Modified BLUE Packet Buffer for Base-Stations in Mobile IP-based Networks

Kyeong Hur, *Member, KIMICS*

***Abstract*—Performance of TCP can be severely degraded in Mobile IP-based wireless networks where packet losses not related to network congestion occur frequently during inter-subnetwork handoffs by user mobility. To solve such a problem in the networks using Mobile IP, the packet buffering method at a base station(BS) recovers those packets dropped during handoff by forwarding the buffered packets at the old BS to the mobile users. But, when the mobile user moves to a congested BS in a new foreign subnetwork, those buffered packets forwarded by the old BS are dropped and TCP transmission performance of a mobile user degrades severely. In this paper, we propose a Modified BLUE(MBLUE) buffer required at a BS to increase TCP throughput in Mobile IP-based networks. When a queue length exceed a threshold and congestion grows, MBLUE increases its packet drop probability. But, when a TCP connection is added at new BS by a handoff, the old BS marks the buffered packets. And new BS receives the marked packets without dropping. Simulation results show that MBLUE buffer reduces congestion during handoffs and increases TCP throughputs.**

***Index Terms*— TCP congestion control, Packet Buffering, Mobile IP, Buffer Management.**

I. INTRODUCTION

IN order to support user mobility in the Internet environment, IETF (Internet Engineering Task Force) has been designed Mobile IP protocol. There are a Base Mobile IP [1] and a Mobile IP with route optimization extension in Mobile IP protocol. The Base Mobile IP protocol can support host mobility without modification to existing routers or correspondent hosts. The Mobile IP with route optimization extension has been designed to solve the well-known triangle routing problem [2]-[4] in the Base Mobile IP protocol by informing correspondent hosts of the mobile host's care-of-address. In the Base Mobile IP protocol, the Home Agent (HA) in the home network of the mobile host(MH) intercepts the packets destined for the mobile host, and then delivers them to the mobile host's care-of-address using a tunneling technique. When a mobile host moves to a new foreign subnetwork, HA sends packets to old foreign subnetwork that mobile host stayed. So, until the old FA receives the binding update message from the new FA, there is a possiblity that packets are dropped in the events of handoffs. When packets are dropped due to mobility of mobile hosts, TCP interprets the packet losses as a sign of network congestion [5]. This makes the TCP exhibit severe performance degradation, especially when a mobile host user visits many subnetworks during the connection.

Because of packet losses by mobile host handoffs, the route optimization extension adopts smooth handoff scheme. In the route optimization extension with smooth handoff, when the new Foreign Agent(FA) receives the registration request message from the mobile host, it sends the binding update message to the old(previous) FA in order to deliver the new care-of-address of the new FA as well as to relay the registration request message. Each time the old FA receives a packet from the correspondent host(CH), it forwards the packets to the new FA. Because the new FA is always near to the old FA, packet losses reduce by using the smooth handoff so that it can reduce the number of packets dropped during that inter-subnetwork handoff. Furthermore, in utilizing the route optimization extension, the TCP performance sometimes becomes worse even than the case of the Base Mobile IP if its smooth handoff fails to avoid losses of four or more packets during an inter-subnetwork handoff[6]. To achieve a seamless handoff for the inter-subnetwork handoff, in the case where only the current FA performs packets buffering [6], [7], the current FA sends the buffered packets to the new FA when receiving the binding update message from the new FA.

However, there is a scalability problem because FA must cover all the TCP connections of the mobile hosts communicating with IP hosts at the outside of the subnetwork. So, in the case where only the current BS performs packet buffering, the scalability problem is alleviated because BS covers smaller TCP connections of the mobile hosts than FA[6], [8], [9]. But in the case where mobile host moves to congestion Base Station(BS) of the new subnetwork, forwarded packets by the old(previous) BS are dropped and TCP performance could be degraded due to the forwarded burst packets [10], [11].

Manuscript received July 11, 2011; revised July 29, 2011; accepted August 5, 2011.

Kyeong Hur is the corresponding author with the Department of Computer Education. Gyeongin National University of Education, Incheon, Korea(e-mail: khur@ginue.ac.kr).

In this paper, we propose a Modified BLUE(MBLUE) buffer required at a BS to increase TCP throughput in Mobile IP-based networks. MBLUE buffer keeps the overall TCP throughput high while it maintains a small average queue length and mitigates congestion at a handoff. When a queue length exceeds a threshold and congestion grows, MBLUE increases its packet drop probability. But when a TCP connection is added at a new BS by a handoff, the old BS marks the buffered packets. And new BS receives the marked packets without dropping. Unlike Drop-tail, RED, and BLUE buffers, MBLUE buffer avoids a large decrease of TCP throughputs due to global synchronization when a TCP connection is added at the new BS by a handoff.

The rest of the paper is organized as follows. In Section 2, we present a brief overview of the BS packet buffering method in Mobile IP based networks. Section 3 includes a detailed description of the proposed MBLUE apparatus for improving TCP performance in Mobile IP with BS packet buffering. In Section 4, through simulations, we first investigate influences of the BS packet buffering method on TCP performance in a congested FA, then we evaluate the impact of the proposed MBLUE apparatus on the TCP performance for the same FA. Section 5 gives concluding remarks.

