インターネットにおけるリアルタイム通信のための冗長パケット挿入アルゴリズムに関する検討

マノーダ ガマゲ　三木 哲也　菅原 真司
電気通信大学 電気通信学部 情報通信工学科
〒182-8585 東京都調布市調布が丘 1-5-1
E-mail: {manodha, miki, sugawara}@ ice.uec.ac.jp

あらまし　本稿では、インターネット上でのリアルタイム通信におけるパケットロスを最小にする目的で、フィードバック制御を用いた冗長パケット挿入アルゴリズムについて提案している。このアルゴリズムでは、ある時点で送信されるパケットのための冗長分データが後に送信されるパケットに添付される方式をとり、その冗長分データは通常のデータよりも低いレートでエンコードされる。しかし、従来のこの方式では冗長度が上がるにつれ、QoS 及び信頼性が向上する反面、有効な帯域を確保できなくなる。提案方式ではネットワークからのフィードバックに基づいて適当な Combination Value に従い、挿入する冗長データブロックを制御する。パケットロス率はアプリケーションによって高い場合（8%以内）と低い場合（4%以内）に制御することを目的としており、提案方式を用いた場合の QoS の改善は、単純な M/M/1/K キューを用いたパケットロスモデルに関して、数学的に、あるいは計算機シミュレーションを行うことによって示されている。

キーワード　冗長データ, プライマリエンコーディング, セカンダリエンコーディング, リアルタイム通信, パケットロス, QoS

Redundant Packet Insertion Algorithm for Real Time Communications over the Internet

Manodha GAMAGE　Tetsuya MIKI　and　Shinji SUGAWARA
Faculty of Engineering, The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585, Japan.
E-mail: {manodha, miki, sugawara}@ ice.uec.ac.jp

Abstract This paper proposes to use UDP and RTP (Over IP) with **Feedback controlled Redundant Packet Insertion Algorithm** to minimize the packet losses in Real-Time communications over the Internet, where the redundant data about present packets are piggybacked along with later packets. The redundant packets are encoded at a lower rate (secondary encoding rate) compared to the primary encoding. With the increase of number of redundancies the QoS and reliability will increase, but it is inefficient with respect to the bandwidth. Therefore the proposed control algorithm inserts redundant blocks based on the feed back from the network by varying the Combination Value (CV) suitably. CV is defined such that number of redundant packets inserted is increased with the CV. The QoS improvement due to the proposal is proved mathematically using simple M/M/1/K queue to model the loss process in the network and also by simulation. This control algorithm is proposed with the aim of controlling packet losses within the limits of **High** and **Low** depending on the application.

Keywords Redundant data, Primary encoding, Secondary encoding, Combination value, Real-Time Communication, Packet losses, QoS

1. Introduction

Initially the Public Switched Telephone Network (PSTN) was designed to deliver a single service: point to point voice communication. Similarly the Internet was introduced with a limited set of services such as email, file transfer, remote terminal emulation etc. Both these networks have evolved over time to provide new user services, but as at present services over IP network are growing very rapidly due to the huge reductions in operating costs compared to PSTN, especially in long distance communications and referred to as Next Generation Communications Networks.

Today the Internet attempts to deliver all traffic as soon as possible within the limits of its ability, but without any guarantees related to throughput, delay, jitter and packet losses. This "best effort" forwarding has worked well for applications running on IP with low priority and low bandwidth data applications with a high tolerance to delay and jitter. It is clear that the field of packet voice is some what matured and the basic building blocks are available. However the real-time (Interactive) multimedia communications such as voice/video conferencing over the Internet is yet to find optimum solutions to overcome the problems such as packet losses, dynamically varying bandwidth, end-to-end delay and jitter.

Using control algorithms to overcome the problems caused by high loss rates, varying bandwidths and delay jitter is very promising for real-time communications over the Internet, in the absence of network support to provide guaranteed QoS. These control algorithms should adapt according to the applications so that they can either eliminate or minimize the impacts of packet losses and delay jitter on the QoS.

