ELSEVIER

Computer Networks 38 (2002) 181–206

COMPUTER NETWORKS

www.elsevier.com/locate/comnet

A network infrastructure for IP mobility support in metropolitan areas

Cristina Hristea [*], Fouad Tobagi

Department of Electrical Engineering, Stanford University, Stanford, CA 94305-9030, USA

Received 24 August 2001; accepted 27 August 2001

Responsible Editor: I.F. Akyildiz

Abstract

The original design of the Internet and its underlying protocols did not anticipate users to be mobile. With the growing interest in supporting mobile users and mobile computing, a great deal of work is taking place to solve this problem. For a solution to be practical, it has to integrate easily with existing Internet infrastructure and protocols, and offer an adequate migration path toward what might represent the ultimate solution. In that respect, the solution has to be incrementally scalable to handle a large number of mobile users and wide geographical scopes, and well performing so as to support all application requirements including voice and video communications and a wide range of mobility speeds. In this paper, we present a survey of the state-of-the-art and propose a scalable infrastructure to support mobility in Internet protocol networks. In that respect, we exploit local area network (LAN) technologies to create the network infrastructure necessary to offer connectivity to mobile users across any geographical area (building, campus and metropolis). The intrinsic properties of LAN technologies and their underlying protocols, namely flat address space, transparent learning and low complexity renders this solution particularly cost effective for supporting user mobility. In particular, we propose a network topology and a set of protocols that render the infrastructure scalable to a large geographical area and many users. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Mobility; IP networks; Local and metropolitan area networks; Overlay infrastructure; Scalability; Seamlessness

1. Introduction

In the broadest sense, the term mobile networking refers to a networking technology that allows users to maintain uninterrupted end-to-end connectivity while moving (at any speed) over a geographical area of any scope (building, campus, metropolis or even the entire world). Such technology is often associated with wireless communication. Mobile networking already exists today for voice communication services offered with cellular telephony, however it relies on circuit switching, which is appropriate for handling voice traffic. With the growing interest in providing data communication services and access to the Internet to wireless mobile users, there is a need to seek a packet-based mobile networking solution to support any communication service. Indeed, packet switching and the Internet protocol (IP) are quickly becoming the networking standards towards which many networks are converging,

---

[*] Corresponding author.

E-mail addresses: christea@stanford.edu (C. Hristea), tobagi@stanford.edu (F. Tobagi).

1389-1286/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.

PII: S1389-1286(01)00257-2

including those used by the traditional telecommunication sector, as less appropriate circuit-switched technologies are abandoned. Interest in having mobile users access the Internet is evidenced by the increase in cellular phones with built-in Internet access, PDAs and laptops with wireless interfaces and car navigation systems also with Internet connectivity.

Although a lot of progress has been made towards supporting mobility in IP networks, this problem still poses difficult challenges [17]. Such challenges stem from some properties of the IP and related implementation practices, as explained in what follows. The IP addressing scheme was designed and optimized for a stationary environment (i.e. in which hosts are fixed). When a host, known as the source, sends a packet to another host, known as the destination, the source has to insert the IP address of the destination in the packet header, to be used by routers in the Internet for packet routing and delivery. In that respect, routers have to be able to route to destinations all over the Internet. To aid in this task, which can pose significant scalability problems, IP addresses are hierarchical, which allows routing decisions to be performed using address aggregates. Indeed, IP addresses consist of fixed fields, which are used individually by routers in the Internet as follows: the first field is the network number, used by routers outside the network to reach that network and thus all hosts within that network; the second field is the sub-network (subnet) field, used by routers within a network to reach that subnet, and thus all hosts within that subnet; finally, the third field is the host field, used by routers inside the subnet to reach the individual host. Due to this addressing structure, IP addresses need to change dynamically as users move from one IP subnet to another (or one network to another), to match addressing at the new subnets (or the new network). [1] If a source sends packets to a destination that is moving and changing IP addresses, the contents of the packet header may refer to a destination address that is stale and the packet may need to be discarded.

One possible solution to this problem is to notify the source of any changes in the IP addresses of the destination. However, packets that are in the process of travelling towards their destination while this destination changes IP addresses, or that are sent by the source prior to its receipt of this notification, will have incorrect header contents and hence will be directed to the wrong IP address. Moreover, requiring that a new IP address be used for communication can pose problems to the higher layers (above IP) in the protocol stack. Indeed, while in principle, higher layers are supposed to be independent of the IP layer, in practice they make use of the IP address for identification purposes. We refer to this problem as the dual significance of IP addressing. For example, the transport layer uses the combination of IP address and port number at both the source and destination of a connection for identifying that connection. If the IP address is changed during the course of that connection, its identity becomes invalid and the connection is lost. Therefore, to maintain seamless connectivity during movement, IP addresses need to be kept fixed, transparent to changes in user locations. [2]

Another possibility is to relax the hierarchical structure of IP addressing and maintain fixed IP addresses at the source and destination while they move. This can be achieved by informing some routers along the path between the connection source and destination of changes in their location. These routers then would perform host-specific routing to properly deliver packets towards their destination. Since it takes away the ability of some routers to exploit the hierarchical nature of IP addressing, this solution may not scale however to a large number of hosts. Host-specific routing requires space in the routing tables proportional to the number of hosts, slows down the routing

---

[1] Multicast (class D) addresses constitute an exception to this rule. The same multicast address can be used, even when crossing different subnets.

[2] Note that, to resolve this duality, in IPv6 a host is allowed to use two addresses. One address is used as a permanent identifier while the other address is used for routing purposes. The permanent address is included in the main IPv6 header, while the routing address is inserted in a special-purpose extension header used for routing.

process and consumes potentially excessive bandwidth in the Internet (due to the constant updating of routing tables for individual host entries).

Mobile IP [4] proposes a way to resolve these challenges without requiring that routers in the Internet learn host-specific addresses. A mobile host (MH) is allowed to hold two addresses simultaneously (this requires software changes at the MH): a permanent address (also known as a home address) and a temporary addresses that conforms with addressing at the subnet the mobile is visiting (known as a care-of-address or COA). Servers known as home agents and foreign agents (HAs and FAs) are deployed at every subnet wishing to support MH. A host registers its home address with the HA of its home subnet. When it enters a visiting subnet, the FA in that subnet provides the MH with a COA for that subnet. Then, the FA or the MH itself inform the HA of the COA of that mobile. The HA maintains an up-to-date database mapping the home address of the mobile user to its COA. Furthermore, when a MH is away, the HA intercepts packets sent to the home address of that MH and tunnels it to the COA of that host. The FA or the MH de-tunnels this packet for final packet delivery. The Mobile IP solution uses an overlay of special-purpose agents that interfaces with the Internet infrastructure, where no changes are required. Thus, Mobile IP is a modular solution to the problem of supporting mobility in IP networks that integrates well with the existing Internet.

Final challenges to be considered are the need to support wireless users, frequently changing their points of attachment to the Internet and multimedia applications with stringent performance constraints, such as low packet loss and high interactivity. As users move, handoff needs to take place between the user's old point of attachment to the network and the new point of attachment to the network. Handoff may require change of state, not only in the vicinity where the user is immediately connected, but also further out in the Internet (like the HA in Mobile IP). If the number of such changes is large, or the distance from the user to the points where changes are needed is large, this change of state can take a long time. An interruption in connectivity due to a slow handoff can cause packet loss, which can significantly lower the perceived quality of these applications by the user. Packet buffering is typically used to handle packet loss. However, packet buffering may result in excessive latency overhead. Real-time applications such as voice depend on packets being delivered at a constant rate and within a certain time budget. If at times of user movement, the network cannot ensure the timely delivery of packets, they become irrelevant and would need to be discarded, to the dissatisfaction of users. Therefore, to assist moving users and maintain the continuity of multimedia traffic, the solution needs to support fast handoffs. To achieve this, handoffs should not involve propagation of information over long distances (hence should be handled in the vicinity of the user location). Also in order to achieve smooth handoffs, it may be necessary for the mobile user to be connected to multiple points of attachments (referred to as diversity in the literature). However, this becomes difficult as speeds increase and users move continuously across space, frequently changing their points of attachment.

The introduction of overlay agents in Mobile IP can make meeting these last challenges difficult, because it adds performance overhead, for a number of reasons: the need to acquire a new IP address every time a mobile crosses subnets, which could be very frequent due to the small size of many subnets deployed in the Internet, and the need to tunnel packets through agents, in order to properly interface to the routers in the Internet. If the HA is used to track mobile users, and happens to be located far away from the location of mobile users, the HA may be slow at receiving location updates for those users, affecting their ability to communicate in a timely manner. To address this issue, Mobile IP allows FAs to be chained, for the purpose of user tracking. Instead of informing the HA, the FA at the new subnet has the option to inform the FA at the subnet that the user visited previously. The FAs need to map the old COA to the new COA of mobile users. The optimization however increases the complexity at the agents and the mobile users and can further slow down the process of packet routing, which can be a problem for real-time applications. More discussion of Mobile IP will take place in Section 2 of this paper.

Many researchers have investigated ways to reduce the overhead of Mobile IP. These schemes seek to accelerate Mobile IP for the cases where frequent movement takes place within a confined geographical region [9,12]. This is achieved in part by allowing MHs to maintain a fixed COA within that region, which reduces the overhead of acquiring new IP addresses and making the necessary changes in the supporting infrastructure to enable the use this new address. Because they are aimed at improving support for movement on a small scale (within a region), such schemes are known as *micro-mobility* schemes. The basic idea is shown in Fig. 1. In the Section 2 we present an in-depth discussion of related work; in this section, we chose to make some high-level comments regarding two micro-mobility schemes, namely Hawaii [12] and Cellular IP [9].

In Hawaii [12], routers that make up the Internet infrastructure of a region, which in fact maps to an IP domain, are used for mobile user tracking within that region. The routers perform host-specific routing to allow users to maintain a fixed IP address while moving inside the domain. By working at the level of the Internet infrastructure, instead of relying on an overlay infrastructure like in Mobile IP, Hawaii may represent a simpler, and more cost effective solution for supporting micro-mobility. The disadvantages of Hawaii stem from the use of host-specific routing, which as pointed out earlier suffers from lack of scalability and inefficiency. Hawaii localizes the use of host-specific routing to within a region of a small scale (an IP domain), in order to reduce the impact of these problems. As a result, the acceleration of Mobile IP can be sustained only within regions of that scale.

In Cellular IP [9], a different approach is taken to accelerate micro-mobility. Cellular IP proposes the deployment of a special-purpose access network, built as one IP subnet, to support wireless mobile users inside a region of some geographical scope. Cellular IP proposes a set of protocols for the operation of this access network. We observe that these protocols are similar to the 802 standards, such as the spanning tree protocol used to compute connectivity inside a subnet. Due to the nature of its design based on layer-2 technology, the access network used in Cellular IP may represent a cost-effective way to support mobility in that region. However, Cellular IP poses some reliability and scalability problems, which are common to LANs, stemming from a number of reasons such as: the computation of a spanning tree, which requires traversal of the entire subnet, the use of a single spanning tree for connectivity, and the use of broadcasts.