II. ROUTE OPTIMIZATION EXTENSION OF MOBILE IP STANDARD WITH PACKET BUFFERING AT THE BS

The network configuration of the Mobile IP based networks shown in Fig. 1 is considered in this paper, where it is assumed that the router in each subnetwork also plays the role of FA. In the route optimization extension shown in Fig. 2, when the new FA receives the registration request message from the mobile host, it sends the binding update message to the old (previous) FA in order to deliver the new care-of address of the new FA as well as to relay the registration request message. Each time the old FA receives a packet from the correspondent host (CH), it forwards the packets to the new FA. Also, it sends the binding warning message to the Home Agent (HA) because the correspondent host cannot know the new care-of address until it is informed by the HA. When the HA receives the binding warning message, it sends the binding update message to the correspondent host in order to inform the new care-of address. After receiving the binding update message, the correspondent host can send packets to the new FA instead of the old FA. Thus, the above process, called a smooth handoff, can reduce the number of packets dropped during the inter-subnetwork handoff.

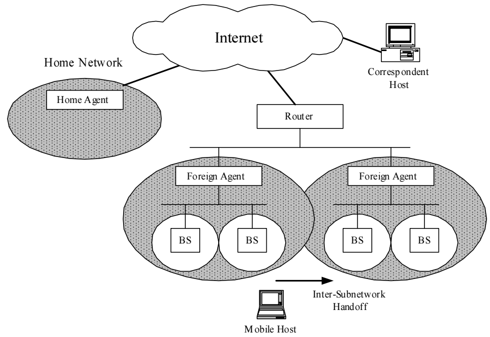


Fig. 1. Network Configuration of Mobile IP based Networks

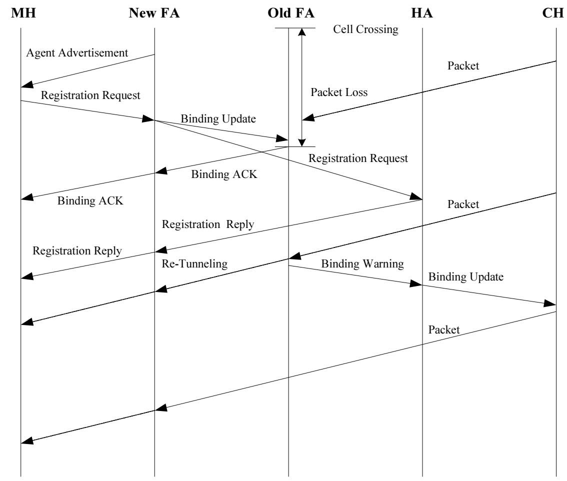


Fig. 2. Mobile IP Protocol with Route Optimization Extension

The Mobile IP standard recommends to limit the maximum sending rate of the agent advertisement message to once per second for reducing the network load caused by the agent advertisement message [1]. Therefore, even if the agent advertisement message is broadcasted at the maximum sending rate by the new FA, the mobile host cannot receive it during one second after moving into the new foreign network in the worst case. It means that during that time, in-flight packets destined for the mobile host are dropped because the mobile host cannot send the registration request message until receiving the agent advertisement message. For this problem, we incorporate a local handoff protocol in the route optimization extension as shown in Fig. 3. After receiving the beacon message, which plays the same role as the agent advertisement message, the mobile host sends the handoff request message to the new BS. The new BS then sends the notification message to the new FA for requesting the agent advertisement message. Upon receiving the notification message, the new FA sends the agent advertisement message to the new BS. By this method, the mobile host can receive the agent advertisement message more quickly, compared to the method which periodically broadcasts the agent advertisement message to all of the BSs in the subnetwork. It is because the beacon message used in the local handoff protocol is usually much shorter than the agent advertisement message and thus its sending rate is much higher than the maximum sending rate of the agent advertisement message [1].

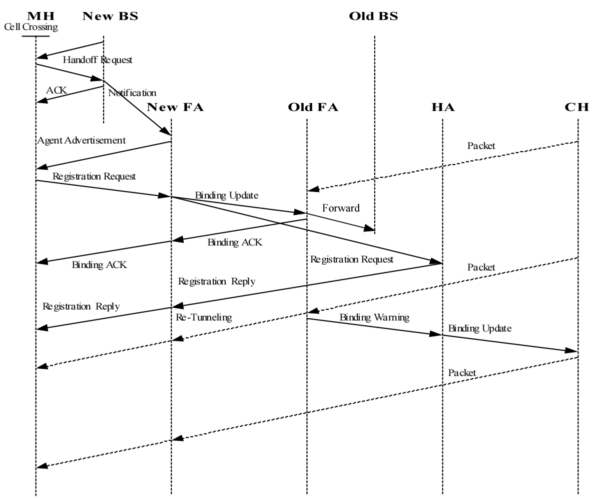


Fig. 3. Route Optimization Extension of Mobile IP Protocol with BS Packet Buffering

