A Distributed Location Management Scheme for Mobile Hosts

Rajan Shankaran, Vijay Varadharajan, Michael Hitchens
Department of Computing
Macquarie University, Australia
{rshankar,vijay,michaelh}@ics.mq.edu.au

Abstract

With the increasing growth in mobile computing devices and wireless networks, users are able to access information from anywhere and at anytime. In such situations, the issues of location management for mobile hosts are becoming increasingly significant. Different location management schemes such as Columbia University's mobile IP scheme and IETF mobile IP have been proposed. In this paper, we propose a new distributed location management scheme and discuss the advantages of the proposed scheme over the others. The paper then considers the issues of multicasting in the proposed architecture.

1. Introduction

The technology of mobile communications is rapidly evolving to give a user the ability to access information and services from anywhere and at anytime. Mobile devices are getting more powerful in terms of their computational capabilities while at the same time getting smaller in size. Regardless of size, most mobile computers nowadays are equipped with wireless connections to the fixed part of the network, and perhaps other mobile computers. The resulting computing environment, which is often referred to as mobile or nomadic computing, no longer requires users to maintain a fixed and universally known position in the network and enables them almost unrestricted mobility. The users' expectations are also on the increase with the growth in mobile technology. A user ideally wishes to have continuous access to information irrespective of its current location. Therefore, the process of being mobile should not impede the user's interaction over the net. In other words, mobility itself is transparent to the user. This implies that even though a host migrates from one cell to another, it continues to maintain its network connections without breaking and re-initialising them. That is, in mobility, unlike portability, a computer remains in continuous contact with the network resources needed by its applications. In other words, there is an uninterrupted connectivity between applications and source. Therefore, neither the system nor any of the applications running on the system need to be re-initialised or restarted, even when network connectivity is frequently broken and re-established at new points of attachment.

Therefore, mobility poses certain fundamental questions such as: how do we reliably know at any given instant of time where a particular mobile user is? Who should we approach to procure this information? And once this information is obtained, how is the packet delivered to the mobile host? That is, which is the best way to send the datagrams (unicast and multicast) to the remote recipient? These questions essentially address the location management problems associated with mobile hosts. These issues are further aggravated as the mobility of the user and the number of mobile users increase and the network size grows bigger.

In the past, a number of approaches have been suggested location management schemes for mobile hosts. Two of these approaches have gained some recognition in the wireless community namely the Columbia approach and the IETF approach. In this paper first we discuss the mobile computing architecture of these two approaches and look at how they address the problem of location management. Then in section 3, we propose our model that aims to solve some of the problems inherent in these approaches. Our model eliminates the problem of triangle routing that arises with the notion of a home agent in [1] or a home campus in [2,3]. Section 4 outlines our model's capability to support multicasting without introducing additional overheads. Finally, section 5 considers some further related work and provides concluding remarks.

2. A Brief Review of IETF and Columbia Mobile IP Schemes

The IETF Mobile IP scheme [1] allows the mobile host to use two addresses. The home address is static and is used, for instance, to identify TCP connections. The care-of address changes at each new point of attachment and can be thought of as the mobile node's topologically significant address; it indicates the network number and thus identifies the mobile node's point of attachment with respect to the network topology. In this scheme, the mobile host has a home network where it is registered with a home agent. Upon migrating to a foreign network, the mobile host registers with its home agent via the foreign agent. When in a foreign network, a mobile host selects a foreign agent (via the advertisement messages or through solicitation) and acquires a care of address. The mobile host then registers its care of address with the home agent either directly or via
the foreign agent. The process begins when the mobile node, possibly with the assistance of a foreign agent, sends a registration request with the care-of address information. When the home agent receives this request, it (typically) adds the necessary information to its routing table, approves the request, and sends a registration reply back to the mobile node. Whenever the mobile node is not attached to its home network (and is therefore attached to what is termed a foreign network), the home agent gets all the packets destined for the mobile node and arranges to deliver them to the mobile node's current point of attachment. In other words, the home agent intercepts mobile host's datagrams and forwards them to its care-of address. The packet is modified so that the care-of address appears as the destination IP address. This modification can be understood as a packet transformation or, more specifically, a redirection. These packets are unravelled by the foreign agent and then delivered to the mobile host. The home address and the care of address mapping is maintained by the home agent on behalf of the mobile node and is referred to as a mobility binding. Each mobility binding has a negotiated lifetime limit. To extend this binding limit, the mobile host has to reregister within the lifetime. Upon returning to home, the mobile host deregisters with the home agent and sets its care of address as the permanent address. Deregistration with the foreign agent is not required. It expires automatically. When a node knows a mobile node's current care-of address, a correspondent node can deliver packets directly to the mobile node's home address without any assistance from the home agent.

