Mobile Location Management in ATM Networks

Malathi Veeraraghavan, *Member, IEEE*, and Gopal Dommety

***Abstract*— This paper presents two mobile location management algorithms for ATM (asynchronous transfer mode) networks based on the PNNI (private network-to-network interface) standard. The first solution is called the *mobile PNNI scheme* because it builds on the PNNI routing protocol. It uses limited-scope (characterized by a parameter S) reachability updates, forwarding pointers, and a route optimization procedure. The second solution is called the LR (*location registers*) *scheme* because it introduces location registers (such as the cellular home and visitor location registers) into the PNNI standards-based hierarchical networks. This scheme uses a hierarchical arrangement of location registers with the hierarchy limited to a certain level S. Analytical models are set up to compare the average move, search, and total costs per move of these two schemes for different values of the CMR (call-to-mobility ratio), and to provide guidelines for selecting parameters of the algorithms. Results show that at low CMR's (CMR <0.025), the LR scheme performs better than the mobile PNNI scheme. We also observe that the two schemes show a contrasting behavior in terms of the value to be used for the parameter S to achieve the least average total cost. At low CMR's, the parameter S should be high for the mobile PNNI scheme, but low for the LR scheme, and vice versa for high CMR's.**

***Index Terms*—Mobility management, PNNI, wireless ATM.**

I. Introduction

MOBILITY management algorithms enable networks to support mobile users, allowing them to move, while simultaneously offering them incoming calls, data packets, and/or other services. In connection-oriented networks, *mobility management* consists of *location management* (tracking mobiles and locating them prior to establishing an incoming call), and *handoff management* (rerouting connections, on which the mobile user was communicating while moving, with minimal loss of user data). Since asynchronous transfer mode (ATM) networks are connection-oriented, both of these aspects of mobility management need to be studied to support mobile users in ATM networks.

This paper addresses the location management problem in ATM networks. The two aspects of this problem, mobile tracking and mobile locating, are also referred to as *MOVE* and *FIND* operations, respectively, in [1] and [2]. Mobile tracking is the procedure by which the network elements update information about the location of the mobile. Mobile location is the procedure by which the network finds the location of the mobile while delivering incoming calls, data packets, or other services. The information acquired during the tracking phase is used in the locating phase.

Since both ATM networks and cellular telephony networks are connection-oriented, the IS-41 and GSM MAP (mobility application part) standards [3], [4] offer a natural starting point for the design of location management algorithms in ATM networks. On the other hand, the PNNI (private network-to-network interface) protocol standard defined by the ATM Forum [5] is also a candidate starting point for an ATM location management algorithm. Using these two starting points, we propose two algorithms for location management in PNNI standards-based ATM networks. In the first algorithm, we add features to the PNNI protocol to enable it to handle mobile users. We refer to this solution as the *mobile PNNI* scheme. At a high level, this scheme is based on mobile IP [6], [7], as will be explained later. In the second algorithm, we introduce location registers of the type used in cellular telephony networks into the PNNI standards-based ATM networks. These location registers are databases that track mobile locations and respond to location queries. This solution is referred to as the LR (*location registers*) scheme. Thus, our two solutions, the mobile PNNI scheme and the LR scheme, can be viewed as representing the "mobile computing" and "cellular telephony" approaches, respectively.

We briefly review prior work on location management in Section II. Next, we describe the proposed *mobile PNNI scheme* and the proposed *LR scheme* for location management in PNNI standards-based ATM networks in Section III. A comparative analysis of these two schemes is included in Section IV. Our conclusions are presented in Section V.

II. Prior Work

Prior work on location management includes the cellular IS-41 MAP (mobility application part) standard [3] and several improvements proposed in [1], [8]–[12]. The cellular IS-41 scheme consists of using a two-level hierarchy of location registers called home location registers (HLR's) and visitor location registers (VLR's) to track mobile locations using *registration notification (REGNOT)* messages. An HLR is assigned to a mobile based on its permanent address, while a VLR, which is typically collocated with a mobile switching center (MSC), is assigned based on the current location of the mobile. Incoming calls to mobiles are delivered after executing a mobile location phase, wherein the call-originating switch generates a mobile *location request (LOCREQ)* to the HLR of the mobile which, in turn, generates another query to the VLR/MSC. Among the improvements proposed to this scheme are the extremes of the "flat" scheme [9] and the "hierarchical"

Manuscript received August 30, 1996; revised May 2, 1997. This work was supported in part by the Advanced Technology Program of the National Institute of Standards and Technology, U.S. Government.

M. Veeraraghavan is with Bell Laboratories, Lucent Technologies, Holmdel, NJ 07733 USA.

G. Dommety is with the Department of Computer and Information Science, The Ohio State University, Columbus, OH 43210 USA.

Publisher Item Identifier S 0733-8716(97)06090-3.

scheme [10]–[12]. The former proposes using a single-level hierarchy of location registers, while the latter proposes building a rooted tree of location registers. In the *flat* scheme, upon receiving a LOCREQ, the HLR assigns a temporary address based on the VLR/MSC at which the called mobile is located rather than require an additional message exchange from the HLR to the VLR/MSC to obtain a temporary address assignment. The mobile's permanent address is tunneled in the call setup message, while the temporary address is used to route the connection from the call-originating switch to the mobile's current switch. The *hierarchical* scheme uses a hierarchy of location registers to localize both mobile tracking and mobile locating messages. A registration is propagated up the hierarchy until it reaches a location register beyond which there is no change of information regarding the mobile's location. The call setup message (or an explicit location query) is sent up the hierarchy until it reaches a location register that knows the location of the mobile, from which point the hierarchy is traced in the downward direction to reach (or determine) the exact switch where the mobile is located.

The flat scheme results in lower computation costs, but incurs larger communication costs than the cellular scheme, while the hierarchical scheme achieves the opposite (lower communication costs, but higher computation costs). By building a rooted tree of location registers, the need for home location registers is eliminated, thus removing the need for "long-distance" signaling messages. On the other hand, to determine the location of a mobile, the location query needs to be stopped and processed at a much larger number of location registers (on the average). This increases the computation costs of the network. In contrast, the flat scheme increases the overall signaling load (communication costs) on the network since registrations and location queries need to be sent "long distance" to the HLR's of mobiles, but it also decreases the computation costs since processing is needed only at one node for both registrations and location queries. The flat scheme also results in a lower mobile location delay due to the one-hop location query processing which, in turn, leads to a lower overall call setup delay. Other improved schemes, such as the forwarding scheme of [1] and the anchor scheme of [8], are in between these two extreme schemes in terms of computation and communication costs.

In contrast, the *location registers (LR)* scheme proposed in this paper for wireless ATM networks is a hybrid scheme whose parameters can be set to default to one of the two improved schemes, i.e., the flat scheme or the hierarchical scheme. It essentially uses a hierarchy of location registers, but limits the hierarchy by lopping off the tree at some level S, beyond which it resorts to the flat scheme approach of updating/consulting a home location register. It also uses the concept of tunneling the mobile's permanent address in the call setup message, as was proposed for the flat scheme. Such a hybrid scheme allows a network provider to implement one of the two extremes or some in-between scheme by selecting S according to computation and communication costs. The novelty of the LR scheme lies in this concept of lopping off the tree at level S, which allows for the simultaneous minimization of both communication and computation costs.

Other related work on location management includes mobile IP [6], [7] with caching and route optimization extensions [13]. Our proposed mobile PNNI scheme is similar to mobile IP in that registrations and triangular routing are used, but with the difference that, since ATM networks are connection-oriented, the principle of routing to the home and forwarding from the home to the mobile's current location is applied on "calls" rather than on "packets." Other differences stem from the fact that the mobile IP scheme does not integrate mobility into routing protocols, such as OSPF (open shortest path first), BGP (border gateway protocol), etc. On the other hand, in our proposed mobile PNNI scheme, the PNNI routing protocol is used to send limited reachability updates for mobiles to a neighborhood of switches, allowing for calls originating in this neighborhood to be routed directly to the mobiles. Thus, the novelty of the mobile PNNI scheme is that: 1) it defines how the location management aspects of tracking and locating are integrated into ATM signaling and routing protocols, 2) it proposes the use of limited reachability updates in the PNNI routing protocol to create a "neighborhood" around the mobile that knows the exact location of the mobile so that calls originating in this neighborhood are routed directly on the shortest paths without triangular routing through the home, and 3) it includes a route optimization scheme for rerouting connections in ATM networks while maintaining cell sequence (a factor that does not need consideration in route optimization associated with mobile IP).

III. Proposed Location Management Algorithms for Mobile ATM Networks

In this section, we propose *two* location management schemes for wireless ATM networks. In the ATM Forum, PNNI standards [5] have evolved to define hierarchical networks that are subdivided into peer groups. We propose both location management schemes for ATM networks based on the PNNI standards.

The *first approach* simply enhances the PNNI protocol to handle mobile users, and is referred to as the *mobile PNNI scheme*. The PNNI routing protocol is used to convey reachability information about endpoints to ATM switches. This is exploited to accommodate mobile endpoints. The "scope" parameter (set to some number S) available in PNNI routing protocol messages is used to limit the region of nodes which receive reachability updates as mobiles move. There is no explicit mobile location phase prior to connection setup. Instead, connections are set up to mobiles according to the reachability information at the switches. Switches within the region defined by S (relative to the position of each mobile) have the correct reachability information for mobiles. Calls originating from such switches will be routed on "shortest paths." However, calls originating at switches outside the region defined by S will be routed toward the home switch of the mobile, and subsequently forwarded to the current location of the mobile. This implies the need for a *procedure to set and/or update forwarding pointers* at the home switch of the mobile, and creates the need for a *procedure to optimize the route* of calls set up on circuitous routes. These enhancements

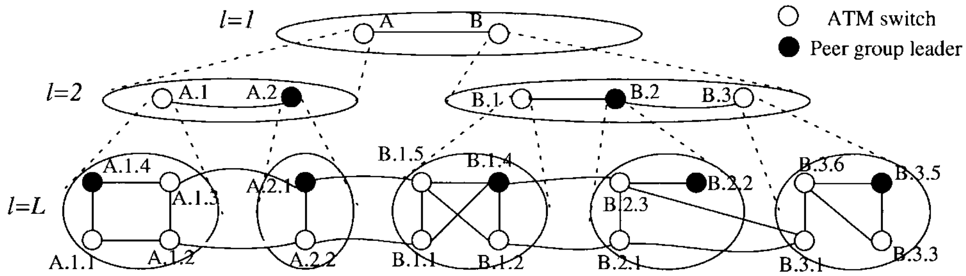


Fig. 1. PNNI-based hierarchical ATM network.

are proposed to the PNNI protocol to enable it to support mobile endpoints.

The *second approach*, the *LR (location registers)* scheme, isolates the effect of mobility from the PNNI routing protocol. The cellular concept of using location registers to handle mobile users is introduced into PNNI-based ATM networks. Location registers (databases) are placed within the peer group structure of these ATM networks. However, instead of directly adopting the IS-41-based location management scheme, we use a combination of the two improved schemes reviewed in Section II. In this approach, location registers track the location of mobiles, and respond to location queries generated prior to connection setup. Thus, unlike the mobile PNNI scheme, the LR scheme uses an explicit mobile location phase prior to connection setup.

A. Mobile PNNI Scheme

We first give a brief overview of the PNNI routing protocol in Section III-A1 to show how *fixed endpoints* are supported in networks based on the PNNI standards. Next, we describe our proposal for supporting mobile endpoints in such networks. In Section III-A2, we describe the *architectural* addition needed to support mobile endpoints. In Section III-A3, we describe the *mobile tracking* procedure. Section III-A4 describes how incoming calls to mobiles are routed in networks using this location management scheme. Finally, *route optimization* is briefly addressed in Section III-A5.