For an intersubnetwork handoff, in a Base Mobile IP and a Mobile IP with route optimization extension, we can calculate of time interval as shown in (1) [6]. T_{loss} denotes the period during which in-flight packets are dropped. T_{RTT} is the round-trip-time of a TCP connection, and T_h is the period from the time when the mobile host moves into the new foreign subnetwork to the time when the correspondent host can know the new-care-of address of the mobile host by receiving the binding update message from the HA.

$$T_{loss\_Base} = T_B + T_N + T_{HA} + D_{HA\_FA} \quad , \quad T_{loss\_Route} = T_B + T_N + T_O$$
$$T_{h\_Base} = T_B + T_N + T_{HA} \quad , \quad T_{h\_Route} = T_B + T_N + T_{HA} + T_{CH} \tag{1}$$
$$T_{RTT\_Base} = D_{CH\_HA} + D_{HA\_FA} + D_{CH\_FA} + a \quad , \quad T_{RTT\_Route} = 2 \cdot D_{CH\_FA} + a$$

D_{HA_FA} : the delay taken to send a packet between HA and old FA
D_{CH_HA} : the delay taken to send a packet between CH and HA
D_{CH_FA} : the delay taken to send a packet between CH and FA
T_B : the period from the time when the mobile host moves into the new foreign subnetwork to the time when the mobile host receives beacon message from the new BS
T_N : the period from the time when the mobile host receives beacon message to the time when the new FA sends the registration request message or the binding update message
T_{HA} : the period from the time when the new FA sends the registration request message to the time when the HA receives it
T_O : the period from the time when the new FA sends the binding update message to the time when the old FA receives it
T_{CH} : the period from the time when the HA sends the binding update message to the time when the correspondent host (CH) receives it ($\approx D_{CH\_HA}$)
a : round-trip-time between FA and mobile host

The buffering method can be implemented at a BS by maintaining one buffer per mobile host, rather than maintaining one buffer per TCP connection, which resolves the possible scalability problem and could support other transport protocol like UDP more easily [6]. In addition, BSs should maintain a table that stores the list of currently attached mobile hosts. When the old BS receives the forward message including the address of the mobile host which has moved to another wireless cell, it just forwards the buffered packets to the mobile host and then deletes the entry for the mobile host from the table. Furthermore, the packets re-ordering problem which causes duplicate ACKs should be considered. When the correspondent host sends packets without knowledge of the new care-of address, the old FA forwards those packets to the new FA in the route optimization extension [7]. However, if those packets arrive at the old FA before it completes the forwarding of the buffered packets from the old BS, packets could arrive at the mobile host in a wrong order. TCP performance could be degraded worse than in the case in which the packet buffering is not supported. This is direct results of the duplicate ACKs caused by a re-ordering of packets. To solve this problem, the old FA forwards those packets to the new FA through the old BS as shown in Fig. 4. Namely, the old FA sends those packets to the old BS, and then the old BS puts those packets into the buffer of the mobile host. Finally, the packets in the buffer are forwarded to the new FA through the old FA. Then, packets for the mobile host are forwarded to the new FA in the correct order.

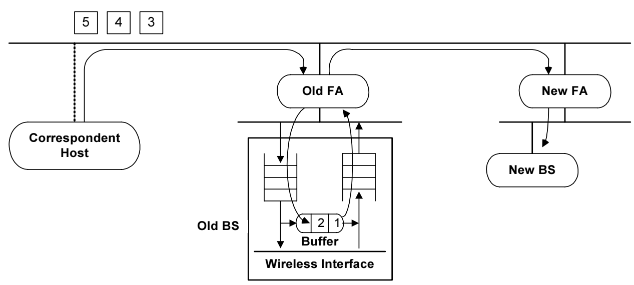


Fig. 4. Packet Delivery Sequence for avoiding re-ordering of packets

III. A MODIFIED BLUE BUFFER MANAGEMENT SCHEME (MBLUE)

A. RED (Random Early Detection)

The problem of congestion control with Drop-tail buffer is the late notification of congestion since the TCP source recognizes packet losses after the Drop-tail buffer overflows and it drops many packets burstly. This property of Drop-tail buffer causes the global synchronization problem of TCP sources. In order to solve this problem, RED buffer management scheme has been proposed [12].

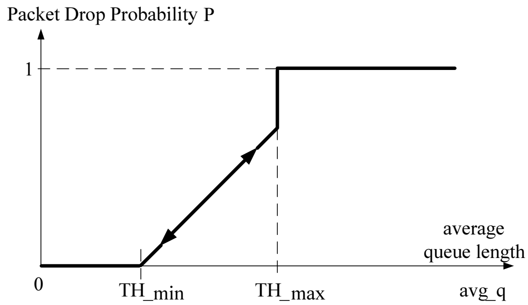


Fig. 5. RED Buffer Management Scheme

As shown in Fig. 5, the parameters of RED buffer algorithm are TH_{min}, TH_{max}, avg_q, P_{max}, and w_q. In [12], authors suggest to set P_{max} to 0.02 and w_q to 0.002. In RED algorithm, if the average queue size(avg_q) exceeds TH_{min}, arriving packets are dropped randomly. If avg_q exceeds a threshold TH_{max}, all arriving packets are dropped. Therefore, before the RED buffer overflows, arriving packets are random early dropped by controlling average queue length. Consequently, RED buffer mitigates the problem of global synchronization caused by simultaneously dropped packets from many TCP flows. The operations of a RED buffer is explained in (2).

