A Diagnosis-Based Soft Vertical Handoff Mechanism for TCP Performance Improvement

Tong Shu[1,2], Min Liu[1], Zhongcheng Li[1], Anfu Zhou[1]
[1]Institute of Computing Technology, Chinese Academy of Sciences
[2]Graduate University of Chinese Academy of Sciences
Beijing, P.R.China, {shutong, liumin, zcli, zhouanfu}@ict.ac.cn

***Abstract*—Most existing soft handoff approaches lead to plenty of out-of-order packets during downward vertical handoffs (VHOs). We have presented a soft VHO scheme, called SHORDER, to avoid packet reordering caused by downward VHOs. In this paper, we analyze the effects of our SHORDER scheme and another typical existing soft VHO method on the handoff latency and the received data size during a downward VHO for TCP applications. Then, we approximately derive the applicable conditions of the two approaches, and further propose a diagnosis-based soft vertical handoff (DSVH) mechanism which can self-adaptively deal with reordering packets. The mechanism has practical advantages of no changes to correspondent nodes and compatibility with various enhanced TCP variants. With numerical analysis and test-bed experiments, we show that the DSVH mechanism has better performance than the SHORDER scheme and the typical existing method. Furthermore, experimental and analytical results are consistent with each other.**

I. INTRODUCTION

To achieve a truly seamless mobile environment, vertical handoffs (VHOs) are a major challenge. Vertical handoffs refer to handoffs between BSs and APs using different wireless network technologies, and are divided into downward vertical handoffs (DVHOs) and upward vertical handoffs. The former are handoffs to a wireless overlay with a smaller cell size and higher bandwidth per unit area, such as handoffs from wireless wide area networks (WWANs) to wireless local area networks (WLANs). The latter are reverse. To improve the performance of vertical handoffs, soft handoff methods are used for avoiding packet loss. In a soft handoff, a mobile node (MN) does not release the old link until it has established the new link [1].

Most existing soft vertical handoff approaches, such as the client-based soft handoff (CSH) method [2], do not take the sequence of packets into account, and thus may result in out-of-order delivery of packets during DVHOs. In general, out-of-order packets can degrade the performance of applications [3]. In order to avoid packet reordering during DVHOs, we have proposed a soft vertical handoff scheme, called SHORDER (Soft Handoff in ORDER) [4], which can effectively eliminate packet reordering as seen by the transport layer. Based on our previous research, this paper quantitatively analyzes the impact of different soft VHO approaches on TCP performance. We discover that the time cost of eliminate packet reordering is sometimes higher than that of packet retransmission. Therefore, we need to make a trade-off between them. From our analytical results, we conclude that the SHORDER scheme is suitable for the case that the previous round-trip time (RTT) between the MN and its correspondent node (CN) is less than triple the new RTT. According to the condition, we further present a diagnosis-based soft vertical handoff (DSVH) mechanism to improve TCP performance.

In this paper, the main contributions of our work are as follows:

- We analyze the effects of the SHORDER and CSH approaches on TCP performance in terms of the handoff latency and the received data size during a DVHO, and approximately derive their applicable conditions.
- Based on their applicable conditions, we propose the DSVH mechanism, self-adaptive to network delays. The network-layer mechanism need not modify CNs and is compatible with various transport-layer protocols.
- With numerical analysis and test-bed experiments, we show that our DSVH mechanism has better performance than the SHORDER scheme and the CSH method.

The rest of this paper is organized as follows. In Section II, we describe the CSH and SHORDER approaches. Section III analyzes the effects of the CSH and SHORDER approaches on the handoff latency and the received data size of the TCP, and then discusses their applicable conditions. In Section IV, we propose our DSVH mechanism based on the applicable conditions. Section V presents our experimental results in a heterogeneous wireless testbed. In Section VI, we give an overview of the related work. Section VII concludes this paper.

II. SOFT VERTICAL HANDOFF APPROACHES

In this section, we introduce two types of soft vertical handoff approaches and discuss their advantages and disadvantages. One type produces out-of-order packets, and we takes the CSH method for instance. The other type causes no reordering packets, such as our SHORDER scheme.