In that respect, what becomes important is the design of a scalable, yet cost-effective overlay infrastructure (i.e. its interconnect and operational pro-

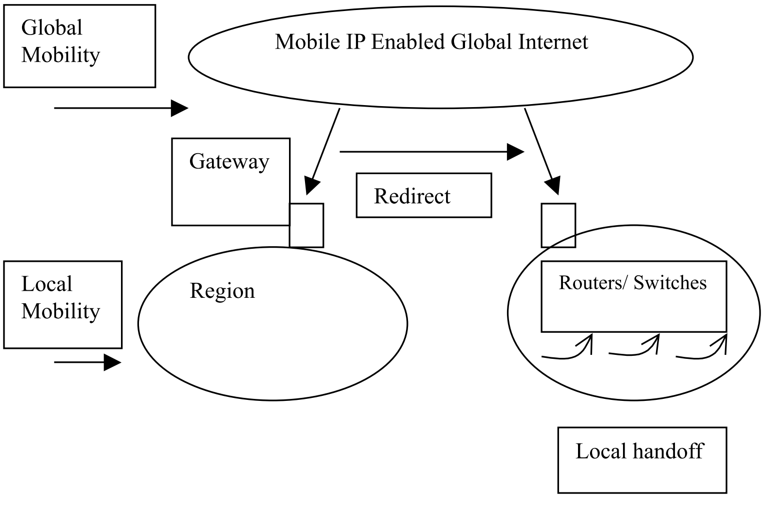

Fig. 1. Typical micro-mobility architecture.

tocols) based on layer-2 technology, to support IP mobility in regions of large geographical scope. This is the main goal of our work described in this paper.

Another technique to improve micro-mobility support is to ensure that DNS holds information on the most up-to-date location of users, such as their temporary IP address (via dynamic DNS updates). This way, users can be reached directly for more efficient communication (e.g. the HA can be bypassed). Along with improving the communication speed between users, this technique reduces the load in the Internet since it eliminates the need for tunneling. It is interesting to note the parallels that exist between dynamic DNS solutions and directories techniques in cellular telephone systems such as the groupe speciale mobile (GSM) or the personal communication system (PCS) [11,13]. In the GSM and PCS systems, users are identified by a unique phone number. One important feature of the GSM/PCS system is the automatic, worldwide localization of users. The system knows where a user currently is, and the same phone number is valid worldwide. A hierarchy of databases consisting of home location registers (HLRs) and visiting location registers (VLRs) is used to track users. The HLR contains information about the current VLR of a user, and the VLR knows the switching center via which the user can be reached. The HLR/VLR databases are similar to the dynamic DNS directories because they store up-to-date information on the location of every user. Notice that this feature of GSM/PCS renders the mobility management scheme a challenging problem. While this approach eliminates system-wide paging, which vastly reduces the radio link signaling, it introduces remote database lookups that may incur a large amount of wired network traffic and long call setup delay. Much effort has been focused on exploring efficient location management techniques. Extensions to standard HLR/VLR schemes, such as partial replication and caching have been developed to improve wireless call setup performance [13]. We believe that these techniques may apply to the IP world in order to achieve an efficient and scalable dynamic DNS implementation.

A radically different way to solve the challenges of supporting mobility in IP networks, proposed in [16], is to modify the higher layers such as TCP to use domain names instead of IP addresses, and to change the routers in the Internet to use domain-name routing. In addition, dynamic DNS is used to track changes in mobile IP addresses. Routers perform DNS lookup to resolve the domain name to the IP address that it can use to route appropriately. Because of its radical nature, this solution does not integrate well with the existing Internet infrastructure and hence it may be prohibitively expensive to deploy.

A different solution that also requires changes at the TCP layer is presented in [3]. Connection migration is performed to maintain connectivity for sessions in-flight at the time of move. Dynamic DNS updates are used to ensure that hosts can be reached directly (as mentioned before, this eliminates the need for HAs). For this solution to work however, the MHs, and fixed hosts in the Internet wishing to communicate with MHs would need to be upgraded to the new versions of software. While upgrading the MHs may be an easier task, upgrading all the hosts in the Internet is not a possibility. Furthermore, achieving good application performance with dynamic IP addresses remains a significant challenge. Simulation results show that significant disruption is incurred during connection migration; moreover the solution limits movement to a single end and also may apply to TCP applications only.

The rest of the paper has the following outline: Section 2 describes details of the state-of-the-art; Section 3 presents our solution for supporting mobility in IP networks; Sections 4–6 describe the experimental methodology, the results gathered while experimenting with our solution, and a summary of its contributions; Section 7 describes challenges posed by the process of migrating this technology from research to practice; finally overall conclusions are presented in Section 8.

2. Related work

2.1. Mobile IP

In the 1990s, the IETF designed a solution for mobility known as Mobile IP [4], which overcomes the challenges of supporting mobility in IP networks without requiring that routers in the Internet learn host-specific routes. Mobile IP solves the problem of IP address duality by allowing a single computer to hold two addresses simultaneously. The first address is permanent and fixed. It is the address that transport and application protocols use. The second address is temporary—it changes as the computer moves, and is valid only while the computer visits a given location.

A MH is assigned a permanent home address and a HA in its home subnet. DNS maps the domain name of the host to its home address. When the MH moves to a foreign subnet, it acquires a temporary COA from an advertised FA in the subnet, and it registers its new address with the HA. The HA uses gratuitous proxy ARP to capture all IP packets addressed to the permanent address of the MH and uses encapsulation, (i.e. IP-in-IP tunneling) to forward them to the current location of the mobile. [3] The process of sending packets to the MH via the HA is known as triangle routing [5].

There are two possibilities for the packets going back from the MH to the corresponding host (CH). One choice is for the MH to send out unencapsulated IP packets with the permanent home address of the MH as the source address and the address of the CH as the destination address. However, some routers in the Internet use address based filtering and discard packets from outside the subnet. To avoid this, the MH needs to encapsulate the packet using its COA as the source address and the address of the HA as the destination address. The HA decapsulates the packet and forwards it to the CH.

One thing to notice is that packets delivered via HA typically travel further through the Internet than they would if delivered by the optimal unicast route. Apart from increasing the round-trip delay observed by the communicating parties, this also affects other users by increasing the overall load on the shared resources of the Internet. A proposed mechanism, known as route optimization [19], attempts to fix this, by having the HA communicate the current COA to the CH by means of the so-called binding updates. A CH with enhanced networking software can then learn the temporary COA and perform the encapsulation itself, sending the packet directly to the MH. This avoids the overhead of indirect delivery. Moreover, as users depart from one subnet and join another, binding updates can also be used to inform an old FA of the new COA of the departed user, allowing it to perform forwarding of packets communicated to the old subnet. This optimization reduces the overhead of the handoff process.

By design, Mobile IP focuses on the problem of long-duration moves, rather than attempting to handle rapid network transitions such as the ones encountered in a wireless cellular system. Because it requires considerable overhead after each move, Mobile IP may not meet the challenge of supporting mobility for wireless users. This overhead arises for a number of reasons, namely: after it moves to a new subnet, a MH must detect that it has moved, communicate across the foreign network to obtain a COA and then communicate across the Internet to its HA to arrange forwarding.

To address this issue, a proposal was introduced in [1], known as hierarchical foreign agents or HFA. In this proposal, hierarchical foreign agents (HFAs) are introduced to smooth out the handoff process when a MH transitions between subnets. This optimization is accomplished via hierarchical tracking of MHs by the FAs, and via packet buffering at FAs.

Within a domain, the FAs are organized into a tree structure that handles all the handoffs in that domain, with one FA serving as the root of the tree. (The tree organization is unspecified and left up to the network administrator of that domain. One popular configuration is to have a FA associated with the firewall to that domain be the root of the tree—also known as a gateway foreign agent or GFA—and all the other FAs provide the second level of the hierarchy.)

A FA sends advertisements called Agent Advertisements in order to signal its presence to the MHs. An Agent Advertisement includes a vector of COAs, which are the IP addresses of all its ancestors (up to the root) as well as the IP address

---

[3] Note that for IPv6, the extension header plays the role of the encapsulation header in Mobile IP.

of that FA. When an MH arrives at a subnet, it records these IP addresses as its COAs. A registration goes through and is processed by the FA, all its ancestors and the HA. The FA processing a registration records the next lower-level FA as the COA to which to forward packets sent for the MH, and the HA records the root FA for this purpose. A registration reply goes through in the opposite direction from the HA to the MH. When a handoff occurs, MH compares the new vector of COAs with the old one it had recorded previously. It chooses the lowest-level FA that appears in both vectors, and sends a regional registration request to that FA, and all the lower level FAs specified in the new vector. Any higher-level agent needs not be informed of this movement.

When a packet for the MH arrives at its home network, the HA tunnels it to the root FA. The root FA re-tunnels it to the lower-level FA, which in turn re-tunnels it to the next lower level FA. Finally, the lowest-level FA delivers it to the MH.

Mobile IP route optimization [19] extends the use of binding cache and binding update messages to provide smooth handoff. However, tunneled packets that arrive at the previous FA before the arrival of the required binding update are lost. To address this issue, HFA includes a mechanism for packet buffering at the FAs. Besides decapsulating tunneled packets and delivering them directly to an MH, the FA also buffers these packets. When it receives a binding update with a new COA, it re-tunnels to that COA the buffered packets along with any future packets tunneled to it. Clearly, how much packet loss can be avoided depends on how quickly an MH finds a new FA, and how many packets are buffered at the previous FA. This in turn depends on how frequently FAs send out beacons or agent advertisements, and how long the MH stays out of range of any FA. To reduce duplicates, the MH buffers the identification and source address fields in the IP headers of the packets it receives and includes them in the buffer handoff request so that the previous FA does not need to retransmit those packets that the MH has already received.

While HFA helps reduce the overhead of handoff by handling handoff closer to the MH, it introduces additional complexity at the MH, and in constructing and maintaining the FA hierarchy, and it may add latency due to the need for packet encapsulation and decapsulation at every FA in the FA tree along the path from the CH to the MH. Moreover, scalability issues may arise at the root FA and the FAs close to the root of the FA tree because of their involvement in packet tunneling for all the MHs of that domain. Finally, packet buffering can result in latency overhead, while encapsulation still generates bandwidth overhead.

Finally, we note that Mobile IP and solutions similar to it are popular choices employed in products and standards aimed to offer IP connectivity to users around the world. In what follows, we describe two examples, namely the Ricochet system from Metricom [16] and universal mobile telecommunication system (UMTS) [10].