1) Overview of the PNNI Routing Protocol (Support of Fixed Endpoints): PNNI standards-based ATM networks are arranged in hierarchical peer groups as shown in Fig. 1. At the lowest level ($l = L$), ATM switches are shown connected in arbitrary topologies. A PGL (peer group leader) is elected in each peer group. This node represents the peer group at the next higher level peer group. In this role, it is termed the LGN (logical group node) representing its lower level peer group. Nodes within a peer group exchange detailed PTSP's (PNNI topology state packets), and hence have complete information of the peer group topology and loading conditions. A PGL summarizes topology/loading information received in its peer group, and generates PTSP's in its role as LGN to members of the higher level peer group. Each member of the higher level peer group receiving this summarized information will send information to members of its child peer group (downward flow). This exchange of *topology and loading information* constitutes the *PNNI routing protocol* [5]. Using this mechanism, each node in the network has the complete

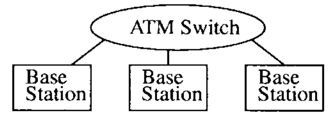


Fig. 2. Configuration of a zone.

topology/loading information of its lowest level peer group, and also the topology/loading information of its ancestor peer groups. This information is used to determine routes of connections when a call arrives.

As part of the PTSP's, *reachability information* is propagated among nodes to indicate where endpoints are located. Endpoint addressing in the ATM Forum standards is based on the NSAP (network service access point) addressing format [14]. All three forms of NSAP addressing support *hierarchical addressing, where the prefix of the address indicates the peer group in which the endpoint is located*. A switch will have exact reachability information for endpoints within its level-L peer group, indicating the switch at which each such endpoint is located. However, for endpoints in other level-L peer groups, the switch will only know the higher level peer group through which the endpoints can be reached. PTSP's carrying reachability updates also propagate up and down the hierarchy, as explained earlier for the PTSP's carrying the topology and loading information. Reachability information advertised by a node has a scope associated with it. The scope denotes a level in the PNNI hierarchy, and represents the highest level at which this address can be advertised or summarized [5].

As an example of a PNNI-based network, consider the network shown in Fig. 1. The numbering scheme used for the nodes reflects the peer group structure. Node A.1.4 belongs to peer group A.1 at level 2, and to peer group A at level 1. Node A.1.4 is the peer group leader of peer group A.1. It advertises that all A.1 nodes are reachable through itself in peer group A. Upon receiving this advertisement, LGN A.2 sends a PTSP to all of its lower level nodes. Using this process, nodes in A.2, such as A.2.1 and A.2.2, learn that all A.1 endpoints are reachable through A.1. Similarly, nodes A.2.1 and A.2.2, learn that all nodes with the address prefix B are reachable through LGN B since they maintain the topology, loading conditions, and reachability data for all of their ancestor peer groups.

2) Architecture to Support Mobile Endpoints: Mobile endpoints can be supported in the PNNI hierarchical architecture in the following manner. Mobiles are located at base stations, which are assumed to be organized as in cellular networks, with multiple base stations connected to each switch as shown

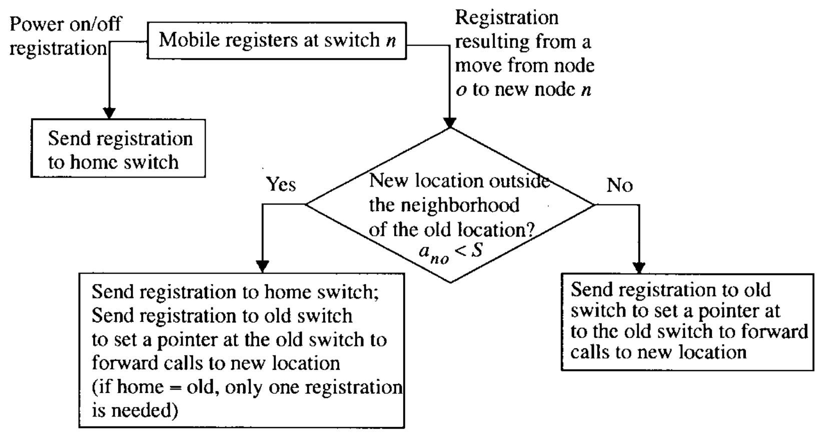


Fig. 3. Flowchart representing how forwarding pointers are set.

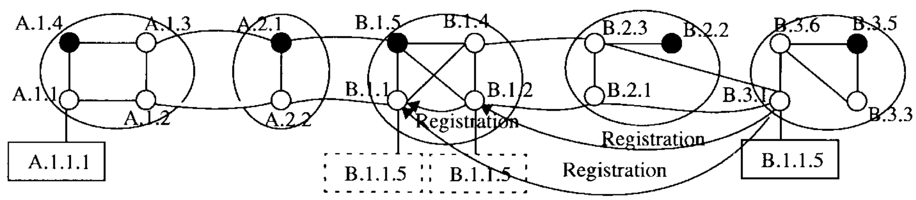


Fig. 4. Registration messages.

in Fig. 2. Zone-change registrations are used to limit air interface registration traffic, where a "zone" consists of all of the base stations under a single switch. When a call setup arrives at a switch, it pages all of its base stations to determine the exact base station on which it is located. General configurations, allowing base stations to be connected to multiple switches, and/or with different definitions of zones, are possible. We have not considered these configurations in this paper for simplicity reasons. The configuration shown in Fig. 2 ties in with the architecture of Fig. 1 at the lowest level. In other words, some of the switches in Fig. 1 are connected to a set of base stations as shown in Fig. 2. These base stations offer wireless access to mobile endpoints.

3) Mobile Tracking: When a mobile powers on or changes locations, the mobile tracking procedure uses a combination of *setting forwarding pointers* at the home and old (in case of a move) locations of the mobile by sending registration messages, and *sending limited reachability updates* (with a scope S) using the PNNI routing protocol to override the *default summarized reachability information*, which indicates that the mobile is at its home.

Setting of forwarding pointers (sending registrations): Fig. 3 shows a flowchart of the actions involved in setting forwarding pointers. In two of the three cases shown in Fig. 3, forwarding pointers have to be set at the *home* in order to route calls generated by nodes outside the neighborhood (defined by scope S). In the third case, when the mobile does not change neighborhoods during a move, the forwarding pointer data in the home switch are accurate even after the move, and hence no registration is sent to the home. Forwarding pointers have to be set at the *old* location in the case of a move for the following reason. The forwarding pointer is needed to handle calls that the home switch may have forwarded to the old location before it receives the registration message updating its forwarding pointer with the new location of the mobile. Also, calls originated within the old neighborhood will be directed to the old location until the old location issues a reachability update to cancel the limited reachability update override issued earlier for the mobile. Such calls will need to be forwarded from the old location to the new location of the mobile.

As examples, consider the mobile B.1.1.5 located under a base station connected to its home B.1.1 as shown in Fig. 4. Let $S = L$, which means that the neighborhood of a mobile is the set of switches in its level-L peer group, as shown in Fig. 4. If the mobile moves to a base station under switch B.1.2 (case of Fig. 3 in which the new location is in the neighborhood of the old location), then a *registration* message will be sent from B.1.2 to B.1.1 (as shown in Fig. 4). Further, if it moves to B.3.1, two *registrations* are sent to set forwarding pointers at the home and old locations (B.1.1 and B.1.2, respectively) of the mobile (case of Fig. 3 in which the new location is outside the neighborhood of the old location).

Reachability updates: After setting forwarding pointers, the feature of "sending triggered updates of topology information for significant change events" in the PNNI routing protocol is used to generate reachability updates to override default summarized reachability information. Before describing how reachability updates propagate, we define *three* terms: *ancestors-are-siblings level*, *scope*, and the *neighborhood* of a node. The *ancestors-are-siblings level* a_{ij} of nodes i and j is the level at which the ancestors of the two nodes i and j belong to the same peer group. The *scope* S is used to set the stopping point for reachability information propagation. If scope is S, reachability data sent by a node i do not propagate to any node j for which $a_{ij} < S$. The *neighborhood* G_i of a node is defined to include all nodes j such that $a_{ij} \geq S$.

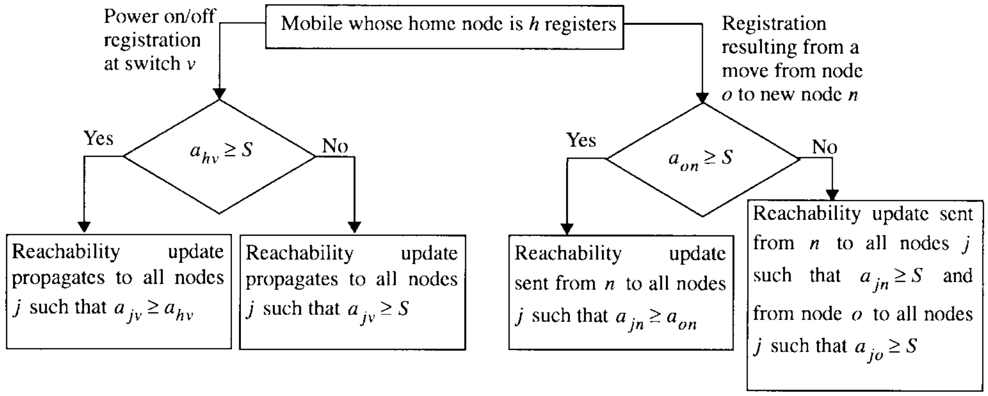


Fig. 5. Flowchart representing how reachability updates are propagated.

As an example, consider the PNNI-based network shown in Fig. 1. The ancestors-are-siblings level of nodes A.1.1 and A.2.2 is 2. If the scope $S = 2$, the neighborhood of node A.1.1 includes all nodes in peer group A, but excludes all nodes in peer group B.

Reachability updates are propagated according to the rules shown in Fig. 5. If a mobile *powers on* within its home neighborhood ($a_{hv} \geq S$), reachability data need to be changed at only a few nodes (the exact set of nodes updated is indicated in Fig. 5). However, if it powers on outside this neighborhood, the whole new neighborhood receives a reachability update about the mobile overriding its default summarized reachability, which indicates that the mobile is at its home. The same rules apply for reachability updates sent when a mobile powers off. *As mobiles move*, if the new location is within the neighborhood of the old location, reachability updates are sent up only to a subset of nodes whose reachability data change as a result of the move (the exact set of nodes is defined in Fig. 5). On the other hand, if the mobile moves outside its current neighborhood, the entire new neighborhood needs to receive a reachability data update that overrides the default summarized reachability data about the mobile. In addition, the old neighborhood is also updated to cancel the limited reachability update that overrode the default summarized reachability data. In effect, this resets reachability information about the mobile to indicate that the mobile is at its home location. Using this approach, all nodes within a mobile's neighborhood know its exact location, while nodes outside its neighborhood believe that the mobile is at its home location. The reason for this arrangement is to allow calls originating from a switch within a mobile's neighborhood to be routed directly to the mobile (without having to be routed to the home switch first). The forwarding pointer at the home switch allows for calls originating at nodes outside the neighborhood to be routed to the home location as per reachability data in these nodes, and then forwarded to the mobile's current location based on the forwarding pointer. The scope parameter S allows the network provider to use a large neighborhood by setting S to a low value (which will lead to a large number of reachability updates, but will result in more calls being routed directly) or vice versa for a small neighborhood.

As an example, consider the PNNI-based network of Fig. 1, and assume that the scope S is 2. If a mobile A.1.1.5 powers on at A.1.2 (i.e., within its home neighborhood), the reachability update overriding the default reachability data only propagates to nodes in peer group A.1. No PTSP is sent from A.1 to A.2 since there is no change of reachability data stored in A.2 nodes regarding mobile A.1.1.5. If, instead, it powers on at a base station connected to node B.1.1 (i.e., outside its home neighborhood), reachability updates will be propagated through peer group B.1, and then upward (i.e., PTSP's carrying reachability updates are sent from B.1 to all nodes of peer group B), and finally, downward from LGN's other than B.1 in peer group B to their child peer groups. Since $S = 2$, no reachability updates are sent to nodes in peer group A, which all believe that the mobile is at its home A.1.1.

Details regarding how registration messages are transported are twofold. First, each mobile maintains the identifiers of its old and home switches, allowing it to communicate this information when registering at a switch (power on, power off, or move). This information is used by the new switch to generate the registration message to the home or old switch. Second, we assume the availability of connectionless transport to send location management messages, such as registrations. One such transport mechanism is connectionless ATM (CL–ATM) proposed in [15]. Without this assumption, on-demand connections would need to be set up and released for the transport of every *registration* message, which creates a considerable processing and signaling overhead.

