Quality of Service Guarantees in Mobile Computing

Suresh Singh

Department of Computer Science
University of South Carolina
Columbia, SC 29208
email: singh@cs.scarolina.edu

Keywords: Mobile Computing, Transport Layer, Quality of Service

Abstract

With rapid technological advances being made in the area of wireless communications it is expected that, in the near future, *mobile users* will be able to access a wide variety of services that will be made available over future high-speed networks. The quality of these services in the high-speed network domain can be specified in terms of several QOS parameters.

In this paper we identify *two* new QOS parameters unique to the mobile environment – guarantee of *seamless* service and ensuring *graceful degradation* of service in situations where user demands exceed the network's capacity to satisfy them. A network architecture and a suite of transport level services are proposed that enables these QOS parameters to be satisfied.

To appear in J. Computer Communications

1 Introduction

Mobile computing refers to an emerging new computing environment incorporating both wireless and wired high-speed networking technologies. In the near future, it is expected that millions of users equipped with *personal digital assistants* (palm-top computers with wireless communications technology) will have access to a wide variety of services that will be made available over national and international communication networks. Mobile users will be able to access their data and other services such as electronic mail, electronic news including special services such as stock market news, videotelephony, yellow pages, map services, electronic banking, etc., while on the move.

Providing such a diverse number of services to mobile users requires high data rates and several authors (see for example [9, 10, 12]) have proposed an average data rate of between 1–2Mbps per mobile user. In order to support such high data rates, a *microcellular* network architecture has been proposed, see [9, 5]. Here a geographical region such as a campus is divided into *microcells* with a diameter of the order of hundreds of meters. All mobile users within a microcell communicate with a central host machine within that cell who serves as a gateway to the wired networks; this machine is called a *mobile support station* (MSS).

What are some of the networking issues related to providing the different types of service discussed above? Some of the broad issues that need to be considered are the design of efficient network architectures to support mobility, protocols to provide uninterrupted service to mobile users, maintaining quality of service guarantees for applications and in some cases renegotiating these guarantees during the lifetime of a connection. Many of these issues have been studied for high-speed networks and in many cases good solutions exist. Unfortunately, most of these solutions cannot be applied to the mobile computing environment because of its inherent properties.

To illustrate some of the unique features of the mobile computing environment, consider a situation where several mobile users have opened high-bandwidth data connections. When these data connections are set up, the network ensures that the users receive some guaranteed quality of service (such as max. loss probability, bounded delay, etc.). Since these users are all mobile, it is possible that many of them could move into the same cell. In such a situation, it is very likely that the available bandwidth of the cell will be exceeded resulting in the original QOS (quality of service) parameters being violated. This situation does not arise in high-speed networks because users are not mobile during the life-time of a connection.

Maintaining *seamless* (i.e., uninterrupted) connections is another problem in mobile environments. As users move rapidly between cells, the data has to be forwarded by the previous mobile support station to the new mobile support station resulting in brief *blackout* periods. For many applications (such as videotelephony) these blackout periods are quite unacceptable particularly if they occur frequently. For a mobile user in a car that is moving at 30-40 mph, these blackouts will occur every 5-10 sec (assuming a microcell size between 100 and 200 meters) and may last for as long as half a second (see [6, 7]).

In this paper, we identify several fundamental open problems related to networking in the mobile environment and propose approaches to solving these problems. These important problems arise mainly because of the *mobility of the users* and the *unpredictability of their motion.* An implication of this is that solutions to similar problems in the high-speed networking domain cannot be carried over to mobile computing.

1.1 Overview of the Paper

The ultimate goal of the mobile computing environment is to provide users access to a wide variety of services that will be made available over future high-speed networks. Thus, the design of the mobile network must be based on user needs specified in terms of quality of service requirements. In section

2 we identify QOS parameters unique to the mobile environment and discuss implications for network design. A proposal for a network architecture is presented in section 3 and transport layer support for providing the QOS requirements is discussed in section 4. Our conclusions are presented in section 5.

2 QOS Parameters

In order to provide an acceptable level of service to users many applications, running over networks, require some minimum grade of service from the underlying network. Furthermore, these network requirements vary from application to application. In the high-speed network domain most of these requirements can be specified as some combination of *quality of service* (QOS) parameters such as: delay & jitter bounds, minimum & maximum bandwidth requirements and maximum loss bound, etc. For example, a video-on-demand application requires data (i.e., the audio and video portions of the movie) to be delivered to the user with low loss and low jitter.

It is clear that these QOS parameters can be used to specify grades of service in the mobile environment as well. However, the fact that users are mobile and unpredictable, results in situations where these QOS parameters do not suffice. In this section we identify *two* additional QOS parameters that are essential to specifying grades of service for mobile users.

2.1 Graceful Degradation of Service

An important problem that is uniquely a feature of the mobile computing environment is the problem of service degradation. This problem is caused, as noted in Section 1, when many users with open connections enter a cell and the total requested bandwidth exceeds the cell's capacity. In this context we need to answer a fundamental question: how do we allocate the limited bandwidth among all the users? This problem becomes particularly knotty when the services being used by the conflicting users have different characteristics.

One scheme is to prioritize all the open connections and then penalize the least priority connections first. Unfortunately it is not always possible to strictly rank-order all the different applications (or services) that different users are currently using. Furthermore, a user with all 'low priority' connections may lose all connections! This is clearly not desirable. In terms of policy, should we reduce the quality of service for all users even if that is not necessary? In other cases, should we try to support as many users as possible at the cost of degraded service quality or should we support a few users only, while providing them with the best possible service?

2.1.1 QOS Parameter: *Loss Profile*

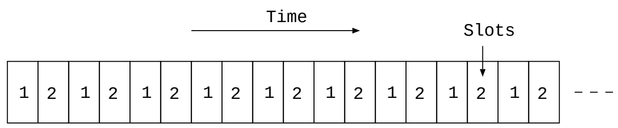
Our approach to the problem is to require users to specify, during connection setup, a preferred way in which data can be discarded in the event that bandwidth requirements within a cell exceed the available bandwidth. This specification is called a *loss profile* and is used, in conjunction with other QOS parameters, to allocate bandwidth to different mobile users within a cell.