2.1.1. Ricochet

The Ricochet system offered by Metricom [15] implements a solution for IP mobility that is somewhat similar to the Mobile IP protocol. It was designed more than a decade ago, and thus predates the Mobile IP protocol. Wireless cells are connected to IP gateways and name servers that provide security, authorization and roaming support to users. When a user first connects to the network, it sends a request to the local gateway and the name server. If validated and authorized, this request allows the user to receive an IP address to connect to the Internet. The IP address identifies a permanent connection between the user and the network and thus remains fixed for the entire time the user is connected to the Internet. All Internet traffic for the user is tunneled through the gateway to which the user was originally connected. Beside the fixed IP address, a MH is given two layer-2 addresses: one is fixed and unique to that user, and the other is dynamic and unique to the cell where a user is located at that point in time. The gateway maps the IP address of the user to the layer-2 address corresponding to the cell where the user is located at any point in time. As the user crosses cells, this mapping changes to reflect the new location of the user. In essence, this gateway performs the function of the HA in Mobile IP, by maintaining an up-to-date mapping of a fixed user address to its temporary address, and by redirecting traffic received for the fixed address of the user, to the temporary address of that user.

2.1.2. Universal mobile telecommunication system

One example of a standard that may employ the Mobile IP protocol is the UMTS, which is proposed in [10]. UMTS aims to provide IP level services via virtual connections between mobile hosts and IP gateways connected to ISPs or corporate networks at the edges of the mobile network. Consistent with the Internet architecture, a user is assigned domain names in order to identify the ISP that can be accessed to provide Internet connectivity to that user. When a user logs on, it is assigned an IP address by the gateway to which that ISP is connected, also known as the home gateway. A connection is established, consisting of two segments: one segment connects the mobile and some foreign gateway, and another, connects the foreign gateway and the home gateway, via the Mobile IP protocol. The connection is maintained as long as the mobile remains on and the foreign gateway can be changed as the mobile roams from the coverage area of one gateway to another. One can think of the mobile as being linked to the home gateway via an elastic global pipe. To the external world, the mobile appears to be located at the home gateway because it is this gateway that provides the IP address for the mobile.

2.2. Cellular IP

Cellular IP [9] is a micro-mobility scheme that aims to accelerate Mobile IP by deploying special-purpose access networks, built as single IP subnets, across some geographical regions. The description of Cellular IP assumes that originally, each wireless cell (or even pico-cell) constituted an IP subnet. Consequently, they propose that multiple wireless cells be grouped into one subnet to improve roaming between the cells of one subnet. However, this concept is not completely new. For example, the 802.11 standard uses extended service sets (ESS) to interconnect multiple 802.11 cells within a single subnet.

Cellular IP access networks, depicted in Fig. 2, are connected to the Internet via gateway routers. Cellular IP uses base stations (BSs) for wireless access connectivity, and for mobility support inside an access network. MHs attached to an access network use the IP address of the gateway as their COA. The gateway de-tunnels packets and forwards them toward a BS. Inside a Cellular IP network, MHs are identified by their permanent home address and data packets routed without tunneling or address conversion. The Cellular IP routing protocol ensures that packets are delivered to the actual location of the host. Vice versa, packets sent by the MH are directed to the gateway, and from there, to the Internet.

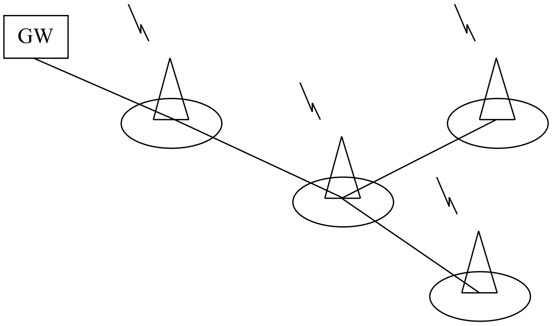

Fig. 2. A wireless access network in Cellular IP.

Cellular IP proposes methods for computing connectivity among base stations and for routing packets inside an access network. Periodically, the gateway sends out beacons that are broadcasted across the access network. Through this procedure, BSs learn about neighboring BSs on the path towards the gateway. They use this information when forwarding packets to the gateway. Moreover, when forwarding data packets from users to the gateway, BSs learn about the location of a user, and use that information to deliver packets sent for that user. We remark that, although Cellular IP proposes protocols for configuration and routing in IP subnets, LAN protocols already exist to accomplish these goals. For example, the algorithms for building a spanning tree and for learning as defined by the 802 standards are widely deployed and well known.

If a packet is received at a BS for a user that is unknown to that BS, a paging request is initiated by the BS. The paging request is broadcast across a limited area in the access network known as a paging area. The MH responds to the paging request and its route to the paging BS gets established. Each MH needs to register with a paging area when it first enters that area, regardless of whether it is engaged in communication or idle. Clearly, how fast paging occurs depends on the size of the paging area and on the efficiency of spanning tree traversal. A small paging area can help reduce the latency of paging, however it increases the number of paging area required to cover a given area, which in turn increases the signaling overhead imposed on MHs.

We observe that the paging techniques in Cellular IP are similar to those existent in the GSM system [11]. In GSM, mobile users are located in system-defined areas called cells that are grouped in paging areas. Every user connects with the base station in his cell through the wireless medium. BSs in a given paging area are connected by a fixed wired network to a switching center, and exchange data to perform call setups and deliver calls between different cells. When a call arrives at the switching center for a given user, a paging request for that user is initiated across all the cells in that paging area. If the user answers, a security check on the user is performed, and if the test passes, the switching center sets up a connection for that user.

Cellular IP supports two types of handoff: hard handoff and semisoft handoff. MHs listen to beacons transmitted by BSs and initiate handoff based on signal strength measurements. To perform a handoff, the MH tunes its radio to the new BS and sends a registration message that is used to create routing entries along the path to the gateway. Packets that are received at a BS prior to the location update are lost. Just like in Mobile IP, packet loss can be reduced by notifying the old BS of the pending handoff to the new BS, and requesting that the old BS forward those packets to that BS. Another possibility is to allow for the old route to remain valid until the handoff is established. This is known as semisoft handoff and is initiated by the MH sending a semisoft handoff packet to the new BS while still listening to the old BS. After a semisoft delay, the MH sends a hard handoff packet. The purpose of the semisoft packet is to establish parts of the new route (to some uplink BS). During the semisoft delay time, the MH may be receiving packets from both BSs. The success of this scheme in minimizing packet loss depends on both the network topology and the value of the semisoft delay. While a large value can eliminate packet loss, it however adds burden on the wireless network by consuming precious bandwidth.

Cellular IP specifies an algorithm to build a single spanning tree rooted at the gateway to the access network as we described above. A spanning tree is necessary for the broadcasting of packets, to avoid packets from propagating to infinity if the topology of the access network has any loops. However, because it uses only a subset of the links inside the access network, a single spanning tree can result in link overload if traffic in the access network is high. This can be a significant drawback of Cellular IP as high-density access networks supporting many Tb/s of traffic become possible to deploy. Moreover, a single spanning tree can be prone to long periods of connectivity loss. Connectivity loss would make this technology unacceptable as a replacement to wired, circuit-switched technology for telephone communications. Finally, Cellular IP specifies an interconnect between BSs that has a flat hierarchy. As access networks cover larger area and exhibit higher pico-cell densities, a flat hierarchy would result in latencies of packet traversal across the access network that may be unacceptable.

In conclusion, it is clear that deploying wireless access networks as single subnets, like in Cellular IP is important for mobility. In this light, it becomes important to increase the size of IP subnets to the largest size possible in order to maximize their effectiveness in supporting IP mobility.

2.3. Hawaii

Hawaii [12] is a micro-mobility scheme intended to accelerate Mobile IP inside portions of the Internet by exploiting host-specific routing at the routers present at those locations. Hawaii segregates the network into a hierarchy of domains, loosely modeled on the autonomous system hierarchy used in the Internet [12]. The gateway into each domain is called the domain root router. When moving inside a foreign domain, an MH

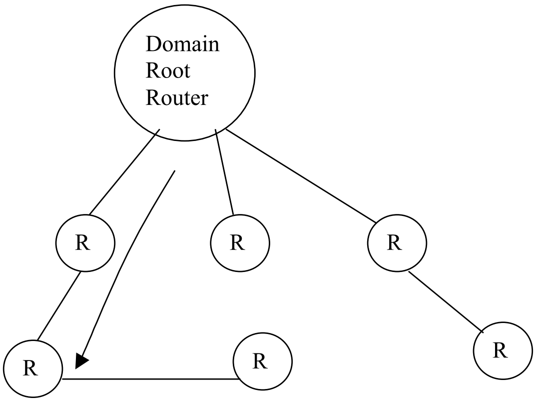

Fig. 3. Diagram of a domain in the Hawaii architecture. A domain root router acts as the gateway to each domain. Paths are established between the routers of a domain.

retains its COA unchanged and connectivity is made possible via dynamically established paths, as shown in Fig. 3. Path-setup update messages are used to establish and update host-based routing entries for the MHs at selective routers in the domain, so that packets arriving at the domain root router can reach the MH. The choice of when, how and which routers are updated constitutes a particular setup scheme. Hawaii describes four such setup schemes, which trade-off efficiency of packet delivery and packet loss during handoff. The MH sends a path setup message, which establishes host specific routes for that MH at the domain root router and any intermediary routers on the path towards the MH. Other routers in the domain have no knowledge of the IP address of that MH. Moreover, the HA and CH are unaware of intra-domain mobility. We observe that, by sending path setup messages to a router in the tree, the MH is doing what a router would normally do in OSPF to load state for that MH, but without having to flood this information to all the routers inside the domain. The state maintained at the routers is soft: the MH infrequently sends periodic refresh messages to the local BS. In turn, the BS and intermediary routers send periodic aggregate hop-by-hop refresh messages toward the domain root router. Furthermore, reliability is achieved through maintaining soft-state forwarding entries for the MHs and leveraging fault detection mechanisms built in existing intra-domain routing protocols.

In conclusion, Hawaii exploits host-specific routing to deliver micro-mobility. By design, routers perform prefix routing to allow for a large number of hosts to be supported in the Internet. While routing based on host-specific addresses can also be performed at a router, it is normally discouraged, because it violates the principle of prefix routing. Furthermore, host-specific routing is limited by the small number of host-specific entries that can be supported in a given router. However, this concern can be addressed by appropriate sizing of the domain and by carefully choosing the routers that are updated when a mobile is handed off. One of the problems with the implementation of Hawaii is that a single domain root router is used. This router, as well as its neighbors inside the routing tree can become bottlenecks routers for the domain for two reasons: first, they hold routing entries for all the mobile users inside the domain, second, they participate in the handling of all control and data packets for the mobile users in that domain. Another disadvantage of Hawaii comes from its use of routers as a foundation for micro-mobility support. With cells becoming smaller, it is possible that a larger number of routers would be needed for user tracking and routing in a given area; however, this can become prohibitively expensive.

