M^f : MultiMedia Mobility Manager – a seamless mobility management architecture supporting multimedia applications

Karl Andersson
Division of Mobile Networking and
Computing
Luleå University of Technology
SE-931 87 Skellefteå, Sweden
+46 910 58 53 64
karl.andersson@ltu.se

Daniel Granlund
Division of Mobile Networking and
Computing
Luleå University of Technology
SE-931 87 Skellefteå, Sweden
+46 706 70 35 34
daniel.granlund@ltu.se

Christer Åhlund
Division of Mobile Networking and
Computing
Luleå University of Technology
SE-931 87 Skellefteå, Sweden
+46 910 58 53 31
christer.ahlund@ltu.se

ABSTRACT

In this paper, a proof-of-concept and a software architecture, M^f (MultiMedia Mobility Manager), is presented. In short, M^f is offering seamless mobility management to multimedia applications using a variety of wireless access networks. First, M^f is built on multihomed Mobile IP building on the principle of soft handovers. Second, network selection in M^f is based on a network layer metric combining round-trip times and jitter in round-trip times. Third, the end-user can enter its own preferences on network selection through a policy-based extension to the proposed network selection algorithm.

The proposed architecture is evaluated in a live heterogeneous networking environment where handover performance is studied in detail. In addition, user-perceived quality of experience for Voice over IP using the M^f software architecture has been studied. Last, downloads of large amount of data using a combination of high and low capacity wireless networks in M^f were studied.

Results indicate that the architecture as a whole and the proposed algorithms perform well. M^f can thus be seen as an implementation of the “*Always Best Connected*” vision.

Categories and Subject Descriptors

C.2.2 [Computer-Communication Networks]: Network Protocols – *routing protocols*.

General Terms

Algorithms, Performance, Design, Experimentation.

Keywords

Seamless mobility, heterogeneous networking environments, multimedia applications, quality of experience, Mobile IP,

multihoming, policy-based networking.

1. INTRODUCTION AND BACKGROUND

A growing portion of mobile phones and PDAs offer connectivity over more than one wireless interface. Typically UMTS and CDMA2000 mobile phones are today equipped with Bluetooth and WLAN (IEEE 802.11) radio interfaces. At the same time existing radio technologies are improved with higher data rates and better coverage. Also, new radio access technologies are currently being deployed in parts of the world, as is the case with WiMAX (IEEE 802.16).

The multimedia capabilities of handheld devices are also significantly improving with higher screen resolutions, new types of input devices and techniques, and better computing power. Battery capacities are typically improved and positioning capabilities are often built in to such devices today. In this way new applications both for leisure and business purposes are currently being developed and deployed to the new devices. Mobile TV, mobile games, and easy-to-use way-finders are popular applications today.

To leverage all those mentioned new features of current handheld devices and to support fully interactive, multimedia applications there is a growing need for seamless mobility management in heterogeneous networking environments following the vision of “*Always Best Connected*” [1]. The basic idea is to take full advantage of the emerging wireless networks using a variety of different access technologies simultaneously.

The M^f software architecture is targeted to support this type of usage scenario.

A mobility management scheme must support two basic operations: handoff and location management [2]. The handoff process needs, in turn, to determine the timing of the handoff, take decision on what access network to transfer the traffic to, and to migrate existing connections smoothly. Location management is the process for locating a mobile node (MN) in order to initiate and establish a connection.

Heterogeneous networking environments require a mobility management function working at the application layer, the transport layer, or the network layer. However, data-link layer-based solutions cannot be considered because of the multi-access

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM'07, December 12-14, 2007, Oulu, Finland.

Copyright 2007 ACM 978-1-59593-916-6/07/0012...\$5.00.

network environment being used and the need for support of vertical handovers.

Mobility management at the application layer was proposed in [3] essentially making use of an extension to the SIP protocol. Location management is handled through SIP Registrar servers while connection migration is handled using the proposed extension with re-INVITE messages. This type of mobility management scheme is suitable for session-based applications using UDP as transport protocol. TCP-based applications cannot be handled this way.