A. The CSH Method

Chakravorty et al. improve the standard Mobile IPv6 (MIPv6) [5] mechanism, and present client-based soft handoffs [2] in the network layer to ensure no loss of packets destined to MNs during VHOs. In the CSH method, a client-based handoff

This work has been supported by the Beijing Nova Program and the National Natural Science Foundation of China (No. 60803140 and No. 60970133).

module is hooked to the MN's IPv6 stack to support soft handoffs, such that after every handoff, it allows all in-flight packets destined to the MN's previous interface to be read and be given to applications. Thus, this method keeps receiving packets from the previous network interface, while at the same time it allows for complete migration of IP points of attachment, before starting to send packets from the new interface. By this method, in-flight packets destined to the MN's previous network interface are not discarded. Nevertheless, the CSH method results in out-of-order packets during DVHOs. The source TCP enters the fast retransmit mode due to the duplicate acknowledgements (ACKs) that are generated by the MN after a DVHO.

B. The SHORDER Scheme

We have proposed the SHORDER scheme in [4] to avoid packet reordering in soft-handoff situations, in which the new access network has shorter delay than the previous access network. The SHORDER scheme utilizes binding acknowledgment (BA) messages which the home agent (HA) simultaneously transmits via the previous and new paths to the MN respectively. The time interval between the reception of the two BAs is used by the MN's network layer to buffer incoming packets from the new interface. Upon the reception of the BA from the previous interface, the MN orderly delivers the buffered packets to the transport layer. Accordingly, the MN's transport layer will not perceive the occurrence of out-of-order packets. Additionally, in order to correctly decapsulate in-flight packets with double IPv6 headers in the previous path, the MN and the HA maintain the tunnel between the MN's previous care-of address (CoA) and the HA for some time after the VHO is triggered. Due to the multi-homing characteristic, more than one CoA needs to be registered to the HA in the process of a soft handoff. We supplement an additional "S" flag in binding update (BU) message of the MIPv6 protocol. The "S" flag indicates simultaneous bindings as the Mobile IPv4 protocol [6]. If the "S" flag is set, the MN requests that the HA retains its prior mobility bindings. With respect to the signaling procedure and the MN / HA operation of the SHORDER scheme, the reader can be referred to [4] for more detailed description.

C. The Trade-off between Buffering Out-of-order Packets and Earlier Retransmission

If the new path delay between the CN and the MN is very short, the TCP is capable of rapidly retransmitting packets lagging in the previous path through the new path. In the case, the MN should not buffer incoming packets from the new interface. If there is a relatively long delay of the new path, the TCP may take quite long time to retransmit out-of-order packets. In the case, it is better to buffer packets from the new network. In sum, it is necessary to make a trade-off between buffering out-of-order packets and retransmitting lagging packets.

III. THE EFFECT OF SOFT VERTICAL HANDOFF APPROACHES ON TCP PERFORMANCE

In this section, we concentrate on analyzing the impacts of the CSH and SHORDER approaches on TCP performance.

A. Formalized Analysis

We focus on the downward vertical handoff process, and pay attention to the case that the CN transmits data to the MN. Initially, the MN sends a BU message to the HA to trigger a DVHO. The HA updates the MN's binding when receiving the BU at the instant t_h. At the same time, the data with the size of $B_P D_P$ are on the path from the HA to the MN's previous CoA. Then, the HA continues receiving packets from the CN, and forwards the packets via the new network to the MN. At the instant $t_h + D_N$, the packets start to successively reach the MN via the new network. At this time, the data with the size of $B_P(D_P - D_N)$ are still on the previous path from the HA to the MN. Thus, the packets destined to the MN's new CoA arrive at the MN earlier than some packets previously sent by the CN.

For the CSH method, duplicate ACKs are generated by the MN, and then are received by the CN at about $t_h + D_N + D'_N + D$. After that, the CN retransmits the data with the size of $B_P(D_P - D_N)$ via the new network, and then continues to send new data. Hence, after receiving the new data with the size of $B_P\,(D_N + D'_N + 2D)$ from the time $t_h + D_N$, the MN starts to receive the retransmitted packets. Since the instant that the CN sends the first retransmitted packets, it takes time of $D + D_N + [B_P\,(D_P - D_N) / B_N]$ for all retransmitted packets to reach the MN. After the DVHO has been completed, the MN keeps on receiving data at a high data rate in the new network. For a downlink TCP transfer, the latency of the DVHO from the time t_h to the arrival of the last retransmitted packet, is

$$\begin{aligned} t_{\mathrm{CSH}} &= (D_N + D'_N + D) + [D + D_N + B_P\,(D_P - D_N) / B_N] \\ &= D_P B_P / B_N + D_N\,(2 - B_P / B_N) + D'_N + 2D. \end{aligned} \tag{1}$$

