From *Proceedings of the USENIX Symposium on Mobile and Location-Independent Computing*, Cambridge, Massachusetts, August 1993.

Providing Connection-Oriented Network Services to Mobile Hosts[1]

Kimberly Keeton, Bruce A. Mah, Srinivasan Seshan, Randy H. Katz, Domenico Ferrari
{kkeeton,bmah,ss,randy,ferrari}@CS.Berkeley.EDU
Computer Science Division
University of California at Berkeley

Abstract

Mobile computers using wireless networks, along with multimedia applications, are two emerging trends in computer systems. This new mobile multimedia computing environment presents many challenges, due to the requirements of multimedia applications and the mobile nature of hosts. We present several alternative schemes for maintaining network connections used to provide multimedia service, as hosts move through a nano-cellular radio network. These algorithms modify existing connections by partially re-establishing them to perform handoffs. Using a simple analytical model, we compare the schemes on the basis of the service disruption caused by handoffs, required buffering, and excess resources required to perform the handoffs.

1.0 Introduction

Two technological trends of the 1990s are the emerging use of wireless computers and network support for multimedia services. The products of these trends hold forth the promise of being combined in innovative ways to provide applications such as mobile digital video and audio conferencing. The new computing environment presented by wireless multimedia personal communication systems such as discussed in [Sheng92] introduces many challenges, because of the requirements of multimedia applications and the mobile nature of the hosts.

Multimedia applications typically have strict requirements such as delay, delay jitter, throughput, and reliability bounds. Real-time network services are designed to guarantee these performance parameters to applications that request them. In order to provide these performance guarantees to individual conversations, approaches such as [Ferrari92] rely on connection-oriented networks and resource reservation. Per-connection resource allocation in a connection-oriented network is necessary to ensure that guarantees will not be violated under conditions of heavy network congestion. (In connectionless network environments, congestion typically causes decreased throughput, increased delay, and even an increased probability of packet loss.) These methods, which are designed for wired networks, do not directly address the mobility of hosts. While these performance guarantees are necessary for the delivery of multimedia data to mobile hosts, our concern in this paper is not providing these guarantees, but instead maintaining the connections in use as a host moves within the network.

User mobility forces networks to cope with new dynamics of routing and resource location. Generally, mobile networks are composed of a wired, packet-switching, backbone network and a wireless (e.g. cellular radio or infrared) network. The wireless network is organized into geographically-defined cells, with a

---

[1] This research was supported by the Department of Energy under grant DE-FG03-92ER25135/A000, by the National Science Foundation and the Defense Advanced Research Projects Agency (DARPA) under Cooperative Agreement NCR-8919038 and by the Corporation for National Research Initiatives, AT&T Bell Laboratories, Hitachi Ltd., Hitachi America Ltd., Pacific Bell, the University of California under a MICRO grant, and the International Computer Science Institute. Kimberly Keeton was supported by an AT&T Graduate Fellowship. Bruce A. Mah was supported by an NSF Graduate Fellowship.

This paper, as well as other works on real-time computer networks is available for anonymous FTP from `tenet.ICSI.Berkeley.EDU`.

control point called a *base station* (BS) in each of the cells. The base stations, which are attached to the wired network, provide a gateway for communication between the wireless network and the backbone interconnect. As a *mobile host* (MH) travels between wireless cells, the task of forwarding data between the wired network and the mobile host must be transferred to the new cell's BS.

This process, known as a *handoff*, must maintain end-to-end connectivity in the dynamically reconfigured network topology. Since our model of network communication is connection-oriented, each of the MH's connections (or channels) and the associated connection state must somehow be transferred to this new BS. The effectiveness of the rerouting performed by the handoff algorithm is determined by several criteria. In particular, it is desirable to minimize the service disruptions (such as loss of motion due to the replay of the last frame when subsequent frames do not arrive on time) and overheads such as latency, MH and BS buffering, and excess reservation of network resources.