2.4. *Multicast-based mobility*

Numerous solutions for supporting mobility via multicast routing have been proposed [2,6,7]. In [2,6], multicasting is used as a sole mechanism to provide addressing and routing services to MHs in the Internet. Each MH is assigned a unique multicast address. In [2], a MH initiates a group membership registration with the multicast router in its subnet (using the IGMP protocol). In turn, this router informs neighboring multicast routers about the multicast group (via one of several known protocols such as MOSPF and PIM). Multicast routers in the neighborhood of the user join this multicast address, and thus form a multicast tree for that address. Packets sent to the multicast address of the MH flow down the multicast distribution tree o reach the MH. In [6], a MH sends source-specific join messages towards every CH with which it wishes to communicate. As the MH moves and connects to another location, the multicast router at that location joins the group address, and so a multicast tree, rooted at the CH gets formed and is used for routing packets between the CH and the MH. In [7], a multicast architecture is used to accelerate Mobile IP, by avoiding the need to inform the HA of every change in location. MHs are assigned prearranged multicast group addresses at the HA. Neighboring base stations in the vicinity of the MH adhere to this group address, and a multicast tree gets formed, rooted at the HA. Packets are then tunneled from the HA to the MH, along this multicast tree.

The most significant drawback of multicast-based solutions is that they require routers to be multicast capable; this capability either does not exist, or it is not turned on in the Internet routers of today. In essence, this solution requires that routers learn multicast addresses, in the same way that routers learn unicast addresses in Hawaii. Unlike LAN switches, routers are not designed to learn host addresses, and therefore they would need to be modified for this purpose. Other drawbacks of mobility schemes based on multicast routing are that they require unique multicast addresses to be used, which creates address management complexity and limits the addressing space.

2.5. Micro-mobility and LAN switching

In all the solutions for micro-mobility that we presented, fixed IP addresses are used to track mobile users inside some region. This is done via learning at wireless BSs, routers or agents. Despite the use of IP addresses, which are hierarchical, routing to mobiles inside a region is performed in a flat-address space, just like in a LAN. Consequently, MH addresses are tracked in a similar fashion as layer-2 addresses in LANs. We further make the observation that, in fact, these addresses are tracked in the same way as virtual channel identifiers in virtual circuit-switched solutions such as ATM (e.g. employed in UMTS for the tracking of users by foreign gateways). In their original design, routers were not meant to perform tracking of individual host addresses, and consequently do not perform host-specific routing in an efficient way. It is unlikely that, in the near future, routers in the Internet will be replaced by routers that perform this function. It is also unlikely that an infrastructure consisting of such modified routers would be deployed to support mobility inside a micro-mobility region. However, it is feasible to imagine that an overlay infrastructure consisting layer-2 switches, which, by design, can track host addresses efficiently, could be deployed in a cost-effective manner to achieve this goal.

3. A network infrastructure for mobility support using layer-2 technology

Over the past decade, we have witnessed tremendous developments in LAN technologies, such as increases in switch processing by a few orders of magnitude, and increases in link bandwidth and distances (owing to the fiber optics technology). These advances resulted in an increase in the size of LANs, and more recently, the deployment of such technologies in metropolitan areas. Mobile users can roam inside a LAN without having to update their addresses. The reason why this is possible is because LAN switches learn the users location and can route packets to them quickly using this information. Furthermore, layer-2 technology is cost-effective by virtue of its simple protocols and large-scale deployment. For these reason, layer-2 technology is at the foundation of our network design for mobility.

In addition to layer-2, functionality at layer-3 and possibly the directory (DNS) layer is needed to support mobility at a global level across the Internet. As described earlier, Mobile IP may be used to tunnel traffic appropriately when users move between subnets. Furthermore, dynamic DNS may be used to improve performance by avoiding triangle routing through the HA, as mentioned in the Introduction. A description of such a solution is depicted in Fig. 4.

In the rest of the paper, we focus on the design and analysis of one possible implementation of our

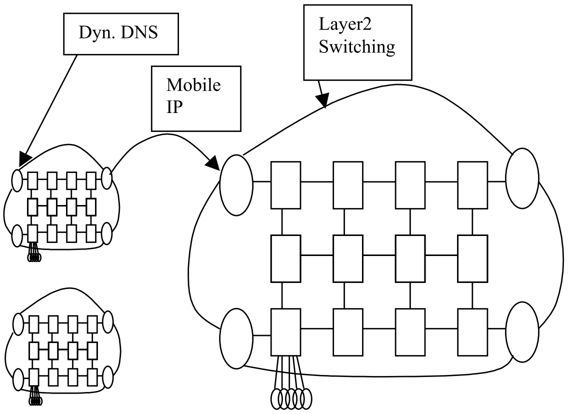

Fig. 4. A multi-layer architecture for IP mobility.

layer-2 for mobility support in metropolitan areas. This network has a number of design goals, namely: fast tracking of users, for the purpose of supporting mobility and applications with time constraints; a simple, low cost design for easy deployment; incremental scalability from small cities to large cities with many users (tens of millions), each consuming high data bandwidth (tens of Mb/s); finally, support for Internet connectivity, at high speeds (microseconds) and high bandwidth (tens of Tb/s).

To achieve the fast tracking of users, it is necessary that a fixed address be maintained for each user while it roams inside the network, to avoid the penalty of acquiring and releasing many different addresses. From this perspective, a mechanism for address tracking is necessary, similar to that performed by the switches in a LAN. However, unlike in LANs, the tracking of users must happen at a faster pace, to accommodate for user mobility at many speeds. In LANs, switches learn about user location via the snooping of data packets originated at that user. Instead, what needs to take place is the explicit registration of users with the switches in the network. This network exhibits functionality that is not typically present in LANs, thus we refer to it as *MobiLANe* (i.e. LAN for mobile users).

The number of switches involved in tracking each user has to be carefully considered so as to ensure the fast completion of location updates for each such user. To achieve this goal, the MobiLANe could be decomposed into control partitions that include only a subset of the switches in the MobiLANe. Each user is tracked by the control partition that includes the switch local to the user. Note that each switch tracks only a subset of the users (according to the placement of users at a given time). Thus, another benefit of this approach is that the tracking databases at each switch can be made smaller and thus the switches can be faster and cheaper. A protocol similar to the generic attribute registration protocol (GARP) [14] is designed to track users inside a given control partition according to the user location. Since the location information for a user is not flooded across the entire MobiLANe, an algorithm must be designed for routing user packets that properly exploits the information at the control partition for that user. This can be achieved by employing the multicast feature of LAN switches, which allows us to selectively multicast data packets towards the control partition for that user, from where the packet can be delivered directly to the user. To this end, the protocol could also decompose the MobiLANe into data partitions. Each control partition must have one or more switches in common with every data partition. Similarly, each data partition must have one or more switches in common with every control partition. Data packets for a given user are propagated along a given data partition (as given by the location where the data packet was first injected into the MobiLANe) until the control partition for that user is reached and the packet delivered to the user. More details are given in the following subsections.

In this paper, we choose to interconnect a large number of switches via the grid topology (e.g. the Manhattan Street Network) as shown in Fig. 5. For this topology, each row in the grid could be a control partition and each column, a data partition, as depicted in Fig. 6. The choice of the grid topology is partly driven by the fact that the grid matches the topology of cities themselves—with the streets being rows and column. In addition, the grid is scalable by virtue of its distributed nature, which does not result in the aggregation of traffic at some central location such as a root. Furthermore, because of its uniformity, the grid lends

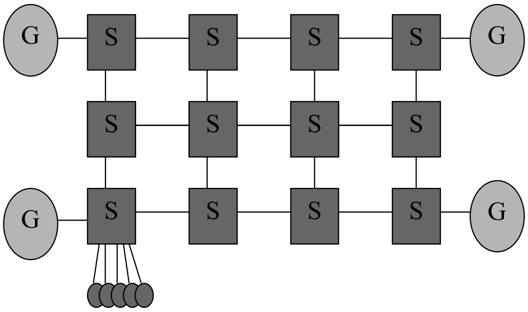

Fig. 5. A MobiLANe implementation using switches connected by a grid topology.

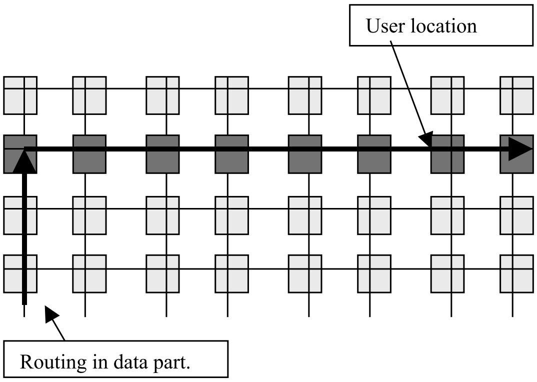

Fig. 6. A user and its control partition along the horizontal arrow.

itself to a simple, low cost deployment, that allows the same switches and link interconnects to be deployed at all locations across the area. This eliminates the need to study and understand the traffic in the area, in order to properly scale each individual switch and link bandwidth in the network. However, due to their large number, wireless cells are connected to MobiLANe switches in a hierarchical fashion. This hierarchy is necessary to reduce the number of hops to be traversed when communicating between two access points in the grid, and therefore reduce latency.

The MobiLANe topology may not be a perfect grid for various reasons: the MobiLANe can not be deployed as a regular structure due to terrain or administrative restrictions; some switches or links may fail, resulting in rows and columns of varying length. To cope with this situation, we developed the following algorithm to render a grid structure over an irregular topology: we define a virtual row/column to be a set of switches and all the edges between them, such that for any pair of switches in that row/column $\langle a, b \rangle$, the path between a and b is entirely contained in that row/column. The following properties are true:

1. The MobiLANe is represented by the sum of all virtual rows and by the sum of all the virtual columns.
2. Each virtual row/column must intersect all virtual columns/rows.

We partition MobiLANe into virtual rows and columns to create the desired grid structure. Let us take the example of a missing switch, or a "hole". Intuitively, work around "holes" in the topology by including the switches that neighbor the hole into virtual rows/columns as follows: for a hole at location $\langle I, J \rangle$, where I is the row number and J the column number:

1. Add switches $\langle I-1, J-1 \rangle$, $\langle I, J-1 \rangle$, $\langle I+1, J-1 \rangle$ to virtual column J (assuming these switches are not holes).
2. Add switches $\langle I-1, J-1 \rangle$, $\langle I-1, J \rangle$, $\langle I-1, J+1 \rangle$ to virtual row I (assuming these switches are not holes).

To connect to the Internet backbone, a scalable and distributed gateway router is necessary to support the aggregate traffic between the Internet and all the cells in the MobiLANe. Indeed, if the numbers of cells and users are large, this bandwidth can become very large. For example, for a LAN supporting two million users, consuming 2 Mb/s each, the routing bandwidth is 4 Tb/s. Furthermore, the router must be physically distributed across many smaller routers to allow for load balancing at the links connecting the LAN switches to the subnet router. To address this, in this implementation we superimpose a number of IP subnets onto MobiLANe (at the logical level), each controlled by one of the IP gateways, which are placed at different locations in the area, as shown in Fig. 5.

