A New Smooth Handoff Scheme for Mobile Multimedia Streaming using RTP Dummy Packets and RTCP Explicit Handoff Notification

Kenichi Kashibuchi†, Tarik Taleb‡, Abbas Jamalipour*, Yoshiaki Nemoto‡, and Nei Kato†
†,‡Graduate School of Information Sciences, Tohoku University, Sendai, Japan
*School of Electrical and Information Engineering, University of Sydney, Australia
Emails: †{buchiken, kato}@it.ecei.tohoku.ac.jp, ‡{taleb, nemoto}@nemoto.ecei.tohoku.ac.jp, *a.jamalipour@ieee.org

***Abstract*— In the near future, RTP/RTCP-based multimedia streaming will become the norm not only in wired networks but also in mobile environments. Presently when a handoff occurs between heterogeneous networks (with different available bandwidths), a RTP sender cannot stream media at a suitable rate over the new network. Furthermore, RTCP fails to precisely adapt to sudden changes in network resources due to handoff.**

In order to solve these issues, 1) senders should be aware when mobile nodes are about to perform a handoff; 2) senders should then efficiently probe the available bandwidth in the new network and accordingly adjust their streaming rates. In this paper, we propose a scheme that allows mobile nodes to explicitly notify their handoff timing by using newly-defined RTCP packets. In the proposed scheme, senders probe the available bandwidth in the new network using low-priority RTP dummy packets.

The performance of the proposed scheme is evaluated and compared with conventional schemes through extensive simulations. The simulation results show that the proposed scheme achieves appropriate bandwidth utilization immediately after a handoff occurrence and lowers packet losses during the handoff. The proposed scheme exhibits also high TCP-friendliness.

I. Introduction

Along with the growth in the users community of high-speed and broadband Internet access technologies, such as FTTH (Fiber To The Home) and xDSL (Digital Subscribers Line), multimedia streaming services are becoming popular. Although TCP (Transmission Control Protocol) is the dominant communication protocol in today's Internet traffic, real-time streaming services are mostly based on the RTP (Real-time Transport Protocol) [1]. RTP is mainly used over UDP (User Datagram Protocol). Whilst RTP does not provide any reliability or congestion control mechanisms, it has the advantage of not introducing additional delays to the transmitted data due to retransmissions, as is the case with TCP. However, when non-congestion controlled RTP is deployed in the Internet, RTP senders transmit packets regardless of the network conditions. This may congest the network. Therefore, RTP is generally used together with RTCP (RTP Control Protocol) [1].

In RTP/RTCP, receivers notify senders of some information related to their perceived Quality of Service (QoS), such as packet losses and jitter. This information is sent in packets called Receiver Report (RR). Then, RTP senders assess the network conditions and control their streaming rates based on RTCP RR [2].

On the other hand, current wireless accesses such as wireless LAN (WLAN) and WiMAX (Worldwide Interoperability for Microwave Access) enable high-speed and broadband communications. Consequently, it will be soon possible to enjoy multimedia streaming services using mobile devices such as laptop computers, PDA (Personal Digital Assistance), and mobile phones. However, mobile networks exhibit different characteristics from their wired counterparts. In deed, when providing multimedia services using RTP/RTCP to mobile users, the following problems may occur. First, when a mobile node performs handoff (handover) between two different base stations (access points), if the two cells have different available bandwidth, the sender cannot stream at a suitable rate for the new cell immediately after the handoff. This bandwidth disparity can be due to difference in traffic load in both wireless cells or use of different wireless access techniques with different link speeds, such as WLAN, 3G (3^{rd} generation cellular system), and WiMAX. When a mobile node performs handoff, two scenarios can be envisioned. If the mobile node moves from a higher bandwidth network (*e.g.* WLAN) to a lower bandwidth network (*e.g.* 3G), and the server continues transmitting data without any adjustments to its streaming rate, the new network may be congested and a number of packets may be dropped. On the other hand, if the mobile node moves from a lower bandwidth network to a higher bandwidth network, no adjustment to the streaming rate will lead to a waste of the network bandwidth and degraded service quality in an environment where higher streaming rates (higher quality) can be obtained. Second, since RR packets are not generated so frequently, the streaming rate cannot adapt to sudden changes in network resources due to handoff. In [1], the fraction of the session bandwidth added for RTCP and the minimum interval for transmitting two consecutive RTCP packets are recommended to be set to 5 % and 5 seconds, respectively. So, it may take a few seconds till the streaming rate becomes suitable for the new network after a handoff. In a multicast environment, this delay may be much longer.