Prior work related to host mobility does not simultaneously address the issues of connection-oriented service and de-centralized control. Much has been accomplished in the context of providing support for the Internet protocol suite in a wireless environment [Ioannidis91, Teraoka91]. However, because its network layer protocol (IP) is connectionless, network performance cannot be guaranteed under conditions of high load. Conversely, the cellular telephone network uses circuit switching to provide connection-oriented services. It maintains these connections during handoffs through highly centralized knowledge and control by the mobile telephone switching offices.

Two straightforward solutions to the challenge of connection-oriented handoffs are forwarding data from the original BS to the new BS and establishing a new connection between the multimedia source and the new BS. Because these approaches incur considerable overheads, we propose several alternate algorithms that modify the existing connections by partially re-establishing them to perform handoffs. The first protocol capitalizes on the logical locality of geographically adjacent cells to partially re-establish the connections during handoff. The second uses multicast facilities to provide connectivity to the new base station when the host moves.

We present a quantitative comparison of these strategies using analytically derived formulas for connection setup latency, required buffering, and excess resource reservation. We compare the algorithms and evaluate their feasibility, given the physical limitations imposed by current wireless network technology. Our basis for comparison is the case where connections are fully re-established to the source.

2.0 Environment

The physical and logical aspects of the networking environment will play a large role in determining the nature of the handoff algorithms and their requirements.

2.1 Computing Environment

The technologies used in the mobile system have a significant impact on the effectiveness of the handoff schemes. We have chosen several technology points, based on typical portable computers and switch-based networks, to characterize our computing environment. For instance, most mobile computers have fully functional CPUs, although the low-power requirements of the mobile host may limit its computational capabilities. We assume, however, that the mobile hosts contain enough processing power to run applications and the necessary communication protocols.

The wireless and wired network technologies available today define various aspects of mobile communication. We assume a nano-cellular radio network (with cells less than ten meters in diameter) covering the interior of a building. Users (and hence mobile hosts) move occasionally and slowly compared to the speed of the network. Mobile hosts communicate through the radio network to a subset of the base stations; each MH can communicate with at most one BS at a time. These base stations are gateways between the radio network and a switch-based store-and-forward backbone network, and facilitate communication between the mobile hosts and multimedia servers on the backbone network. Base stations are single-homed; that is, they have only one physical connection to the wired network. For simplicity, hosts and switches are interconnected in a hierarchical topology resembling a tree. The switches support duplex connections. (For convenience we refer to the direction towards the MH as the *downlink* and the direction away from the MH as the *uplink*.) Finally we assume that the switches and BSs can be programmed to support our handoff algorithms.

The characteristics of Code Division Multiple Access (CMDA) radio (< 2 GHz carrier frequency) nano-cellular networks [Schilling91] define some of the wireless communication patterns. In a radio network, there is generally considerable overlap between the cells of the network. (As stated before, however, a MH can communicate with only one BS at a time.) Each host in a radio network knows which BSs are "in range" and which of those BSs has the strongest radio signal. The relative strength of the radio signal from the current BS and the next strongest radio source gives the MH an indication of when it is close to a new cell. The MH can also use the CDMA codes to determine which cell it is approaching.

However, handoff algorithms cannot rely with perfect certainty on the availability of this information. Radio networks may have *null regions*, areas in which a MH loses wireless contact with its BS, possibly due to interference with radio propagation. The cell overlap information cannot be completely relied upon because it may be possible to enter a null region in one cell and exit the region in a different cell. Some radio networks may also have sharp boundaries between some cells (for example, doorways). Transitions across these boundaries may occur too quickly for any advance warning to be useful. Due to these two characteristics of wireless networks, our approach uses information about an impending cell transition as a hint to improve performance, rather than as an integral component of the handoff algorithms.