$$
\begin{array}{l}
\textit{if} \;\; avg\_q < TH_{\min} \\
\qquad \textit{queue packet} \\
\textit{else if} \;\; TH_{\min} \le avg\_q \le TH_{\max} \\
\qquad \textit{calculate probability} \;\; P \\
\qquad \textit{with probability} \;\; P \\
\qquad\quad \textit{discard packet} \\
\qquad \textit{else queue packet} \\
\textit{else if} \;\; avg\_q \ge TH_{\max} \\
\qquad \textit{discard packet}
\end{array}
\tag{2}
$$

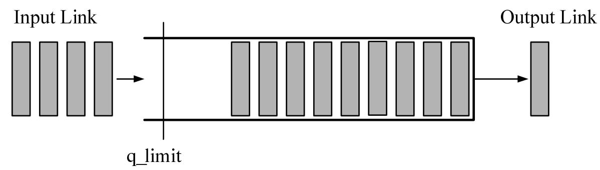
B. BLUE

The buffer management scheme for congestion control has to minimize packet losses and queuing delay while maintaining high link utilization by preventing the global synchronization. But, RED buffer must have sufficient queue length and accurate parameter values for optimal congestion control. BLUE buffer management scheme has been proposed in order to solve this problem of RED buffer [13]. BLUE algorithm manages its buffer directly only with both packet losses and current link utilization, which makes BLUE buffer different from other buffers using current occupation ratio of the buffer. When packets go into a BLUE buffer, they are discarded according to a packet drop probability, P_m [13]. When a packet is dropped due to buffer overflow at BLUE buffer, *d1* is added to P_m. On the other hand, when BLUE buffer is empty, *d2* is subtracted from P_m. The operations of BLUE buffer is explained in (3).

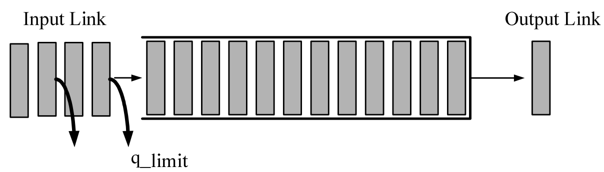
With probability P_m
discard packet
else queue packet

When a packet is randomly dropped or dropped due to buffer overflow:
if ((now_update - last_update) > freeze_time) then
$P_m = P_m + d1$
last_update = now_update
(3)

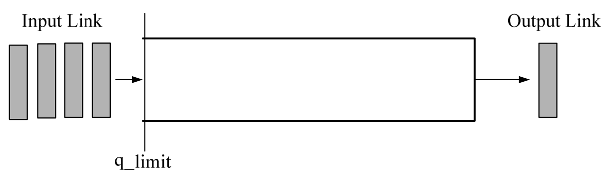
When the BLUE buffer is empty:
if ((now_update - last_update) > freeze_time) then
$P_m = P_m - d2$
last_update = now_update



(a) Maintenance of packet drop probability when queue length is less than *q_limit*



(b) Increase of packet drop probability when packet losses due to buffer overflow occur



(c) Decrease of packet drop probability when queue is empty

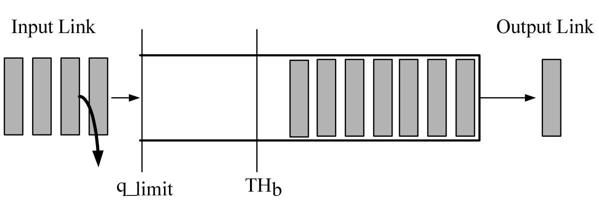
Fig. 6. Relationship between queue length and packet drop probability in BLUE buffer

In controlling P_m, *freeze_time* determines minimum time interval between two consecutive updates. The value of *freeze_time* has to be established basically based on RTT periods in order to notify packet drops periodically to TCP flows. And the values of *d1* and *d2* have to be set optimally according to characteristics of aggregate TCP traffic. In general, *d1* is greater than *d2* because BLUE should responds to variations of traffic fast. Fig. 6 explains that the relationship between queue length and packet drop probability in the BLUE buffer.

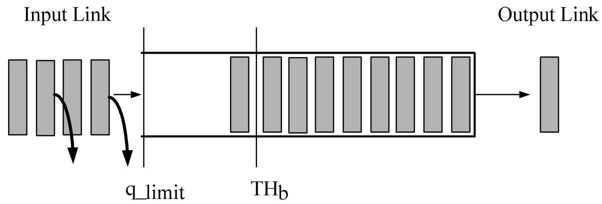
C. MBLUE

As is explained before, Drop-tail buffer is not suitable for mitigating TCP performance degradation of mobile hosts in Mobile IP-based networks with packet buffering. RED buffer shows better performance than the Drop-tail, but TCP performance degradation due to global synchronization still appears in the RED buffer. MBLUE buffer is designed to decrease packet losses of the mobile host, and to prevent global synchronization, which increases TCP throughputs when a handoff TCP connection is added to a new BS. Therefore, MBLUE algorithm sets up a new threshold TH_b to admit the buffered burst packets forwarded by the old BS into a new space of ($q\_limit - TH_b$), in contrast to the original BLUE algorithm. In MBLUE, when queue length is greater than TH_b, MBLUE buffer early increases P_m. This operation gives a remaining space of ($q\_limit - TH_b$) to MBLUE buffer. Consequently, the average queue length is maintained below TH_b. And, it can avoid buffer overflow drops when burst packets are forwarded from the old BS during handoffs.

