IP Multicast for Mobile Hosts

George Xylomenos and George C. Polyzos

{xgeorge,polyzos}@cs.ucsd.edu

Computer Systems Laboratory
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, CA 92093-0114

Abstract

We present alternative designs for efficiently supporting multicast for mobile hosts on the Internet. Methods for separately supporting multicasting and mobility along with their possible interactions are briefly described, and then various solutions to the combined problem are explored. We examine three different multicast delivery mechanisms and compare them based on their efficiency and impact on host protocol software.

Introduction

Communication modes can be characterized by the number of receivers targeted by a sender. Traditional modes have been one-to-one or *unicast*, and one-to-all or *broadcast*. Between these extremes we find *multicast*, the targeting of a single message or data stream to a select set of receivers. Multicast is a generalization of unicast and broadcast and a unifying communication mode. This model of communication naturally supports applications where data and control are distributed over multiple actors, such as updates to replicated databases, contacting one of a group of servers, and interprocess communication among co-operating processes.

A basic motivation for using multicast is resource conservation via sharing: instead of transmitting information from a sender to each receiver separately, we can arrange for links that are shared to carry the data only once. We can picture a multicast route as a tree rooted at the sender with a receiver at each leaf, (and possibly on internal nodes). Where paths in the tree diverge, network nodes (routers for IP) must duplicate the information in order to forward it further. The tree can be designed so as to maximize shared links and thus minimize resource consumption. In addition, there are resource savings at the sender since it is only required to transmit a single copy of the data.

Support for multicasting in the current version of IP (IPv4) has been evolving for years on the worldwide Internet, while its next version (IPv6) emphasizes multicasting and encourages the replacement of broadcasting with multicasting wherever possible. Using multicasting, services offered by a group of hosts can be identified by a single IP address, resulting in easy resource location and promoting distributed and replicated services.

The explosive growth of wireless communications has attracted interest in the integration of wireless networks with wireline ones and the Internet in particular. Wide area wireless networks allow devices to be connected to the network while roaming freely from area to area. The goal of the Internet designer is to achieve seamless communication for applications as hosts move, without disruptions, while preserving the current routing and addressing mechanisms. This can be achieved by extending IP to transparently handle mobile hosts that attach themselves to various network access points, hiding mobility from the transport service.

In this article we describe proposals for integrating multicasting and mobility in the Internet architecture. We first present IP extensions for host mobility and other extensions for multicasting. We then examine local multicasting mechanisms, focusing on a group membership protocol that is optimized for wireless point to point links. Next, we examine the problems of sending and receiving multicast datagrams in a wide area network. For multicast reception, we describe three alternative proposals and compare them by examining both their applicability and their performance, as well as possible tradeoffs among the two.
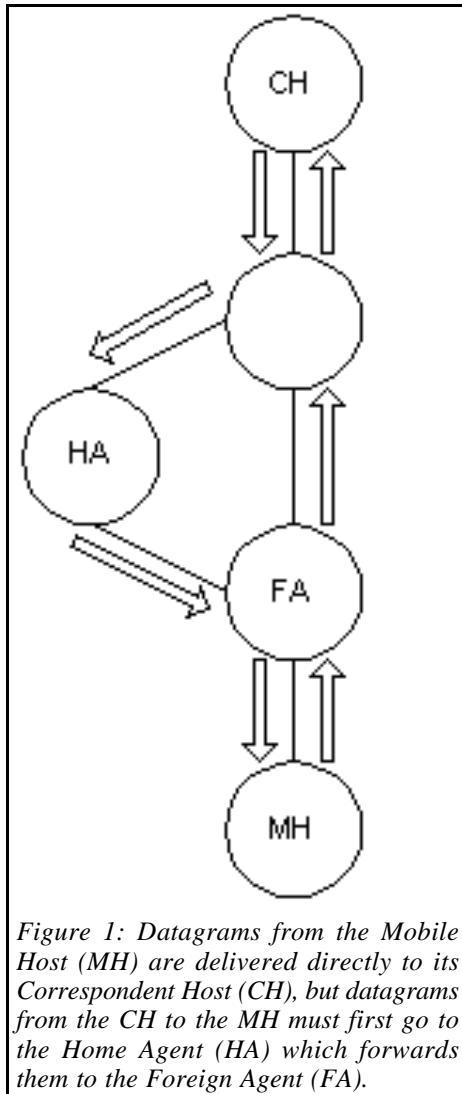
IP Mobility

The goal of IP mobility support is to allow a *mobile host* (MH) to change its point of attachment to the network without losing connectivity at the transport layer, even though Internet transport layer protocols (TCP/UDP) assume that a host's address is fixed. *Mobile IP* provides a mechanism for a MH to retain one address, called its *home address*, as it roams around the network, so that transport associations are not disrupted.