When a user enters the MobiLANe, it is given an IP address by the gateway controlling one of the subnets (this gateway is, say closest to where the user is located). Evidently, the user IP address does not change while the user is still in the MobiLANe. It is the responsibility of the switches inside the MobiLANe to appropriately route packets to and from the user regardless of the location of that user. Switches track a user by learning either the MAC or the IP address of that user. The advantage of MAC addresses is their use in LAN switching today. One disadvantage of using MAC addresses is that not all devices may come with a built-in MAC address. Another disadvantage is that some form of address resolution like ARP needs to be performed at the MobiLANe, to map the IP address of the user to its MAC address. If the MobiLANe is large, this operation may take a long time. To cope with the later challenge, a database is used at each gateway to map the IP addresses of users to their MAC addresses. This database holds an entry for every user belonging to the IP subnet for which that gateway is responsible. The database of a gateway is updated at the same time that the user enters the MobiLANe and receives an IP address from that gateway.

3.1. Registration protocol

As mentioned earlier, the propagation of tracking information in the partition is by means of a dynamic registration protocol similar to the GARP [8]. The protocol, depicted in Fig. 7, consists of a number of control messages:

1. The JOIN message informs switches of the arrival of one or more users.
2. The LEAVE message informs switches of the user(s)'s departure from the switch's jurisdiction.
3. An ACK message confirms that the registration updates have been received.

A switch of a partition is a local switch if it is directly connected to the access point where a user resides. The other switches in the partition are known as remote switches. A local switch propagates control information to one or more remote switches in the partition. If rings are used to connect the switches along rows and columns, then the control messages at the local switch are propagated as follows:

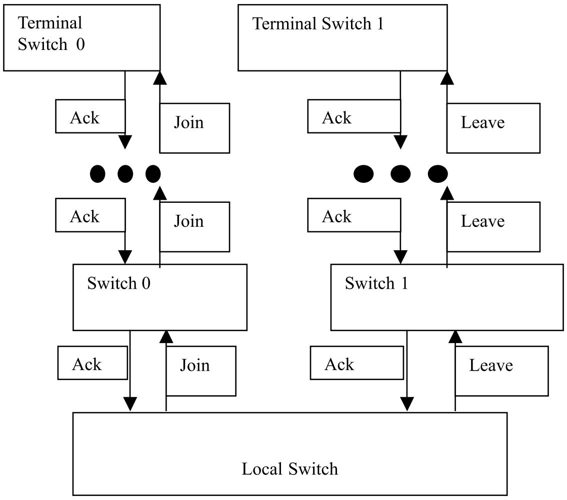

Fig. 7. Registration protocol.

1. either left or right when the control partition is a row or a virtual row,
2. either up or down when the control partition is a column or a virtual column,
3. either left and down, left and up, right and down or right and up when a control partition is a pair of rows (or virtual rows) and columns (or virtual columns).

When the MobiLANe is configured, switches are programmed to propagate control messages in a fixed direction selected form the choices outlined above.

When a control message arrives at a remote switch, it is used to configure the routing tables appropriately. It is also remembered and sent out on the port directly opposite to the port where the control message was received (e.g., if the control message is received on the left port, it is propagated on the right port). When a control message arrives to a switch that has already seen it, it is consumed and removed from the ring.

If a ring topology is not used to connect switches in a row or column, the control message at the local switch is multicast in two directions when the control partition is either a row (virtual row) or column (virtual column) (e.g. up and down that column), and four directions when pairs of rows (virtual rows) and columns (virtual columns) are used (up, down, left and right). Note that if a given control message did not cause the routing tables for a given port to be updated, then the message is not propagated along that port. One such case occurs when the control partition is a row and the user moves from one switch to the next switch in that row. The new local switch observes that the routing tables for the horizontal ports need not be updated, hence a control message is not sent to the remote switches.

Every time a mobile crosses cell boundaries, a JOIN message is sent to the local switch by the wireless cell to which the user is connected. The JOIN is propagated across the new control partition. An ACK reply is propagated in the reverse direction across the same control partition. Upon reception of an ACK message for the JOIN at the local switch, a LEAVE message is sent to the old switch port, which then gets propagated across the old control partition (we assume that communication between the new and old local switches is possible). An ACK reply is then propagated across the old control partition and sent to the new local switch, which completes the registration process.

Acknowledgements are used to ensure that user registration and de-registration with the switches of a partition are atomic. When moving from one partition to another, a user leaves the old partition only after it has joined the new partition and received an acknowledgement from it. Notice that simply receiving a packet from the switches in the new partition is not enough to guarantee atomicity because some switches may not have been properly set. This is because the packet may come from another user within the partition without traversing all the switches in that partition. However note that acknowledgements can be piggybacked to the user packets that may be traversing the partition, in order to reduce latency and bandwidth consumption. This technique assumes that cell overlap is possible and its extent long enough to allow for the new JOIN to complete, and the ACK to be received.

A terminal switch is a switch that determines not to propagate a registration message received at an input port to any output port. Each switch that is not a terminal switch waits for acknowledgements from other switches before submitting an acknowledgement. Terminal switches update their databases and then send acknowledgements without further wait. When single rows or columns are used for control partitions, the local switch waits for a single acknowledgement before proceeding with the LEAVE. When pairs of rows and columns are used as control partitions, then the local switch waits for two acknowledgements, one from the row switches, and one from the column switches. Note that if a switch moves from one partition to another partition that have a common row or column, then a single acknowledgement is needed, corresponding to the column or row that is different.

3.2. Routing

In our routing protocol, we capture the mechanism by which routing at the switches is to be performed along data and control partitions. When a switch receives a data packet for some user, the following possibilities arise:

1. If the user is known to the switch (i.e. the switch belongs to the control partition for that user), then the packet is routed according to the information in the table.
2. If the user is not known to the switch (i.e. the switch does not belong to the control partition for that user but belongs to the data partition for that data packet), then the data packet is routed according to some routing algorithm selected when the MobiLANe was originally configured.

We identified three such routing algorithms:

(1) *Vertical*. If the switch is the first switch in the data partition to see the packet, and the ring topology was not used to connect the switches of a row or a column, then the packet is routed BOTH

up and down. If the switch is the first switch in the data partition to see the packet, and the ring topology was used to connect rows and columns, then the data packet is routed either up or down, depending on the switch configuration. This routing scheme is called vertical-first routing.

(2) *Horizontal*. If the switch is the first switch in the data partition to see the packet, and the ring topology was not used to connect the switches of a row or a column, then the packet is routed BOTH left and right. If the switch is the first switch in the data partition to see the packet, and the ring topology was used to connect the switches of a row or a column, then the packet is routed either left or right, depending on the switch configuration. This routing scheme is called horizontal-first routing.

(3) *Sometimes horizontal and sometimes vertical*. Here we combine the two algorithms above. Some packets are routed using vertical-first routing while others are routed using horizontal-first routing.

We describe each algorithm in more detail in the following subsections.

3.2.1. Vertical-first routing

Vertical-first routing refers to a scheme where a switch in the grid routes vertically a packet that it receives for a user unknown to it. Eventually, the packet arrives at a switch that knows the user (at the intersection of the control partition for that user and the data partition for that data packet), from where the packet is then routed home.

For this routing scheme to work, the tracking protocol requires that the control partition for every user include the row in which that user is located. Vertical-first routing with row control partitions is depicted in Fig. 6. At first, the packet is routed up the column, where it eventually reaches a switch that “knows” the user. If the column is a ring, the packet is only routed in one direction (either up or down). Otherwise, the packet is multicast up AND down. The first switch that knows the user routes the packet horizontally. If the row is a ring, the packet is only routed in one direction (either left or right). Otherwise, the packet is multicast left AND right. From there on, the packet is routed horizontally until the switch to which the user is attached is found and the packet gets delivered.

3.2.2. Horizontal-first routing

This works identically to vertical-first routing, except that the direction taken by a packet at a switch is horizontal instead of vertical when that switch does not recognize the user for which the packet is intended.

3.2.3. Alternating vertical-first and horizontal-first routing

Intuitively, one can see how vertical-first routing or horizontal-first routing can lead to load unbalance if the users or the input/output vertices are concentrated along a particular row or column. For example, if the users are concentrated along some column and horizontal-first routing is used, then traffic is routed along the vertical axes by using primarily that column. Also, if the input to the grid is concentrated along one column and vertical-first routing is used, then traffic is routed vertically by using primarily that column. To avoid this situation, the routers of the MobiLANe must be distributed in a balanced fashion across the MobiLANe. However, depending on the region, it may not be possible to control the distribution of routers so to ensure uniform coverage of all rows and columns. Under these circumstances, a routing algorithm that distributes the load fairly across the MobiLANe becomes essential. This is particularly important when the link bandwidth is scarce (e.g., microwave radio links or low-cost MobiLANe infrastructure).

Consequently, in order to balance the load, we designed a scheme that alternates vertical-first routing with horizontal-first routing. The scheme works as follows: instead of always routing horizontally or vertically when the destination user is unknown, switches route vertically sometimes and horizontally other times. By alternating between vertical-first routing and horizontal-first routing, we ensure that, in the long run, the load is divided fairly across all rows and all columns, regardless of the user and router distribution. In order to guarantee that all packets reach their intended destinations, we must ensure that packets are routed in the same fashion (vertically or horizontally) at all the switches that do not recognize the user for whom a given packet is intended. This is achieved by having the first switch that routes the packet select the routing direction. This switch marks the packet to communicate to the other switches that a routing direction had been selected for that packet. The other switches observe that the packet is marked and are required to follow suit and route in a similar fashion. They route the packet to the port directly opposed to the incoming port (e.g. the left port if the packet was received on the right port). The first switch selects the routing direction by alternating between the vertical and horizontal direction in a round-robin or random fashion.

3.2.4. Clustering

It is expected that users visit cells in the neighborhood of their location more frequently than remote cells. In our schemes, when rows are used as control partitions, and the user moves to the next row, registration with the new row and deregistration with the old row need to take place. When pairs of rows and columns are used as control partitions and the user moves either to the next row or the next column, registration with the new row or new column need to take place. If the user moves frequently, then these updates can be expensive. One solution is to perform user tracking in a hierarchical way that exploits the spatial locality of user movement. Instead of a local switch, we use a cluster of switches, called the local cluster. As before, we use rows (or columns, or both) to track and locate users. However, while before the row (or column, or both) was used to obtain the location of the user exactly, now it is used to obtain the location of the user cluster to which the user belongs. To find the actual the location of the user, we use the tracking at the switches inside the local cluster. Tracking inside a local cluster is performed as follows:

(1) If rows are used as control partitions to find the local cluster, then columns (or segments of columns) must be used as control partitions to find the user inside that cluster. By using columns as control partitions inside the local cluster, we ensure that the registration information can be propagated between the local cluster and the row that tracks that cluster.

