Multi-path Mechanism for Audio / Video Streaming Based on Bandwidth Estimation

Joonsuk Kang † and Koji Okamura ††,

†Department of Computer Science and Communication Engineering, Graduate Student
††Research Institute for Information Technology

Summary
By the remarkable improvement of network performance which consist of Internet and the diffusion of high-end terminals such as smart-phones, media streaming service is becoming one of the major communication technologies. However current packet networks still can not provide a stable transmission quality and it can cause stream quality degradation. Especially, the main factor affecting quality is congestion across a network. Efficient utilization of available bandwidth over multiple access networks of multi-homed devices can be a reasonable solution to provide good quality for real-time media streaming applications. In this paper, we propose Real Time & Multi-path Transmission Protocol (RTMTP), a transport layer protocol in which multiple path real-time transport is available. Our protocol exploits RTP's real-time features and SCTP's multi-homing capability by enabling the use of multiple paths to transfer media streams. RTMTP uses the end-to-end sender-based bandwidth estimation mechanism to measure the actual available bandwidth of each path. To maximize network bandwidth efficiency of multiple paths, the RTMTP shares traffic among those paths and the transmission rate for each path is determined according to the measured available bandwidth. RTMTP also helps users to use their preferred access network as much as possible. Our simulation results show that RTMTP improves streaming transmission throughput

Key words:
Real-time streaming, multi-path delivery, bandwidth-estimation, SCTP, RTP/RTCP

1. Introduction

With improved terminal performance and demand of users for more sophisticated services, it has even become possible that one terminal supports several interfaces, that is, one terminal can connect to a number of access networks simultaneously. The demand and necessity for the multiple paths use of an end-terminal are increasing, as it can give benefits of advanced services and efficient management of network resources and traffic. Especially, using multiple paths can help to guarantee quality of service (QoS) for media streaming applications with high data rate.

The quality of media streaming applications is affected by network performance such as latency, jitter, or packet loss.[1-7] Especially, network congestion causes high packet loss rate. The packet loss is bursty in nature and bursty packet loss has a severe impact on streaming quality. In order to guarantee QoS, congestion should be under control. To avoid network congestion, sender needs to control sending rate. However, Real-Time Transport Protocol (RTP)/ RTP Control Protocol (RTCP) [8] which is widely used for real-time streaming applications can not support an effective method to control it.

For the purpose of providing a reliable end-to-end message transportation service, the Stream Control Transmission Protocol (SCTP)[9] and its various extended versions[10-15] were proposed. They support TCP-like congestion control and well-designed multi-homing solution. However, for streaming media content, the reliability could be rather the disturbance for the guarantee of a desired quality.

We therefore propose a new protocol named Real Time & Multi-path Transmission Protocol (RTMTP), a transport layer protocol that is capable of supporting multiple path real-time transport and effective congestion control. RTMTP data transfer is designed to be suitable for transporting data with real-time character. It does not support reliable transmission i.e. there is no congestion window or slow start phase.

RTMTP supports multi-homing by exploiting SCTP's multi-homing features. Especially, it can utilize multiple paths simultaneously to transfer data. This enhances the throughput by bandwidth aggregation over multiple paths.

The primary goal of our protocol is sharing outgoing streams among multiple paths efficiently, with congestion avoided. For this, RTMTP continuously monitors the status of each path such as round trip time (RTT), available bandwidth (ABW) etc.

In order to maximize the use of the user preferred path, each path has a user preference value set up by a user. A higher priority path is fully utilized preferentially to transmit traffic.

An important aspect of RTMTP is that it is completely end-to-end.

The rest of this paper is organized as follows. In section 2, the related works are introduced briefly. In section 3, we describe in detail the proposed Real Time & Multi-path Transmission Protocol. Section 4 presents the performance

Manuscript received February 5, 2013
Manuscript revised February 20, 2013

evaluation with a simulator. Finally, in section 5, we make some conclusions.

2. Related Works

2.1 RTP/RTCP

The Real-Time Transport Protocol (RTP)[8] is an Internet protocol standard that provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video or simulation data, over multicast or unicast network services. RTP is regarded as the primary standard for audio/video transport in IP networks.