IP datagrams are delivered to their destinations via a series of *routers*. When receiving a datagram, a router examines the *network part* of its destination address. If the network part indicates that the host is local, the datagram can be directly delivered to the host indicated by the *host part* of its destination address, since each router has detailed knowledge of its network(s). However, if the host is not local, the datagram is forwarded towards a router advertising reachability to the destination network. Thus, each router is statically associated with a set of addresses for its directly attached networks, while keeping track of remote hosts via their network addresses.

A MH, on the other hand, attaches itself to various networks, called *foreign networks*, whose network addresses will differ from the one indicated by its unique home address. IP routing however insists on delivering datagrams for the MH to the *home network*. On the draft standard for mobile IP [1], this problem is solved by having a router on the home network of the MH, called the *home agent* (HA), and the router on the foreign network to which the MH is currently attached, called the *foreign agent* (FA), co-operate. When the MH reaches a foreign network, it locates the FA and *registers* with it. Then, it contacts its (permanent) HA, informing it of the FA currently serving it. Subsequently, the MH sends its datagrams through the FA, which then routes them normally, since unicast IP routing typically ignores their source addresses. In contrast, datagrams destined for the MH, are first delivered to the HA on the home network. The HA consults its tables, locates the FA serving the MH, and then *encapsulates* the datagrams within new IP datagrams from the HA to the FA. The FA decapsulates these datagrams and delivers them to the directly attached MH. The method of encapsulating datagrams to work around normal IP routing is called *tunneling*.

Figure 1: Datagrams from the Mobile Host (MH) are delivered directly to its Correspondent Host (CH), but datagrams from the CH to the MH must first go to the Home Agent (HA) which forwards them to the Foreign Agent (FA).

Using these mechanisms, communication with the MH can proceed uninterrupted despite its mobility, using the MH s home address. Tunneling however results in suboptimal *triangle routing* (see Figure 1), because datagrams to the MH are delivered via its HA. Since the MH may move outside the FA s area at any time, there is no de-registration procedure. Instead, information on visiting MHs is deleted from FA tables if renewal registrations are not received periodically.

IP Multicasting

IP multicasting is based on the concept of the *host group* [2]: a dynamic set of hosts identified by a single, class D, IP address. A host can *join* or *leave* a group at any time, in order to start or stop receiving datagrams sent to the group. Sending datagrams to a group is not limited to group members. To deliver datagrams to a group, we need mechanisms to *track group membership* and *route datagrams* towards group members. In the following we categorize the required mechanisms into *local*, which deal with group membership management and local datagram delivery, and *global*, which deal with multicast routing from senders towards remote group members.

Local Multicasting Mechanisms

Locating hosts belonging to multicast groups is the task of the *Internet Group Management Protocol* (IGMP) [2]. Each local network that supports multicasting designates one *multicast router* (MR) as the group manager. This router periodically sends *queries* for group membership in its local area, and the attached hosts reply with *reports* identifying the groups that they participate in. The group manager can then build a list of all groups that are present in the local area and arrange for datagrams sent to these groups to reach the MR, using global mechanisms.

IGMP and the local multicast delivery architecture were designed for multiaccess LANs, where packets are broadcast on the physical medium, so that *native* multicasting is available. The queries are multicast to an address to which everyone is listening and each report is sent to the multicast address for the group in question. Both the router and all group members listen to this address, so that the router can learn of the need for the group and other members can suppress their reports. Queries are periodically repeated, and if no reports are received for a previously present group, the router assumes that it is no longer required. Thus, joining a group causes reports to be periodically sent, while leaving a group does not lead to any explicit action. For a shared medium network, one query and one report per group are needed per period, and the router need only record the presence of a group rather than its recipients.