To explain this idea better let us consider an example. Let us assume that the multiaccess protocol within a cell is a simple form of TDMA[1]. Let us further assume that the bandwidth of a channel is 32kbps (as in [10]) and that the cell currently has two mobile users, M1 and M2, each with an open constant bit-rate connection (see [3] and section 4) operating at 16kbps. Initially, M1 and M2 receive

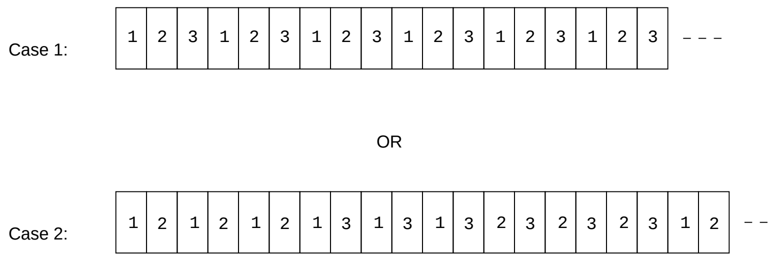
[1]The reason for this choice is that most proposed standards for communication within a cell such as, the European standard (GSM), the North American standard (IS-54) and another Pan-European standard (DECT) use TDMA (see [10]). A proposal for third generation mobile networks is the PRMA protocol, that uses a combination of TDMA and ALOHA (see [8]).

data from the MSS in alternate slots as shown in Figure 1(a). Let us assume that a third mobile user M3, also with an open constant bit-rate connection, enters the cell. If this user's connection is also a 16kbps connection then consider two possible allocations of slots as shown in Figure 1(b). Figure 2 illustrates the arrival process at M1. As we can see in Figure 2, in Case 1, every third chunk of data of s bytes (TDMA slot size) is lost and delays are increased while maintaining a smaller but constant bit rate. In Case 2, data is lost in bursts and the bit rate is unchanged when M1 is not losing data.

If the application being used by the mobile user M1 is audio then perhaps Case 1 (Figure 1(b)) is preferable. If, however, M1 is receiving compressed video then this type of loss is quite unacceptable[2] and a bursty loss, Case 2 in Figure 1(b), is preferred. This is because the reconstructed video in this case will consist of several seconds of clear video signal followed by a period of outage.



(a) Initial Channel Allocation.



(b) Two Possible Channel Allocations.

Figure 1: Slot Assignment.

Note that this type of a situation does not arise, in most cases, for high-speed networks because the loss rates there are typically far below 10^{-5}. By comparison the loss rate in our example above is 33%. What does all this mean from an implementation standpoint? There is a clear need to require service-providers to specify *acceptable loss behavior* (or **loss profile**) for their respective services and to have channel allocation policies that try to follow this type of loss behavior (when channel capacity is exceeded). These loss profiles may, in general, be specified in terms of distributions or, in the case of constant bit-rate connections, in a deterministic manner (in Case 1 above, for M1, the loss profile specifies that losses be evenly spaced in the data stream).

2.2 Seamless Communication

Recall the car example from the introduction. A rapidly moving user has a small *latency* in each cell (staying time) and as a result may see periodic breaks in service. This situation may be unacceptable

[2] A subband coded compressed image (using, for example, JPEG or the wavelet transform) may be viewed as a 2D matrix of integer values with a large percentage of 0s. The data stream can be viewed as being composed of two major parts – a *runlength* that indicates the positions of non-zero coefficients and the coefficient values themselves. If coefficient values are lost, the reconstructed image gets blurred but if the runlength (or part of it) are lost, the reconstructed image is, in most cases, of unacceptable quality (because the coefficients get misplaced) and the reconstructed video is rendered useless.

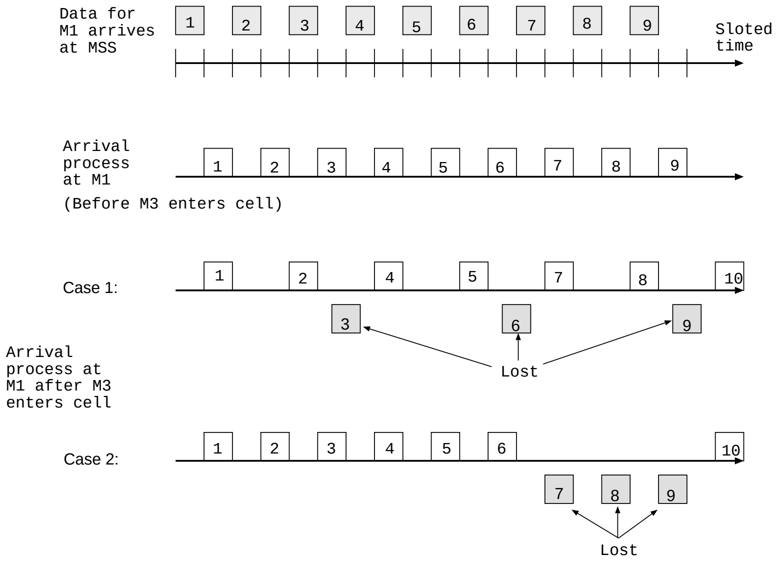


Figure 2: Effect of different losses on M1's connection.

for some applications such as audio. In [6] we have presented a network-level solution that ensures users do not see any breaks in service.

The basic concept behind this solution is the idea of a *group*. In Figure 3 let us assume that the mobile user is in cell c_i at some given time. A collection of cells surrounding c_i that the mobile user can move into is defined as the current group of the mobile user and is denoted by g_i (the shaded region). Any message destined for the mobile user is multicast to all the cells in g_i. This ensures that even if the mobile user moves rapidly out of c_i upon arriving at a new cell it finds messages waiting for it. This method of anticipating the arrival of the mobile user and prebuffering is called *predictive buffering*. Note that the composition of the group changes as the mobile user moves between cells, see Figure 4.

The shape and composition of a group is determined by several factors such as the set of neighboring cells that can be accessed from the present one (e.g., if a cell is made up of one room in a building – a *picocell*– then the user has limited choices of neighboring picocells to move into), the speed at which the mobile user moves between cells and the direction of motion. Cells that are separated by a wall in a building or in a campus will not belong to the same group; likewise cells on different floors of the same building will not belong to the same group. The direction of motion also has a bearing upon the composition of the group. In Figure 5 the groups defined for mobile users a and b are different because they have different directions of motion. Note that three cells in the immediate vicinity of a and b do not belong to either group because these cells are separated by walls.

2.2.1 QOS Parameter: *Probability of Seamless Communication*

One drawback with the above approach is the enormous overhead in terms of storage space required at the mobile support stations. If the average size of a group is k cells then there is an overhead of $k-1$ messages per message in the network (because the same message will be delivered to k mobile support stations). How can this buffering overhead be reduced?