2.2 Communication Paradigm

We assume that the MHs will provide various multimedia services, as described in [Sheng92]. Many multimedia services, such as audio-video conferencing or video playback, have associated with them performance requirements that must be met to guarantee acceptable service to the users. [Ferrari90] describes the requirements that some typical applications place on networks. The Tenet Real-Time Protocol Suite [Ferrari92] is one approach to providing these *real-time performance guarantees* in packet-switching networks; this approach relies on resource reservation in a connection-oriented network environment.

A connection-oriented network paradigm with per-conversation resource reservation is needed to provide performance guarantees. In a connectionless scheme, successive packets may follow different paths in the network, and hence incur different delays in reaching the destination. Packet delays in this environment are much harder to control than if all packets followed the same path, as in a connection-oriented scheme. Per-conversation resource reservation is necessary to ensure that sufficient resources are available for real-time traffic at all times, including periods of high network load and congestion. Because network resources are finite, an admission control policy must be administered, to ensure that resources are not over-subscribed.

Before communication can take place, a real-time channel (a connection with performance guarantees) must be established. Channel establishment involves a single source-routed round-trip pass of control messages from the source to the destination of the connection and back.[2] On the forward pass, admission control tests are performed to ensure that the network will be able to satisfy this channel's performance requirements without violating the guarantees of existing real-time channels. Assuming the tests succeed, node resources (such as bandwidth) are tentatively allocated, pending the acceptance of this channel at all other participating nodes. On the return pass of a successful establishment, the resources of each node are committed to the channel. [Mah93] describes in detail a mechanism and protocol by which real-time channels can be established and managed.

We also assume that all connections have at most one MH as an endpoint (e.g., at least one endpoint is a fixed host on the wired network). This assumption somewhat simplifies the design of the algorithms, as they do not need to consider the case where both ends of a connection move simultaneously.

---

[2] The Tenet real-time channels are simplex unicast, but it is a fairly simple matter to support duplex real-time channels. Ongoing work is aimed at providing real-time multicast network services.

2.3 Topological Knowledge

The algorithms that we present modify existing connections to perform handoffs. We call the point where a connection is modified the *crossover point*. Choosing this crossover point requires some knowledge of the network topology. We assume, however, that control entities have only local knowledge of the network (such as next-hop routing information); this allows them to function without needing large status updates from their peers. This assumption also prevents a single entity from making rerouting decisions independently. Therefore, multiple control entities must collaborate using a distributed algorithm to determine the crossover point.

3.0 Algorithms

In this section we discuss three different schemes for handoffs in connection-oriented mobile networks. We present a Full Re-Establishment scheme as a basis for comparison. We then introduce two new algorithms, Incremental Re-Establishment and Multicast-Based Re-Establishment. The algorithms all rely on a number of pre-existing connections in the network, to be used primarily for control purposes. We assume that any two network nodes connected by a link have a control channel between them. Every pair of base stations responsible for physically adjacent cells also has a control connection between them. (We note that these base stations may not be directly connected by a physical link.) In addition, these algorithms require the BSs to buffer a bounded amount of data that has already been transmitted to the mobile host, to ensure that no data is lost when the mobile host moves to an adjacent cell. The algorithms make no attempt to re-order out-of-order packets. They will, however, attempt to prevent data from becoming out of order as the result of a handoff operation.

For each algorithm, we give a short description, followed by a brief example of its application re-rerouting a single channel. We also present the optimized case which takes advantage of the cell overlap hint. In each of the examples, the MH moves from the cell whose point of control is BS 1 into the cell corresponding to BS 2. Thus, in all cases, BS 1 is considered to be the “old BS” and BS 2 is considered the “new BS”. Numbered lines correspond to control messages exchanged between network entities to complete the handoff.

3.1 Full Re-Establishment

The Full Re-Establishment (FR) algorithm executes handoffs by establishing a set of completely new channels between the MH and the servers.