RTP runs on the top of the User Datagram Protocol (UDP). It does not in itself address resource reservation and does not guarantee quality of service (QoS) for real-time services. (since it is dependent on network characteristics) RTP is used in conjunction with the RTP Control Protocol (RTCP)[8] to monitor data delivery statistics and QoS.

RTP provides facility for jitter compensation and detection of out of sequence arrival in data. However, RTP/RTCP can not quickly respond to dynamically changing networks.

2.2 SCTP

The Stream Control Transmission Protocol (SCTP)[9] is a transport layer protocol that provides a connection oriented, full duplex, reliable data communication path and in-sequence transport of messages with congestion control (TCP like). It was defined by the IETF Signaling Transport (SIGTRAN) working group.

SCTP serves in a similar role to the Transmission Control Protocol (TCP). However, unlike TCP, SCTP provides message-based multi-streaming.

SCTP supports a good solution for multi-homing. During an SCTP association initialization, the two end-hosts exchange their multiple IP addresses. However, the SCTP multi-homing supports only communication reliability. Only one primary path is used for data unit transmission. Secondary paths are used to deal with all kinds of retransmissions or as a backup path. It is not used for load balancing.

In order to enhance the multi-homing ability of the original SCTP, extended versions of SCTP such as LS-SCTP (Lord-Sharing SCTP), cmpSCTP (concurrent multi-path SCTP) and CMT-SCTP (Concurrent Multi-path Transfer - SCTP) were proposed [10 - 14]. These protocols are designed to use all the available paths for the association at the same time, instead of using only the primary path. Therefore it can support full multi-homing by utilizing all available paths simultaneously, which achieves effective load-balancing and increases an application’s throughput.

To support the flexibility to provide intermediate reliability levels, the Partial-Reliable SCTP (PR-SCTP) [15] was proposed. PR-SCTP is an extension to SCTP for partial reliability that enables a content sensitive transport service where the reliability of messages can be individually controlled. It allows an SCTP sender to assign different levels of reliability.

2.3 Bandwidth Estimation

An accurate estimation of available bandwidth helps network applications to adjust their behavior accordingly. Numerous end-to-end available bandwidth measurement techniques have been developed. These methods inject probe packets into the network and collect the feedback information including transmission delay, packet arrival-interval time and so on.

[16 – 20] use the packet-pair rate probing technique to estimate the bottleneck available bandwidth by measuring the inter-arrival time between back-to-back packets

[21, 22] use periodic-stream techniques. The bandwidth estimation is derived by monitoring one-way delay variations of an equal-size packets train, transmitted at a constant bit-rate. Because of its adaptive search, this technique may have long convergence times and uses a large number of probe packets per estimate.

[23] uses the dispersion of longer packet bursts (packet trains) to estimate the available bandwidth of a path.

3. Real Time & Multi-path Transmission Protocol

Our main goal is to design a new multi-path real-time transport protocol, and it has to satisfy the following features:

- Suitable for the delivery of time-sensitive streaming
- Support effective congestion control mechanism
- Maximize the effect of path diversity
- Considering user’s path preference

In this section, we introduce RTMTP in which multi-path real-time transport and effective congestion control are available.

3.1 RTMTP Design

The RTMTP is a transport layer protocol, serving in a similar role to the cmpSCTP and RTP/RTCP.

RTMTP borrows a multi-homing concept from cmpSCTP and a data transmission feature from RTP, and adds a bandwidth estimation based congestion control concept.

RTMTP doesn't support end-to-end flow control and reliability, but congestion control is performed per each path. Especially, congestion avoidance is achieved by traffic shifting mechanism. RTMTP also applies play-out buffer mechanism to minimize the unexpected effect due to transmitting traffic through multiple paths which have different attributes.