2. Currently Available Solutions and Suggestions to Improve Real-Time Transmission over the Internet

Considerable research effort has been directed towards transporting real-time traffic over IP network. The motivation for transporting real time traffic over IP networks is the potential cost savings achievable by eliminating or bypassing the circuit switched telephony infrastructure. In this research we are more focused on voice communication as a real time transmission over the Internet. The end-to-end delay for real-time interactive communications should be less than 250ms to 300ms. The famous Transmission Control Protocol (TCP)[1,4] with Automatic Repeat reQuest (ARQ), which is a closed loop mechanism based on retransmission of packets that were not received correctly at the destination, is not suitable for real time traffic due to the delay restrictions. User Datagram Protocol (UDP)[1,4], which provides faster transmission by sending pieces of information in short bursts with the hope of successful delivery, is used as the layer 4 protocol in real time communications. But it doesn't have any flow or error controls and therefore UDP messages can be lost, delayed or delivered out of order. Real Time Protocol (RTP) is used on top of UDP to support the transport of real time traffic and it provides time stamping, sequence numbering etc.[4,5,6]. Forward Error Correction (FEC), which is an open loop mechanism based on transmission of redundant information is an attractive alternative to ARQ for providing reliability with an acceptable latency and an increase in bandwidth.

The IP networks are inherently best effort networks with variable packet delays and losses. While voice traffic can tolerate some amount of packet loss, a packet loss rate greater than 5%-8% is considered harmful to voice quality. The amount of packet loss rate that can be tolerated depends on the sampling rate and the encoding algorithm used. It also depends on the application. Even though changing IP infrastructure to support guaranteed bandwidth sessions would allow effective transport of voice traffic, it is not an easy proposition compared to applying techniques for compensations for packet delay, jitter and loss.

Various studies in the past have shown that the number of consecutively lost packets is small, except for very busy hours and thus FEC can be used for real time interactive communication over the Internet as a packet loss compensation technique. Algorithms such as Reed-Solomon codes are not much suitable for interactive real-time traffic. This is because these algorithms require the data to be broken down into blocks and the contribution of this 'block delay' to the end-to-end delay is not negligible.

We propose in this paper to use UDP and RTP (Over IP) with **Feedback Controlled Redundant Packet Insertion Technique** (Fig. 1) to minimize packet losses, where the redundant data about present packets are piggybacked along with later packets.

For example N^{th} packet will carry redundant data of $(N - i)^{th}$ packet and if $(N - i)^{th}$ packet is lost, the application waits for the N^{th} packet to recover the lost data with i packets worth of delay.

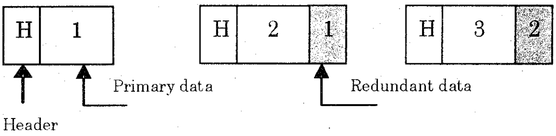

Fig. 1: Redundant Packet Insertion

Implementation of any FEC technique would require additional bandwidth and it would introduce an additional delay. In our algorithm it is proposed to discard any redundant data received after a specified maximum tolerable delay in order to control the end-to-end delay. The redundant packets are encoded at a lower rate (secondary encoding) compared to the primary encoding of the original data in order to control the additional bandwidth required to implement this technique. For example if PCM (64kbps), ADPCM (32kbps) or Low Delay Code Excited Linear Prediction (LDCELP- 16kbps) is used as primary encoding, then Conjugated Structure-Algebraic CELP (CS-ACELP-8kbps), GSM (less than 16kbps), LPC (less than 5kbps) or Harmonic Vector Excitation Coding (HVXC-2/4kbps) can be used as the secondary encoding to generate redundant data[3, 7, 9]. In this paper we propose to use the frame formats in fig. 2, where we use LDCELP as the primary encoding and HVXC as the secondary encoding along with the proposed control algorithm. We considered 40ms voice packets in our frames. LDCELP coding always consider 2.5ms frames and therefore it is necessary to have 16 such frames as primary data to form a packet of 40ms. On the other hand HVXC always consider frames of 20ms, so that it is necessary to have two such frames as secondary data in one packet. With the increase of number of redundancies the reliability will increase, but it will lead to inefficiencies with respect to the bandwidth. Therefore insertion of redundant blocks should be decided based on the feed back the source gets from the network.

