Performance comparison of end-to-end mobility management protocols for TCP

Peer Azmat Shah\textsuperscript{a,c,d,*}, Muhammad Yousaf\textsuperscript{b}, Amir Qayyum\textsuperscript{c}, Halabi B. Hasbullah\textsuperscript{a}

\textsuperscript{a} Department of CIS, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia
\textsuperscript{b} Faculty of Computing, Riphah International University (RIU), Islamabad, Pakistan
\textsuperscript{c} Center of Research in Networks & Telecom (CoReNet), M.A. Jinnah University, Islamabad, Pakistan
\textsuperscript{d} Department of Computer Science, COMSATS Institute of Information Technology, Pakistan

Abstract

Mobility management for TCP connections has been discussed in literature since last several years. Many mobility management solutions have been proposed including end-to-end solutions and infrastructure dependent solutions. These solutions operate at different layers of TCP/IP protocol stack. End-to-end mobility management solutions have the advantage of not requiring the deployment of additional entities in network infrastructure to handle mobility management as compared to the infrastructure dependant solutions. Performance analysis of these end-to-end mobility management protocols seems missing in literature. In this paper, we analyze the performance of different end-to-end mobility management protocols using mathematical modeling. Evaluation has been carried out for evaluating the handover delay, throughput degradation time and protocol overhead. Some security and implementation issues pertaining to these protocols have also been discussed. Results show that protocols that allow simultaneous communication through multiple network interfaces provide throughput stability during vertical handovers, thus facilitating smooth handovers.

© 2012 Elsevier Ltd. All rights reserved.

Contents

1. Introduction ................................................................. 2
2. Related work ............................................................. 3
3. End-to-end mobility management protocols for TCP ............... 3
   3.1. HIP based mobility management. .......................... 3
   3.2. TCP-redirection. .................................................. 3
   3.3. Socketless-TCP (SL-TCP). ................................. 3
   3.4. TCP-Migrate ...................................................... 4
   3.5. pTCP ............................................................... 4
   3.6. Tsukamoto’s proposal ......................................... 4
   3.7. LCT-based handoff protocol .............................. 4
   3.8. VA-TCP .......................................................... 4
   3.9. Session layer mobility management (SLM). ............. 4
   3.10. End-to-end mobility management framework (EMF) ..... 4
4. Performance evaluation model for handover delay .................. 5
   4.1. Vertical handover scenarios ................................. 5
   4.2. Modeling end-to-end handover delay .................... 6
   4.3. Results comparison for handover delay .................. 8
5. Performance evaluation model for throughput degradation time ...... 10
   5.1. Modeling throughput degradation time after vertical handover .... 10
   5.2. Results comparison for throughput degradation time .......... 11

\* Corresponding author at: Department of CIS, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia. Tel.: +92 321 582 2507.
E-mail addresses: peer.azmat@yahoo.co.uk, peer.azmat@comsats.edu.pk (P.A. Shah).

1084-8045/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2012.05.002

Please cite this article as: Shah PA, et al. Performance comparison of end-to-end mobility management protocols for TCP. Journal of Network and Computer Applications (2012), http://dx.doi.org/10.1016/j.jnca.2012.05.002
1. Introduction

Mobile Internet applications have become popular and are being used widely. With the evolution of 4G wireless heterogeneous networks, mobile devices nowadays are equipped with multiple network interfaces, e.g., WiFi, WiMAX, UMTS, etc. Capabilities of these heterogeneous access networks vary largely with respect to data rate, coverage area and cost. Users having multi-homed mobile devices like to use the best access network according to the network capabilities and their own preferences. Heterogeneous wireless networks mostly use a common IP core network. More than 90% of Internet traffic uses Transmission Control Protocol (TCP) as reliable transport layer protocol. However, whenever a mobile device having Internet connectivity moves from one access network to another, the TCP 4-tuple is disturbed. This results in service disruption for the established connection with remote node in the network.

To handle this problem of service disruption, variety of mobility management protocols have been proposed in literature. These protocols operate at different layers of TCP/IP protocol stack. For example, mobility has been handled at Link layer, Network layer (Bhagwat et al., 1996), layer 3.5 (Host Identity Protocol) (Moskowitz and Nikander, 2006), Transport layer (Atiquzzaman and Reaz, 2005) and at Session layer (Landfeldt et al., 1999; Banerjee et al., 2006). At each of these layers, mobility management schemes have corresponding strengths and weaknesses.

Link layer mobility management is used for handover within the same link layer technology hence providing only horizontal handover. As our current focus is on the mobility of a node in heterogeneous environment that requires the vertical handover capabilities therefore, we are not considering link layer mobility solutions in this paper.

Network layer mobility management solutions have following shortcomings (Eddy, 2004):

- Deployment of additional network entities (Home Agent, Foreign Agent, Mobility Anchor Points, etc.) is required in network infrastructure. From the perspective of end users, the requirement of infrastructure support may be the largest problem, as end users suffer in places where network administrators have not provided support for Mobile IP (Perkins, 1996) due to additional deployment cost.
- Mobile Node (MN) always uses a static address as its home address in all IP packets. After mobility, when MN moves out from home network, routers and firewalls may assume that the node is spoofing its address as part of some attack, rather than due to mobility, and block its packets. Thus, using Network layer mobility management in an attempt to maintain connectivity may instead cause a loss of connectivity.
- The interface between IP and transport protocols is not enough for upper layers to be notified when mobility is taking place. After a movement across networks, TCP's roundtrip time and congestion window estimates may be invalid and need to be reset.
- Several steps are involved in a Mobile IP (Perkins, 1996) handover, e.g., detection of motion, finding new networks, updation of HA with its current location and performing authentication procedures at each step. While this transition is in process, the MN may be completely disconnected and the network will lose packets that are destined for the MN.

Transport layer mobility management techniques have been characterized into two main types: end-to-end and proxy based. Proxy based transport layer solutions like MOSCKS (Maltz and Bhagwat, 1998), etc. require additional network entities and face the same limitations as network layer approaches do. End-to-end mobility management techniques require no changes in the network infrastructure and mobility is handled only by the two end nodes. Therefore, these schemes are considered strongest, despite the fact that they require changes to the existing protocols like TCP. As cost of deploying additional network entities is not involved in end-to-end solutions thus end users are facilitated without having any dependence on network operators to deploy mobility management solutions. Some other characteristics of end-to-end mobility management solutions are:

- No need to maintain states in the network (like tunnel states in Mobile IP).
- Direct communication between two communicating nodes is done thus avoiding packet redirection.
- Low handover delay is involved.
- Authentication remains simple.

At transport layer, SCTP based end-to-end mobility management solutions (Koh et al., 2004) have also been discussed in literature but as they do not facilitate TCP traffic therefore, in this paper we are not considering the SCTP based solutions.

Similarly, Session layer mobility management approaches (Landfeldt et al., 1999; Banerjee et al., 2006) are also there. Some of them are SIP based solutions, like Banerjee et al. (2006) and used only for real time multimedia applications. They do not facilitate TCP traffic; therefore, in this paper we are not considering such solutions.
Rest of the paper is organized as follows: In Section 2, related work is discussed. In Section 3, a description of different end-to-end mobility management protocols for TCP is provided. Sections 4–6 present the performance models and results for handover delay, throughput degradation time and protocol overhead. In Sections 7 and 8, analysis of security and implementation issues has been provided. Section 9 discusses the qualitative comparison of different end-to-end mobility management schemes and in the end; Section 10 concludes the paper with future directions.