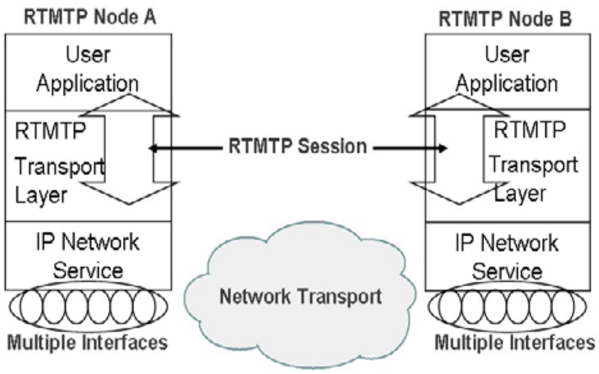

Fig. 1 RTMTP protocol stack.

Figure 1. shows the protocol stack of RTMTP.

RTMTP is between Application layer and Internet layer. Each RTMTP node can utilize more than one IP address in a RTMTP session and uses multiple paths to deliver media streams. The RTMTP connection between two RTMTP nodes is called a "RTMTP session" and RTMTP does not allow a half-open connection. A 4-way handshake is used for establishing a RTMTP session. (We borrowed this concept from SCTP)

3.2 Architectural Overview

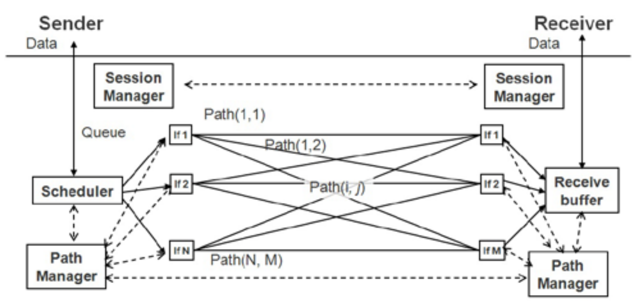

Fig. 2 RTMTP architecture.

Figure 2 provides an architecture model of RTMTP.

The session manager is responsible for managing session information such as paths list. During initiation of the RTMTP session, two end-hosts exchange active interfaces lists consisting of their priority (see section 3.3) information and make out all available paths list. Session information can be dynamically negotiated between two end-hosts during communication. A single port number is used across the entire address list at an endpoint for a specific session.

After session initiation, media traffic is transferred to the peer host through paths.

The scheduler is responsible for allocating RTMTP data packets to paths. In the non-congested case, the scheduler tries to maximize the use of a more preferred path. The scheduler adopts a heap up buffer to distribute the data packets via available paths. ([24])

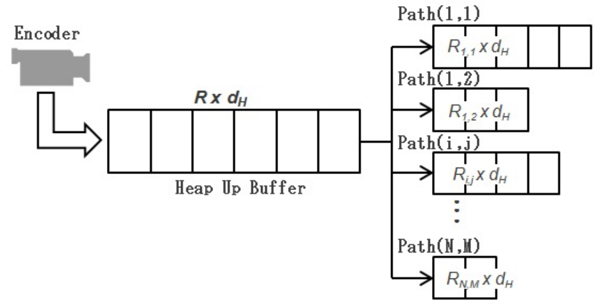

Fig. 3 Data packet allocation.

The buffer heaps up encoded data packets (data rate R). In every d_H time the stored data packets are distributed between the paths and the heap up procedure starts again.

The heap up buffer is connected to the paths through path buffers, which are modeled as FIFO queues. These queues are refilled at every d_H. The path buffers' sizes are determined by the allocated data rate ($R_{i,j}$) of each path and the heap up buffer delay (d_H). The path buffers must be purged in d_H time to be able to receive the next amount of data.

The receive buffers minimize the effect of out-of-order packet deliveries. Due to the different path characteristics such as RTT(Round Trip Time), data packets can arrive out-of-order. It intensifies packet loss, because RTMTP doesn't support session flow control. The role of the receiver buffer is to adjust delay difference among the paths. If the play-out buffer size is enough, the receive buffer is not necessary. (The play-out buffer is used at the receiver-end to compensate for variable network delays (jitter) and to maintain packet order.)

