts. The play out ratios of primary audio with different proposed FEC codes are shown in Fig. 6. The play out ratio of primary audio decreases as random loss ratio increases in all methods. It also decreases as the block length of SPC code increases in the proposed method.

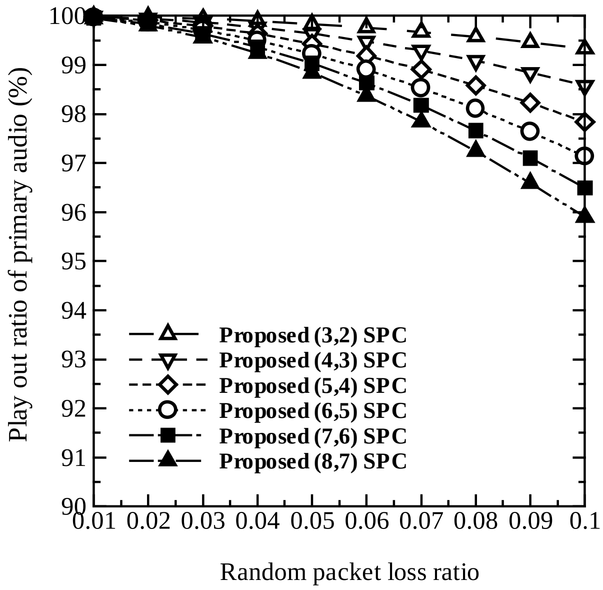


Figure 6. Play out ratio of primary audio for different FEC block codes in RLR between 2 end hosts

Usually, the jitter is overcome by pre-buffering of media stream in conventional audio streaming and the proposed FEC method can accommodate the jitter without pre-buffering as described in section 3.1. The network jitter for audio packet delivery can be approximated to the standard deviation of a normal distributions that represent the inter arrival time of packets [19,24]. Simulations were performed for an IP telephony system with 20 ms audio frames. Therefore, a normal distribution with an average of 20 ms was used for the evaluation of jitter. In this simulation, any delay beyond a threshold of 200 ms from the playout point was considered as a packet loss. Then the pre-buffering time required to accommodate the jitter that can be accommodated by the proposed FEC method for a given packet loss ratio was found for different jitter values, as shown in Fig. 7. Since audio packets are generated with silence suppression, the play out point can be adjusted during silent periods at the receiver end. Adjusting the play out point during a talk spurt will cause an audible glitch in the output, but a small change in the length of the silent period between talk spurts will not be noticeable [24]. It was assumed that the play out point is adjusted by changing the length of the silent period in the simulations. Fig. 7 indicates that the jitter that can be accommodated by the proposed (3,2) SPC method is the same as that of around 209 ms of pre-buffering. This is the best possible performance that can be achieved by the proposed FEC methods and the amount of jitter that can accommodate decreases with the increase of the block length in SPC code. Each FEC code has an ability to accommodate jitters more than 200 ms pre-buffering. According to [23], typical buffer sizes for accommodating jitter range from 50 to 100 ms. From this evaluation, it is clear that the proposed FEC methods have enough ability to accommodate jitter. Therefore, proposed RLR can omit pre-buffering of media stream and this will reduce the start-up delay that is longer than conventional method.