If we know all future movements of a mobile user then we could virtually eliminate the storage overhead by anticipating the future position of the mobile user and sending its messages to the appropriate mobile support station. In some circumstances, where the mobile user is in a car traveling on a highway, we can fairly accurately predict the future positions of the user and buffer accordingly. However, this

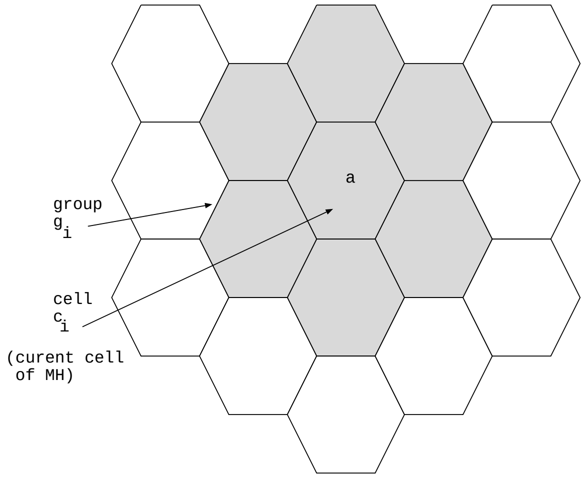


Figure 3: *Group* for mobile user a.

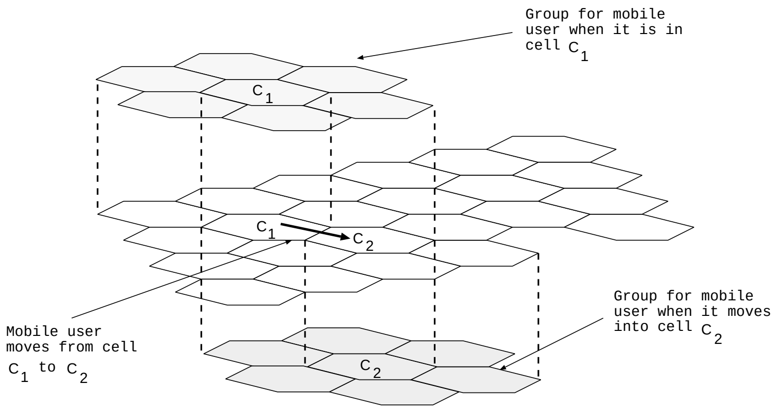


Figure 4: Group changes as MH moves from cell c_1 to c_2.

is not always possible.

In [7] we proposed two orthogonal approaches for solving this problem. The first approach is to estimate the latency of the user in a cell and begin prebuffering at neighboring cells (i.e., cells of the group) only after a certain amount of time. Thus, if the latency of a user in a cell is t, we begin predictive buffering at cells of the group starting at τ (our estimate for t which is, of course, not known a priori). Therefore the buffering overhead is only $(k-1)*(t-\tau)*d$ every t sec, where d is the data rate of the mobile user's open connections. If the value of τ can be estimated accurately, the predictive buffering overhead can be substantially reduced. In [7] we show that if the cell latency of the mobile user is a normal distribution then, in most cases, the buffer overhead is only double. A drawback of such delayed buffering is that if the user moves rapidly through the cell, there may be a break in service when it enters the next cell because buffering has not been done there. If the mean cell latency is 5 sec we see from Figure 6 that if $\tau = 4.0$ sec the buffer overhead *per* group member is less than 20%. That means if the number of group members (not including the current cell of the MH) is 5 the overhead is 100 % (rather than 500 % if we do not use delayed pre-buffering). The probability of a break in service is less than 0.1, see Figure 7.

The second approach is to restrict the group membership based on observed user behavior. Suppose 90% of mobile users move from cell a to either cell b or cell c. Then if we multicast messages only to

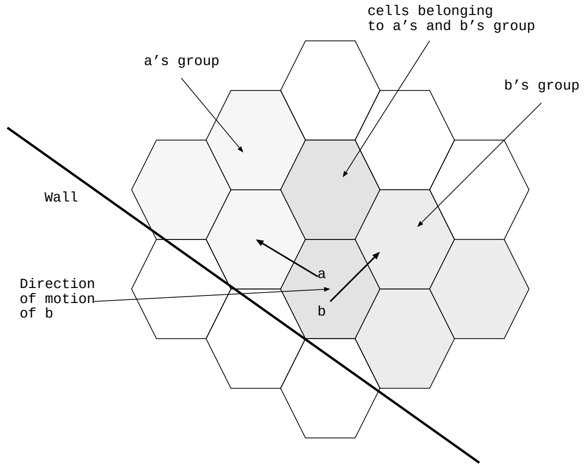


Figure 5: Group composition depends upon direction of motion.

neighboring cells b and c (for a mobile user currently in cell a) we can guarantee that with a probability of 0.9 there will be no break in service. This type of a guarantee is sufficient for many applications. For example, if the mobile user has an open audio channel, in many instances it does not matter if there are occasional breaks in service. On the other hand, if the application being run is online banking, breaks in service may be unacceptable. Clearly, for different applications an acceptable probability of break in service can be between 1.0 and 0.0.

Thus we define the **probability of seamless communication** as another QOS parameter. Depending on the type of application, a mobile user may require a small probability of seamless communication, as in an ftp application where breaks in service are of little concern so long as lost data is retransmitted, or a high probability of seamless communication, as in a videophone application where breaks in service every 5 sec are unacceptable (recall the car example from section 1). Therefore, based on user requirements, the composition of the group can be altered. Clearly, since a high probability of seamless communication requires a large amount of buffer space, these users will be required to pay more for the service.

2.3 Implications for Network Design

A *mobile network* refers to a collection of nodes, both fixed and mobile, that constitute the mobile computing environment. The architecture of this network must be such that the two QOS parameters, loss profiles and probability of seamless communication, can be easily incorporated into the communication structure.

In order to implement seamless communication it is necessary to be able to track users as they move, predict future position of these users in order to define groups and ensure that data gets multicast to the appropriate group. All these functions need to be implemented efficiently because otherwise, since the number of mobile users is likely to be huge, there will be a large amount of bookkeeping traffic.

It is noteworthy that the MSS nodes by themselves cannot implement loss profiles because as a user moves between cells (all of which maybe crowded and thus require selective discarding of data) its open connections will see discarded data in each cell. It is likely that, *the cumulative loss over a long time period* will typically not match the specified loss profile. This is because a individual MSS node is unaware of the past history of a connection (i.e., before the user moved into its cell) and therefore bases its decision to discard data on local information only (i.e., bandwidth limitations, other user needs, etc.). This is clearly a problem and needs to be addressed in any architecture. Note that requiring the

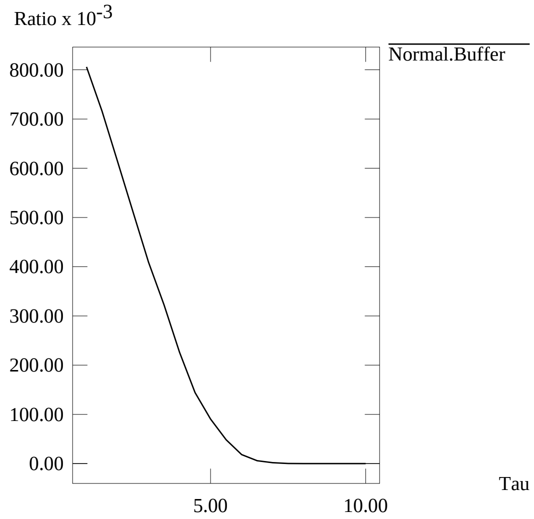


Figure 6: Buffer overhead at group members.

service-provider (from the fixed high-speed network) to implement loss profiles for mobile users is not an acceptable solution because, ideally, user mobility must be made transparent to nodes of the fixed network. In the next section we present our proposal for a mobile network architecture that allows for an efficient implementation of these two QOS parameters.