The Path manager checks each path continuously. (See detailed explanation in section 3.5)

3.3 User Preferred Path

RTMTP can maximize the use of user preferred interfaces. In case of using multiple-interfaces, the mobile users may choose their preferred paths to send data traffic probably based on the price paid and on the quality of the service offered by various service providers. For example, assuming that Joon can connect to the Internet via 3G network and WLAN. When WLAN is available, he would prefer accessing the Internet service through WLAN.

In order to reflect user's demands, each interface has a priority in RTMTP. User can set up the priority of each interface.

During session initialization, all the paths priorities are decided according to the interface priority.

RTMTP tries to allocate data traffic into more preferred path.

3.4 Packet Format

RTMTP packet is uses two types of packet, data packet and control packet.

Figure 4. shows RTMTP packet structure.

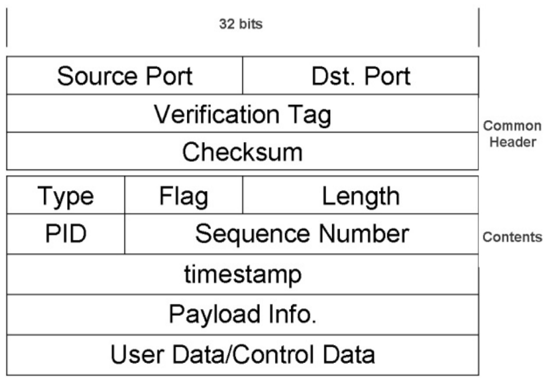

Fig. 4 RTMTP packet structure.

An RTMTP packet is composed of a common header and contents. Contents contain data or control information. RTMTP packet contains a common 12-byte header containing source/destination port numbers, the verification tag and checksum. The tag is a session identifier. Data content contains a user data. Path ID (PID) indicates transmission path. Each path in a session should have a unique ID. The sequence number increases by one for each RTMTP data packet sent, and is be used by the receiver to detect packet loss and to restore packet sequence. The timestamp reflects the sampling instant of the first octet in the RTMTP packet by which RTT and jitter can be estimated. The role of Payload information is same as RTP's payload type (PT).

The following control packet types are defined in RTMTP

- Session Initiation / Session Initiation ACK :
- Heartbeat / Heartbeat ACK
- ABW(Available Bandwidth) Request_DATA
  / ABW Request_DUMMY
  / ABW Request ACK
- Delay Difference Checker
- Session Update Request
  / Session Update Request ACK
- Path Status Request / Path Status Request ACK
- Path Status Inform / Path Status Inform ACK
- Path Re-assignment Request
  / Path Re-assignment Request ACK
- Session Shutdown
  / Session Shutdown ACK
  / Session Shutdown Complete

3.5 Path Monitoring

Because RTMTP dynamically utilizes multi-paths to transfer streams, path attributes should be monitored continuously. 'Path monitor' is a component to check path status.

The path manager monitors the actual available bandwidth and Round Trip Time (RTT) of each path and delay differences among all active paths.

3.5.1 Bandwidth Estimation for Path

The bandwidth of each path is estimated independently.

We adopt packet-pair rate probing technique to estimate the bottleneck available bandwidth by measuring the inter-arrival time between back-to-back packets [16].

If a receiver receives an ABW Request packet, it should immediately send an ABW Request ACK packet along the path. The ack packet must consist of the original information of the received ABW Request packet (such as PID, Sequence Number, timestamp and packet size). Usually, normal data packets can be used as ABW Request packets. To estimate bandwidth, a number of continuous ABW Request packets need to be sent periodically for every path.

A sender uses the value of the RTT and the amount of data confirmed by the received ACKs to estimate the available bandwidth. If, in the generic kth RTT, Δ_k, one or more ACKs notify that a total of d_k bytes have been received, the bandwidth sample, B_k, can be computed as

$$B_k = d_k / \Delta_k \qquad (1)$$

In order to average the sampled measurements and filter out the high-frequency components, a low pass filtering is necessary. RTMTP adopt the discrete approximation already used in Westwood+ TCP [25] so that the filtered bandwidth $\hat{B}_k$ can be written as

$$\hat{B}_k = \alpha\hat{B}_{k-1} + (1 - \alpha) B_k, \qquad (2)$$

The parameter α is a weighting factor that determines how much the two most recent samples should be weighed against the history of the bandwidth estimate.