In contrast, when the router has to support a set of *point to point* (PtP) links, each multicast datagram has to be separately unicast to each interested host. This means that separate queries and reports are required for *each* link and more detailed information must be kept in router tables, either as a host list for each group, or as a group list for each host, even though the local router only needs a simple group list to determine which datagrams should be delivered to it. Many wireless networks provide only PtP links, while some proposals for combining multicasting and mobility use *virtual* PtP links among the multicast router and the receivers.

Since IGMP detects implicitly that a previously present group has no more local members, multicast datagrams are delivered to networks without any recipients for a period, called the *leave latency*. To avoid it, the draft standard for IGMP v2 [3] defines a *leave group* message: when a host leaves a group for which it was the *last* to send a report, it sends a leave group message. This message is only a hint however, so the MR multicasts a *group specific query* to trigger possible membership reports, in order to determine if there are any remaining group members. Thus, leave latency *may* be shortened but not eliminated, while leave messages that do not actually cause the MR to drop a group *increase* overhead.

Global Multicasting Mechanisms

Delivering multicast datagrams to routers serving the corresponding group members requires global router co-operation. The earliest routing mechanism was the *Distance Vector Multicast Routing Protocol* (DVMRP) [4]. With DVMRP each router keeps track of the best paths to the *sources* of multicast datagrams. Whenever a multicast is received, if it arrived via the best path to its source, it is forwarded through all other interfaces of the router, else it is discarded. Thus, datagrams are distributed over a tree composed of the best *reverse* paths, that is, from the best paths from each receiver to the sender. Datagrams are essentially broadcast to all MRs which forward only the required groups locally. DVMRP discovers the best routes to *networks*, to conserve routing table space, using a *distance vector* algorithm.

Since multicasting is not universally supported, multicast routers must frequently communicate over non-multicast aware areas. This is achieved by setting up fixed *tunnels* among routers, where multicast datagrams are encapsulated within unicast IP datagrams at the one tunnel endpoint and are decapsulated at the other, transparently to the intervening routers. These tunnels are *virtual* links, and the collection of multicast aware areas connected by them is a virtual network known as the MBone [5]. In IP, datagram delivery is limited by the *time to live* (TTL) field, which is usually implemented as a limit to the hops that a datagram can make before it is discarded. Since virtual links look like a single hop, MRs only forward multicast datagrams via tunnels if their TTL values exceed certain thresholds, to limit their scope.

A second routing mechanism, the *Multicast Open Shortest Path First* (MOSPF) [6] protocol, is based on a *link state* algorithm, where each router floods information about its adjacent links, including its group membership list, throughout the network, so that all routers know the complete network topology and the location of all group members. Whenever a multicast datagram arrives to a router, the shortest path tree from the sender to all receivers is calculated (and cached), using Dijsktra's algorithm, and the datagram is forwarded accordingly. Again, to reduce routing table sizes, routers only keep track of networks, for both senders and receivers. With MOSPF, datagrams are only propagated when actually needed, a marked improvement over DVMRP.

A third proposal, *Core Based Trees* (CBT) [7], employs a single tree for each group rather than one tree per source. A router called the *core* is chosen in an ad hoc way for each group, and all multicast datagrams are initially sent there. Multicast routers contact the appropriate core before reception begins, building a *reverse* shortest path tree rooted at the core and extending to all receivers. Whenever a datagram is subsequently delivered to any router in the group tree on its way to the core, it is forwarded through all tree links. Routing is normally worse than in the other two proposals, as all messages must first reach the tree. CBT can co-operate with *any* underlying unicast routing algorithm and, as it uses a single tree per group, it makes routing decisions without considering the source address of datagrams. *Protocol Independent Multicast* (PIM)[8], combines the core based and shortest path tree mechanisms.

Local Multicasting Mechanisms for Mobile Hosts

Regardless of the local delivery mechanisms in use, multicast routers need group membership information for their attached hosts. If the MR communicates with its MHs via point to point (PtP) links, either because the local wireless network only offers PtP service, as in cellular telephone networks, or because virtual PtP links (tunnels) are used, additional state should be kept beyond the list of present groups: either which hosts participate to each group or which groups are required by each host. This enables the router to selectively unicast the multicast datagrams over the appropriate PtP links only. Transmission overhead can be reduced by employing this additional state to optimize IGMP behavior, by having each MH *explicitly* join *and* leave a group [9]. The router keeps track of the groups each MH is currently a member of, updating this state when a new join or leave message arrives, or when the MH leaves the area.