To solve these issues, the sender should be aware of the handoff timing of mobile nodes and accordingly adjust its streaming rate to the new network. In the proposed mechanism,

1-4244-0270-0/06/$20.00 (c)2006 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

mobile nodes explicitly notify their handoff timings to senders by using newly-defined RTCP packets and senders probe the available bandwidth in the new network by using low-priority RTP dummy packets. Extensive simulations are conducted to evaluate the performance of the proposed scheme. The results reveal that the proposed scheme achieves appropriate throughput and reduces the number of packet losses during the handoff operation.

The remainder of this paper is structured as follows. Section II surveys background activities on streaming rate control and handoff. Section III introduces the low priority RTP packets and the newly-defined RTCP packets, and describes the key concept behind the proposed scheme. In Section IV, the simulation environments are described and the simulation results are discussed. Following this, the paper concludes in Section V.

II. Related Work

RTP/RTCP communication over mobile networks has mainly two problems: streaming rate control and handoff management. In this section, we argue about these problems.

As previously mentioned, since RTP, itself, does not have any congestion control function, several streaming rate control techniques have been proposed in the recent literature [2]–[6]. In RTP/RTCP, RTP receivers send back RTCP RR packets which include the quality information, such as packet losses and jitter. RTP senders adjust their streaming rates based on these RR packets. However, there are two problems with streaming rate control: fairness towards competing TCP connections (*i.e.* TCP-friendliness) and robustness to sudden changes in network conditions.

In general, when TCP connections which constitute most of today's Internet traffic share one link with UDP type connections, the UDP connections conquer most of the bandwidth as TCP senders react to congestion by reducing their bandwidth consumption and UDP senders do not. Therefore, a UDP type connection should use the same bandwidth as a TCP connection when they traverse the same path. In recent years, a number of schemes have been proposed to achieve TCP-friendliness. [7] gives an overview of the recent research activities in this area. In TFRC (TCP Friendly Rate Control) [3][8], receivers monitor the loss event rate and continually report this information back to the sender. The latter measures then RTT (Round Trip Time) based on the feedback it receives. Using these two measurements, the sender adjusts its sending rate based on the complex TCP equation presented in [9]. LDA+ (Loss-Delay based Adaptation algorithm) [4] also relies on the RTCP messages and the equation of [9]. The sender adjusts its transmission rate based on end-to-end feedback information about losses, delays, and bandwidth capacity measured by the receiver. When no losses are observed, the sender can additively increase its transmission rate, otherwise it needs to multiplicatively reduce it as in TCP. However, in each method based on RTCP feedback, because RTCP messages may not be generated so frequently, the streaming rate cannot precisely adapt to the sudden changes of network.

On the other hand, when a mobile node moves between different base stations, a handoff is required. Furthermore, a mobile node needs to change its network address if the new point-of-attachment to the network differs from the old one. Mobile IP [10] tackles this problem. To reduce signaling overhead, handoff delay, or packet losses due to handoffs, many extensions such as Fast Handovers for Mobile IPv6 (FMIPv6) [11] and Hierarchical Mobile IPv6 (HMIPv6) [12] have been proposed. However, until the sender is notified of the new network address, transmitted packets are dropped during handoff. These schemes are thus not adequate for streaming in mobile networks. As a remedy to this issue, and to guarantee smooth handoff, several approaches [13]–[16] have been proposed. In [13] multiple paths are established between the server and a mobile node during handoff. Admittedly this scheme provides smooth handoff for streaming media. Nevertheless it uses multiple paths all the while a mobile node exists in the cell overlapping area. As in the case of a handoff between 3G and WLAN, if the distance of the cell overlapping area is long, it is unacceptable to use multiple paths for a long time as it causes redundant transmission of important data. [14] realizes a seamless handoff of streaming video by using two simultaneous connections through two separate WLANs and computing handoff occurrence time based on the delay difference. This approach is however effective for only horizontal handoffs as the delay differences may be due to difference of wireless access techniques in case of vertical handoffs.

III. Smooth Handoff Scheme with Low Priority RTP Packets and RTCP Packets

This section presents in detail the proposed scheme. First, we describe the preconditions that need to be applied to our scheme, and then define the new RTP/RTCP packets.