There are also proposals on transport layer-based solutions. The Stream Control Transmission Protocol (SCTP) [4] is an end-to-end, connection-oriented protocol that supports transport of data in independent sequenced streams. SCTP supports multihoming and combines the datagram orientation of UDP with the sequencing and reliability of TCP. Cellular SCTP (cSCTP) [5] is an extension to SCTP making handoffs smoother by sending data via multiple paths during handoffs. Transport-layer mobility management is gaining interest in the research community [6], but deployment of a new transport layer protocol such as SCTP is not made easy to a large amount of applications.

Finally, mobility management handled by the network layer using Mobile IP (MIP) has been proposed [7, 8]. The most notable benefit of this approach is that any application can stay unchanged using a stable (fixed) IP address. MIP, being essentially an extension to standard IP, takes care of the delivery of packets to the user's current point of attachment to the Internet. Both TCP-based and UDP-based applications are supported.

The M^f software architecture is implementing the network layer mobility management approach.

The rest of the paper is organized as follows. Mobility management with multihomed Mobile IP is presented in section 2, while section 3 describes network selection techniques, the relative network load metric, and introduces the policy-based decision model. Section 4 describes the M^f software architecture itself. Results and the evaluation framework and scenarios used are presented in sections 5 and 6 respectively. Section 7 presents related work, and finally section 8 discusses the results and indicates future work.

2. MOBILITY MANAGEMENT WITH MULTIHOMED MOBILE IP

Since IP addresses are used both for routing purposes and end-point (host) identification, MIP separates those two functions by using two IP addresses for the mobile node (MN), namely a home address (HoA) being stable over time and a care-of address (CoA) giving the location of the MN. A home agent (HA) located on the home network is responsible for forwarding packets to the MN when it connects to foreign networks.

The MN is informing the HA of its current point of attachment to the Internet, by sending binding update (BU) messages periodically. The HA replies to BU messages by sending binding acknowledgements (BA) messages back to the MN.

Incoming traffic, i.e. packets originating from other hosts referred to corresponding nodes (CN), travel via the HA and are then tunneled through an IP tunnel using IP in IP encapsulation to the MN. Finally the MN takes care of decapsulation and delivery of

packets to upper layers. Outgoing traffic, i.e. packets originating from applications in the MN, are handled vice versa.

Different types of route optimization techniques exist in the MIP standards for IPv6. When used, all traffic does not have to pass through the HA.

The possibility to register more than one CoA to the HA for a given HoA, often referred to as M-MIP (multi-homed MIP), is described in [9]. The M^f software architecture is using this technique in order to support soft hand-overs.

3. NETWORK SELECTION TECHNIQUE AND POLICY-BASED DECISION MODEL

Network selection in the M^f software architecture is using a fully mobile controlled hand-over (MCHO) scheme based on a simplified version of the *Relative Network Load* [10] metric.

Round-trip times (RTT) and jitter in RTT values are calculated on the BA messages forming an RNL metric representing a quality value for each access network. RTT and RTT jitter values are access technology independent and good indicators on congestion in networks and limitations in available bandwidth.

RTT jitter, being the variation in RTT, is calculated using formulas in RFC 3550 [11]. The formula is adjusted with a variable history window instead of using a fixed history window of 16 (as in RFC 3550) giving the following formulas:

$$RNL_n = \bar{z}_n + c J_n \quad (1)$$

$$\bar{z}_n = \frac{1}{h} RTT_n + \frac{h-1}{h} \bar{z}_{n-1} \quad (2)$$

$$RTT_n = R_n - S_n \quad (3)$$

$$D_n = R_n - R_{n-1} - (S_n - S_{n-1}) = (R_n - S_n) - (R_{n-1} - S_{n-1}) = RTT_n - RTT_{n-1} \quad (4)$$

$$J_n = \frac{1}{h} |D_n| + \frac{h-1}{h} J_{n-1} \quad (5)$$

where, S_i and R_i are defined as

S_i = the time of sending BU message i

R_i = the time of arrival of BU message i

h determines the history window for the weighted average calculations. For example, when $h = 5$, the most recent value will contribute to the calculated $\bar{z}_n$ and J_n values with 20%.

c determines the weight of the RTT in comparison to the RTT jitter value. For example, when $c = 5$, the RTT jitter value is contributing five times more to the RNL metric value than the RTT value does.

