Henrik Petander, Optimizing localized mobility management for make-before-break handoffs, in Proceedings of NEWCOM-ACoRN Joint Workshop, Vienna, Austria, 20-22 September 2006.

OPTIMIZING LOCALIZED MOBILITY MANAGEMENT FOR MAKE-BEFORE-BREAK HANDOFFS

Henrik Petander

National ICT Australia,
Bay 15, Australian Technology Park,
1435 Eveleigh, NSW, Australia
Phone: +(61)2 8374 5540, fax: +61 2 8374 5579,
email:henrik.petander@nicta.com.au
web:www.nicta.com.au

Laboratory of Theoretical Computer Science,
Helsinki University of Technology,
Konemiehentie 2, 02150 Espoo, Finland
email:henrik.petander@hut.fi

ABSTRACT

Localized mobility management protocols are designed to improve the handoff performance of mobile nodes. Most of these protocols have been designed for mobile nodes able to connect to only one network at a time. However, emerging wireless technologies, such as mobile WiMAX, enable mobile nodes to perform make-before-break handoffs by connecting simultaneously to the old and new access network during the handoff. To maximally exploit this capability, localized mobility management protocols need to adjust their forwarding and signalling schemes during handoffs. This paper analyzes the performance of Fast Mobile IPv6 localized mobility management protocol for make-before-break handoffs. Based on the analysis a novel forwarding scheme is proposed using a combination of buffering and bicasting. The performance of the proposed protocol is analyzed and measured in a wireless test bed.

1. INTRODUCTION

Convergence of voice and data communications is leading towards a unified IP based communications infrastructure. In mobile communications this trend is creating a need for IP based mobility management for mobile users. Macro level IP mobility management protocols such as Mobile IPv6[1] allow a mobile node to remain reachable and retain existing communications in spite of mobility between multiple networks. Large handoff latencies of up to 2s per handoff inherent in the design of Mobile IPv6 and other macro level mobility management protocols make them unsuitable as a stand alone solution for frequently moving mobile users. A latency of this magnitude will lead to disruption of TCP connections and noticeable degradation of real time communications, such as Voice over IP. Localized mobility management schemes, such as Hierarchical Mobile IPv6[2] and Cellular IP[3], have been proposed to address the handoff latency. So far the most complete proposal is the IETF standardized Fast Handovers for Mobile IPv6[4] protocol (FMIPv6), which introduces a localized forwarding and context transfer framework for seamless handoffs.

Most IP mobility management schemes are designed for break-before-make handoffs, in which a mobile node disconnects from its previous access network before connecting to a new access network. Make-before-break handoffs offer a parallel way to localized mobility management to decrease the impact of handoffs. In a make-before-break handoff a mobile node connects to the new network before breaking off its connection with the previous network. Make-before-break handoffs have been used in cellular networks and have been standardized as a part of mobile WiMAX (IEEE 802.16e), an upcoming wireless Metropolitan Area Network protocol. Make-before-break handoffs can be also performed independently of the underlying access technology by using multiple network interfaces in a mobile node as proposed in [5]. Moreover, a mobile node moving between different network types, e.g. between WLAN and UMTS networks, would typically be able to perform a make-before-break handoff. A lossless make-before-break handoff can be achieved, if mobile node can retain the connectivity at its old access network until it finishes the handoff. The risk of packet loss due to early disconnection from previous access network can be minimized by reducing the handoff latency. For this reason, it is desirable to combine make-before-break handoffs with localized mobility management schemes.

This paper analyzes the applicability of the FMIPv6 localized mobility management framework for mobile nodes with the capability to perform make-before-break handoffs. Theoretical analysis of the performance of FMIPv6 with buffering and bicasting for make-before-break handoffs is presented in Section 2. Based on this analysis, a new protocol is proposed in Section 3 and experimental results on its performance are presented in Section 4.