A. Preconditions

Firstly, it is assumed that wireless cells overlap with each other as shown in Fig. 1. To access two networks in parallel, a mobile node needs to be simply equipped with two wireless interfaces. It is difficult to imagine a mobile node with two WLAN cards, but [17] presents a mechanism where a single physical WLAN interface can be used to simultaneously access multiple WLANs. Moreover, if the wireless networks get integrated further, it will become normal for a mobile node to have interfaces of different wireless technologies. A special hardware that can simultaneously access all types of wireless technologies will be developed in the future. Secondly, the proposed scheme requires that all network elements (*e.g.* router) along the RTP/RTCP connection path support some priority disciplines. Currently, most networks are best-effort and most routers in the Internet do not apply any priority policy. However, in the near future, routers will be able to support multiple service classes in order to support real-time applications such as VoIP (Voice over IP). Lastly, a server-side application should be able to adjust its streaming rate by changing the quality of multimedia contents.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

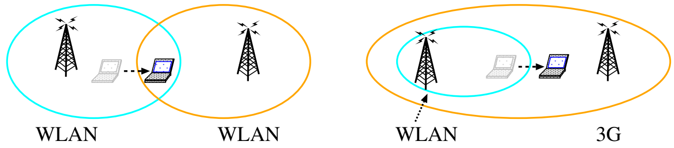


Fig. 1. Two different scenarios of handoff

B. Definition of New RTP/RTCP Packets

1) RTP Dummy Packets: RTP dummy packets are generated by the sender and are treated as low priority packets. They are used to estimate the available bandwidth in the new network. To distinguish RTP dummy packets from normal RTP packets, the payload type field in the RTP header of dummy packets is defined as a new payload type.

In research work related to TCP, several schemes which use low priority packets to probe the availability of network resources have been proposed. Notable examples are TCP-Peach [18] and Dummy Segment based Bandwidth Probing (DSBP) [19]. In these papers, low priority packets are called dummy segments since they do not carry any new information.

In [19], we used dummy segments to probe the available bandwidth for a mobile node after handoff in the new cells. Similarly to TCP dummy segments, RTP dummy packets are treated as low priority packets. Accordingly they do not affect the delivery of data traffic. Indeed, when a router on the connection path is congested, RTP dummy packets are discarded first.

2) RTCP Handoff Notification (HN): RTCP Handoff Notification (HN) packets notify the streaming server that a mobile node is about to perform a handoff. HN packets consist of only RTCP common header, and in order to distinguish HN from other RTCP packets, the packet type field in the RTCP header is set to an unused value. While normal RTCP packets are compounded and transmitted, RTCP HN packets are independently transmitted to the server upon detecting degradation in the link quality in order to immediately notify the handoff occurrence to the server.

3) RTCP Rate Estimation (RE): RTCP Rate Estimation (RE) is used so that the server calculates the available streaming rate for a mobile node. RE packets conform to RTCP RR, and their packet type fields are set to an unused value in order to distinguish RE from RR.

C. Algorithm of the Proposed Scheme

Fig. 2 portrays the procedures of the proposed scheme. A mobile node instantly measures radio strength or link quality. (1) It is assumed that a mobile node MN uses wireless interface IF1 in the cell of base station BS1 and receives data over RTP. (2) When MN moves into the new cell of base station BS2, a new network address is given to the MN's wireless interface IF2. And then, (3) when radio strength or link quality through IF1 goes down below a predefined threshold, (4) MN immediately transmits a RTCP HN message to the server through IF2. To guarantee a quick notification of the handoff occurrence, and thus a smooth handoff, this threshold should be set in function of the mobile node speed. Indeed, the