3.1.1 Full Re-Establishment without Hints

The “Without Hints” part of Figure 1 depicts the communication necessary to perform the FR scheme in the general case where the MH has no advanced warning that a handoff from BS 1 to BS 2 is about to occur. Once the MH enters the new cell, it identifies itself to the new cell’s BS, BS 2, and requests that its connections be rerouted through the new BS (message labeled “1” in the diagram). Included in this greeting message is an identifier for the old BS, BS 1, as well as a list of the identifiers for the various connections originating or terminating on the MH. After BS 2 acknowledges this greeting with message (2), data transmission from the MH may begin.

In order to avoid an interruption in data service, BS 2 immediately uses the pre-existing BS-to-BS control channel to request that downlink data from each of the MH’s connections be forwarded from the original BS (3). Message (3) also requests that BS 2 be allowed to forward uplink data from the MH through BS 1 to the Server. BS 1 acknowledges these requests, and may begin forwarding data to BS 2 just after (4). Once (4) has been received, BS 2 may begin forwarding data transmitted by the MH through BS 1.

While this data forwarding is occurring, BS 2 begins establishing connections to each of the appropriate Servers on the network (5). Once a connection is fully established (6), the new BS informs the MH (7) and sends a message (8) to the Server requesting that it redirect all data transmissions down the newly estab-

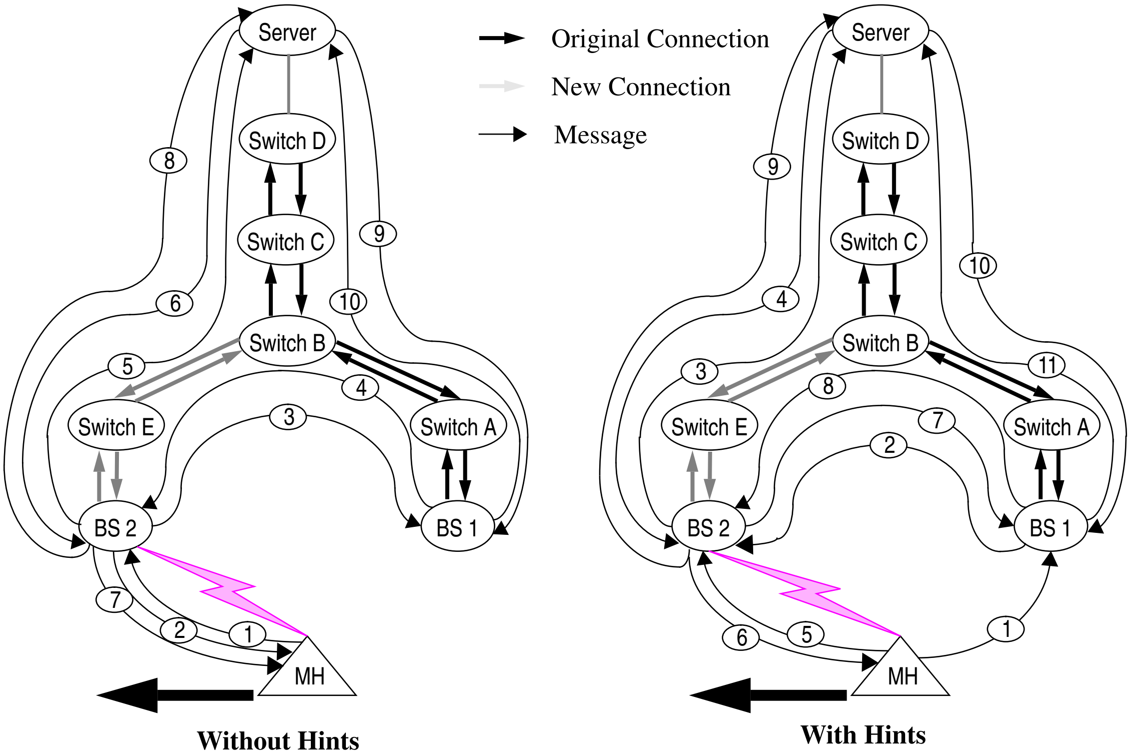


FIGURE 1. Full Re-Establishment

lished downlink. The downlink portion of the connection between Switch B and BS 1 is then torn down in step (9). The redirected downlink data, and the remainder of the data forwarded through BS 1 to the new BS, is buffered at BS 2 to ensure in-order delivery to the MH.