2. Related work

Performance analysis of some of the mobility management protocols has been performed in varying contexts in the literature, e.g., Atiquzzaman and Reaz (2005) has analyzed the transport layer mobility management solutions, however it is only a survey paper that has identified the pros and cons of different Transport layer mobility management solutions. It has made classification of protocols on the basis of their characteristics and behavior during the handover execution. There is no performance analysis made in this work. Mohanty and Akyildiz (2007) has identified mobility management solutions at different layers of TCP/IP protocol stack and derived an analytical model for the performance analysis of Mobile IP, TCP-Migrate and SIP. The focus of this work is to compare the performance of Mobile IP with TCP-Migrate and SIP. Zeedally and Siddiqui (2007) discussed an empirical analysis of SIP, Mobile IP and SCTP. A comparison of host mobility for IP networks using TCP-Migrate, HIP and Mobile IP is done in Henderson et al. (2003) on the basis of security, deployment, scalability and robustness, but there is no performance analysis, rather a comparison of these protocols is done. Also, focus of this research is to highlight the weaknesses of Mobile IP as compared to HIP and TCP-Migrate. In Diab et al. (2007) a generic mathematical model is proposed for performance analysis of mobility management protocols but the focus of this model is also Mobile IP and it does not discuss the end-to-end solutions. Murthy and Phiri (2005) have discussed handover latency in WLAN/GPRS inter-working system and identified the parameters that affect the handover latency. A survey of mobile management protocols has been carried out in Shah and Yousaf (2008) and Al-Surmi et al. (2012).

As performance analysis of end-to-end mobility management protocols seems missing in the literature, there is a need to analyze different end-to-end protocols for mobility management. In this paper, we have discussed some scenarios where a MN moves across heterogeneous access networks. We have discussed performance metrics like handover delay, throughput degradation time, protocol overhead, security and implementation issues, to analyze the end-to-end mobility management solutions.

3. End-to-end mobility management protocols for TCP

To manage the mobility in end-to-end way, various mobility management protocols have been proposed, operating at different layers of TCP/IP protocol stack. These solutions eliminate the need for making changes in network infrastructure and tunneling of data packets. The protocols covered in this paper for performance evaluation are shown in Fig. 1. Protocols are grouped on the basis of TCP/IP protocol stack layer at which they operate.

3.1. HIP based mobility management

In current Internet, IP address is used for location identification as well as for end point identification. Host Identity Protocol (HIP) (Moskowitz and Nikander, 2006; So and Wang, 2006) separates the dual role of IP address as an end point identifier and as a location identifier. It introduces, a Host Identifier (HI) used as end point identifier at transport layer and IP address is used as location identifier at network layer. This is termed as a new layer, Host Identity layer, which is seen as layer 3.5 in TCP/IP protocol stack in Fig. 1.

Host Identity Protocol introduces a new network entity rendezvous server for location management and simultaneous mobility support.

3.2. TCP-redirection

For the continuity of existing TCP connection on a Mobile Node, TCP-Redirection (TCP-R) (Funato et al., 1997) revises the pair of IP addresses and port numbers of the connection. Whenever there is a change in the MN’s IP address, MN sends Redirect message to Remote Node (RN). On receiving the Redirect message, RN sends Authentication request to MN. MN replies with connection Authentication reply message. RN checks the authentication reply message and if found correct, then it revises the pair of addresses of existing TCP connection. In the meanwhile, MN also revises its pair of addresses. DNS is proposed for location management in this protocol.

3.3. Socketless-TCP (SL-TCP)

Socketless-TCP (SL-TCP) (Beda and Ventura, 2005) is not a complete mobility management protocol rather it only provides services for vertical handover execution. No details are given for location management but DNS dynamic updates can be used for this purpose. The protocol introduces a unique number as Connection Identifier (CID) at both ends separately. CID is a combination of local port & destination IP and does not depend on the physical attachment of MN in the network; hence it remains unique for the duration of the communication session on MN. CID depends upon the location of RN in the network; hence whenever MN moves across heterogeneous networks and obtains a new IP address then CID at RN is updated. SL-TCP performs its own connection initialization, connection management and connection migration. These functionalities are implemented within the
transport layer, as a modification to the TCP (Beda and Ventura, 2005).

When a MN moves and changes its point of attachment (IP address), then SL-TCP connection migration depends upon the role of the end node. In case of MN, it has the correct destination IP address of RN within its CID mapping, thus outgoing packets will be addressed correctly. In case of RN, whenever a packet is received from network layer, SL-TCP retrieves the source IP address from network layer header, and extracts the CID from transport layer header. It then checks the CID to (local port, destination IP) mapping and compares the source address obtained from the network layer with the destination IP, if the address is different, it updates the destination IP address in CID, otherwise no action is taken. Packets are then forwarded to respective application via the applications port number.

3.4. TCP-Migrate

In TCP-Migrate (Snoeren and Balakrishnan, 2000), for connection migration, authors have proposed a Migrate option for TCP that allows an existing TCP connection to be migrated by either host from an old IP address to a new IP address. This TCP connection migration is done by exchanging proposed TCP SYN with Migrate option. In Migrate option, a token (identifier) is included that identifies the previously established connection. This token is negotiated during initial connection establishment using migrate-permitted option. Location management is done through DNS dynamic updates. Simultaneous mobility management support was not available in basic TCP-Migrate, however, Wu et al. (2007) proposed simultaneous mobility support feature for TCP-Migrate through DNS assistance.

3.5. pTCP

pTCP (Hsieh et al., 2004) is a transport layer solution for multi-homed mobile devices that provides mobility and bandwidth aggregation services. It creates and maintains one TCP state for each active network interface. TCP-virtual (TCP-v) pipes are created for each active interface in a pTCP connection. Whenever a pTCP socket is opened by application for a connection, a TCP-v pipe for that connection using active interface is created. When application on the MN opens a pTCP socket with RN, pTCP creates the first TCP-v pipe. This pTCP with only one pipe behaves similarly as traditional TCP connection. When MN moves to a new network and gets a new IP address then pTCP creates a new pipe but does not closes the previously established pipe.

3.6. Tsukamoto’s proposal

Tsukamoto proposed a new mobility management scheme (Tsukamoto et al., 2007; Tsukamoto and Hori, 2004) that modifies the transport protocol for handling multiple connections during handover process. A cross layer approach is used that detects any change in the status of wireless networks and selects the optimal interface. A Handover Manager (HM) is implemented at transport layer that performs the vertical handover. Link layer information is sent to the HM at transport layer, bypassing the intermediate layers using cross layer manager. Using this information HM can detect a link-up or link-down state. During communication, if HM detects a link-up state for a new wireless network, then it starts a new TCP connection via that interface and starts parallel transmission using two TCP connections. For each connection, end-to-end performance measures, such as Bandwidth, Bandwidth Delay product or bottleneck bandwidth are calculated and compared, using first few packets in the start-up period (slow start). On the basis of this comparison, handover is initiated or ignored.

3.7. LCT-based handoff protocol

To maintain connection continuity Local Connection Translation (LCT) based handoff protocol (Guo et al., 2004) introduces two components, Connection Manager (CM) and Virtual Connectivity (VC). Responsibility of CM includes the detection of link layer conditions, like signal strength, and network conditions like available bandwidth and end-to-end delay, etc. VC is responsible for maintaining connection continuity in case of mobility. VC has an internal component LCT, which maintains mapping relationship between the original connection information and the current connection information for each active connection, thus making the mobility transparent to applications. Location management is done through DNS. In order to support mobility, proposed protocol introduces Subscription/Notification (S/N) service at application layer. To support these two functionalities, an additional entity, S/N server, is required in the network, implementing the subscription/notification functionalities. An S/N client is also required at each end node to communicate with the S/N server using S/N protocol.

