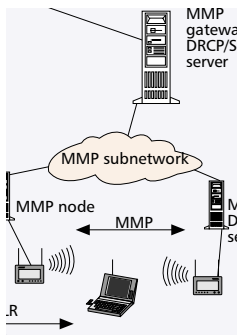


A MULTILAYERED MOBILITY MANAGEMENT SCHEME FOR AUTO-CONFIGURED WIRELESS IP NETWORKS

K. DANIEL WONG, ASHUTOSH DUTTA, JIM BURNS, RAVI JAIN, AND KENNETH YOUNG

TELCORDIA TECHNOLOGIES

HENNING SCHULZRINNE, COLUMBIA UNIVERSITY



The convergence of wireless and Internet Protocol (IP) has led to the need for IP to handle mobility. The Mobile IP protocol was developed to facilitate IP mobility. However, it has a number of shortcomings for dynamically auto-configured networks.

ABSTRACT

The convergence of wireless and IP has led to the need for IP to handle mobility. The Mobile IP protocol was developed to facilitate IP mobility. However, it has a number of shortcomings for dynamically auto-configured networks. Mobility protocols like Mobile IP with Location Registers (MIP-LR) and Session Initiation Protocol (SIP) have been developed to address some of its shortcomings. Micromobility protocols like Cellular IP have been developed to address other shortcomings of Mobile IP. In this article we present a new integrated mobility management scheme that advantageously combines the strengths of SIP and MIP-LR with the benefits of a micromobility management protocol similar to Cellular IP. A prototype implementation of our scheme is explained, and lessons learned in the prototyping process are presented.

INTRODUCTION

Internet routing was designed based on the assumption that nodes do not move. As the need for merging wireless and IP arose, a mechanism was needed to handle mobile nodes. Mobile IP (MIP) [1] was created to provide this mobility support. However, there is a need to add new features to mobility handling for auto-configured wireless networks. In this article we introduce a new multilayered mobility management scheme for auto-configured wireless IP networks, explain the design issues, and document a laboratory prototype implementation of our scheme and share lessons learned in the prototyping process.

We distinguish between real-time and non-real-time traffic. Real-time traffic is streaming traffic where the time relation between successive data packets must be preserved. In other words, only small deviations can be tolerated between the packet arrival times. Real-time traffic is often carried over Real-Time Transport Protocol (RTP)/User Datagram Protocol (UDP)

[2]. Non-real-time traffic includes all other kinds of traffic without such strict delay, jitter, and loss requirements, and is mostly carried over TCP. Real-time traffic is especially important because it is the kind of traffic used by voice and video-conferencing, including IP telephony. We are interested in mobility management for wireless IP networks that handle real-time traffic while also supporting traditional non-real-time traffic.

The mobility management scheme we present is flexible enough to apply to both traditional infrastructure-backed networks and quasi-static ad hoc networks. In traditional infrastructure-backed networks, on one hand, the network infrastructure topology changes only infrequently. There are also mobile hosts that may move around fairly frequently. In traditional ad hoc networks, on the other hand, the topology constantly changes as every node moves and can be a router. Our concept of quasi-static networks is somewhere between a traditional infrastructure-backed network and a traditional ad hoc network in terms of network constancy. For example, a quasi-static network may have whole subnets that are mobile; however, a subnet moves together and maintains a relatively stable internal topology. An example of a quasi-static network might be a makeshift communication network in a disaster area. It needs to be rapidly deployed, but once deployed has a relatively stable central core alongside mobile hosts (MHs) and mobile subnets.

Since the underlying components in a quasi-static network offer less stability than an infrastructure-backed network, the mobility protocols must be designed with greater care for robustness and survivability. For our purposes we maintain that the design considerations for quasi-static networks are in this sense more stringent than for infrastructure-backed networks. Therefore, our schemes, designed to meet the tighter survivability requirements, work well in both types of networks. Many mobility management schemes have been developed for IP

networks to support both interdomain and intradomain mobility of MHs supporting both real-time and non-real-time traffic. There are significant challenges, though, regarding the robustness, management overhead requirements, and latency of some of these existing approaches; hence, none of these traditional mobility management schemes alone sufficiently supports our requirements. This will be elaborated later.

SYSTEM ASSUMPTIONS

In this article we make the following assumptions:

- All real-time sessions are managed by Session Initiation Protocol (SIP) [3]; that is, SIP is used to initiate, modify, and terminate real-time sessions. This is a reasonable assumption to make, given the momentum SIP has gained in recent years.
- When an MH moves into a foreign network, it needs to obtain an IP address in that network and arrange for traffic to be routed to that IP address. While MIP provides a means to do both, we assume that the former is performed as part of a possibly independent auto-configuration protocol, and the latter is the concern of the mobility management scheme.
- Interdomain authentication, authorization, and accounting (AAA) is handled together with auto-configuration.

Furthermore, we use the terms *home network*, *home agent* (HA), and so on with implicit reference to the MH being discussed, unless otherwise noted.

MOTIVATIONS AND REQUIREMENTS

MIP is the standard scheme for IP mobility management.

MIP has several strengths, including:

- Transparency to upper layers. MIP is designed as an overlay over the IP layer in the protocol stack. Therefore, its operation is transparent to upper layers.
- No modifications are needed in the correspondent host (CH). Therefore, existing IP nodes can be CHs without modification.

However, basic MIP has some shortcomings, including:

Routing efficiency problems. Having to route through the HA is an inefficiency known as *triangular routing*. However, a route optimization enhancement [4] has been introduced to fix this problem (unfortunately, this requires CH modification to understand binding updates).