2. BACKGROUND AND RELATED WORK

2.1 Fast Handovers for Mobile IPv6 protocol

FMIPv6 allows a mobile node to perform a predictive handoff from its *previous access router (pAR*) to a *new access router* (*nAR*), if it can anticipate the handoff event before disconnecting from pAR. In case mobile node loses connectivity with pAR before performing a predictive handoff, the mobile node will perform a reactive handoff after connecting to nAR. In both predictive and reactive mode, mobile node establishes forwarding from its *previous care-of address* (*pCoA*) on the link of pAR to its *new care-of address* (*nCOA*) on the link of the nAR. As a part of the handoff, pAR and nAR exchange state information for mobile node, such as quality-of service state, network access service state and security associations. This state transfer mitigates the need for mobile node and nAR to establish the state after mobile node has connected to nAR. The state transfer together with the localized forwarding scheme reduces the handoff latency even in the case of reactive handoff.

Predictive handoff in Fast Mobile IPv6, as shown in Figure 1, consists of the following steps:

1. Discovery of routers behind access points adjacent to the access point mobile node is connected to using proxy router solicitation and advertisement messages.
2. Initiation of handoff via pAR. Mobile node proposes a Care-of address at link of nAR in *Fast Binding Update* (FBU) and if nAR accepts the address (Handoff Acknowledge), pAR starts tunnelling packets destined to previous care-of address to new care-of address. During the link layer handoff of mobile node nAR receives the tunnelled packets and buffers them
3. Attachment to link of nAR. After performing a link layer handoff to nAR, mobile node sends a *Fast Neighbour Advertisement* to nAR. Upon receiving the message, nAR delivers any buffered packets to mobile node.

In reactive mode mobile node forms a new care-of address and sends a Fast Binding Update upon connecting to nAR. The pAR and nAR then perform context transfer and pAR starts tunnelling packets for previous care-of address to the new care-of address. In the case of a break-before-make handoff, packet loss would occur between disconnection from pAR to receiving of Fast Binding Update at pAR.

2.2 Performance of FMIPv6 with buffering for make-before-break handoffs

FMIPv6 predictive mode handoff enables a mobile node to perform a seamless handoff, if the buffering in nAR can hide the effects of the handoff from applications. With infinite buffer size, there are four factors which affect the effectiveness of buffering: 1) *Handoff latency*, i.e. how long the link layer handoff from pAR to nAR takes, 2) *Bandwidth of traffic,* 3) *Available bandwidth* on link of nAR and 4) *Delay sensitivity of application*. Firstly, if the handoff latency exceeds the delay threshold of a delay sensitive application, the application performance will be affected in spite of the buffering. Secondly, the emptying of the buffer creates a burst of traffic on the new link which, depending on the bandwidth of the link and traffic, may lead to delayed delivery of new packets. Thus, the use of buffering may have negative side-effects in spite of no packet loss.

To analyze the performance of FMIPv6 for make-before-break handoff capable mobile nodes in detail, the handoffs can be divided into four distinct cases:

1. Mobile node can perform a complete predictive make-before-break handoff, i.e. it can finish the attachment to nAR, before losing connectivity with pAR. In this case mobile node could receive all packets during the handoff period directly via pAR.
2. Mobile node can perform only a partial predictive make-before-break handoff: it loses connectivity with pAR after the start of the handoff but before attaching with nAR. Mobile node could receive packets at the start of the handoff directly via pAR, but would miss some packets due to early disconnect from link of pAR.
3. Mobile node performs a predictive break-before-make handoff: it loses connectivity with pAR at the start of the predictive handoff.
4. Mobile node cannot anticipate the handoff in time and performs a reactive break-before-make handoff after attaching to nAR.

In the first case, i.e. complete make-before-break handoff, the early establishment of tunnelling and buffering prevent mobile node from receiving packets directly via pAR, in spite of the connectivity. Use of reactive mode in this case would allow mobile node to receive packets directly at pCoA until the completion of the handoff at link of nAR. The first case is likely to be the most common one, since the handoff latency in FMIPv6 consists only of link layer handoff latency. However, use of buffering improves the handoff performance of the second and the third cases, since mobile node is able to receive all the packets from the buffer. In the fourth case, mobile node has no alternative to using reactive handoff mode and cannot take advantage of its make-before-break handoff capability.