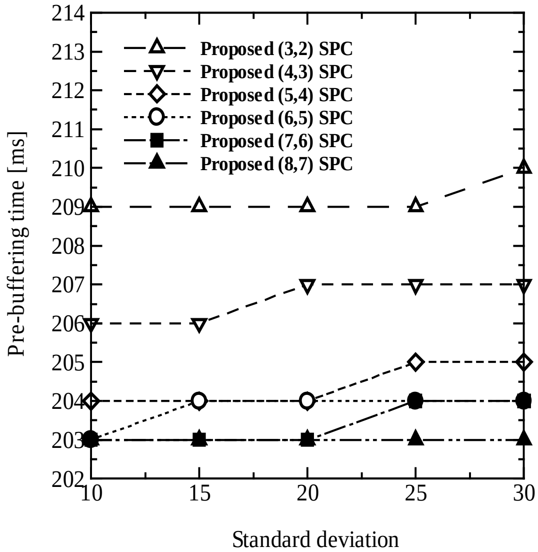


Figure 7. Pre-buffering time

5.2. Evaluation of application level multicast

Application Level Multicast (ALM) [20-22] is used as an example of P2P streaming and its characteristics are evaluated by a 3-level binary tree, as illustrated in Fig. 8. There are many routers between each end hosts. Table 1 shows 3 proposed methods with the adoption patterns of delivery of base and ext layers at each level in the 3-level binary tree. The performance of each proposal is evaluated when random packet loss ratio at each link varies from 0.01 to 0.1. Although many kinds of multicast trees are constructed in ALM, the evaluation using binary tree is enough because the play out ratio of primary audio is observed according to the variation of random packet loss ratio in each link. Therefore, the play out ratio of primary audio is not sensitive for different multicast trees. From the results of Fig. 6, all proposed FEC methods have high play out ratios as close as 100%. The (5,4) SPC and (3.2) SPC codes are selected for the base and ext layers of FEC content respectively as a design example of RLR. Proposal 1 in Table 1 should provide the high performance with respect to play out ratio of primary audio since both base and ext layers are applied for all levels. The play out ratios of primary audio of other proposals are also evaluated for comparison. The effects of FEC overheads for ARC and conventional parity FEC methods are calculated considering the silence suppression. In conventional parity FEC method, the applied patterns of FEC codes at each level in ALM tree are the same as Table 1 for comparison.

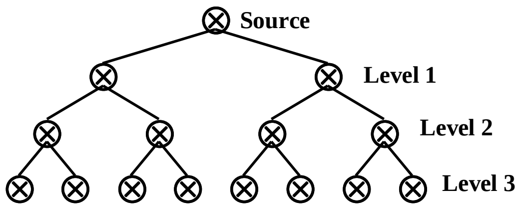


Figure 8. 3-level binary tree

TABLE I. DELIVERY WAY OF BASE AND EXT LAYERS TO END HOSTS

Method	Level in binary tree		
	1	2	3
Proposal 1 Parity FEC 1	Base + Ext	Base + Ext	Base + Ext
Proposal 2 Parity FEC 2	Base + Ext	Base + Ext	Base
Proposal 3 Parity FEC 3	Base + Ext	Base	Base

Figures 9, 10 and 11 show the play out ratio of primary audio of all 3 proposals at levels 1, 2 and 3 respectively are highly improved compared to the conventional methods. It is observed that ARC and conventional parity FEC schemes degrade the play out ratio of primary audio as the level of binary tree increases. Therefore, when an n-level binary tree where n is a large number is assumed, the audio quality of end hosts at n-level is highly degraded. On the contrary, the proposed methods keep high play out ratios close to 100% even if the level of binary tree increases. There is a huge improvement of play out ratio of primary audio in the proposed method. It is clear that the proposed RLR scheme can provide much higher audio quality regardless of the number of levels in the multicast tree. From these evaluations, the proposed FEC method, which can recover lost packets at each end host without the degradation due to FEC overheads, is suitable for a wide-area P2P streaming services to provide higher media quality.