Overhead problems. Encapsulated packets are at least 8–12 bytes larger than the original packets. There is also signaling overhead from the MIP registration requests and replies.

Handoff latency problems. In addition to the handoff latency related to handoff at the physical and link layers of the MH links, MIP signaling could add significant latency. There is ongoing work to enhance MIP to reduce the handoff latency.

Survivability problems. The HA is a single point of failure in MIP routing. If a single node, the HA, is unavailable, packets will not be routed correctly to a roaming MH.

The first major requirement for our mobility management scheme is that it handle mobility without MIP's shortcomings. A second requirement is that real-time traffic must be handled with special care. For example, handoff latency is especially disruptive to real-time traffic, even if most of the packets in transit during the handoff are not lost but buffered and eventually delivered to the MH. A third requirement is that it must be survivable and robust in a quasi-static dynamically auto-configured network.

ARCHITECTURE

Our two-layer integrated mobility management scheme is designed keeping in mind the requirements just discussed. There is currently no single protocol that handles global (macro-) mobility as well as micromobility, but both are important and necessary. We designed Micromobility Management Protocol (MMP) to handle micromobility for our integrated mobility management scheme. For macromobility we chose SIP to handle mobility for real-time traffic [5] and MIP with Location Registers (MIP-LR [6]) to handle the mobility for non-real-time traffic. In either case, MMP [7] handles micromobility.

Our architecture introduces several novel features:

- Policy-based usage of SIP for macromobility signaling for real-time traffic and MIP-LR for macromobility signaling for non-real-time traffic
- Integration of SIP for macromobility with MMP for micromobility
- Integration of MIP-LR for macromobility with MMP for micromobility

The reader is referred to [5–7] for details on these base schemes, beyond the brief introduction we provide next.

A BRIEF INTRODUCTION TO COMPONENT MOBILITY PROTOCOLS

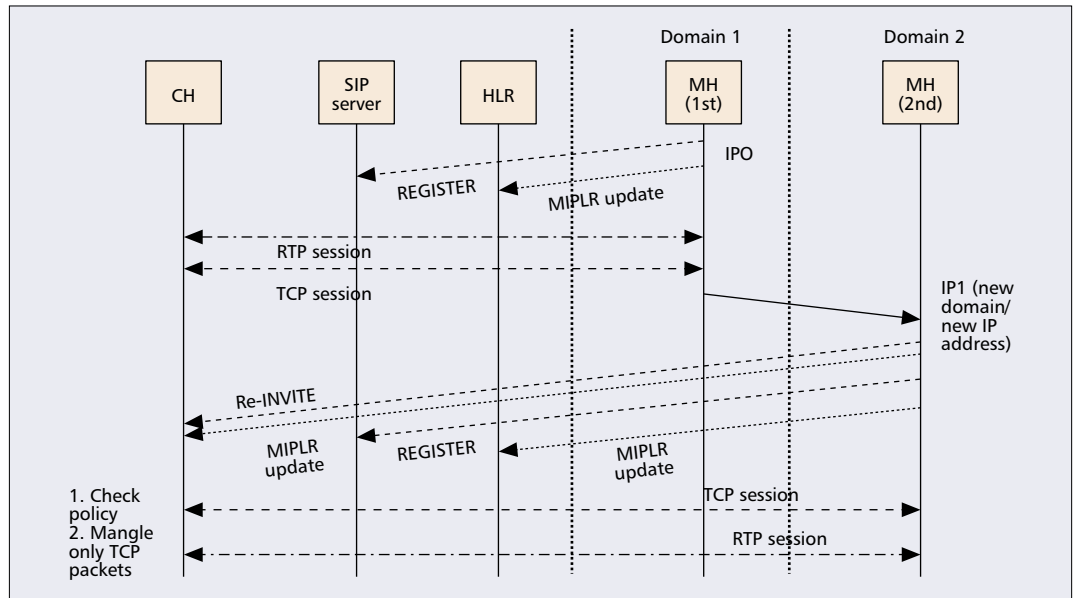
SIP uses INVITE messages to initiate sessions. SIP-based terminal mobility uses re-INVITE messages to provide fast handoff for real-time multimedia traffic as the MH moves from one cell to another. SIP user agents (UAs) and SIP servers interact with other entities such as AAA servers, and DHCP/DRCP or PPP servers to provide macromobility support.

MIP-LR provides a more efficient approach than MIP by taking care of forwarding and profile replication. In MIP-LR, the database mapping of the MH's IP address to its care-of address is maintained by a home location register (HLR) that is queried similar to how the HLR is queried in cellular systems for MH location. Unlike the HA, it need not necessarily be located in the home network and can be replicated for survivability.

MMP is an extension of Cellular IP suitable for ad hoc networks, where the nodes and gateways are auto-configured using protocols such as Dynamic and Rapid Configuration Protocol (DRCP). It provides mobility support when the client moves within a domain by using host-based routing internal to the domain. After auto-configuration, the MH sends an update to

Our two-layer integrated mobility management scheme is designed keeping in mind the requirements discussed later. There is currently no single protocol that handles global (macro-) mobility as well as micromobility, but both are important and necessary.

For macro-mobility, we use both SIP and MIP-LR. Although MIP-LR alone can handle the macro-mobility for both real-time and non-real-time traffic, we let it handle only non-real-time traffic, and use SIP for macro-mobility for real-time-traffic.



■ Figure 1. Integration of SIP and MIP-LR.

the gateway, including its new IP address. This IP address is stored in host-based route caches along the path to the gateway. When handoffs occur, route caches are updated, so handoff latency is at most the time it takes for the update to reach the gateway within the domain. MMP also optionally provides survivability features by adding multipath and multigateway features for the same domain.