In Columbia's scheme [3], a virtual mobile subnet is created using a set of routers called Mobile Support Routers. Such a virtual mobile subnet is called a campus. The coverage area under a MSR is called a cell and the communication within the cell is via a wireless medium. The mobile host is given a constant address, which remains unchanged within the campus. When the mobile host moves to a new campus it acquires a temporary nonce address from the mobile subnet of the foreign campus. Each MSR within a campus advertises its presence via beacon messages. These messages are transmitted at regular intervals. Upon receiving such a message, a mobile host initiates the registration process with the MSR. Now suppose a correspondent node (CN) wishes to communicate with a mobile host MH and assume that MH is on a different subnet under MSR1. The packets from CN will initially be routed to its nearest MSR say MSR2. If MSR2 is not aware of MH's current location, it dispatches a WhoHas packet to all the other MSRs within the campus, to locate the whereabouts of MH. MSR1 sends back a reply with an IHave packet revealing the current location of MH. All subsequent packets are now tunneled by MSR2 to MSR1. MSR1 decapsulates the packets and delivers them to MH. If MH migrates to a new segment under a different MSR, say MSR3, then it becomes the responsibility of MSR to send a FwdPkt message to the previous MSR (ie MSR1) informing it of MH's new location. Subsequent packets will now be tunneled by MSR1 to MSR3 and MSR1 will also inform MSR2 using a redirect message of the current location of MH.

When a MH migrates to a "new" campus it becomes its own MSR and is called a popup. In this mode it acquires a new nonce address and registers this with its home campus MSR. This MSR is now responsible for answering WhoHas queries on behalf of this MH. It tunnels all the packets for this MH to its current foreign address where the MH performs its own de-encapsulation. Any packets that the MH transmits are tunneled back to the home MSR for forwarding.

3. Our Architectural Approach

Our architecture is a collection of autonomous systems and backbone. The backbone network consists of several routers called Backbone Routers (or BRs) each of which serves a specific region within the backbone called a cell. The cell helps to exploit frequency reuse and also helps to maintain mobility of hosts, as the bandwidth of mobile systems is limited. The BR consists of a transmitter for communication as well as a database for responding to the information processing needs of a MH. The total area covered by all the BRs comprises the total coverage area of the backbone. The process of crossing a cell boundary and entering into the other is known as handoff. The entire process of handoff is fully transparent to the mobile host and serves to maintain end-to-end data connectivity. A MH can therefore enjoy unrestricted movement within the backbone area. The backbone network itself can support both wireless and wireline environments. BR acts as a gateway between the wireless cell and the wired network. MH uses a wireless link layer to communicate with its BR. It is assumed that a mobile host can reside in only one cell at a given instant of time under the jurisdiction of one BR. Where two BRs overlap, MH communicates with the BR with whom it is currently registered with. On the other hand, BR can have several mobile hosts in its cell.

The backbone network is connected to other network systems called autonomous systems (AS) or domains. An autonomous system is a collection of hosts interconnected by a set of routers and subnets under a common and often a single administration. This is referred to as the Autonomous System Controller that acts as an entry point to AS. In some sense, the Autonomous System Controller acts as a border backbone router. Hence we will use the notation BBR. BBR serves an autonomous domain whereas a BR does not. BBR also acts as a location information server for hosts in the AS. It caches the current location information of a mobile host. Any inter-AS communication must pass through the backbone. All traffic generated within an AS to the backbone and from backbone to an AS must pass through the BBR.