4) Connection Setup/Mobile Locating: In the mobile PNNI approach, there is no explicit mobile location procedure prior to connection setup. Instead, connection setup proceeds with every switch "believing" its reachability information. Fig. 6 shows how incoming connections to mobiles get routed in the mobile PNNI scheme. The path taken depends on the locations of the calling party, and the home and visiting locations of the called mobile. If the calling party is in the mobile's neighborhood or the mobile is in its home neighborhood, the call is routed directly to the mobile. Otherwise, the call is routed to the home switch first since the switches outside the neighborhood of the mobile have default reachability information. In this scenario, the home switch forwards the call to the current location of the mobile. If the mobile has moved recently, the path may also depend on whether the call arrives after the reachability updates have propagated and forwarding

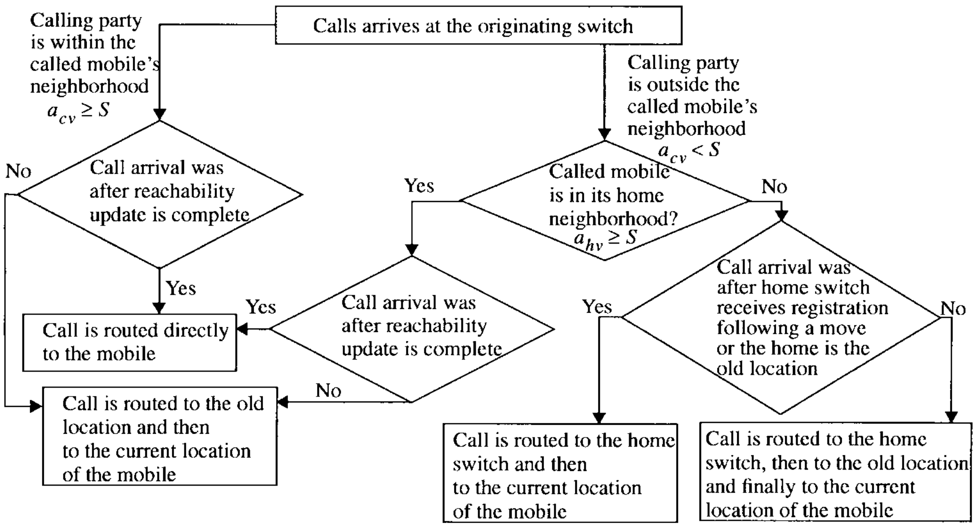


Fig. 6. Flowchart representing how connections are routed.

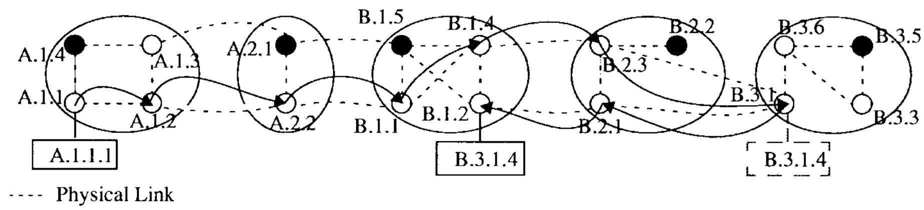


Fig. 7. Connection setup.

pointers have been set, or whether the call arrives prior to the completion of reachability update propagation and/or the setting of forwarding pointers.

We illustrate some of the cases shown in Fig. 6 with examples. Consider that A.1.1.1 issues a call setup to the mobile B.3.1.4, currently located at B.1.2, as shown in Fig. 7. If $S = 2$, then the called mobile is in its home neighborhood, and the calling party is outside the called mobile's neighborhood. Call setup proceeds through peer groups A.1 and A.2, and arrives at peer group B (at a node B.1.1, which is in peer group B) since the reachability information in the nodes of peer group A indicates that the mobile is in its home peer group (B). Once the call setup message arrives at peer group B, it is routed efficiently to B.1.2 since all nodes in the neighborhood (which includes all nodes in peer group B) have accurate reachability for mobile B.3.1.4. A second example illustrates the case when the calling party is outside the called mobile's neighborhood, the called mobile is in its home neighborhood, and the call arrives at a switch before it is updated with the correct reachability information about the mobile. Consider the situation in which the mobile B.3.1.4 has just moved from B.3.1 to B.1.2 and a call arrives at switch B.1.1 before B.1.1 is updated with reachability information for the mobile endpoint B.3.1.4. In this case, switch B.1.1 may choose[1] $\{$B.1.1, B.1.4, B.2, B.3$\}$ as the shortest path by which to reach peer group B.3 based on its current reachability information for mobile B.3.1.4 (which points toward B.3). The call is then routed from the old location B.3.1 to B.1.2. Thus, the connection route will be inefficient, as shown in Fig. 7 (with the arrows indicating the connection route). As a third example, consider that $S = 3$, and that a call to mobile B.3.1.4 is generated by an endpoint attached to switch B.2.1. Since $S = 3$, the B.2 nodes are not updated about the move of the mobile B.3.1.4 from B.3.1 to B.1.2. This is an example of the case when the calling party is outside the called mobile's neighborhood and the called mobile is not in its home neighborhood. In this case, the connection will be routed to the home and then to the new location (from B.2.1 to B.3.1, back to B.2.1, and then to B.1.2).

For connections being forwarded from the old (or home) location of the mobile to its current location, the call setup message needs to "tunnel" the mobile's home address while using the mobile's current (temporary) address to perform connection routing [9], [16]. In the second example described above (also shown in Fig. 7), when the connection is rerouted from B.3.1 to switch B.1.2, if the called party number parameter in the SETUP message indicates the home address of the mobile, then node B.2.1 will again turn the connection setup back toward B.3 (if its reachability information for B.3.1.4 is not yet updated). To avoid this, switch B.3.1 must use a temporary address, such as B.1.2.0 (where the "0" extension indicates "mobile users") in the called party number field. This will allow node B.2.1 to route the connection toward node B.1.2. Upon receiving the setup, B.1.2 will recognize the B.1.2.0 number in the call setup to indicate a mobile. It then looks for the tunneled mobile's home address in the setup

[1]The first node receiving the call setup message in each peer group determines the route of the connection through that peer group.

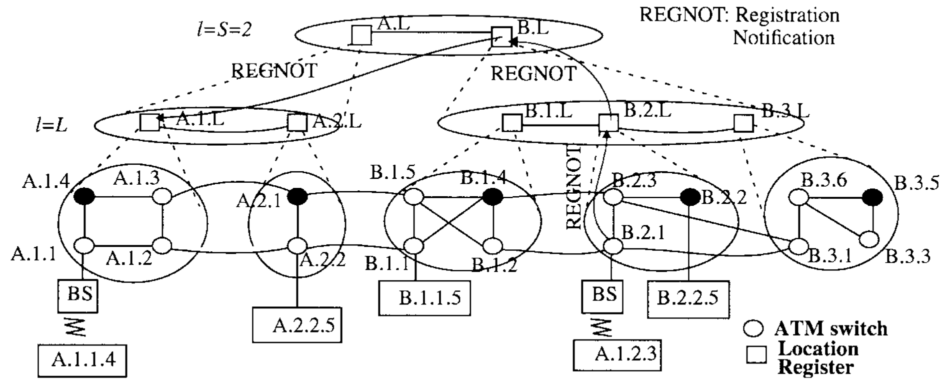


Fig. 8. Location registers arranged hierarchically.

message, i.e., B.3.1.4, to page for the mobile and complete connection setup.

5) Route Optimization: As shown in the examples of Section III-A4), connections may be inefficiently routed due to the lack of correct reachability information. This implies a need for route optimization. A similar need for route optimization exists in mobile IP networks [13]. However, unlike IP networks, ATM networks deliver cells in sequence. The ATM route optimization procedures should maintain cell sequence. The route optimization is performed in two steps. First, a "switchover node" at which the connection is to be rerouted from the old path to the new path is found, and a new segment is set up from the route optimization-initiating switch to the switchover node. Second, user data are switched over from the old path to the new path using "tail" signals and buffering to perform this action without loss of cell sequence. Details of the switchover node selection procedure are presented in a separate paper [17]. Comparisons of our proposed method for switchover node selection to other schemes, such as those proposed [18], are described in [17]. Details of how *tail* signals and buffers are used to switch data from the old path to the new path are also described in [17]. *Tail* signals were originally proposed for improving routes of handed-off connections in [19], [20].

B. Location Registers Scheme

In this scheme, the PNNI routing protocol reachability information is disregarded for mobile endpoints. Instead, an explicit tracking and locating procedure is overlaid on a PNNI-based network using location registers. The LR scheme architecture, the mobile tracking procedure, and the mobile locating procedure are defined in the following subsections.

1) LR Scheme Architecture: Fig. 8 shows hierarchically organized location registers (LR's). The switches are represented by circles. Location registers only exist from level L up to some level S (as explained in Section II, we lop off the tree at level S). We make an assumption that each peer group is assumed to have one LR. This assumption can be relaxed, and multiple LR's may be located in each peer group. This is effectively equivalent to creating a sublayer under the lowest layer of switches, and applying the same concept of allocating one LR per peer group of this new sublayer.

Location register A.1.L (we use the .L extension to avoid confusing this node with the A.1 logical node shown in Fig. 1 for the mobile PNNI scheme) is assumed to track all mobiles attached to switches within peer group A.1 (i.e., mobiles located at base stations connected to switches A.1.1, A.1.2, A.1.3, and A.1.4). Similarly, A.2.L is assumed to track all mobiles located at base stations connected to switches A.2.1 and A.2.2. A home LR is assigned to a mobile based on its permanent address, e.g., A.1.L is the home LR of the mobile A.1.2.3.

The hierarchy of location registers helps localize mobile tracking and locating costs. However, if the hierarchy is carried to the topmost level ($l = 1$) as in the hierarchical scheme of [10], the processing requirements could be high. If computation costs are more than communication costs, it is more expensive to stop and process REGNOT (registration notification) or LOCREQ (location request) at each LR in the hierarchy than to send one such request as a connectionless message to the home. Hence, we limit the hierarchy to level S, and resort to the flat scheme approach of updating and/or querying the home LR of the mobile (see Section II). However, if the home LR were to track the lowest level ($l = L$) LR currently tracking the mobile as in the flat scheme, the long-distance signaling costs of updating or querying the home LR would be high. Hence, the home LR only tracks the Sth level LR for each mobile, and only receives location queries if none of the LR's up to level S of the calling mobile's switch can respond to the query. The parameter S allows the LR scheme to be flexibly implemented as a flat structure, or as a rooted hierarchical tree, or as a mixed structure combining these extremes.

2) Mobile Tracking: When a mobile *powers on*, the switch connected to its base station receives a power-on registration message. This switch sends a REGNOT registration notification to its LR at level L. This, in turn, causes REGNOT's to be generated to the ancestor LR's upstream up to an LR at level S. If the visiting switch is distinct from the home switch, the LR at level S sends a REGNOT to notify the home LR of the mobile that the mobile is currently in its domain. REGNOT's are sent as connectionless packets using the ATM NSAP address of the mobile as the destination [15]. The home LR's of all mobiles visiting at switches other than their home switch track the Sth level LR of the mobile in

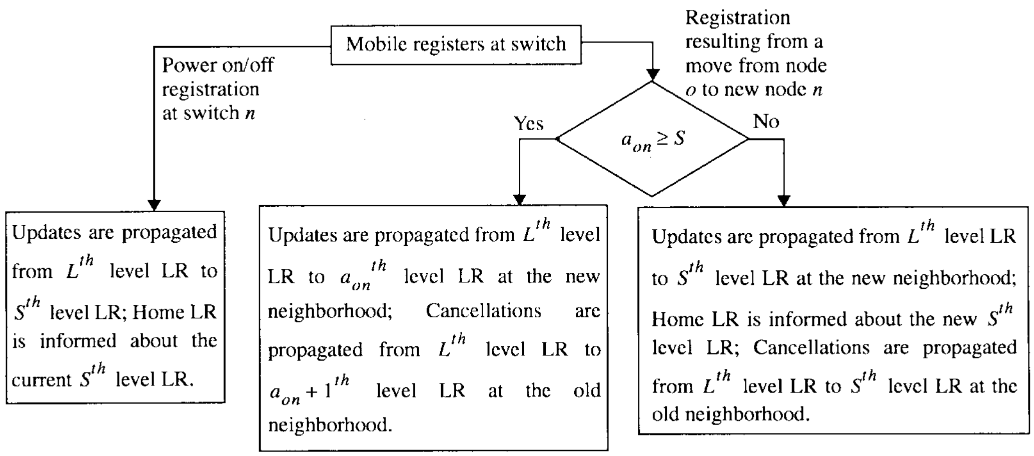


Fig. 9. Flowchart representing how location registers are updated in the LR scheme.