Once the new portion of the connection has been established, data from the MH can flow directly to the Server over the new connection. To preserve in-order delivery of the data from the MH to the Server, this uplink data must be initially buffered at BS 2 until the uplink data forwarded through BS 1 is drained from the old portion of the connection. This buffering is proportional to the difference between the transmission delays along the new path and the forwarding/old path. Once all messages forwarded through BS 1 have been delivered to the Server, the uplink portion of the old connection is torn down (10) and uplink data may flow directly through BS 2. The handoff is completed once all of the new connections have been created and the old connections destroyed.

3.1.2 Full Re-Establishment with Hints

In a radio network, the MH can take advantage of the overlap between adjacent cells to detect that it is entering a new cell before it loses contact with its current BS. This “hint” allows the MH to request that the current BS establish in advance new connections between the Servers and the new BS. This scenario is shown in the “With Hints” portion of Figure 1. At the hint time, the MH requests (1) that its current BS (BS 1) send a list (2) of active connections to the new BS (BS 2). The new BS may then establish each of the connections to the appropriate network Server (3). A successful establishment is indicated to BS 2 by the acknowledgment in step (4).

During this pre-establishment, the MH ceases communications with the old BS and begins communicating with the new BS (5). As in the more general algorithm, this greeting is then acknowledged (6). Depending on how much time has elapsed, the new connections may or may not have been established to the new BS. In the best case, establishment for each of the new connections has been completed, and message (6) also notifies the MH of which connections have been established. As in the more general algorithm without hints, BS 2 then initiates a forwarding request for all of the data transmitted during the radio communication switchover period on a given connection from the Server to BS 1 (7, 8). Downlink data can then be forwarded across the pre-existing channel by BS 1. However, the uplink data is not forwarded, but sent only on the newly established connection to the Server. To ensure in-order delivery of the data from the MH to the Server, this uplink data must be initially buffered at BS 2 until all of the data that was transmitted by the MH (while in the old cell) can be sent along the old connection. This buffering is proportional to the difference between the transmission delays along the two paths, taking into consideration the time for the MH to make contact with the new BS.