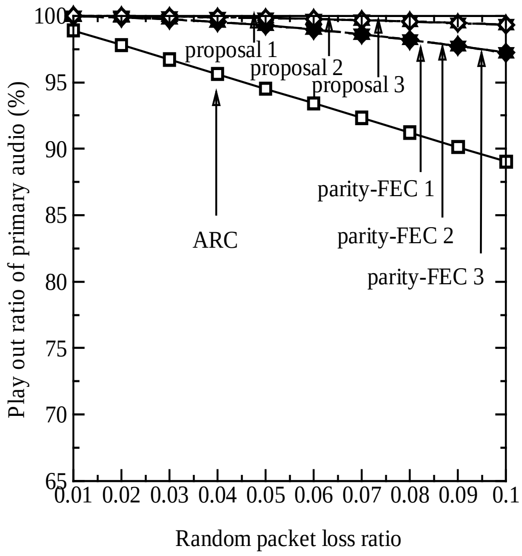


Figure 9. Play out ratio of primary audio at level 1 in binary tree

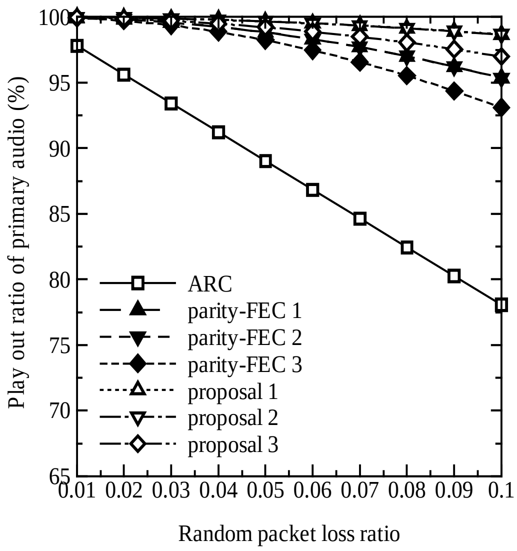


Figure 10. Play out ratio of primary audio at level 2 in binary tree

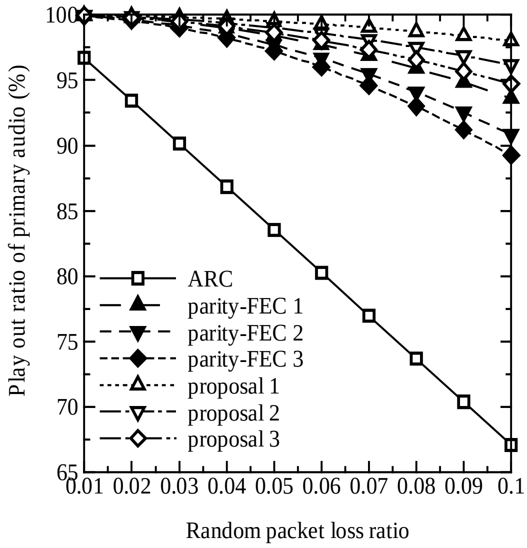


Figure 11. Play out ratio of primary audio at level 3 in binary tree

Finally, the FEC content size is calculated because it affects the start-up delay time, which is an important factor to be considered in RLR scheme. Its size depends on the media content length, the bit rate of audio media and the applied FEC code. Here, G.726 is adopted as an audio encoding. The (5,4) SPC and (3,2) SPC codes are applied as FEC codes of base and ext layers, respectively. The silence suppression is also applied. Table 2 shows the FEC content size of each base and ext layers when different media content lengths are applied. All content sizes are small even for 90 minutes of media content lengths. In the future, the FTTH technology will be deployed, increasing the capacity of the access network drastically. In this scenario the download time of FEC content will be very short for both base and ext layers and the start-up delay time will be within the acceptable limits.

TABLE 2. FEC Content Size

Media Playout Time (min.)	FEC Content Size (MB)	
	Base layer	Ext layer
30	0.68	0.68
60	1.36	1.36
90	2.04	2.04

Table 3 summarizes performances of all 3 proposals in terms of primary audio play out ratio, accommodation of delay jitter and start-up delay time. The start-up delay time varies subject to the FEC content size. The proposal 1 is the best performance of primary audio play out ratio but increases the start-up delay time as both base and ext layers are delivered to each end host. Proposal 3 is vice versa to proposal 1. From Fig. 7, both (3,2) SPC and (5,4) SPC codes have an enough ability to accommodate the delay jitter. Therefore, all proposals can cope with delay jitter. It can be concluded that when each end host has a higher bandwidth the proposal 1 is suitable. If the bandwidth of each end host is limited, the proposal 3 is suitable to reduce the start-up delay time.