The difference between explicit join/leave messages and IGMP v2 leave messages is that the former exploits the fact that in PtP links only one group member may exist. Therefore, there is no gain by periodically repeating queries/reports since no membership reports are suppressed, and leave messages are not just hints but authoritative information. The result is that both periodic queries/reports and leave latency are *completely* eliminated, thus decreasing management and delivery overhead. Furthermore, there is no need for battery powered MHs to wake up periodically for IGMP processing. Existing IGMP report and leave messages can be employed to implement the join/leave mechanism.

When the network uses a shared medium, all datagrams are received by all MHs, therefore native multicasting minimizes data transmissions. However, using the standard IGMP method for group management may be wasteful, as the periodic queries disrupt the operation of every host. Increasing the query interval reduces management overhead, but increases leave latency overhead. The join/leave mechanism is an alternative that minimizes data transmission overhead. However, the group management overhead may be either reduced or increased, depending on MH population, membership dynamics and group membership overlap. If the join/leave mechanism is deemed to be profitable for the MHs, the additional state required at the MR should be acceptable.
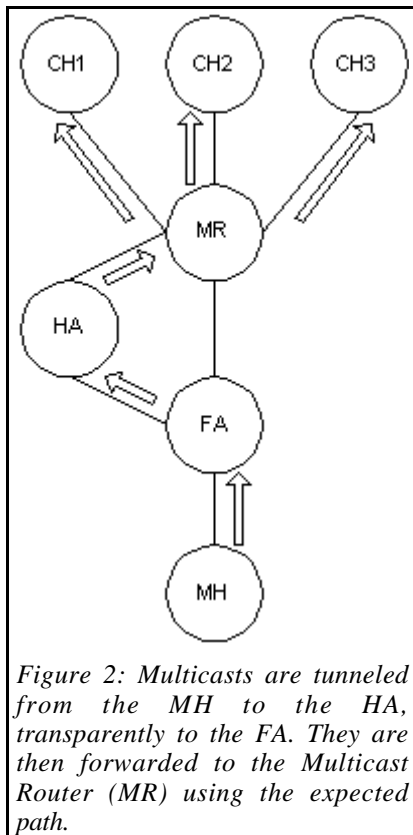
Multicasting from Mobile Hosts

Figure 2: Multicasts are tunneled from the MH to the HA, transparently to the FA. They are then forwarded to the Multicast Router (MR) using the expected path.

IP unicast routing depends solely on a datagram's destination, so MHs simply forward their datagrams to the FA which routes them normally. With DVMRP and MOSPF however, multicast routing relies on a datagram's source, represented by the network part of its IP address. A MH's multicasts are expected from the link used to reach its home network, but when the MH moves to a foreign network its datagrams will arrive on many routers via unexpected links. DVMRP drops such datagrams, while MOSPF forwards them based on an erroneous distribution tree, so that in both cases some destinations are not reached. Since CBT uses a single group distribution tree, datagrams are routed based only on their destination, permitting a MH to correctly send multicasts from any point in the network.

To overcome such routing problems, one approach is to disguise multicasts as originating from an address in the foreign network. Using the FA's address as the source would cause replies to multicasts to be delivered to the FA, while using a temporary local address for the MH, besides stressing the nearly depleted IPv4 address space, would cause replies to a MH's multicasts to be delivered to the wrong MH after the sender moves. A solution is to modify the routes used by MH originating multicasts by tunneling them first to their HAs. From the HA, they can be forwarded as if they had originated from their home network (see Figure 2), thus arriving at each MR via the expected path. As HAs are required to process encapsulated datagrams, recognizing tunneled multicasts at the HA and treating them accordingly is trivial. Tunneling leads to suboptimal routing until the HA, but from that point on standard multicast routing is employed. The draft mobile IP standard allows sending multicasts either using temporary addresses or via tunnels to the HA.
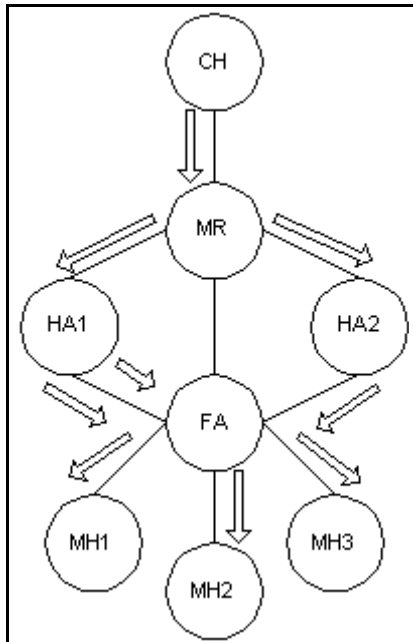
Figure 3: All the MHs served by the FA belong to the group to which the CH sends. The MR delivers the datagrams to HA1 and HA2 which run IGMP. HA1 needs two separate tunnels to MH1 and MH2 converging with another tunnel from HA2 to MH3.