The variables $\bar{z}$, D , and J are initialized with the following values:

$$\bar{z}_0 = RTT_0$$

$$D_0 = 0$$

$$J_0 = D_1$$

The RNL metric is beneficial to use for its access network independence feature. The fact that no synchronized clocks are needed is also favoring this solution.

Before any handover decision is finally taken, the direction of the vertical handover is determined. Nasser et al. [12] defined an upward vertical handover as roaming to an access network with a larger cell size and lower bandwidth, and a downward vertical handover as roaming to an access network with a smaller cell size and larger bandwidth. Using those definitions, a handover is decided to take place when

$$S_{\text{new}} < S_{\text{old}} - a$$

for downward handovers where a is a positive constant used in order to avoid hysteresis. Upward handovers are decided to take place when

$$S_{\text{new}} < S_{\text{old}}$$

This asymmetric decision model is used in order to let handovers to access networks with high but unstable capacities wait until the policy value is significantly lower compared to the old access network with lower capacity but better coverage. When doing handovers to access networks with less capacity but better coverage, the handover decision should be executed immediately in order not to loose the connection. This is especially important at high speeds and steep cell edges.

The M^d software architecture is implementing the IETF's proposed model for policy handling [13] depicted in figure 1. The policy repository (PR), the policy decision point (PDP), and the policy enforcement point (PEP) are in the M^d software architecture all located in the MN.

Figure 1. IETF's model for handling of policies.

The overall aim of the policy model is to choose the best available access network taking the end-user's preferences on various variables into account. Therefore, a cost function has been defined using three parameters: monetary cost, power consumption, and network load. The policy function is defined as:

$$S_n = w_p \ln P_n + w_c \ln C_n + w_b \ln L_n \quad (6)$$

where

$$w_p + w_c + w_b = 1$$

S_n is the policy value for access network n and is the weighted sum of normalized policy parameters. P_n represents power consumption while C_n is the monetary cost for access network n respectively. Those parameters are dimensionless constants entered by the end-user in the M^d software architecture graphical user interface. L_n is the RNL metric for access network n and calculated according to the formulas above.

The network selection decision is taken whether to make a handover or not, and in the case of such a decision, to which new access network to transfer the connection to.

4. THE M^d SOFTWARE ARCHITECTURE

The M^d software architecture is composed of a HA software component and a MN software component building an overall architecture together with the access networks, the Internet backbone, and any correspondent node (see figure 2).

All traffic is sent via the HA in a bidirectional tunnel, hence no route optimization is used. Also, the MN uses collocated CoAs, meaning no Mobile IP support is needed in foreign networks. BU messages are sent prior to tunnel setup and transmitted out of band. This means they are being sent directly to the HA outside the tunnel. The BU message contains CoA, HoA, life time, sequence number, check sum, and flags.

Figure 2. Overall architecture.

4.1 M^d MN software component

The M^d MN software component is running on Windows XP using WinpkFilter 3.0 [14] for packet filtering. It is composed of functions for PR, PDP, and PEP respectively. Other functions handling routing, tunneling, and physical interfaces are also parts of the M^d MN software component shown in figure 3.

Figure 3. The M^d MN software architecture.

The M^d tunnelling mechanism is implemented using a virtual network interface on the MN and uses UDP datagrams in order to support NAT and firewall traversal at the cost of an UDP header

overhead. The solution basically provides connectivity to the home network via the virtual interface using IP communication. The virtual interface is configured with home network parameters where IP address is the HoA and the subnet mask is configured with a point-to-point mask. The interface appears in the Windows environment and provides direct communication to the home network, see figure 4.

Name	Type	Status	Device Name	Phone # or Host Address	Owner
Connected					
CDMA2000	Dial-up	Connected	AnyDATA CDMA USB Modem (PID 6501)	#777	System
WLAN	LAN or High-Speed Internet	Connected	Intel(R) PRO/Wireless 3945ABG Network Connection		System
M4	LAN or High-Speed Internet	Connected	Microsoft Loopback Adapter		System

Figure 4. Network connections in the MN.

The virtual network interface is a standard loopback adapter. The tunnel functionality captures packets on the virtual interface (i.e. the default gateway interface) and encapsulates them in UDP datagrams.