1 Routers within an autonomous system are referred to as Autonomous System Router (ASR), which are capable of supporting mobility.
The backbone network has a principal location server called Rendezvous Point (RP). This is a central repository of location information. It maintains a database of mobile hosts that are currently active in the backbone. When a mobile host successfully migrates to a new cell, it becomes the responsibility of its current BR to pass the location information of this host to the RP for updating its database. When a mobile host enters an AS, first it registers with the BBR. BBR caches this information and sends a copy to RP for it to update its database. Upon receiving this message RP realizes that the concerned mobile host is no longer in the backbone area and therefore deletes the relevant entry from its database. A Backbone Router (BR) advertises routes within its cell range. It performs registration of the mobile host when the mobile host first enters the backbone. It routes packets to and from MHs, tunneling them to RP and BBRs if necessary. BR maintains a database of all mobile hosts currently residing in its cell. It conveys the location information of a mobile host to the RP. The mobile host related information maintained by a BR includes the following: mobile host identity (MHi), mobile host entry and expiry times, the multicast groups that a mobile host belongs to as well as mobile host security information.2

BBR maintains a database of all mobile hosts currently located in the AS. It performs the registration of a mobile host in its AS and assigns a temporary care of address (CoA) for a certain duration. It maintains a mapping of the permanent and the temporary address of the MH. The information about a mobile host that it stores is similar to that maintained by a BR. The Rendezvous Point (RP) maintains a central repository of all mobile hosts that are roaming in the backbone area (obtained via BRs). It does not include entries for mobile hosts located under BBRs (i.e., those in autonomous systems). It should be noted that BBR has two interfaces, one for the AS and the other to the backbone cell. For the hosts located in the cell, RP must maintain relevant entries in its database as they belong to the backbone. It contains such information as the mobile host identity, its current BR identity and security related information. Following multicast groups exist in this architecture: an AllBBR Group: This group includes all the BBRs as well as RP.

3.1 Functions of Principal Components

In the DLM architecture, illustrated in figure 1, the location information is distributed. The location information of a mobile host may be cached with multiple entities. We shall now examine entities that play a principle role in this architecture.

**Backbone Router (BR):** The functions of a BR are as follows:

1. Advertises routes within its cell range.

2. It performs registration of the mobile host when the mobile host first enters the backbone.

3. Routes packets to and from MHs, tunneling them to RP and BBRs if necessary.

4. It maintains a database of all mobile hosts currently residing in its cell.

5. It informs the mobile node about special features provided by it, for example, alternative encapsulation techniques.

6. It lets mobile nodes determine the network number and status of their link to the Internet.

7. It takes part in the handoff procedure.

8. It conveys the location information of a mobile host to the RP.

9. It exchanges information with BBRs about the location of MHs that it needs to serve.

A BR maintains static and a dynamic database. Static database consists of information concerning the BR itself such as its address, and security related information, which are fairly static in nature. The dynamic database contains mobile host related information, which is fairly transient in nature. MH related information is maintained as a data structure indexed by the MHi. See [8] for details of the structure.

**Backbone Border Router (BBR):** BBR has two interfaces, one representing the backbone cell and the other representing the AS. The functionality of a BBR is the same as BR for hosts residing in the backbone cell. It performs some additional functions for hosts residing in the AS. It acts as a gateway into the AS. For hosts within an AS, it:

1. Maintains a database of all MHs. These MHs could be visitors or could be indigenous to AS.

2. Performs registration procedure for the MH and assigns it a temporary care-of-address (CoA) within the AS for a certain duration.

3. Maintains a mapping of the permanent and the temporary address of the MH.

4. Caches the current location information of MH within an AS.

Thus at any given instant of time, the current location of a mobile host is cached with the BR and the governing ASR under whose cell MH resides. It furnishes this information to other backbone routers as and when required. It does not convey this information to RP. It should be noted that BBR conveys the location information for hosts residing in the backbone cell to the RP.