(2) If columns are used as control partitions to find the local cluster, then rows are used as control partitions to find the user inside that cluster.

(3) If pairs of rows and columns are used to find the local cluster, then pairs of rows and columns are used to find the user inside that cluster.

When a user moves inside a local cluster, tracking updates need to take place only at switches in that cluster. When a user leaves a local cluster and joins another, we update the tracking databases in the rows and columns of the grid that track the new local cluster, as well as inside the new local cluster. Depending on the size of the cluster, updating tracking databases inside the local cluster can be significantly less expensive than updating an entire row or column of the MobiLANe. Clustering can significantly reduce the cost of completing a location update.

Routing is performed in two steps: first, the local cluster is found, via one of the routing algorithms we described before. Second, the actual user location is found, via routing at the switches inside the cluster. Fig. 8 shows the switches clustered in different clusters corresponding to different quadrants of the grid. The lines corresponding to the column and row used for tracking the cluster in which the user resides are shown by the dotted lines. The switch where these lines intersect is known as the head (or pivot) of the cluster. The pivot is dynamic for each user, and is given by the

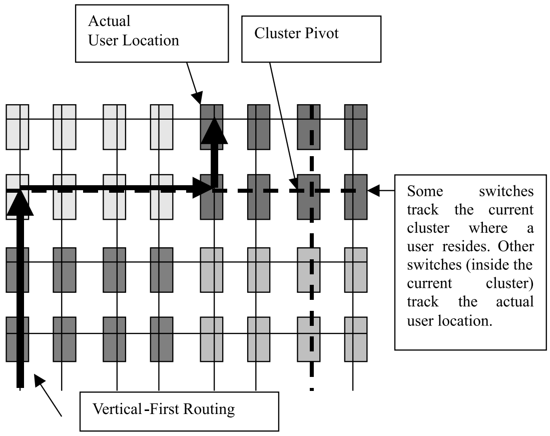

Fig. 8. MobiLANe clustering.

switch via which the user had joined the cluster. By making the pivot dynamic, we allow the traffic load to be more uniformly distributed across the rows and columns of MobiLANe.

3.3. Contributions of MobiLANe design

The MobiLANe design has a number of advantages. The MobiLANe does not rely on a single spanning tree or root switch. This is important for scalability as the MobiLANe extends to large geographical scopes. By exploiting control and data partitions, it minimizes the latency of user location updates without affecting the latency of packet routing inside the MobiLANe. Finally, its operation relies significantly on existing switching techniques and protocols, which makes the solution simple, inexpensive and easy to implement and deploy.

4. Simulation environment

We use a stochastic mobility model from the University of Waterloo [9], which simulates daily movements of mobile subscribers, incorporating realistic activity patterns. The input data to the model includes an activity transition matrix and an activity duration matrix, derived from the trip survey [18] conducted by the Regional Municipality of Waterloo in 1987. For this survey, a travel diary was completed by each household member over five years of age, in which details were recorded on all trips taken during the day of the survey. Included for each recorded trip were the trip start and end times, the trip purpose at the origin and destination, and employment or student status.

There are nine types of activities defined in the model, which are as follows: work, school, shopping, personal business, work-related, serve passenger, social/recreation, return home and other. Each activity has an associated time of day, duration and location (at the level of a radio cell). In a given simulation run, activities are selected based on the previous activity and the current time period. The probability of transition from one activity to another uses the activity transition matrix. Once the next activity is selected, its duration is determined using the activity duration matrix. Finally, the location of the activity is selected, based on certain heuristics and the activity type (e.g., if the activity purpose is "shopping", a cell is randomly selected from a set of cells neighboring the subscriber, within a radius as given by the retail characteristics of the region). Once the location of the next activity is selected, the intermediate route (in terms of cells crossed) and the total distance are determined from a geographical lookup table. Using a user-defined system-wide average speed, the time spent in each intermediate cell is calculated. The subscriber stays in the destination cell for the duration of the activity, and the sequence is repeated. The output from the model is thus a trace of the daily movement of individual cellular subscribers over a period of several days, in terms of cells crossed and time spent in each cell.

The geography of the region simulated, which is shown in Fig. 9 (approximated from the geographical lookup table), covers 312 km^2 and is divided into 45 cells, each covering an average of 7 km^2. For each subscriber, a home cell, a work cell and a school cell are selected randomly from among the cells in the region. Starting at home, users take daily trips such as going to work and back, going shopping and going to the movies. We simulated 35 days during which subscribers were tracked on a per minute basis. The simulation is event-driven and has a granularity of one minute. An event that occurs between two minutes is considered to have occurred at the earlier minute.

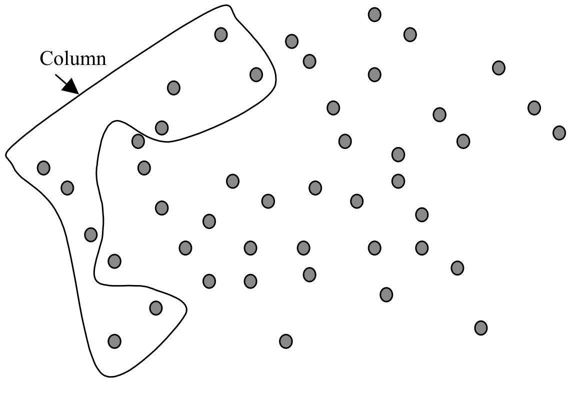

Fig. 9. Cell division for the region simulated.

Since they happen during the same minute, the JOIN, LEAVE and ACK corresponding to a cell crossing occur simultaneously in our model.

In our simulations, two wireless cell types are employed: macro-cells and micro-cells. A macro-cell maps directly to a cell as defined in the mobility model discussed above. The characteristics of the wireless macro-cells are summarized in Table 1.

An architecture where cells are further subdivided into micro-cells is also discussed (Table 2). Macro-cells are subdivided into micro-cells to increase user bandwidth. Mainly, one macro-cell is subdivided into 700 smaller cells, allowing short-range wireless protocols (like 802.11b) to be used inside a micro-cell. This results in approximately 7.5 Gb/s bandwidth to be available inside a cell, i.e. an increase by 770,000× in bandwidth per cell. Higher user bandwidth is crucial for delivering multimedia to mobiles. A single H.263 video stream consumes 10 Kb/s, so this would allow for approximately 770,000 users to receive different video streams simultaneously inside a cell. For this configuration, multiple switches are used to balance the load across the region and to reduce the number of ports needed for connecting the micro-cells at a switch. Since the total load across all the micro-cells in the region is about 340 Gb/s, we chose to spread this load across 45 switches, each responsible for 7.5 Gb/s sent to/from users in the 700 micro-cells of a cell. The switches are grouped in a grid consisting of nine rows and five columns (9 × 5).

Table 1
Macro-cell characteristics

Range	7 km^2
Bandwidth	100 kb/s
Total number of cells	45
Example of protocol	Cellular telephony

Table 2
Micro-cell characteristics

Range	0.01 km^2
Bandwidth	11 Mb/s
Total number of micro-cells	31,500
Example of protocol	802.11b

To compare the performance of our system against the performance of other schemes, we consider three scenarios as follows:

(1) The Mobile IP scenario: nine foreign agents populate the region, each responsible for one-ninth of space (as given by a row in the grid). For experimental purposes, the HA is considered to be outside the region, with each FA being directly connected to the HA (at equal distances). Location updates are sent to the HA. To reach its destination user, traffic arrives at the HA from where it gets forwarded to one of the foreign agents inside the region. This is shown in Figs. 10 and 11 by the interrupted line between the CH, HA and the region. Notice that Mobile IP allows the CH to receive binding updates with the latest COA of the user. However, in our simulations, each call is generated by a different CH, hence the information in the binding cache is irrelevant.

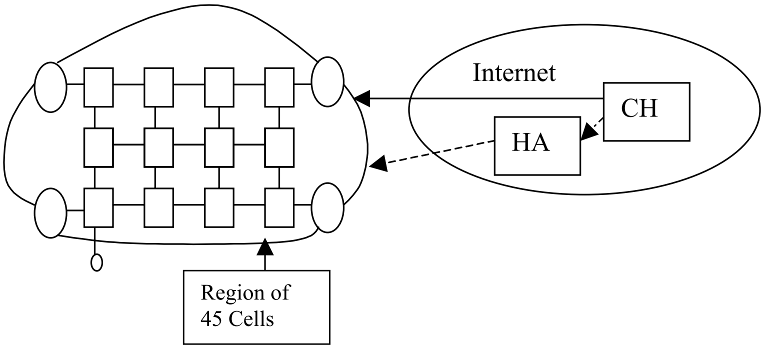

Fig. 10. Simulation scenario with wireless macro-cells.

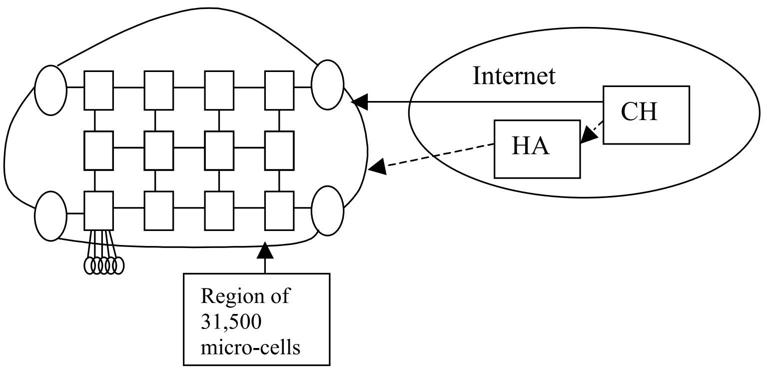

Fig. 11. Simulation scenario with wireless micro-cells.

(2) A typical micro-mobility algorithm (such as HFA or Cellular IP), labeled Micro IP. 45 routing devices (such as FAs in HFA or switches in Cellular IP) populate the region, and are connected via a hierarchical interconnect as shown in Fig. 12. A gateway connects the root of the tree to the Internet. Location updates are handled locally by the hierarchy and do not need to reach the HA. However, all traffic from the CH traverses the HA prior to arrival at the region.

(3) MobiLANe: The MobiLANe is populated by 45 switches interconnected via a regular, 9 × 5 grid topology. Each switch controls one macro-cell or 700 micro-cells in a cell area. To superimpose the grid structure on the region, the switches are grouped into irregular columns and rows (an example of a column is shown in Fig. 9). Figs. 10 and 11 depict the MobiLANe implementation for each wireless cell type we considered. In this paper, vertical-first routing is used, and no clustering. For the users in the MobiLANe, DNS is updated with the COA of the gateway responsible for the region. Traffic from the CH arrives directly at the gateway from where it is forwarded to the switches. This is shown in the figure by the uninterrupted line between the CH and the region. Location updates are handled by the switch local to the moving user and do not need to be forwarded to the HA.