POLICY-BASED USAGE OF SIP AND MIP-LR

For macromobility, we use both SIP and MIP-LR. Although MIP-LR alone can handle macro-mobility for both real-time and non-real-time traffic, we let it handle only non-real-time traffic, and use SIP for macromobility for real-time-traffic because:

- SIP is already used for session control signaling for real-time applications, and mobility for these applications can be handled using the same signaling mechanisms.
- SIP handling of terminal mobility integrates well with SIP personal mobility (employing a unique URI for the user and obtaining the assistance of SIP proxies).
- A SIP-based solution exists for smooth handoffs of real-time traffic streams [8].

On the other hand, could SIP be used to handle macromobility for *both* real-time and non-real-time traffic? There are at least two proposals that describe how SIP can be used to handle macromobility for non-real-time traffic as well [9, 10]. This attractively provides a uniform macromobility protocol using an application layer signaling protocol. However, we believe this work is still under development, so it has not been used in the presently proposed scheme. Therefore, we chose an IP-layer macromobility solution to handle non-real-time traffic. We chose MIP-LR for its advantages over MIP, especially its survivability, reduced overhead, and routing efficiency.

In order to use both SIP and MIP-LR for macromobility management, we use a policy

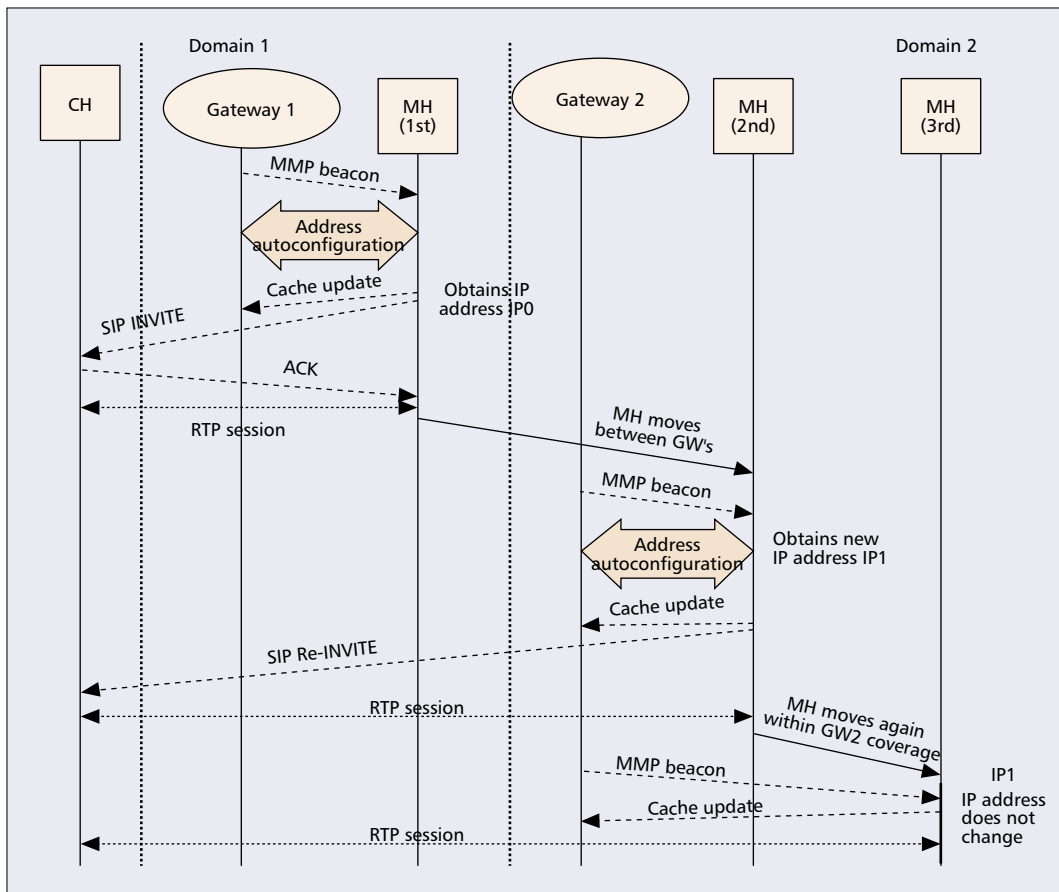
table. Abstractly, between the IP level and link layer processing, there is an entity that examines each IP packet and dispatches it to the appropriate handler. The decision is based on the policy table. For example, the MIP-LR software could capture every IP packet and process every packet that is not related to real-time traffic (i.e., RTP packets or SIP signaling). The real-time traffic passes through untouched, and is redirected by the SIP application when the IP address changes.

Figure 1 shows how SIP and MIP-LR can both manage mobility at the same time for RTP and TCP packets, respectively. Suppose a voice or video session (carried by RTP) and a file transfer (e.g., using ftp over TCP) are in progress at the same time. The MH starts in domain 1, where it is labeled MH (1st), referring to the first phase of its movement. The MH then moves to domain 2, where it is labeled MH (2nd), referring to the second phase of its movement. The solid arrow shows the movement of the MH between domains. When the MH detects that it is in a new domain (after arriving in domain 2), it performs auto-configuration. MIP-LR then updates the CH and HLR(s) with this new address, so the CH can update the destination IP address of the TCP packets. At the same time, SIP (on the MH) issues a re-INVITE and also updates the SIP registrar for location management. The SIP UA on the CH then informs the real-time applications that the address of the MH has changed.

Additionally, for real-time traffic a fast handoff scheme could be deployed [4] without affecting MIP-LR mobility management.