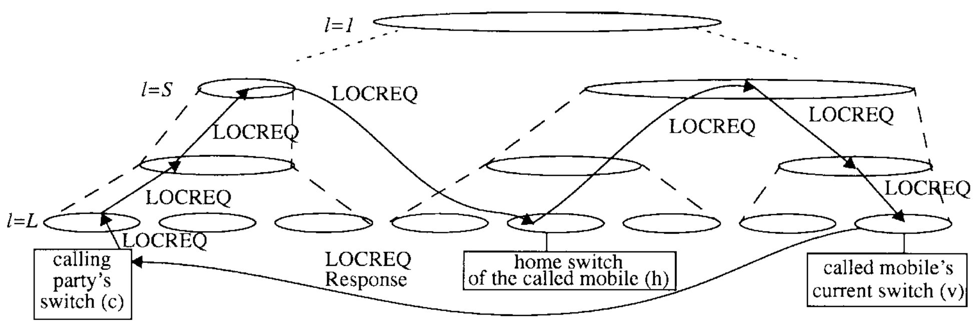


Fig. 10. Mobile locating procedure in the LR scheme (worst case).

its current location. Power-off registrations are handled in a similar manner as power-on, whereby LR's up to level S are informed that a mobile powered off, and if the mobile was visiting (away from home), its home LR is also notified. Fig. 9 shows how location registers are updated in the LR scheme.

Next consider *zone-change registrations*, which are generated as mobiles move from a base station connected to one switch to a base station connected to another switch. The hierarchy of LR's is exploited to limit the propagation of registration information for such movements. On receiving the registration message, the new switch sends a REGNOT message to its level L LR. This, in turn, propagates the REGNOT message upward to the LR which is a common ancestor of the LR corresponding to the old switch and the LR corresponding to the new switch, or up to level S, whichever is lower in the hierarchy (higher in numerical value). A message is sent by the new switch to the old switch, informing the old switch about the movement of the mobile. The old switch then generates a REGCANC (registration cancellation), which is sent to its level L LR. This, in turn is propagated upward, canceling the old information in the LR's. If the Sth level LR tracking the mobile changes due to the move, then the home LR of the mobile is updated.

For example, if the mobile A.1.1.4, shown in Fig. 8, moves to a base station connected to switch A.1.2, only LR A.1.L needs to be updated. One cancellation is required at the switch. On the other hand, if it moves from switch A.1.1 to a switch A.2.2, then REGNOT's are sent to LR A.2.L from switch A.2.2, and subsequently from LR A.2.L to LR A.L since LR A.L is the common ancestor LR of the LR's corresponding to the old and new switches. Since the LR at level S (A.L) did not change, there is no REGNOT sent to the home LR of the mobile. However, a cancellation message is sent from A.2.2 to A.1.1, which in turn generates a REGCANC from A.1.1 to A.1.L. Finally, if the mobile moves from switch A.2.2 to switch B.1.1, REGNOT's propagate from switch B.1.1 to LR B.1.L, and then to LR B.L. Since there is a change in the Sth level LR tracking the mobile, LR B.L notifies home LR A.1.L. In addition, a REGCANC is generated by B.1.1 to A.2.2, which passes upward to LR A.2.L, and then to A.L.

3) Mobile Locating: To find a mobile prior to call setup, a chain of location registers is traced. The length of the chain depends on the location of the calling party and the current location of the mobile. The called party's switch begins by checking to see if the called mobile is located at a base station in its domain. If so, it completes the call without generating any LOCREQ's.

If the called mobile is not located at a base station within its domain, it generates a LOCREQ to its LR. Such requests are forwarded upward in the hierarchy of LR's. If an LR at some level k has information (pointer to a child LR) regarding the location of the mobile, then it sends LOCREQ's downward toward the called mobile's current location. The location query will be resolved by the level-L LR of the switch at which the called mobile is located, and the response will be sent directly to the calling party's switch.

If, however, none of the LR's, from the level-L LR of the calling party's switch to the Sth level LR, knows the location of the called mobile as shown in Fig. 10, the Sth level LR sends a LOCREQ to the home LR of the called mobile. It uses connectionless transport [15] to send this message. The called mobile's home switch will then forward this message to the home LR of the mobile. Since the home LR tracks the Sth level LR of its mobiles, it forward the LOCREQ to the Sth level LR tracking the mobile in its current location. This LR generates downward LOCREQ's according to the information it has about the called mobile. The LOCREQ will reach the level-L LR of the called mobile's switch. The response is sent directly from this LR to the calling party's switch, as shown in Fig. 10. The address tunneling concept of the flat scheme described in Section II is also used in the LR scheme.

As examples, we consider call originations from three endpoints, B.2.2.5, B.1.1.5, and A.2.2.5, all targeted at mobile A.1.2.3 (see Fig. 8). In the first example, when switch B.2.2 generates a LOCREQ for mobile A.1.2.3 to its LR B.2.L, the latter can immediately respond since the called mobile A.1.2.3 is located within its region. In the second example, switch B.1.1 sends the LOCREQ (in response to the call setup request from its endpoint B.1.1.5 to mobile A.1.2.3) to its LR B.1.L. Since it has no pointer regarding this mobile, it simply generates a LOCREQ to the higher level LR B.L. This register has a pointer indicating that B.2.L is tracking the mobile. Hence, a LOCREQ is sent to this LR. Since B.2.L is the level-L LR for the called mobile, it responds, indicating that the mobile is located at switch B.2.1. This response is sent directly to switch B.1.1 (instead of retracing the pointers backward), allowing it to initiate call setup to the called mobile's switch. In the third example, where endpoint A.2.2.5 generates the call setup to mobile A.1.2.3, the LOCREQ sent by switch A.2.2 traverses the chain of LR's, A.2.L, and A.L. Since neither of these LR's has information on the location of the called mobile and $S = 2$, A.L sends a LOCREQ to the home LR of the called mobile A.1.L. This LR forwards the LOCREQ to LR B.L since each home LR tracks the Sth level LR of its mobiles. LOCREQ's are then sent downward from B.L to LR B.2.L, which responds with a temporary address for the mobile, indicating that the mobile is located at switch B.2.1.

IV. Performance Analysis

In this section, we analyze the two schemes described in Section III and compare their costs. We determine the tracking and locating costs for a mobile in each of these schemes using analytical models. The analysis also allows us to provide insights into the effects of different parameters on the performance of each scheme. In the mobile PNNI scheme, we do not account for the route optimization phase in this analysis. After addressing certain preliminaries in Section IV-A, Sections IV-B–IV-D describe how the average tracking cost per move, average search cost per call setup, and average total cost per move are computed, respectively. Numerical results are provided in Section IV-E.

A. Preliminaries

We first define the notation used in this analysis, and then describe a basic property of PNNI standards-based hierarchical networks that is repeatedly used in the analysis.

1) Notation: Table I lists the symbols used in this analysis section, along with their definitions. Fig. 11 shows the notation used in this paper for numbering levels in a PNNI network.

2) Property: For three nodes x, y, and z, the following relations hold between their *ancestors-are-sibling levels* (for the definition of the term "ancestors-are-sibling level," see Section III-A3):

$$\text{Case I:} \quad a_{xy} < a_{xz} \Rightarrow a_{yz} = a_{xy} \tag{1}$$

$$\text{Case II:} \quad (a_{xy} = a_{xz}) \Rightarrow a_{yz} \leq a_{xz}. \tag{2}$$

The proof of this property can be inferred from Fig. 12. If $a_{xy} < a_{xz}$, the arrangement of nodes is shown in Case I of Fig. 12, from which it is clear that $a_{yz} = a_{xy}$. A similar argument extends for Case II. Also, note that since $a_{ij} = a_{ji}$, we use these terms interchangeably.

B. Mobile Tracking Costs

The cost of tracking a mobile includes the costs incurred during power-up, move, and power-down procedures. We first characterize these costs, and then compute the average move cost.

1) Tracking Cost in the Mobile PNNI Scheme: Since the tracking procedure in this scheme uses the PNNI routing protocol for sending reachability updates, we first quantify the cost of a reachability update U_K (defined in Table I). The cost of updating reachability data for a mobile using the PNNI routing protocol at all nodes whose ancestors-are-siblings at levels l such that $l \geq K$ is given below:

$$\begin{aligned} U_K &= m_K + \sum_{i=K+1}^{L} m_i \left(\prod_{j=K+1}^{i} (m_{j-1} + 1) \right) \\ &= \left(\prod_{i=K}^{L} (m_i + 1) \right) - 1. \end{aligned} \tag{3}$$

Equation (3) can be explained as follows. We assume that the PTSP's flooded in a peer group are routed on an MST (minimum spanning tree). Although peer groups at the same level may have MST's of different lengths, we make a pessimistic assumption that all peer groups at a given level have MST's of the same length as the peer group with the longest MST. If the MST of a peer group at level i is of length m_i, then the cost of sending a PTSP with the updated reachability information within a peer group of level i is m_i. This explains the first term in (3) for the topmost level peer group witnessing this update (K). The cost of updating all other peer groups from levels $K+1$ to the lowest level L is the second term in (3).

The costs of individual tracking procedures in the mobile PNNI scheme are summarized in Table II. The cost of tracking a mobile in the mobile PNNI scheme includes the cost of updating reachability data for a mobile, and the cost of sending messages to the home (and old) locations of the mobile to set

TABLE I
NOTATION

Symbol	Meaning
L	Number of peer group levels in the network; level 1 is the topmost level and level L+1 represents individual switches in the network (see Fig. 11).
a_{ij}	The ancestors-are-siblings level of two nodes i and j (see Section 3.1.3 for definition).
h, v, o n, c	Subscripts used to represent the home, visiting, old and new locations of a mobile, and the calling party, respectively.
S	Scope indicating the stopping distance for reachability update propagations.
G_i	The neighborhood of node i (see Section 3.1.3 for definition).
U_K	Cost of updating reachability data sent by a node i to all nodes j such that $a_{ij} \geq K$.
m_i	Length of the longest MST (minimum spanning tree) among all peer groups at level i for $i = 1, 2, \ldots L$; $m_{L+1} = 1$; $m_i = 0$ for $i > L+1$. There are $m_i + 1$ peer nodes in a peer group of level i.
p_i	Average (among all node pairs) length of the "shortest-path" between nodes of a peer group at level i.
N_b	Number of base stations in a registration area (i.e., a switch in this case).
D_{ij}	Distance, in terms of the number of nodes on the route, from node i to node j.
h	Cost of sending a long distance message (such as the registration message).
R_k	Cost of updating or querying LRs up to level k from a switch.
c_i	Cost of updating or querying a level i location register, where $c_i = 0, i > L$.
ρ	CMR (Call-to-Mobility Ratio). It represents the number of calls per move.

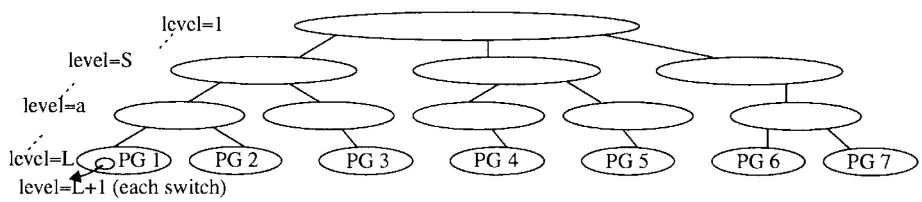


Fig. 11. Illustration of the hierarchy numbering notation.