Multicast Reception on Mobile Hosts

Home Agent Routing

A direct mechanism for achieving multicast reception on MHs is to let the HA handle multicast routing, by executing IGMP and delivering multicasts to the MH as if it was at home. Datagram delivery is achieved by tunneling via the FA, while membership reports from the MH can be unicast to the HA (see Figure 3). Since the HA and the MH communicate via virtual PtP links, which may include a wireless link, per MH information must be kept in the HA and IGMP operation could be modified to use explicit join/leave messages to optimize transmission.

A similar tunneling mechanism has been designed to handle local *broadcasts* from the MH's home network. Broadcast datagrams are encapsulated twice: the inner IP header indicates that the datagram's destination is the MH and the outer indicates that it is the FA. The FA receives the datagram, throws away the outer header, and delivers it to the MH by looking at the inner header. The MH then throws away the inner header, and the broadcast datagram emerges. This scheme can also be used for multicast delivery. However, instead of tunneling *all* datagrams, the MH and the HA exchange IGMP messages in order for the HA to tunnel only the required groups to the MH.

The advantage of this approach is its interoperability with existing networks. Multicasting is completely transparent to the various FAs that a MH may use, while the MH and the HA are generally under the same administrative control and therefore may be modified at the same time.
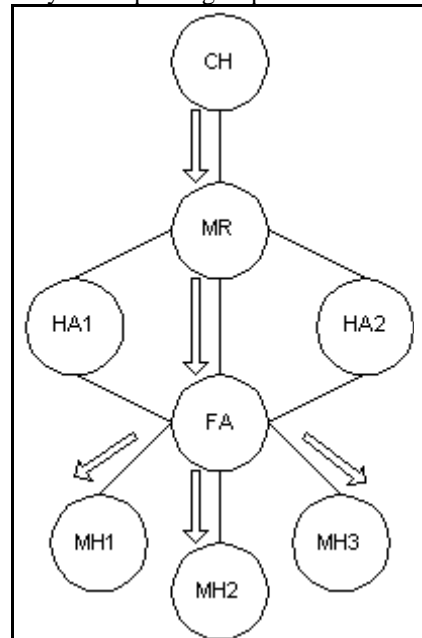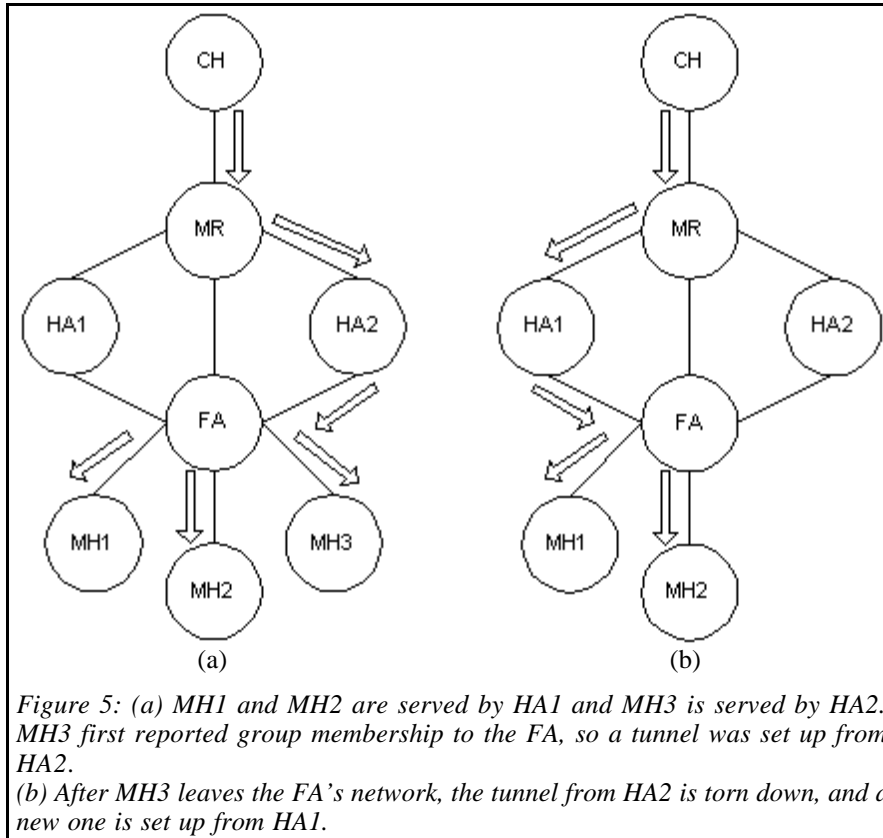