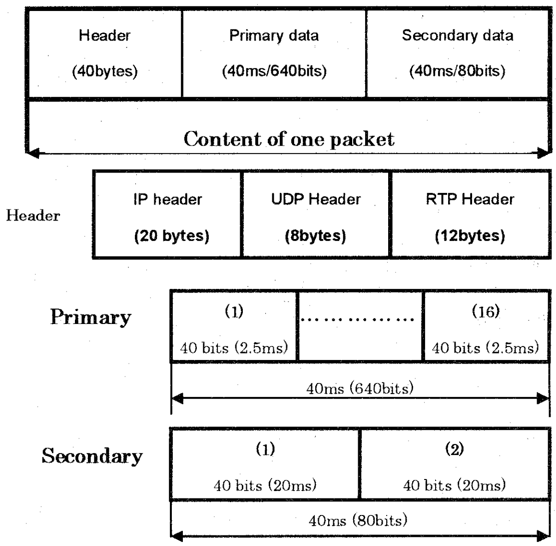

Fig. 2: Frame Formats

3. Proposed Algorithm

Two limits namely **High** and **Low** are defined with the aim of controlling packet losses within these limits. According to the encoding methods and frame formats we have proposed, it can be proved that if we send the redundant data four or more packets after the original data it will not meet the end-to-end delay requirements demanded in interactive real-time communications. Considering this limitation, four Combinations are defined as shown in the Table 1.

Combination Value (CV)	Packets sent
0	0
1	0, 2
2	0, 2, 3
3	0, 1, 2, 3

Table 1: Combinations used in the Control Algorithm

CV= '0' means no secondary data of original data are transmitted and all the sessions are to be started with CV= '0' (Fig. 3 (a)), expecting the network quality to be good and the packet losses are within the tolerable limits. CV= '1' (Fig. 3 (b)) means secondary data of each primary (original) packet is piggybacked in the 2^{nd} packet after the primary packet. Similarly CV= '2' (Fig. 3 (c)) describes the scenario that two secondary data packets of each primary packet are piggybacked in the 2^{nd} and 3^{rd} packets after the primary packet. CV= '3' (Fig. 3 (d)) describes the scenario that three secondary data packets of each primary packet are piggybacked in the next three packets followed by the primary packet.

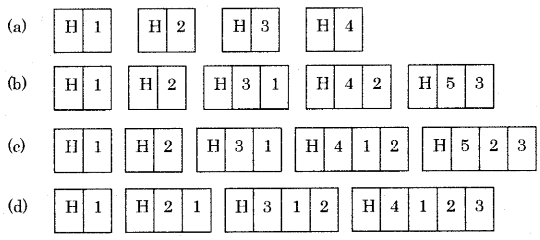

Fig 3: Redundant Packet Insertion based on the Combination value

Feedback information such as number of packets lost before reconstruction (i.e. without considering the redundant information)- L_b, number of packets lost after reconstruction (i.e. considering the redundant information)-L_a, number of event losses with two consecutive packet losses-EL(2), event losses with three consecutive packet losses-EL(3), event losses with four or more consecutive packet losses-EL(4m), total number of packets transmitted(T_N) during the period considered etc. are received periodically via RTCP reports. Using these information, Packet Loss Rate before reconstruction, $LR_b = L_b / T_N$, Packet Loss Rate after reconstruction, $LR_a = L_a / T_N$, Event Loss Rates, ELR (2) = EL (2) / T_N, ELR (3) = EL (3) / T_N, ELR (4m) = EL (4m) /T_N etc. are calculated.

If LR_a>**High** it is necessary to increase the amount of redundancies. If the event losses are high there must be lots of burst losses (i.e. if ELR (4m) is high compared to ELR (2) & ELR (3)) and then there is no much use of increasing redundancies. In fact it will worsen the situation by creating more congestion. Therefore in our algorithm we calculate effective $LR_a = \{L_a - 4\{ELR\ (4m)\}\} / T_N$ (i.e. loss rate after eliminating burst losses) and increase the redundancies only if effective LR_a >**High**. On the other hand if LR_a<**Low** the CV should be reduced to reduce the number of redundancies inserted. Here again this is done only if three consecutive LR_a values are smaller than **Low**. This is done to avoid continuous fluctuations of the quality.

4. A Model for Verification

In a large network as the Internet, a flow of packets crosses several routers before reaching the other end. Most of the losses from a flow occur at the router having the smallest available bandwidth in the chain of routers, so that the whole network can be modeled by a single **bottleneck** router. Simple M/M/1/K queue can be used to model the loss process in the network. [2,8] Assume that audio packets arrive at the bottleneck according to a Poisson distribution of parameter λ and the process time at the bottleneck is exponentially distributed with parameter μ, so that the traffic intensity, $\rho = \lambda / \mu$. At the steady state the traffic intensity, $\rho < 1$.