In the duration, the size of the data received by the MN is

$$d_{\mathrm{CSH}} = B_P D_P + B_P\,(D_N + D'_N + 2D). \tag{2}$$

For the SHORDER scheme, the MN's network layer buffers the packets from the new interface, and does not pass them to the transport layer until it receives all the packets via the previous network. In this period, for the segments from the previous network, the MN ceaselessly returns their ACKs via the new network to the CN, so the retransmission timer does not expire at the source. There is no retransmission. Therefore, the latency of the DVHO from the instant t_h to the arrival of the last packet via the previous network is

$$t_{\mathrm{SHORDER}} = D_P. \tag{3}$$

In the period, the size of the data received by the MN is

$$d_{\mathrm{SHORDER}} = B_P D_P + B_P\,(D_P - D_N) = 2B_P D_P - B_P D_N. \tag{4}$$

B. Applicable Conditions

If (5) and (6) are both satisfied, the MN can receive more data by the SHORDER scheme.

$$t_{\mathrm{SHORDER}} < t_{\mathrm{CSH}} \tag{5}$$

TABLE I. PARAMETER VALUES

Parameter	Value	Parameter	Value	Parameter	Value
B_N	11 Mbps	B_P	384 Kbps [a]	D_P	100 ms

a. The downlink data rate of the ZXTR B30, one type of ZTE BSs, is 384 Kbps.

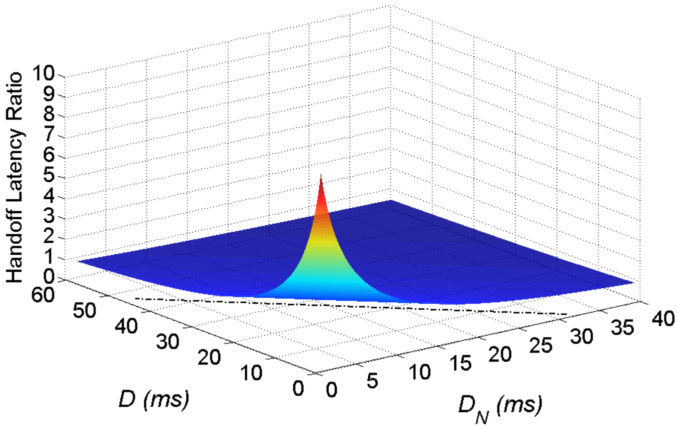


Figure 1. The ratio of TCP handoff latency with the SHORDER scheme to the CSH method (The dash-dot line represents projection of the contour line that the ratio equals 1)

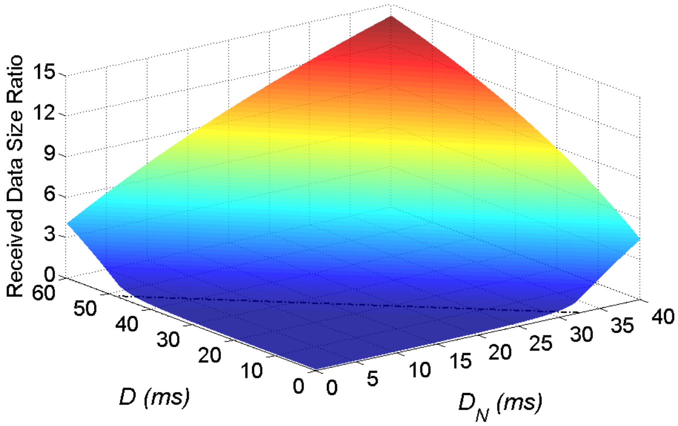


Figure 2. The ratio of received data size during the same period with the SHORDER scheme to the CSH method (The dash-dot line represents projection of the contour line that the ratio equals 1)

$$d_{\text{SHORDER}} + B_N (t_{\text{CSH}} - t_{\text{SHORDER}}) > d_{\text{CSH}} \quad (6)$$

By substituting (1) for t_{CSH} and (3) for t_{SHORDER} in (5), we can obtain

$$D_P < \frac{(2B_N - B_P)D_N + B_N D'_N + 2B_N D}{B_N - B_P} \quad (7)$$

Based on direct substitution from (1), (2), (3) and (4) in (6), we have

$$D_P < \frac{(2B_N - 3B_P)D_N + (B_N - B_P)D'_N + 2(B_N - B_P)D}{B_N - 2B_P} \quad (8)$$

Since $B_P \ll B_N$ and $D_N \approx D'_N$, (7) and (8) can both be simplified to (9) approximately.