The BBRs in the backbone forms a multicast group called the "AllBBR" group. The multicast messages to this group are used for querying for location information of mobile hosts. Just as BR, BBR also maintains two databases: static and dynamic. Static database consists of information concerning the BR itself such as its identity, and its security related information that is fairly static in.
nature. The dynamic database contains mobile host related information which is fairly transient in nature.

The static database is indexed by BBRid. See [8] for details of the structure.

Rendezvous Point (RP): It maintains a central repository of location information. All mobile hosts currently roaming in the backbone area register their current location with RP. So, at any given instant of time, RP and BR (in whose cell MH is at present located) are the only two entities that are aware of the current location of mobile host.

The information held by RP is as follows:

- MH ListBB: A list of mobile hosts currently roaming in the backbone, their current cell location (Cellid), governing BR address (BRid). It should be noted that MHs residing in AS under the jurisdiction of a BBR are not recorded in RPs database.
- MH lifetime: Provided to the MH during registration process indicating its duration of stay in the backbone.

Mobile Host (MH):

MH moves around in the backbone as well as in AS areas. It always relies on BR or a BBR for service.

1. It engages in the registration procedure with the BR/BBR.
2. Upon entering a new cell, it resets its default route.
3. It conveys its security related credentials as well as group membership information to the BR.

MH maintains static and a dynamic database. The static portion contains details pertaining to MH whereas the dynamic portion consists of information pertaining to MH's location. The static MH related information is maintained as a data structure indexed by the MHid. See [8] for details of the structure.

3.3 Distributed Location Management Scheme (DLM)

A mobile host (MH) is assumed to be assigned a permanent address called MHid. This identity may be assigned to the host by its organisation or through some other independent scheme (such as through an Internet Service Provider). The mobile host retains this identity in the backbone. However, when visiting an autonomous system it may acquire a temporary identity called the care of address (CoA) whose significance is only local. Let us first consider the mobile router discovery process. Our method is similar to the one proposed in [1]. A mobile router advertisement is formed by including a Mobility Router Advertisement Extension in an ICMP conventional router advertisement message. This advertises the services of a mobile router on a link. By means of these
advertisements, a host can determine if it has changed its current cell location or moved from the backbone to an autonomous system region. However we require slight modifications in the message structure, see [7]. For instance, the Registration Lifetime field is not included in the backbone router advertisements as the duration of stay of a mobile host in the backbone area is determined by the RP. However, it is included in the BBR advertisements. In such an advertisement, it indicates the longest lifetime (measured in seconds) that this router is willing to accept in any registration request.

Assume that a MH is roaming in an autonomous system region. Upon receiving an advertisement message from a backbone router, MH realises that it is no longer in the AS region. MH then responds to this advertisement message by sending out a registration request that includes its permanent MHiid. MH sends a registration message to the BR, which then passes it on to RP. The RP generates a roaming profile for MH and caches MH related information in its database. This roaming profile also indicates the duration of stay of MH in the backbone. Since MH is roaming, it moves from one cell to another, each of which is under a different BR. The only two entities which need to be aware of the current location of an MH is its current BR and the RP. When the MH switches its cells, the BR in its new cell informs the RP as well as the previous BR about MH’s movement. The previous BR then transfers the state information related to the MH to the new BR. After the successful completion of handoff procedure, the previous BR deletes the state information associated with this MH. The new BR informs RP of MH’s presence in its cell. RP updates its database.

When a MH receives an advertisement message from a BBR, it realises that it is within an AS. It sends a registration request to the BBR and supplies its original MHiid along with the lifetime desired. Upon successful completion of the registration process, MH acquires a Care of Address (CoA). The significance of CoA is only local to the autonomous system. BBR caches this information within its database, which contains a list of MHs that are currently residing in the AS.

BBR also forwards this message to RP. RP upon receiving such an advertisement learns that a MH is currently residing in a specific AS. RP goes through its database to locate the entry for the concerned mobile host. Upon locating this information, it deletes this entry from the database.