SIP AND MIP-LR INTEGRATION WITH MMP

Global update signaling time in SIP, as in MIP, can result in significant handoff latency. It has previously been suggested at a high level [5] that micromobility schemes could be used together with SIP to improve its performance for micromobility situations. Here we provide the details of how these two can coexist. We also developed a prototype of the integrated approach.



■ Figure 2. SIP/MMP integration call flow.

We consider an example scenario in which an MH moves from one domain to another. While in the first domain, it initiates a SIP session with a CH. The MH then moves into the second domain (macro-mobility), continuing the session. Within the second domain, the MH moves again (micro-mobility), and the session continues. Figure 2 shows the signaling that takes place. The solid arrows show how movement of MH between domains and within domain 2. In general, there might be a number of intermediate route caches between the MH and the gateway in each domain. These are not shown to reduce clutter.

The scenario starts with the MH entering domain 1. From the MMP beacon, it knows it is in a new domain; it auto-configures. There are several ways to do this, and we illustrate an example later, where DRCP [11] is used for auto-configuration. Having obtained a local address in domain 1, IP0, it updates the MMP gateway. It should then send one or more SIP REGISTER messages to appropriate SIP servers (not shown in the figure to reduce the clutter). Some time later, it initiates a SIP session with a CH. After a subsequent move into domain 2, the MH hears the gateway beacon and realizes that it is in a new domain. It auto-configures and sets itself up for micromobility management with its new local address. It then sends a SIP re-INVITE to the CH with its new address, so the SIP handoff can be completed with the CH changing the destination address of the packets it sends to the MH. The MH also sends one or more SIP REGISTER

messages to appropriate SIP servers, which are not shown for brevity. When the MH moves again, it is within domain 2. Hence, it hears the MMP beacon and knows the move is only a local move. Therefore, it only updates the MMP gateway. SIP is completely uninvolved in the process because the IP address is unchanged. Compared to the interdomain handoff, this intradomain handoff occurs with very low handoff delay.

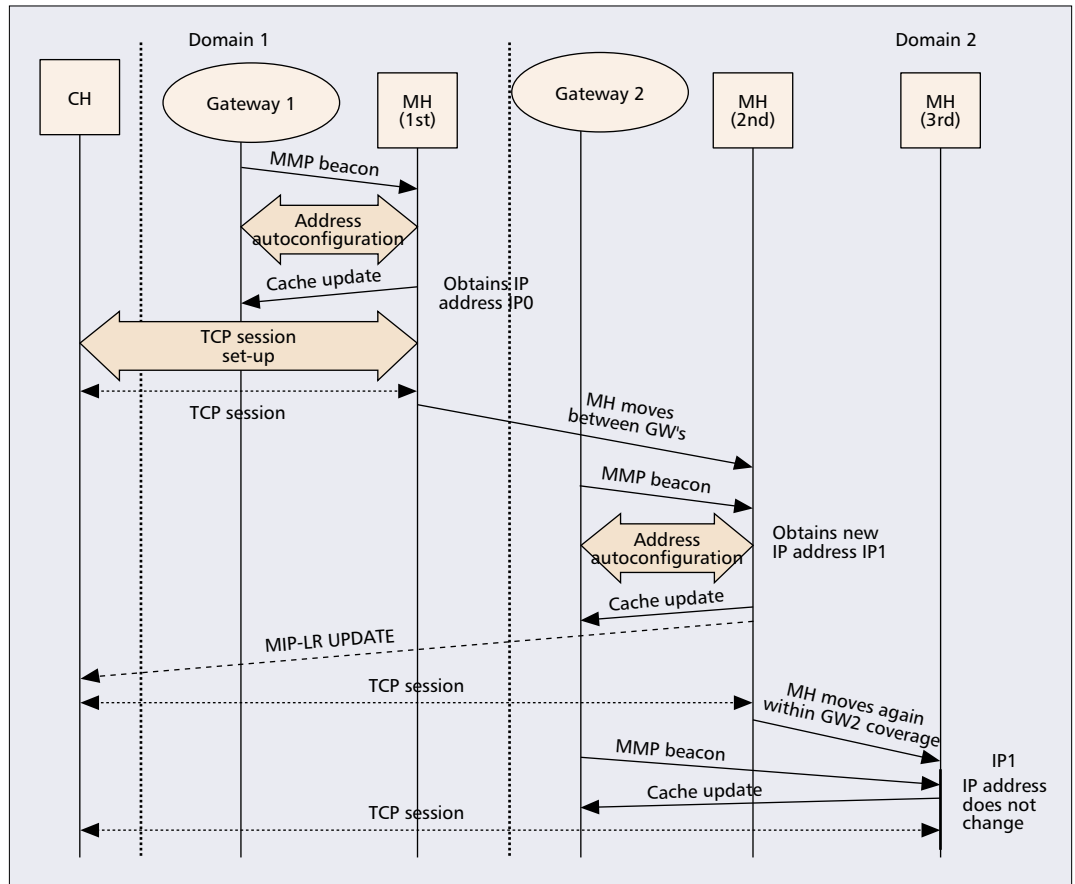
We illustrate the integration of MIP-LR and MMP in Fig. 3. While in the first domain, the MH initiates a TCP session (e.g., a file transfer) with a CH. In domain 1, the MH sends MIP-LR update messages to appropriate HLRs (not shown for brevity). Then it initiates a file transfer session. After moving into domain 2, the MH hears the gateway beacon, auto-configures, and performs micromobility setup signaling. It then sends a MIP-LR UPDATE to the CH with its new address. The MH should also send MIP-LR UPDATE messages to appropriate HLRs.