TABLE II. PERFORMANCE PARAMETER VALUES

Wireless Technologies		Downlink Rate	RTT [b]
WWAN	3G LTE[7]	100 Mbps	10 ms
	IEEE 802.16e[8]	70 Mbps	25-40 ms
	CDMA2000 1xEV-DO Revision A	3.1 Mbps[9]	30 ms
	HSDPA[10]	14.4 Mbps	70 ms
	CDMA 1x	153.6 Kbps	300 ms
	GPRS	144 Kbps	800 ms
WLAN	IEEE 802.11n[11]	600 Mbps	1 ms
	IEEE 802.11a/g[12]	54 Mbps	2 ms
	IEEE 802.11b	11 Mbps	2 ms

b. In the case of no congestion

$$D_P < 3D_N + 2D \quad (9)$$

If (9) is satisfied, the SHORDER scheme improves the handoff latency and the the received data size during a DVHO, in comparison to the CSH method.

C. *Numerical Results*

Based on the above analysis, we adopt the parameter values in Table I to calculate the numerical results of the handoff latency and the received data size during a DVHO. Figure 1 displays the ratio of the DVHO latency with the SHORDER scheme to the CSH method for the TCP transfer from the CN to the MN. Figure 2 reveals the ratio of the received data size during the time of $\max\{t_{\text{CSH}}, t_{\text{SHORDER}}\}$ after t_h with the SHORDER scheme to the CSH method. As evident from the plots, when D and D_N are quite short, the DVHO latency is lower and the received data size is higher by the CSH method than by the SHORDER scheme. With D and D_N increasing, the CSH method gradually underperforms the SHORDER scheme in terms of the DVHO latency and the received data size of the TCP. The division represented by a dash-dot line denotes the cases that the two approaches have the same performance, and is close to the line of $3D_N + 2D = D_P$ as expected.

IV. A DIAGNOSIS-BASED SOFT VERTICAL HANDOFF MECHANISM

From the above analysis, it is known that the SHORDER scheme is superior to the CSH method if the inequality (9) holds. Otherwise, the SHORDER scheme is inferior to the CSH method. Whether the condition (9) is violated depends on the path delay between the HA and the MN / CN. According to the general performance of wireless networks in Table II, the condition (9) is often not satisfied for handoffs from CDMA 1x networks to IEEE 802.11 WLANs, while it is usually true for handoffs from 3G LTE networks to IEEE 802.11n WLANs. Hence, combining both approaches by deciding the applicable condition can improve the TCP performance during DVHOs.

A. *The Principle*

Assume that $D_N \approx D'_N$ and $D_P \approx D'_P$. We have $RTT_P \approx 2D_P + 2D$ and $RTT_N \approx 2D_N + 2D$. Thus, (9) can be converted to

$$RTT_P < 3RTT_N. \quad (10)$$

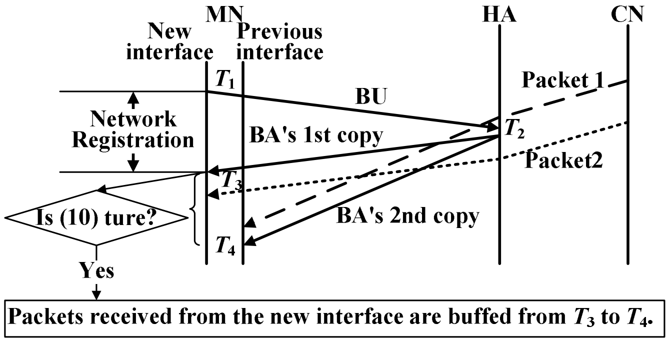


Figure 3. The DSVH mechanism

We need to estimate RTT_P and RTT_N. According to the MIPv6 protocol, whenever the MN moves to a new network or its registration expires, the MN sends a BU message to its HA and receives the corresponding BA message from the HA. The registration time is approximately the round-trip time of the current path between MN and HA. Therefore, $2D_P$ is the latency of the last registration before the MN moves to the new network. $2D_P$ can be computed as the average registration time if the MN updates its registration several times in the previous network. In the current registration process, the MN can estimate the registration time in the new network as $2D_N$, just after receiving the BA message through the new path. In addition, the TCP protocol specifies a timestamp option for round-trip time measurement (RTTM), so the value of RTT_P can be measured by the RTTM mechanism. Then, compute $RTT_N \approx RTT_P - 2D_P + 2D_N$. Thus, the MN can obtain the values of RTT_P and RTT_N in time, and can determine whether or not to buffer the packets received from the new network.