Under these conditions the loss probability is given by $P(\rho) = \frac{(1-\rho)}{(1-\rho^{K+1})}\rho^{K}$

5. Quality Function

A quality function is defined to analyze the improvements in quality that can be achieved by introducing the proposed feed back controlled redundant insertion technique in real time communications. In defining the quality function it is assumed that the audio quality is proportional to the amount of information received at the receiver. Therefore if primary data is received correctly the quality is assumed to be equal to 1 and if a primary packet is lost and information is recovered from a redundant packet, then the quality should be less than 1 as the information received with secondary encoded redundant data is less compared to the primary data. This quality is assumed to be equal to:

$$\alpha = \frac{\text{Volume of information in a redundant packet}}{\text{Volume of information in an original packet}} < 1$$

The buffer size is assumed to be a function of number of packets and not a function of a packet length. The effects of redundancy insertion to the buffer capacity are assumed to be negligible. Therefore $K_a = \frac{K}{1+i\alpha} \approx K$, Where K and K_n are the buffer sizes with and without redundancy insertion respectively. Then the loss probability with the insertion of redundancies will be,

$$P(\alpha) = \frac{1-\rho}{1-\rho^{K_a+1}}\rho^{K_a} \approx P(\rho)$$

Define a continuous two state Markov variable $\{X_n\}$, where $X_n \in (1, 0)$ such that

-If packet n is lost $X_n = 1$

-If packet n is correctly received $X_n = 0$

The quality function depends on the Combination Value (CV) applied.

For CV = '0'

$$Q(\alpha) = P(X_n = 0) = 1 - P(\rho) \approx 1 - P(\alpha)$$

For CV = '1'

$$\begin{aligned} Q(\alpha) &= P(X_n = 0) + \alpha P(X_n = 1) P(X_{n+2} = 0 \mid X_n = 1) \\ &= 1 - P(\alpha) + \alpha P(\alpha) P(X_{n+2} = 0 \mid X_n = 1) \\ &= 1 - P(\alpha)[1 - \alpha P(X_{n+2} = 0 \mid X_n = 1)] \end{aligned}$$

Therefore $Q(\alpha)_{CV=0} < Q(\alpha)_{CV=1}$

Similarly it can be shown that Quality Function will improve when the combination value is increased.

6. Analysis of the Simulations

We simulated a 10Mbps channel for long durations by randomly varying the amount of traffic and the number of users to verify the validity of our theoretical findings. These simulations were run continuously for over 24 hours so that it will verify the algorithm both during peak and off peak times. We set the values of **High** and **Low** of the algorithm to be 8% and 4% respectively in all our simulations. Figure 4 shows the results of the simulation with our proposed redundant packet insertion algorithm applied and the average packet loss rate in this case is 6.29%. Also it is very clear from the graph that we can have a steady packet loss rate by applying the proposed dynamically adaptive algorithm. In other words it is possible to achieve a steady improved QoS irrespective of the variations in the traffic conditions of the network. On the other hand figure 5 shows the results when we simulated the channel under the same conditions, but without the proposed algorithm. Here the average packet loss rate is 11.37%. Besides the high average packet loss rate we see rapid fluctuations of the packet loss rate, with the traffic variations of the network over time. Comparing figure 4 and 5, we can conclude that there is a 50% improvement in the average packet loss if the proposed algorithm is implemented.

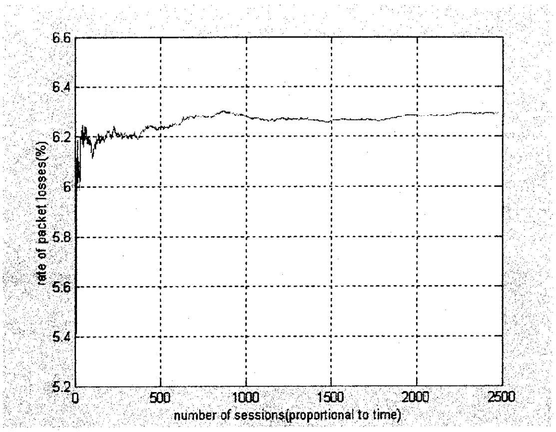

Fig 4: Packet loss rate with the proposed algorithm

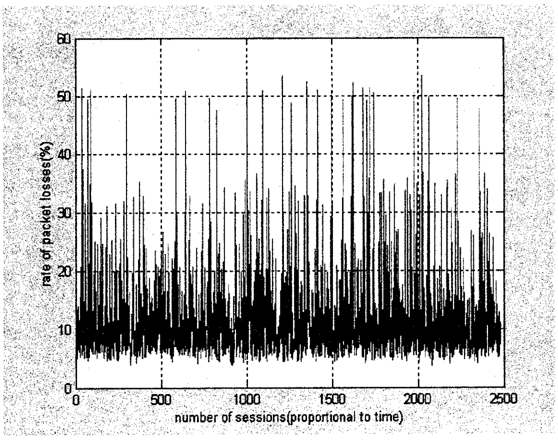

Fig 5: Packet loss rate without the proposed algorithm

We also simulated the proposed algorithm without applying the '**Low**' limit and the results are shown in figure 6. In this scenario if there are more packet losses in the network the 'Combination Value' is increased to insert more redundant information, but if packet losses in the network is very low the redundancy insertion is not reduced. This will achieve very low packet loss rates, but the usage of the bandwidth is not very efficient as we keep on inserting redundant information unnecessarily even when the loss rate in the network is very low.