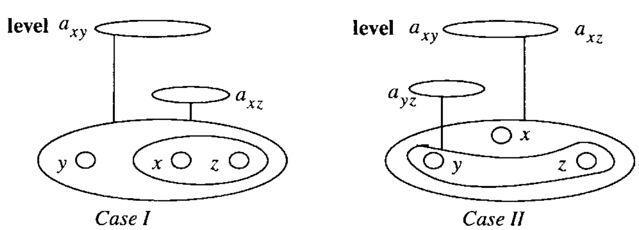


Fig. 12. Relative positions of ancestors-are-siblings levels for nodes x, y, and z.

forwarding pointers. The cost of sending a message from node i is assumed to be 1 if the recipient is within the neighborhood G_i (as defined in Table I). If the recipient is outside the neighborhood G_i, the cost is h, where $h > 1$.

The cost of updating reachability information depends upon the relationship between the ancestors-are-siblings levels and the scope parameter S (see Fig. 5). If a mobile powers on such that $a_{hv} \geq S$ (where a_{hv} is the ancestors-are-siblings level as defined in Table I for the home and visiting nodes of the mobile), then reachability updates are sent to only to those nodes whose siblings are ancestors at the a_{hv} level or at a lower level in the hierarchy. This explains the reachability update cost $U_{a_{hv}}$ shown in Table II for this case, where $U_{a_{hv}}$ is given by (3). The cost of setting a forwarding pointer at the home location is assumed to be 1 since the home is in the neighborhood of the mobile ($a_{hv} \geq S$). The cost of the second row in Table II can be similarly reasoned.

TABLE II
TRACKING COSTS IN THE MOBILE PNNI SCHEME

Procedure	Relative locations			Cost
Power on	$a_{hv} \geq S$			$U_{a_{hv}} + 1$
	$a_{hv} < S$			$U_S + h$
Move	$a_{on} \geq S$			$U_{a_{on}} + 1$
	$a_{on} < S$	$a_{hn} \geq S$		$U_S + U_{hn} + h + 1$
	$a_{on} < S$	$a_{hn} < S$	$a_{oh} \geq S$	$U_S + U_{oh} + 2h$
	$a_{on} < S$	$a_{hn} < S$	$a_{oh} < S$	$2U_S + 2h$
Power off	$a_{hv} \geq S$			$U_{a_{hv}} + 1$
	$a_{hv} < S$			$U_S + h$

Move costs depend upon the relative distances between the old and new locations of the mobile, and the home and the new locations of the mobile (see Figs. 3 and 5). Four cases are possible, as shown in Table II. Depending on the case, the reachability update cost depends upon one or more of the following: a_{on}, a_{hn}, a_{oh}, and S, where the subscripts o, n, and h represent the old, new, and home location of the mobile, respectively. For example, in case $a_{on} < S$, $a_{hn} < S$, $a_{oh} \geq S$, the reachability update has to be sent to the entire neighborhood G_n around the new switch. Reachability cancellations around the old switch need only propagate up to level a_{oh} since other nodes within the neighborhood G_o will continue storing the same reachability as before (pointing toward home for this mobile). Other cases can be similarly reasoned. Before generating these reachability updates, registrations are sent to set forwarding pointers as described in Section III-A3). The terms 1 and h in the move costs shown in Table II represents the costs of these registrations, depending upon whether the registration has to be sent inside or outside the neighborhood of the new location. In cases when a mobile powers on/off at the home, it is not necessary to send an explicit registration message to home. We ignore this minor optimization in the analysis. The power-off costs are the same as power-on costs, as shown in Table II.

The average tracking cost per move in the mobile PNNI scheme ($\overline{M_{\text{PNNI}}}$) is given by (4)

$$\begin{aligned}\overline{M_{\text{PNNI}}} &= \overbrace{\sum_{i=S}^{L} P(a_{on}=i)(U_{a_{on}}+1)} + \sum_{i=1}^{S-1} P(a_{on}=i)\\ &\cdot \left\{ \overbrace{\sum_{j=S}^{L} P(a_{nh}=j)(U_S+U_{hn}+h+1)} \right.\\ &+ \sum_{j=1,\, j\neq i}^{S-1} P(a_{nh}=j)(2U_S+2h) + P(a_{nh}=i)\\ &\cdot \left\{ \sum_{j=i}^{S-1} P(a_{oh}=j \mid a_{nh}=a_{on}=i)(2U_S+2h) \right.\\ &\left.\left. + \overbrace{\sum_{j=S}^{L} P(a_{oh}=j \mid a_{nh}=a_{on}=i)(U_S+U_{oh}+2h)} \right\}\right\}. \end{aligned} \tag{4}$$

The first two terms account for the first two rows of the move cost in Table II. The third row of the move cost from Table II accounts for the last term of the equation. It is nonzero only if $a_{nh} = a_{on}$. This is because, if $a_{hn} \neq a_{on}$, (1) shows that a_{oh} is the smaller of the two values. But since both values are smaller than S, $a_{oh} \geq S$ cannot occur. For the case when $a_{nh} = a_{oh}$, a multiplicative factor showing the conditional probability $P((a_{oh} = j | a_{nh} = a_{on} = i))$ is needed since a_{oh} is dependent on the values of a_{nh} and a_{on}. The fourth row of the move cost shown in Table II corresponds to the third and fourth additive terms in (4). The third term shows the case when $a_{nh} \neq a_{oh}$, and the fourth term shows the case when $a_{nh} = a_{oh}$. The latter requires a conditional probability since it involves all three terms a_{on}, a_{nh}, and a_{oh}. Expressions for the probabilities in (4) are derived in the Appendix.

TABLE III
TRACKING COSTS IN THE LR SCHEME

Procedure	Relative locations	Cost
Power on	$a_{hv} \geq S$	$R_S + 1$
	$a_{hv} < S$	$R_S + h$
Move	$a_{on} \geq S$	$R_{a_{on}} + R_{a_{on}+1} + 1$
	$a_{on} < S$ $a_{hn} \geq S$	$2R_s + h + 1$
	$a_{on} < S$ $a_{hn} < S$	$2R_s + 2h$
Power off	$a_{hv} \geq S$	$R_S + 1$
	$a_{hv} < S$	$R_S + h$

2) Tracking Cost in the LR Scheme: Mobile tracking costs in the LR scheme are shown in Table III. Tracking in the LR scheme essentially requires updating pointers at LR's (see Fig. 9). The cost of updating pointers at LR's up to and including a level k LR is given by R_k as defined in Table I. The cost of updating/querying LR's in the path from the level-L LR to the level-k LR is given by

$$R_k = \sum_{i=k}^{L} c_i, \qquad \text{where } c_i \text{ is defined in Table I.} \tag{5}$$

The entries in Table III are explained as follows. When a mobile powers on/off, all LR's in the path up to level S are updated, and a message is sent to the home LR. In cases when a mobile powers on/off at the home, it is not necessary to send an explicit registration message to the home LR. We ignore this minor optimization in the analysis. The message to the home is treated as a long-distance message of cost h if the home LR is not in the neighborhood of the mobile.

Tracking cost during a move consists of the cost incurred to set up new pointers, delete old pointers, and send a message to the old switch, and/or to the LR of the home switch. When $a_{on} \geq S$, pointers are updated at LR's up to level a_{on} relative to the new location, and are deleted at LR's up to level $a_{on}+1$ relative to the old location. When $a_{on} < S$, pointers are updated at LR's up to level S relative to the new location, and are deleted at LR's up to level S relative to the old location. The cost of updating the home LR is given by h or 1, depending on a_{hn}. The average tracking cost per move in the LR scheme, $\overline{M_{\text{LR}}}$ is given below

$$\begin{aligned}\overline{M_{\text{LR}}} &= \sum_{i=S}^{L} P(a_{on}=i)(R_{a_{on}} + R_{a_{on}+1} + 1)\\ &+ \sum_{i=1}^{S-1} P(a_{on}=1)\end{aligned}$$

$$\cdot \left\{ \sum_{j=S}^{L} P(a_{hn} = j)(2R_S + h + 1) + \sum_{j=1}^{S-1} P(a_{hn} = j)(2R_S + 2h) \right\}. \quad (6)$$

The above equation can be readily explained from the move costs shown in Table III. We also refer to the cost $\overline{M_{LR}}$ as the "average move cost."

C. Mobile Search Costs

In this section, we define the average "search" cost incurred during call setup to a mobile. In the LR scheme, this is the cost of determining the mobile location since this scheme has an explicit location phase. This cost may be mobile location delay or the signaling load (in Mbits/s) incurred to send location queries and receive responses. On the other hand, the search cost for the mobile PNNI scheme is more difficult to define. If bandwidth is of concern, the overhead of the scheme can be characterized by the average extra bandwidth required for the connections that are routed inefficiently. We first define the search costs for these schemes, and then formulate the average search costs in these schemes.

1) Search Cost in the Mobile PNNI Scheme: Given that there is no there is no explicit mobile location phase in the mobile PNNI scheme, we define the "search" cost in this scheme as the number of extra hops needed to route forwarded connections. In other words, for connections that are misrouted, the search cost in this scheme is

$$S_{\text{misroute}} = D_{ch} + D_{hv} - D_{cv} \quad (7)$$

where D_{ij} is defined in Table I as the number of nodes on the route from node i to node j, and the subscripts c, h, and v represent the calling party, home location of the called mobile, and visiting location of the called mobile, respectively. As described in Section III-A4), some of the calls originating at nodes outside the neighborhood of the current location of a called mobile will be routed inefficiently. In addition, some calls originating within this neighborhood may also be routed inefficiently if the call setup request arrives soon after the called mobile moved, and the reachability update has not propagated to all of the relevant nodes. For purposes of this analysis, we ignore this cost for two reasons. First, if the reachability update limiting level S is high (numerical value is large), reachability updates will presumably propagate fast, allowing us to ignore this cost. On the other hand, if S is low, the mobile tracking costs become significant, thus allowing us to ignore search costs while comparing the total costs of the two schemes. The search cost in the mobile PNNI scheme is listed as shown in (8), at the bottom of the page.

The first case of (8) is justified because, when a mobile is in its home neighborhood, the entire network has the correct reachability information about the mobile, and hence no calls are misrouted. All of the nodes within the neighborhood of the mobile have correct reachability information due to the reachability updates propagated as part of the mobile PNNI scheme. The summarized reachability information in the nodes outside the neighborhood of the mobile (default information) indicates that the mobile is in its home neighborhood. In the second case of (8), since the calling party is located in the neighborhood of the called mobile, the call is not misrouted (see Fig. 6).

The last three cases shown in (8) represent the cases where the mobile PNNI scheme may incur a search cost by setting up a connection that later requires route optimization. Pessimistically, we treat all such calls as misrouted calls. First, we provide a method for estimating D_{ij}. The distance D_{ij} between nodes i and j is approximated as

$$D_{ij} = \prod_{k=a_{ij}}^{L} p_k \quad (9)$$

where p_k is the average (among all node pairs) length of the "shortest path" between nodes of a peer group at level k. For the worst case performance, the maximum length of the "shortest path" can be taken. By this definition, the distance D_{ij} between nodes and depends on the value of their *ancestors-are-siblings level* a_{ij}.

From the property described in (1) and (2), we know the relationship between the *ancestors-are-siblings level* of different nodes. When $a_{hv} < a_{ch}$, the cost associated with misrouting, given by (7), $D_{ch} + D_{hv} - D_{cv}$ is equal to D_{ch} since $a_{cv} = a_{hv}$. Similarly, when $a_{ch} < a_{hv}$, the cost associated with misrouting, given by (7), $D_{ch} + D_{hv} - D_{cv}$ is equal to D_{hv}. This case is illustrated in Fig. 13. It also demonstrates that this estimate of the cost is approximate because the border node in the peer group PG1 which receives the call setup may be closer to the home node than to the visiting node, making the exact number of extra hops in the misrouted connection different from D_{hv}. Finally, if $a_{hv} = a_{ch}$, then the search cost is $2D_{ch} - D_{cv}$ and $a_{cv} \geq a_{ch}$ [as per (2)].