Let us now consider the various scenarios whereby a mobile host moves between autonomous domains and the backbone. We will first consider five cases and then put them together and describe the overall behaviour of each of the components involved in our distributed location management architecture. We will refer to the host that the mobile host is trying to communicate with as the Correspondent Host (CH).

Case 1: Mobile Host and the Correspondent Host in the same AS. Assume both the mobile host MH1 and the correspondent host CH are currently registered in an autonomous system AS1. It is assumed that MH1 has already undergone the process of registering with the autonomous system controller BBR1. Assume that CH wishes to communicate with MH1. CH generates packets for MH1 which are received by its local router (some ASR) and then forwarded to BBR1. BBR1 looks at its visitor's directory to locate MH1. If BBR1 finds MH1 (as it will in this case), it tunnels the packets to MH1’s current location using its care of address. Thereafter, MH1 can communicate directly with CH.

Case 2: Mobile Host and the Correspondent Host are in different AS. Assume that a mobile host MH1 is in an autonomous system AS1 and the correspondent host CH is in an autonomous system AS4. CH wishes to communicate with MH1. It is assumed that MH1 has registered with the autonomous system controller BBR1. BBR1 updates its domain list entry. Packets generated by CH are received by its BBR4. BBR4 checks its registration list to verify if MH1 is currently in its autonomous system. If MH1 is not located in AS4, as it will be in this case, BBR4 dispatches a "WhoHas" query to AllBBR group. Note that in this case RP will not contain location information of the MH as it is residing in an AS. Upon receiving this query message, BBR1 checks its database and finds the concerned MH to be under its jurisdiction. It sends back to BBR4 a "IHave" reply. From BBR1’s reply, BBR4 can find out that the MH is in BBR1’s jurisdiction. Now BBR4 can forward the packets to BBR1, which in turn sends them to MH1.

Case 3: Mobile Host in the Backbone and the Correspondent Host in an AS. Assume a mobile host MH1 is in the backbone currently under the jurisdiction of BR1. We will assume that the MH1 has already undergone the process of registering with the RP via BR1. Assume that a CH is in an autonomous system AS1 and wishes to communicate with MH1. The packets generated by CH are forwarded to its BBR1.

BBR1 checks its registration list to verify if MH1 is currently in its domain. If not found, as will be in this case, BBR1 dispatches a "WhoHas" query to AllBBR group which includes the RP. RP checks its domain list entry and finds the concerned MH1 under BR1 in the backbone. It sends back to BBR1 a "IHave" reply message. This message contains the current location id of MH1. BBR1 can forward the packets to BR1 which then forwards them to MH1. However, if we assume that MHs change their location quite frequently within the backbone and a BR does not cache the location information of MH after a hand-off. In such situations, it may be more appropriate for BBR1 to send the packets to RP which can then transfer to the right BR and MH1. On the other hand, this approach could lead to RP bottleneck.

Case 4: Mobile Host and the Correspondent Host in the Backbone. Assume that both the mobile host MH1 and the Correspondent Host CH are residing in the backbone under the jurisdiction of routers BR1 and BR2 respectively. It is assumed that both MH1 and CH have already
undergone the process of registering with the RP via their respective mobile routers. Let us assume that CH wishes to communicate with MH1. CH generates packets for MH1 and forwards them to BR2. BR2 checks its local cache to find out if MH1 is currently registered. If so, the packets are forwarded to MH1. If not, as will be in this case, BR2 sends a "WhoHas" query to "AllBBR group". RP sends back a "IHave" reply in which case BR2 forwards the packets to RP and then RP forwards them to BR1. Alternatively, RP can send to BR2 the information that MH1 is under BR1 and then BR2 forwards the packets to BR1. The same issues discussed in Case 3 above apply here.

**Case 5:** Mobile Host in the AS and the Correspondent Host in the Backbone. Assume that the mobile host MH1 is in an autonomous system AS1 and the CH is in the backbone. Let us again consider the case where CH wishes to communicate with MH1. The earlier part of the process is same as the one described in Case 4 above. BR2 sends a "WhoHas" query to AllBBR group. The BBR1 responds with a "IHave" message. All the packets are then subsequently routed to BBR1 and from there get delivered to the intended recipient MH1.