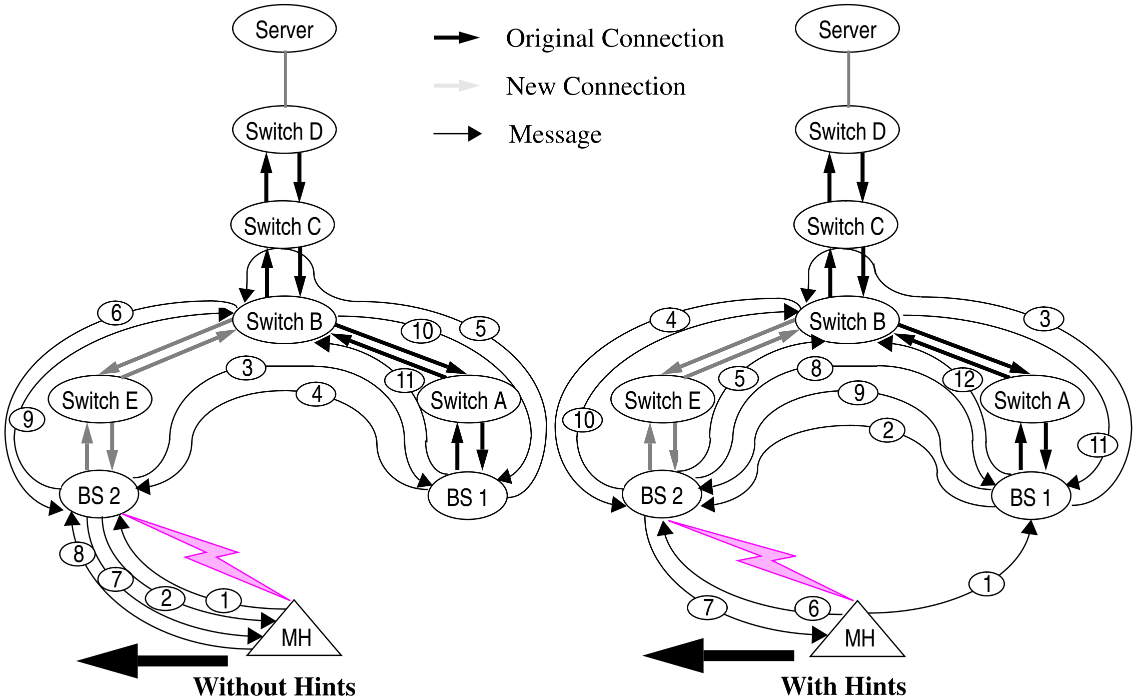
While downlink data is being forwarded, the new BS sends a message (9) to the server requesting that data be rerouted to the new connection. Once the Server switches active connections, it begins deleting the channel to the old BS (10). After all uplink data has been transported to the Server, the old connection is completely torn down (11).

If the MH arrives at the new BS before its connections have been established (but after the hint processing has begun), forwarding for both the down and uplink is initiated as in the general case with no hints. The MH is then notified as each connection is established. At this point, the new BS sends a message to the Server indicating that it may begin sending downlink data over the newly established channel. The teardown of the old channel proceeds as in the general case with no hints.

If the MH does not end up fully entering the cell it was approaching but instead eventually moves out of its range, it must send another message to its original BS revoking the hint. This results in a tear-down of the connections that had been set up in anticipation of the arrival of the MH.

3.2 Incremental Re-Establishment

In contrast to the FR algorithm, the Incremental Re-Establishment (IR) scheme attempts to re-use as much of an existing connection as possible, creating only the portion between the crossover point and the new cell's BS. The corresponding portion of the original connection is then torn down.



Without Hints **With Hints**

FIGURE 2. Incremental Re-Establishment

3.2.1 Incremental Re-Establishment without Hints

The “Without Hints” portion of Figure 2 illustrates the application of the IR scheme in the general case where the MH has no advanced warning of the impending handoff. When the MH first arrives in a new cell, it sends a greeting message (1) to the new BS, BS 2, which contains a list of connections to be rerouted as well as the identity of the old BS. Data transmission from the MH to BS 2 begins after BS 2 acknowledges this greeting (2). As in the FR scheme, BS 2 requests that BS 1 forward the downlink and uplink data from each of the MH’s connections across the BS-to-BS connection (3, 4).

In order to reuse a portion of the existing connection, message (3) also implicitly requests on behalf of the new BS that the old BS invoke the distributed crossover point location process. This location algorithm is initiated by the old BS because the new BS has no knowledge of the connection path to the old BS and possesses only local topological knowledge. BS 1 invokes this decision process by backtracking along the original connection route one hop at a time (5). Each switch along this route decides whether it knows the appropriate crossover point by examining its routing tables. If the switch uses different ports to reach the old and new BSs, it continues to forward this reroute message (Switches A and B in step 5). If the switch uses the same port to reach both the old and new BSs (Switch C), it knows that the switch one hop in the downlink direction (Switch B) is the crossover point. Switch C then communicates this fact to Switch B.

Once the crossover point has been identified, the new portion of the connection must be established and the old portion torn down. The first step, establishing the new partial connection between the crossover point (Switch B) and the new BS (BS 2), is performed as it would be in a wired network (6). This connection is established to the MH using message (7). Assuming establishment is successful, acknowledgments are sent from the MH to BS 2 (8) and from BS 2 back along the path to Switch B (9).