The average search cost per call in the mobile PNNI scheme $\overline{S_{\text{PNNI}}}$ is given by (10). Since the search cost is 0 for $a_{hv} \geq S$, the first (ith index) summation is from 1 to $S - 1$. The first and the second additive terms are due to the third and fourth cases of (8). In the third case, $a_{hv} < a_{ch}$. Hence, the jth-

$$S_{\text{PNNI}} = \begin{cases} 0, & a_{hv} \geq S \quad \text{independent of } a_{cv} \\ 0, & a_{cv} \geq S \quad \text{independent of } a_{hv} \\ D_{ch}, & a_{hv} < S \quad a_{cv} < S \quad a_{hv} < a_{ch} \\ D_{hv}, & a_{hv} < S \quad a_{cv} < S \quad a_{ch} < a_{hv} \\ 2D_{ch} - D_{cv}, & a_{hv} < S \quad a_{cv} < S \quad a_{ch} < S \quad a_{ch} = a_{hv} \end{cases} \quad (8)$$

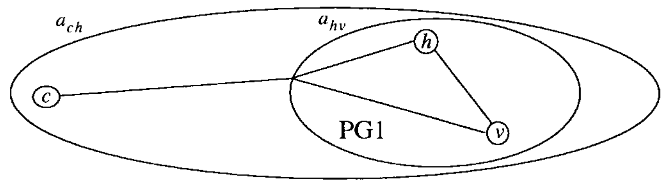


Fig. 13. Distances between the calling party and the home and visiting locations of the called mobile.

index summation varies a_{ch} from $i+1$ to L. The condition that $a_{cv} < S$ is automatically satisfied since $a_{cv} = a_{hv}$ as per (1), and $a_{hv} < S$. Similarly, the fourth case of (8) requires only the probability distributions of a_{hv} and a_{ch}. In this case, a_{ch} should be varied from 1 to $i-1$ since $a_{ch} < a_{hv}$. The last term of (10) originates from the second and fifth cases of (8). In the previous two terms, since the value of a_{cv} is determined from the values of a_{hv} and a_{ch}, and these latter values were less than S, the condition $a_{cv} \geq S$ is always false. However, if $a_{hv} = a_{ch}$, $a_{cv} \geq a_{hv}$, allowing a_{cv} to vary from i, where $a_{hv} = a_{ch} = i$, to L. The second case of (8) shows that if $a_{cv} \geq S$, the search cost is 0. This implies that a_{cv} should only be varied from i to $S-1$, as indicated in the last term in (10). Expressions for the probabilities in (10) are derived in the Appendix

$$\begin{aligned}\overline{S_{\mathrm{PNNI}}} &= \sum_{i=1}^{S-1} P(a_{hv}=i)\Bigg\{\sum_{j=i+1}^{L} P(a_{ch}=j)D_{ch} \\ &\quad + \sum_{j=1}^{i-1} P(a_{ch}=j)D_{hv} + P(a_{ch}=i) \\ &\quad \cdot \Bigg\{\sum_{j=i}^{S-1} P(a_{cv}=j|a_{ch}=a_{hv}=i)(2D_{ch}-D_{cv})\Bigg\}\Bigg\}. \qquad (10)\end{aligned}$$

2) Search Cost in the LR Scheme: In the LR scheme, since there is an explicit mobile location phase, the search cost is the cost of sending LOCREQ's (location requests) up and down the chain of LR's while locating a mobile. This search cost is given below:

$$S_{\mathrm{LR}} = \begin{cases} R_{a_{cv}} + R_{a_{cv}+1} + 1, & a_{cv} \geq S \quad \text{independent of } a_{hv} \\ 2R_S + 2h + 1, & a_{cv} < S \quad a_{hv} \geq S \\ 2R_S + 1 + 2h, & a_{cv} < S \quad a_{hv} < S \quad a_{ch} \geq S \\ 2R_S + 3h, & a_{cv} < S \quad a_{hv} < S \quad a_{ch} < S \end{cases} \qquad (11)$$

where R_i is defined in (5).

When the calling party is located in the neighborhood of the called party [case 1 of (11)], search cost consists of the cost of querying LR's from the level L up to level a_{cv}, and then down from the LR at level $a_{cv}+1$ to the level-L LR of the called mobile's switch. These two costs correspond to the terms $R_{a_{cv}}$ and $R_{a_{cv}+1}$, respectively. The cost of sending the final response directly from the level-L LR to the calling party's switch, as shown in Fig. 10, is 1 since the two ends of this response message are within the same neighborhood.

When the calling party is outside the neighborhood of the called party's current location [cases 2, 3, and 4 of (11)], the relative location of the home address of the mobile and its current location become relevant. If it is in its home neighborhood (case 2), then the search propagates up to the level-S LR from the calling party (at a cost R_S), which then sends a message to the home LR of the mobile (at a cost h since the home LR of the called mobile is outside the neighborhood of the calling party). The home LR sends a location request to the Sth level LR currently tracking the mobile at a cost of one unit (since both of these LR's are in the same neighborhood). This is followed by a set of location requests which follow the trace of the pointers from the LR at level S to level L (at a cost R_S). The final reply costs h units since the LR at the visiting location of the mobile is outside the neighborhood of the calling party. Costs shown in cases 3 and 4 of (11) can be similarly reasoned.

The average search cost per call in the LR scheme ($\overline{S_{\mathrm{LR}}}$) is given by (12). The first two terms correspond to the first two cases, and the fifth term corresponds to the third case of (11). The third and the fourth terms represent the fourth case of (11):

$$\begin{aligned}\overline{S_{\mathrm{LR}}} &= \sum_{i=S}^{L} P(a_{cv}=i)(R_{a_{cv}} + R_{a_{cv}+1} + 1) \\ &\quad + \sum_{i=1}^{S-1} P(a_{cv}=i)\Bigg\{\sum_{j=S}^{L} P(a_{hv}=j)(2R_S+2h+1) \\ &\quad + \sum_{j=1,\, j\neq i}^{S-1} P(a_{hv}=j)(2R_S+3h) + P(a_{hv}=i) \\ &\quad \cdot \Bigg\{\sum_{j=i}^{S-1} P(a_{ch}=j|a_{hv}=a_{cv}=i)(2R_S+3h) \\ &\quad + \sum_{j=S}^{L} P(a_{ch}=j|a_{cv}=a_{hv}=i)(2R_S+1+2h)\Bigg\}\Bigg\}. \qquad (12)\end{aligned}$$

D. Average Total Costs

The total cost of a location management scheme depends on the move cost and the search cost of that scheme. In order to be able to estimate the average total cost, the rate of call arrival at a mobile λ_c and the rate at which the mobile moves between base stations λ_m are needed. The average move cost per unit time, average search cost per unit time, and the average total cost per unit time are given in (13), where $\overline{M}$ and $\overline{S}$ represent the average move cost and average search cost per call, respectively:

$$\hat{M} = \lambda_m \overline{M} \qquad \hat{S} = \lambda_c \overline{S} \qquad T = \hat{M} + \hat{S}. \qquad (13)$$

Since we do not have exact numbers for λ_c and λ_m, but are interested in the impact of these parameters, we use the CMR (call-to-mobility ratio), denoted ρ (defined as the number of calls arrivals per move) [1], and quantify the dependence of the average total costs on the CMR. The average total cost per

TABLE IV
INPUT PARAMETERS

Parameter	Value
m_i, $1 \le i \le L$	4
p_i, $1 \le i \le L$	2
c_i, $1 \le i \le L$	1
N_b	37
L	10

move T^m for the two schemes is given by

$$T^m_{\rm PNNI} = \overline{M_{\rm PNNI}} + \rho\overline{S_{\rm PNNI}} \qquad T^m_{\rm LR} = \overline{M_{\rm LR}} + \rho\overline{S_{\rm LR}} \quad (14)$$

where $\rho = \lambda_c/\lambda_m$, and $\overline{M_{\rm PNNI}}$, $\overline{M_{\rm LR}}$, $\overline{S_{\rm PNNI}}$, and $\overline{S_{\rm LR}}$ are given in (4), (6), (10), and (12), respectively.

E. Numerical Results

In this section, we quantitatively compare the mobile PNNI scheme described in Section III-A, and the LR scheme described in Section III-B. The measures of comparison include the average move cost per mobile (derived in Section IV-B), the average search cost per call (derived in Section IV-C), and the average total cost per move (derived in Section IV-D). This analysis also *provides insights into the effect of key parameters of these algorithms*, such as S, the reachability update limiting scope, which is also the highest level of the hierarchy of location registers in the LR scheme, h, the cost of "long-distance" signaling, and CMR (call-to-mobility ratio).

Input Data: Values of input parameters assumed for this numerical computation are shown in Table IV (see Table I for definitions of these parameters). We observe that the exact numerical results are dependent on the exact values chosen for these parameters. However, the trends observed are more or less independent of these values. Sensitivity of the comparative results to these input parameters has been studied, but not included in this paper due to space considerations.

1) Comparison of the Two Schemes: Plots of (14), showing the variation of average total cost in the two schemes with CMR, are given in Fig. 14. This figure shows that the mobile PNNI scheme incurs a lower average total cost at high CMR's, while at low CMR's, the LR scheme performs better. These plots depend on the value of S, which can potentially be different in the two schemes. For example, if the CMR is 0.02, the LR scheme, when operated with $S = L - 1$, gives the lowest average total cost. But if CMR is 0.03, the mobile PNNI scheme should be chosen and operated with $S = L + 1$. We observe that the CMR at which the mobile PNNI scheme does better than the LR scheme is at 0.025 (we designate this the "breakpoint CMR").

A second level of comparison is to understand the behavior of the two schemes relative to increasing CMR and S. Since S is a parameter of the two algorithms, these results provide significant insight for the selection of this parameter. To study the effect of S on the average total costs of the two schemes, consider the plots shown in Fig. 15. The mobile PNNI plots demonstrate that at higher CMR's, a low value of S should

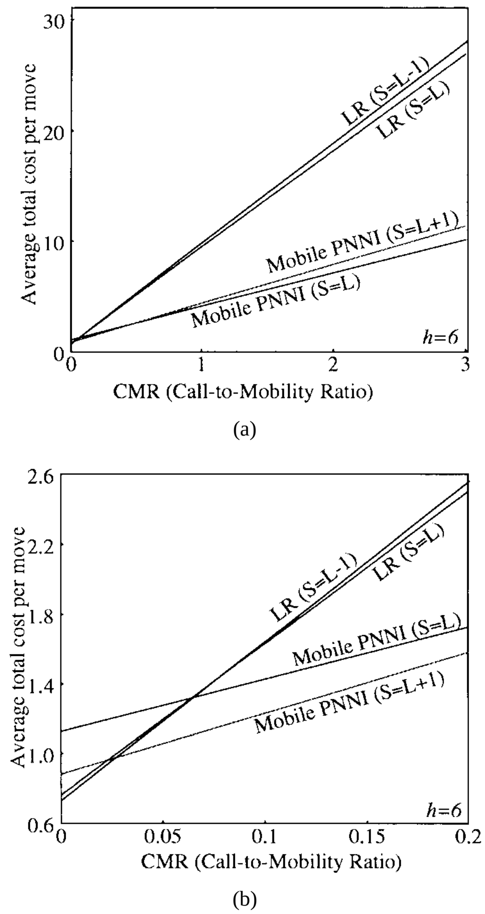


Fig. 14. Comparison of the average total costs of the two schemes.

be selected, while at low CMR's, high values of S should be chosen. This is illustrated in Fig. 16. The opposite behavior is seen in the plots for the LR scheme.

In the mobile PNNI scheme, the move costs increases with a decrease in S since reachability updates have to propagate to a wider area. Thus, if the CMR is low, a high value of S should be chosen to limit the move costs. On the other hand, the search cost increases with an increase in S since more calls are likely to be misrouted as the neighborhood containing the reachability information is small. For high CMR's, a low S should be chosen to limit the search costs. Fig. 15(a) for the mobile PNNI scheme shows that the $S = L + 1$ plot offers the lowest average total cost at very low CMR's (0–0.49), the $S = L$ plot becomes the best (minimum average total cost) solution for the next range of CMR's (0.49–2.55), the $S = L - 1$ plot becomes the best for CMR's ranging from 2.55 to 6.13, and the trend continues.