Figure 4: Multicasts from the CH to each MH are forwarded to the local multicast router (FA) and are then delivered to all receivers using any appropriate mechanism. Routing and transmission efficiency are optimized.

The disadvantage of the approach is its inefficiency. First, datagram delivery is suboptimal due to triangle routing. Second, native multicasting cannot be exploited even when supported by the network: multiple MHs receiving the same group need separate tunnels, originating from the same or different HAs, leading to the *tunnel convergence problem*. Since multicasts are doubly encapsulated, they cannot be recognized as duplicates by an unmodified FA.

Foreign Agent Routing

When the FA is willing to support multicasting, the existing IP multicasting model can be used with the FA gathering IGMP information from the MHs and forwarding multicast datagrams to them (see Figure 4). Since global multicast routing is only concerned with forwarding multicasts to routers, the FA acting as a MR *hides* the MH addresses. IGMP operation and local multicast delivery can be optimized, transparently to the rest of the network.

The advantage of this scheme is its complete transparency. By simply gathering membership information from its local network, the FA can interoperate with other routers using any protocol, and routing will be always optimal. Internally, the FA can choose the optimum delivery and group management scheme for the network at hand and enforce the policies and tradeoffs set by the local network provider. The main drawback is that the local network owner may not want to provide multicast service to visiting MHs, either due to the associated datagram delivery overhead or due to multicast routing related resource overhead. Since the MHs can receive multicasts anyway using tunnels from their HAs, causing high overhead due to duplicate and converging tunnels, the former objection is not valid. Therefore, routing overhead should be weighed against the savings in delivery overhead achieved by this scheme.

Combined Routing

Figure 5: (a) MH1 and MH2 are served by HA1 and MH3 is served by HA2. MH3 first reported group membership to the FA, so a tunnel was set up from HA2.
(b) After MH3 leaves the FA's network, the tunnel from HA2 is torn down, and a new one is set up from HA1.

A final approach for receiving multicasts mixes tunneling from the HA with local multicast service from the FA [10]. The FA gathers membership information and arranges for *unique* tunnels to be set up for each group. Thus, the FA carries out local tasks using any appropriate mechanisms, while global delivery is arranged between the FA and the HAs. The tunnel is set up from the HA whose MH first asks for a group (see Figure 5.a). If all the MHs that this HA serves leave the FA's network, the tunnel is torn down by the HA which notifies the FA, and a new one is set up from another HA (see Figure 5.b). The HAs must notify the FAs when tunneling is to be discontinued, since the FA cannot distinguish between inactive and disconnected tunnels.

Local multicast operations can be optimized by the FA transparently to the global mechanism. A claimed advantage is that this approach works without local multicasting support, since the FA is not an MR attached to the MBone. The first disadvantage of this approach is the suboptimal triangle routing used. A second problem is the overhead associated with dynamic tunnel management and double encapsulation. A third problem is determining when the HA should start and stop tunneling datagrams, given that the HA and the FA are under separate administrative control and unlikely to trust each other. The main objection however relates to the scheme s applicability: as both the FA and the HA must be modified to handle multicasts using a non-standard protocol, interoperability is limited. Contrast this with the other schemes, where either only the MHs and their HAs or only the FAs need to be modified.