COMPARISON TO RELATED WORK

Columbia University's work on integrating MIP and Cellular IP considered a foreign agent (FA) collocated with the gateway, and the MH used its home address in the micromobility domain. The MH cannot use its foreign network address in MIP collocated mode in the foreign network, unless Cellular IP is modified. Therefore, it is known that Cellular IP integrates best with MIP with FAs at the gateway, and not using a temporary foreign network address in the micromobility domain [12].

In order to use both SIP and MIP-LR for macro-mobility management, we use a policy table. Abstractly, between the IP-level processing and the link layer processing, there is an entity that examines each IP packet and dispatches it to the appropriate handler. The decision is based on the policy table.

SIP and MMP integration, as well as MIP-LR and MMP integration, avoids the complications of using Cellular IP with co-located care-of-address. This is because at the MH, packets are sent and received using the foreign network address, whereas with MIP, packets are sent using the MH home address.



■ Figure 3. MIP-LR/MMP integration flow.

SIP and MMP integration, as well as MIP-LR and MMP integration, avoids the complications of using Cellular IP with collocated care-of-addresses. This is because at the MH packets are sent and received using the foreign network address, whereas with MIP packets are sent using the MH home address.

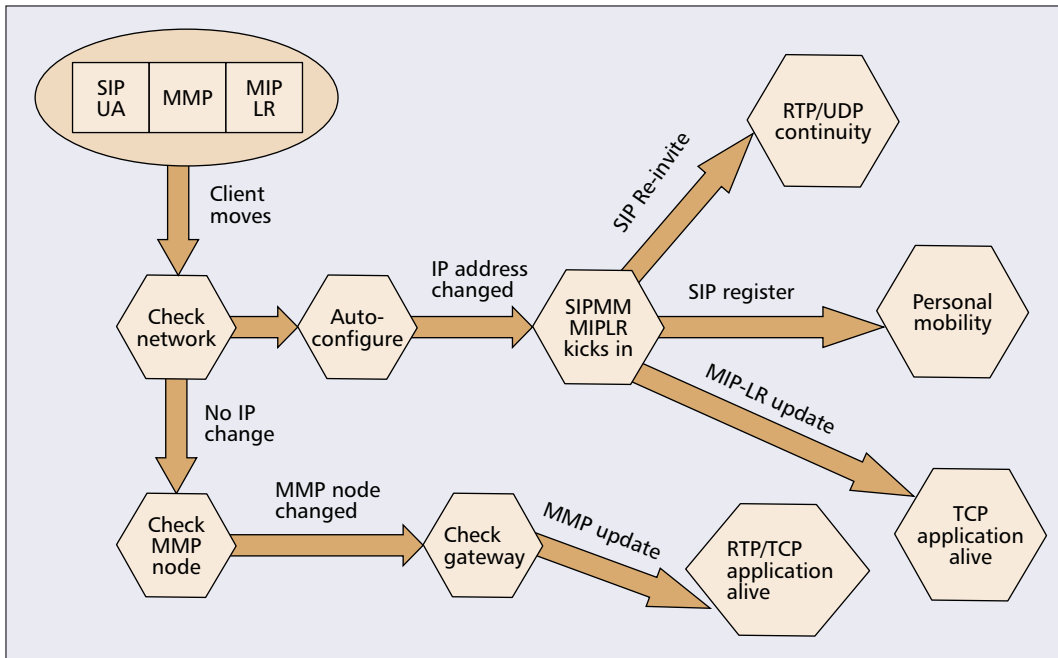
Putting all three components (SIP, MIP-LR, and MMP) together, we have a protocol flow illustrated in Fig. 4. Whenever the MH moves, it checks the network to discover if it is an interdomain or intradomain move it has made (based on MMP beacons or other means). If it is an interdomain move, it auto-configures with a new IP address and establishes MMP connectivity if needed. Then SIP and MIP-LR kick in, as shown in Fig. 1 (whether or not there is a current SIP session, at least the SIP registrar is informed of the move; similarly, whether or not there are current non-real-time sessions, at least the MIP-LR location register is informed of the move). Otherwise, if it is an intradomain move, only MMP updates happen, and the move is transparent to both SIP and MIP-LR.

IMPLEMENTATION AND PERFORMANCE ISSUES

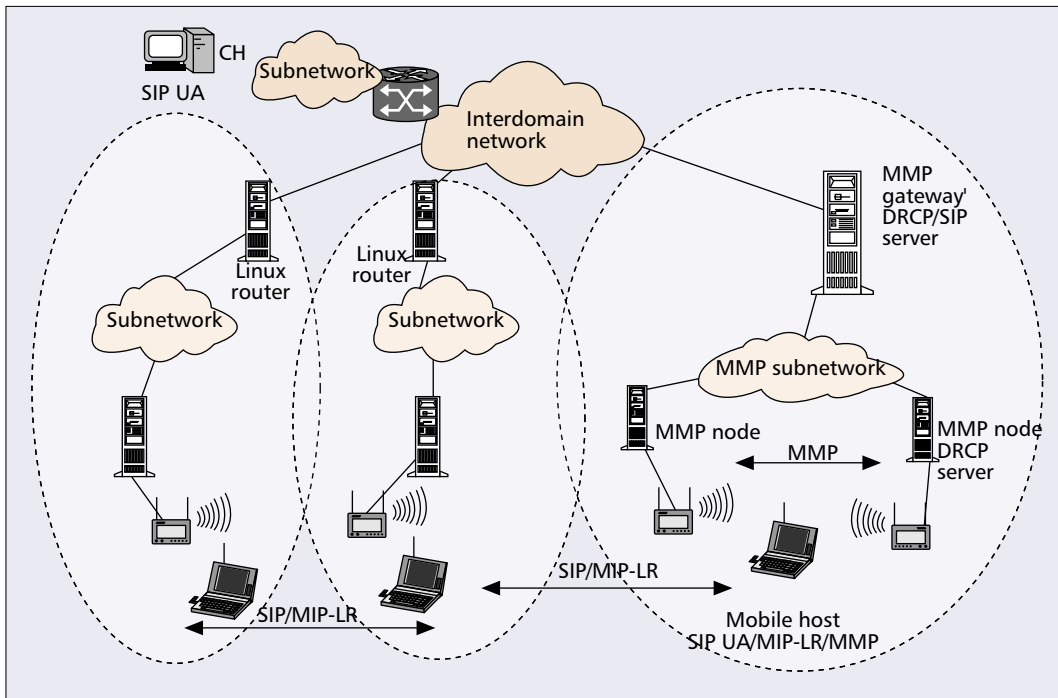
Figure 5 shows the setup of our Linux-based laboratory prototype using 802.11 wireless LAN (WLAN) for the wireless links. IP address management (including auto-configuration) is pro-