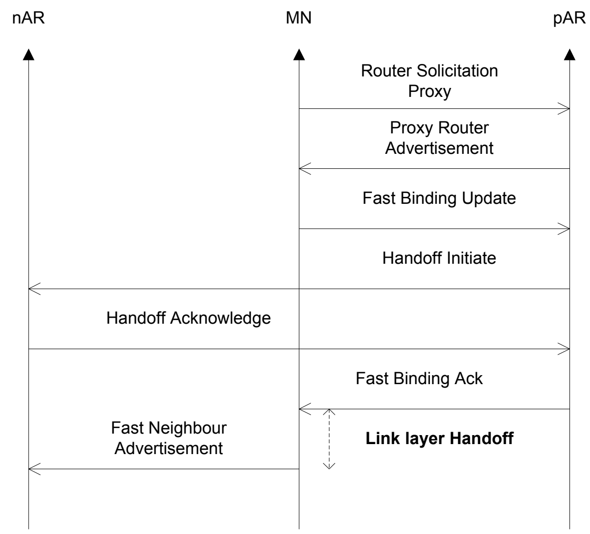

Figure 1. FMIPv6 signalling in predictive mode.

The performance of FMIPv6 predictive and reactive mode is summarized in Table 1. In order for a mobile node to get optimal performance in cases 1-3, it would need to be able to estimate in advance whether it will be able to perform a complete make-before-break handoff. In practice this is hard and optimistic mispredictions would lead to packet loss and pessimistic ones to unnecessary buffering.

Table 1. Performance of predictive and reactive mode.

Case	*Predictive mode*	*Reactive Mode*
1. Complete make-before-break handoff	Buffering is unnecessary.	No packet loss.
2. Semi-complete make-before-break handoff	Buffering is needed.	Packet loss occurs.
3. Break-before-make anticipated handoff	Buffering provides optimal performance.	High packet loss.
4. Unanticipated handoff	N/A.	Only option, high packet loss.

2.3 Bicasting and other alternative mechanisms

As an alternative to buffering at nAR, pAR can perform bicasting[6], and send packets to mobile node via two alternative routes. In the case of FMIPv6, pAR would bicast a copy of each packet it would tunnel directly to previous CoA of MN. In the case of a complete make-before-break handoff, mobile node would receive copies of packets at link of pAR until establishing connectivity at link of nAR. In case of incomplete make-before-break or break-before-make handoff, bicasting would still provide better performance than use of reactive handoff. This makes bicasting a desirable technique for enhancing handoff performance of make-before-break capable mobile nodes. However, bicasting has two inherent problems: A large data transmission overhead from duplication of the packets and a potentially negative impact on application performance from the duplication and reordering of packets at mobile node. Reordering is caused by mobile node receiving the bicasted packets at different times via the two routes due to the different network latencies on the two paths. Both packet reordering and duplication have been shown to have an adverse effect on TCP, which may mistake them to be results of congestion. In [7], TCP was shown to perform better with buffering than with bicasting due to these problems.

A form of bicasting, known as soft handoff, is already used in cellular networks. In a soft handoff, the data transmission overhead of bicasting is resolved by using a coding algorithm to divide the traffic between the multiple paths with some redundancy for error correction. However, the soft handoff mechanism can be only employed after mobile node has established connectivity with the new network, i.e. the handoff is completed. Before mobile node has attached to nAR, the bicasted traffic needs to be fully redundant. Thus, soft handoffs do not provide a solution to the problem of optimal packet forwarding during handoffs, but instead improve the performance immediately after the handoff.

S-MIP protocol[8] combines bicasting and buffering for a more optimal local forwarding solution for mobile nodes not able to perform make-before-break handoffs. However, the protocol does not solve the performance problems for make-before-break capable mobile nodes.

A new forwarding solution is needed in order for the packet delivery system to minimize packet loss and communications overhead regardless of the completion of the make-before-break handoff. To this end a new protocol which uses a combination of bicasting and buffering is proposed in the following section.

