Increasing communication availability with signal-based mobile controlled handoffs

D. Forsberg, J. T. Malinen, J. K. Malinen, Hannu H. Kari
TSE-Institute, Telecommunications and Software Engineering
Laboratory of Information Processing Science
Helsinki University of Technology, P.O. Box 9700
FIN-02015 HUT, Finland
E-mail: {dforsber,jtm,jkmaline,hhk}@cs.hut.fi

Abstract— **We present an experiment on mobile-controlled handoffs (MCHO) where the signal-based information is used for determining the best communication partner. We argue that by a simple MCHO we can get a straightforward and scalable support for fast, seamless, and glitchless handoffs in IP-based wireless access networks. With glitchless we mean that no packets are lost during a handoff. We present a modular solution available as a part of our hierarchical mobility management.**

Keywords— **frequent handoffs, mobile controlled handoff, handoff signaling, hierarchical tunneling, IP routing, Mobile IPv4, mobility management, wireless mobility management.**

I. Introduction

Various portable computing devices such as laptops, handheld computers, and other personal digital assistants (PDAs) with networking capabilities increase the demand for seamless communication both in wired and wireless networks. Increased use of multimedia content with mobile computers makes seamless communication an essential and required feature expected in mobile connections. Practical mobility management should provide a seamless handoff where the user does not observe communication disruptions.

Internet Protocol (IP) [1] based mobility management implementations have traditionally ignored link-layer information with this respect. However, many link-layer technologies provide signal based information that can be used by the network layer mobility control.

Traditionally, a user does not need to know the communication partner that provides the connection to the Internet. This is convenient but causes some restrictions to the system. When several possibilities for connecting to the Internet are available the user may want to choose or change the communication partner dynamically. For example, the cost, bandwidth, and available services may cause the user to change the communication partner, or more precisely the policy for choosing the gateway to the Internet.

In Section II, we explain the communication availability problem with Mobile IP in wireless local area networks (WLANs) and the criteria for an efficient solution. We discuss the related work in Section III. The discussion is divided into two parts—network level handoffs and handoff prioritization and policies. We present our solution in Section IV with similar division and describe the deeper aspects of the solution and the implementation in Section V. We evaluate the system and different performance tests in Section VI. Conclusions follow in Section VII.

II. Communication availability

A. Hierarchical Mobile IP

Mobile IP [2], [3] is an addition to the IP that allows nodes to continue processing datagrams no matter where they happen to be attached to the Internet. Control messages allow the involved IP nodes to manage their routing tables reliably.

A *mobile node* (MN) is a host that can change its point of attachment from one network or subnetwork to another without changing the IP address. It may continue to communicate with other Internet nodes called *correspondent nodes* (CN) at any location using the same home IP address, assuming link-layer connectivity to the point of attachment is available.

A *home agent* (HA) is a host in the *home network* (HN) of an MN. It tunnels datagrams for delivery to the MN when the MN is away from home and maintains current location information for the MN. A *foreign network* (FN) is any other network than the HN. A *foreign agent* (FA) is a host in an FN. It provides routing services to the registered MNs in the FN. FAs deliver decapsulated datagrams to the MNs tunneled by the HA and they may also act as default routers for registered MNs. Both HAs and FAs are called *mobility agents* (MA).

B. Improving communication availability

We will consider the problem of communication availability under signal quality (SQ) based handoff management in WLANs with Mobile IP.

The **communication availability** is readiness for usage [4]. Communication availability is defined as the ability of the server to deliver the service that fulfills and maintains the requirements of service quality to the MN. We measure this both from the point of mobile users and the networks.

Signal quality gives information about the ability to transfer information in the wireless data path.

Handoff is an event that occurs between communicating MAs and an MN. At least three nodes are involved in a handoff; one is the MN and the other two are the old and the new MA. Handoff starts when the decision for changing the MA is made and finishes when the MN has changed the MA. Thus, a handoff is the process during which a node is "handed over" between two designated MAs [5]. During handoff the MA that is responsible of routing the packets to and from MN is changed.

In *soft handoff* [6], [7] an MN can communicate with both the new and the old MA. This is not possible in *hard handoff* [7], because the MN can listen only one MA at a

time.

Network controlled handoffs (NCHO) are handoffs where the network makes the handoff decision. In *mobile assisted handoff* (MAHO) the MN makes the handoff decision together with the network and in the *mobile controlled handoff* (MCHO) the MN decides itself when to make handoffs. [8]

One method to separate MCHOs is to divide them into *forward* and *backward handoffs* [7]. In backward handoff the MN sends the handoff request to the current MA. In forward handoff MN initiates the handoff by sending request to the new MA.

Wireless network interfaces can have different cell sizes, e.g., in-room, in-building, campus, metropolitan, and regional. *Wireless overlay networks* are a combination of wireless networks that have different cell sizes. If the wireless network interface and the cell size is changed during the handoff process, this is a *vertical handoff* [9], otherwise it is a *horizontal handoff* where the old and the new MA uses same radio technology. Depending on the cell size, handoffs can be classified into *macro, micro*, and *pico handoffs*. Macro level cells have at least several kilometers long radius and lower bandwidth than micro cells or pico cells. The radius in micro level cells is in tens or hundreds of meters and with pico level cells the radius is in meters.

Handoff management includes the procedures and required information needed to make handoffs. In this paper it is done in the network layer. The handoff management problem can be divided into two subproblems. In MCHO the MN has to be aware of available MAs and the services they offer. This can be expressed as *mobility agent detection* (MAD) problem. Secondly, the *mobility agent selection* (MAS) problem arises when the MN detects several MAs and requires communication availability. One of the detected MAs has to be selected. Communication availability requires routing of the signaling messages that the handoff management must take care of.

C. Criteria

The criteria for the solution should be fulfilled at least with one MN in the system.

- **Minimal impact on data transmission:** In the ideal situation the handoff is transparent and has no impact on the data transmission. This means that the throughput and latency of the routed packets are not affected by the handoff management. Sessions are maintained and no packets are lost. Minimal impact on data transmission can be evaluated with throughput analysis and measurements of signaling latency and packet loss.
- **Tolerance for congestion:** The solution should tolerate congestion occuring when the data path is fully used and there is demand for more capacity. The word "tolerate" means that the MN and the MA are able to communicate with each other. This can be measured with signaling latency under heavy load on the data path generated by the MAs and the MN.
- **Efficiency:** The solution should be efficient. An efficient solution uses the radio bandwidth sparingly. Signaling messages are small and the signaling itself is lightweight. This can be measured with the number and size of the signaling messages required in the handoff management protocol.
- **MN should use the MA with best communication availability:** To measure communication availability, readiness of the service must be measured. Readiness of the service is related to the throughput and packet latency of the communication path. Smaller latency indicates better readiness. Higher throughput gives better readiness for services that require more bandwidth. Throughput and latency can be used to measure this criteria.
- **Independence of the underlying radio technology:** The solution should not be dependent on the physical characteristics of the underlying radio technology. This can be measured by identifying the layer in which the solution is functional.
- **Modular node selection system:** Flexibility to add new and modify existing node selection policies affects most the implementation. *Node selection policy* is a set of rules that affect the handoff decision. The implementation should be modular and easy to improve. This involves clear interfaces between modules and data flows.

III. Related work

Handoffs have been studied widely in recent years. G. P. Pollini presents an overview of published work on handoff performance and control [10]. He also discusses current trends in handoff research. Furthermore, he presents different handoff methods based on signal strengths. Challenges in seamless handoff design in mobile multimedia networks are handled by L. Taylor et al. [11].

Mobile IP is not specifically planned to support micro mobility [3] and it has not been considered to be a good solution for network-level micro mobility [12], [13], [14], [15]. Thus, several micro mobility proposals with and without Mobile IP have been introduced [16], [17], [18]. M. Stemm and R. H. Katz described vertical and horizontal handoffs [9]. C. Toh et al. presented different handoff protocol design issues [19].

A. Network level seamless handoffs

If the user or a program that uses the network bandwidth does not notice the handoff by only examining the data stream over the network, the handoff is said to be *seamless*. In a *glitchless handoff* delays due to the handoff are eliminated from the data stream. Multicast and buffering are the most used methods to provide seamless and glitchless handoffs. R. Cáceres and V. N. Padmanabhan describe a buffer-based solution with four-packet buffer in the access point (AP) [13].

K. Brown and S. Singh researched User Datagram Protocol (UDP) [20] for mobile cellular networks. They use Mobile IP together with buffered UDP packets and achieve a 50% increase in throughput with M-UDP compared to UDP [12]. Bakre and Badrinath published a similar analysis [21] of Transmission Control Protocol (TCP) [22]. They split the TCP connection into wireless and wired parts to get better throughput.