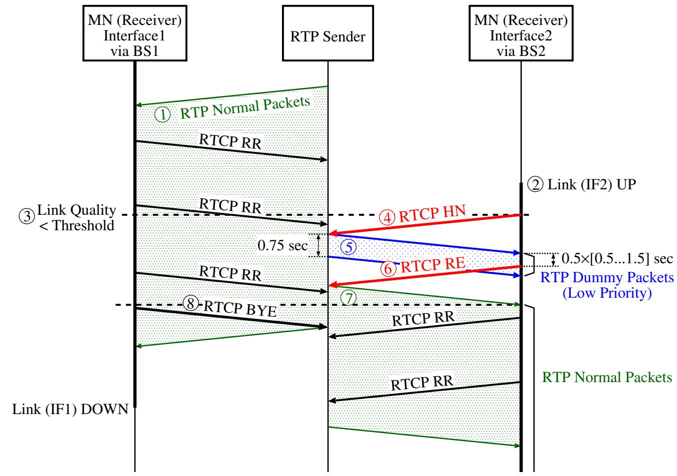


Fig. 2. Procedures of the proposed scheme

faster the mobile node, the higher value the threshold should be set to. Upon receiving HN, (5) the server transmits RTP dummy packets to MN's IF2 at the maximum streaming rate of contents for 0.75 seconds[1]. After receiving dummy packets for 0.5 seconds[1], (6) MN's IF2 returns RTCP RE to the server. Upon receiving RE, (7) the server calculates the appropriate streaming rate from the packet loss event rate and RTT as in the case of receiving a normal RTCP RR, and keeps streaming normal RTP packets (not dummy packets) at the computed rate. During these operations, MN keeps receiving the streaming data through IF1. (8) Once MN's IF2 starts receiving normal RTP packets, MN's IF1 transmits RTCP BYE to the sender, and the sender accordingly stops streaming data to IF1.

IV. Performance Evaluation

In order to verify the effectiveness of our scheme, we implemented the algorithm in the Network Simulator (ns-2.28) [20] and carried out extensive simulations. This section gives a detailed description of the simulation environment and discusses the simulation results.

A. Simulation Setup

In our simulations, the configuration of the considered network is depicted in Fig. 3. This topology imitates a common scenario where two base stations BS1 and BS2 are connected to the wired networks through a wireless gateway WG and a mobile node performs a handoff between BS1 and BS2. All wired links have a bandwidth of 155 Mbps (*e.g.* OC3). As for the wireless links capacity, a number of test scenarios were created by setting their capacity to different values from 384 kbps (*e.g.* 3G) to 11 Mbps (*e.g.* IEEE 802.11b). All links are error-free throughout this paper.

While it is more general to consider a sequence of handoffs, the behavior of the proposed scheme will be best understood by considering a single handoff. We thus focus on analyzing

[1]Actual receiving period is set to a random uniformly distributed between 0.25 and 0.75 ($= 0.5 \times [0.5 : 1.5]$) seconds in order to avoid traffic bursts from unintended synchronization with other sites. Since maximum period is 0.75 seconds, the server sends dummy packets for 0.75 seconds.

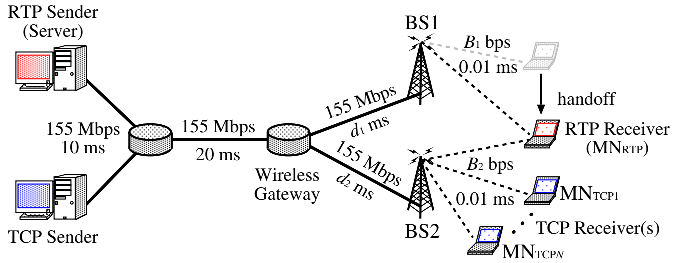


Fig. 3. Simulation network topology

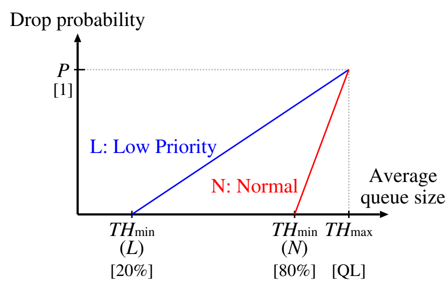


Fig. 4. Parameter configuration of WRED queue

a single handoff in all simulations. In our scheme, all network elements in the connection path need to support some priority disciplines as previously mentioned. In this paper, we use WRED (Weighted Random Early Detection) [21] for queuing. While the average queue size is between the minimum threshold $TH_{\min}$ and the maximum threshold $TH_{\max}$, the arriving packets are either dropped or queued, depending on the packet drop probability (Fig. 4). If the average queue size is greater than $TH_{\max}$, the packet is automatically dropped. WRED can selectively discard lower priority traffic (*i.e.* dummy packets) when the router begins to get congested and provide differentiated performance characteristics for different classes of service. The parameters of WRED are listed in Table I.