B. Mechanism Description

Based on the estimation method, we present the DSVH mechanism as shown in Figure 3. After estimating the values of RTT_P and RTT_N at the time T_3, the MN judges whether the inequality (10) is satisfied. If the inequality (10) holds, the MN continues to perform the DVHO according to the SHORDER scheme. Otherwise, the MN subsequently uses the CSH method. In detail, the signaling procedure and the MN / HA operation of the DSVH mechanism is as follows.

T_1: Having initiated a DVHO, the MN creates a local interface of an IPv6-in-IPv6 tunnel between its new CoA and the HA, but does not delete the local interface of the IPv6-in-IPv6 tunnel between its previous CoA and the HA. Then, it sends the BU message via the new network to the HA, with an additional “S” flag set to 1. After that, the MN continues sending packets via the previous network, and starts to buffer the packets received from the new network interface in a first-in-first-out (FIFO) queue in sequence.

T_2: On the arrival of the BU message with the “S” flag set to 1, the HA updates the binding cache entry for the MN, and creates a local interface of an IPv6-in-IPv6 tunnel between itself and the MN’s new CoA. However, it does not immediately delete the local interface of the IPv6-in-IPv6 tunnel to the MN’s previous CoA. The HA creates a timer with enough time. If the timer expires, it will delete the old tunnel interface. Furthermore, the HA stops forwarding packets from the CN via the old tunnel, and starts to forward them via the new tunnel. Then, it generates two copies of the BA message with the “S” flag set to 1, and sends one to the MN’s new CoA and the other to the MN’s previous CoA simultaneously.

T_3: If the delay from the HA to the MN via the new network is higher than via the previous network, there are no out-of-order packets. Otherwise, the BA message via the new network anticipates the arrival of the simulcasted BA message via the previous network to the MN. On the receipt of one copy of the BA message, the MN starts transmitting packets via the new network. At this time, if the inequality (10) does not hold, the MN no longer buffers the packets received from the new network interface.

T_4: When the other copy of the BA message reaches the MN, the MN deletes the local interface of the IPv6-in-IPv6 tunnel between its previous CoA and the HA. Moreover, if the queue is unempty, the MN orderly passes all the packets buffered in the queue to the transport layer, and does not buffer packets any longer.

TABLE III. PARAMETER VALUES

Parameter	Value	Parameter	Value	Parameter	Value
D_P	100 ms	D'_P	100 ms	D	20 ms
B_P	384 Kbps	B_N	11 Mbps	$D_N = D'_N$	1 - 40 ms