C. Perkins and K-Y. Wang present a scheme for optimized *smooth handoffs* [23]. They use buffering with Mobile IP as a basis for the handoff. FAs buffer packets for MNs and when the MN switches FAs, the old FA is signaled to send the buffered packets to the new FA

which then forwards the packets to the MN. Packet identifiers are used to eliminate duplicate packets sent to MN. Packet buffer is required for every MN and in multiple APs. This increases the requirements for resources and decreases the scalability of the system.

K. Keeton et al. present an incremental reestablishment scheme, which modifies an existing connection by establishing only the portion of the channel between the AP and the MN where the old and new channels would diverge [24]. They also present multicasting support for handoffs.

Jayanth Mysore and Vaduvur Bharghavan describe a multicasting-based solution for handoffs [15]. Every MN has a unique multicast address and packets destined to MNs have this multicast destination address. Packets from the MN have unicast destination addresses. Neighbor multicasting routers join to the same multicast address as the MN. When the MN initiates handoff with a new AP it is already in the multicasting address of the MN and thus the handoff can be made seamless.

C. L. Tan et al. describe a fast handoff scheme for wireless networks using a multicast based handoff [14]. They describe a *domain FA* (DFA) which assigns a multicast address unique within its domain to each MN. The domain FA has logically many APs in the lower level. Thus, the approach can be seen to have a two level hierarchy. The AP, to which the MN is connected, has joined to the multicast group of the MN and actively forwards packets to the MN. Adjacent APs have also joined to the same multicast group but do not send packets to the wireless network.

Further, the Ph.D. thesis of S. Seshan [25] and the paper of H. Balakrishnan, S. Seshan, and R. H. Katz [26] favor multicast-based handoff solutions. They also introduce a *snoop module* that listens TCP traffic between the AP and the MN. The idea of the snoop module is to resend packets lost between the MN and the AP by monitoring the acknowledgments to TCP packets.

The drawback in multicast solutions is that multicasting has to be supported by the routers and the network bandwidth is wasted since the data stream is duplicated to several APs. The APs have to allocate resources for every MH that is directly connected to it or to the adjacent APs. Thus, resources are not used efficiently. The buffering in APs may also affect the packet routing latency between a CN and an MN.

If multicasting and buffering are used together the resource requirement becomes more demanding. The wired network has to be capable of handling more bandwidth than the wireless network. This is not a problem if the bandwidth difference is considerably high. Today, especially in WLANs the bandwidth is increasing. The cell structure of the wireless networks and the many radio channels make it possible to overload the wired core network multiple times the wireless bandwidth.

B. Handoff prioritization and policies

S. Tekinay and B. Jabbari describe a measurement prioritization scheme for handoffs in mobile cellular networks [27]. The prioritization is made in the wired network side. They also illustrates how the SQs can be used to prioritize handoffs to get better performance from the system. *Adaptation and Mobility in Wireless Information Systems* by R. H. Katz [6] presents generally the problem of improving communication through situation awareness. N. D. Tripathi et al. discuss the handoffs in cellular systems [28]. They state that a handoff algorithm with fixed parameters cannot perform well in different system environments. Handoff prioritization schemes are described and the prioritization is based on the SQ.

Different prioritization schemes are related to the handoff policies. H. J. Wang et al. introduce handoff policies that take into consideration many different aspects of the handoff [29]. For example, performance, power consumption, and cost can be measured and compared to calculate the best wireless system at any moment. Wang et al. introduce a policy calculation function that uses different cost parameters as input and produces the total cost of the network. The total cost can also be thought as the priority for the network. In addition, APs may also have different priorities which Wang et al. do not handle in their paper. They separate the networks but do not separate different APs in the same network.

Wang et al. also describe a handoff synchronization problem [29]. If several MNs are using the same policy in the same place, they may change the network simultaneously and affect the dynamic parameters of the network. These parameters affect the policy and, thus, may cause MNs to oscillate between different networks. This can also happen between APs in the same network. As the solution for this problem, Wang et al. introduce a *stability period*, a time period that the MN waits before initiating a handoff. The reason for this time period is stabilization. This kind of approach adds latency to the handoff.

IV. Mobility agent switching

Mobile IP proposes a macro mobility solution for the mobility problem. Therefore, we are using it as a basis for the solution to the problem described in Section II. More precisely, the HUT Dynamics Mobile IP is used for the mobility management to achieve communication availability. The hierarchical structure of HUT Dynamics provides an efficient platform for fast macro mobility [30].

The MN makes handoff decisions and the FA hierarchy assists the MN in the handoff management. This makes it possible for the MN to use different handoff policies independently of the MAs. Thus, the solution uses MCHO scheme. The MN can compare different MAs and gather information about them from agent advertisements. MNs may use different policies and criteria to choose the communicating MA. The disadvantage of the MCHO is that MN has to support handoff management. This includes ability to make handoff decisions and initiate handoffs. With NCHOs the access network is responsible for handoffs and the MN can be made simpler.

The whole idea for mobility lies on the *mobility management* system which includes also the handoff management. We divided the system into *MA detection* and *MA selection*. Fig. 1 shows the data flow in the mobility management system. MA detection includes a *signal quality sensor* and *signal quality collector*. A *node selector* and a *signal quality analyzer* belong to the MA selection part of the handoff management. *Signal quality history* data is used by both the MA selection and the MA detection.

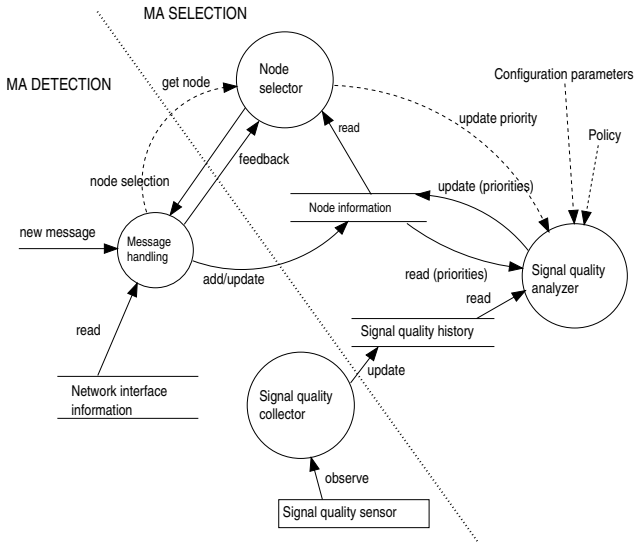

Fig. 1. Data flow

Node information data contains information about FAs that the MN has received in the agent advertisements. This information storage is used and updated by the MA selection component. The concept is clarified below.

A. MA detection

The expiration of the agent advertisements in Mobile IP provides a method for MA detection. An MN receives advertisements from MAs and knows which MAs are available. The agent advertisement lifetime is the maximum length of time that the advertisement is considered valid in the absence of further advertisements [3]. It is used to get some resolution for mobility in time. If the MN moves it can detect movements from the advertisements. Either it does not receive agent advertisements from certain FAs anymore, or advertisements itself contain some information from which the MN can notice the movement. This is sufficient in wired static networks, where the MN is switched from one subnet to another, but in WLANs the movement and, thus, mobility is different. In a WLAN the mobile user can move inside the wireless cell of the MA without losing connection to it. The MA is acting also as an AP for the MN. If the MN is in range of several MAs it has to decide which one to use as a gateway for the communication with CNs. When the mobile user moves outside the current MA, the MN has to initiate a handoff with a new MA.

Signal quality collector

If an SQ sensor is available the solution has much more possibilities. The SQ sensor monitors the link quality to other nodes in range. SQ is measured from received packets and is related to the data throughput between the signal source and the receiver.

An SQ collector is a component that reads in the SQ values from the SQ sensor. Values are converted into a more general form and stored in a *signal quality history* data storage. Conversion is needed to support different kinds of SQ sensors. The collector makes the conversion so that the data storage contains comparable values from possibly different SQ sensors. This simplifies the SQ analyzer because it does not have to be aware of different SQ sensors.

Message handling

Connections, location updates and disconnects are handled in the *message handling* component. It receives all agent advertisement messages, parses them and saves the information into a *node information* data storage. Message handling communicates with the node selector which controls the location update decisions. When location update or connection is made, the message handling component sends the registration request to the selected MA and handles the received registration reply from the MA.

B. MA selection

Mobility agent selection is based on priority comparison. Priorities are modified and analyzed in the SQ analyzer. Node selector makes the final decision based on the priority and currently used MA. Different *MA selection policies* are used to help the decision.

There is only one priority variable for each available MA. Priorities are based on the SQ values received via the interfaces with an SQ sensor. With interfaces that do not have an SQ sensor a specific interface priority is used as a basis for the MA priority. The whole monitoring system is built upon the idea that different MAs can be separated by some means related to the communication availability. Priorities have been chosen to separate the MAs in the monitoring system because they are flexible and abstract enough.