TABLE I
WRED PARAMETERS

Factor	Parameters
Queue Length (QL)	64 Packets
Min. Threshold for Normal Packets $TH_{\min}(N)$	80 % of QL
Min. Threshold for Dummy Packets $TH_{\max}(L)$	20 % of QL
Max. Threshold for All Packets $TH_{\max}$	100 % of QL
Mark Probability P	1

Although the proposed scheme can be used with any streaming rate control, we use LDA+ algorithm [4] in this paper. The reason behind this choice underlies beneath the fact that LDA+ achieves relatively good TCP-friendliness even when RTCP generates feedback messages with low frequency. In this context, it should be noted that whilst frequent transmissions of control messages is beneficial for quick adaptation to sudden changes in network conditions, it incurs overhead in terms of bandwidth consumption. Additionally, RTP is independent of the underlying protocol. Indeed, it can work on any type of network, such as TCP/IP, ATM, and frame relay. We apply UDP as transport protocol and IP as network protocol in the simulations. RTP can be also used in multicast environments. In this paper, we conduct all simulations only in unicast environments.

As comparison terms, we use two conventional schemes: a single-path scheme and a multi-path scheme. In the single-path scheme, we assume that there is no delay and no packet loss due to the handoff. In case of the multi-path scheme, two paths, via BS1 and BS2, are established after radio strength through one wireless interface IF1 goes down below a predefined threshold, as in the case of the proposed scheme. The sender then streams data to the other interface IF2 of the receiver without changing its rate. Table II shows an overall list of the simulation parameters.

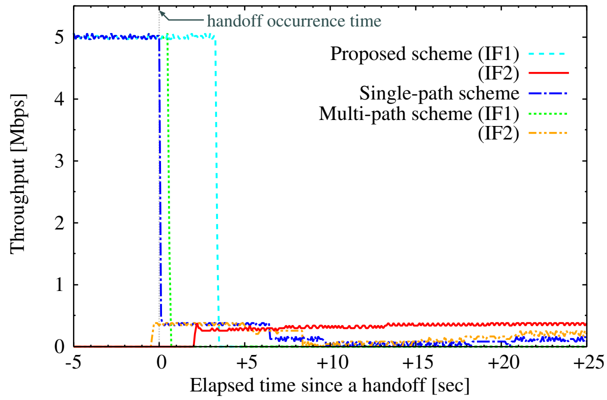
TABLE II
SIMULATION PARAMETERS

Factor	Simulation Parameters
Wired Links Capacity	155 Mbps
Wireless Links Capacity B_1, B_2	384 kbps – 11 Mbps
Link Delay between BS–WG d_1, d_2	1 ms
RTP Packet Size	512 Bytes
RTP/UDP/IP Packet Size	552 Bytes
TCP/IP Packet Size	1500 Bytes
Maximum Rate of Contents	5 Mbps
Total Number of TCP Receivers N	0 – 10
BER (Bit Error Rate)	0 (Error-Free)

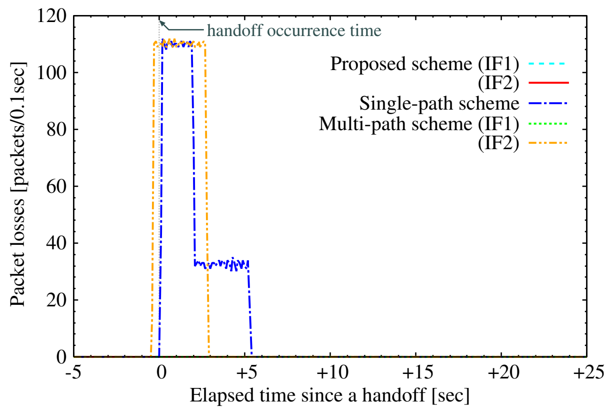
In this paper, two metrics are used to evaluate the performance of the proposed scheme: throughput and packet losses. The throughput indicates the number of bytes received by a mobile node (*i.e.* the receiver). The packet losses are the number of lost packets in the connection path per 0.1 seconds. It should be emphasized that RTP dummy packets are not considered in these computation since they do not carry any new information.

B. Simulation Results

As a first step, we investigate the performance of our scheme when only a single RTP connection exists ($N = 0$). In this simulation, two situations are envisioned: a mobile node moves from a higher bandwidth cell to a lower bandwidth cell or vice versa. Fig. 5 graphs the transition of throughput and packet losses when the RTP receiver MN_{RTP} performs a handoff from 11 Mbps to 384 kbps (*i.e.* $B_1 = 11$ [Mbps], $B_2 = 384$ [kbps]). Fig. 5(a) demonstrates that the conventional schemes achieve a little higher throughput compared to our scheme. However, they cause large number of packet losses as shown in Fig. 5(b). This is because the server keeps transmitting data at a high rate, which is suitable to the old cell, until a RR packet is received. Moreover, in the single-path scheme, after receiving a RR, due to the fact that the computed streaming rate depends on the old information, it does not suit the resource of the new cell. On the other hand, in the proposed scheme, RTCP HN notifies the sender that a mobile node is about to perform a handoff, and then the server appropriately adjusts its streaming rate to the new cell by using RTP dummy packets and RTCP RE. Therefore, no packet loss is observed during handoff, as shown in Fig. 5.