A similar behavior is observed in the plots for the LR scheme, shown in Fig. 15(b), with the exception that the trends are in the opposite direction. In other words, the LR scheme executed at lower values of S tends to perform better at lower CMR's, and at higher values of S, it tends to perform better at higher CMR's. The LR scheme plots in Fig. 15(b) show that for CMR's below 0.053, the $S = L - 1$ choice is a better one, while for CMR's above this value, the $S = L$ choice is better (this point has been chosen to illustrate the trend, although it is higher that the breakpoint CMR). Unlike the

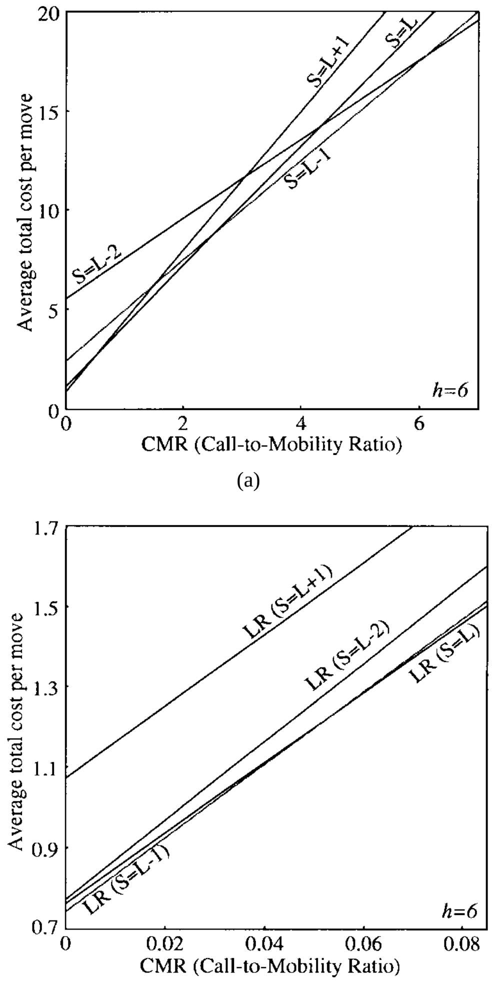


Fig. 15. Effect of S on the average total costs in the two schemes: (a) mobile PNNI scheme and (b) LR scheme.

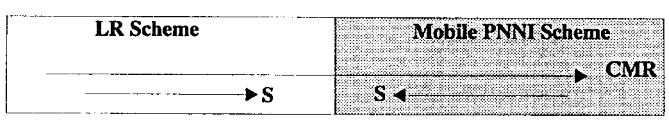


Fig. 16. Contrasting behavior of the two schemes with varying S and CMR.

mobile PNNI scheme, as CMR increases, larger values of S should be chosen in the LR scheme.

Interestingly, in the LR scheme plots, for some values of S, there does not exist a range of values of CMR where operating with that value of S gives the minimum average total cost. For example in Fig. 15(b) plots, $S = L-2$ and the $S = L+1$ plots incur higher costs at all values of CMR. This is explained by the effect of the parameter h, an important parameter excluded from the above discussion. Next, we address the effect of this parameter.

2) Effect of Key Parameters h and S:

a) Variation of average costs with changing h for different values of S: The value of h affects the average move cost in both schemes [see (4) and (6)], and the average search cost in the LR scheme [see (12)], as seen in the plots shown in Fig. 17. It does not effect the search cost in the mobile PNNI scheme [see (10)]. For large values of S, the move cost in both schemes is dominated by the cost of setting forwarding pointers (in the mobile PNNI scheme) or updating the home/old LR (in the LR scheme), and hence, h becomes a more significant parameter (the average move plots corresponding to $S = L+1$ in both schemes have a steep slope). If $S = 1$, there are no updates to the home LR of a mobile as a user moves since the Sth level LR never changes (there being only one Sth level LR). Searches also do not require location queries to distant location registers since every location query will be resolved during the upward propagation of queries. Hence, in Fig. 17, the plots corresponding to $S = 1$, for the LR scheme, are flat.

b) Variation of average costs with changing S for different values of h: From the plots in Fig. 17, we can make observations about the effect of S on the average costs in the two schemes at different values of h (note that CMR is not involved in these plots since they show the average move and search costs, and not the average total costs). Instead of selecting specific values of h and showing the variation of the average costs with respect to S, we show that at different ranges of h, the average move and search costs in the two schemes follow certain trends. These trends are shown for the two schemes in Figs. 18 and 19 (arrows pointing upward indicate an increase in cost). These results provide important information in helping us select values of S for a given value of h.

For the mobile PNNI scheme, Fig. 18 shows that at "low and medium values" of h, the average move cost decreases with increasing S. This is observed in Fig. 17, which shows that up to $h = 7.5$, the average move cost incurred is consistently higher for lower values of S. However, at high values of h, Fig. 18 shows that the average move cost decreases up to a value, and then increases as S increases. This is seen in the mobile PNNI average move plot of Fig. 17 where, for example, if $h = 12$, changing S from $L-1$ to L to $L+1$ causes the cost to first drop and then increase. The mobile PNNI search cost simply increases with S irrespective of h (seen from Fig. 17 and shown in Fig. 18).

For the mobile PNNI scheme at *high values of h*, the move cost decreases with increase in S up to a value $S_{\max}$, as shown in Fig. 18, beyond which it increases. Since the mobile PNNI scheme search cost increases monotonically with increasing S, to minimize the average total cost, for the mobile PNNI scheme, the values of S should be chosen such that $S \leq S_{\max}$. Such a statement cannot be made if h is in the *low or medium ranges* since the average move cost decreases with increasing S, while the average search cost increases with increasing S. The optimal value for S is then determined by the value of CMR. For example, Fig. 15 showed the variation of the average total costs with varying CMR for different values of S at an operating point where h is in the medium range ($h = 6$).

The LR costs show a slightly different trend. At "low" values of h, the LR scheme experiences decreasing average move and search costs with increasing S, at "high" values of h, both costs increase with increasing S, and at "medium" values of h, both costs first decrease and then increase, as shown in Fig. 19. This is seen in the LR scheme plots of

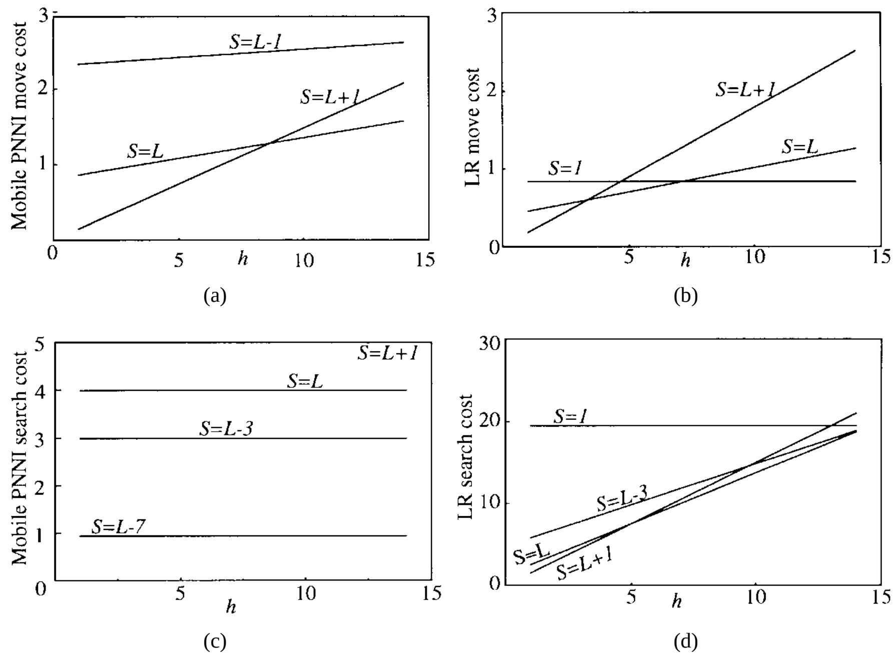


Fig. 17. Effect of h on the average costs in the two schemes.

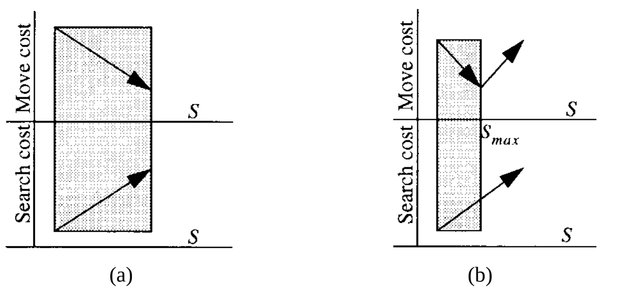


Fig. 18. Effect of change in S on the average costs in the mobile PNNI scheme. (a) Low and medium values of h. (b) High values of h.

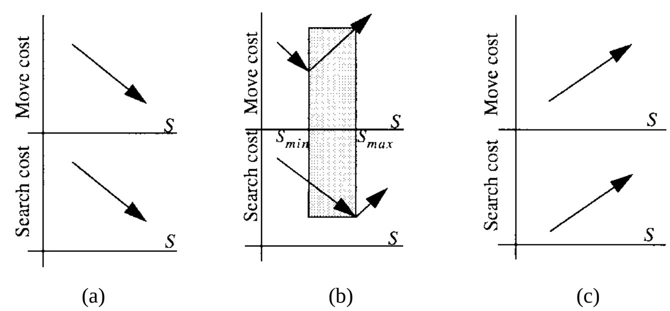


Fig. 19. Effect of change in value of S on average costs in the LR scheme. (a) Low values of h. (b) Medium values of h. (c) High values of h.

Fig. 17. The reason for such behavior is that, if h is small, the LR scheme should be operated more like the "flat" scheme of Section II by choosing a large S. In other words, all nodes know the home LR's of mobiles, and directly send registrations and location queries to the home LR's. At large values of h, the LR scheme should be operated more like the "hierarchical" scheme of Section II, by making S equal to 1 since the best result (lowest average total cost) is obtained at the smallest value of S. For the medium range of h (such as $h = 6$, for which we provided detailed plots in Fig. 15), the optimal value of S depends upon the CMR. In this range, the average move cost decreases with increasing S up to $S_{\min}$, and the search cost decreases up to a value of $S = S_{\max}$. The minimum value of average total cost is obtained for some value of S that lies between $S_{\min}$ and $S_{\max}$. For example, in Fig. 15(b), $S_{\min} = L - 1$ and $S_{\max} = L$.

In summary, there are three important parameters: S, the reachability update limiting scope, which is also the highest level of hierarchy in the LR scheme, h, the cost of "long-distance" signaling, and CMR (call-to-mobility ratio). As to which location management scheme incurs a lower average total cost, the mobile PNNI scheme or the LR scheme, depends on these three parameters. Typically, at *low values* of h, operating the LR scheme at high values of S leads to a lower average total cost than the PNNI scheme because, at high values of S, while the mobile PNNI scheme does not incur a move cost, it does incur a search cost, while in the LR scheme, by virtue of h being low, both move and search costs are small. On the other hand, if the cost of *long-distance signaling h is high*, either the LR scheme or the mobile PNNI scheme could lead to minimal average total cost, provided the correct value of S is selected. For the mobile PNNI scheme, this depends on the CMR expected, but in the LR scheme, one needs to select a low S (preferably $S = 1$). Finally, if *h is in the medium range* (which we expect will be the range of operation), there will be a breakpoint CMR below which the LR scheme will perform better, and above which the PNNI mobile scheme will incur lower costs. A significant point to note is that, for a number of cases, for example, when mobiles are located close to their home locations (which we expect will be a high percentage), or if the calling party is close to the visiting location of the called mobile, the mobile PNNI scheme incurs a zero search cost. This leads to the mobile PNNI scheme performing better in most regions of operation expected in low-tier (i.e., slowly moving users) PCS applications.