VC handles the mobility by sending a Connection Update (CU) message to RN. Connection Update ACK (CUA) is an acknowledgment message for a received CU when RN trusts that the MN is really at its claimed current address carried in the CU. If RN does not trust the CU message, then it sends a Connection Update Challenge (CUC), which is acknowledged by MN using Connection update Challenge Response (CCR) message.

3.8. VA-TCP

Vertical handoff Aware-TCP (VA-TCP) (Lin and Chang, 2007) is an end-to-end mobility management scheme that uses the abstraction layer introduced by LCT-based handoff protocol to maintain the application transparency. For connection continuity, this protocol adds some new messages to support vertical handover. When a MN roams in a new network, then VA-TCP dynamically estimates the connection parameters like bandwidth, delay and the bandwidth-delay product from the information provided by the lower layers. MN sends a notification message to RN telling that it has changed its access network and IP address. RN updates the connection parameters. In the meanwhile MN also updates the connection parameters.

3.9. Session layer mobility management (SLM)

Session Layer mobility Management (SLM) (Landfeldt et al., 1999) is a framework that supports mobility management by introducing Session layer between Transport and Application layers. This layer operates above TCP and switches data streams between multiple connections. The end-to-end network path is divided into three separate paths. Two paths are between the applications on the two hosts and socket connectors and the third path between the socket connections of two hosts. Location management is done through a proposed User Location Server (ULS).

3.10. End-to-end mobility management framework (EMF)

End-to-end mobility management framework (EMF) is a Session layer mobility management protocol that enables the multi-homed mobile devices to roam across heterogeneous wired/wireless networks for service continuation (Yousaf and Qayyum, 2008; Yousaf et al., 2011). It uses Association ID (AID) as session identifier. Vertical handover is performed after establishing a new TCP connection from the new network interface and then exchanging handover request and handover confirm messages.
This protocol also supports simultaneous mobility, willful hand-over and has intelligence of IEEE 802.21 MIH link going down prediction. These features distinguish EMF from the rest of end-to-end mobility management protocols that has been discussed earlier.

As there are many proposed protocols, therefore before practically implementing each proposed technique and then checking its performance, there is a need to evaluate the performance of these protocols via some low cost methods. One such method is to do performance evaluation using mathematical modeling. Table 1 shows the current status of these protocols. As simulation patches or real world implementation of most of these protocols is not available, therefore, performance evaluation based on actual implementation or via simulation methods is not possible at this stage.

This paper is the first attempt of our long-term efforts to evaluate the performance of end-to-end mobility management protocols. Our intention is to first perform some mathematical analysis of these protocols and then on the basis of this analysis recommend that which protocols are needed to be implemented due to their good performance in different scenarios.

In next sections, performance evaluation models for handover delay, throughput degradation time, protocol overhead, security and implementation has been discussed.

4. Performance evaluation model for handover delay

In this section, handover delay of end-to-end mobility management protocols in different vertical handover scenarios is discussed. Different mathematical modeling techniques of performance evaluation are used and the handover delay is analyzed using common metrics.

4.1. Vertical handover scenarios

Consider a multi-homed Mobile Node (MN) with two network interfaces. MN can move across overlapping as well as non-overlapping access network coverage areas, as shown in Fig. 2.

Scenario 1: Movement of MN in non-overlapping regions

When MN moves out of one network’s coverage area and enters into the coverage area of another access network, connectivity of MN in old access network is no more active. In this scenario, MN is connected to single access network at a time, shown as movement scenario 1 in Fig. 2. Here break-before-make handover is performed.

Scenario 2: Movement of MN in overlapping regions

In this scenario, MN may enter into the overlapping region or may leave the overlapped region of two different access networks shown as movement scenarios 2 and 3 respectively in Fig. 2.

In movement scenario 2, MN moves into the coverage area of new access network while remaining in the coverage area of old access network. Here, MN is accessible through two different access networks.

In movement scenario 3, MN moves out of the overlapping region and looses connectivity to the network through which it was connected earlier. Here MN can perform a make-before-break handover, if the mobility management protocol has the intelligence to detect a link going down event.

Figures 3 and 4 show timing diagram of handover delay for scenario 1 when MN is receiving and sending data respectively. Here, MN moves between non-overlapping regions of two different access networks. It loses its connectivity in the old access network and becomes unable to send/receive data until it moves into the coverage area of a new access network and performs vertical handover. In these figures, $T_{12}$ is the time required for layer 2 connectivity, $T_{IP}$ is the time required to acquire new IP address, $T_{seg}$ is the time required for the signaling of handover control messages between two communicating nodes and $T_{d}$ is the one way delay for data packets to travel between MN and RN in new access network.

In scenario 2, when MN enters into the overlapping region of two access networks, the handover delay depends upon the nature of mobility management protocol. Some mobility solutions stop sending/receiving data through old network interface, after MN moves into the coverage area of new access network and perform the same procedure as discussed above for scenario 1. If the mobility management protocol allows simultaneous communication through two different interfaces, then MN will be receiving data through old network while it is performing layer 2 connectivity and then exchanging handover control messages (connection migration/new connection establishment) with RN through new network interface. Here, MN will not suffer from any

Table 1
Current status of end-to-end mobility management protocols.

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>HIP</th>
<th>TCP-R</th>
<th>SL-TCP</th>
<th>TCP-Migrate</th>
<th>pTCP</th>
<th>Tsukamoto's proposal</th>
<th>LCT-based</th>
<th>VA-TCP</th>
<th>SLM</th>
<th>EMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citation count</td>
<td>74</td>
<td>84</td>
<td>2</td>
<td>619</td>
<td>33</td>
<td>14 and 6</td>
<td>153</td>
<td>7</td>
<td>61</td>
<td>9</td>
</tr>
<tr>
<td>Simulation patches</td>
<td>OMNeT++</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
</tr>
<tr>
<td>Implementation</td>
<td>Available for Linux</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Available for Linux</td>
<td>Not available</td>
<td>Not available</td>
<td>Not available</td>
<td>Available for Linux</td>
<td></td>
</tr>
</tbody>
</table>

Please cite this article as: Shah PA, et al. Performance comparison of end-to-end mobility management protocols for TCP. Journal of Network and Computer Applications (2012), http://dx.doi.org/10.1016/j.jnca.2012.05.002
handover delay because MN is still receiving data through old network interface while it is performing handover signaling through new network interface.

In all these scenarios for simplicity, handover signaling between communicating nodes is considered as three-way message exchange. However, different end-to-end mobility management protocols exchange different number of control messages for handover signaling, thus their handover delay will be different.

4.2. Modeling end-to-end handover delay

Handover delay of end-to-end mobility management protocols is the time between receiving the last packet in old access network and the first packet in new access network. It is the sum of all individual delays involved in handover. Usually handover includes the following delay components:

a. Layer 2 connectivity delay ($T_{L2}$)
b. IP address acquisition delay in new network ($T_W$)
c. Handover control messages signaling delay ($T_{sig}$)
d. One-way delay for data packet in new network ($T_d$)

a. Layer 2 connectivity delay:

Link layer connectivity delay ($T_{L2}$) is the time required to establish layer2 connectivity between a mobile node and an Access Point (AP) or Base Station (BS) of new access network. For example for WLAN, it includes the time spent in scanning for available APs, selecting appropriate channel, authenticating and then associating with one of the APs. AP broadcasts the
Manitover delay is the one-way delay between communicating nodes (Kim and Koh 2008). According to this model, signaling delay of some mobility management protocols, we consider the values of TL2 and TIP in new access network.

**c. Handover control messages signaling delay:**

After making decision for handover and getting new IP address, mobility management protocols either migrate the old connection or establish a new TCP connection and exchange handover control messages, depending upon the mobility management solution. Time consumed to exchange these control messages is referred to as handover control messages signaling delay (Tsig).