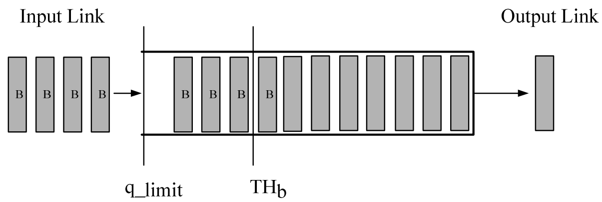
Furthermore, the old BS marks the buffered packets during handoff using ToS(Type of Service) field of Mobile IP packet header and forwards the marked packets to the new BS equipped with MBLUE buffer. And MBLUE buffer does not drop the marked packets from the old BS and sets the packets drop probability P_m to zero. This operation of MBLUE buffer solves the problem of the global synchronization during handoffs because the arrivals of marked packets mean the occurrence of handoffs and MBLUE buffer does not perform burst packet drops even though there exists increased congestion at handoffs. Also, these operations of MBLUE protect the forwarded packets of TCP mobile host and then the seamless handoff is offered to the mobile host with increased TCP throughput. Fig. 7 shows the relationship of queue length and packet drop probability in MBLUE buffer. The operations of MBLUE buffer is explained in (4) and in Fig. 8.



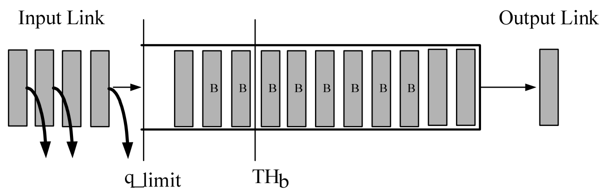
(a) Maintenance of packet drop probability when queue length is less than TH_b



(b) Increase of packet drop probability when queue length is more than TH_b



(c) No packet drop(P_m=0) when marked packet from the old BS arrives



(d) Increase of packet drop probability when queue length is more than TH_b in the case that an unmarked packet arrives

Fig. 7. Relationship of queue length and packet drop probability in MBLUE buffer

when an unmarked packet arrives:
if $q\_length \geq TH_b$
  if $((now\_update - last\_update) > freeze\_time)$ *then*
    $P_m = P_m + d1$
    $last\_update = now\_update$

when a marked packet arrives: (4)
    queue packet
    $P_m = 0$

When the MBLUE buffer is empty:
  if $((now\_update - last\_update) > freeze\_time)$ *then*
    $P_m = P_m - d2$
    $last\_update = now\_update$

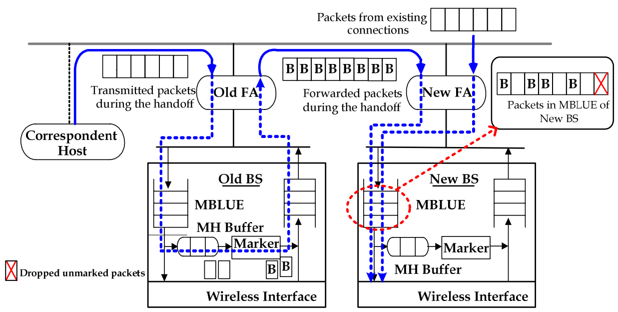


Fig. 8. Illustration of the MBLUE operations with BS packet buffering

IV. SIMULATION RESULTS AND ANALYSES

A. Simulation model

To investigate the influence of the buffering method on TCP performance in Mobile IP with route optimization extension, we developed the model shown in Fig. 9. The model includes a mobile host, which moves between two wireless cells in different subnetworks while maintaining a TCP connection with a fixed correspondent host, and the bandwidth of the wireless link of the new BS is shared among three TCP connections. In this case, because the bottleneck may be the wireless link, the buffer in the new BS is likely to be fully utilized by those TCP connections already presented in the new BS. Fig. 9 explains Mobile IP signaling and delivery sequence of a TCP packet flow in case that when the mobile host moves to the congested new BS in a new subnetwork.