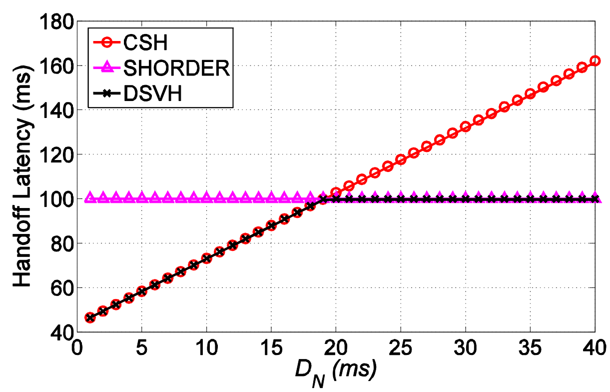


Figure 4. The impact of soft VHO approaches on TCP handoff latency

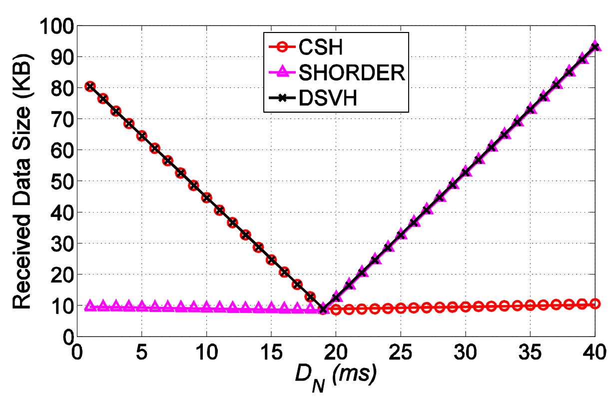


Figure 5. The impact of soft VHO approaches on TCP received data size

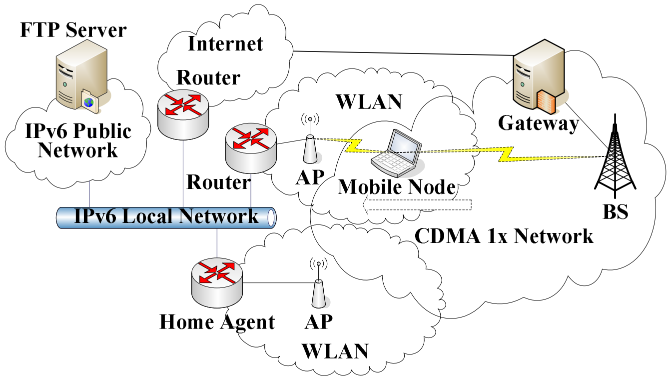


Figure 6. The Mobile IPv6-based WWAN-WLAN experimental test bed

C. Numerical Analysis

According to the parameter values in Table III, we evaluate the performance of the DSVH mechanism in terms of the handoff latency and the received data size from t_h to t_h+max{t_{CSH}, t_{SHORDER}} for the TCP. Numerical results are shown in Figure 4 and 5. From the figures, it can be seen that the performance of the DSVH mechanism is the same as the better one of the CSH and SHORDER approaches.

V. Experimental Results

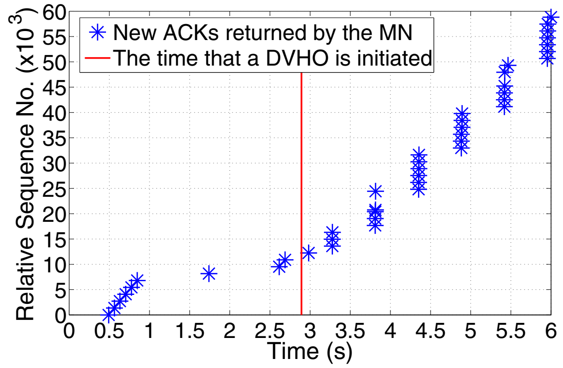
In order to demonstrate the feasibility and effectiveness of the DSVH method, we have implemented a prototype system. In this section, we observe the effects of the CSH, SHORDER and DSVH approaches on the handoff latency and the received data size of a TCP application by experiments.

Our experimental setup is shown in Figure 6. The mean RTT between the MN and its access routers in the WLAN and the CDMA 1x network is about 3ms and 400ms, respectively. Besides, the mean RTT between the router in the IPv6 local network and the IPv6 FTP server is nearly 35ms. In our experiments, the MN performs DVHOs by the CSH, SHORDER and DSVH approaches separately, while downloading files from the IPv6 FTP server.

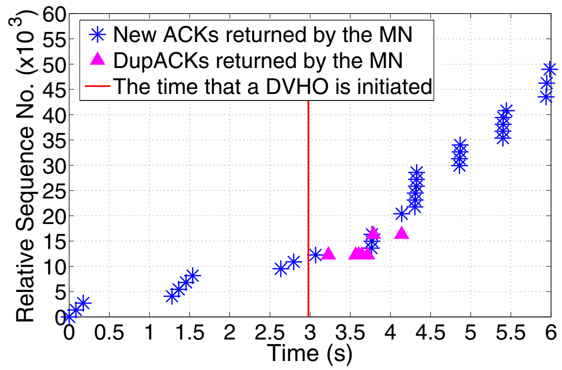
A. The Case of $RTT_N < RTT_P \leq 3 \cdot RTT_N$

In order to observe the impact of the DSVH mechanism on TCP performance when the condition (10) is true, we adjust the experimental scenario in Figure 6. A module for delaying packet forwarding is separately hooked to the IPv6 stacks of the HA and the foreign WLAN access router, such that the path delay between the HA and the MN's foreign WLAN router becomes 100ms, as well as the path delay between the HA and the FTP server increases by 150ms. Then, we conduct our experiment in the adjusted scenario.