**d. One-way delay for data packet in new network:**

It is the one-way delay between MN and RN in new access network. After handover control messages signaling, RN sends first data packet to the MN’s new location which requires one way delay (Td) in new network.

Summing up all the individual delay components, total handover delay is

\[
T_{\text{handover}} = T_{L2} + T_{IP} + T_{\text{sig}} + T_d
\]

As TL2 and TIP does not depend upon the mobility management protocol therefore for the performance comparison of different mobility management protocols, we consider the values of TL2 and TIP as constant.

A mathematical model for the evaluation of handover delay of Mobile IPv6 (MIPv6) and mSCTP has been discussed in Kim and Koh (2008). According to this model, signaling delay of some mobility management protocol x can be modeled as a function of number of signaling messages exchanged and the end-to-end delay between communicating nodes (Kim and Koh 2008).

\[
T_{\text{sig},x} = N(D_{MN-RN} + M(D_{RN-MN}))
\]

where \(T_{\text{sig},x}\) is the handover signaling delay for protocol \(x\), \(N\) is the number of handover signaling messages sent from MN to RN, \(M\) is the number of handover signaling messages sent from RN to MN and \(D_{x,y}\) is the one way end-to-end delay for handover signaling messages between nodes \(x\) and \(y\).

Handover delay of end-to-end mobility management protocols in non-overlapping regions includes all the four delay components discussed in Eq. (1) above. In this scenario MN loses connectivity from the old access network and enters into the coverage area of new access network, where it has to face all the handover delay components as shown in Fig. 5.

\[
T_{\text{handover},x} = T_{L2} + T_{IP} + T_{\text{sig},x} + T_d
\]

Handover delay, when MN enters into the overlapping coverage area of two different access networks, depends upon the mobility management protocol’s capability of whether it supports multi-homing and allows simultaneous communication through two network interfaces or not. If mobility management protocol supports multi-homing and allows simultaneous communication through two network interfaces, then handover delay will be zero (MN will keep on receiving data through old network interface while it is performing handover signaling through new network interface), otherwise it will be the sum of handover signaling delay and \(Td\), as shown in Fig. 6.

\[
T_{\text{handover},x} = \begin{cases} 
0 & : \text{Protocol allows simultaneous communication} \\
T_{\text{sig},x} + T_d & : \text{Protocol does not allow simultaneous communication} 
\end{cases}
\]

In case MN moves out of overlapping regions, handover delay will depend upon the presence or absence of Link Going Down (LGD) prediction intelligence (Lampropoulos et al., 2008). If mobility management protocol has the intelligence to predict LGD trigger, then MN can initiate handover signaling through alternate interface before the old link goes down, resulting in zero handover delay, otherwise handover delay will be equal to the sum of handover signaling delay and one way delay for data packet between MN and RN in new network, as shown in Fig. 7.

\[
T_{\text{handover},x} = \begin{cases} 
0 & : \text{Protocol has LGD intelligence} \\
T_{\text{sig},x} + T_d & : \text{Protocol has no LGD intelligence} 
\end{cases}
\]

Majority of end-to-end protocols for mobility management use connection re-establishment or connection migration, between MN and RN in new network, to perform the handover. Cardwell et al. (2000) has modeled the TCP connection setup delay as a function of end-to-end packet loss probabilities in forward and reverse directions.

To calculate the end-to-end packet loss probabilities, we use the reference scenarios as shown in Fig. 2. MN is attached to wireless Access Point (AP)/Base Station (BS) and AP/BS is connected to the Internet with wired medium. The path between MN and RN consists of two parts: wireless link between MN and BS/AP and the wired link between BS/AP and the RN.
The end-to-end packet loss probability between MN and RN as discussed by Mohanty and Akyildiz (2007)

\[ p = 1 - (1 - P_{\text{rol}})(1 - P_{\text{wr}}) \]  

where \( P_{\text{rol}} \) is the packet loss probability in wireless part and \( P_{\text{wr}} \) is the packet loss probability in wired part.

Packet loss probability in wireless link can be represented as a function of link layer frame error rate and the number of link layer frames per packet (Mohanty and Akyildiz, 2007).

\[ P_{\text{rol}} = 1 - (1 - \text{FER})^k \]  

where FER is the link layer Frame Error Rate (FER) and \( k \) is the number of link layer frames per packet. \( k \) can be calculated by dividing the packet length with link layer frame length. Packet loss probability in the wired link is very small. Thus the end-to-end packet loss probability between MN and RN from Eqs. (6) and (7) is

\[ p = 1 - (1 - \text{FER})^k(1 - P_{\text{wr}}) \]  

The model proposed by Mohanty and Akyildiz (2007) discussed the connection setup/migration delay as function of end-to-end packet loss probability, number of messages exchanged, their number of retries and the retransmission timeout. Signaling delay of protocols that exchange three messages for handover, like TCP-Migrate, TCP-Redirect (TCP-R), Tsukamoto’s proposal and parallel-TCP (pTCP) can be modeled as:

\[ L_{\text{sig}} = 1.5RTT + \sum_{i=1}^{N} 2^i \text{RTO} + \sum_{b=0}^{k-1} 2^b \text{RTO} + \sum_{c=0}^{k-1} 2^c \text{RTO} \]  

\[ L_{\text{sig}} = 1.5RTT + (2^j + 2^i + 2^k - 3) \text{RTO} \]  

where \( i, j, k \) are the number of unsuccessful tries for handover control messages and protocol doubles Retransmission Time Out (RTO) after each RTO expiration.

For protocols that exchange two control messages for handover, this signaling delay is

\[ L_{\text{sig}} = \text{RTT} + \sum_{i=0}^{j} 2^i \text{RTO} + \sum_{b=0}^{k-1} 2^b \text{RTO} \]  

or

\[ L_{\text{sig}} = \text{RTT} + (2^i + 2^j - 2) \text{RTO} \]  

Let \( p_1, p_2 \) and \( p_3 \) are the end-to-end packet loss probabilities of three handover control messages, which are calculated using Eq. (8), then probability of handover signaling completion for different protocols, discussed in Eqs. (9) and (10) is

\[ P = p_1(1-p_1)p_2(1-p_2)p_3(1-p_3) \]  

\[ P = p_1(1-p_1)p_2^2(1-p_2) \]  

where \( P \) in Eqs. (11) and (12) is the overall probability of handover completion. Let \( P_{\text{sig},x} \) is the probability that handover message for protocol \( x \) will reach successfully after some retries, then the average value of handover signaling delay is

\[ E[T_{\text{sig},x}] = \sum_{i=0}^{N} \sum_{k=0}^{n} P_{\text{sig},x} I_{\text{sig},x} \]  

where \( N \) is the maximum allowed number of retries for handover signaling messages. Most TCP implementations abort connection establishment attempt after 4–6 failures (Cardwell et al., 2000).

Analytical model discussed in Munasinghe and Jamalipour (2007) modeled the signaling delay during handover. Focus of this model is to find the handover delay of Mobile IP but it can be used for end-to-end mobility management solutions as well. It has discussed signaling delay as a function of packet size, bandwidths and delays of wired and wireless parts, average number of hops and the processing delay at each hop.

\[ T_{\text{sig},x} = N \left[ \left( \frac{S}{B_{\text{rol}}} + L_{\text{rol}} \right) + \left( \frac{S}{B_{\text{wr}}} + L_{\text{wr}} \right) H_{a-b} + L_{\text{proc}}(H_{a-b} + 1) \right] \]  

where \( S \) is the average size of handover signaling messages exchanged between MN and RN, \( B_{\text{rol}} \) and \( B_{\text{wr}} \) are the bandwidths of wireless and wired links, \( L_{\text{rol}} \) and \( L_{\text{wr}} \) are the latencies of wireless and wired links respectively, \( H_{a-b} \) is the average number of hops between two communicating nodes, \( L_{\text{proc}} \) is the processing delay at each node and \( n \) is the number of handover control messages exchanged during handover between MN and RN. We used the same model to analyze the handover delay of end-to-end mobility management protocols as a function of link bandwidths.