The M^d Policy Engine, being the PDP component in the M^d MN software component, calculates and updates policy values continually for all active interfaces. Information on what is the best interface is sent to the MIP Driver which, in turn, enforces the best interface to be used. The Policy Engine takes network selection decisions at a rate of twice the highest BU message frequency. What network selection decision that has been made can be viewed in real-time through the $M4$ Monitor window (see figure 5).

Figure 5. The M4 Monitor window.

The M^d routing functionality handles the routing table, i.e. adding and deleting entries in the Windows routing table. When a new route is added it is assigned a lower routing metric than automatically generated routes by Windows XP itself. This way a new route is preferred over existing ones. The M^d MIP Driver is set as the default gateway. A full match mask (255.255.255.255 for IPv4) is used in order to make sure the tunnelled traffic destined to the HA is sent through the selected interface.

The M^d interface functionality is a software entity, representing each physical interface selected in the M^d MN software's graphical user interface. Each physical interface runs a thread that handles BU messages sent out of band (i.e. they are not tunnelled). The BU message mechanism is also used to estimate network performance by measuring the round-trip time of BU messages. RTT and jitter in RTT values are used for RNL calculation.

The RNL calculation algorithm includes a filtering function to eliminate spikes in the RTT jitter. This action can be motivated

when one single BU message is lost due to some other cause than network performance degradation.

Figure 6. Network selection algorithm.

Filtering is accomplished by altering the weights for the sliding average calculation. If the jitter is below 2 milliseconds the weight is set to 0.9 to allow increases in performance to be detected rapidly. If the momentary jitter is significantly (150 milliseconds) higher than the calculated average, the momentary measurement is considered to be a possible random peak caused by a dropped BU message. Therefore only 1/100 of the value is included in the calculation. A counter keeps track of how many values that were filtered, and if more than two sequential values are significantly higher than the average during four calculations they are most likely caused by bad network performance and should be included in the calculation. The algorithm used for the calculations is depicted in figure 6.

Since high speed links with limited coverage, such as WLAN, are more likely to have rapid fluctuations in performance BU messages are sent with a higher frequency over those interfaces. For links with more stable characteristics but with lower data rates, e.g. UMTS and CDMA2000, BU messages are sent less frequently.

The M^d policy repository holds user preferences on weights for policy calculation parameters.

The graphical user interface allows the user to select what interfaces to be included in the mobility management handling and to set weights for each parameter on each access network. It is also possible to let the user exclude interfaces. Finally, the IP address of the HA is entered here. Figure 7 shows the M^d MN graphical user interface.

Figure 7. The M^d MN graphical user interface.

By using Internet Connection Sharing on the MN's virtual network interface mobile routing is also provided. This way multiple users may use the MN's global connectivity enabling e.g. flight connection services.

4.2 THE M^d HA software component

The M^d HA software component is running on Linux distribution Fedora core 5 [15], with kernel 2.6.15 that has to support IP forwarding and the universal TUN/TAP [16] virtual network kernel driver. The M^d HA software component is developed using ANSI C and is multi-threaded with one socket listening for incoming traffic and one socket used for outgoing traffic. One common socket handles BU messages.

The M^d HA software component is a combined tunneling end point, a router, and a simple server. It holds a binding cache with entries for each connected MN. When a BU message is received the HA checks if the MN is already in the cache and if not, it is added. The BU message is verified using a checksum calculated as the sum of byte values modulus 256. If the received BU message is incorrect it is discarded. Otherwise it is mirrored back to the MN as a BA message.

A TUN interface acts as a common tunnel endpoint for all MNs.

Outgoing traffic is decapsulated at the tunnel end-point and sent out on the home link.

Incoming traffic is captured in the HA using proxy Address Resolution Protocol (ARP) functionality. This is handled by making a published static ARP entry in the Linux kernel. The HA responds to ARP requests on the home link on behalf of the MN. When a MN is added a static routing table entry is made to route traffic destined for the MN directly to the TUN interface which in turn sends it to the M^d HA software. The captured packets are inspected to determine the destination MN, encapsulated and sent to the MN through the tunnel.

5. EVALUATION FRAMEWORK AND SCENARIOS

To evaluate the M^d software architecture, a live heterogeneous networking environment was set up. A WLAN (IEEE 802.11g) access point was installed and a commercial CDMA2000 (operating on the 450 MHz band) was used. The topology is shown in Figure 8.

Figure 8. Evaluation topology.

