A Reliable Multicast Protocol in Networks with Mobile Hosts

Byung-Won On, Myong-Soon Park

Dept. of Computer Science and Engineering, Korea University

{bwon, myongsp}@iLab5.korea.ac.kr

Abstract

As portable computing devices become more popular, increasing the wide use of group communication applications that need reliable delivery of data, the need for reliable multicast protocol in networks with mobile hosts also increases. The highest protocol for reliable multicast in networks with mobile hosts uses an ack-based approach and consists of mobile support stations (MSSs) and supervisor hosts (SHs). The MSSs communicate with the mobile hosts within their cells via wireless links and forward ack messages from mobile hosts to SHs. Each SH controls a group of neighboring MSSs and delivers one ack message to the sender, after collecting ack messages from MSSs, each of which contains some mobile hosts joined in a multicast group. In this protocol, if the mobile hosts within the domain of particular SHs are continuously crowded according to the regional property such as downtowns and suburbs, an ack implosion of these SHs can occur due to a number of ack messages from mobile hosts. Therefore, the multicast group communication is limited by the imbalanced processing overhead among SHs. To solve the above problem, we propose an efficient reliable multicast protocol in networks with mobile hosts.

1.Introduction

As the demand for group communication increases, Internet Protocol (IP) Multicast is widely used to support multicast communication. The IP Multicast offers many advantages from the point of view of network bandwidth and CPU utilization by transmitting the same data to multiple receivers in one transmission rather than the number of receiving hosts [10, 11].

Portable, personalized computing devices such as PDA (Personal Digital Assistant) and HPC (Hand-held PC) have recently become popular, increasing the wide use of networks with mobile hosts for group communication applications based on the protocol to provide continuous computing ability while the mobile user is moving [3]. Examples of these applications are transmission of stock market quotes to subscribers, transmission of television programs (new services or pay-per-view services), or transmission of weather advisories, etc. To support the use of these applications, a protocol must be able to:

- Provide efficient 1×N multicast from a fixed sender to N users
- Guarantee delivery of all transmissions to each user
- Hide mobility from applications that use multicasting
- Allow dynamic and distributed adds and drops to the multicast group
- Allow users to belong to several multicast groups simultaneously

In this way, there is the need for a reliable multicast protocol in networks with mobile hosts to support the use of applications that are sharing information without a limitation of place. In these protocols, reliable delivery of multicast messages to all the mobile hosts in a multicast group, requires the sender to receive ack messages from all the mobile hosts [6, 7, 8, 16]. The Host View Multicast Protocol (HVMP) [1, 2] and the RelM protocol [3] are these protocols.

The HVMP augments the ability of MSSs to manage mobility of mobile hosts and to provide reliable delivery of multicast messages. A MSS is located in a cell. Each MSS communicates directly with mobile hosts, usually via wireless links. In addition, each MSS is responsible for buffering messages, processing ack messages for the various connections opened by the mobile hosts, tracking mobile hosts, and forwarding packets to its new cells. The disadvantage of this protocol is that the exchange cost of handoff messages is high whenever the mobile host moves between cells. The RelM protocol improves the performance by solving the disadvantage of the HVMP. The RelM protocol has a SH that control a group of neighboring MSSs and process most functions that MSS does in the HVMP with the exception of communicating with mobile hosts via wireless links. But if the mobile hosts within the domain of particular SHs are continuously crowded according to the regional property such as downtowns and suburbs, an ack implosion of these SHs can occur because of a number of ack messages from mobile hosts.

We propose a reliable multicast protocol in networks with mobile hosts. Our proposed protocol processes ack messages in both the MSS and the SH to alleviate the ack implosion of particular SHs. We expect the multicast groups to be very large. We assume that mobile hosts in a multicast group are lopsidedly located in the network and that both the fixed and the mobile networks have a relatively high transmission error rate. We also believe that applications must be unaware of the possible mobility of subscribers. This will make it possible to make all services available to mobile hosts, over high-speed networks, with no change at service providers. And we expect cell sizes to be small (100m) in order to accommodate high-bandwidth applications. Our proposed protocol may be

0-7695-0819-7/00 $10.00 © 2000 IEEE

implemented on top of the existing TCP/IP protocol suite, and it takes advantage of existing multicast protocols developed for the fixed network and message delivery and tracking within the mobile network and allows efficient and distributed adds and drops to the group.