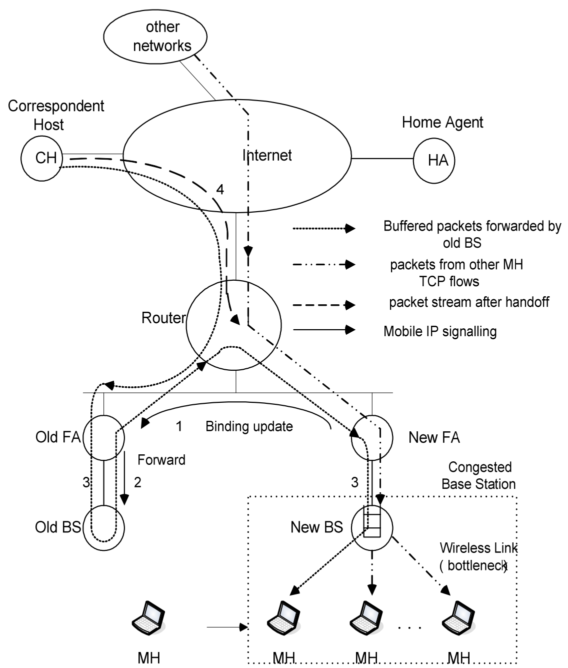


Fig. 9.Forwarding buffered packets to a congested new BS

TABLE I
DELAYS BETWEEN NODES CONSTRUCTING HANDOFF PROCESSING DELAY

Parameter	Value
T_B	50 ms
T_N	12 ms
T_{HA}	60 ms
T_{CH}	60 ms
T_O	1 ms
D_{CH_FA}	40 ms
α	8 ms
T_{h_Route}	182 ms
T_{loss_Route}	63 ms
T_{RTT_Route}	88 ms

TABLE II
PARAMETER OF RED BUFFER MANAGEMENT SCHEME

Parameter	Value
Queue Size (packets)	120
Minimum Threshold (TH_{min})	40
Maximum Threshold (TH_{max})	80
Maximum value (P_{max})	0.02
Queue Weight (w_q)	0.002

TABLE III
PARAMETER OF BLUE AND MODIFIED BLUE BUFFER MANAGEMENT SCHEME

Parameter	Value
Queue Size (packets)	120
Threshold (TH_b)	80
Increasing Drop Probability ($d1$)	0.0001
Decreasing Drop Probability ($d2$)	0.00001
Freeze time period (s)	0.001

Fig. 10 shows our simulation model. Each BS has its wireless link of 2 Mbps bandwidth. And it is connected to 10 Mbps Ethernet. The delay taken to send a packet between adjacent FAs is set as 1 ms by considering the 10 Mbps Ethernet. Similarly, the delay taken to send a packet between FA and BS is also set as 1 ms. Table 1 shows values of the handoff delay components for the route optimization extension which are determined as shown in (1). The period of beacon message broadcastings, T_B, is set as 50 ms [6]. The RTTs of three TCP connections in the new BS have 86, 88, and 90 ms, respectively. Also, we consider a congestion control of TCP Reno sources. And we set packet size as 512 bytes and *mws* to 64 packets. We assume that there is no packet loss by transmission link errors. The parameter values of RED buffer are shown in Table 2. And those ones of BLUE and

MBLUE buffers are set as shown in Table 3. We executed simulations with ns-2 simulator.

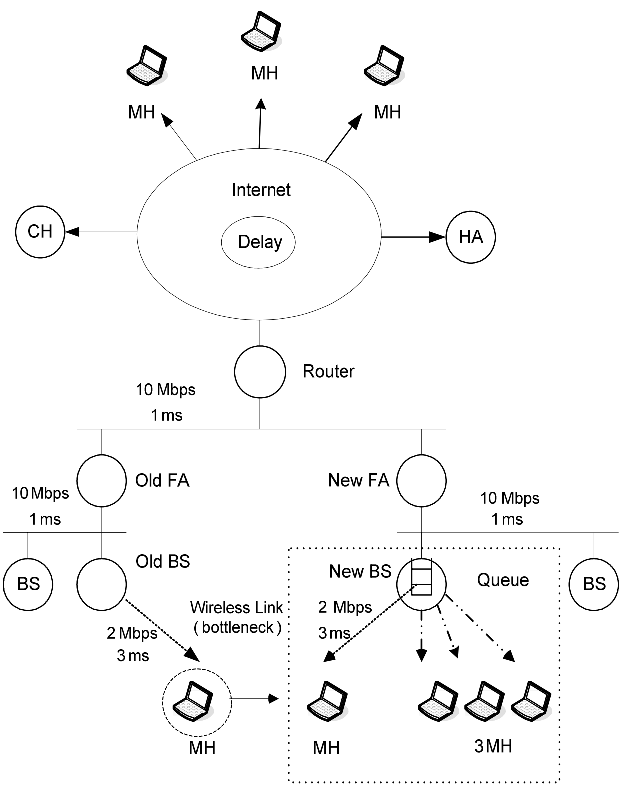


Fig. 10. Simulation model to evaluate buffer management schemes

B. *Performance Analysis of each buffer for Mobile Host Handoffs*

RED, BLUE, and MBLUE buffers have been tested respectively at the new BS shown in Fig. 10. While three TCP mobile hosts which respectively have 88, 90, and 92 ms RTTs are communicating in the new BS.