4.3. Results comparison for handover delay

For our performance comparison, we assume that Layer 2 connectivity delay and IP address acquisition delay remain same for all end-to-end protocols in a particular access network. The only thing that affects the handover delay is the handover signaling delay that depends upon the nature of mobility management protocol.

To compare the handover delay performance of end-to-end mobility management solutions, we have taken following values for different parameters through simulation: \( T_{\text{ip}} = 10 \text{ ms}, T_{\text{ip}} = 20 \text{ ms}, T_{\text{ip}} = 50 \text{ ms} \). Packet loss probability in wired network is considered as \( P_{\text{rol}} = 1 - e^{-6}, RTT = 100 \text{ ms} \) and \( RTO = 200 \text{ ms} \). These values are not constant all the times; they may vary from network to network depending upon network conditions. But, the reality with these values is that, when we change the values, behavior of all the protocols will be affected in the same manner in which they are behaving in the results. These are just test values for the evaluation of different protocols. The variation in
values will affect all the protocols, e.g., if we change the value of RTT from 100 ms to 200 ms, then the handover delay for all the protocols will be affected but the manner in which they will behave will be just like same as represented in the current graphs.

Figure 8 shows the comparison of handover delay as a function of packet loss probability of different end-to-end mobility management protocols in scenario 1, when MN moves across heterogeneous networks with non-overlapping coverage areas. According to the results, the protocols can be categorized in three groups. This categorization is made on the basis of number of handover signaling messages exchanged. VA-TCP performs much better than other protocols. For example, for a loss probability of 0.15, the delay for LCT-based protocol is twice than that for SL-TCP and VA-TCP whereas for the TCP-Migrate and related protocols, the delay is about 3.6 times that of the SL-TCP and VA-TCP. The difference in delay performance of different protocols increases with increasing packet loss probability. It can be seen that SL-TCP and VA-TCP are least affected with the increase in packet loss probability whereas TCP-Migrate, pTCP, HIP, Tsukamoto’s proposal and TCP-R are the most affected. The reason is that, for SL-TCP and VA-TCP the number of handover control messages is smaller as compared to other protocols and an increase in packet loss probability increases the chances of dropping handover control messages causing higher handover delays for TCP-Migrate, pTCP and others.

Figure 9 shows the handover delay in second scenario, when MN enters into overlapping region of two access networks. Behavior of TCP-Migrate, TCP-R and SL-TCP is same as was in Fig. 8. The reason is that these protocols do not support multi-homing and simultaneous communication through parallel network interfaces, causing higher handover delay. VA-TCP, SLM, pTCP, LCT-based handoff protocol, EMF, Tsukamoto’s proposal and HIP have zero handover delay. These protocols allow MN to exchange data through old network interface while performing handover signaling through new interface thus zero handover delay is observed in scenario when MN moves into overlapping region.

Figure 10 shows the handover delay in scenario when MN moves out of overlapping region. Here handover delay of VA-TCP has become smaller than SL-TCP, as compared to Fig. 10, where both have same handover delay. The reason for this decrease is that, VA-TCP allows multi-homing, thus MN a priori performed the layer-2 connectivity and IP address acquisition in new access network. Similarly TCP-Migrate and TCP-R have higher delays as compared to SLM, pTCP and HIP-based solutions, due to non-multi-homed nature. In this scenario, as discussed earlier, handover delay for protocols that have LGD prediction intelligence will be zero. As EMF has LGD prediction intelligence, and it can predict a link that is about to break, hence it initiates the vertical handover through alternate available network interface before the old link is down causing zero handover delay in this scenario.

To compare the handover delay as a function of bandwidth, we used following values for different parameters: $S=60$ bytes, $L_{proc}=1e^{-3}$ ms, $H_{a-b}=10$, $L_{w}=0.1$ ms against different values of $B_{s}=1$, $L_{w}=1e^{-3}$ ms and $B_{s}=1$ Mbps. Against different values of $B_{s}$, we found that signaling delay decreases exponentially up to an increase in bandwidth of 500 Kbps. After which this decrease becomes linear by increasing the bandwidth but up to a certain limit, 1.5 Mbps, and after which it remains same irrespective of the increase in bandwidth as shown in Figs. 11–13.

![Fig. 8. Handover delay as function of packet loss probability in non overlapping region.](image)

![Fig. 9. Handover delay as function of packet loss probability, where MN enters into overlapping region.](image)

![Fig. 10. Handover delay as function of packet loss probability, where MN leaves the overlapping region.](image)
Figure 11 shows the handover delay as a function of bandwidth in scenarios where MN moves across non-overlapping regions of two access networks. Behavior of protocols is same as was in Fig. 8. It can be seen that SL-TCP and VA-TCP perform better as compared to other protocols. These two protocols, as discussed earlier, exchange less number of handover messages resulting minimum handover delay.

Figure 12 shows the handover delay as a function of bandwidth in second scenario, when MN while connected to an access network enters into the overlapping region. Here handover delay of a group of protocols is found to be zero. These protocols (VA-TCP, SL-TCP, pTCP, LCT-based, HIP, Tsukamoto's proposal, EMF) allow MN to exchange data through old network interface while performing handover signaling through new interface. Thus MN will be receiving data through old network interface while mobility management protocol is performing handover process, resulting in zero handover delay.

In scenario where MN leaves the overlapping coverage area, handover delay as a function of bandwidth is shown in Fig. 13. Results show that TCP-Migrate and TCP-R have higher handover delays as compared to SLM, pTCP, Tsukamoto’s proposal and HIP, due to non-multi-homed nature. Again EMF has zero handover delay because of LGD prediction intelligence.

5. Performance evaluation model for throughput degradation time

After moving out of the coverage area of an access network, Mobile Node (MN) becomes unable to exchange data, thus zero throughput is observed. When MN enters into a new network, end-to-end mobility management protocols migrate the old TCP connection or establish a new one. Normally a new TCP connection starts from slow start and gradually moves to a steady state. In this situation, MN suffers from the throughput degradation.

5.1. Modeling throughput degradation time after vertical handover

Throughput degradation time during vertical handover is the sum of handover delay and time spent by TCP connection to reach steady state. If the handover delay is zero (as discussed in Section 4), then throughput degradation time will be the time spent by TCP connection to reach steady state.

\[ T_{TD} = T_{H0} + T_{ss} \]  
(15)

where \( T_{TD} \) = throughput degradation time; \( T_{H0} \) = time for handover; \( T_{ss} \) = time to reach at steady state.

To model the throughput degradation time, we calculate the time spent by TCP connection in slow start phase, i.e. time to reach steady state. Time to reach steady state, as a function of number of slow start rounds and the RTT value, has been modeled by Mohanty and Akyildiz (2007) as

\[ T_{ss} = \frac{\log_2(1 + CW_{ss}) - 1}{RTT} \]  
(16)

where \( \log_2(1 + CW_{ss}) - 1 \) is the number of slow start rounds, and RTT is the average round trip time. For simplicity, we assume that there is no packet loss during slow start and TCP doubles its congestion window every RTT and after time \( T_{ss} \) TCP connection reaches at steady state. Congestion window at steady state,
\( CW_{ss} \) is (Mohanty and Akyildiz 2007)

\[
CW_{ss} = 1 + 2 + 2^2 + 2^3 + \ldots + 2^{i-1} - 1
\]

Here \( i \) is the number of rounds taken by TCP to reach at steady state.