Priority balancing (PB) technique compares the best MA candidate that has the highest priority to the priority of the current MA. If PB occurs the priority of the current MA is set to the same value as the value of the best MA. When the best MA candidate has same priority than the current MA, the node selector does not make a decision to change the MA. PB occurs if the compared priorities are close enough.

Priority decreasing technique (PDT) decreases the MA priority with a certain percent value in the MA selection system. PDT uses triggers to modify the *degradation percent value* used to degrade the priority for the MA. The degradation percent value is increased every time the PDT is triggered, until the maximum of hundred percent is reached. The function for the degradation percent value that the PDT uses is exponential. *Priority increasing technique* (PIT) is used in addition to PDT. It has an opposite effect to the PDT. The value of the degradation percent variable is decreased with PIT and increased with PDT. PIT is also trigger-based. It multiplies the degradation percent value with a constant fraction when triggered. *Configurable interface priorities*, SQ values, and SQ averages affect the priority of an MA, but also priority balancing, PIT, and PDT are used to enhance the MA selection system.

Node selector

Node selector is a simple component that compares the priority values. It communicates with the message handling module and decides whether the MN should change the MA. If change decision is made, the node selector gives

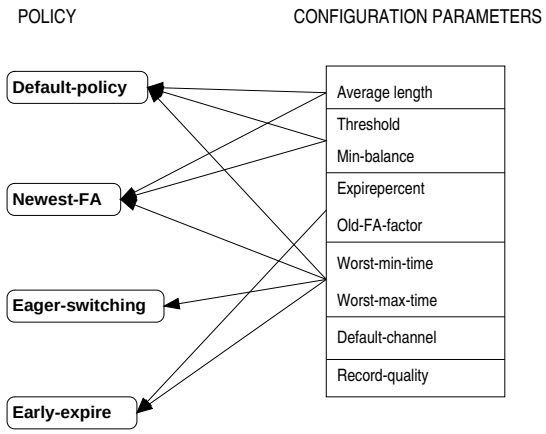

Fig. 2. Policy and configuration parameter relations

information about the node it has selected to the message handling component.

Message handling module provides feedback for the node selector. If the registration process is unsuccessful the node selector has to decide what to do with the problem. The MA cannot be used if the registration fails. After a time period the MA may become functional and the MN should be aware of it in order to use it. The possibility to separate functional and nonfunctional MAs increases the fault tolerance, efficiency, and functionality of the system.

If registration fails through the MA, the PDT is used to decrease the priority of the MA. Every time the MA sends an agent advertisement, PIT is used for that MA. This enables MAs to become slowly available again. This technique makes the system more robust since it recovers from small temporary failures in the MAs or the network.

Signal quality analyzer

A *signal quality analyzer* is the main component for the MA selection process. It modifies and prepares the priorities for the node selector, which finds the MA with highest priority and handles the registration process feedback.

C. Selection policies

The node selector uses certain rules to make selection decisions. The set of rules that affects the node selection process is called a *policy*. Comparisons and selections are based on the node priority, which the policies modify to achieve the needed results. The node selector picks up the node that has the highest priority.

We use currently four different policies for the node selector: *eager-switching, newest-fa, early-expire*, and the *default-policy*. The SQ analyzer contains some configurable parameters that are related with policies. Fig. 2 illustrates the relationships between configuration parameters and policies. It also shows how the parameters are conceptually related together.

With *eager-switching* the node selector takes the MA with the highest priority and does not calculate any averages from the link quality values. This means that the MA with the immediate highest SQ is used. Depending on the SQ sensor characteristics the SQ values for the FAs can vary even when the MN is not moving at all. Thus, frequent location updates are characteristic for this policy.

Every agent advertisement has a lifetime that starts from zero when a new agent advertisement is received from the MA. The default policy is to use the agent advertisement lifetime expiration entries from the node selector. *Early-expire* policy uses *expiration-time* configuration parameter to calculate the validity for MAs in the node selector. If the entry becomes older than the expiration-time, the *old-FA-factor* percent value is used to degrade the priority.

Newest-FA policy selects always the most recently detected MA and it acts like the default policy when no new FAs are detected. Consider a situation where an MA is in an area where no other MAs are heard. The MA can be in a different radio channel than other surrounding MAs or there can be a wall between the MAs that does not pass the radio waves through. The mobile user can enter this area very quickly from an area where many MAs are heard. This can happen when the mobile user changes radio channel or walks around a dense wall. Due to the nature of the expiration process for the MAs, the node selector will remember the old FAs when mobile user has entered the new area.

When MN detects the FA in the new area, newest-FA policy sets the priority for this FA at maximum and thus the node selector will select it. After the second agent advertisement from the new FA, the old FA entries may not have expired yet. The SQ value is used as a basis for the priority for the new FA and at this point it may become lower than the priorities of the old nonexpired FAs. This brings up a problem for the MN. After the second advertisement from the new FA the SQ analysis of the old FAs may still be valid, even if they are not actually reachable. If the node selector selects one of the old FAs, it is clearly a mistake since the MN cannot communicate with it. Solution for this problem is to set the priorities of the other FAs to the lowest possible, when new FA is detected. Old FA priorities are restored with new agent advertisements from each of the old FAs. This prevents MN from registering to the old FAs that may not be reachable when new FAs are detected. On the other hand, if some of the old FAs are still reachable and heard, the priorities will become normal with the next agent advertisements from these FAs.

The *default-policy* uses averages of the last received SQ values to calculate priorities for different FAs. The number of SQ values used in the average calculation is configurable. The three other policies can be combined with the default-policy, since they have slightly different effects. Different combinations make the system more flexible.

V. HUT Dynamics Mobile IP

The HUT Dynamics Mobile IP [30] system has been developed at Helsinki University of Technology (HUT). It is a scalable and hierarchical Mobile IP implementation for the Linux operating system [31].

In a hierarchical Mobile IP several FAs are put into FNs, so that the FAs make up hierarchical structures (Fig. 3). The registrations need to travel only a minimal distance when the MN has already registered via the FA hierarchy. HUT Dynamics supports arbitrary number of FAs and hierarchy levels.

The tunnel is established between the HA and the reg-

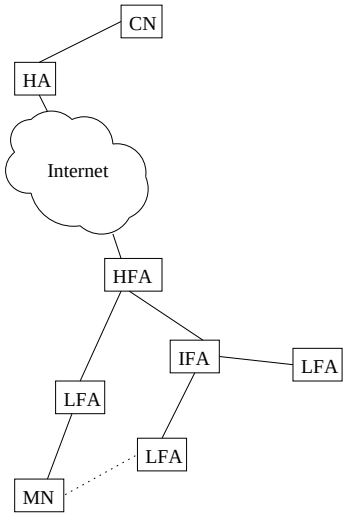

Fig. 3. Hierarchical foreign agents

istered location of the MN. Tunnel is built with segments between the HA and the root FA, between each FA in the path down to the *lowest FA* (LFA). In *FA decapsulation* mode the LFA decapsulates the encapsulated packets and sends them directly to the MN. A tunnel may also continue down to the MN. This is called *MN decapsulation* mode, since MN decapsulates the tunneled packets.

If MN decapsulation is used the MN needs a *co-located care-of-address* (CCOA) in the FN. In MN decapsulation mode the tunnel endpoints are between the MN and the HA. With FA decapsulation the home IP address of the MN is sufficient for registration and tunnel endpoints are between LFA and HA.

When MN moves to the FN the *registration protocol* (RP) is used for registration with HA. RP is hierarchical and the mobility binding is created through the FA hierarchy step by step. Each FA on the path from MN to the HA examines if it already has a binding for the specified MN. This allows them to perform local location updates. For a new registration, the protocol reaches the HA which then confirms the mobility binding creation [30].

An *access point* (AP) is an entity that has the station functionality and provides access to the distribution services, via the *wireless medium*. When MN moves to another place in the FN and connects to a new AP, the RP is executed again. This time a *localized location update* [30] is performed in the FA hierarchy. This includes creating new tunnels, if needed, to the new path location of the MN. *Switching FA* (SFA) is the FA that replies to the registration request of the MN. Localized location update is illustrated in Fig. 4.

A. Policy-based MA selection and detection

The interface priority is combined with the priorities of each MA in the MA selection system. The Monitor component can be configured to give a weight for the interface priority. By default, the interface priority is used as the default priority for FAs that are reachable via interfaces without an SQ sensor. With wireless interfaces that are capable of measuring SQ values no interface priority is used but the SQ value is used to produce the priority.