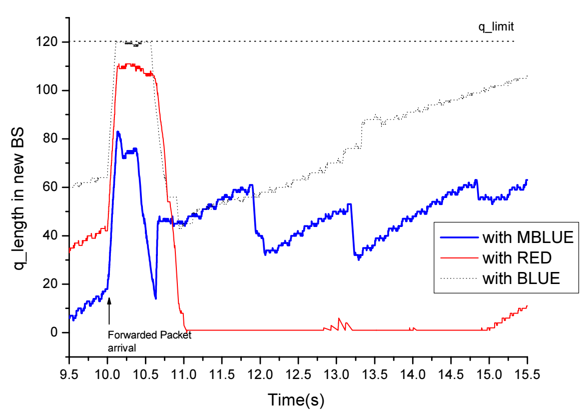


Fig. 11. Arrival property of buffered packets at the new BS

Fig. 11 shows *q_length* variation at each buffer when a mobile host with a TCP connection moves into the new BS at 10 s. In Fig.11, it is shown that a maximum of one window of burst 64 packets arrive from the old BS during the short packet forwarding period. That is, from 10s to 10.182s, the buffered maximum window size packets are forwarded to the new BS.

In the case of BLUE, queue length increases rapidly as many as 64, *mws* during 10s to 10.182s. And following burst packets are dropped due to buffer overflow so that global synchronization appears. Similarly, queue length increases quickly in RED buffer and its *avg_q* is also increased that its packet drop probability is raised. This operation of RED buffer causes global synchronization of TCP flows after a mobile connection handoff as shown in Fig. 11. On the contrary, MBLUE buffer's queue length increases at a handoff. But, MBLUE's packet drop probability, P_m decreases close to zero when MBLUE buffer detects the marked handoff packets by receiving from the old BS. This operation of MBLUE that MBLUE buffer reduces packet drops whenever a handoff occurs in Mobile IP-based networks with packets buffering prevent global synchronization of TCP flows at handoffs as shown in Fig. 11.

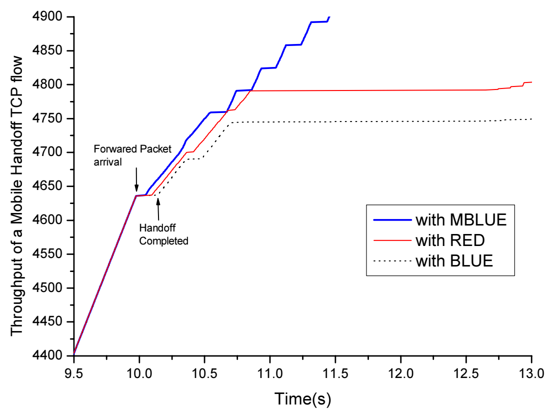


Fig. 12. TCP throughput of the handoff mobile host

Fig. 12 compares TCP throughput of the handoff mobile host at new BS with each buffer. When using BLUE or RED buffer at new BS, TCP throughput of the handoff mobile host gets worse due to global synchronization than when using MBLUE buffer. This results can be inferred from results in Fig. 11. From this simulation result, we can know that using MBLUE buffer at a BS is effective to increase TCP throughputs in a Mobile IP with route optimization extension with packet buffering.