vided by DRCP/DCDP servers. DRCP is a version of Dynamic Host Configuration Protocol (DHCP) optimized for wireless environments. A DRCP server configures a node's interface with an IP address, and provides the addresses of DNS server, SIP server, and so on. Dynamic Configuration and Distribution Protocol (DCDP) is a new protocol that distributes pools of IP addresses to the nodes in a quasi-static ad hoc network so that they become DRCP servers. Our implementation of MIP-LR eliminates the tunneling function (and its encapsulation overhead) by using Linux's new libipq and iptables utilities to "mangle" packets (change IP header fields) appropriately at the endpoints.

The MH obtains a new IP address once it moves to a new domain, and keeps this IP address as long as it remains within this domain. This is handled "automatically" by DRCP. As shown in Fig. 4, as the mobile node moves between domains it uses SIP or MIP-LR depending on the type of application being supported. But while moving within a domain, mobility management is handled by MMP, where the gateway would act as a DRCP/DCDP server, and one of the MMP nodes acts as a DRCP server. For convenience in this testbed, all access points within a domain use the same WLAN frequency, whereas access points in different domains use different frequencies. It is reasonable for all access points within a domain to use the same frequency, and micromobility handoff is optimized in this manner, as discussed next.



■ **Figure 4.** Protocol flow of our multilayered mobility management scheme.



■ **Figure 5.** The laboratory prototype of the multilayered mobility management scheme.

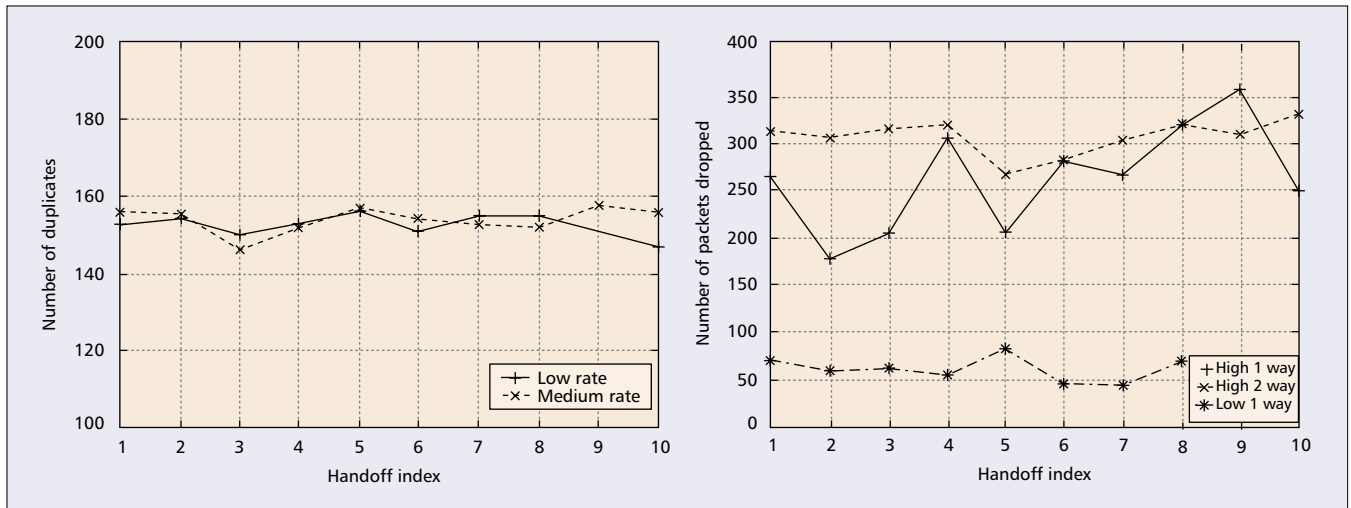
PERFORMANCE

SIP, MIP-LR, and MIP all provide a binding update mechanism that updates a mapping between a permanent address and a temporary one. With SIP, this is done with REGISTER (for pre-session mobility) and re-INVITE (for mid-session mobility). With MIP and MIP-LR, this is done with registration (with HAs and HLRs, respectively). MIP (with route optimization), SIP, and MIP-LR all allow binding updates for CHs to route packets directly to the MHs after mid-session mobility. SIP servers and MIP-

LR location registers can be replicated for survivability.