To calculate the congestion window size and time spent in slow start, Cardwell et al. (2000) considered \( \gamma \) to denote the rate of exponential growth of congestion window during slow start. According to this model, amount of data sent to reach steady state, \( data_{ss} \) in \( i \) number of rounds can be modeled, using geometric series, as

\[
data_{ss} = w_1 + w_1 \gamma + w_1 \gamma^2 + \ldots + w_1 \gamma^{i-1} \\
\text{or}
\]

\[
data_{ss} = w_1 \frac{\gamma^i - 1}{\gamma - 1}
\]

where \( w_1 = \text{initial congestion window} \). Normally \( w_1 = 1 \) (Stevens 1997).

Number of rounds \( i \), that TCP spends to reach at steady state, can be derived from Eq. (18) as (Cardwell et al., 2000)

\[
i = \log_{\gamma} \left( \frac{data_{ss} (\gamma - 1)}{w_1} + 1 \right)
\]

The expected time spent in slow start can be expressed as (Cardwell et al., 2000)

\[
E[T_{ss}] = RTT \cdot i
\]

\[
E[T_{ss}] = RTT \cdot \log_{\gamma} \left( \frac{E[d_{ss}] (\gamma - 1)}{E[w_1]} + 1 \right)
\]

Here \( E[d_{ss}] = \text{avg}(data_{ss}) \)

In case of MN’s movement in non-overlapping regions, throughput degradation time for protocols that do not have any throughput degradation avoidance mechanism is the sum of handover delay and time spent in slow start phase. In this scenario, MN loses its connectivity in old access network and after some time, enters into coverage area of another access network where it first establishes a new TCP connection or migrates the existing connection and then starts exchanging data with the peer node in slow start phase. For protocols that use throughput degradation avoidance mechanism (like VA-TCP), this time will be equal to the time spent for handover. Let \( T_{D,x} \) is the throughput degradation time in vertical handover for protocol \( x \), then:

\[
T_{D,x} = \begin{cases} 
T_{H0} + T_{ss} & \text{No throughput degradation avoidance mechanism} \\
T_{H0} & \text{Having throughput degradation mechanism}
\end{cases}
\]

(21)

In case MN moves into overlapping region of two access networks, throughput degradation time depends upon the capabilities of mobility management protocol. For the protocols that do not use any throughput degradation avoidance mechanism and allow exchanging data through old connection until new connection is established, throughput degradation time is the time spent by new TCP connection to reach steady state. Let \( T_{D,x} \) is the throughput degradation time after vertical handover for protocol \( x \), then:

\[
T_{D,x} = \begin{cases} 
T_{ss} & \text{No throughput degradation avoidance mechanism} \\
0 & \text{Having throughput degradation avoidance mechanism}
\end{cases}
\]

(22)

In case MN moves out of overlapping region and the mobility management protocol has the IEEE 802.21 MIH LGD (Link Going Down) trigger intelligence (Lampropoulos et al., 2008), then MN can detect in advance that interface through which it is connected is going to down. In this case, MN can initiate handover through new interface before losing connectivity of old access network, throughput degradation time is same as discussed in Eq. (22).

5.2. Results comparison for throughput degradation time

To compare the throughput degradation time during vertical handover using models discussed in Section 5.1, we used \( w_1 = \text{MSS} \) (Stevens 1997), \( d_{ss} = 1024 \text{ segments} \) (Cardwell et al., 2000). Again, as discussed in Section 4.3, these values are not constant all the times; they may vary from network to network depending upon network conditions. These are just test values for the evaluation of different protocols. The variation in values will affect all the protocols but the manner in which protocols will behave, will be just like same as represented in the current graphs.

By using these values, and for simplicity assuming that there is no packet loss during data transmission (Mohanty and Akyildiz 2007), we calculated the number of slow start rounds \( i \), \( E[W_1] \), \( E[T_{ss}] \), \( CW_{ss} \) and \( T_{ss} \).

Figure 14 shows the throughput degradation time for non-overlapping region scenario. Here it includes the time to reach at steady state and time for handover except for VA-TCP which has the lowest throughput degradation time. It is because, when roaming into a new network, VA-TCP allows MN to dynamically estimate the bandwidth, delay and the bandwidth-delay product. On the basis of estimated bandwidth-delay product, VA-TCP adjusts slow start threshold, congestion window, RTO, and RTT appropriately to fit into the new network conditions. Hence, VA-TCP does not start from slow start phase thus avoiding throughput degradation time.

Figure 15 shows the throughput degradation time of different protocols in scenario 2 where MN moves into the overlapping region. Mobility management protocols that allow sending/receiving data through old connection, e.g. VA-TCP, SLM, EMF and pTCP have zero throughput degradation time. It is because, pTCP, EMF and SLM allows simultaneous communication through parallel interfaces, while VA-TCP adjusts the congestion window according to the new network conditions. HIP allows simultaneous communication through two interfaces but restricts that when data is traveling on old network interface, handover control messages will be exchanged...
through new network interface. And after handover TCP starts from
slow start causing throughput degradation.

Figure 16 shows the throughput degradation time in scenario
3 where MN moves out of overlapping region. As only EMF has
the intelligence to use the IEEE 802.21 MIH LGD (Link Going
Down) trigger and as connection over new network interface
reaches steady state before the old link goes down therefore,
throughput degradation time is zero. For rest of the other
protocols, handover delay is added in the throughput degradation
time.

Results show that protocols that allow simultaneous commu-
nications after vertical handover (like pTCP, Tsukamoto’s proposal,
EMF and SLM) perform better in overlapping regions with minimum
or zero throughput degradation time as compared to protocols that
do not allow simultaneous communication through parallel inter-
faces. Also, protocols that use mechanisms to avoid throughput
degradation time after vertical handover, like VA-TCP, performs
better in overlapping as well as in non-overlapping regions.

6. Performance comparison for protocol overhead

Protocol overhead is the total amount of additional control
information sent by the mobility management protocol with each
data packet and/or in handover control messages. This overhead
amount depends upon the nature of mobility management
protocol.

6.1. Modeling protocol overhead

Protocol overhead for protocol \( x \) “PO\(_x\)” is the sum of overhead
created by protocol for handover signaling and the overhead of
the additional bytes sent with each data packet, i.e.

\[
PO_x = \text{Handover overhead} + \text{Additional bytes sent with each data packet}
\]  \hspace{1cm} (23)

where \( \text{Handover overhead} = a(S_{MN-RN}) + b(S_{RN-MN}) + c(S_{HP}) \)

\hspace{1cm} (24)

where \( a \) is number of messages sent from MN to RN for con-
nection migration, \( S_{MN-RN} \) is the size of message sent from MN to RN
for connection migration, \( b \) is the number of messages sent from
RN to MN for connection migration, \( S_{RN-MN} \) is the size of message
sent from RN to MN for connection migration, \( c \) is number of
handover packets and \( S_{HP} \) is the size of handover packet sent from
MN to RN.

Handover of additional bytes sent with each data packet for \( n \)
data packets can be computed as

\[
\text{Handover overhead} = \sum_{i=1}^{n} S_{Add_bytes}
\]  \hspace{1cm} (25)

where \( S_{Add_bytes} \) is the additional bytes sent with each data packet
by the protocol and \( n \) is the number of data packets.

Table 2 shows a comparison of different end-to-end mobility
management protocols regarding protocol overhead. Here, hand-
over signaling message packet size includes an average of 60
bytes of TCP, IP and link layer headers also, except for HIP that does not contain TCP header.