3 Architectural and Internetworking Issues

3.1 Network Design Issues

Our *conceptual view* of the future mobile computing environment is shown in Figure 8. Mobile users communicate with each other and with nodes in the high-speed network via connection endpoints. Thus, to receive a service (e.g., yellow pages) the mobile user **M** needs to set up a connection via host **S**. However, as the mobile user moves about, it may move away from **S** and will need to set up new connections with another connection endpoint such as **R**. Nodes **M, R, S**, etc. form the *mobile network*. Our focus in this section is on developing a mobile network arhitecture that provides a simple way to satisfy the QOS requirements of diverse applications. Corresponding transport layer issues are discussed in section 4.

Some authors, for instance [14], view the mobile network as an extension of ATM networks and suggest that the mobile network architecture and protocols be compatible with ATM in terms of functional layers and services so that there is a minimum of protocol processing at the interface between the mobile network and the ATM network. The architecture proposed consists of base stations and mobile users at the lowest level. The base stations are connected to ATM multiplexers which are in turn connected to an ATM-switching fabric. Thus the mobile network looks exactly like the ATM network except that the users may be mobile. As in ATM networks, the data services to be made available to users are CBR (constant bit-rate), VBR (variable bit-rate), connectionless and connection-oriented.

In our view this approach suffers from several drawbacks.

- Let us consider the situation discussed in section 2.1 where a mobile user with open connections moves into a cell causing the total requested bandwidth to exceed the total available bandwidth in

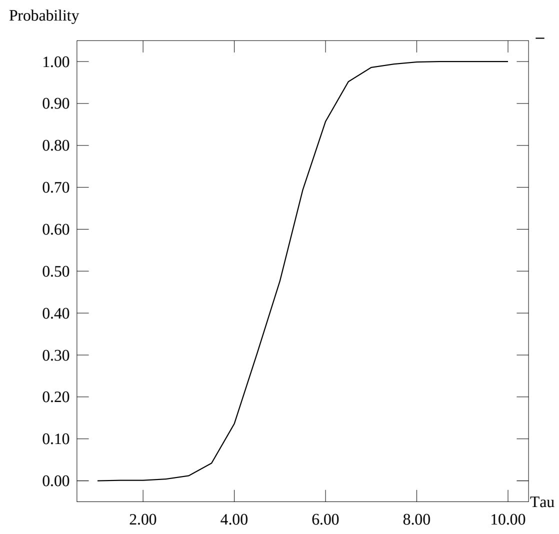


Figure 7: Probability of break in service.

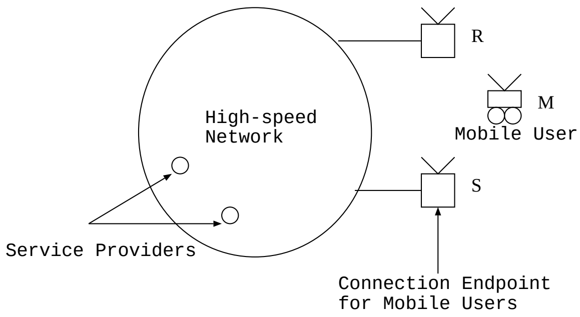


Figure 8: Conceptual View of mobile Network.

this cell. If we adapt the approach in [14], since the QOS parameters for an open connection can no longer be satisfied, the mobile user may, (1) close connections for which the QOS parameters can no longer be met, or (2) renegotiate QOS parameters for these connections. The first solution is clearly undesirable. The second solution requires renegotiation of QOS parameters most every time the user crosses cell boundaries (which can happen every 5–10 sec in many cases) – an undesirable situation.

- As mobile users move between microcells it is possible that some ATM cells for this user may be delivered *out-of-order* because the previous base station will forward undelivered cells to the current base station. This problem can be overcome by requiring the new base station to fetch untransmitted ATM cells from the previous base station before transmitting any new cells to the mobile user. This, unfortunately, leads to delays which may or may not be acceptable depending on the *type* of application being run. Thus for an audio application it may be preferable for the old base station to discard the untransmitted ATM cells while for an ftp application it is necessary to ensure correct and in-order delivery of all ATM cells. This means that the base stations need to know the nature and type of *all* services being received by all mobile users within their cell so that they can decide, for new users, whether undelivered cells from the previous base station need to be fetched or not. Thus the base stations need to be devices operating at the *transport layer* rather than at the network-layer thus adding, significantly, to their cost and complexity.

- There appears to be no simple way of providing seamless communication via groups because this architecture provides no support for deciding on appropriate multicast groups as users move about. There is also the problem of the mobile user receiving *duplicate* ATM cells when it moves into a new microcell. This is because all ATM cells are multicast to base stations in the group. When the mobile user moves to a new microcell, the base station there begins transmitting all buffered ATM cells, some of which may already have been transmitted to the user in the previous cell.

- Another drawback of this approach is that as the mobile user moves between cells, the path needs to be updated (i.e., modify tables in ATM switches).

3.2 Our Approach

As we discussed in sections 1 and 2, the mobile environment has unique characteristics not found in high-speed fixed networks. We believe that these characteristics combined with end-user expectation, rather than compatibility with ATM, must guide the design of mobile networks.

In our approach we view the mobile network as being *logically different* from fixed high-speed networks at the *transport layer*. Mappings between transport classes of high-speed networks and the mobile network are handled by special nodes (called *Supervisor Hosts*). All connections between mobile users and fixed service-providers in the high-speed network are handled by these Supervisor Hosts. This view is illustrated in Figure 9.

The architecture of our system may be viewed as a three-level hierarchy. At the lowest level are the *Mobile Hosts* or mobile users (MH) who may be viewed as hand-held computers equipped with a radio transmitter/receiver. They communicate with one another and with stationary hosts as they move around. At the next level we have the *Mobile Support Stations* (MSS) – one to each cell. The MSSs provide mobile users with connectivity to the underlying network and to one another. Finally, several MSSs are controlled by an assigned supervisor machine called the Supervisor Host (SH). The SH is connected to the wired network and it handles most of the routing and other protocol details for the mobile users. *In addition it maintains connections for mobile users, handles flow-control and is responsible for maintaining the negotiated quality of service.* If the subnet spans a large physical area we may choose to have more than one SH per subnet e.g., if a subnet straddles many floors of a building then we may have one SH per floor. On the other hand, we may choose to have one SH for several floors. Notice that the two Supervisor Hosts in Figure 9 share cells at the boundary of their respective domains. This is needed to ensure that groups can be implemented across SH boundaries.