How well does the new mobility management scheme meet the requirements stipulated earlier? By virtue of the use of macromobility protocols like SIP and MIP-LR, the triangular routing problem of MIP is eliminated. We have found that this significantly increases routing efficiency when the home network of the MH is far from the visited network and the CH is closer to the MH. Our scheme has much less overhead than MIP because encapsulation is not used by any of the component protocols, and the use of micro-

Our implementation of MIP-LR eliminates the tunneling function (and its encapsulation overhead) by using Linux's new libipq and iptables utilities to "mangle" packets (change IP header fields) appropriately at the endpoints.



■ Figure 6. a) Duplicate packets arriving at an MH during micromobility handoff; b) packets dropped during macromobility handoff.

mobility significantly reduces the global signaling overhead. Avoidance of triangular routing and absence of encapsulation contribute to low latency in both real-time and non-real-time communication. The scheme is survivable by having SIP proxies and multiple HLRs that act like dynamic HAs. In general, the MH maintains a current list of SIP proxies or HLRs that can be contacted prior to a session or during communication.

We investigated the performance of the multilayer mobility management scheme using the laboratory testbed. We used SIP to initiate a video session between the MH and CH. During the movement of the MH, both micromobility and macromobility handoffs occurred.

For micromobility handoffs (within a domain), since the two access points are on the same frequency, the handoff does not require a change in frequency. The IP address also remains unchanged. The only difference (for MH transmitting) is that the default gateway and medium access control (MAC) address of the outgoing packets are changed to the new access point. This results in practically no disruption in outgoing packets. For incoming packets, it can receive packets from both access points (same frequency), so we measured no dropped packets. However, there was a short time during the handoff when the same packets were transmitted through both access points, resulting in duplicate packets. Figure 6a shows the number of duplicate packets measured at different handoffs. By “handoff index” we simply mean which handoff out of all the handoffs considered. The variation is low (less than 5 percent), and the number does not change significantly when the video bit rate doubles from 10 kb/s (low rate) to 20 kb/s (medium rate); we suspect this is because the packet size changes, so the packet rate is roughly the same. Duplicate RTP packets should not pose a problem to most streaming video receivers. However, duplicates could be eliminated by performing a hard switch in the MMP gateway between sending the packets to the old and new access points. The trade-off is that there may be a couple of dropped packets if this is done.

Figure 6b, on the other hand, shows handoff behavior when the same MH moves across domains. It acquires a new IP address using DRCP, which triggers SIP handoffs. However, it takes time to change frequencies and resume the physical layer connectivity and then to auto-configure with a new IP address. Furthermore, more packets are lost in the longer “pipeline.” Therefore, dropped packets are observed (the measurements of dropped packets are made at the MH). The high-rate video traffic is 200 kb/s (whether one-way, “high 1 way” or in both directions, “high 2 way”), and the “low 1 way” (low-rate one-way) is 10 kb/s. The rate of dropped packets increases slowly with the data rates. However, a SIP-based fast handoff mechanism [8] can be introduced here to reduce packet loss.

OTHER LESSONS LEARNT

In the course of developing and designing the prototype testbed, we made the following observations:

Care must be taken to be consistent regarding the IP address the MH uses to identify itself in micromobility zones. The IP address the MH uses to identify itself is the address that is stored in the route caches in MMP. When this is its home address, we found it works best with FAs collocated with the micromobility gateway (e.g., MIP-LR can be used with FAs). Otherwise, packets will arrive for the MH addressed to its auto-configured foreign network address, and the route caches need to associate the two addresses (this can be handled by an MMP extension, but is less elegant). Conversely, when identifying itself by its foreign network auto-configured address, it works best without FAs, since the route caches would be set up to forward with the foreign network address in this case.

Separation of real-time and non-real-time traffic is becoming more practically reasonable. With standard tools like iptables for Linux 2.4.7-10 and above, it is easy to set policy-based handling of different types of traffic, say, to do MIP-LR processing only for non-RTP packets, bypassing SIP signaling packets, and RTP packets based on the port numbers.

There are other significant contributors to macromobility handoff latency besides MIP signaling latency. We found that the complete auto-configuration process of IP address distribution using DCDP and IP address configuration using DRCP can take a few seconds, including reconfiguration of the wireless interface. In fact, our testbed typically did not have high network latency, but macromobility handoff latency was still significantly higher than that of micromobility handoff.

Changing the IP address as a result of mobility may require slight application-level changes. For MIP-LR macromobility, applications are unaware of IP address changes with mobility. However, for SIP macromobility we had to modify our video and audio applications (VIC and RAT, respectively, both available as freeware on Linux), and added hooks for interprocess communication with SIP UAs. In general, a mobility-aware RTP stack should be built to adapt itself to IP address change. Some recently built RTP stacks (www.vovida.org) are in fact mobility-aware.

CONCLUSIONS

In this article we introduce a new multilayered mobility management scheme for auto-configured wireless IP networks. Our scheme integrates SIP and MIP-LR for macromobility and MMP for micromobility. This integrated scheme provides the desired features and requirements for a survivable ad hoc network.