To study handover performance in detail a 120 second Voice over IP (VoIP) call of 6 kbps two-way traffic was simulated using the Iperf [17] traffic generator.

The MN was installed in a car traveling at 30 km/h starting in a point outside the WLAN cell. After approximately 25 seconds the car reached the cell edge of the WLAN cell where a handover was expected to take place. After an additional 40 seconds the WLAN cell edge was reached again where a handover back to the CDMA2000 system was expected to take place.

Outdoor antennas were used both for WLAN and CDMA2000. The WLAN access point was configured to a capacity of 54 Mb/s giving 27 Mb/s in practice. The CDMA2000 network offered bandwidths between 0.5 and 1 Mb/s downlink and 64 kb/s uplink.

The experiments conducted used $h = 5$ (in formulae (2) and (5)), $c = 5$ (in formula (3)) and $a = 1$ (the hysteresis avoidance constant). The cost and energy consumption weights were set to zero on all interfaces in order to evaluate a policy based only on RNL. The BU messages were sent with a one second interval on the CDMA2000 interface and with 100 milliseconds as interval on the WLAN interface. The selected intervals relate to how fast the MN reacts to variations in RTT and RTT jitter where shorter intervals improve reactivity at the cost of a higher network overhead. A shorter interval is motivated especially for highly fluctuating access networks with high throughput like WLAN.

To study user-perceived quality of service parameters the NetAlly software [18] from Viola Networks was used. This part of the experiment took place in a laboratory environment where handovers were forced. The G.723.1 codec using approximately 6 kbps was studied for WLAN, CDMA2000, and the combination of

Figure 9. Results from hand-over performance studies.

those networks combined in M^f with 50% of the time spent on WLAN and the rest on CDMA2000. Two-way calls of 60 seconds were studied.

A third study on downloads large amounts of data was also performed in the laboratory environment. The highest possible throughput is measured with Iperf using M^f with CDMA2000 during 30 seconds and WLAN during 30 seconds. The purpose of this study was to see how download times were affected of the introduction of M^f using a mixture of high and low performance access networks.

6. RESULTS

To evaluate the effectiveness of the architecture a number of parameters were studied. Throughput, delay, delay jitter, packet losses, and packet reordering were all studied by looking at the output from Iperf. In figure 9 a graph of one experiment showing received bandwidth (black dotted line), jitter for selected access network at each time (cerise dashed line), and packet loss rate (yellow straight line). The calculated policy values for each access network are also plotted (light green and blue lines respectively) as well as information on what interface that was selected at each time (dark green and brown lines respectively). Most notably, packet losses vary from 0 to 3 lost packets for each test in the first study. The experiment shown in the graph had no packet losses at all.

RTTs were typically 10 milliseconds for WLAN and 150 milliseconds for CDMA2000 while RTT jitter were typically 10 milliseconds for WLAN and 20 milliseconds for CDMA2000 respectively. With the weights and constants used, the policy value for WLAN varied according to networking conditions with a minimal value around 3.0 while the policy value for CDMA2000 stayed close to 6.0 constantly. Furthermore, the data presented in the graph shows that received bandwidth is very

stable during handovers and that the idea of using multihomed Mobile IP makes soft handovers work in reality. VoIP applica-

Table 1. Results from study of user-perceived quality of services parameters.

	MOS	R-value	Jitter	Delay	Loss
CDMA2000 (downlink)	3.4	67.2	12.5	104.8	0.03
CDMA2000 (uplink)	2.8	54.9	15.4	169.3	0.02
WLAN (downlink)	3.9	76.1	0.8	9.1	0.02
WLAN (uplink)	3.9	76.1	0.5	9.7	0
M4 (downlink)	3.7	72.9	3.8	71.2	0.01
M4 (uplink)	3.6	69.6	4.4	70.7	0.01

tions using the G.723.1 codec can handle approximately 1% packet loss without problems. The highest packet loss rate we experienced during measurements was 3 out of 2249 lost packets, corresponding to a packet loss rate of 0.13%. The highest number of consecutive packets lost during handovers was 2 indicating that quality of experience in the studied VoIP application is only affected to minimal extent. The results from the second study where user-perceived quality of service parameters were measured are presented in Table 1.