A number of papers [25 - 27] have shown that this technique provides a reliable estimate.

If there are no allocated data packets on a path, a number of continuous probing packets (ABW Request_DUMMY packet) might be generated and sent on the path to measure the available bandwidth.

3.5.2 Delay Differences

To measure delay differences among all active paths, a Delay Difference Checker control packet is utilized. The Delay Difference Checker control packet includes timestamp and PID. A sender sends Delay Difference Checker control packets to all paths simultaneously. The receiver can calculate the delay differences among all paths by checking the timestamp information of all received Delay Difference Checker control packets.

The acceptable delay depends on the used multimedia application. A slow path should be avoided for very time-sensitive applications.

3.6 Rate Scheduling

Considering a set of active paths Pth = {1, 2, 3,…,P}, each path p is characterized by its available bandwidth ABW_p , path priority Pr_p and the allocated rate R_p.

The RTMTP distributes the data rate of the stream R among the active paths. Rate is preferentially allocated to the path having higher priority and the available bandwidth of the path is fully utilized. To avoid network congestion, the allocated rate R_p to a path p can not exceed the available bandwidth. (i.e. $R_p \leq ABW_p$, $R \geq= \sum_p R_p$)

4. Performance Evaluations of RTMTP

We implemented RTMTP in NS-2.31[28] to analyze the performance of the proposed architecture.

Figure 5 shows the network topology used for simulation. We implemented two network models with RTP and RTMTP respectively for a performance comparison. We tested our implementation under a variety of traffic mixes. To investigate the impact of network congestion, we generate several VoIP traffic (G.711, GSM-AMR) and UDP traffic and concentrate it into the congested link. Especially, to simulate realistic voice traffic, we adopted a talkspurt-silence model for VoIP traffic.

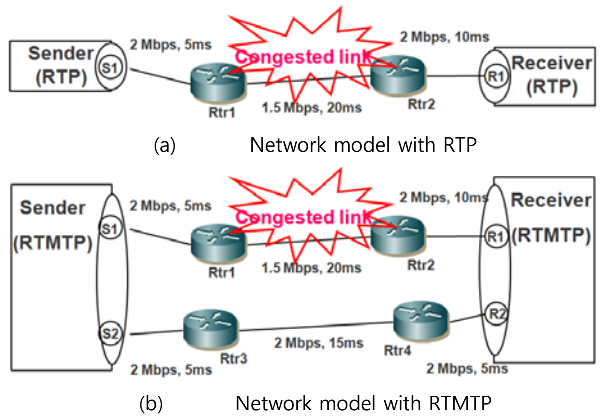

Fig. 5 Network model for simulation.

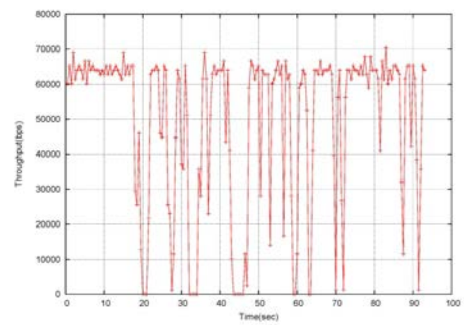

Fig. 6 G.711 VoIP flow generated by the sender.

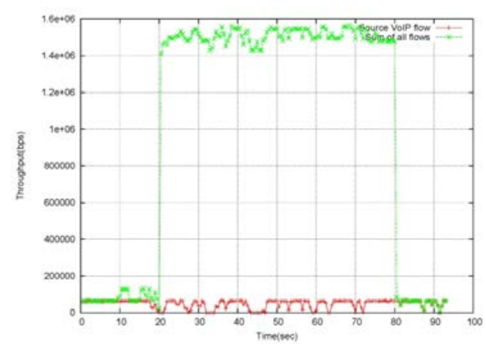

Fig. 7 Traffic flowed into the link between Rtr1 and Rtr2.