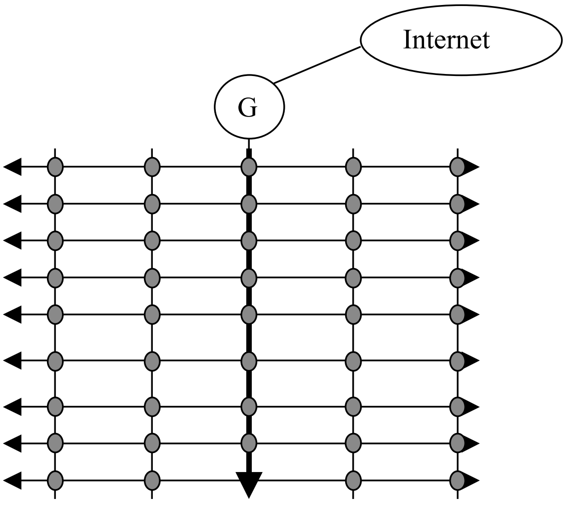

Fig. 12. The tree hierarchy in Micro IP. The gateway (G) is connected to the root of the tree, and to the Internet. The region is populated by 45 routing devices.

A stochastic model for user voice traffic is assumed in order to test the impact of the mobility architecture on user traffic performance. Traffic is originated from a CH, located outside the area, and is addressed to a user inside the area. Users receive phone calls at random times throughout the day. The number of phone calls per day and the duration of each phone call are entered statically as parameters to the simulation.

We identified a number of important metrics for evaluating our architecture: the peak per minute control load, including the total number of JOIN's, LEAVE's and ACK's at a switch BW_CTL; the peak per minute data load at a switch BW_DTA; the time to complete a location update, in terms of the number of switches traversed LAT_CTL. Note that, to complete an update, multiple switches may be traversed in parallel (e.g., across the row and column of a control partition). In this case, we consider update latency to be the longest of the parallel tracks traversed to complete the update. In the case of LAT_CTL, we consider the time to setup paths to users entering the region (Init), as well as the peak (Peak) and average (Avg) per minute values across all users. We also consider the time for a packet to travel from source to destination, which we call routing latency or LAT_DTA. We measure routing latency in terms of the average number of switches traversed per packet across all users.

To compute BW_CTL, we count the number of control messages per minute, multiply it by the length of the control message and divide it by 60. The length of the control message is assumed to be 16 B. We think 16 B is a good estimation since it includes 12 B for the source and destination addresses, 2 B for the checksum and 2 B for various attributes and options.

To compute LAT_CTL, we measure the number of cell crossings that occur during a simulation, and multiply this number by the number of switches involved in a location update for each algorithm. The duration of the location update depends on the scenario as follows: for the Mobile IP scenario, it represents the time to update the database at the local FA, to transfer the update message between the FA and the HA and the time to update the database at the HA; for the Micro IP scenario, it represents the time to update the appropriate routing devices in the tree; finally, for MobiLANe, a location update includes the time to update the appropriate switches in the MobiLANe.

To compute LAT_DTA, we measure the number of switches involved in transferring each packet across the MobiLANe and multiply it by the time to transfer the packet at the switch. We add the time to transfer the packet between the CH and the gateway in charge of the mobile. For the Mobile IP and Micro IP scenarios, we add to this number the time to transfer the packet between the HA and the region. We show the sensitivity of LAT_DTA to transfer time between the HA and the region. We assume that the distances between the CH and the HA, between the HA and the region and between the CH and the region are the same.

5. Simulation results

5.1. Control load

In this section, we investigate the scaling of control load with the size of population in the region, in the case of MobiLANe with macro-cells and MobiLANe with micro-cells. It is important that the control load amount to a small number (relative to the capacity of the switches of today) for MobiLANe to be a feasible solution, for a variety of population sizes.

Fig. 13 shows BW_CTL for MobiLANe with macro-cells and MobiLANe with micro-cells, and

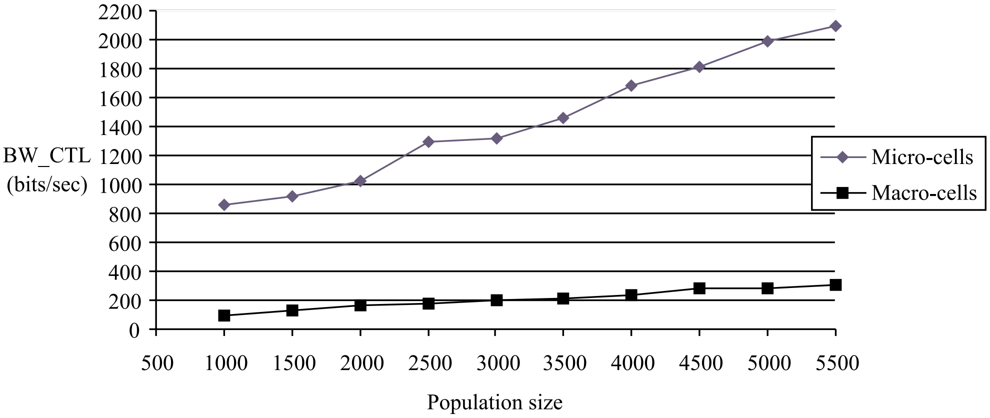

Fig. 13. BW_CTL for different population sizes for the cases of MobiLANe with macro-cells and MobiLANe with micro-cells.

for various population sizes between 1000 and 5500 people. Prohibitively long simulation times limited the population size studied to this range.

The dependency of BW_CTL on population size is sub-linear. When scaling from 1000 to 5500 people, we noticed sub-linear effects: less than 3.3× and 2.5× increase for MobiLANe with macro-cells and MobiLANe with micro-cells, respectively. This is explained by the fact that, for this population range, an increase in population size contributes primarily towards the spreading of traffic across more cells and more time slots and only secondarily towards increasing peak traffic in a given time slot and a given cell. More substantial increases in BW_CTL are expected for high population ranges due to the increased user concentration in a given time slot.

As expected, cell subdivision results in a significant increase in BW_CTL because of the increase in the rate of cell crossings. The subdivision of each macro-cell into 700 micro-cells results in about 7× increase in BW_CTL. Assuming linear increase for high population densities, a population of 500,000 people (which is large for a region of 312 km^2), would generate about 200 Kb/s in control traffic in the case of MobiLANe with micro-cells. Assuming a registration latency of 1 μs, (or the equivalent of ~10 SDRAM lookups), a single switch can support 1,000,000 registrations per second, or an equivalent of 122 Mb/s. Since the maximum value achieved by BW_CTL is ~200 Kb/s, there is more than enough bandwidth to support the registration bandwidth generated by this region fully populated (at 500,000 people).

In this section we demonstrated the feasibility of MobiLANe with macro-cells and MobiLANe with micro-cells, with regard to supporting registration traffic for tracking user movement. In the rest of the paper, we present results only for MobiLANe with micro-cells, because it represents a more interesting experimental case, and it delivers higher data bandwidth to users.

5.2. Control latency

Fig. 14 shows the behavior of the average LAT_CTL as a function of distance to the HA for Mobile IP. LAT_CTL for MobiLANe with one gateway is plotted for reference. A 1 ms processing delay at the FA and every switch in the network is assumed. Processing time at the HA in considered part of the distance to the HA. As we can see from the graph, Mobile IP does generally worse than MobiLANe because of the overhead of communicating with the HA upon every location change. However, if the HA is nearby to where the user is located, HA updates may take less time than local updates in MobiLANe. This is because, for MobiLANe, a larger number of electronic switches need to be involved in one location update than for Mobile IP.

Fig. 15 shows Init, Peak and Avg. LAT_CTL for MobiLANe, Micro IP and Mobile IP. The distance to HA is arbitrarily set to 20 ms. We observe that MobiLANe performs better than both Mobile IP and Micro IP in all three cases: the time to setup connectivity to subscribers (Init), and the peak and average LAT_CTL. The reason

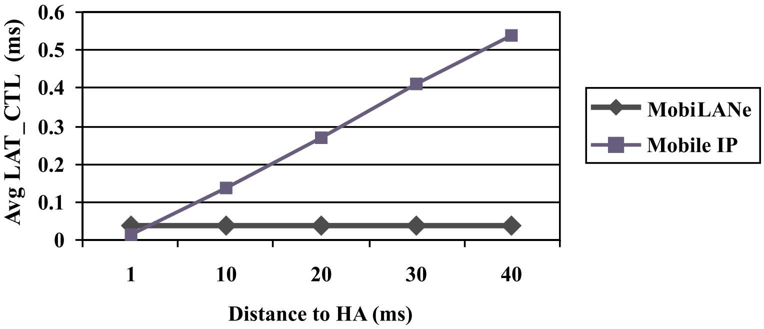

Fig. 14. Avg. LAT_CTL for Mobile IP, for different distances between the local FA and the HA.

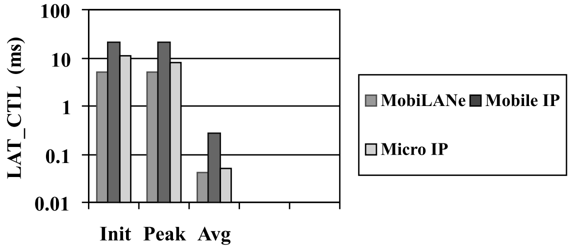

Fig. 15. Init, Peak and Avg. LAT_CTL for MobiLANe with micro-cells and one gateway, Micro IP and Mobile IP.

MobiLANe performs better than Mobile IP is because no HA overhead is incurred. The reason MobiLANe performs better than Micro IP is because fewer switches need to be informed of the location of users, due to the selective multicast nature of the routing algorithm of the MobiLANe used in MobiLANe.

5.3. Data load

Fig. 16 plots BW_DTA for different numbers of gateways, different per-user application bandwidth (8 Kb/s, 128 Kb/s and 1 Mb/s) and for three different schemes: Mobile IP, Micro IP and MobiLANe (with five different numbers of gateways). We notice that the best results are achieved for Mobile IP. This is because, in this implementation of Mobile IP, FAs only track and route to/from users in their portion of the region and do not transfer traffic sent for another foreign agent in that region. When multiple gateways are used, BW_DTA decreases considerably from the case of a single gateway. When four gateways are used, less than half as much peak traffic is born by the switches than when one gateway is used. This means that, when multiple gateways are used, smaller, cheaper switches can be deployed in the MobiLANe.