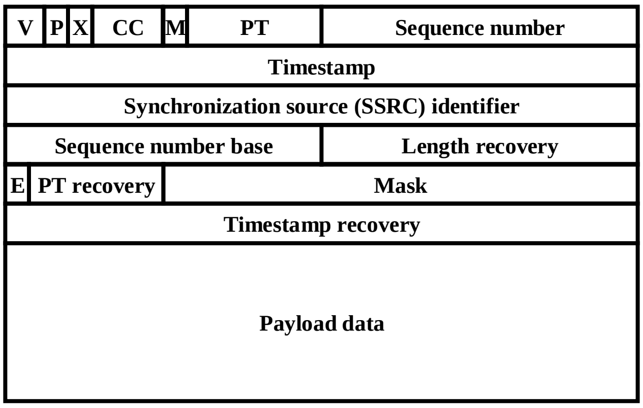
TABLE 3. Summary of 3 Proposals

Method	Comparison Parameters		
	Play out ratio of primary audio	Accommodation of delay jitter	Start-up delay time
Proposal 1	Best	Possible	Good
Proposal 2	Better	Possible	Better
Proposal 3	Good	Possible	Best

6. Consideration of Implementation

In the conventional media streaming, the FEC packets are usually sent as a separate RTP stream, on a different UDP port but to the same destination address, in order to make FEC methods backward-compatible [24]. Therefore, the synchronization between FEC and media packets is important. In [9], the FEC packet format is described as shown in Fig. 12. The sequence number base and mask fields in the payload header are used for the synchronization. The sequence number base field indicates the minimum sequence number of the original media packets composing this FEC packet. In other words, this field indicates the first media packet in the FEC block length. The mask field is a bit mask indicating which of the packets following the sequence number base are included in the FEC operation. These 2 fields can detect the appropriate media packets that each FEC packet can be correctly applied for.

The difference between RLR and conventional streaming is the location of FEC packets. Since the FEC stream generated from the FEC content already at the receiver ends in the RLR can be considered as a separate RTP stream in conventional streaming, the RLR is feasible with the same packet format and software as [9], and P2P software.



V: version number **M: marker**
P: padding **PT: payload type**
X: extensions **E extension**
CC: list of contributing sources

Figure 12. FEC packet format

7. Conclusion

The current and future multimedia delivery services via the Internet are provided with many media. Especially, the audio media requires a high quality. The demand for wide-area streaming services rapidly increases and the P2P streaming is highly expected as one of the implementations of such services. In this paper, we propose a novel FEC method to provide a guaranteed QoS for on-demand P2P audio streaming. In the proposed method, it is suggested to generate the FEC content when media content is created and transmit them before media streaming starts. It avoids the increased congestion caused by FEC overhead. By the implementation of layered coding of FEC, it can provide different QoS according to the user requirements and equipment. Packet losses and jitter are recovered at each end host and do not affect the next-hop streaming. Thus, it is possible to provide an original primary audio rather than low quality secondary audio, accommodate delay jitter and omit the pre-buffering of media stream. This scheme does not need the feedback information and highly improves the audio quality compared to the conventional methods. The FEC content size for audio media is small, and it is a suitable audio streaming scheme for the future broadband Internet.

In the future, the defined protocol should be evaluated in the real Internet.