The rest of this paper is organized as follows. Section 2 describes related work. Section 3 outlines the basic idea of our proposed protocol. Section 4 provides the details of our proposed protocol. Section 5 describes our simulation model and its' results. This paper concludes in Section 6.

2. Related Work

Many solutions to this reliable multicast problem have been proposed. The HVMP provides reliable multicast in networks with mobile hosts. In this protocol, architects for networks with mobile hosts view the system as a two-tier hierarchy, with mobile hosts comprising the lowest level and MSSs providing network connectivity to the mobile hosts. Mobile hosts can connect to the network from different locations at different times. Each MSS passes multicast messages to mobile hosts within its cell and receives ack messages related to the correspondent message via wireless links, buffering multicast messages for the various connections opened by the mobile hosts. If each MSS receives ack messages related to the correspondent message from all mobile hosts in a multicast group within its cell, each MSS delivers one ack up to the sender. Each MSS maintains ordering information of multicast messages from mobile hosts within its cell and the data structure related to ack messages. Each MSS tracks the mobile hosts within its cell. If the mobile hosts move from the previous cell to the new one, the MSS in the previous cell passes undelivered multicast messages, the ordering information of the correspondent message, and the information related to ack messages for the mobile host to the MSS in the new cell. The disadvantage of HVMP is that the exchange cost of handoff control messages is high whenever the mobile host moves between cells.

The RelM protocol has special nodes, SHs that control a group of the neighboring MSSs, manage mobility of mobile hosts, and provide reliable delivery of multicast messages. The RelM protocol has a three-tier hierarchy that consists of mobile host, MSS, and SH. In this architecture, the exchange cost of handoff messages is reduced whenever the mobile host moves between cells. The buffer usage is also decreased compared with the HVMP, since the roaming mobile host remains within the same SH domain for long periods of time. The detail functions of each component in the RelM protocol are described as follows. The MSS passes multicast messages received from the SH at the high level to mobile hosts within its cell and forwards ack messages for the correspondent message received from the mobile host in a multicast group within its cell to the SH at the high level, as the service of the base station that communicates with mobile hosts within its cell via wireless links. The SH forwards multicast messages to the MSSs that are controlled by itself and receives ack messges from the MSSs, buffering multicast messages for the various connections opened by the mobile hosts. If each SH receives ack messages related to the correspondent message from all mobile hosts in a multicast group within its domain, each SH delivers one ack up to the sender. The SH maintains ordering information of multicast messages that are delivered to mobile hosts within its domain and the data structure related to ack messages. The SH tracks the mobile hosts within its domain. If the mobile hosts move from the previous domain to the new one, the SH in the previous domain passes undelivered multicast messages, the ordering information of the correspondent message, and the information related to ack messages for the mobile host to the MSS in new one. But if the mobile hosts in a multicast group are continuously crowded around the domain of particular SHs according to the regional property such as downtowns and suburbs, an ack implosion of these SHs can occur due to a number of ack messages from mobile hosts. Therefore, the multicast group communication is limited by the imbalanced processing overhead of these SHs.

3. Proposed Reliable Multicast Protocol

3.1. Architecture

We assume that our proposed protocol has a three-tier hierarchy, with SHs plus mobile hosts and MSSs that are entities in traditional architecture for networks with mobile hosts, as shown in Figure 1 [3]. In this architecture, the SH instead of the MSS is responsible for tracking the mobile hosts. If mobile hosts move from the previous domain to a new one, the SH of the previous domain passes undelivered multicast messages, the ordering information of the correspondent message, and the information related to ack messages for the mobile hosts to the SH of the new domain. We expect that the exchange cost of handoff messages is reduced when mobile host moves between cells. The buffer usage is also decreased as compared with the HVMP, since the roaming mobile host remains within the same SH domain for long periods of time.