6.2. Discussion regarding protocol overhead

Figure 17 shows the protocol overhead of different end-to-end mobility management protocols after handover initiation for \( n = 1 \) to 10. Protocols that send additional information with each data packet have higher protocol overhead as compared to protocols that do not send any additional information with data packets. Protocol overhead of HIP is highest of all because an additional HIP base header of 40 bytes is sent with each data packet. Protocol overhead for TCP-Migrate and TCP-R, remains constant after exchanging the handover control message. It is due to the fact that these protocols do not send any additional information with data packets, keeping the protocol overhead constant. For SL-TCP, protocol overhead starts from 66 bytes and is increased by 6 bytes, by sending an additional field, \( CID \), with each data packet. This increase does not create as much overhead as created by HIP, pTCP, EMF and SLM (Table 3).

Figure 18 shows the overhead comparison of protocols that send additional information with each data packet. This overhead is computed for \( n = 1000 \) using Eq. (25). It was found that protocol overhead of HIP, pTCP, SLM and SL-TCP for sending 1000 packets is 40,252, 12,252, 4244 and 6066 bytes respectively. Thus HIP generates an overhead traffic of around 40KB, pTCP around 12KB, SLM around 4KB, EMF around 20KB and SL-TCP generates an overhead traffic around 6KB, for sending 1000 packets over the network.

7. Discussion on security issues

End-to-end mobility management approaches simplify the trust relationship between two communicating nodes, as compared to the network-layer approaches like Mobile IP or proxy based solutions. Several Mobile IP based approaches require authentication of MN with Home Agent and arbitrary number of Foreign Agents.

In end-to-end mobility management solutions the only trust relationship that is required is between the two communicating nodes. No intermediate entity is required with which additional trust relationship is needed. In spite of this simplified trust relationship, there are possibilities that attacker can launch attacks against MN and RN or against the connection between them. Two different types of attacks that can be launched by an attacker on the mobility management protocols are:

- Fake location update attack: Sending a fake location update control message.
- Redirection attack: Sending a fake handover update control message. It may also be termed as session hijacking attack.

Discussion of security mechanisms used by different end-to-end mobility management solutions has been made next.

7.1. Security in HIP

IPSec is used to secure the HIP message exchange. HIP defines a base message exchange for connection initiation, containing four messages, a four-way handshake. During this message exchange, Diffie-Hellman key exchange mechanism is used to create a session key and to establish a pair of IPSec ESP Security Association (SA) between two communicating nodes. An ESP SA pair is indexed by the SPIs and the two HITs (Moskowitz and Nikander, 2006). The SAs need not be bounded to IP addresses because all internal control of the SA is by the HITs. Thus, a node can easily change its IP address and still maintain the SAs. There are no chances of redirection attacks because all communication is secured by IPSec.

7.2. Security in TCP-R

The only vulnerability point in this protocol is the Session key. If a malicious node knows the session key of any ongoing communication, it can initiate various types of attacks to the connections in this session. To secure this session key, TCP-R has a
built-in feature. Two nodes exchange authentication keys at the
time of connection start up to establish the session key. Whenever
a MN changes its point of attachment in the network, it sends a
Redirection request to RN, containing the old and new IP addresses
encrypted with the session key. RN sends Authentication request.
As a response MN sends Authentication reply. Both Authentication
request and reply messages are encrypted, thus reducing the
chances of man-in-the-middle attack for authentication request
response messages.

End nodes that are unconcerned with connection security, can
do this by just sending type without any authentication key
values in the connection establishment messages.

7.3. Security in TCP-migrate

TCP-Migrate has a built-in mechanism to secure the handover
update message including migrate options. End nodes negotiate
an unguessable connection token (connection identifier) secretly
by encrypting it with session key, which is computed earlier by
two nodes separately and is never transported over the network.
This scheme reduces the security vulnerabilities of TCP-Migrate
connection as compared to that of a traditional TCP connection.
Without finding the token (connection identifier), attacks are
difficult to launch against TCP-Migrate connection. Since any host,
that obtains the secret connection key, can make the token and
can issue a Migrate request, thus to avoid this discrepancy, TCP-
Migrate chooses the session key with Elliptic Curve Diffie-Hellman
key exchange at connection startup time.

End nodes that are using IPSec for security or those uncon-
cerned with connection security, may choose to disable key
negotiation to avoid computational overhead.

7.4. Security in pTCP

pTCP has no built-in feature for security and is susceptible to
redirection attacks. Source and destination Identifiers along with
pTCP sequence number are included in pTCP header of handover
control messages to identify the proper source and destination.
If an intruder gets a pTCP packet then he can launch redirection
attack using the handover control message. The intruder can
initiate a new TCP-v pipe with RN and can send a false pTCP packet
to RN for handover the transmission to the new TCP-v pipe. RN has
no capability to check that this handover update message is sent
from authenticated user or not.

7.5. Security in Tsukamotos' proposal

Tsukamoto et al. have not discussed any security mechanism.
Thus, this proposal is vulnerable to redirection attacks. As no
session identifier is used and no security measure is incorporated
therefore, intruders can send a false handover control message on
behalf of communicating node and session can be redirected
easily.

7.6. Security in LCT-based handoff protocol

Possible vulnerability in this protocol can be the CID. To
overcome this vulnerability, this protocol has its own security
mechanism that sends encrypted handover control messages over
the Internet using a shared secret key. Shared secret key is
generated through Diffie-Hellman key exchange between two
communicating nodes, at the connection startup time. Thus no
significant security risk is involved in this protocol regarding the
redirection attack.

7.7. Security in VA-TCP

This protocol has no built-in feature for security and is
susceptible to redirection attacks. Authors have not discussed
any mechanism to secure the handover update control messages.
Redirection attack can easily be launched by sending a false
handover control message to RN. As no authentication is required
from RN for the handover control message, thus an attacker can
simply send a handover control message (notification message),
containing new IP address and port number, to RN for doing false
handover to any new IP address.

7.8. Security in SLM

No security mechanism is defined to secure the handover
signaling messages. If a malicious user gets the Session ID then he
can easily launch different security attacks. So a secure mechan-
ism is required to send encrypted Session ID, when transmitting it
over the network in a handover signaling message.

7.9. Security in EMF

EMF has built-in security mechanism to avoid the redirection,
man-in-the-middle and replay attacks. For this purpose EMF
implemented the Elliptic Curve Diffie-Hellman key exchange
mechanism for shared secret key exchange. Association Identifier
(AID) of EMF is encrypted with this shared secret key. Only
 certain fields of handover control messages are encrypted while
data traffic is not encrypted in EMF.

7.10. Recommendations

As no security mechanism is proposed for pTCP, SLM, VA-TCP
and Tsukamoto’s proposal therefore, IPSec with renegotiation of
security associations on handover initiation can be used to secure
the control signaling of these protocols. If low handover delay is
desirable, then MObIKE (Eronen, 2006) can be used to secure
these protocols, as IPSec with renegotiation of association has
high handover delay. MOBIKE handles the problem of security
association renegotiations upon the mobility of nodes across
heterogeneous networks. These security recommendations do
not require any changes in proposed protocols design.

HIP and TCP-Migrate uses encrypted data traffic. As, mobility
management solutions are required to provide mobility manage-
ment service in a secure manner and data confidentiality is out of
the scope of mobility management solutions therefore, it is
recommended that encrypting each data packet in HIP and TCP-
Migrate should be avoided.

8. Discussion on implementation issues

We have discussed different mobility management solutions
with their strengths and weaknesses. Some of these solutions
have implementation concerns and require extensive efforts to
implement them. They require major changes in communication
protocol stack or require changing some parameters at different
layers of the protocol stack. Also, no socket option or ioctl is
available that can be used easily from the upper layers, to change
these parameters during the communication. The only mechan-
ism to incorporate these changes is to change the System Kernel.
In this section, we have highlighted the implementation issues that
exist for the implementation of these protocols. Table 4 shows a
comparison of protocols with respect to implementation issues.