Our architecture offers several advantages:

- As we discussed in [6] seamless communication using groups can be very easily implemented in this architecture. The SH tracks users within its domain and maintains group information for each user. Packets for a particular mobile user are multicast to the MSS nodes that form its current group. This means that nodes of the fixed high-speed network (e.g., service-providers) do not have to worry about providing seamless communication to the mobile users themselves. Furthermore, as we will discuss in section 3.4, communication between the SH and MH nodes takes place via packets containing sequence numbers. Therefore the problems of duplicate or out-of-order data arriving at the MH can be easily handled. Recall that this was a problem with the approach in [14].

- An implication of our approach is that it is now possible to have one set of QOS parameters for the mobile part of a connection (between the mobile user and the Supervisor Host) and another set defined for the high-speed fixed network. Doing this allows us to implement loss profiles in the mobile network in a way *transparent* to the fixed high-speed network. Thus, the SH decides which parts of a data stream to discard based upon user specified loss profiles. Recall from section

2.3 that in order to implement loss profiles correctly the past history of a connection must be considered. The SH is ideally suited for this task since it manages all aspects of a user connection so long as the user is within its domain.

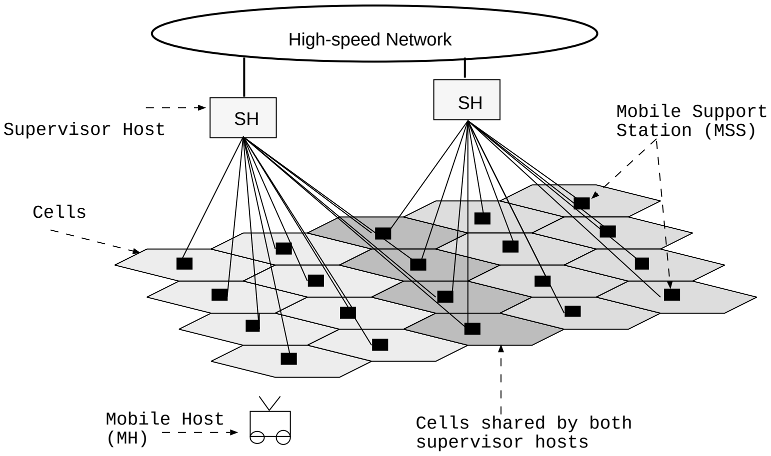


Figure 9: Logical view of our architecture.

3.2.1 Proposed Architecture

In order to design a workable architecture it helps to visualize the mobile computing environment of the future. The number of users expected is, of course, very large. But more importantly, the *spatial density* of these users will vary widely depending on the region we look at. Thus, in a campus or office building we will expect a higher density of users as compared with a highway. In fact, in a building the density of users will be very high per square foot compared to almost any other environment. What does all this mean from an architectural standpoint?

As mentioned earlier, the average bandwidth that will be made available to mobile users is 1–2Mbps. If, however, the *average* density of mobile users *per microcell* is very high then it may be impossible to guarantee this bandwidth. We believe this situation will occur frequently in high-rise office buildings and shopping centers.

To deal with such situations we propose a cellular network architecture made up of both microcells and *picocells*. Note that the diameter of a picocell is of the order of 10m as opposed to a microcell whose diameter is of the order of 100m. Office buildings and other high-density sites will be served by a picocellular network while lower density regions will be serviced by a microcellular network, see [6, 9]. An advantage of a smaller cell size is higher throughput (because there will be fewer users per cell), greater frequency reuse (i.e., while adjacent cells use different radio frequencies, non-adjacent cells may use the same frequency) and low-power transmitters (resulting in longer battery life). The main drawback is that small cells imply *faster* inter-cell user mobility resulting in a host of routing, tracking and hand-off problems.

Figure 10 shows our proposal for an architecture that incorporates both the microcellular and the picocellular architectures. A cell in the picocellular network is typically restricted to a room and corridors may be made up of several cells. The cell shape may not be uniform and is constrained by the architecture of the building as is movement between cells. The intra-campus microcellular network consists of larger cells whose diameter is of the order of 100 meters. There are few restrictions on the movement between microcells. Note, however, that if a microcell is placed in a street it will have fewer entry points compared to a microcell placed in a park.

All the MSSs are connected to a LAN either directly or through a hub or concentrator to the SH. The MSSs are small in size and are typically placed on the ceiling of each room in the picocellular environment or in lamp posts (or other appropriate locations) in the microcellular environment. Each MSS contains a large amount of buffer space in order to support high-bandwidth applications for the mobile users.

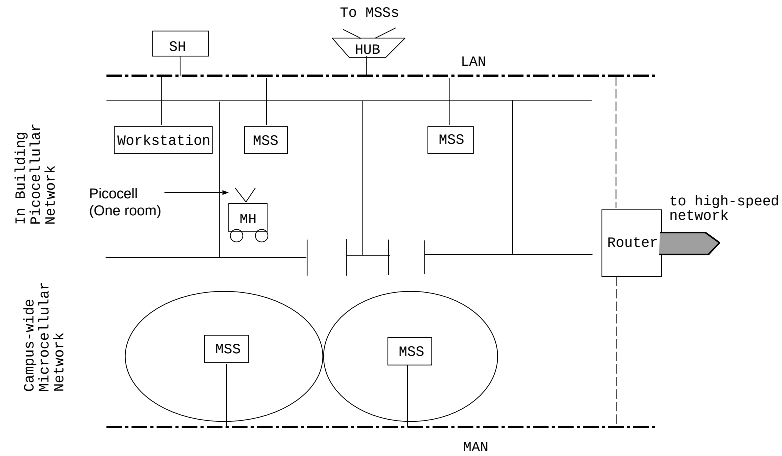


Figure 10: Proposed Architecture.

Our proposed architecture differs from traditional microcellular architectures, such as [5, 14], in important ways. The network intelligence in those architectures resides in the MSS nodes. The MSS is responsible for routing decisions, tracking of mobile users and forwarding packets for mobile users that have moved into new cells. It is also responsible for providing the negotiated quality of service. In our view this is not a scalable architecture because, as the size of the mobile network grows, the MSSs will have to contend with increasingly complicated routing, tracking and data forwarding decisions. The problem becomes even more severe when we consider in-building picocells with very rapid user mobility between cells (because of their small size users move faster through them).