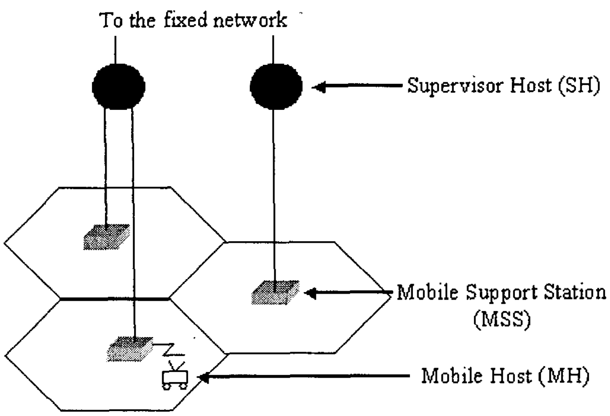


Figure 1.Three-tier model

The functions for each component are described as follows. At the lowest level are the mobile hosts, which move between cells. The mobile host is a host that can move while retaining its network connections. At the next level are the MSSs. Each MSS communicates with the mobile hosts within its cell via a wireless link. MSSs implement a cell discovery that enables mobile hosts to know the identity of its current MSS node. At the third level, a group of the neighboring MSSs is controlled by an assigned SH. The SH is part of the fixed network and handles protocol details for mobile hosts. In other words, the SH is responsible for forwarding multicast messages to the MSSs that are controlled by itself and receives ack messages from the MSSs and buffering multicast messages for the various connections opened by the mobile hosts. If each SH receives ack messages for the correspondent message from all mobile hosts in a multicast group within its domain, each SH delivers one ack up to the sender. The SH maintains ordering information of multicast messages that are delivered to mobile hosts within its domain and the data structure related to ack messages. The SH tracks the mobile hosts within its domain. If the mobile hosts move from the previous domain to the new one, the SH of the previous domain passes undelivered multicast messages, the ordering information of the correspondent message, and the information related to ack messages for the mobile host to the MSS of the new domain. We assume that each SH has sufficient memory to buffer multicast messages. In this architecture, if the mobile hosts within the domain of particular SHs are continuously crowded according to the regional property such as downtowns and suburbs, an ack implosion of these SHs can occur due to a number of ack messages from mobile hosts. The necessary state information to mobile hosts within the domain of the SH and the data structure related to ack messages is increased, and the imbalanced processing overhead among all the SHs in a multicast group is incurred because the overhead of the particular SHs that process ack messages is higher than that of other SHs in a multicast group. In reliable multicast group communication, the sender must send the next message to all the SHs, after collecting ack messages related to the previous message from all the SHs. Therefore, the multicast group communication is limited by the imbalanced processing overhead among SHs.

3.2. The basic idea of our proposed protocol

To solve the above problem, we propose an efficient reliable multicast protocol, in which both MSS and SH are responsible for processing ack messages from mobile hosts. Each MSS passes one ack message up to the SH that controls itself, after collecting ack messages from mobile hosts in a multicast group within its cell. The SH also forwards one ack message up to the sender after the SH receives ack messages from all MSSs, each of which have mobile hosts in a multicast group.

Figure 2 illustrates processing ack messages in our proposed protocol. S and Q denote SHs. A, B, A', and B' denote MSSs. m1, m2, m3, m4, m5, and m6 denote mobile hosts. A and B are controlled by S. A', and B' is controlled by Q. Acklist is the data structure processing ack messages from the mobile hosts. Arrows show the direction to process ack messages.

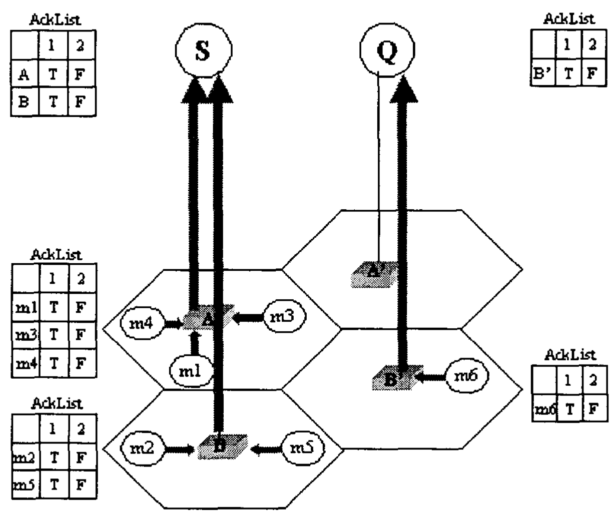


Figure 2. The ack messages processing model in our proposed protocol.