5.4. Data latency

5.4.1. Impact of dynamic DNS

Fig. 17 displays LAT_DTA for various distances between the MH and the HA and the CH and for three different scenarios: MobiLANe with one gateway, Micro IP and Mobile IP. LAT_DTA increases monotonically with distance; a slower rate of growth is observed for MobiLANe (owing to the use of dynamic DNS which circumvents the need for tunneling through the HA). For a distance value of 40 ms, LAT_DTA is almost twice smaller if dynamic DNS is used. Notice that for

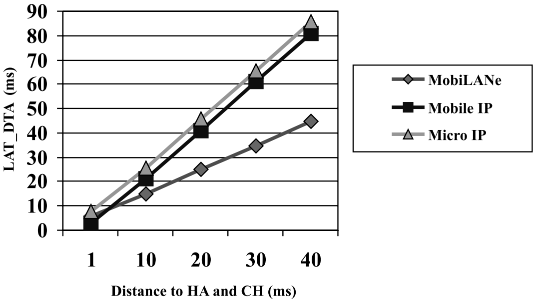

Fig. 17. LAT_DTA for MobiLANe with micro-cells and one gateway, Mobile IP, Micro IP.

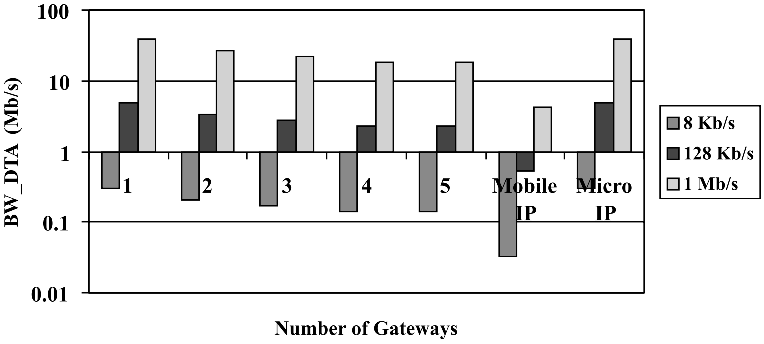

Fig. 16. BW_DTA for MobiLANe with micro-cells for different number of gateways.

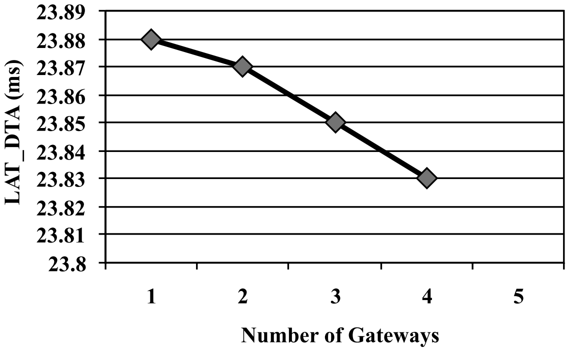

Fig. 18. LAT_DTA for MobiLANe with micro-cells for different numbers of gateways.

small distance values (a few ms), LAT_DTA is actually larger in the case of MobiLANe and Micro IP than for Mobile IP. This is because of the inherent overhead of traversing the switches populating the region. In the case of Mobile IP, fewer electronic components are traversed as packets are routed between the CH, HA and the MH.

5.4.2. Impact of multiple gateways

Fig. 18 shows the variation in LAT_DTA for MobiLANe in terms of the number of gateways. We notice that LAT_DTA decreases slightly as more gateways are used. This is for two reasons: when the MobiLANe is populated with more gateways, these gateways are placed closer to subscribers than when fewer gateways are used. Secondly, subscriber movement exhibits a small degree of locality; on average, subscribers are closer to their home gateways than to any other gateway in the MobiLANe. A distance of 20 ms between the MobiLANe and the CH, and a 1 ms processing delay at the gateway are assumed.

6. Contributions of MobiLANe

Our simulations demonstrate that MobiLANe is a scalable, feasible solution for supporting IP mobility over a large metropolitan area. By virtue of localized registration and distributed routing algorithms and the use of dynamic DNS, MobiLANe fairs better in performance than other existing schemes. Reliance on existing, low-cost switching technologies, as well as load balancing techniques and the use of multiple distributed gateways which have the effect of lowering the peak load and memory requirements at the switches, make it possible to implement MobiLANe with small and inexpensive technology.

7. Migration path

To transition to the mobile network of tomorrow, it may not be possible to design the supporting network infrastructure from scratch. Instead, support for mobility may need to be built on existing network structures, such as small subnets controlled by LAN switches and interconnected by IP routers with a small number of host-specific entries. Under such circumstances, one possibility is the use of virtual private networks (VPN) to offer extended LAN connectivity across multiple small subnets. In order to support mobile users in the most effective fashion, the protocol to handle mobile users needs to be flexible enough to operate at different layers in the protocol stack, and versatile enough not to require changes in the implementation of the LAN switches and IP routers of that network. In particular, the protocols running on LAN switches should be based on existing 802 protocols, since they are implemented in hardware and therefore cannot be easily replaced or reprogrammed. The main challenge becomes how to use and optimize existing protocols for the purpose of efficient support for mobility.

8. Conclusions

This paper surveys the state-of-the-art in providing mobility support to mobile users in the Internet. In particular, emphasis is placed on micro-mobility techniques designed to accelerate Mobile IP. One observation is that all micro-mobility solutions work in a similar way by requiring that network devices inside a given geographical area learn about the location of users and keep track of them as they move inside that area. The differences among these techniques are the type of device required to do the learning (it could be an IP router, Mobile IP agent or LAN switch) and the protocols for routing packets using the learning databases. This paper also proposes the deployment of an infrastructure based on simple and cost-effective layer-2 technology, to support user mobility in geographical areas of many scopes. One important feature of this infrastructure is its scalable and efficient MobiLANe design, geared at accelerating Mobile IP in large geographical regions. By relying on existing technologies, and by virtue of working with Mobile IP, this solution is global, cost-effective, easily deployable and compatible with the Internet of today.

References

[1] E. Perkins, K.Y. Wang, Optimized smooth handoffs in mobile IP, Proceedings of the IEEE Symposium on Computers and Communications, June 1999, Red Sea, Egypt.

[2] J. Mysore, V. Bharghavan, A new multicasting-based architecture for Internet host mobility, Mobicom 1997, Budapest, Hungary, September 1997.

[3] A. Snoeren, H. Balakrishnan, An end-to-end approach to host mobility, Mobicom 2000, Boston, MA, August 2000.

[4] C. Perkins, IP Mobility Support, RFC 2002, October 1996.

[5] S. Cheshire, M. Baker, Internet mobility 4 × 4, Proceedings of the ACM SIGCOMM 1996, Stanford, CA, August 1996.

[6] A. Helmy, A multicast-based protocol for IP mobility support, Second International Workshop on Networked Group Communication, Palo Alto, CA, November 2000.

[7] S. Seshan, H. Balakrishnan, R. Katz, Handoffs in cellular wireless networks: the daedalus implementation and experience, Wireless Networks (1995).

[8] J. Scourias, T. Kunz, An activity-based mobility model and location management simulation framework, MSWiM, Seattle, WA, 1999.

[9] A. Campbell, J. Gomez, S. Kim, A. Valko, C. Wan, Design, implementation and evaluation of cellular IP, IEEE Personal Communications (June/July, 2000).

[10] Naming, Addressing and Identification Issues for UMTS, UMTS Forum, December 2000.

[11] J. Schiller, Mobile Communications, Pearson Education Limited, 2000.

[12] R. Ramjee et al., HAWAII: a domain-based approach for supporting mobility in wide-area wireless networks, Proceedings of the IEEE International Conference on Network Protocols, 1999.

[13] Y. Cui et al., Efficient PCS call setup protocols, IEEE Infocom, San Francisco, CA, 1998.

[14] IEEE 802.1d MAC Layer Bridging Standard.

[15] http://www.metricom.com.

[16] D. Cheriton, M. Gritter, TRIAD: a scalable deployable NAT-based Internet architecture, http://www-DSG.Stanford.edu/triad/index.html.

[17] C. Hristea, F. Tobagi, IP routing and mobility, IWDC, Italy, 2001.

[18] Tranplan Associates, Waterloo Region Travel Survey 1987: An Overview of the Survey Findings, Regional Municipality of Waterloo, Department of Planning and Development, October 1989.

[19] C. Perkins, D. Johnson, Route Optimization in Mobile IP, Mobile IP Working Group, Internet Draft-work in progress, November 1997.

Cristina Hristea received the International Baccalaureat Diploma from Lester B. Pearson College of the Pacific, British Columbia, Canada in 1993, and the B.S. and M.Eng. degrees in Computer Science from Massachussets Institute of Technology, Massachussets, USA, in 1997. Currently she is working toward a Ph.D. degree in Electrical Engineering at Stanford University, California, USA.

From 1995 to 2000, she was with Silicon Graphics, where she worked on networking and multiprocessor computer systems. From 2000 to 2001, she was with Globespan where she worked on DSP systems. Her research interests include networking, computer systems, hardware and software design technologies.

Fouad A. Tobagi received the Engineering Degree from Ecole Centrale des Arts et Manufactures, Paris, France, in 1970 and the M.S. and Ph.D. degrees in Computer Science from the University of California, Los Angeles, in 1971 and 1974, respectively.

From 1974 to 1978, he was a Research Staff Project Manager with the ARPA project at the Computer Science Department, University of California, Los Angeles, and engaged in research in Packet Radio Networks, including protocol design, and analysis and measurements of packet radio networks. In June 1978, he joined the faculty of the School of Engineering at Stanford University, Stanford, California, where he is Professor of Electrical Engineering and Computer Science. In 1991, he co-founded Starlight Networks, Inc., a venture concerned with multimedia networking, and served as Chief Technical Officer until August 1998. His research interests include packet switching in ground radio and satellite networks, high speed local area networks, fast packet switching, broadband integrated services digital networks, asynchronous transfer mode, multimedia networking and communications, and modeling and performance evaluation of network systems.

Dr. Tobagi is a Fellow of the IEEE for his contributions in computer communications and local area networks. He is the winner of the 1981 Leonard G. Abraham Prize Paper Award in the field of Communications Systems for his paper "Multiaccess protocols in packet communications networks" and co-winner of the IEEE Communications Society 1984 Magazine

Prize Paper Award for the paper "Packet radio and satellite networks". He has served as Associate Editor for Computer Communications in the IEEE Transactions on Communications for the period 1984–1986, Editor for Packet Radio and Satellite Networks in the Journal of Telecommunications Networks for the period 1981–1985, Co-Editor of the special issue on local area networks of the IEEE Journal on Selected Areas in Communications (November 1983), Co-Editor of Advances in local area networks, a book in the series Frontiers in Communications (IEEE Press, New York), Co-Editor of the special issue on packet radio networks of the Proceedings of the IEEE (January 1987), and Co-Editor of the special issue on large scale ATM switching systems for B-ISDN of the IEEE Journal on Selected Areas in Communications (October 1991). He has also served as Co-Editor of Advances in local area networks, a book in the series Frontiers in Communications (IEEE Press, New York). He is currently serving as editor for a number of journals in high speed networks, wireless networks, multimedia and optical communications. He is a member of the Association for Computing Machinery and has served as an ACM National Lecturer for the period 1982–1983. He is a co-recipient of the 1998 Kuwait Prize in the field of Applied Sciences.