Some protocols like TCP-Migrate and TCP-R introduce new TCP
options to send handover control information. Hence both ends
require changes in TCP implementation to process these additional options. Some protocols like TCP-R and VA-TCP require changes in Transmission Control Block (TCB) of TCP at both ends. As no socket option or system call exists that can be used to make these changes in TCB, hence it would be difficult to implement this functionality.

pTCP has introduced a new concept of TCP-v pipes that has the similar functionality to that of a standard TCP connection. Each TCP-v pipe has its own congestion control scheme. This requires the implementation of complete congestion control mechanism. Also pTCP has its own flow control mechanism as compared to standard TCP, thus complete new TCP implementation is required.

SLM and EMF do not introduce any changes in TCP implementation and Kernel, rather they use the conventional TCP. They only send the data/handover control information, as TCP payload which requires no changes in TCP implementation. The only thing required, is to incorporate the mobility management mechanism at an intermediate layer between Transport and Application layers of protocol stack on both ends. As this approach can introduce the application transparency issue therefore, to overcome this issue, EMF has implemented the application transparency module between EMF layer and application layer. However, SLM lacks this functionality thus, suffers from the application transparency problem.

9. Summary of qualitative comparison of mobility management protocols

Besides presenting the comparison of quantitative parameters like handover delay, throughput degradation time and protocol overhead, we have also discussed some qualitative parameters such as security issues and implementation issues of the end-to-end mobility management protocols. Continuing this qualitative comparison, in this section we discuss some other qualitative parameters to compare the capabilities of these protocols.

Table 5 shows a comparison of different end-to-end mobility management protocols. Each column corresponds to an end-to-end mobility management protocol and each row represents a characteristic of these protocols. In the following, we briefly discuss these characteristics.

- **Application transparency:**
  A mobility management scheme is said to be application transparent if user application remains unaware of the underlying handover process. Almost all the protocols provide their services in an application transparent manner except SLM.

- **Multi-homing support:**
  Although using multi-homing, an already established communication between two nodes can be spanned across multiple network interfaces however, this feature depends upon the capability of mobility management protocols. Not all mobility management protocols are able to support this capability, e.g. TCP-R, SL-TCP and TCP-Migrate do not support the multi-homing feature.

- **Willful handover:**
  Handover should satisfy the preferences of end users. For example, in case of overlapping regions, a user might be willing to switch its connection from an access network with strong signal strength but having higher monetary cost to an access network with relatively weak signal strength but having lower monetary cost. This is termed as willful handover. Only EMF has provided this capability.

- **Seamless handover:**
  A seamless handover allows mobile devices to continue communication without any service disruption. As TCP-R, SL-TCP and TCP-Migrate do not support multi-homing therefore, these protocols also do not provide the make-before-break facility that is the key to the seamless handover.

- **Simultaneous movement awareness:**
  In some scenarios, it is possible that both the communicating nodes move simultaneously. A mobility management scheme is said to be simultaneous movement aware if it supports simultaneous movement of two communicating nodes. Not all protocols support this feature. Only HIP, LCT-based solution and EMF provide this support.

- **Requirement of Additional entity in network:**
  A mobility management scheme may require additional support from the network for handover control and data signaling, for example use of gateways, proxies, mobility agent, etc. HIP and LCT-based handoff protocols require additional support from network, but are still included in the list of end-to-end protocols. Reason is that, the additional network entities are not required for simple handover management, but they are required to support simultaneous mobility feature only.

### Table 4
Comparison of different end-to-end mobility management solutions w.r.t. implementation issues.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Changes in OS kernel</th>
<th>Changes in protocol stack</th>
<th>Changes at both ends</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIP</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>TCP-R</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>TCP-Migrate</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>pTCP</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Tsukamoto’s proposal</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>LCT-based</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>VA-TCP</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>SLM</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>EMF</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Table 5
Qualitative analysis matrix for end-to-end protocols.

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>HIP</th>
<th>TCP-R</th>
<th>SL-TCP</th>
<th>TCP-Migrate</th>
<th>pTCP</th>
<th>Tsukamoto’s proposal</th>
<th>LCT-based</th>
<th>VA-TCP</th>
<th>SLM</th>
<th>EMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol layer</td>
<td>Layer 3.5</td>
<td>Layer 4</td>
<td>Layer 4</td>
<td>Layer 4</td>
<td>Layer 4</td>
<td>Layer 4.5</td>
<td>Layer 4.5</td>
<td>Layer 4.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application transparency</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Multi-homing support</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Willful handover</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Seamless handover</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Simultaneous handover</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Requirement of additional network entity</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Lower layer awareness</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Link going down (LGD) prediction intelligence</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Similarly, SLM requires additional locator server for its location updates.

**Lower layers awareness:**
A mobility scheme that takes information from lower layers, e.g., about signal strengths etc. for handover decision, is said to be lower layer aware. Cross-layer techniques are normally used to pass the lower layer information to upper layers, bypassing the intermediate layers. Majority of the protocols nowadays use this cross layer optimization. However, TCP-R and SL-TCP do not posses any mechanism to handle lower layer information.

**Prediction for link going down:**
It is the ability of a MN to predict that the link through which it is connected is about to break or not. If a MN is moving out of the coverage area of an access network and it has the intelligence for link going down prediction, then MN can initiate its handover signaling through alternate available active network interface thus minimizing the handover delay (Yousaf et al., 2009). This feature is only supported by the EMF architecture and none of other protocol has implemented such intelligent mechanism.

10. Conclusion and future directions

In this paper, we analyzed and compared different end-to-end mobility management protocols in terms of handover delay, throughput degradation time and protocol overhead. Some security and implementation issues related to these protocols, are also discussed. Handover delay and throughput degradation time have been discussed in three different scenarios on the basis of MN’s movement across heterogeneous networks. Different mathematical modeling techniques were used to compare the performance of end-to-end mobility management protocols.

Results show that when MN moves between two non-overlapping regions, throughput degradation time is lowest for VA-TCP. The reason is that VA-TCP uses a throughput degradation avoidance mechanism in which new TCP connection does not start from slow start phase and congestion window is adjusted according to the new network’s conditions. In scenarios where MN moves between regions with overlapping coverage area, handover delay is found to be zero for a group of protocols including pTCP, VA-TCP, Tsukamoto’s proposal, EMF and SLM. These protocols use simultaneous communication through two different network interfaces which makes handover delay as zero. Similarly in overlapping regions, throughput degradation time for these protocols is also zero. This is because MN keeps on sending/receiving data through old network interface while it performs handover signaling through new network interface.

In scenarios, where MN moves out of overlapping region, handover delay of EMF is found to be zero. EMF incorporates the link going down prediction intelligence that is used to predict that MN is moving out of the coverage area of the access network and the link may soon go down. EMF predicts link going down and initiates a new TCP connection in-advance using the alternate available network interface, thus reducing the handover delay and throughput degradation time.

Similarly, protocol overhead of mobility management protocols that do not send any additional control information with data packets, like TCP-Migrate and TCP-R, is lowest as compared to the protocols that send additional control information with each data packet like pTCP, SLM, EMF and HIP.

Research community can use the results of this mathematical analysis as the first step towards performance evaluation of different end-to-end mobility management protocols. Protocols with distinguished performance from this study can be selected and implemented first as simulator patches and then as actual implementation. This will help to analyze the performance of these end-to-end protocols in real scenarios.

16

References


Moskowitz R, Nikanter P. Host identity protocol (HIP) architecture. IETF RFC 2006;4423.


Yousaf M, Qayyum A, Malik SA. An architecture for exploiting multihoming in mobile devices for vertical handovers & bandwidth aggregation. Springer Wireless Personal Communications Journal, ISSN 0929-6212, Published online May 8, 2011.