We assumed that voice data packets only are sent from the sender to the receiver and smaller numbered interface has higher priority. The sender generates VoIP traffic using the G.711 codec which has a fixed bit rate of 64 Kbps. The payload size is 80 bytes.

It is assumed that the path (S1, R1) is preferred than the path (S2, R2) and data traffic tends to be assigned to the path (S1, R1).

Figure 6 shows the voice flow generated by the sender.

Figure 7 shows all the traffic flowed into the link between Rtr1 and Rtr2.

4.1 RTP Performance Evaluation

In case of RTP, after about 20 seconds, Rtr1-Rtr2 link starts to drop packets. The red solid line in Fig. 8 shows the dropped VoIP traffic during the congested period. The green dotted line shows the traffic arrived at the receiver without being dropped. Total packet loss ratio is depicted in Fig.9. Congested link causes packet loss and it affects VoIP call quality.

We calculated "R" value as the measure of voice quality. (Fig.10) During the congested period, voice quality is severely degraded.

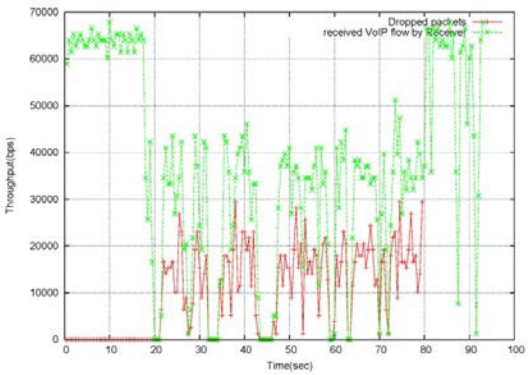

Fig. 8 Dropped VoIP traffic.

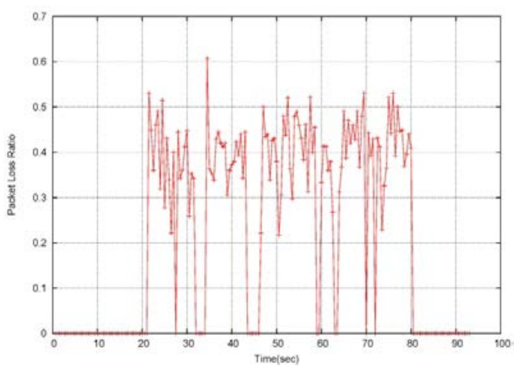

Fig. 9 Packet loss ratio.

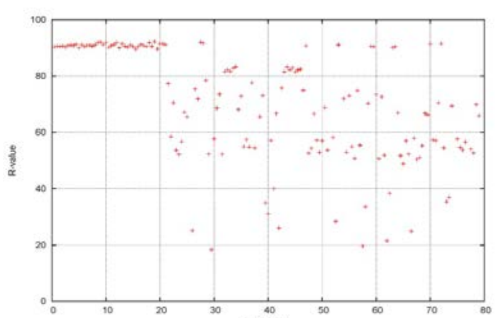

Fig. 10 R-value(RTP).

4.2 RTMTP Performance Evaluation

Now, we show the RTMTP performance evaluation result. The traffic patterns used for the RTMTP system are same as the RTP's

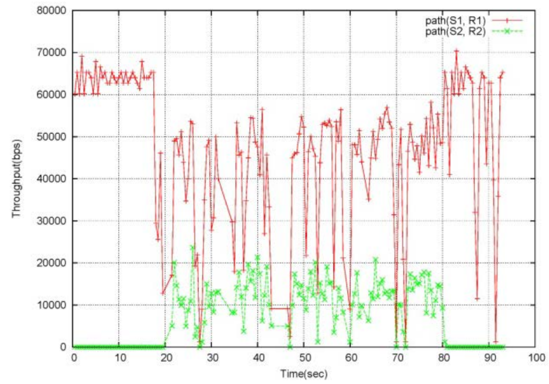

Fig. 11 Trace of rate over paths.