Figure 7 depicts TCP performance based on the DSVH and CSH approaches respectively, when RTT_P is a little less than triple RTT_N. The handoff latency with the CSH method is 0.79s and the handoff latency with the DSVH mechanism is 0.38s. The received data size by the CSH method is 38.1KB and the received data size by the DSVH mechanism is 46.6KB within 3s after a DVHO is triggered. In this case, by the DSVH (or SHORDER) approach, the TCP sender avoids false congestion

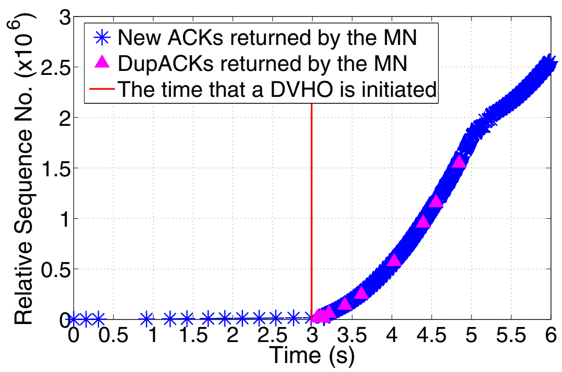


(a) The DSVH mechanism

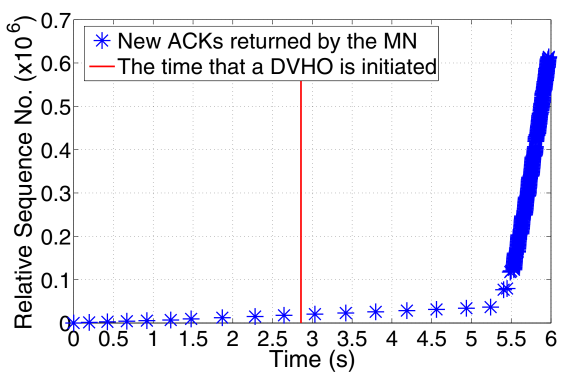


(b) The CSH method

Figure 7. The case that the inequality (10) is satisfied



(a) The DSVH mechanism



(b) The SHORDER scheme

Figure 8. The case that the inequality (10) is NOT satisfied

control and thus can achieves the higher performance, in comparison with the CSH method.

B. The Case of $RTT_P > 3 \cdot RTT_N$

For the case that the condition (10) is false, we can directly place the MN in our test bed in Figure 6. Figure 8 illustrates TCP performance based on the DSVH and SHORDER approaches respectively, when RTT_P is far more than triple RTT_N. The handoff latency with the SHORDER scheme is 2.55s and the handoff latency with the DSVH mechanism is 0.16s. The received data size by the SHORDER scheme is 0.46MB and the received data size by the DSVH mechanism is 2.54MB within 3s after a DVHO is triggered. In this scenario, with the DSVH (or CSH) approach compared to the SHORDER scheme, the MN finishes the DVHO more rapidly, and receives more data during the same period by earlier retransmitting data.

VI. Related Work

Recently, many researchers pay attention on the approaches against the side effect of out-of-order packets during DVHOs on TCP performance. In this section, we give an overview of the related work about these approaches.

In some proposals [13, 14, 15], when a handoff is impending, the receiver sends to the sender a cross-layer handoff notification regarding the changes in the access link characteristics. According to the notification, the sender selects proper values for TCP parameters. However, these approaches all require the modification of the TCP stack of the sender, so they are not easy to be put in practice. Hence, we focus on designing a solution without modification of the sender.

Other approaches try to develop mobile-receiver centric loosely coupled cross-layer designs. In [16], Hansmann et al. give the Nodupack approach which suppresses the transmission of duplicate ACKs during a handoff, when some conditions are met. In [17], Rutagemwa et al. also advance a proactive RTT Equalization approach to prevent false fast retransmit due to packet reordering during DVHOs, by equalizing the round-trip delay experienced by all packets. Nevertheless, these methods are not appropriate to the case that there is a Timestamps option in TCP segments for protecting against wrapped sequence numbers [18]. The Timestamps option is an important option and has been implemented in various Linux kernel versions. In this paper, our proposed network-layer soft vertical handoff approach is independent of the transport layer and is compatible with the TCP Timestamps option.