Different wireless interface cards may produce different

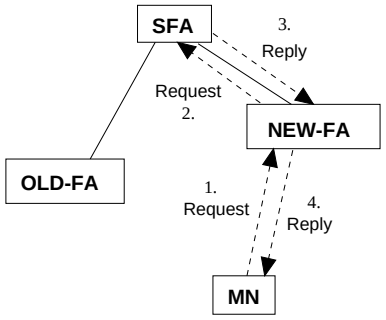

Fig. 4. Local registration update

signal quality values. Additionally, in some cards the SQ value may change rapidly while in other cards it is quite steady. For this purpose, the SQ analyzer normalizes the priorities to a range from 0 to 100. The maximum value is queried from the driver and is used to normalize the values.

After the normalization all MAs that are reachable via different devices can be compared roughly since the normalization does not take into account the different characteristics of the SQ values of different cards. The scale may not be linear compared to the real signal to noise ratio. The normalization is not distorted if the scale is linear. But if the scale is not linear the normalization distorts the SQ values. For accurate comparisons the compared normalized values should be equally distorted.

B. Configurable Monitor

Monitor contains the FA selection mechanism. It is used with wireless interfaces that have SQ sensor. Monitor includes SQ collector that monitors the SQ values and keeps a SQ history in memory for further processing in SQ analyzer.

In addition to different policies the monitoring system in MN is configurable. Configurable parameters include: *threshold*, *min-balance*, *expire-percent*, *old-FA-factor*, *worst-min-time*, *worst-max-time*, and *average-length*. Each parameter has a default value that can be changed with the `dynmn_tool(8)` configuration tool. Fig. 2 shows the relations between the configuration variables. By tuning the values we can meet the requirements for different environments and needs.

Agent expiration

The MA selection system chooses the communicating MA for the MN from a list of MAs. Each entry in the list has a certain expiration time that is bound to the MA agent advertisement lifetime. This lifetime is three times the agent advertisement interval which is configurable in the MA. When the MA agent advertisement lifetime is expired the entry is removed from the SQ collector and all information about the MA is cleared. The MA is handled as newly detected next time an agent advertisement is heard from it.

The *node selector* can be configured to use its own expiration method with early-expire policy for the MA selection system. The expire-percent configuration parameter value that is used to calculate the expiration time for the MA from the agent advertisement lifetime. After the calculated expiration time is exceeded the *old-FA-factor* is

used to decrease the MA priority. The idea behind this is that MAs may not be reachable when the MN moves relatively to the MAs. To speed up the MA detection this variable is used to decrease the priority of the FAs that are heard more seldom than the true FA agent advertisement interval is.

Threshold and minimum balance

The MN changes MAs based on the priorities. The *MA switching threshold* is a percent value that is used in priority comparisons. If the current MA priority is at least *threshold* percents of the compared MA, the MN will not change the current MA, but priority balancing is used.

When the compared priority is below *minimum balance* percents of the maximum, threshold is not used, but the MA with the highest priority is chosen. The best value for this variable depends on the underlying link-layer technology.

The number of monitored MAs

The MA is monitored when the SQ values are measured from the received packets of the MA. The Linux kernel limits the maximum number of MAs that the SQ collector can monitor at the same time to eight. When SQ collector monitors maximum number of MAs and a new MA is detected, some compromises have to be made. Either one entry is dropped from the currently monitored entries in the SQ collector or the new MA is discarded. When an MA is discarded it is not available for comparison with other MAs and, thus, a potential candidate for the communication partner is lost. This affects the communication availability.

When space is needed for the newly detected MA and all slots are reserved, the following procedures are taken to drop one of the monitored MAs.

- If the MA is monitored, but no advertisements are heard from it in *worst maximum time* seconds it is marked old. Old entries are dropped if new slots are needed for newly detected MAs.
- If no old entries are found then the entry with the worst signal quality value is dropped. The *worst minimum time* is a configurable parameter that tells the minimum time in seconds that the worst entry has to be in the SQ collector before it can be dropped.

An SQ cache in the device driver is a way to get rid of the limit in the Linux kernel for the maximum number of monitored nodes. The purpose for the SQ cache is to use the Linux kernel limit in a different way. The SQ cache makes sure that the SQ is available for at least eight last received packets. The problem in this approach is that the SQ values need to be queried fast enough before the old values get replaced by the new values.

Average length (N)

$$a_{i+N} = \frac{\sum_{j=i}^{N} a_j}{N} \quad (1)$$

The SQ collector uses this configuration variable to decide how many last received SQ values are used in calculation of the SQ average (a). If set to 1, the effect is the same as with the eager-switching policy.

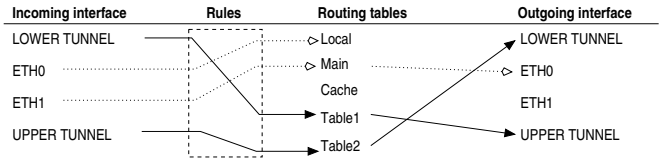

Fig. 5. An example packet handling in an FA

The greater the value for this variable is, the slower the system is in detecting changes in SQs, which is highly related to the communication availability. If the changes in priorities with surrounding MAs are not detected fast enough the node selector may not select the optimal MA for the mobile user. On the other hand if we use eager-switching or value 1 for the average length, the system may over-react depending on the SQ sensor characteristics. The node selector changes the MA more often than is really needed to maintain or increase the communication availability. Equation 1 shows how the average is calculated using the average length.

C. Seamless handoff

We use the routing capabilities of the Linux operating system for routing and tunneling of data packets. An *upper tunnel* in an FA is an IP-in-IP encapsulated data path to the upper MA. A *lower tunnel* in an FA is an IP-in-IP encapsulated data path to the lower FA or, when in MN decapsulation mode, to the MN. Fig. 5 shows how two tunnels, upper and lower, are connected together so that the data packets go from the lower tunnel to the upper tunnel and vice versa. This is the basic mechanism that the intermediate FAs (IFA) use to handle routing of data packets to and from MNs. An IFA is an FA that is at least one level up from the LFA and at least one level down from the *highest FA* (HFA). HFA is an FA that is the root for the FA tree hierarchy.

Delayed deletion and enhanced message processing

We made some enhancements similar to caching to speed up the signaling in FAs. *Delayed mobility binding deletion* in FAs is one of these optimizations. Another caching optimization is the *delayed forward deletion* in the SFA.

A *forwarding* includes a rule, a route, and a tunnel that together make up a tunneled data path for packets to and from the MN in the FA. With delayed deletion optimizations the location update time for the MN that is switching FAs inside the same hierarchy is faster. Also some packets that may be under way in the data path are not lost so easily.

Additionally, the registration reply processing in LFAs was enhanced. When the MN receives the registration reply, it changes the default route to the new FA. In this stage the FA hierarchy must have data path ready for packets coming from the MN. Like in the registration request processing, the reply processing in an *intermediate FA* (IFA) was enhanced. The IFA creates forwarding entries downwards after forwarding the request to the child FA. An LFA cannot do this since the reply may reach the MN before the forwarding entries are ready. The LFA is an exception to this enhancement since it creates the tunnel with MN decapsulation to the MN before forwarding

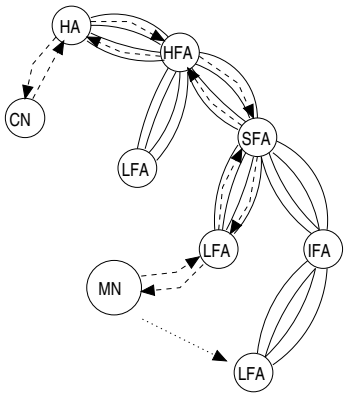

Fig. 6. Reverse tunneling and FA decapsulation

the reply to the MN.

The impact on data transmission with hierarchical Mobile IP

While the MN changes the current MA, the route for packets to and from the MN changes. This requires routing updates in both MAs and the MN. Packets destined to MNs are encapsulated in the HA and decapsulated either in the MN or in the lowest FA. Fig. 6 shows tunnels between FAs and the HA with *reverse tunneling* [32] and FA decapsulation. The HUT Dynamics FAs use explicit tunnels in both directions. This means that the *highest FA* (HFA) and IFAs have to make one tunnel upwards and another downwards. The LFA also uses the tunnel upwards but does not make a tunnel downwards if FA decapsulation is used with the MN.

When the MN changes the LFA in Fig. 6 the new LFA forwards the request to the next upper FA. This IFA forwards it again up to the SFA which notices that the downward route for this MN has changed. The SFA sends a reply to the new location of the MN via the IFAs and the LFA. Then it connects the lower tunnel to the upper tunnel. Finally, it changes the route for packets destined for the MN and coming from the upper tunnel.