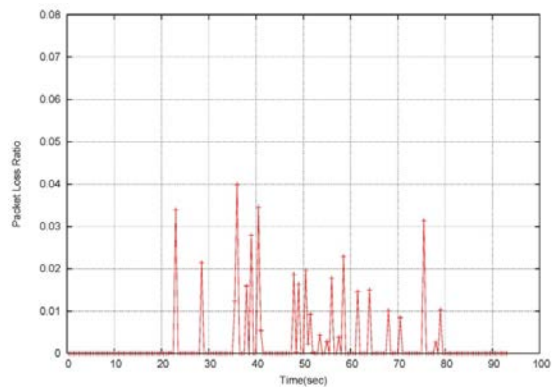

Fig. 12 Total packet loss ratio.

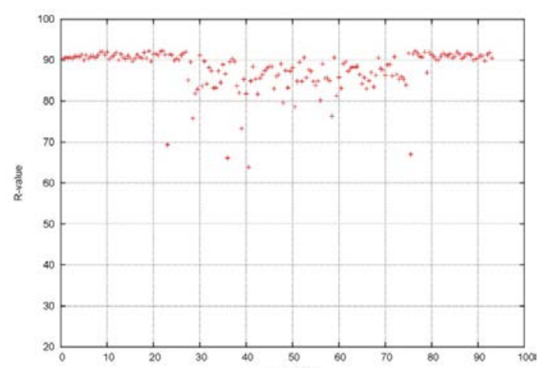

Fig. 13 R-value(RTMTP).

The RTMTP scheduler controls the transmission rate of each path according to the estimated bandwidth.

In Fig.11, the red solid line and the green dotted line indicate the traffic flowed into the path (S1, R1) and the path (S2, R2) respectively. Figure.12 shows the total packet loss ratio detected by the playout buffer of receiver. It is caused by delay jitter and packet reordering rather than network congestion. In Fig.13, we also calculated voice call quality of RTMTP system. As compared to the RTP case, RTMTP assures reasonable voice call quality by avoiding network congestion.

5. Conclusions and Future Works

In this paper we suggested a new protocol named Real Time & Multi-path Transmission Protocol (RTMTP) that is capable of supporting multiple path real-time transport and effective congestion control.

We optimized the data transmission scheme of RTMTP for real time transmission. Furthermore, in order to guarantee service quality, RTMTP supports multi-homing and path diversity utilizing.

RTMTP uses the end-to-end sender-based bandwidth estimation mechanism to measure the actual available bandwidth of each path. This enhances the throughput by bandwidth aggregation over multiple paths.

To enhance the usage of user preferred paths, a simple priority policy is supported. For an effective management of network resources, the status of every path is monitored periodically such as RTT and available bandwidth. RTMTP uses the end-to-end sender-based bandwidth estimation mechanism to measure the actual available bandwidth of each path.

To analyze the performance of the proposed architecture, we implemented RTMTP in NS-2. The result showed that RTMTP is suitable for real-time streaming applications and achieves the good performance for data transmission.

References

[1] J. M. Boyce et al, “Packet loss effects on MPEG video sent over the public Internet”, ACM Multimedia, Bristol, England, September 1998.

[2] D. Loguinov et al, “Measurement study of low-bitrate Internet video streaming” ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA, November 2001.

[3] D. Loguinov et al, “End-to-end Internet video traffic dynamics: Statistical study and analysis”, IEEE INFOCOM, New York, NY, June 2002.

[4] A. Basso et al, “Transmission of MPEG-2 streams over non-guaranteed quality of service networks”, Picture Coding Symposium, Berlin, Germany, September 1997.

[5] Y. Wang et al, “An empirical study of Real Video performance across the Internet”, ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA, USA, November 2001.

[6] A. C. Dalal et al, “A new architecture for measuring and assessing streaming media quality”, the 3rd Workshop on Passive and Active Measurements (PAM 2003), San Diego, CA, April 2003. 14

[7] J. van der Merwe et al, “A tool for monitoring Internet multimedia traffic”, ACM Computer Communication Review, 30(4), October 2000.

[8] H. Schulzrinne et al, "RTP: A Transport Protocol for Real-Time Applications", RFC 3550, July 2003.