Comparison of Approaches

We examine the multicast reception approaches from two main perspectives: how easily they can be integrated with existing mechanisms and how efficient they are (see Table 1). Regarding interoperability, the *Modification Scale* and *Modified Entities* criteria show the extent and location of required modifications to existing host protocol software. Regarding performance, *Protocol Overhead* and *Delivery Overhead* show whether additional protocol and data transmissions are required over standard multicasting, *Multicast Routing* compares each approach with standard routing, and *Local Operation* examines whether the approach supports local IGMP and delivery optimizations.

An additional criterion, *Locality Model*, shows which multicast messages are considered local by the MH. This depends on which entity acts as the MR. Support for local multicasting on the foreign network is useful for resource discovery. Finally, there is the *Security Support* criterion: a MH may participate in restricted or encrypted multicast groups. Since the HA and the MH are required by the draft mobile IP standard to maintain security associations, authenticated and encrypted delivery paths can be extended from group senders to MHs via their *own* HAs (and not via *any* HA, as in *Combined Routing*).

	Home Agent Routing	Foreign Agent Routing	Combined Routing
Modification Scale	Minor	Minor	Major
Modified Entities	HA, MH	FA	HA, FA, MH
Protocol Overhead	Yes	No	Yes
Delivery Overhead	Yes	No	No
Multicast Routing	Suboptimal	Optimal	Suboptimal
Local Operation	Unoptimized	Optimized	Optimized
Locality Model	Home Network	Foreign Network	Both Networks
Security Support	Yes	No	No

Table 1: Comparison of multicast reception approaches.

Under all criteria, *Combined Routing* is no better than *Foreign Agent Routing*. A tradeoff exists between ease of application/security, where *Home Agent Routing* is superior, and efficiency, where *Foreign Agent Routing* is best. For networks that do not support multicasting, the former approach will be used by necessity, while the latter approach may eventually prevail due to its superior performance. Migration to the *Foreign Agent* routing would be eased by a dual mode of operation, choosing either approach during registration. Both *Home Agent* and *Foreign Agent* routing are allowed by the draft mobile IP standard.

Conclusion

We have seen how multicasting and mobility can interoperate in the Internet. Although performance and compatibility problems as well as tradeoffs among them are still being investigated, the existing proposals and standards are adequate to support full participation of MHs to multicast groups. Simple modifications to the still evolving protocols can further improve performance, easing the migration of multicast based applications to both mobile and wireless hosts.

Acknowledgments

References

[1] C. Perkins (ed.), IP Mobility Support, *Internet Draft* (work in progress), May 1996.

[2] S. Deering, Host extensions for IP multicasting, *Internet Request For Comments*, August 1989, RFC 1112.

[3] W. Fenner, Internet Group Management Protocol, Version 2, *Internet Draft* (work in progress), September 1996.

[4] S. Deering, C. Partridge, and D. Waitzman, Distance vector multicast routing protocol, *Internet Request For Comments*, November 1988, RFC 1075.

[5] H. Eriksson, MBONE: The multicast backbone, *Communications of the ACM*, 37(8):54-60, August 1994.

[6] J. Moy, Multicast routing extensions for OSPF, *Communications of the ACM*, 37(8):61-66, August 1994.

[7] A. Ballardie, J. Crowcroft, and P. Francis, Core based trees (CBT) - An architecture for scalable inter-domain multicast routing, *Computer Communications Review*, 23(4):85-95, October 1993.

[8] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, and L. Wei, An architecture for wide-area multicast routing, *Computer Communications Review*, 24(4):126-135, October 1994.

[9] G. Xylomenos and G. C. Polyzos, IP Multicasting for Point-to-Point Local Distribution, to appear in *Proceedings of the IEEE INFOCOM 97*.

[10] V. Chikarmane, R. Bunt, and C. Williamson, Mobile IP-based multicast as a service for mobile hosts, in *Proceedings of the 2nd IEEE International Workshop on Services in Distributed and Networked Environments*, pages 11-18, 1995.

Internet Requests for Comments and current *Internet Drafts* are available at *http://www.internic.net*.