[4] also discusses a three-level hierarchy for supporting cellular mobile telephone systems. The lowest level of this hierarchy is called the radio infrastructure and consists of mobile users and base transceiver stations (MSSs). The next level consists of base station controllers each of which controls several base transceiver stations. Many such base station controllers are controlled by a mobile switching center. The goal of this architecture is to optimize the overhead of mobility management. The base station controllers and mobile switching centers are switches whose main responsibility is to forward calls to the correct base transceiver station and track mobile users. In our approach, on the other hand, we require the three-level hierarchy to do much more that switching and tracking. As we have mentioned earlier, the SH performs a vital role in maintaining QOS guarantees for mobile users. Furthermore, the MSS and SH are responsible for flow-control.

Our solution to the problem is more cost effective and easily scalable. By having MSS nodes act simply as a connection between the mobile users in a cell and the SH, we can reduce the cost of these devices. Further, by having a single SH serve a 'large' geographical area, it becomes easier to track mobile users even if they do move rapidly between cells (e.g., a person walking rapidly in a building will move rapidly between picocells but all of these cells will probably be connected to the same SH). It also provides a natural way of guaranteeing the negoiated quality of service to mobile users as discussed in section 3.5. Our architecture is also more secure because the SH nodes participate in all calls made by mobile users. Thus we can build firewalls within the SH to avoid unauthorized access to network resources. In the traditional architectures, each MSS would have to have security features built in – an added cost.

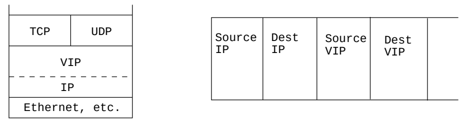
3.3 Internetworking in our Architecture

Several authors have presented different solutions to the problem of *addressing* mobile users. [16] describes a solution for addressing mobile users in the ARPAnet. A global database combined with source routing was the proposed solution. Obviously this does not scale to networks with a large number of very mobile users.

[11] proposes a more appropriate solution for mobile computing called the **IPIP** ("IP-within-IP") protocol. Every MH has a unique IP address called its 'home address'. To deliver a packet to a remote MH, the source MSS first broadcasts an ARP to all other MSS nodes to locate the MH. Eventually some MSS responds. The source MSS then encapsulates each packet from the source MH within another packet containing the IP address of the MSS in whose cell the destination MH is located. Upon receiving this packet the destination MSS extracts the original packet and attempts to deliver it to the destination MH. If the MH has moved away, the destination MSS attempts to locate it by broadcasting an ARP request. As discussed in [17] this method is not easily scalable.

[17] proposes a very flexible solution to the problem of addressing mobile users – the Virtual Internet Protocol or **VIP**. Here every host has a *virtual network address* (VIP address) that is unchanging. In addition, hosts have associated *physical network addresses* (traditional IP addresses) that may change as the host moves around. At the transport layer, the target node is always specified by its VIP address only. The address resolution from the VIP address to the IP address takes place at either the network layer of the same machine or at a gateway. Both the host machines and gateways, maintain a cache of VIP to IP mappings with associated time stamps. This information is in the form of a table and is called AMT (or *address mapping table*).

The network layer in this approach is divided into a VIP sublayer sitting on top of the traditional IP sublayer, see Figure 11(a). The VIP sublayer at a node or the gateway performs the VIP to IP address conversion. The underlying network then uses the traditional IP protocols to send packets from the source to the destination. The structure of packet headers is shown in Figure 11(b).



(a) Protocol Layering. (b) Packet Header.

Figure 11: VIP approach adopted from [17].

Because of the many benefits of the VIP solution we believe that it will be eminently suitable for our proposed mobile network architecture. We assume that every mobile host has a globally unique VIP address. Furthermore, we assume that all the stationary hosts (MSS, SH and other network-wide servers) have both a VIP and a unique IP address. Unlike [17], however, we do not *require* MH nodes to have an IP address. This is because MH nodes move around so rapidly that there would be little point in assigning them IP addresses.

The SH keeps a list of VIP addresses of all the mobile users within its purview. Furthermore, all these VIP addresses are *associated with this SH as aliases.* Thus, if a packet needs to be sent from an external node to a local MH, the external node simply addresses it with the VIP address of the MH node. Network routers associate this VIP address with the IP address of the SH and route the packet accordingly. The SH then forwards the packet to the MH via the MSS in whose cell the MH is currently located.

3.4 Connection Management

Unlike [14] we believe that the smallest unit of communication between mobile users and SH nodes must be *variable-sized packets.* The reason is that every packet can be numbered and duplicates can be easily discarded by the MH (this is a problem in [14] as discussed in section 3.1). This enables an easy implementation of groups for seamless communication and simplifies flow-control.

We proposed a connection setup and maintenance protocol in [6] that uses packets. As discussed there, whenever an MH needs to open a new connection it sends a request to its SH. The SH sets up the connection and returns an identifier to the MH. Henceforth every packet sent along this connection carries its identifier. Similarly, every packet received by an MH (for every open connection) carries the identifier and a sequence number. The MH keeps track of the next sequence number expected and can, thus, discard duplicates and, in the case of connection-oriented service, request retransmission of missing packets. It is important to note that a packet must be transmitted completely while the MH remains in the same cell. This is necessary to ensure duplicate data is not received by the MH.

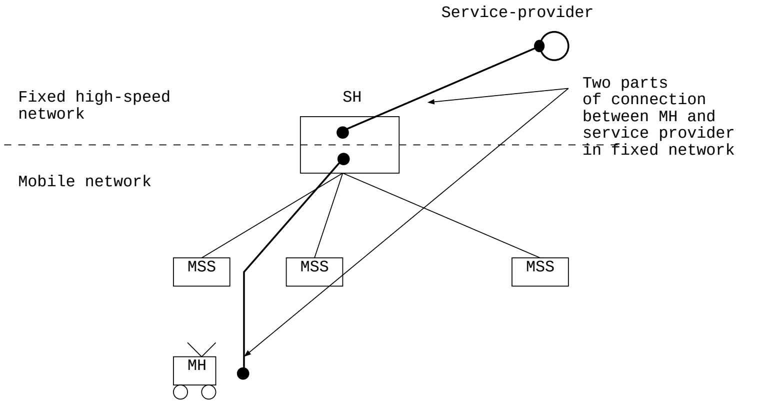
In our approach, flow-control occurs between the SH node and the MH node. The MSS nodes act simply as communications devices with large *caches.* Thus, if the MH has a reliable connection set up, then the SH sends packets to the MSS but *does not discard them until the MH acknowledges correct receipt.* The reason is that the MH may move out of range of an MSS before receiving all packets from it. In such situations, in our model, the old MSS simply discards these packets and the SH forwards them to the new MSS. In [1, 5] the old MSS would be responsible for forwarding these packets. The drawbacks of this approach have been discussed in [7].