[9] R. Stewart et al, "Stream Control Transmission Protocol", RFC 4960, September 2007.

[10] Ahmed Abd El Al et al, "LS-SCTP: a bandwidth aggregation technique for stream control transmission protocol", Computer Communications, 27(10):1012–1024, 2004.

[11] J Iyengar et al, "Concurrent multipath transfer using SCTP multihoming", SPECTS 2004, 2004.

[12] J. Iyengar et al, “Concurrent Multipath Transfer using SCTP Multihoming over Independent End-to-end Paths,” IEEE/ACM Transaction on Networking, Oct. 2006, 14(5): 951-964.

[13] Jianxin Liao et al, "cmpSCTP: An Extension of SCTP to Support Concurrent Multi-path Transfer", Communications, 2008. ICC '08. IEEE International Conference on, pp.5762-5766, May 2008.

[14] P. Natarajan et al, “CMT using SCTP Multihoming: Transmission Policies using a Potentially-failed Destination State,” TR 2007/338, CIS Dept, University of Delaware, Feb. 2007.

[15] R. Stewart et al, “Stream Control Transmission Protocol (SCTP) Partial Reliability Extension,” IETF RFC 3758, May 2004.

[16] Keshav et al, “A Control-theoretic Approach to Flow Control”, ACM Sigcomm 1991, (Zurich, Switzerland, September 1991), 3-15.

[17] K. Lai et al, “Measuring Bandwidth”, INFOCOM, Apr. 1999.

[18] R.L. Carter et al, “Measuring Bottleneck Link Speed in Packet-Switched Networks”, Technical report, 1996.

[19] R. L. Carter et al, “Measuring bottleneck link speed in packet-switched networks”, Tech. Rep. TR-96-006, Boston University Computer Science Department, Mar. 1999.

[20] B. Melander et al, “A new end-to-end probing and analysis method for estimating bandwidth bottlenecks”, IEEE GLOBECOM 2000, Nov. 2000.

[21] M. Jain et al, “End-to-End Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput”, IEEE/ACM Trans. Networking, vol. 11, no. 4, Aug. 2003.

[22] V. Ribeiro et al., “pathChirp: Efficient Available Bandwidth Estimation for Network Paths”, Proc. Passive and Active Measurement Workshop, Apr. 2003.

[23] J.C. Bolot, “End-to-End Packet Delay and Loss Behaviour in the Internet”, ACM SIGCOMM Computer Comm. Rev., vol. 23, no. 4, Oct. 1993.

[24] Huszák, Á. et al, “Content-aware Interface Selection Method for Multi-Path Video Streaming in Best-effort Networks”, 16th International Conference on Telecommunications, Marrakech, Morocco, Jul. 2009, pp. 196-201.

[25] L.A. Grieco et al, “Performance Evaluation and Comparison of Westwood+, New Reno, and Vegas TCP Congestion Control”, ACM SIGCOMM Computer Comm. Rev., vol. 34, no. 2, Apr. 2004.

[26] L.A. Grieco et al, "End-to-End Bandwidth Estimation for Congestion Control in Packet Networks", Proc. IEEE Int'l Workshop Multiservice IP Networks (QoS-IP '03), Feb. 2003
[27] R. Fracchia, et al, "WiSE: Best-path selection in wireless multihoming environments", IEEE Transactions on Mobile Computing, Vol. 6, No. 10,pp. 1130-1141, October 2007.
[28] The network simulator : ns-2, http://www.isi.edu/nsnam/ns/.

JoonSuk Kang was received the B.S. and M.S. degrees in Information and Communications Engineering from Chun-gnam National University, Korea in 2000 and 2002, respectively. He is now studying as Phd student in the department of Advanced Information Technology, Graduate school of Information Science and Communication Engineering, Kyushu University.

Koji Okamura received the B.S., M.S., and Ph.D. degrees from Kyushu University in 1988, 1990, and 1998. He is currently a professor at Research Institute for Information Technology of Kyushu University and a vice director of Research Institute for Information Technology of Kyushu University. His research interests are 'The Future Internet Technologies' and 'Power Aware Network Management'.