When the IFA receives the reply from the SFA, it confirms the request and connects the lower tunnel to the upper tunnel as the SFA did. The IFA also adds a routing entry for the MN so that packets coming from the upper tunnel go to the right lower tunnel. When the MN receives the registration reply, it changes the default route to this new FA. This rises a race problem. When the SFA changes the route to the new location of the MN, all packets destined for the MN will be routed to this new path. Depending on the efficiency of the SFA and FAs, network latency, and network capacity the forwarding in the LFA may not be ready when the data already comes from the next higher FA to the LFA. This means that some of the packets in certain time period do not have a route to the MN. Further, this means that the MN will not receive some packets during the handoff and the deeper the hierarchy is the bigger is the possibility for packet loss.

Our solution for this problem is to enhance the functionality in the FA. This decreases the location update latency. Following steps are taken in the FA:

1. RECEIVE registration request
2. FORWARD request upwards
3. create tunnel downwards **if LFA and FA decapsulation mode in use**
4. add a route for packets incoming from upper tunnel destined for MN to the lower tunnel

—

5. RECEIVE registration reply
6. **if IFA** FORWARD reply downwards
7. connect tunnel upwards
8. **if LFA** FORWARD reply downwards

In the LFA the lower tunnel is created after the request has been forwarded upwards. Inside the FA hierarchy tunnels exist between FAs already. With this approach the tunnel creation does not delay the message processing in the LFA. Additionally, the data path is ready for downstream packets when the SFA switches the route. The MN changes the default route as before when the registration reply is received.

Packet loss prevention without multicasting or buffers

Soft handoff is a powerful method to change the communicating MA. With soft handoff, the MN can hear the old and the current FA simultaneously. This ability can be exploited in the system to prevent packet loss.

The FA hierarchy should not lose any packets from or to the MN when the MN is switching the FAs. In the routing engine of the Linux kernel route changes can be made as an atomic operation so that no packets are lost between the route update process.

We enhanced the packet loss prevention in FA hierarchy so that the data path for downstream flow is generated in the request handling stage. When the SFA changes atomically the route to the new MN location, data path down to the MN is ready. This makes the downstream direction of the data stream to the MN more reliable. Unfortunately the request processing stage in FAs requires more resources compared to the solution where all tunnels and routes are done in the reply processing stage. The effect to the handoff latency is, however, minimized since the request stage changes are performed while the FA is waiting for the registration reply.

The changes to the forwarding entries during the request handling does not compromise the security of the system since the data stream path is changed in the SFA or the HA and both can validate the request of the MN because of the security associations. Additionally, the data stream upwards from the MN is not opened during the registration request stage but in the reply processing stage.

VI. Performance analysis

In this section we explain the motivation for different tests. Different test setup environments are described before we show the test results and describe the tests more accurately. The focus was to test software, not hardware. Tests for the handoff management and packet routing while nodes are moving proportionally to each other are included. Multiple MNs are not tested. We also compare the results with related work.

A. IEEE 802.11 communication modes

The scope for IEEE 802.11 standard is to develop a media access control (MAC) and physical layer (PHY) specification for wireless connectivity for fixed, portable, and

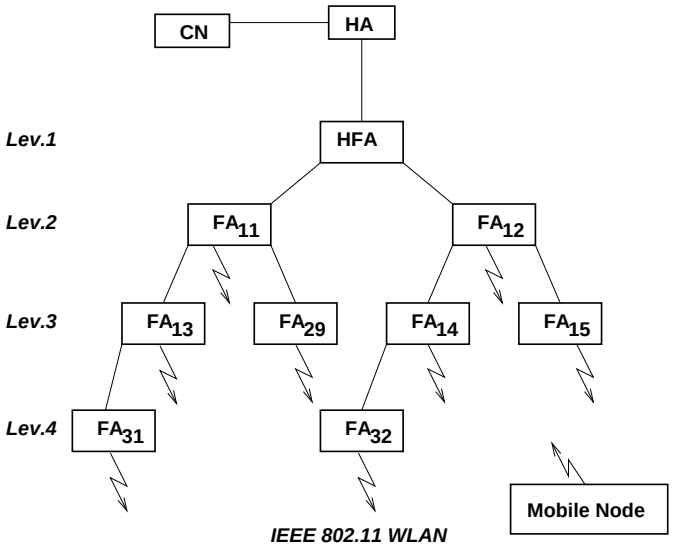

Fig. 7. A four-level test bed setup

moving stations within a local area [33]. The standard describes two different operational modes for communication between nodes, an ad hoc (IBSS) and an infrastructure network.

In an infrastructure network an AP always exists and the communication is controlled by it. The AP usually acts as a gateway, or a portal, to other parts of the network, e.g., to the Internet. Infrastructure networks are formed around APs and moving nodes roam from one AP to another. The handoff is handled in the link-layer and the network layer cannot decide which AP to use or when to initiate handoffs.

In the ad hoc network mode every node in range participates to the communication control and can directly communicate with each other. There is no need for an AP and thus no link-layer handoff management like in an infrastructure network.

B. Test bed setup

Test bed includes hardware and software components. Some software components are made only for the tests to support emulation, measurements, and result processing, e.g., logging and log parsing.

Dynamics – HUT Mobile IP MN, FA, and HA version 0.7pre3 were used in the test bed. All MAs resided in physically different hosts where the MN, the HFA, the HA, and the CN were Pentium class hosts whereas all the other FAs were less efficient, custom-built, 486-based embedded AP hosts, called Martnodes [34], with a wired and a wireless network interfaces. Martnodes and the CN used the Linux kernel version 2.2.9, the MN the Linux kernel version 2.2.13 and the HA version 2.2.12. IEEE 802.11 [33] compatible 2 Mbps WLAN adapters from Lucent were used in the FAs and the MN in ad hoc mode. Device driver version 1.0.1 from Andreas Neuhaus was used in the MN and in the Martnode FAs.

The HA and the CN resided in a 100 Mbps switched Ethernet laboratory network and the FA hierarchy in a dedicated switched 10 Mbps network. The wired FA hierarchy was in a private address-space subnet as shown in Fig. 7.

The MN used the wireless network for all its communication with the FAs while all the other data between the FAs and the HA and between the HA and the CN were transferred in the wired network. The clear bottleneck on the network was the wireless part. The maximum obtainable throughput without location updates was 1.4 Mbps using TCP and 1.6 Mbps using UDP.

Real time link quality recording and emulation

The monitor module supports SQ recording into files. Produced data files can be post processed with `gnuplot(1)` to produce graphical representations of the SQs as a function of time. Monitor records the SQ values of agent advertisements from FAs. The agent advertisement interval defines the sampling frequency for each FA. The recorded trace can be used to measure the SQs for different environments and it helps in setting up the wireless network. Monitor configuration variables such as the average-length affects the recorded trace. Thus, the monitor module SQ recorder is on top of the SQ analyzer.

To support easily configurable sampling frequencies, a more specific tool called `iwspy-gather` was made for SQ environment recording. The tool can be used to collect SQ information from different FAs in varying sampling frequencies. The collecting system is based on the MN agent solicitation [2] messages. Every time an agent solicitation is sent to the broadcast address, every FA that hears this message will reply automatically to it. This approach is not mandatory since ICMP echo messages could be used in environments that do not have FAs installed. The `iwspy-gather` tool gets messages from all heard FAs and collects the SQ values.

`Iwspy-gather` writes SQ values with timestamps into files. Each file, called an *signal quality tape* (SQT), contains information for only one FA. A group of files produced in a recording session is called an *SQT set*. The timestamps in the SQT files are synchronized within an SQT set. An SQT set can, thus, encode variations of the physical link and movements of all communicating parties within the time period.

`Iwspy-sim` is a program that uses SQTs for real life emulations. It communicates with the enhanced wireless interface network card device driver in the Linux kernel. It reads SQTs produced by the `iwspy-gather` and feeds SQ and timestamp value pairs to the driver which then replaces the actual SQ values sensored by SQ sensor. Each SQT is mapped to a MAC address so that SQTs do not get mixed. If the driver has a mapped SQT for a MAC address it is said to be in emulation mode for that MAC address. Thus, the driver can emulate received SQ values for some nodes simultaneously with nodes that do not have the SQT mapping. If the received packet is originated from an emulated node, the SQ value that was measured by SQ sensor will be replaced with a value from the mapped SQT. Emulation mode stops when the emulation starting time added with the last timestamp in the SQT is reached. The SQ values can be queried normally from the driver but the output is synchronized in time with the mapped SQT. Additionally, packet dropping can be emulated in the device driver level. The packet dropping percent is bound to the SQ value.