(a) Throughput



(b) Packet losses

Fig. 5. Handoff from 11 Mbps to 384 kbps (A single RTP connection)

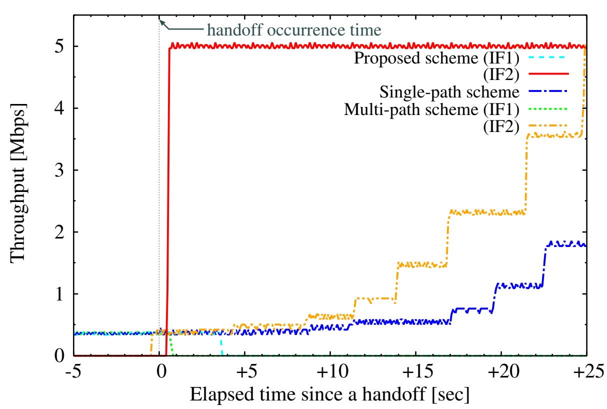


Fig. 6. Handoff from 384 kbps to 11 Mbps (A single RTP connection)

To consider the opposite scenario, we plot the throughput transition of three schemes when $\mathrm{MN_{RTP}}$ performs a handoff from 384 kbps to 11 Mbps. Here, since we observed no packet loss in all schemes, we do not graph packet losses. Fig. 6 illustrates that the proposed scheme achieves higher throughput compared to the conventional schemes immediately after the handoff. The obvious reason behind this performance consists in the fact that the conventional schemes keep transmitting data at the old rate which is inadequate for the higher bandwidth of the new cell.

Next, we direct our focus to evaluating the performance of the proposed scheme when a RTP connection shares one link with N TCP NewReno connections after the handoff. When the simulation starts, RTP receiver $\mathrm{MN_{RTP}}$ resides in the cell of BS1 and all TCP receivers $\mathrm{MN}_{\mathrm{TCP}i}$ $(i = 1, 2, ..., N)$ are in the cell of BS2 as shown in Fig. 3. $\mathrm{MN_{RTP}}$ then moves into the cell overlapping area and performs a handoff. $\mathrm{MN}_{\mathrm{TCP}i}$ do not roam. Fig. 7 shows the transition of throughput and packet losses of the RTP connection and a competing TCP connection when $\mathrm{MN_{RTP}}$ performs a handoff between 5 Mbps cells (*i.e.* $B_1 = B_2 = 5$ [Mbps], $N = 1$). For the sake of figure clarity and as the multi-path and single-path schemes exhibit relatively same behavior, Fig. 7 plots only the result of the single-path scheme and the proposed scheme. A glance at Fig. 7(a) reveals that the proposed scheme enables $\mathrm{MN_{RTP}}$ to enter into the new cell without affecting $\mathrm{MN_{TCP1}}$'s traffic. Furthermore, in our scheme, it takes less time to get a certain-level of TCP-friendliness than the conventional schemes. It is true that the throughput of $\mathrm{MN_{RTP}}$ in the conventional schemes is higher than that in our scheme after handoff, but $\mathrm{MN_{RTP}}$'s traffic in the conventional schemes causes a large number of packet losses not only in $\mathrm{MN_{RTP}}$'s traffic but also in $\mathrm{MN_{TCP1}}$'s traffic, as shown in Fig. 7(b). The throughput of $\mathrm{MN_{TCP1}}$ is also decreased unfairly for a few seconds, as shown in Fig. 7(a). As a result, use of conventional methods yields network congestion and degradation of TCP performance. Although there is a number of packet drops in the proposed scheme too, they are caused by the saw-like pattern of TCP congestion window during the congestion avoidance phase. Despite these packet drops, the proposed scheme succeeds in fairly dividing the bandwidth of the new network between the RTP and TCP connections, as shown Fig. 7(a). This performance is achieved by gradually decreasing the throughput of $\mathrm{MN_{TCP1}}$ and slowly increasing that of $\mathrm{MN_{RTP}}$.