**Case 6:** Mobile host and the corresponding host are under the same BR. Assume both the mobile host MH1 and the correspondent host CH are currently registered in the backbone under a BR BR1. It is assumed that MH1 and CH have already undergone the process of registering with the BR1. Assume that CH wishes to communicate with MH1. CH generates packets for MH1 which are received by BR1. BR1 looks at its visitor's directory to locate MH1. If BBR1 finds MH1 (as it will in this case), it tunnels the packets to MH1. Thereafter, MH1 can communicate directly with CH.

### 3.4 Remarks

In summary, if a host in an autonomous domain AS and wishes to communicate with another host, then it first contacts its local BBR in its AS. The BBR checks whether the target host is in all AS. If not, it sends a "WhoHas" query to the AllBBR group. If the host is in the backbone, RP responds with a "IHave" query. Alternatively, one of the BBRs responds if the host is in its autonomous domain. RP maintains a list of mobile hosts in the backbone. If a host in the backbone wishes to communicate with another host, it first contacts its local BR. The BR checks whether the target host is in its cache. If not, it sends a query to "AllBBR group" to determine the BR or BBR under which the target host is residing.

A mobile host may get disconnected while on the move. The disconnection may be transient in nature such as during handoffs or it may go down for an indefinite period of time. MH may also go into a "doze" mode. In such a case, MH can actually inform its peers before disconnecting, so that preventive action can be taken both at the MH and on the wired network to lessen the effect of disconnection on applications. To handle such cases, the BR maintains a timeout element MHTimeout. This information element indicates the amount of time elapsed since MH last contacted the BR. If the breakdown of MH is unprecedented and it does not get an opportunity to inform its BR of this event, then BR simply waits for MHTimeout interval after which it simply purges this MH's entry from its database. RP is also notified after MHTimeout so that the entry is deleted from RP's database. Before the expiry of the timeout, BR caches all packets destined to MH. MH also caches MHTimeout. When in doze mode, MH is aware of the buffer time it has in hand before getting back to BR. If the MH is in an autonomous domain, then it is the BBR that takes on the role of BR. A similar procedure to the above occurs and after timeout, BBR deletes the MH's entry from its database and informs the RP.

From an architectural point of view, as the size of the backbone increases, RP has to accommodate more and more hosts that can cause RP to be a bottleneck. We extend our architecture to include multiple backbone segments. Each of these segments is governed by a RP and may be connected to one or more autonomous systems. We refer to this as a zone. Relationship between zones can be hierarchical or peer to peer in nature. In the hierarchical model, each child RP has a unique parent RP. A child RP sends summary advertisements of its location information to its parent. One can have a hybrid model whereby there can be certain peer to peer links between parent RPs at a given level. These RPs at a given level can then form a multicast group.

### 4. Multicasting

Many emerging Internet applications are one-to-many or many-to-many, where one or multiple sources are sending to multiple receivers. Examples include transmission of corporate messages to employees, communication of stock quotes to brokers, video and audio conferencing for remote meetings, and replicating databases and web site information. Multicast communication is becoming increasingly popular for large scale Internet applications and services. IP multicast [4] is a relatively new feature of IP in which applications send one copy of the information to a group address. However, there are several issues associated with multicast communication that are yet to be resolved. Multicast requires a non-trivial amount of state and complexity in both core and edge routers. These requirements are at odds with the long-standing belief that intelligence should be pushed to the edges of the network. Multicasting is connectionless, which means that a multicast datagram is neither guaranteed to reach all members of a group nor guaranteed to arrive in the same order as they were sent. The protocol delivers a multicast datagram to the destination group members on a best-effort basis. Mobility further aggravates the problem of multicasting. For instance, if the source of the multicast packet is itself mobile, the packet may not get delivered to some members of the group. Also when a mobile host moves from one cell to another or from AS to backbone or
vice versa, it may experience a delay in receiving multicast datagrams. A uniform time to live (TTL) may not hold true as a mobile host is constantly on the move, which causes the number of hops to vary.