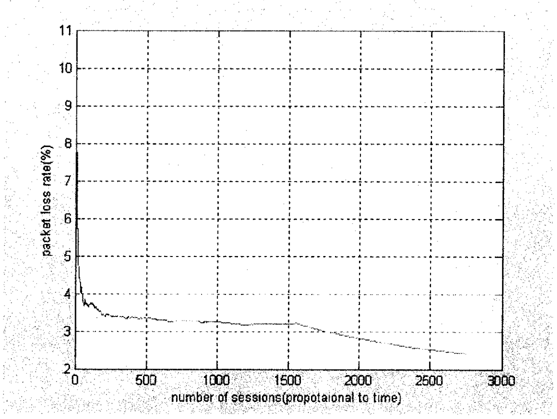

Fig 6: Packet loss rate with the proposed algorithm applied without '**Low**' limit

7. Conclusions

Various FEC techniques for real-time multimedia applications over the Internet have been proposed recently. However we need to analyze them very carefully as FEC will increase the bandwidth requirements and the end-to-end delays. Network congestion is one of the main causes for packet losses. Increased bandwidth requirements due to the application of FEC may result in more congestion leading to more packet losses. This is one of the major factors that should be concerned when deciding a FEC as a compensation technique for packet losses. Further more these conditions in the network vary dynamically.

We have proposed an adaptive algorithm to control packet losses based on the feed back from the network. This will of course, not provide a guaranteed QoS for real-time communications over the Internet, but it will definitely provide a noticeable improvement in the quality even at very highly varying network loss rates. In this algorithm the speed of adaptation depends on the feed back and therefore it is very important to receive timely and accurate reports on the status of the network.

The best improvements for QoS of real-time traffic can be achieved by minimizing packet losses, end-to-end delay and jitter and by varying the transmission rate according to the bandwidth variations. Therefore the optimum results of this proposed algorithm can be achieved by coupling it with an algorithm to control the transmission rate according to the varying bandwidth. Also it will give better results in high speed networks such as MPLS, where the delay jitter is less.

We will pursue this study to improve this algorithm to be used in the multicast environment as well. In the future, we wish to couple this algorithm with an algorithm to control the transmission rate according to the bandwidth variations. Also it is necessary to research more on the hardware requirements, both at the sender and receiver end, in order to implement the proposed algorithm.

References

[1] Douglas E Comer "Internet working with TCP/IP" - volume 1, 3rd edition published by Prentice Hall - New Jersey in 1995.

[2] Leonard Kleinrock "Queuing Systems" - Volume 1: Theory, published by John Wiley & Sons - New York.

[3] Thomas Wedi, "Hybrid Video Coding Based on High-Resolution Displacement Vectors", "Adaptive Interpolation Filter for Motion Compensated Hybrid Video Coding", 110th Convention of Audio Engineering Society May 2001.

[4] Rakesh Arora "Voice over IP: protocols and standards", www.cis.ohio-state.edu/~jain/cis788-99/voip_protocols/index.html

[5] H. Schulzrinne, S. Casner, R. Federick, V. Jacobson, "RTP: A transport protocol for real-time applications", RFC 1889.

[6] H. Schulzrinne, "RTP Profile for Audio and Video Conferences with Minimal Control", RFC 1890, Internet Engineering Task Force Jan. 1996

[7] R.V. Cox and P. Kroon, "Low Bit-rate Speech Coders for Multimedia Communication", IEEE Communications Magazine, pages 34-41, Dec. 1996.

[8] Jean-Chrysostome Bolot, "Characterizing End-to-End Packet Delay and Loss in the Internet", High-Speed Networks, vol. 2, no. 3, pp. 305-323, December 1993.

[9] A white paper on "Speech Codecs", Wipro Technologies, http://www.wipro.com/prodesign/focusareas/multimedia/knowledge.htm