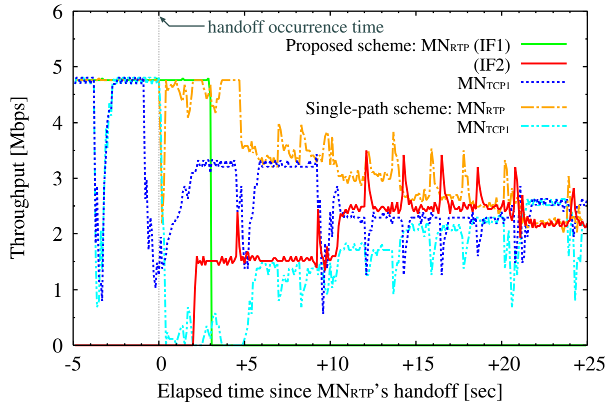
Lastly, we examine the TCP-friendliness in case of larger values of N. As a friendliness index, we define the following metric $f(x, \overline{y})$ based on Jain's fairness index [22].

$$f(x, \overline{y}) = \frac{(x + \overline{y})^2}{2\left(x^2 + \overline{y}^2\right)} \tag{1}$$

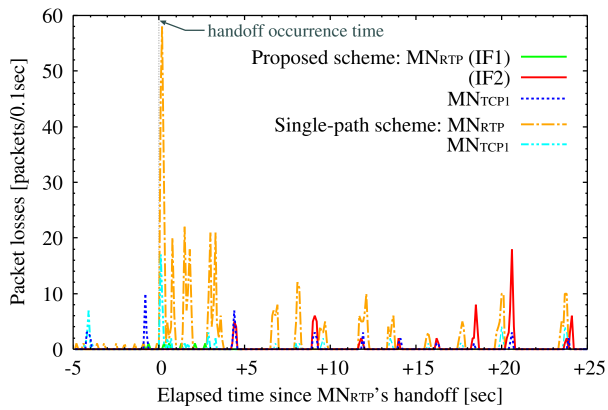
Here, x and $\overline{y}$ indicate the throughput of RTP connection and the average throughput of N TCP connections, respectively. In the proposed scheme and multi-path scheme, x refers to the throughput of the RTP flow via BS2. Each throughput is measured for 10 seconds after the handoff occurrence time. Fig. 8 plots the friendliness index of the three schemes as a function of the total number of competing TCP connections N. The results are an average of multiple simulation runs. From this figure, it can be easily understood that the proposed scheme achieves TCP-friendliness faster than the conventional schemes regardless of the total number of TCP flows N. Finally, it should be noted that similar results were obtained even when we change the parameters of WRED.

V. Conclusion

In this paper, we proposed a scheme based on low priority packets called RTP dummy packets and newly-defined RTCP packets for streaming data to mobile users over RTP/RTCP.



(a) Throughput



(b) Packet losses

Fig. 7. Throughput and packet losses variation of a RTP connection and a TCP connection sharing the bandwidth of the new network

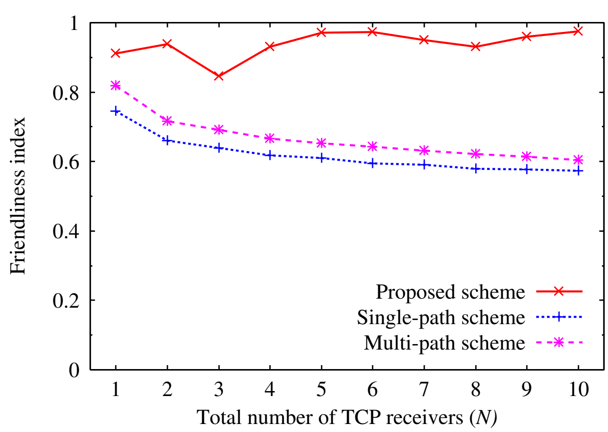


Fig. 8. Friendliness index vs. total number of TCP receivers

In our method, a mobile node is assumed to have at least two wireless interfaces. Prior to handoff occurrence, a mobile node explicitly notifies its handoff timing to its correspondent server by submitting a RTCP HN packet via a given interface. In response, the server estimates the available bandwidth in the new network by transmitting RTP dummy packets to the same interface of the mobile node. During this operation, the mobile node keeps receiving streaming data via the initially-used interface. By so doing, the server adjusts its streaming rate to the new network and accordingly guarantees smooth handoff management.