Each MSS maintains the data structure processing ack messages to mobile hosts within its cell. When a mobile host moves between cells of MSSs controlled by the same SH, the additional handoff messages are exchanged between two MSSs to delete the data structure to process an ack message from the correspondent mobile host in its previous MSS and to add the data structure in its current MSS. To remove the additional cost of handoff messages, we use a register message whenever the mobile host moves between MSSs. The SH forwards the register message with the necessary information related to the ack messages of the correspondent mobile host to a new MSS. For example, we assume that a mobile host moves from the previous MSS' cell into a new MSS'cell. After detecting that it is in a new cell by comparing signal strengths of different MSSs, the mobile host waits for the new MSS to transmit a beacon[1] message. It eventually transmits a greeting (the ID of the mobile host, the ID of the previous MSS, the ID of the previous SH, and the sequence number of the latest delivered message) to the current MSS. The current MSS passes the greeting message to SH, which sends the current MSS a message to register (the ID of the mobile host, the IDs of the multicast groups, the sequence number of the latest delivered message). This means that the current MSS adds the mobile host to its Acklist. The SH sends the previous MSS a message to deregister (the ID of the mobile host). The previous MSS deletes the mobile host's entry from its Acklist.

Figure 3 illustrates the idea for removing the additional cost of handoff messages within the domain of same SH. Q denotes SH and maintains m_R, ω_m, and Ω_m information. A mobil host can belong to several multicast groups simultaneously[2]. m_R denotes the set of multicast groups. ω_m is used during handoff[3] and indicates the next message to be transmitted to the mobile host for each of the mobile host's connections. Ω_m indicates the sequence numbers of all the messages of all multicast connections in m_R in this ordering as they come in. Multicast group Ri contains 1 and 3. Pij is the oldest undelivered jth message of multicast group Ri. A and B denote MSSs and are controlled by SH Q. MSS A has Acklist 1 and 3 in the left side of Figure 3. MSS B also has Acklist 1 and 3 in the right side of Figure 3. m1, m2, and m3 is mobile host and m1 is in A's cell and m2 and m3 are in B's cell in the beginning. Arrows indicate the basic handoff messages occurred by m2's moving from the previous cell (B) into the new cell (A).

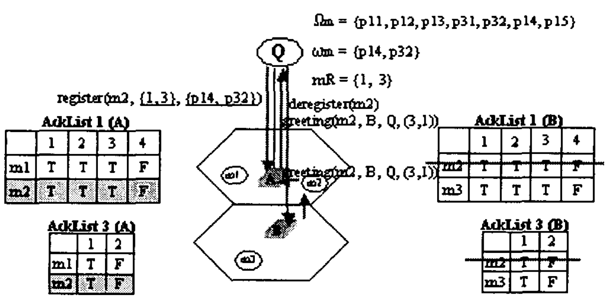


Figure 3. The idea to remove the additional cost of handoff messages within Q's domain

[1] MSSs periodically transmits a beacon message inviting all new mobile hosts to announce themselves.

[2] For example, we imagine a user receiving weather reports via one multicast connection and stock news via another.

[3] When mobile host moves one SH's domain into the domain of another SH.

We use register message with the necessary information related to ack messages of the correspondent mobile host to remove the additional cost of handoff messages occurred when m2 moves from the previous cell (B) into the new cell (A). Our proposed protocol alleviates the ack implosion problem that occurs in three-tier architecture, with the same cost of handoff messages as the RelM protocol.