SQT sets can be divided and combined with other SQT

TABLE I

LOCATION UPDATE LATENCIES FOR SOME TRANSITIONS

Handoff type	Avg. (ms)	Std. dev. (ms)
$FA_{11} \rightarrow FA_{12}$	19.1	1.2
$FA_{12} \rightarrow FA_{11}$	19.2	1.4
$FA_{13} \rightarrow FA_{14}$	30.4	2.0
$FA_{14} \rightarrow FA_{13}$	30.3	1.0
$FA_{31} \rightarrow FA_{32}$	41.4	1.5
$FA_{32} \rightarrow FA_{31}$	41.1	1.3
$FA_{13} \rightarrow FA_{29}$	23.3	0.8
$FA_{29} \rightarrow FA_{13}$	23.5	0.9
$FA_{31} \rightarrow FA_{12}$	19.2	1.4
$FA_{12} \rightarrow FA_{31}$	41.5	1.7
$FA_{32} \rightarrow FA_{13}$	30.1	2.3
$FA_{13} \rightarrow FA_{32}$	41.3	1.6
$FA_{32} \rightarrow FA_{12}$	14.6	0.9
$FA_{12} \rightarrow FA_{32}$	37.4	1.4
$FA_{31} \rightarrow FA_{13}$	14.9	0.9

sets. Furthermore, imaginary SQTs can be created from scratch and combined with existing recorded SQT sets. This enables production of scenarios that would be otherwise hard to generate. SQT sets can be replayed with `iwspy-sim` multiple times which makes the emulation model convenient for testing different kinds of node selection policies. Also the characteristics for different configuration parameters can be examined.

C. System performance tests

We ran all the tests using FA decapsulation and reverse tunneling modes on the wireless environment. Security associations were configured between the HA and the MN, and separately in the FA hierarchy between FAs. The HFA and the HA did not have a preconfigured shared secret. Therefore, they used RSA public key encryption with 768-bit public keys for session key distribution. All the other key distribution operations used keyed MD5 [35] algorithm. This kind of configuration corresponds to the case where we do not have the complexity of managing shared secrets between the HN and each FN. However, it is feasible to use shared secrets between FAs in one administrative organization as is the usual case with FA hierarchies.

In handoff latency and packet traffic tests the MN was forced to follow a predefined FA path and handoff frequency in the FA hierarchy illustrated in the Fig. 7. Therefore, the MN did not use the agent discovery part of the Mobile IP. In practice, the MN received the agent advertisements, but it completed a location update only when requested by a test script.

Handoff latency

We measured the handoff latency by forcing the MN to initiate a handoff between different FAs once in 100 ms. Table I contains the resulted handoff latencies. The purpose for this test was to find the effect of the hierarchy level to the handoff time.

The results show that the handoff latency increases linearly with the hierarchy level at least up to the fourth level. In this test bed the delay due to one hierarchy level is 11 ms.

Packet loss, latency, and order with location updates

This test describes the packet routing characteristics with handoffs. To better understand end-to-end effects requires a closer analysis of what happens to individual packets during a handoff. `Udpcat` and `udplisten` programs were made for packet *loss*, *latency*, *duplicates*, and *order* testings. `Udpcat` sends UDP [20] packets across the IP network to `udplisten` in a certain interval. Every UDP packet contains an increasing serial number. `Udplisten` saves the serial number and timestamp of the received packets into a log file. `Udplisten` can also initiate location updates in the MN via API calls if it is started in the same host as the MN is running in.

In the test the UDP packet size was 1024 bytes and the throughput 100 kB/s, 100 packets per second. Thus the average interval between packets was 10 ms. Both the directions from CN to MN and from MN to CN were tested by sending 30000 packets several times. Generated log files were parsed to obtain needed information. We tested the system without location updates when the MN was registered to the FA_{31}. Without location updates no packet order changes or duplicates occurred in both directions. Maximum delay was between 20 ms and 400 ms in both directions. 270000 packets were sent from the CN to the MN and 2 packets were lost (0.0007%). From the MN to the CN direction 840000 packets were sent and 10 packets was lost (0.0012%). The packet loss was negligible but it shows that packets are lost also without location updates.

Table II contains the packet losses per location update with data stream from CN to MN. The test included almost 8000 location updates per transition. The location update interval was one second. If every packet during the handoff is lost the estimated packet loss depends on the transition. If all packets are lost during the handoff, in 20 ms handoff two packets are lost. The minumum handoff latency in Table I is 14 ms and maximum 41 ms. Thus, with non-optimized handoffs the packet loss per location update would be around one to four packets.

Transitions were tested in a group. First the MN forced location update to the FA_{11} and then to the FA_{31} then to the FA_{29}, etc. `Udplisten` received UDP packets and forced location updates in the MN, logged every received packet into a log file and marked location updates into the log file. A location update started when `udplisten` used the MN API to change the forced FA. After that the MN was forced to update location to the forced FA IP address. When the MN received the reply and the location update was successful it replied to `udplisten` through the API. `Udplisten` marked the location update end to the log file. All lost packets between the location update starting mark and 100 ms after the ending mark were included into the packet loss calculations. The average number of duplicated packets in the 30000 packet sending session was 0.98 packets (0.003% of all packets) and the average number of packets that changed order was 0.76 packets (0.003% of all packets). The maximum packet delay between received packets per 30000 packets session changed between 23 ms and 700 ms. Average delay was 10 ms, as expected.

Table III contains the packet losses per location up-

TABLE II

Data stream from CN to MN: packet loss

transition	lost packets/update
$FA_{11} \leftrightarrow FA_{31}$	0.00
$FA_{31} \leftrightarrow FA_{29}$	0.00
$FA_{29} \leftrightarrow FA_{32}$	0.00
$FA_{31} \leftrightarrow FA_{13}$	0.00
$FA_{12} \leftrightarrow FA_{15}$	0.00
$FA_{15} \leftrightarrow FA_{31}$	0.03
$FA_{32} \leftrightarrow FA_{11}$	0.07
$FA_{13} \leftrightarrow FA_{12}$	0.10

TABLE III

Data stream from MN to CN: packet loss

transition	lost packets/update
$FA_{11} \leftrightarrow FA_{31}$	0.27
$FA_{31} \leftrightarrow FA_{29}$	0.27
$FA_{29} \leftrightarrow FA_{32}$	0.00
$FA_{31} \leftrightarrow FA_{13}$	0.15
$FA_{12} \leftrightarrow FA_{15}$	0.14
$FA_{15} \leftrightarrow FA_{31}$	0.00
$FA_{32} \leftrightarrow FA_{11}$	0.00
$FA_{13} \leftrightarrow FA_{12}$	0.00

date with data stream from the MN to the CN. The test included 20000 location updates per transition. The location update interval was 300 ms. Transitions were tested separately. The average number of duplicates in the 30000 packet sending session was 0.006 packets (0.000% of all packets) and the average number of packets that changed order was 0.44 packets (0.002% of all packets). The maximum packet delay between received packets per 30000 packets session changed between 20 ms and 400 ms. Average delay was 10 ms, as expected. In this test it was expected that every lost packet was due to the location update. The number of lost packets varied depending on the transition.

The results show that packets are lost during a handoff in some transitions. Additionally, the proportional packet loss distribution differs with data streams from the MN to the CN and from the CN to the MN.

End-to-end performance

End-to-end performance was measured as throughput. In the test the location update frequency was increased and UDP and TCP data stream throughput was measured. A throughput suitable for video streams of 1.4 Mbps, such as a near TV-quality MPEG-1 [36], was chosen as the speed for the data streams. Motivation for the latter measurements was to find out how frequently the location updates could be performed with the system when using a representative multimedia application. End-to-end performance depends on the packet routing characteristics that the previous test identified. Especially the TCP protocol is more sensitive for packet loss and delay than UDP which can be seen from the Fig. 8.

Location updates were forced so that the MN started a new registration with given intervals. If the MN could not complete the previous registration before the new one

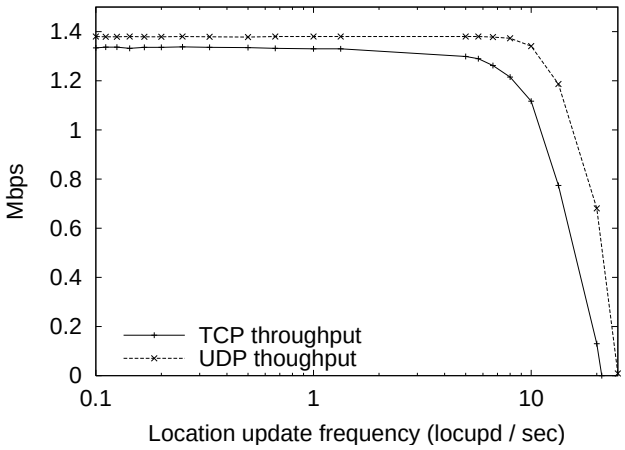

Fig. 8. UDP and TCP throughput with location updates

TABLE IV

UDP and TCP throughput with location updates

Location updates per second	throughput UDP (Mbit/s)	standard deviation (Mbit/s)	throughput TCP (Mbit/s)	standard deviation (Mbit/s)
50.00	0.00	0.00	0.00	0.00
20.00	0.68	0.16	0.13	0.03
13.33	1.19	0.17	1.12	0.10
10.00	1.34	0.07	1.12	0.10
8.00	1.37	0.02	1.22	0.05
6.67	1.38	0.01	1.26	0.04
5.00	1.38	0.00	1.30	0.02
1.00	1.38	0.00	1.33	0.02
0.10	1.38	0.00	1.33	0.04

began, a script added a firewall filter to drop the incoming packets from the CN until a registration succeeded. This corresponds to the situation in which the MN is too fast for the registration procedure to complete in time. Fig. 8 contains the throughput graph and Table IV shows the data points more accurately.