As a baseline to make comparisons against, Mean opinion score (MOS) and R-values, as well as application-layer jitter, delay, and loss rate, are shown for WLAN and CDMA2000 in uplink and downlink directions respectively. It should be noted that the best possible MOS value for the G.723.1 codec is 3.9 which was also the measured value for WLAN in both uplink and downlink directions.

MOS and R-values are also shown for the combined case where WLAN and CDMA2000 were used together with M^d . The expected result would have been to observe an average of the values observed for the two included access technologies. However, as shown in the table, results are significantly better in the uplink direction.

The results from the third study focusing on the amount of data being downloaded using maximal throughput are found in figure 10.

Figure 10. Results from study of amount of data being downloaded using maximal throughput.

Not surprisingly, there is a significant gain of letting a mobility management function like M^d letting the user to take advantage of high capacity access networks like WLAN when such technology is present.

To summarize, mobility scenarios using only one single access technology like CDMA2000 less throughput will be experienced compared to usage of M^d where access networks offering higher capacities will be used on-the-fly. In the case of only using WLAN the communication will break when leaving the WLAN cell disrupting the communication.

7. RELATED WORK

There is a large amount of related work in the area of mobility management architectures, network selection algorithms, quality of service management methods, policy-based behaviors, and implementation prototypes supporting heterogeneity in upcoming fourth generation mobile networks, IMT-Advanced, and the “Always Best Connection” vision.

Yiping et al. [19] proposed a new architecture including a access discovery mechanism, personalization of network selection, and mobility management based on Mobile IP.

Nguyen-Vuong et al. [20] proposed an architecture using multihomed Mobile IP including a network selection algorithm focusing on power-saving interface management.

Puttonen et al. [21] proposed using link layer information improving vertical handovers.

Several network and interface selection algorithms have been proposed. Puttonen et al. [22] suggested using algorithms combining Simple Additive Weighting, weighted product, and TOPSIS (Technique for Order Preference by Similarity to the Ideal Solution) while Song et al. [23] suggested using Analytic

Hierarchy Processing and Grey Relational Analysis. Furthermore, Isaksson et al. [24] proposed using Multi-Criteria Decision Making based on the Analytic Hierarchy Process for network selection decisions.

Papers addressing subjective QoS of multimedia applications in real multi-access network environments are less frequent. However, Sutinen et al. [25] performed a case study in assessing subjective QoS of a mobile multimedia web service. Promising results from an empirical task-based user evaluation of the end-user QoS were presented.

This paper suggests a mobility management architecture based on multihomed Mobile IP, a network selection algorithm based on end-user’s preferences handled through policies and a network layer metric calculating relative capacities among multiple access networks. Also, the software architecture behind M^d is presented in detail.

8. DISCUSSION AND FUTURE WORK

We have proved that the ideas and concepts behind M^d work in real life situations where multimedia applications are used. Handovers are handled in a seamless way and the performance of vertical handovers is good with respect to packet losses.

Total user-perceived quality of service is better than only using CDMA2000 and mobility is improved compared to only using WLAN. Furthermore, users of applications downloading large amounts of data can leverage a wide range of wireless networks making download times shorter. The M^d software architecture could be seen as a concrete implementation of the vision “Always Best Connected”.

We intend to extend the developed software leveraging the soon upcoming concept of media-independent handover services being standardized by the IEEE 802.21 working group [26]. The network selection techniques and algorithms will also be further optimized to adapt to varying networking conditions and access technologies.

Finally, we are currently working on an implementation of the M^d software architecture for the Windows Mobile platform. We will study overall performance and the requirements to make sure that such a platform is suitable for the M^d architecture. Furthermore, we also intend to add support for load balancing enabling concurrent use of more than access network.

9. ACKNOWLEDGMENTS

The work presented in this article is based on results from the HybriNet@Skellefteå [27] project supported by Skellefteå Kraft.

10. REFERENCES

- [1] E. Gustafsson, A. Jonsson, Always best connected, IEEE Wireless Communications, Volume 10, Issue 1, pp. 49 – 55, February 2003.
- [2] I.F. Akyildiz, J. McNair, J.S.M. Ho, H. Uzunalioglu, W. Wang, Mobility Management in Next-Generation Wireless Systems, Proceedings of the IEEE, Volume 87, pp. 1347 – 1384, August 1999.
- [3] H. Schulzrinne, E. Wedlund, Application-layer mobility using SIP, Mobile Computing and Communications Review archive, Volume 4, Issue 3, pp. 47 – 57, July 2000.