The performance of the proposed scheme was investigated through extensive simulations. Performance evaluation relied on computer simulation. The obtained results showed that our scheme achieved appropriate streaming rate compared to the available bandwidth immediately after a handoff occurrence without unfairly decreasing the throughput of competing TCP connections. Additionally, in the proposed scheme, a mobile node experienced lower packet drops during the handoff when it moves into a lower available bandwidth cell.

References

[1] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," RFC 3550, July 2003.

[2] I. Busse, B. Deffner, and H. Schulzrinne, "Dynamic QoS Control of Multimedia Applications based on RTP," *Computer Commun.*, vol. 19, no. 1, pp. 49–58, Jan. 1996.

[3] L. Gharai, "RTP Profile for TCP Friendly Rate Control," IETF Internet Draft, draft-ietf-avt-tfrc-profile-04.txt, work in progress, July 2005.

[4] D. Sisalem and A. Wolisz, "LDA+ TCP-Friendly Adaptation: A Measurement and Comparison Study," in *Proc. of 10th Int. Workshop on NOSSDAV*, Chapel Hill, NC, June 2000, pp. 1619–1622.

[5] J. Viéron and C. Guillemot, "Real-Time Constrained TCP-Compatible Rate Control for Video over the Internet," *IEEE Trans. Multimedia*, vol. 6, no. 4, pp. 634–646, Aug. 2004.

[6] D. Sisalem and A. Wolisz, "Towards TCP-Friendly Adaptive Multimedia Applications based on RTP," in *Proc. of 4th IEEE Sympo. on Computers and Commun. (ISCC '99)*, Red Sea, Egypt, July 1999, pp. 166–172.

[7] J. Widmer, R. Denda, and M. Mauve, "A Survey on TCP-Friendly Congestion Control," *IEEE Network*, vol. 15, no. 3, pp. 28–37, May/June 2001.

[8] M. Handley, S. Floyd, J. Pahdye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," RFC 3448, Jan. 2003.

[9] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation," in *Proc. of ACM SIGCOMM '98*, Vancouver, CA, Canada, Sept. 1998, pp. 303–314.

[10] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," RFC 3775, June 2004.

[11] R. Koodli, "Fast Handovers for Mobile IPv6," RFC 4068, July 2005.

[12] H. Soliman, C. Castelluccia, K. E. Malki, and L. Bellier, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)," RFC 4140, Aug. 2005.

[13] Y. Pan, M. Lee, J. B. Kim, and T. Suda, "An End-to-End Multipath Smooth Handoff Scheme for Stream Media," *IEEE J. Select. Areas Commun.*, vol. 22, no. 4, pp. 653–663, May 2004.

[14] G. Cunningham, S. Murphy, P. Perry, and L. Murphy, "Seamless Handover of Streamed Video over UDP between Wireless LANs," in *Proc. of IEEE CCNC 2005*, Las Vegas, NV, Jan. 2005, pp. 284–289.

[15] H. Matsuoka, T. Yoshimura, and T. Ohya, "A Robust Method for Soft IP Handover," *IEEE Internet Comput.*, vol. 7, no. 2, pp. 18–24, Mar./Apr. 2003.

[16] A. Helmy, "A Multicast-based Protocol for IP Mobility Support," in *Proc. of ACM SIGCOMM 2nd Int. Workshop on Networked Group Commun.*, vol. 7, no. 4, Palo Alto, CA, Nov. 2000, pp. 49–58.

[17] R. Chandra, Paramvir Bahl, and Pradeep Bahl, "MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Card," in *Proc. of IEEE INFOCOM 2004*, vol. 2, Hong Kong, China, Mar. 2004, pp. 882–893.

[18] I. F. Akyildiz, G. Morabito, and S. Palazzo, "TCP-Peach: A New Congestion Control Scheme for Satellite IP Networks," *IEEE/ACM Trans. Networking*, vol. 9, no. 3, pp. 307–321, June 2001.

[19] T. Taleb, K. Kashibuchi, N. Kato, and Y. Nemoto, "A Dummy Segment Based Bandwidth Probing Technique to Enhance the Performance of TCP over Heterogeneous Networks," in *Proc. of IEEE WCNC 2005*, vol. 4, New Orleans, RA, Mar. 2005, pp. 2400–2405.

[20] The Network Simulator - ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns/

[21] Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2. [Online]. Available: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_c/qcfbook.pdf

[22] R. Jain, A. Durresi, and G. Babic, "Throughput Fairness Index: An Explanation," ATM_Forum/99-0045, Feb. 1999.