Throughput tests were made with `netperf`. It is a benchmark that can be used to measure various aspects of networking performance. The primary focus of the `netperf` is on bulk data transfer and request/response performance using either TCP or UDP and the Berkeley Sockets interface [37]. Socket send and receive sizes were 1024 bytes with UDP stream and 4096 bytes with TCP stream. 60 second throughput test was repeated several times with each location update interval.

Different policies and configurations

Monitor supports four different policies: newest-FA, eager-switching, short-interval, and the default policy. Additionally different configuration parameters are available for fine tuning and adjusting in different environments and with different link technologies. In this test the SQ environment recording and replaying system was used. An SQT set was recorded in an office environment with eight FAs.

TABLE V

MONITOR SETTINGS

	Setting 1	Setting 2
Threshold	50	1
Min-balance	10	13
Expirepercent	50	50
Old-FA-factor	50	50
Worst-min-time	10	10
Worst-max-time	20	20
Average-length	1	3
Early-expire	OFF	OFF
Newest-FA	OFF	OFF
Eager-switching	ON	OFF

TABLE VI

PACKET DROPPING PERCENT BOUND TO THE SQ

SQ (dB)	Packet drop percent
≤ 4	100%
5	90%
6	75%
7	33%
8	20%
9	10%

Monitor testings

The monitor was tested with two different settings using a recorded SQT set. Table V shows the used settings. Additionally a test was made without the monitor, i.e., without the SQ sensor, the SQ collector, and the SQ analyzer. The device driver was modified to drop packets when the SQ is low. Table VI shows the corresponding SQ for each dropping percent. Packet dropping was used to simulate the wireless media and to find out the differences between the tests.

The SQT set was replayed with the two different monitor settings and without the monitor. `Udplisten` and `udpcat` were used to flood 100 UDP packets per second from the CN to the MN while replaying the SQT set. Thus, the maximum packet loss amount is 100. Table VII shows the number of location updates and lost packets with these three scenarios.

With plain Mobile IP settings, SQ values are not used in FA selection. The MN changes the FA if the agent advertisement lifetime expires. The agent advertisement interval is crucial since it determines the lifetime for the advertisement. By default it is three times the agent advertisement interval.

With plain Mobile IP setting the MN loses considerably more packets than with the monitor. The MN with monitor setting 1 does location updates very eagerly compared

TABLE VII

MONITOR TESTING RESULTS

	Plain MIP	Setting 1	Setting 2
Lost packets	2179	66	117
Location updates	8	63	9

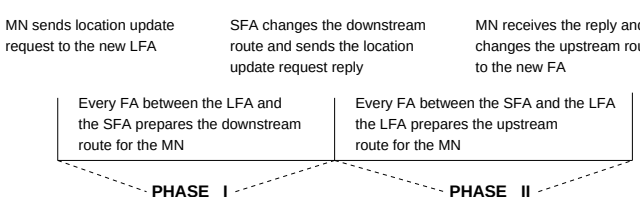

Fig. 9. Two-phase handoff

to the other two, but the packet loss is lower. Eager-switching is not the best policy since it makes location updates much more frequently than the other two. When monitor setting 2 is used the packet loss is low and the number of location updates is almost as low as with the plain Mobile IP setting. The threshold with monitor setting 2 was set to 1 and min-balance to 13 which makes the MN switch the MA when the current MA priority is below 13. Additionally, the average-length was 3 with setting 2 which makes the priorities more stable than with the setting 1. All these changes in the setting 2 decreases the location updates compared to the setting 1.

D. Handoff protocol analysis

The handoff protocol that the system provides uses horizontal MCHOs. They are soft and classified as forward type handoffs. Additionally, the handoff can be described as a *two-phase handoff*. In the first phase the route for downstream packets is changed and in the second phase the upstream route for the packets is changed. In the latter the system is in a state where the up- and downstream packets are routed via different APs. After the second phase, the handoff has completed. This is possible only with soft handoff. Fig. 9 illustrates the two-phase handoff.

Locality is exploited because of the hierarchical structure of the FAs. The localized location updates reuse partially the path between the MN and the HA, and depending on the FA hierarchy the reused path may be relatively long. The lower in the hierarchy the SFA is, the longer the reused path is in the FA hierarchy during the location update. In the MA selection process, radio hints are used to achieve seamless and glitchless handoffs. This is possible because of the finer granularity in the FA comparison and direct knowledge of the wireless data path SQ characteristic. Coarsely, the better the SQ is, the better the packet delivery, and thus the communication availability, becomes. No handoff request queuing is performed in the FAs.

Scalability issues were not tested. The soft handoff does not use neither specific buffers nor multicasting for packet loss prevention. Signaling load is shared in the hierarchical structure with localized location updates. These two things may improve the scalability with multiple MNs but they need to be tested and analyzed more throughly. The HUT Dynamics Mobile IP supports signaling prioritization. Signaling is prioritized over the data packets, which makes it more tolerable for congestion in the network.

Service disruption time is comparable to the glitches that the MN or the CN experience during data stream transfers. Lost packets, packet order changing, and relatively high latencies are sources for service disruptions. Service disruption affects communication availability. The

more the service is disrupted, the worse the communication availability becomes. With WLANs packet loss cannot be eliminated completely if the AP coverage is sparse. Even in dense WLANs the reflection and interference may cause service disruption. The soft handoff with the localized location updates completes relatively fast and the packet loss rate is negligible. Data streams like UDP and TCP perform well with up to five location updates a second. In an office environment such a high location update frequency is highly improbable and may indicate that the WLAN architecture should be replanned.

E. Comparison with related work

Srinivasan Seshan made handoff latency, packet loss, and packet duplicate measurements in his Ph.D. thesis [25]. In his implementation the handoff latency is measured between the registration request message and the first data packet coming from the new AP. The implementation from Seshan does not use any registration reply messages. Our test measured the latency between the registration request message and the registration reply message. In our two-phase handoff the MN may receive data packets from the old AP before the registration reply has arrived at the MN. Additionally, our handoff protocol uses replay protection and authentication which increases the handoff delay. The implementation from Seshan does not take care of the security issues. Thus, the handoff latency results are not directly comparable.

Seshan measured the packet loss during handoffs with a bit higher data rate as we have done, 1024 byte packets with 1.0 Mbit/s data rate. Without buffering or multicasting the implementation of Seshan lost several packets per handoff (2-5). Even with multicast-based handoffs the packet loss rate was several packets. When buffering was used with multicasting the packet loss rate was negligible. In our implementation the packet loss rate is negligible without multicasting or buffering.

Fikouras et al. measured the traffic disruption time with Mobile IP and with different handoff policies [5]. The traffic disruption time with Mobile IP handoffs and UDP traffic was up to six seconds and with TCP traffic more than ten seconds. They did not use hierarchical Mobile IP or SQ values to determine the best FA. Their measurements showed that eager switching was the best choice when traffic disruption time is minimized. In our tests the eager switching policy behaved worse than the default-policy. Additionally, service disruption times were negligible in our environment.

VII. Conclusions

We developed a general event driven node selection mechanism based on the radio link signal qualities. Furthermore, we enhanced the Dynamics – HUT Mobile IP system to support glitchless and seamless handoffs in WLANs. The configurable MA selection system in the MN is based on prioritization and techniques that affect the priorities.

The enhanced system improves packet delivery to and from a moving MN. Glitchless and seamless handoffs are automated in the MN. Handoff management does not affect the packet routing latency although the latency may change if the hierarchy level of the LFA changes. Different policies and configurable handoff management with a modular implementation fulfills the modular node selection criteria. Additionally, the mobility agent detection and MA selection systems enable the MN to choose the MA that offers the best communication availability.