3. FMIPV6 BICASTING WITH SELECTIVE DELIVERY

This paper proposes a new protocol, FMIPv6 Bicasting with Selective Delivery (FMIPv6-BSD), for make-before-break handoffs. The FMIPv6-BSD protocol uses a combination of buffering and bicasting to provide improved handoff performance for mobile nodes capable of performing make-before-break handoffs.

3.1 Protocol description

The proposed FMIPv6-BSD protocol uses bicasting as proposed in [6]: pAR bicasts a copy of the packet directly to the pCoA, in addition to tunnelling packets to nCoA. The protocol operation is shown in Figure 2.

In contrast to the protocol in [6], the FMIPv6-BSD protocol minimizes the overhead of bicasting and minimizes packet duplication. The protocol introduces a counter value to packets forwarded by pAR to mobile node after the start of the handoff. The nAR sets the counter value in packets bicasted to nAR and directly to mobile node at the link of pAR. Mobile node records the counter values of the bicasted packets it receives at the old link.

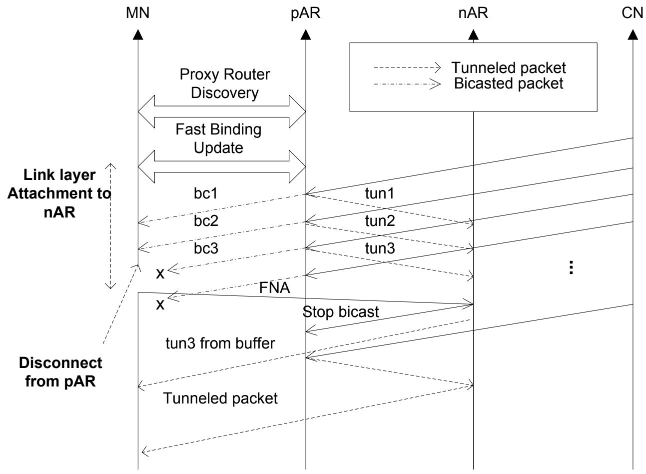

Figure 2. FMIPv6-BSD protocol operation for semi-complete make-before-break handoff with early disconnect from pAR

When attaching to nAR, mobile node includes a list of received packets in Fast Neighbour Advertisement to nAR. Based on this list, nAR can deliver packets from its buffer selectively, so that only the packets mobile node did not receive directly from pAR are delivered from the buffer. Upon receiving the Fast Neighbour Advertisement, nAR sends a Stop Bicast message to pAR to stop the bicasting of packets to the old link.

3.2 Protocol analysis

The proposed protocol does not affect the performance of reactive FMIPv6 handoffs, since by definition they are always break-before-make handoffs, as mobile node loses connectivity with pAR before it can initiate the handoff with nAR. Analysis of the three cases of predictive make-before-break handoffs is presented below:

1. Mobile node can finish the attachment to nAR, before losing connectivity with pAR. In this case, mobile node can receive all packets during the handoff period directly via pAR. When it connects to nAR, no delivery of the buffered packets is done and nAR flushes all duplicate packets.
2. Mobile node loses connectivity with pAR after the start of the handoff but before attaching to nAR. Mobile node receives packets at the start of the handoff directly via pAR, but misses some packets due to early disconnect from link of pAR. nAR delivers the missing packets from its buffer.
3. Mobile node loses connectivity with pAR at the start of the handoff. This case is identical to the standard handoff in FMIPv6 with buffering, as mobile node receives all packets from the buffer.

In the first case, the performance of selective bicasting with buffering is equivalent to the use of bicasting, but without the packet duplication and double data overhead on the wireless links. In the second case the amount of data received from buffer is: *buffered data = bandwidth of the traffic * (time of connection to nAR – time of disconnection from pAR)*. As the time between disconnection from pAR and connection to nAR increases, the performance approaches that of FMIPv6 performance. Thus, the worst case would be equivalent to normal FMIPv6 with buffering. The handoff performance of the protocol is analyzed empirically in the following section.

4. IMPLEMENTATION AND RESULTS

4.1 Implementation and test bed

An experimental prototype of the FMIPv6-BSD protocol was developed for Linux based on the fmipv6.org[9] FMIPv6 implementation. The fmipv6.org implementation was extended to support make-before-break handoffs using two network interfaces as described in [5] and the FMIPv6-BSD protocol.