3.3 Reliable Multicast Model on the Fixed Network

The sender sends multicast messages to all SH's in a multicast group to provide the mobility of the hosts, regardless of their location. Each SH that controls the sequence number of multicast messages delivered to mobile hosts and the location of mobile hosts within its domain passes multicast messages to the mobile hosts in the multicast group. Here, all the mobile hosts in a multicast group lie in the domain of some SHs. We call the set of SHs the host group. The sender delivers multicast messages to all SHs in the host group and each SH forwards the multicast messages to all the mobile hosts within its domain. The sender and all the SHs in the host group are part of the fixed network. Therefore, to support reliable multicast on the fixed network, our proposed protocol is needed to multicast from the sender to the host group. The proposed protocol also provides the added functions of signaling changes occurring in the host group, allowing dynamic adds and drops, and providing the ability to query for group membership.

Our model is based on the Single Connection Emulation (SCE) protocol [4] of the reliable multicast protocols. SCE provides reliable multicast delivery on the fixed network. It is implemented as a layer above multicast IP and below TCP. By adding the SCE sublayer, multicast may be supported without changing the original TCP protocol. For every multicast message sent, SCE collects ack messages from SHs in the host group and passes one ack message up to TCP in the sender when all the SHs in the host group have replied. Before connection setup, the multicast application will pass down parameters associated with the multicast session to SCE through the direct interface, as shown in Figure 4.

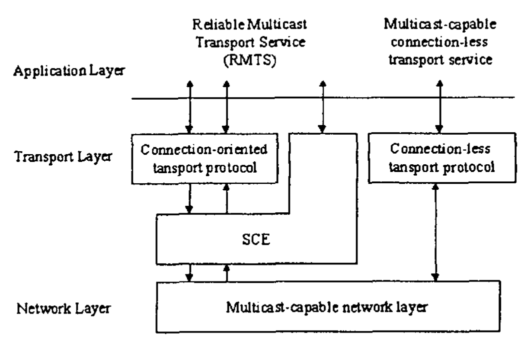


Figure 4. Reliable Multicast Model (from [4])

To retransmit multicast message due to the loss of packet, SCE maintains its own timer that can be derived from the upper TCP timer, and retransmission is handled by upper TCP protocol, which will multicast to the entire group.

Currently, SCE allows dynamic drops, i.e., a participant may drop out of the connection without affecting the rest of the participants and may rejoin at a later time. Dynamic joins, however, are not implemented in SCE due to the problem of defining the semantics of a member joining a reliable multicast connection. For our purposes, we would like to have the ability to allow dynamic joins and it is easy to define adequate semantics for our system. We require that after the SCE at the sender has established a connection to the new member, all future multicast messages will be sent to it. No guarantees are provided for delivery of old or current multicast messages.

When a SH wishes to join a host group, it contacts an existing SH of the group. This SH sends an add message containing the address of the new SH back along its connection with the SCE. Once the SCE has the new address, the add process is the same process used when the multicast connection is first set up.

Another modification to SCE is the ability to signal the SCE client (at the sender of the multicast) whenever new nodes join or leave the host group. This functionality is added by passing this information to the client via a new Groupchange system call. Finally, we can query the SCE about the current membership of the host group. Since the SCE has a list of host group member addresses it is easy to provide this information to the client.

4. The Details of the Proposed Reliable Multicast Protocol

Figure 5 illustrates the major modules of our proposed protocol.

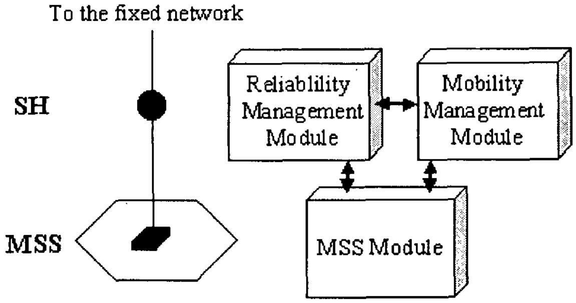


Figure 5. Overview of our proposed protocol

The notations of Table 1 are used in this section:

Table 1. The notation of our proposed protocol

Notation	Define
m	Mobile host
MSS (the ID of the SH, the ID of the MSS)	MSS is controlled by SH - The ID of the SH: S (previous SH), Q (new SH) - The ID of the MSS: a, b
GRi	The host group of multicast group Ri
msgij	The jth message of multicast group Ri
Ackij	The ack message for the jth message of multicast group Ri
Acklist(i, j)	The data structure processing the ack message for the mobile host's jth message of multicast group Ri
Location Table	The current location of each mobile host within S's domain
CSH	The counter variable per each SH of the host group maintained by the sender has the sequence number of the oldest undelivered multicast message

The details for our proposed protocol are described as follows.