Message (9) is also an implicit request for Switch B to redirect all data transmitted by the Server down the newly established downlink. As in the FR algorithm, buffering at BS 2 is used to allow in-order delivery of downlink data to the MH. The downlink portion of the connection between Switch B and BS 1 is then torn down in step (10). After step (9), data from the MH can flow directly to the Server through BS 2. This uplink data is buffered at BS 2 to allow the uplink data forwarded through BS 1 to be delivered to the crossover point. Once all messages forwarded through BS 1 to the Server have been delivered to Switch B, the uplink portion of the old connection is torn down (11) and uplink data may flow over the new connection. When all new connections have been created and old connections destroyed, the handoff is complete.

3.2.2 Incremental Re-Establishment with Hints

The “With Hints” portion of Figure 2 shows how the cell overlap information can be used to facilitate the handoff. The MH informs the current BS, BS 1, of the potential new BS, BS 2, and requests that new partial connections be established through the appropriate crossover points to the new BS (1). BS 1 communicates the list of connections about to be re-established to the new BS (2), and then invokes the crossover location algorithm (3). Once located, the crossover point, Switch B, begins to establish a new connection to the new BS (4). A successful establishment is indicated to Switch B by the acknowledgment in step (5).

While the connection to the new BS is being pre-established, the MH moves completely into the new cell, where it identifies itself (as in the more general algorithm) to BS 2 (6) and is acknowledged (7). Analogous to the FR scheme with hints, in the optimal case where all connections are established, message (7) contains identifiers for the already-established connections. Similarly, BS 2 initiates downlink data forwarding from BS 1 (8, 9) and then requests that the crossover point redirect downlink data over the new connection (10). Again, uplink data is not forwarded through the old BS. Buffering to ensure in-order delivery of both downlink and uplink data is performed as in the FR scheme with hints, except that the endpoint of the two paths under consideration is the crossover point rather than the server. Finally, suboptimal cases are handled in a manner analogous to that of the FR scheme with hints.

3.3 Multicast-Based Re-Establishment

To support video conferencing and other distributed applications, some networks support multicast connections with dynamically changing memberships. The use of multicasting has several interesting ramifications. Because data for the downlinks are transmitted simultaneously to multiple base stations during the interim, the actual switchover can be fairly quick, with decreased buffering. This scheme also has an advantage in that only a small layer of functionality needs to be added on top of the multicast facility in order to support mobility.

3.3.1 Multicast-Based Re-Establishment without Hints

The “Without Hints” diagram in Figure 3 illustrates the operation of the MB handoff algorithm. The MH detects that it is entering a new cell, so it acquires a wireless channel to the new BS, BS 2, and sends it a greeting message (1). This message contains an identifier for the old BS, BS 1, as well as a list of the identifiers for the various multicast channels originating or terminating on the MH. This greeting is immediately acknowledged by BS 2 (2). BS 2 then sends this channel list to the old BS along the pre-existing BS-to-BS connection (3), requesting that all data for each channel be forwarded to the new BS. In addition, BS 2 requests that it be allowed to forward data from the MH through BS 1. Upon receipt of an acknowledgment (4), BS 2 begins transferring forwarded data from BS 1 to the MH and forwarding MH data destined for the server.

Concurrently with the forwarding requests, BS 2 executes a multicast join operation (which establishes a new branch connecting BS 2 with the existing channel) for each existing channel to add itself to the multicast channel (5). Upon receiving an acknowledgment indicating a successful establishment (6), the new BS notifies the MH of the successful join operation (7). BS 2 synchronizes the data arriving down the new branch with the data being drained from the old BS. To preserve in-order delivery of the data from the MH to the server, this data must be initially buffered at BS 2 until the data forwarded through BS 1 is drained from the old portion of the connection.