Let us now consider a multicast scheme for our distributed location management (DLM) architecture. We suggest the use of Core Based Tree (CBT) [5] for constructing multicast deliver trees in the backbone. CBT has been developed to provide a multicast routing protocol that provides efficient communication between members of sparsely distributed groups; these types of groups are likely to be common in wide-area networks such as the backbone network defined in our scheme.

In our DLM model, the backbone forms the sparse-mode domain. It has a core router that acts as an RP a la PIM-SM [6]. We will refer to this router as mRP (multicast Rendezvous Point)\(^1\). Any multicast communication occurring between two ASs or within the backbone must pass through the core multicast router mRP. Each AS is free to choose its own multicasting scheme (such as DVMRP). In the backbone, CBT is implemented. CBT is designed to work more efficiently when there are only a few widely distributed group members. Instead of broadcasting traffic and triggering prune messages, receivers are expected to send explicit join messages. These join messages are sent to a router acting as a core. Sources are expected to send their data traffic to this same node. The use of a core as a "meeting place" for sources and receivers facilitates the creation of the multicast tree. By deploying CBT, we do not flood the backbone with unnecessary multicast messages.

Let us now examine how the multicast communication can be carried between group members comprising of mobile hosts as well as fixed hosts.

Whenever a mobile host changes its cell, it sends a gratuitous membership report in the new cell for each multicast group in which it is a member. This message forms a section of the registration message. The BR (CBT Router) can start sending the multicast packets addressed to the corresponding groups as soon as it sees membership reports and discovers that it already supports these groups. Alternatively whenever a mobile host enters a cell, the BR as part of its advertisement message also includes the groups that are currently active in this cell. The mobile host either responds with a join message or advertises groups to which it belongs that are missing from the BR's list. This reduces the wireless traffic caused by a premature query cycle. However the latter approach is not suitable from a security point of view. BR then issues a JOIN_REQUEST hop by hop toward the group's core. If the JOIN_REQUEST encounters a router which is already in the group's shared tree before it reaches the core router, the router issues a JOIN_ACK hop by hop back toward the sending router.

The core router (mRP) is ultimately responsible for responding with a JOIN_ACK if the JOIN_REQUEST does not encounter an on-tree CBT router along its path toward the core. As each router forwards the JOIN_REQUEST toward the core they are required to create transient 'join state'. This transient join state includes the multicast group, the JOIN_REQUESTs incoming and outgoing interfaces. This transient state information allows an intermediate router to forward returning JOIN_ACKs along the exact path back to the CBT router which originated the JOIN_REQUEST. In the case of an AS, the BR's function is performed by BBR.

Case 1: MH in the Backbone. Consider a mobile host MH roaming in the backbone area. Upon receiving the advertisement message from the local BR, MH initiates the authentication process\(^2\). Along with the authentication message, MH also indicates what multicast groups it is currently a member of. The local BR serves as a CBT designated Router (DR). It should be noted that a BR can be directly connected to an RP or may be connected via a wired network consisting of several routers. In any case, BR serves as a leaf router in the shared tree. BR only joins the shared tree if they have at least one directly attached group member. When the BR receives a message (IGMP) indicating the membership of a MH to this group G, it first looks into its database to verify if it is already receiving packets for such a group. If yes, then the packets get delivered to MH straightaway. Otherwise, BR forwards a JOIN_REQUEST message to the mRP. The BR and intermediate routers create a transient join state. This allows forwarding of multicast messages coming from the mRP of that multicast group to the BR and group members.

When the MH sends a multicast message to a certain group, the BR of that source encapsulates the first message and sends it to the mRP of that group as a unicast message. After receiving this message, the mRP sends back a PIM JOIN_ACK message to the BR of the MH. While this message is being forwarded to BR, all intermediate routers add a new entry in their multicast forwarding tables for the new (source, group) pair. In other words, each router adds the interface over which the join-request was received to an existing group's forwarding cache entry. This enables multicast messages from this MH to be forwarded to the mRP. mRP will then be responsible for forwarding these multicast messages to other members of the group. It should be noted that until these entries have been added in all intermediate routers' tables, all multicast messages would be forwarded as encapsulated unicast messages. Because the mobile hosts frequently change their locations maintaining a transient state at routers introduces overheads. An alternate design solution will be to send all multicast packets to and from RP as unicast packets.