In order to verify the performance of the proposed protocol a FMIPv6 test bed was built. The test bed consists of two access routers and a correspondent node connected together using a 100Mbits/s Ethernet switch.

The two access routers are Linux PCs running the extended FMIPv6 software. The routers acted as access points in managed mode and were set to use the same ESSID, but different channels (1 and 11) to minimize interference. Mobile node was a Linux laptop running the FMIPv6 software. The mobile node was equipped with two 802.11b interfaces, an integrated Intel IPW2100 card and a Prism 2.5 based PCMCIA card, to enable make-before-break handoffs. An external antenna was used with the PCMCIA card to minimize the interference from the two co-located cards. Standard Linux drivers without modifications were used for the WLAN cards and the 802.11b handoff latency was on average 200ms.

4.2 Experimental results and discussion

The performance of the proposed forwarding scheme was evaluated using the test bed presented in the previous section for fixed data rate 100kbits/s UDP and bulk data TCP traffic generated and measured with *iperf* [10] traffic generator. In all the experiments, the mobile node initiated a download from the correspondent node and performed a handoff. The handoffs for FMIPv6-BSD are complete make-before-break handoffs. Semi-complete make-before-break handoffs would show results which would vary between the standard FMIPv6 results and the FMIPv6-BSD results depending on the timing of the disconnection from pAR and the consequent amount of packets delivered from the buffer of nAR.

The performance of FMIPv6 and FMIPv6-BSD was evaluated using constant bit rate (100kbits/s) UDP stream in Figure 3. The FMIPv6-BSD handoff at t=4.2s does not affect the traffic noticeably. In the standard FMIPv6 handoff, the mobile node stops receiving packets from pAR at t=3.8s. The missing packets are delivered from the buffer of nAR at t=4s, after mobile node has attached to nAR.

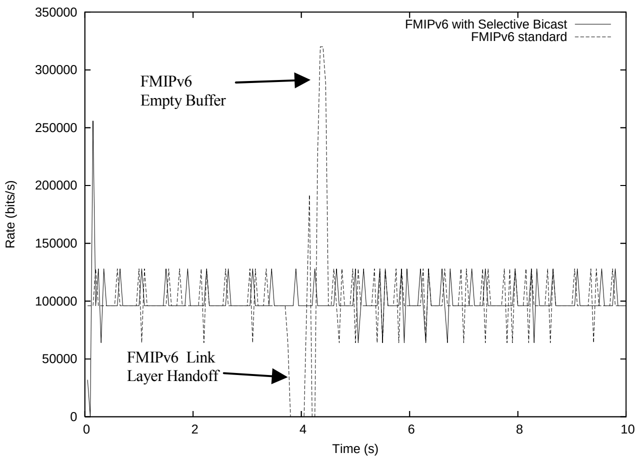

Figure 3. UDP handoff performance comparison.

The handoff performance for TCP traffic is shown in Figure 4 for the proposed scheme and for regular FMIPv6 with buffering. Both FMIPv6 and the FMIPv6-BSD handoffs have an impact on the TCP performance. FMIPv6 with buffering performs as expected based on the UDP experiment; the TCP progression stops at the start of the handoff and continues with some resending after it. The use of FMIPv6-BSD causes mobile node to receive duplicate packets between t=4.4s and t=4.5s. This is due to the delay between mobile node attaching to nAR and pAR receiving the Stop bicast message. The TCP transfer progresses only after the duplicate packets have been handled which limits the performance of the scheme. This may be due to TCP congestion control mechanisms. A possible solution to this would be dropping of the duplicates in mobile node before they are processed by TCP. In spite of this impairment, the use of FMIPv6 with selective bicasting reduced the impact of the handoff on TCP progress by approximately 48%, when compared to FMIPv6 with buffering.

The performance of the proposed forwarding scheme is dependent on the ability to predict handoffs accurately. With accurate handoff timing make-before-break handoffs can be completed with negligible impact on on-going communications. With less accurate timing, handoffs will degenerate into semi complete make-before-break handoffs or break-before-make handoffs. However, even the worst case performance is similar to FMIPv6 without the modifications, and in the more common cases the performance is improved.