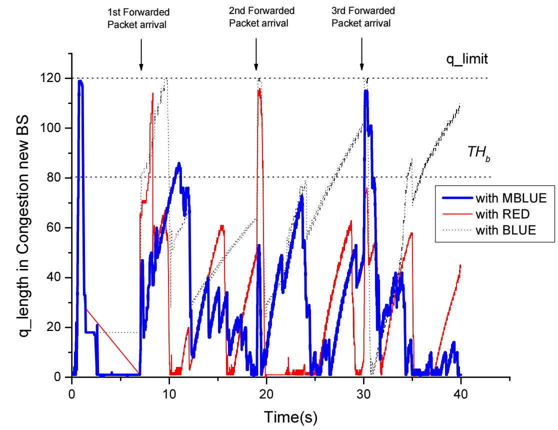


Fig. 13. Queue length in the congested new BS

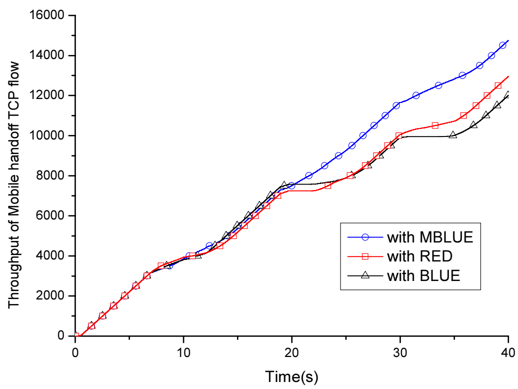


Fig. 14. TCP throughput of the handoff mobile host

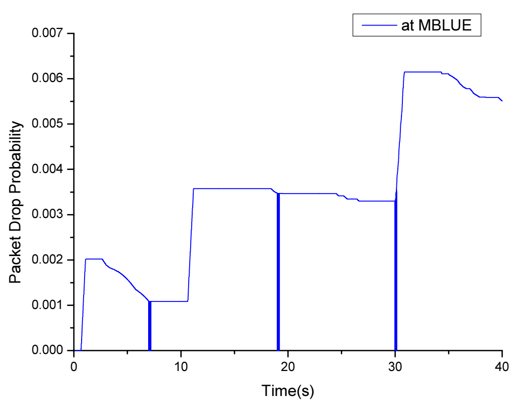


Fig. 15. Packet drop probability variation of MBLUE buffer

In Fig. 13, we investigate arrival property of buffered packets from old BS at BLUE, RED, and MBLUE buffers. In Fig. 13, a mobile host with a TCP connection moves into the new BS where three TCP connections already exist 3 times for 40s. At the first handoff, a few packet drops are found in all buffers. However, after the second handoff, global synchronization occurs commonly due to forwarded burst packets. Fig. 14 compares TCP throughput of the handoff mobile host at each buffer. Because MBLUE's queue length keeps below TH_b and P_m is set as zero at handoffs as shown in Fig. 15, packet drops almost do not occur at handoffs in MBLUE buffer. Then, using MBLUE buffer reduces packet losses of both the mobile host and already communicating hosts. So, the MBLUE buffer increases TCP throughputs more than RED buffer as shown in Fig. 14. RED buffer management scheme which determines packet drop probability, P_b based on avg_q is difficult to stabilize the queue length. On the other hand, MBLUE stabilizes more than RED at queue length because of regular packet drop. Because packet drop probability increases when queue length is larger than TH_b in MBLUE, the queue maintains stability and global synchronization due to buffer overflows is prevented. Because packets buffered at the old BS do not be discarded, proposed MBLUE buffer guarantees improved throughput of a handoff mobile host and a seamless handoff.

ACKNOWLEDGMENT

This work was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (MEST) (2010-0002366) and in part by Mid-career Researcher Program through NRF grant funded by the MEST (2011-0016145).

IV. CONCLUSIONS

In this paper, to solve the problem of BS buffer in Mobile IP-based Networks with packet buffering, the MBLUE buffer is proposed. Unlike the previous BS buffers, the MBLUE buffer solves the global synchronization problem due to burst packet drops at handoffs without additional buffer. Because MBLUE's queue length keeps below TH_b and packet drop probability is set as zero at handoff, packet drops almost do not occur at handoffs in MBLUE buffer. Simulation results show that the MBLUE buffer increases TCP throughputs larger than the conventional BS buffer management schemes.

REFERENCES

[1] C. E. Perkins, "IP Mobility Support for IPv4," revised draft-ietf-mobileip-rfc2002-bis-03.txt, 2000.

[2] C. E. Perkins, "Mobile IP," *International Journal of Communications Systems*, pp. 3-20, 1998.

[3] C. E. Perkins, “Mobile IP,” *IEEE Communications Magazine*, pp. 84-99, May 1997.
[4] C. E. Perkins and D. E. Johnson, “Route optimization in Mobile IP,” draft-ietf-mobileip-optim-10.txt, IETF Network Working Group, 2000.
[5] R. Caceres and L. Iftode, “Improving the performance of reliable transport protocols in mobile computing environments,” *IEEE Journal on Selected Areas on Communications*, vol. 13, no. 5, pp. 100-109, Nov. 1995.
[6] D. S. Eom, M. Sugano, M. Murata, and H. Miyahara, “Performance Improvement by Packet Buffering in Mobile IP Based Networks,” *IEICE Transactions on Communications*, vol. E83-B, no. 11, pp. 2501-2512, Nov. 2000.
[7] D. S. Eom, H. S. Lee, M. Sugano, M. Murata, and H. Miyahara, “Improving TCP handoff performance in Mobile IP based networks,” *Computer Communications*, vol. 25, no. 7, pp. 635-646, May 2002.
[8] R. Caceres and V. Padmanabhan, “Fast and scalable handoffs for wireless networks,” *Proceedings of ACM Mobicom'96*, pp. 56-66, Nov. 1996.
[9] H. Balakrishanan, V. N. Padmanabhan, S. Sehan, and R. H. Katz, “A comparison of mechanisms for improving TCP performance over wireless links,” *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 756-769, Dec. 1997.
[10] S. B. Lee, K. Hur, J-K. Kim, and D-S. Eom, “A Handoff Packet Marker for DiffServ in Mobile IP-based Networks with Packet Buffering,” *IEEE Transactions on Consumer Electronics*, Vol.55, No.2, May 2009.
[11] S. B. Lee, K. Hur, J. Park and D-S. Eom, “A Packet Forwarding Controller for Mobile IP-based Networks with Packet Buffering,” *IEEE Transactions on Consumer Electronics*, Vol.55, No.3, August 2009.
[12] S. Floyd and V. Jacobson, “Random Early Detection Gateways for Congestion Avoidance,” *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397-413, Aug. 1993.
[13] W. Feng, K. G. Shin, D. D. Kandlur, and D. Saha, “The BLUE active queue management algorithms,” *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, pp. 513-528, Aug. 2002.

Kyeong Hur (corresponding author) is currently an Assistant Professor in the Department of Computer Education at Gyeongin National University of Education, Republic of Korea. He was senior researcher with Samsung Advanced Institute of Technology (SAIT), Korea from September 2004 to August 2005. He received a M.S. and Ph.D. in Department of Electronics and Computer Engineering from Korea University, Seoul, Korea, in 2000 and 2004, respectively. His research interests include; computer network designs, next generation Internet, Internet QoS, and future All-IP networks.