- The enhancements help the user to switch from a wired office network to a WLAN. They also improve the communication availability since the user can attach to the network more easily and still maintain connections over different media. The mobile user can change the policy and handoff management parameters dynamically while moving and without disturbing the communication sessions. Thus, the system is flexible and adjusts adaptively to the needs of the user.
- With soft handoffs neither buffering nor multicasting is required to achieve seamless handoffs. Soft handoff capable peer-to-peer link technologies enable simplicity both on the link and on the network layer. Thus, the implementation is simpler and more robust. On the other hand, this solution requires soft handoff capable point to multipoint link-layer, such as the Ethernet like IEEE 802.11 ad hoc mode network.
- The solution is not dependent on the handoff management below the network layer. Thus, it is also independent of the underlying physical characteristics of the link level radio technology. It can, however, use the link level SQ values when available to achieve the needed communication availability. In Mobile IP the MN controls the handoff management in the network layer and the ad hoc mode suits, thus, well for this system. With newest-FA policy the MN can also be used in a wireless IEEE 802.11 infrastructure network.
- The priority-based FA comparison is feasible because it is not bound to the SQ values and, thus, not only to WLANs. With priorities, different value functions can be combined to get the overall priority and the best choice over different possibilities. For example, priority degradation and increasing are value functions used to improve communication availability on temporary and static failures of APs or network connections.
- SQ awareness is a simple but effective way to improve the communication availability without extending mobility protocols. It is scalable and independent, from intra-WLAN through micro mobility to macro mobility.
- The tests showed that hierarchical Mobile IP with SQ awareness and two-phase handoff supports micro mobility. This has not been previously reported. Handoffs can be done more frequently than is on average needed in an office environment. Five handoffs per second with negligible packet loss and with session maintenance is sufficient for even higher needs.

The handoff protocol uses the radio channel sparingly since it does not send multiple signaling messages during a handoff. The registration request is used to initiate location updates and the registration reply is used for authentication purposes and to finish the two-phase handoff. The solution is architecturally natural with Internet mobility on WLANs where the smart mobile hosts can operate independently and the network is simple. Thus, we have demonstrated that a network-layer handoff support model in the MN is sufficient for continuous communication availability in mobile networks.

A. Future work

Scalability with multiple MNs under the same FA hierarchy and an HA should be analyzed. Hard handoff management is required when channel switching occurs in an ad hoc mode WLAN. Its use with the presented system should be evaluated. Furthermore, dynamic parameters in the FA and agent advertisements to support MA selection in the MN should be studied.

References

[1] J. Postel. RFC 760: DoD standard Internet Protocol. Jan. 1980. Obsoleted by RFC0791, RFC0777. Obsoletes IEN123. Status: UNKNOWN.

[2] C. E. Perkins. *Mobile IP Design Principles and Practice*. Addison-Wesley Publishing Company, One Jacob Way, Reading, Massachusetts 01867, 1. edition, Oct. 1997.

[3] C. Perkins. RFC 2002: IP Mobility Support. Oct. 1996. Updated by RFC2290. Status: PROPOSED STANDARD.

[4] H. Laamanen. Serveability Issues, Series of Publications C, Report C-1998. Technical report, University of Helsinki, Department of Computer Science, 1998.

[5] N. Fikouras, K. E. Malki, S. Cvetkovic, and C. Smythe. Performance of TCP and UDP during Mobile IP handoffs in single-agent subnetworks. In *IEEE Wireless Communications and Networking Conference, WCNC*, pages 1258–1262, Sept. 1999.

[6] R. H. Katz. Adaptation and mobility in wireless information systems. *IEEE Personal Communications*, 1(1):6–17, 1994.

[7] J. Sevanto, M. Liljeberg, and K. Raatikainen. Introducing Quality-of-Service and Traffic Classes into Wireless Mobile Networks. In *WoWMoM'98. Proceedings of first ACM international workshop on Wireless mobile multimedia*, 1998.

[8] M. E. Kounavis, A. T. Campbell, G. Ito, and G. Bianchi. Supporting Programmable Handoff in Mobile Network. In *IEEE International Workshop on Mobile Multimedia Communications*, 1999.

[9] M. Stemm and R. H. Katz. Vertical handoffs in wireless overlay networks. 3(4):335–350, Winter 1998.

[10] G. P. Pollini. Trends in Handover Design. *IEEE Communications Magazine*, 34(3):82–90, Mar. 1996.

[11] L. Taylor, R. Titmuss, and C. Lebre. The challenges of seamless handover in future mobile multimedia networks. *IEEE Personal Communications*, 6(2):32–37, 1999.

[12] K. Brown and S. Singh. M-UDP: UDP for Mobile Networks. *ACM Computer Communication Review*, 26(5):60–78, 1996.

[13] R. Cáceres and V. N. Padmanabhan. Fast and Scalable Wireless Handoffs in Support of Mobile Internet Audio. *ACM Journal on Mobile Networks and Applications*, 3(4), Dec. 1998.

[14] C. Tan, S. Pink, and K. Lye. A Fast Handoff Scheme for Wireless Networks. In *2nd ACM International Workshop on Wireless Mobile Multimedia (WoWMoM'99), Seattle, Washington*, pages 83–90, Aug. 1999.

[15] J. Mysore and V. Bharghavan. A New Multicasting-based Architecture for Internet Host Mobility. In *MOBICOM 97 Budapest Hungary*, pages 161–172, Sept. 1997.

[16] A. Campbell, J. Gomez, C.-Y. Wan, Z. Turanyi, and A. Valko. Cellular IP. Internet Draft. October 1999. *(work in progress)*

[17] R. Ramjee, T. L. Porta, S. Thuel, K. Varadhan, and L. Salgarelli. IP micro-mobility support using HAWAII. Internet Draft. Jun 1999. *(work in progress)*

[18] P. McCann, T. H. J. Wang, A. Casati, C. Perkins, and P. Calhoun. Transparent Hierarchical Mobility Agents (THEMA). Internet Draft. Mar. 1999. *(work in progress)*

[19] C.-K. Toh. The design & implementation of a hybrid handover protocol for multi-media wireless LANs. In *The first annual international conference on Mobile computing and networking, MOBICOM*, pages 49–61, 1995.

[20] J. Postel. RFC 768: User Datagram Protocol. Aug. 1980. Status: STANDARD. See also STD0006.

[21] A. Bakre and B. Badrinath. I-TCP: indirect TCP for mobile hosts, Distributed Computing Systems. In *Distributed Computing Systems, Proceedings of the 15th International Conference*, pages 136–143, June 1995.

[22] J. Postel. RFC 793: Transmission Control Protocol. Sept. 1981. See also STD0007. Status: STANDARD.

[23] C. E. Perkins and K.-Y. Wang. Optimized Smooth Handoffs in Mobile IP. In *The Fourth IEEE Symposium on Computers and Communications*, pages 340–346, 1999.

[24] K. Keeton, B. A. Mah, S. Seshan, R. H. Katz, and D. Ferrari. Providing Connection-Oriented Network Services to Mobile Hosts. In *USENIX Symposium on Mobile and Location-Independent Computing*, Aug. 1993.

[25] S. Seshan. *Low-Latency Handoff of Cellular Data Networks*. PhD thesis, University of California at Berkeley, 1995.

[26] H. Balakrishnan, S. Sseshan, and R. Katz. Improving reliable transport and handoff performance in cellular wireless networks. *Wireless Networks*, 1(4):469–481, 1995.

[27] S. Tekinay and B. Jabbari. A measurement-based prioritization scheme for handovers in mobile cellular networks. *IEEE Journal on Selected Areas in Communications*, 10:1343–1350, 1992.

[28] N. D. Tripathi and J. H. R. H. VanLandinoham. Handoff in Cellular Systems. *IEEE Personal Communications*, 5(6):26–37, Dec. 1998.

[29] H. J. Wang, R. H. Katz, and J. Giese. Policy-Enabled Handoffs Across Heterogeneous Wireless Networks. In *2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 51–60, Feb. 1999.

[30] D. Forsberg, J. Malinen, J. Malinen, T. W. Tom, and M. Tiusanen. Distributing Mobility Agents Hierarchically under Frequent Location Updates. In *Sixth IEEE International Workshop on Mobile Multimedia Communications (MoMuC'99), San Diego.*, 1999.

[31] L. Torvalds. Linux: A Portable Operating System. Master's thesis, University of Helsinki, Department of Computer Science, Dec. 1997.

[32] G. Montenegro. RFC 2344: Reverse Tunneling for Mobile IP. May 1998. Status: PROPOSED STANDARD.

[33] IEEE. *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*. Institute of Electrical and Electronics Engineers, inc., New York, NY, Nov. 1997.

[34] H. Arppe, D. Forsberg, J. Malinen, P. Massetti, and J. Salmi. Helsinki University of Technology research project: Mobile Ad-hoc Routing Testbed, MART, 1999. http://www.cs.hut.fi/Research/Mart/.

[35] R. Rivest. RFC 1321: The MD5 Message-Digest Algorithm. Apr. 1992. Status: INFORMATIONAL.

[36] ISO/IEC. ISO/IEC 11172:1993 Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s, 1993.

[37] Hewlett-Packard Company. Netperf: A Network Performance Benchmark, revision 2.1, Feb. 1996. http://www.netperf.org/netperf/NetperfPage.html.