Reference

1 S. Tasaka, and Y. Ishibashi, "Mutually Compensatory Property of Multimedia QoS," in Proc. of ICC2002, pp.1105-1111, April/May 2002.
2 S. Tasaka, and Y. Ito, "Psychometric Analysis of the Mutually Compensatory of Multimedia QoS," in Proc. of ICC2003, pp.1880-1886, May 2003.
3 M. Handly, S. Floyd, J. Padhye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," IETF, RFC 3448, 2003.
4 R. Rejaie, M. Handley, and D. Estrin, "RAP: An End-to-End Rate-Based Congestion Control Mechanism for Real-time Streams in the Internet," Infocom, Mar. 1999.
5 C. Perkins, O. Hodson, and V. Hardman, "A Survey of Packet Loss Recovery Techniques for Streaming Audio," IEEE Network, September/Octorber, 1998.
6 C. Perkins, et al, "RTP Payload for Redundant Audio Data," RFC 2198, Sep. 1997.
7 M. Podolsky, C. Romer, and S. McCanne, "Simulation of FEC-Based Error Control for Packet Audio on the Internet," IEEE Infocom '98, Apr. 1998.
8 J.C. Bolot, and A.V. Garcia, "The case for FEC-based error control for packet audio in the Internet," to appear in ACM Multimedia Systems, 1997.
9 J. Rosenberg and H. Schulzrinne, "An RTP Payload format for Generic Forward Error Correction," RFC 2733, Dec. 1999.
10 H. Ohta, and T. Kitami, "A Cell Loss Recovery Method Using FEC in ATM Networks," in IEEE Trans.,J-SAC, vol.9, no.9, pp.1471-1483, December 1991.
11 R. Iwase, and H. Obara, "A Bit Error and Cell Loss Compensation Method for ATM Transport Systems," in IEICE Trans.,Vol.J75-B-I,no.1,pp.1-11, January 1992.
12 T. Kitami, "Coding Matrix Structure and Performance Evaluation for FEC based Cell Loss Recovery Method," in IEICE Trans., vol. J82-B, no. 4, pp.580-590, April 1999.
13 C. Carle, and E.W. Biersack, "Survey of Error Recovery Techniques for IP-based Audio-Visual Multicast Applications," IEEE Network, Vol.11, no.6, December 1997.
14 M. Hayasaka, M. Gamage and T. Miki, "Referential Loss Recovery for Streaming Audio using Application Level Multicast," in Proceeding of APCC2005, pp.264-268, Octorber 2005.
15 W. Tan, and A. Zakhor, "Video Multicast Using Layered FEC and Scalable Compression," in IEEE trans., VOL.11, No.3, March 2001.
16 R. V. Cox, and P. Kroon, "Low Bit-rate Speech Coders for Multimedia Communication," IEEE Communications Magazine, pages 34-41, December 1996.
17 A white paper, "Speech Codecs," Wipro Technologies, http://www.wipro.com/prodesign/focusarea/multimedia/knowledge.htm
18 ITU-T Rec. P.59, "Artificial conversational speech," 1993.
19 S. Moon, J. Kurose, and D. Towsley, "Packet Audio Playout Delay Adjustment: Performance Bounds and Algorism," ACM/Springer Multimedia Systems, Vol. 6, No.1, January 1998.
20 Y.H. Chu, S.G. Rao, and H. Zhang, "A case for end system multicast," IEEE J. Sel. Areas Commun., vol.20, pp.1456-1471, Oct.2002.
21 S. Banerjee, B. Bhattacharjee, and C. Kommareddy, "Scalable application layer multicast," Proc. SIGCOMM 2002, pp.205-217, Pittsburgh, PA, Aug. 2002.
22 B. Zhang, S. Jamin, L. Zhang, "Host Multicast: A Framework for Delivering Multicast To End Users," IEEE Infocom'02, June 2002.
23 A technical note and white paper, "Internet Telephony – Service, Technical Challenges and Solutions," http://www.sysmaster.com/brochures/internet_telephony.pdf
24 C. Perkins, "RTP: Audio and Video for the Internet," Addison-Wesley, 2003.
25 Y. Takahashi, H. Yamamoto, H. Yoshino, and A. Toda, "Queuing System Primer: Theory and Practice," IEICE publications, 2003.

figures: 12; tables: 3

Article received: 2005-05-05