V. CONCLUSIONS

This paper presented two mobile location management algorithms for ATM networks based on the PNNI (private network-to-network interface) standard. The first solution is called the *mobile PNNI scheme* because it builds on the PNNI routing protocol. It uses limited-scope (characterized by a parameter S) reachability updates, forwarding pointers (setting and clearing of these pointers occur at a cost h), and a route optimization procedure. The second solution is called the *LR (location registers) scheme* because it introduces location registers (such as the cellular home and visitor location registers) into the PNNI standards-based hierarchical networks. This scheme uses a hierarchical arrangement of location registers with the hierarchy limited to a certain level S. It also requires the update of home and old location registers at a cost h.

Analytical models were set up to compare the average move, search, and total costs per move of these two schemes for different values of the CMR (call-to-mobility ratio), and to provide guidelines for selecting the critical parameters of the algorithms. Results showed that at low CMR's (CMR < 0.025), the LR scheme performs better than the mobile PNNI scheme. We also observed that the two schemes show a contrasting behavior in terms of the value to be used for the parameter S to achieve the least average total cost. For the mobile PNNI scheme, the parameter S should be high at low CMR's (within the range in which the mobile PNNI scheme should be used), and low at high CMR's. However, in the LR scheme, the parameter should be low at low CMR's (within the range in which the LR scheme should be used), and high for high CMR's. These observations are made for a region of operation in which h, the cost of setting forwarding pointers, and updating distant LR's, is of medium value. If this cost is low, the LR scheme always outperforms the mobile PNNI scheme. For other ranges of h, the scheme selected, and the S at which the scheme is operated, depend on the CMR.

APPENDIX

In this section, we describe the probability distributions $P(a_{cv}), P(a_{ch}), P(a_{hv}), P(a_{hn}), P(a_{ho}), P(a_{on})$ and the conditional probabilities used to compute the average costs in (4), (6), (10), and (12).

We assume that the distribution of calls to a mobile follows a uniform distribution, i.e., it is equally probable for a mobile to receive a call originating at any switch. So the probability of a call originating from a switch for which $a_{cv} = x$ depends on the number of nodes, for which $a_{cv} = x$, and the total number of nodes in the network. Thus, the probability $P(a_{cv} = x)$ is given by

$$P(a_{cv} = x) = \frac{\left(\prod_{i=x+1}^{L} (m_i + 1)\right) m_x}{\prod_{i=1}^{L} (m_i + 1)}. \quad (15)$$

This distribution makes long-distance calls very highly probable. For example, $P(a_{cv} = 1) \approx 0.8$ for $L = 10, m_i = 4$. This, in effect, makes our cost estimates pessimistic. Since a call can originate from anywhere, the distribution of a_{ch} is the same as the distribution of a_{cv}.

Next, we consider the distribution of a_{hv}. The choice of this distribution is based on the premise that a majority of the mobiles roam in and around their respective homes. We choose a distribution such that the probability of being located close to home is high. The distribution is such that the probability of being located at the home switch is f, in the Lth level peer group of the home switch is f^2, in the $(L-1)$th peer group is f^3, etc. Given that a mobile is located somewhere in the network,

$$f + f^2 + f^3 + \cdots + f^{L+1} = 1 \quad \text{or} \quad \frac{1 - f^{L+1}}{1 - f} = 2. \quad (16)$$

For $L = 8$, $f \approx 0.5$. This can be interpreted as 50% of users being in their home registration area. This is also somewhat pessimistic since the majority of the users are typically in and around their home location. Under this assumption, the distribution of a_{hv} is given by (17):

$$P(a_{hv} = x) = f^{L-x+2}. \quad (17)$$

The probability distributions of a_{oh} and a_{hn} are assumed to be the same as the distribution of a_{hv}.

In order to determine the probability distribution of a_{on}, we assume that the base stations under a peer node of a peer group are arranged in a hexagonal fashion. We model each peer node as a macrocell in which all of the base stations under the peer node are arranged in a hexagonal fashion. The results from [21] are used to approximate $P(a_{on})$. Base stations in a macrocell are arranged in layers, with the jth layer base stations arranged around the $j-1$th layer base stations. A macrocell of i layers with all of its base stations arranged in a hexagonal fashion has $3i^2 - 3i + 1$ base stations in it. The layers are numbered 0 through $i-1$, with layer j $(j > 0)$ having $6j$ base stations in it [21]. For example, a three-layer macrocell consists of 19 cells, with layer 0, layer 1, and layer 2 having 1, 6, and 12 base stations, respectively. The probability that a mobile in an $i-1$th layer base station moves to an ith layer base station $P(i-1 \rightarrow i)$ (i.e., out of the macrocell since a layer-i macrocell has layers numbered 0 to $i-1$) is given by (18) [21]:

$$P(i-1 \rightarrow i) = \frac{2((i-1)+1)}{6(i-1)}. \quad (18)$$

Let $N(k)$ be the number of base stations in a peer node of level k. $N(k)$ is given by (19), where N_b is the number of base stations under a switch (peer group of level $L+1$):

$$N(k) = \left(\prod_{i=k+1}^{L} (m_i + 1)\right) N_b. \quad (19)$$

The total number of base stations in a k-level macrocell is also given by $3 \times [I(k)]^2 - [3 \times I(k)] + 1$ if we arrange the $N(k)$ base stations in this macrocell as a hexagon of $I(k)$ layers. Thus,

$$3 \times (I(k))^2 - (3 \times I(k)) + 1 = N(k). \quad (20)$$

Solving (20) for its positive root, we obtain $I(k)$:

$$I(k) = \frac{3 + \sqrt{9 + 12(N(k) - 1)}}{6}. \tag{21}$$

"Border" base stations are base stations that lie on the boundary of the macrocell (i.e., base stations from which a user can move to other registration areas). Let $B(k)$ denote the number of border base stations in a level-k peer group. Since there are $I(k)$ layers in a level-k peer node and the layers are numbered 0 to $I(k) - 1$, the border base stations of a level-k peer node are the base stations in level $I(k) - 1$. Thus, there are $6[I(k) - 1]$ border base stations for a level-k peer node, as shown in (22)

$$B(k) = 6(I(k) - 1). \tag{22}$$

The probability of moving out of a level-k peer group $P(k)$ can be approximated as

$$P(k) = \frac{B(k)}{N(k)} \left(P(I(k) - 1) \rightarrow I(k)\right)). \tag{23}$$

Assuming that each level-k peer node contributes equally to the border base stations of level-$(k-1)$ peer node, the probability distribution of a_{on} is obtained from $P(k)$ and $P(k-1)$, and is given in (24):

$$P(a_{on} = k) = P(k) - P(k-1). \tag{24}$$

The conditional probability $P(a_{ch} = j | a_{cv} = a_{hv} = k)$ is given by (25):

$$P(a_{ch} = j | a_{cv} = a_{hv} = k) = \frac{\left(\prod_{i=j+1}^{L} (m_i + 1)\right) m_j}{\prod_{i=k}^{L} (m_i + 1)}. \tag{25}$$

The conditional probability $P(a_{oh} = j | a_{nh} = a_{on} = i)$ is given by (26):

$$P(a_{oh} = j | a_{nh} = a_{on} = i) = \frac{P(a_{oh} = j)}{\sum_{k=i}^{L} P(a_{nh} = k)}. \tag{26}$$

ACKNOWLEDGMENT

The authors would like to thank R. Caceres, P. Pancha, and the anonymous reviewers for their suggestions on this work.

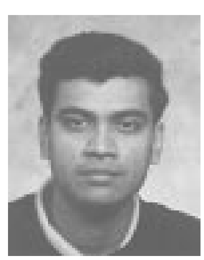
REFERENCES

[1] R. Jain and Y.-B. Lin, "An auxiliary user location strategy employing forwarding pointers to reduce network impacts of PCS," *ACM/Baltzer Wireless Networks J.*, vol. 1, pp. 197–210, July 1995.
[2] B. Auwerbuch and D. Peleg, "Online tracking of mobile users," *J. Assoc. Comput. Mach.*, pp. 1021–1058, Sept. 1995.
[3] EIA/TIA IS-41 Rev. C, *Cellular Radio Telecommunications Intersystem Operations*," TIA/EIA PN-2991, Nov. 1995.
[4] M. Mouly and M. B. Pautet, "The GSM system for mobile communications," 49 rue Louise Bruneau, Palaiseau, France, 1992.
[5] ATM Forum Technical Committee, *Private Network-Network Specification Interface v1.0 (PNNI 1.0)*, af-pnni-0055.000, Mar. 1996.
[6] C. Perkins, "IP mobility support," RFC 2002, Oct. 1996.
[7] D. B. Johnson and C. Perkins, "Mobility support in IPv6," Internet Draft, draft-ietf-mobileip-ipv6-02.txt, Nov. 1996, work in progress.
[8] J. S. M. Ho and I. F. Akylidiz, "Local anchor scheme for reducing location tracking costs in PCN's," *IEEE/ACM Trans. Networking*, vol. 4, pp. 709–725, Oct. 1996.
[9] M. Veeraraghavan, T. F. La Porta, and R. Ramjee, "A distributed control strategy for wireless ATM networks," *ACM/Baltzer Wireless Networks J.*, pp. 323–339, 1995, short form in *Proc. ICC'95.*
[10] J. Z. Wang, "A fully distributed location registration strategy for universal personal communication systems," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 850–860, Aug. 1993.
[11] R. Jain, "Reducing traffic impacts of PCS using hierarchical user location databases," in *Proc. IEEE ICC'96*, Dallas, TX, pp. 1153–1157.
[12] C. Eynard, M. Lenti, A. Lombardo, O. Marengo, and S. Palazzo, "A methodology for the performance evaluation of data query strategies in universal mobile telecommunication systems (UMTS)," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 893–907, June 1995.
[13] D. B. Johnson and C. Perkins, "Route optimization in mobile IP," Internet-Draft, draft-ietf-mobileip-optim-04.txt, Feb. 1996, work in progress.
[14] ATM Forum Technical Committee, *ATM User-Network Interface (UNI) Signaling Specification Version 4.0*, ATM Forum/95-1434R9, Jan. 1996.
[15] M. Veeraraghavan, P. Pancha, and G. Dommety, "Connectionless ATM using an ATM switch router," in *Proc. ECMAST'97*, Milan, Italy, May 1997.
[16] J. Porter and D. Gilmurray, *Tunneled Signaling for the Support of Mobile ATM*, ATM Forum Contribution, ATM Forum/96-1699, San Diego, CA, Dec. 1996.
[17] G. Dommety, M. Veeraraghavan, and M. Singhal, "Route optimization in mobile ATM networks," Comput. Inform. Sci., Ohio State Univ., Tech. Rep.
[18] C.-K. Toh, "Performance evaluation of crossover switch discovery algorithms for wireless ATM LAN's," in *Proc. IEEE INFOCOM'96*, San Francisco, CA, pp. 1380–1387.
[19] M. Veeraraghavan, M. Karol, and K. Y. Eng, "Mobility and connection management in a wireless ATM LAN," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 50–68, Jan. 1997.
[20] G. P. Pollini and K. S. Meier-Hellstern, "Efficient routing of information between interconnected cellular mobile switching centers," *IEEE/ACM Trans. Networking*, vol. 3, pp. 765–774, Dec. 1995.
[21] Y.-B. Lin, L.-F. Chang, and A. Noerpel, "Modeling hierarchical microcell/macrocell PCS architecture," Tech. Rep., PCS-NCTU-96-05 (http://liny.csie.nctu.edu.tw).

Malathi Veeraraghavan (M'88) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Madras, in 1984, and the M.S. and Ph.D. degrees in electrical engineering from Duke University, Durham, NC, in 1985 and 1988, respectively.

She is currently a Distinguished Member of Technical Staff at Bell Laboratories, Lucent Technologies, in the Networking Research Laboratory, Holmdel, NJ. Her research interests include signaling and control of networks, and network and mobility management.

Dr. Veeraraghavan served as an Associate Editor of the IEEE TRANSACTIONS ON RELIABILITY from 1992 to 1994.



Gopal Dommety received the B.Tech. degree (with honors) in computer science and engineering from the Indian Institute of Technology, Kharagpur, in 1993, and the M.S. degree in computer and information science from The Ohio State University, Columbus.

He is currently a Ph.D. candidate in the Department of Computer and Information Science, Ohio State University. His research interests include mobility management, internetworking and interoperability, performance analysis, and parallel and distributed systems.

Mr. Dommety is a student member of the ACM.