Case 2: MH in the Autonomous System. If a mobile host makes its entry into an AS then it exchanges a similar set of messages as above with the BBR during the

---

\(^1\) A set of routers can also act as mRPs. Deciding how many mRPs to have and where to place them in the network is a network planning issue and is beyond the scope of this paper.

\(^2\) Not considered in this paper.
registration process to identify the currently active multicast groups in the AS. If the BBR is already receiving the packets for this group then it simply directs them to MH. If not, BBR forwards a unicast JOIN_REQUEST message to the mRP. Then the packets are routed from mRP to the BBR and then sent to MH. As the mobile hosts change their location frequently, the transient join state in each router need to refreshed periodically. In CBT, each child router monitors the status of its parent router with a keepalive mechanism, the CBT "Echo" Protocol. A child router periodically unicasts an ECHO-REQUEST to its parent router, which is then required to respond with a unicast ECHO_REP message.

5. Discussion and Further Work

Let us now briefly compare our proposed scheme with those of IETF and Columbia. In the IETF Mobile IP specification, each mobile host belongs to a home network and is assigned a dedicated home agent. The home agent is primarily responsible for maintaining the location information about the current location of the mobile host and tunnelling packets to that location. This centralized location management architecture leads to a number of issues. First, it results in routing inconsistencies. Packets from a correspondent host first arrive at the home agent (of the mobile host) from where it is tunnelled to the current location of the mobile host. However packets from the mobile host, on the other hand, can go directly to the correspondent host without any tunnelling. This phenomenon, referred to as triangle routing, is alleviated by use of techniques in the route optimization but doing so requires changes in the correspondent hosts that could take a long time to be deployed for IPv4. It is hoped that triangle routing will not be a factor for IPv6 mobility. The other inconsistency is the dogleg routing. When the mobile host moves to the same network as the correspondent host, the packets for the mobile host may still be routed via the distant home agent. Second, although the single home agent model is simple and easy to implement, it can become a single point of failure. In case the home agent fails, it is not possible to reach the mobile host. Third, frequent reporting back to home agent can cause inefficiency when the mobile hosts move frequently. In our DLM approach, no single entity maintains a permanent location directory for mobile host. This is a fundamental difference compared to the IETF scheme. BBR, BR and RP maintain location information. The problem of triangle and dogleg routing do not arise. Since the location directory itself is distributed, a failure of one entity does not render a mobile host inaccessible.

In the Columbia scheme, there is no concept of a single home location directory. Rather the location directory is spread across several mobile support stations within a virtual subnet called the campus. This leads to the concept of a home campus rather than a home agent. When a mobile host leaves its campus and registers elsewhere in a foreign campus, it may still need to inform its home campus of its current location. Columbia scheme uses close to optimum routes when operating in the local area. However in a wide area, it generally uses sub-optimum routes, that is, it again introduces triangle routing. For instance, if two mobile hosts from the same mobile subnet are connected to a foreign campus network, the packets between the two hosts may have to be routed through the home campus. In our scheme, even though we may have a dedicated home network associated with a mobile host, this information does not play a critical role in formulating routing decisions. For instance, if two mobile hosts residing in the backbone under different BRs need to communicate with each other, a search is made in the RP. In other words, the search is first local and then remote.

In the area of security, we have developed a security framework for the DLM model. We have considered both intra and inter AS movements and developed security protocols. These protocols enabled authentication as well as the establishment of a shared secret key between two mobile users within an AS and residing in different ASs. These issues are addressed in a separate paper [7]. We have also extended this to consider secure multicasting and have explored the development of mechanisms for achieving security and efficiency. In particular, how to achieve security and efficiency when it comes to mobile hosts joining and leaving multicast groups is not trivial.

References