REFERENCES

- [1] C. Perkins *et al.*, "IP Mobility Support for IPv4," IETF RFC 3344, Aug. 2002.
- [2] H. Schulzrinne *et al.*, "RTP: A Transport Protocol for Real-Time Applications," IETF RFC 3550, July 2003.
- [3] J. Rosenberg *et al.*, "SIP: Session Initiation Protocol," IETF RFC 3261, June 2002.
- [4] C. Perkins and D. B. Johnson, "Route Optimization in Mobile IP," IETF, Nov. 2000, work in progress.
- [5] H. Schulzrinne and E. Wedlund, "Application-Layer Mobility using SIP," *ACM Mobile Comp. and Commun. Rev.*, vol. 4, no. 3, July 2000, pp. 47-57.
- [6] R. Jain *et al.*, "Mobile Internet Access: Enhancing Performance, Interoperability and Survivability," *IEEE INFOCOM*, Mar. 1999.
- [7] K. D. Wong *et al.*, "Performance of IP Micro-Mobility Management Schemes Using Host Based Routing," *Wireless Pers. Multimedia Commun. 2001*, Aalborg, Denmark, Sept. 2001, pp. 773-89.
- [8] A. Dutta *et al.*, "Optimized Fast-handoff Scheme for Application Layer Mobility Management," *ACM Int'l. Conf. Mobile Comp. (and Mobile Comp. and Commun. Rev.)*, Nov. 2002.
- [9] F. Vakil *et al.*, "Supporting Mobility for TCP with SIP," IETF draft-itsumo-sipping-mobility-tcp-00.txt, June 2001, work in progress.
- [10] P.-Y. Hsieh, A. Dutta, and H. Schulzrinne, "Application Layer Mobility Proxy for real-time communication," *3G Wireless 2003*, San Francisco, CA.
- [11] A. McAuley *et al.*, "Automatic Configuration and Reconfiguration in Dynamic Networks," *Army Science Conf.* 2002, Dec. 2002.
- [12] K.D. Wong, "Architecture Alternatives for Integrating Cellular IP and Mobile IP," *IEEE Int'l. Perf., Comp. and Commun. Conf. 2002*, Phoenix, AZ, Apr. 2002, pp. 197-204.

BIOGRAPHIES

K. DANIEL WONG [SM] (dwong@research.telcordia.com) is currently a research scientist at Telcordia Technologies. He received a B.S.E. degree in electrical engineering (with

highest honors) from Princeton University, New Jersey, in 1992, and a Ph.D. degree in electrical engineering from Stanford University, California, in 1998. His research interests include wireless IP networking, UMTS and WLAN interworking, and mobility management. He has served as Vice Chair of the IEEE New Jersey Coast Section Communications Chapter, and has been active in various IEEE conferences on Technical Program Committee, and as a tutorial presenter and in other roles. He received the Telcordia CEO Award in 2002, and is a member of Tau Beta Pi, Phi Beta Kappa, and Sigma Xi.

ASHUTOSH DUTTA [SM] is currently a research scientist in Telcordia Technology's Internet Network Research Laboratory. Prior to joining Telcordia, he was director of Central Research Facilities at Columbia University, from 1989 to 1997. His research interests include session control protocols, streaming multimedia, wireless multicast, and mobile wireless Internet. He is a recipient of a Telcordia CEO Award and winner of SAIC's ESTC 2002 best paper award. He has a B.S. in electrical engineering (1985) from National Institute of Technology, Rourkela, India, an M.S. in computer science (1989) from New Jersey Institute of Technology, and a professional engineering degree in electrical engineering from Columbia University. He is currently pursuing his Ph.D. degree at Columbia University related to MarconiNet and streaming multimedia. He is a member of ACM and is currently serving as secretary of the IEEE PCJ section.

RAVI JAIN [SM] (ravi.jain@acm.org) received a Ph.D. in computer science from the University of Texas at Austin in 1992. From 1992 to May 2002 he was at Applied Research, Telcordia Technologies, New Jersey, most recently as director of the Middleware and Mobile Applications Research Group. His research interests include design and analysis of algorithms, architectures, and protocols for mobile computing and communications, as well as middleware and APIs for supporting mobile users and for advanced next-generation network applications. He is currently vice president and director in the Autonomous Communications Laboratory, NTT DoCoMo USA Laboratories, San Jose, California. He has numerous publications in his research area and several issued and pending patents. He is a member of Upsilon Pi Epsilon, Phi Kappa Phi, and the ACM.

JAMES E. BURNS is a research scientist in the Software Technologies Group at Telcordia Technologies. He has been a long time researcher in distributed computing systems, especially in regard to self-stabilizing systems, a specialized area within fault tolerance. He has been involved in developing and simulating protocols for new and evolving areas, with an emphasis on next-generation networks. An important part of this work is the analysis of live network traffic in order to understand the complex interactions of different protocol mixes over the Internet. Other projects have been concerned with wireless communication and securing computer networks. He received a Ph.D. in information and computer science from Georgia Institute of Technology and a B.S. from California Institute of Technology.

KEN YOUNG [SM] is executive director of Government Project Development at Telcordia Technologies. He currently leads research on mobile ad hoc networking technology for the Army Research Laboratory Communications and Networks Collaborative Technology Alliance, the CERDEC Multifunctional On-the-Move Secure Adaptive Integrated Communications (MOSAIC) Advanced Technology Demonstration, and the Adaptive Joint C4ISR Node (AJCN) program. He has more than 60 research publications. He received a Ph.D. in physics from the University of Pennsylvania in 1978 and a B. S. in physics from St. Joseph's University in 1972. He chairs the Communications Society's Tactical Communications Technical Committee and is a member of the American Physical Society.

HENNING SCHULZRINNE (hgs@cs.columbia.edu) is an associate professor in the Departments of Electrical Engineering and Computer Science of Columbia University, New York. He received his Ph.D. from the University of Massachusetts. He has worked at Bell Laboratories, Murray Hill, and GMD Fokus, Berlin. His research interests include Internet multimedia and telephony services, signaling, network QoS, scheduling, multicast, and performance evaluation. He is a co-author of the Internet standards track protocols RTP, RTSP, and SIP.

Our scheme integrates SIP and MIP-LR for macro-mobility and MMP for micro-mobility. This integrated scheme provides the desired features and requirements for a survivable ad hoc network.