In our model connection maintenance is the responsibility of the SH. When the MH moves between cells in the same subnetwork (i.e., same SH) nothing has to be done in terms of connection maintenance except updating the entry for the present location of the MH and redefining the group (if the user has specified seamless communication). When, however, the MH moves between cells belonging to different Supervisor Hosts, the new SH *adopts* all the open connections of the MH. This involves informing the old SH so that packets can be forwarded to the new SH, see [6]. Giving new identifiers to all the open connections (since these identifiers must be unique within the subnet). In addition the new SH adds the VIP of the MH to its list of aliases (this involves informing the location servers) so that, in the future, packets for this MH will not be sent to the old SH but, rather, will be sent to the new SH.

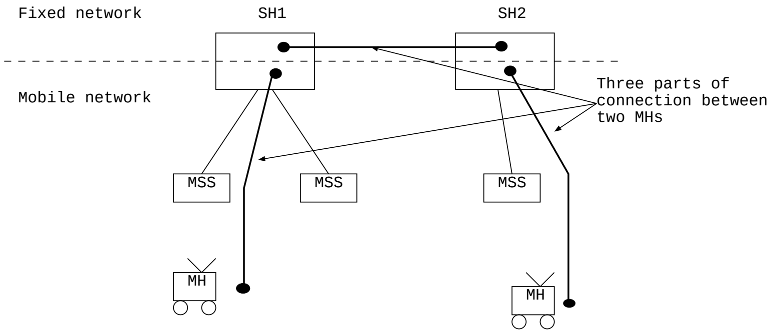
3.5 Architectural support for Satisfying QOS Requirements

Our architecture separates the mobile network from the high-speed wired network and provides connectivity between the two via supervisor hosts. Thus, a mobile user may set up connections where the other end-point is either another mobile user or a fixed host (e.g., a service-provider) in the fixed network. In either case the connection is managed by the current SH of the mobile host(s). Figure 12 illustrates both types of connections. As we can see the connection between the MH and a fixed host is broken in two – one between the MH and a SH and another between the SH and the fixed host. Connections between two mobile users, on the other hand, are broken into three parts as shown. This is necessary because the SH nodes form part of, and communicate via, the high-speed network.

The reason for splitting the connection between the MH and the service-provider is to shield fixed nodes from the idiosyncrasies of the mobile environment. Thus, the service-provider sets up a connection with the SH (also part of the fixed network) assuming the SH is the other end-point of the connection. The SH sets up another connection to the MH. Thus for every **MH – service-provider** connection the QOS parameters are defined *separately* for the **MH – SH** part and for the **SH – service-provider** part of the connection. The traditional QOS parameters for high-speed networks will be used to specify service requirements for connections between the SH and the service – providers. The two new QOS



(a) MH to fixed host connection.



(b) MH to MH connection.

Figure 12: Connection details.

parameters we defined in section 2 will primarily be used for the SH – MH part of the connection.

How should the QOS parameters be defined between the SH and the service-provider? We believe that the QOS parameters here should be negotiated assuming that the mobile user is connected *directly* to the Supervisor Host (SH) via a dedicated link. The reason for doing this is simply that if the mobile user is in a lightly populated cell, it will receive a quality of service equal to that negotiated between the SH and the service-provider (i.e., the best possible service). On the other hand, if the MH is in a crowded cell, it will receive a quality of service dictated by the available bandwidth and the loss profiles specified.

A problem arises when a mobile user with open connections moves from the domain of one Supervisor Host (SH) into the domain of another. Since all the QOS parameters for all of its open connections were negotiated with respect to its old SH, the new SH will have to renegotiate these parameters with all the service-providers currently providing service to the mobile user. This involves delays and may therefore be undesirable. We propose two related approaches to handling this problem.

In the first approach, the old Supervisor Host *anticipates* the move and informs the new Supervisor Host who then begins negotiations with the service-providers *before* the mobile user moves into the domain of the new Supervisor Host. Thus the MH will see no break in service (but the quality of this service could be different). Unfortunately, this approach means that the service–provider will need to set up *new connections* to deliver its service to the new SH. This may not be a good idea because there is a significant cost to setting up new connections in high-speed networks. The second approach is to set up *static* connections[3] between neighboring (or even distant) Supervisor Hosts (like *virtual paths* or *pipes* in ATM) in the high-speed network so that all traffic for the mobile user can be *forwarded* from the old SH to the new SH along these connections obviating the need for the new SH to renegotiate QOS parameters with the service-provider.

We believe that a combination of these two approaches will provide the best solution. Thus, a geographical region (e.g., a city) is broken into regions. All SHs within a region have static connections established as in the second approach. When an MH moves between these regions, however, new QOS parameters will have to negotiated between a SH in the new region and the service-provider(s). If these regions are large enough the MH will, hopefully, see few breaks in service (when moving between regions) and, in many cases, will receive service while located within one region. The precise boundaries of regions will depend upon geographical and demographical characteristics of the city.

4 Traffic Classes at the Transport Layer

As we mentioned earlier, it is expected that mobile users will have complete access to network-wide services available in future high-speed networks. These services will be implemented on top of new technology, like ATM, using the four traffic classes [3] – connectionless, connection-oriented, constant bit-rate and variable bit-rate (referred to as AAL 1 thru AAL 5 in [13]). Clearly, if mobile users are to have access to these services, the mobile networks will have to provide equivalent traffic classes at the transport layer of the mobile users.

Unfortunately, the unique physical characteristics of the mobile computing environment makes it impossible to have the *same interpretation* of constant bit-rate and variable bit-rate traffic classes. Mobility allows users with open connections to congregate in a cell thus overwhelming the available bandwidth. In such situations constant bit-rate connections may see losses (unlike the case in ATM) implying a need for different framing strategies at the transport layer to detect and maybe correct these losses. Similarly variable bit-rate connections may see unusually high losses resulting in a similar need to have better framing strategies (losses of this type of traffic are very small, perhaps 10^{-5}, in ATM but could be of the order of 10^{-1} in mobile environments).

Does this mean that the transport entities in the high-speed networks have to be redefined? The answer is **no** because our architecture isolates the mobile network from the high-speed network and all connections pass through Supervisor Hosts. We propose that similar traffic classes be defined for the mobile environment and that the *transport entity at the Supervisor Host has the responsibility of translating between the framing protocol of the mobile environment and the framing protocol of the high-speed network environment.* Furthermore, flow-control issues and service degradation issues will also be handled by the supervisor host. It is noteworthy that in the architectures used in [1, 5], the MSS nodes or MH nodes will have to perform this task – adding to their complexity and hence cost.

Another advantage of providing different traffic classes for the mobile environment is the ability to provide users with *differential service* based on the type of application they are running, the *loss profiles* for these applications and the amount they are willing to pay to maintain their service quality.