VII. Conclusions

Most existing soft handoff methods lead to plenty of out-of-order packets during downward vertical handoffs. We have presented a soft vertical handoff scheme without packet reordering, called SHORDER. In this paper, we analyze the impacts of our SHORDER scheme and another typical existing method on TCP performance in terms of the handoff latency and the received data size during a handoff. Based on analytical results, we propose a diagnosis-based soft vertical handoff mechanism, which can self-adaptively deal with out-of-order packets according to network delays. With experiments, we validate good performance of our DSVH mechanism. Furthermore, a match between experimental and analytical results is shown with respect to the effect on TCP performance.

References

[1] N. Nasser, A. Hasswa, and H. Hassanein, "Handoffs in Fourth Generation Heterogeneous Networks", IEEE Communications Magazine, vol. 44, no. 10, pp. 96 - 103, 2006.

[2] R. Chakravorty, P. Vidales, K. Subramanian, I. Pratt, and J. Crowcroft, "Performance Issues with Vertical Handovers -- Experiences from GPRS Cellular and WLAN Hot-spots Integration", IEEE 2nd Conference on Pervasive Computing and Communications (PerCom), Orlando, USA, pp. 155 - 164, Mar. 2004.

[3] D. Tandjaoui, N. Badache, H. Bettahar, A. Bouabdallah, and H. Seba, "Performance Enhancement of Smooth Handoff in Mobile IP by Reducing Packets Disorder", IEEE 8th International Symposium on Computers and Communications (ISCC), Antalya, Turkey, vol.1, pp. 149 - 154, Jun. - Jul. 2003.

[4] T. Shu, M. Liu, Z. Li, and K. Zheng, "Network-layer Soft Vertical Handoff Schemes without Packet Reordering", IEEE 34th Conference on Local Computer Networks (LCN), Zurich, Switzerland, pp. 285 - 288, Oct. 2009.

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

[6] C. Perkins, "IP Mobility Support for IPv4", RFC 3344, Aug. 2002.

[7] J. Liu, R. Love, K. Stewart, and M. E. Buckley, "Design and Analysis of LTE Physical Downlink Control Channel", IEEE 69th Vehicular Technology Conference, Barcelona, Spain, pp. 1 - 5, 2009.

[8] C. Ribeiro, "Bringing Wireless Access to the Automobile: A Comparison of Wi-Fi, WiMAX, MBWA, and 3G", the 21st annual Rensselaer at Hartford Computer Science Seminar, pp. 1 - 7, 2005.

[9] N. Bhushan, C. Lott, P. Black, R. Attar, Y. Jou, M. Fan, D. Ghosh, and J. Au, "CDMA2000 1xEV-DO Revision A: A Physical Layer and MAC Layer Overview", IEEE Communications Magazine, vol. 44, no. 2, pp. 75 - 87, 2006

[10] "HSDPA and beyond", http://faculty.stut.edu.tw/~cwycwy/951/N9490006.pdf

[11] E. Perahia, "IEEE 802.11n Development: History, Process, and Technology", IEEE Communications Magazine, vol. 46, no. 7, pp. 48 - 55, 2008

[12] IEEE 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 1999.

[13] L. Daniel, and M. Kojo, "Adapting TCP for Vertical Handoffs in Wireless Networks", IEEE 31st Conference on Local Computer Networks (LCN), Tampa, USA, pp. 151 - 158, Nov. 2006.

[14] L. Daniel, I. Järvinen, and M. Kojo, "Combating Packet Reordering in Vertical Handoff using Cross-layer Notifications to TCP", IEEE 4th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Avignon, France, pp. 297 - 303, Oct. 2008.

[15] N. C. Wang, Y. Y. Wang, and S. C. Chang, "A Fast Adaptive Congestion Control Scheme for Improving TCP Performance during Soft Vertical Handoff", IEEE Wireless Communications and Networking Conference (WCNC), Hong Kong, China, pp. 3641 - 3646, Mar. 2007.

[16] W. Hansmann, and M. Frank, "On Things to Happen During a TCP Handover", IEEE 28th Conference on Local Computer Networks (LCN), Bonn, Germany, pp. 109 - 118, Oct. 2003.

[17] H. Rutagemwa, S. Pack, X. Shen, and J. W. Mark, "Robust Cross-Layer Design of Wireless-Profiled TCP Mobile Receiver for Vertical Handover", IEEE Transactions on Vehicular Technology, vol. 56, no. 6, part 2, pp. 3899 - 3911, Nov. 2007.

[18] V. Jacobson, R. Braden, and D. Borman, "TCP Extensions for High Performance", RFC 1323, May 1992.