5. CONCLUSIONS AND FURTHER WORK

In this paper, the performance of FMIPv6 was analyzed for mobile nodes with the make-before-break handoff capability and it was shown that the use of buffering does not provide optimal performance. A new localized forwarding protocol combining bicasting and selective buffering was proposed to solve the observed performance limitations. It was shown that the proposed protocol improves the handoff performance significantly in the most common cases without increasing the signalling or data transmission overhead of FMIPv6 with buffering.

The performance of the proposed FMIPv6-BSD protocol could be further improved by reducing the number of duplicate packets received by the mobile node. In addition to improving the handoff performance, this would also reduce the overheads of the protocol.

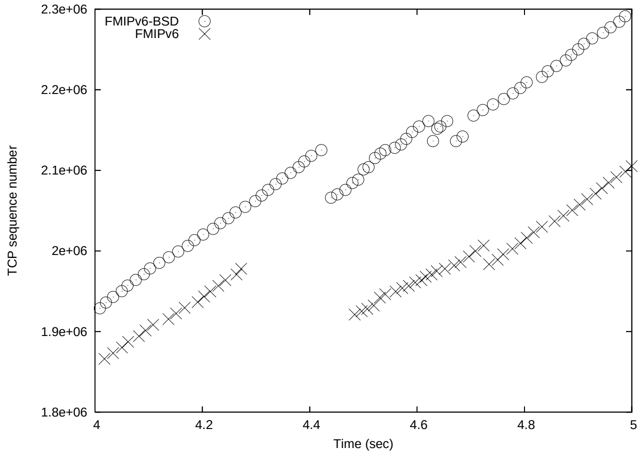

Figure 4. Effect of handoff on TCP sequence number progression.

6. ACKNOWLEDGMENTS

Thanks to Eranga Perera and Aruna Seneviratne for their helpful comments on this paper and the reviewers on their constructive feedback.

REFERENCES

[1] D. Johnson, C. Perkins, and J. Arkko, “Mobility Support in IPv6”, RFC3775, *Internet Engineering Task Force,* June 2004.

[2] H. Soliman, C. Castelluccia, K. El-Malki, and L. Bellier, “Hierarchical Mipv6 Mobility Management”, draft-ietf-mipshop-hmipv6-04.txt, *Internet Engineering Task Force,* July 2002

[3] A. Campbell, J. Gomez, S. Kim, and C. Wan, “Comparison of ip micro-mobility protocols”, *in IEEE Wireless Communications*, pp. 72-82, February 2002.

[4] R.Koodli (editor), "Fast Handovers for Mobile IPv6", RFC 4068, *Internet Engineering Taskforce*, 2005

[5] H. Petander, E. Perera, K-C. Lan, and A. Seneviratne, “Measuring and improving performance of network mobility management in IPv6 network”, in *IEEE Journal on Selected Areas in Communications special issue on Mobile Routers And Network Mobility.,* in press

[6] K. Malki, and H. Soliman, “Simultaneous Bindings for Mobile IPv6 Fast Handoffs”, draft-elmalki-mobileip-bicasting-v6-02.txt, *Internet Engineering Task Force,* June 2002. Work in Progress.

[7] G. Bhaskara, and A. Helmy, “TCP over Micro-mobility Protocols: Systematic Ripple Effect Analysis”, in proceedings of *IEEE Vehicular Technology Conference (VTC)*, Vol. 5, pp. 3095 - 3099, Sept 2004.

[8] R. Hsieh, and A. Seneviratne, “A Comparison for Improving Mobile IP handoff Latency for End-to-End TCP”, In *Proceedings of Mobicom ’03: the 9th annual international conference of Mobile Computing and networking*, San Diego, USA, pp. 29-41, 2003

[9] fmipv6.org implementation of FMIPv6, http://www.fmipv6.org

[10] Iperf traffic generator and measurement tool, http://dast.nlanr.net/Projects/Iperf/.