We propose five traffic classes for the mobile computing environment – connectionless, connection-oriented, VBR-P (a generalization of variable bit-rate), CBR-P and CBR-C (variations of constant bit-rate). These classes of traffic are direct analogues of the four traffic classes defined for ATM networks. Figure 13 illustrates the place these traffic classes will occupy in the network. As we can see, the Supervisor Hosts (SH) serve as the gateway between the mobile part of the network and the wired part. The service providers live in the wired part of the network.

[3] These connections exist forever.

When a mobile user needs to access some service, the SH sets up a connection with the service provider and *behaves as the connection end-point for the service from the point of view of the service provider.* The mobile user then receives the service from its local SH. Any renegotiation of the quality of service will then be handled between the mobile user and the SH – this is because the poor quality is a result of the mobile environment and not the wired-network.

The traffic classes we propose (for communication between supervisor hosts and mobile users) are:

1. **Connectionless Service:** Like UDP with no guarantee of correct delivery.

2. **Connection-oriented Service:** Guaranteed in-order delivery of packets.

3. **Variable Bit-Rate – Priority:** VBR-P is a generalization of the VBR service class in high-speed networks. Here the mobile network *tries* to provide *bounded delay* and *bounded loss* service. Unlike VBR, however, VBR-P is a *best effort* service class. That means if the bandwidth available decreases, the loss probability increases while maintaining the same delay characteristics. From the point of view of video applications this type of service degradation appears to be more acceptable than increasing delays while maintaining low loss probabilities.

4. **Constant Bit-Rate – Priority:** CBR-P is a generalization of CBR in high-speed networks. The main difference is that it is only a *best effort* guarantee with increasing losses as the available bandwidth shrinks. Note however that delays *do not* increase or change. This is important for some applications such as audio. Different priority levels make it easier for the SH to discard data in situations where this becomes necessary.

5. **Constant Bit-Rate – Critical:** CBR-C is a special class of service that must be made available for critical applications such as 911, medical services (EMS), paging service etc. This is a low bit-rate service that takes priority over all others and suffers no loss or queueing delays.

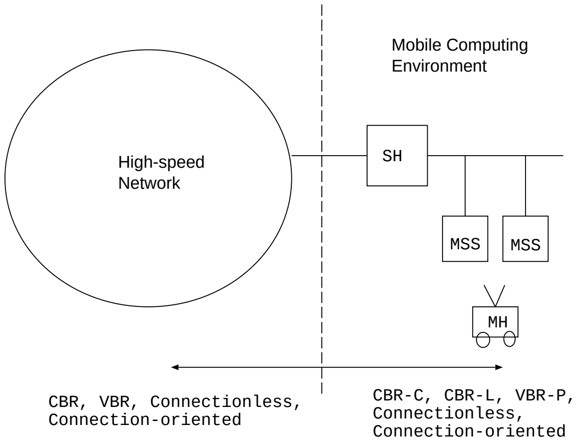


Figure 13: Proposed Traffic Classes.

The reason for making this traffic class available is that users will, in all likelihood, use the mobile PCS (personal communication system) as a substitute for current day telephones, beepers, pagers etc. Thus, in order to continue to support such services, it is necessary to have a different traffic class that is guaranteed priority access to the network. Observe that if CBR-P or VBR-P were used to implement these type of services, they would compete for bandwidth with other high-priority (but non-critical) applications such as video – resulting, possibly, in situations where important messages get blocked.

5 Conclusions

In this paper we identified two QOS parameters that are unique to the mobile environment. A network architecture and transport protocol suite are proposed that makes mobility invisible to nodes of the fixed high-speed networks and which, furthermore, allows easy implementation of these new QOS parameters in the mobile environment.

References

[1] A. Acharya and R. Badrinath, "Delivering multicast messages in networks with mobile hosts" *13th Intl. Conf. on Distributed Computing Systems*, May 1993.

[2] B. Awerbuch and D. Peleg, "Concurrent Online Tracking of Mobile Users", *Sigcomm '91*, pp. 221-233.

[3] D. Bertsekas and R. Gallagher, Data Networks, 2nd edition, Prentice Hall, 1992.

[4] I. Arieh Cimet, "How to Assign Service Areas in a Cellular Mobile Telephone System", *IEEE ICC'94*, pp. 197-200, May 1994.

[5] D. Duchamp, Steven K. Feiner and G. Q. Maguire, "Software technology for wireless Mobile computing" *IEEE Network Mag.*, pp 12-18, November 1991.

[6] R. Ghai and S. Singh, "A Protocol for Seamless Communication in Picocellular Networks", *Proceedings IEEE ICC'94*, May 1-5, 1994, pp. 192-196.

[7] R. Ghai and S. Singh, "A Network Architecture and Communication Protocol for Picocellular Networks", *IEEE Personal Communications Magazine*, submitted.

[8] D.J. Goodman, R.A. Valenzuela, K.T. Gayliard and B. Ramamurthi,"Packet Reservation Multiple Access for Local Wireless Communications", *IEEE Trans. on Communications*, Vol. 37, pp. 885-890, Aug. 1989.

[9] David J. Goodman, "Cellular Packet Communications", *IEEE Trans. on Comm.*, vol. 38, no. 8, pp 1272-1280, August 1990.

[10] David J. Goodman, "Trends in Cellular and Cordless Communications",*IEEE Communications Magazine*, pp 31-40, June 1991.

[11] J. Ioanidis, D. Duchamp and G. Q. Maguire, "IP-based protocols for mobile internetworking" *Proc. of ACM SIGCOMM'91*, pp 235-245, September 1991.

[12] C.S. Joseph, *et al*, "Propagation Measurement to Support Third Generation Mobile Radio Network Planning", *43rd IEEE Vehicular Tech. Conf.*, May 1993, pp. 61-64.

[13] Craig Partridge, Gigabit Networking, Addison Wesley, 1993.

[14] D. Raychaudhuri, "ATM Based Transport Architecture for Multiservices Wireless Personal Communication Networks", *Proceedings IEEE ICC'94*, May 1-5, 1994, pp. 559-565.

[15] R. Steele, "The cellular environment of lightweight handheld portables" *IEEE Commun. Mag.*, vol 27, no. 7, pp 20-29, July 1989.

[16] C. Sunshine and J. Postel, "Addressing Mobile Hosts in the ARPA Internet Environment", *IEN* 135, March 1980.

[17] F. Teraoka and M. Tokoro, "Host Migration Transparency in IP Networks: The VIP Approach", *SIGCOMM*, Vol. 23, No. 1, Jan 1993, pp. 45-65.

[18] J.Z. Wang, "A Fully Distributed Location Registration Strategy for Universal Personal Communication Systems", *IEEE JSAC*, Vol. 11, No. 6, pp. 850-860, August 1993.