- [4] R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, V. Paxson, Stream Control Transmission Protocol, IETF, RFC 2960, October 2000.
- [5] I. Aydin, W. Seok, C. Shen, Cellular SCTP: A Transport-Layer Approach to Internet Mobility, In Proceedings of the 12th International Conference on Computer Communications and Networks, pp. 285 – 290, October 2003.
- [6] W. M. Eddy, At what layer does mobility belong?, IEEE Communications Magazine, Volume 42, Issue 10, pp. 155 – 159, October 2004.
- [7] C. Perkins (ed.), IP Mobility Support for IPv4, IETF, RFC 3344, August 2002.
- [8] D. Johnson, C. Perkins, J. Arkko, IP Mobility Support in IPv6, IETF, RFC 3775, June 2004.
- [9] C. Åhlund, R. Brännström, A. Zaslavsky, M-MIP: Extended Mobile IP to Maintain Multiple Connections to Overlapping Wireless Access Networks, In Lecture Notes in Computer Science, Volume 3420/2005, pp. 204 – 213, April 2005.
- [10] C. Åhlund, R. Brännström, A. Zaslavsky, Traffic load Metrics for Multihomed Mobile IP and Global Connectivity, Telecommunication Systems, Volume 33, Number 1 – 3, pp. 155 – 185, October 2006.
- [11] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, RTP: A Transport Protocol for Real-Time Applications, IETF, RFC 3550, July 2003.
- [12] N. Nasser, A. Hasswa, H. Hassanein, Handoffs in fourth generation heterogeneous networks, In IEEE Communications Magazine, Volume 44, Issue 10, pp. 96 – 103, October 2006.
- [13] R. Yavatkar, D. Pendarakis, R. Guerin, A Framework for Policy-based Admission Control, IETF, RFC 2753, January 2000.
- [14] WinpkFilter, version 3.0, www.ntkernel.com.
- [15] Linux, Fedora version 6, fedoraproject.org.
- [16] vtun.sourceforge.net/tun.
- [17] Iperf, version 1.7.0 and version 2.0.2, dast.nlanr.net/Projects/Iperf.
- [18] NetAlly Base System, SMB Edition, version 5.2.31, www.violanetworks.com.
- [19] C. Yiping, Y. Yuhang, A New 4G Architecture Providing Multimode Terminals Always Best Connected Services, In IEEE Wireless Communications, Volume 14, Issue 2, pp. 36 – 41, April 2007.
- [20] Q.-T. Nguyen-Vuong, N. Agoulmine, Y. Ghamri-Doudane, Terminal-Controlled Mobility Management in Heterogeneous Wireless Networks, In IEEE Communications Magazine, Volume 45, Issue 4, pp. 122 – 129, April 2007.
- [21] J. Puttonen, G. Fekete, J. Makela, T. Hamalainen, J. Narikka, Using link layer information for improving vertical handovers, In Proceedings of IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. Volume 3, pp. 1747 – 1752. September 2005.
- [22] J. Puttonen, G. Fekete, Interface Selection for Multihomed Mobile Hosts, In Proceedings of IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 1 – 6, September 2006.
- [23] Q. Song, A. Jamalipour, A network selection mechanism for next generation networks, In Proceedings of IEEE International Conference on Communications, Volume 2, pp. 1418 – 1422, May 2005.
- [24] L. Isaksson, M. Fiedler, Seamless Connectivity in WLAN and Cellular Networks with Multi Criteria Decision Making, In Proceedings of 3rd EuroNGI Conference on Next Generation Internet Networks, pp. 56 – 63, May 2007.
- [25] T. Sutinen, T. Ojala, Case Study in Assessing Subjective QoS of a Mobile Multimedia Web Service in a Real Multi-access Network, In Proceedings of Thirteenth International Workshop on Quality of Service, Passau, Germany, pp. 298 – 312, June 2005.
- [26] IEEE P802.21/D7.1, Draft IEEE Standard for Local and Metropolitan Area Networks: Media Independent Handover Services, www.ieee802.org/21. August 2007.
- [27] www.hybrinet.org.