4.1. Module at MSS

This module provides reliable unicast from the MSS to mobile hosts within its cell. This module forwards one ack messages up to the SH's Reliability Management Module at the high level, after collecting ack messages for the correspondent message from all the mobile hosts within its cell.

This module is also responsible for implementing the handoff procedure in conjunction with Reliability Management Module and Mobility Management Module for mobile hosts entering or leaving a cell. The protocol is best described by dividing its function into two parts.

A. Reliable Unicast Delivery of Message and Processing Ack Message

This part of the protocol deals with the problem of reliably transmitting messages to mobile hosts within a cell and forwarding one ack message up to the SH, after collecting ack messages for the correspondent message from all the mobile hosts within its cell.

1. Transmit msgij to all m within a cell and wait for ackij from m.

```
a. if (no ackij in timeout)
   then {
     do {
         retransmit msgij
     } while (ackij is not received || the MSS is informed
              that m has left its cell or dropped out of the
              multicast group Ri)
   }
```

2. When a msgij that is the jth message in multicast group Ri is received, a list called the Acklist(i, j) is associated with this msgij. This list contains a boolean entry for each m $\in$ Ri, initialized to false. When an ack is received for msgij from mobile host, this module changes the entry of mobile host from false to true in Acklist(i, j). When all entries of Acklist(i, j) are true, this module sends a delivered(j) message to the SH
3. When all entries of Acklist(i, j) are true, this module sends a delivered(j) message to the SH and delete Pij from the local buffer of the MSS.
4. m sets a LatestReceived variable to (i, j). This indicates the last multicast message received and is used to indicate the next message expected by the m, in the event of a move.

B. Handoff

We assume that m moves from MSS(S, a)'s cell into MSS(Q, b)'s cell. MSS(S, a) needs to be informed of the move so that it can clean up its data structure by discarding all references or messages for m. MSS(Q, b) needs to create entries for m. If S = Q, then the SH simply forwards all messages for m to MSS(S, a). IF S $\neq$ Q, then S needs to ensure that Q has all undelivered messages for m. Here we describe the part of the protocol that occurs between MSS(Q, b) and m.

1. m detects that it is in a new cell by comparing signal strengths of different MSSs and waits for the new MSS to transmit a beacon and eventually transmits a greeting(m, MSS(S, a), S, (i, j)) message to MSS(Q, b). This message contains the ID of the old SH and the old MSS as well as the sequence number of the last message received by the mobile host.
2. MSS(Q, b) passes this greeting message to Q.
3. if S = Q, then m is still within the domain of the same SH. If S $\neq$ Q, SH's Mobility Management Module completes a handoff between S and Q.
4. Q sends MSS(Q, b) a message to register(m, mR, (i, j)). This means that MSS(Q, b) adds m to its list of local mobile hosts. It also creates a data structure to store messages that need to be transmitted to m.
5. Q sends MSS(Q, b) messages starting after sequence (i, j).
6. MSS(Q, b) sends a init message to m containing the ID of Q and its own ID.
7. S sends a deregister(m) message to MSS(S, a). MSS(S, a) deletes m's entry from its list of local mobile hosts and discards all pending messages for m.

4.2. Modules at SH

This module is devided into two modules: Reliability Management Module, which performs reliable delivery of data, and Mobility Management Module, which manages mobility of mobile hosts within SH's domain.

4.2.1. Reliability Management Module

This module maintains message buffers, processes ack messages, maintains the message ordering information for each mobile host and determines when to discard messages from its buffers.

It also interacts with the modules at each MSS and with the Mobility Management Module to manage mobility. The functioning of this module is best explained by first describing its role at the source of a multicast and then describing its role at each SH belonging to a host group.

A. The sender in a multicast group

For every SH $\in$ G_{Ri}, the sender maintains C_{SH} (a multicast message is considered undelivered if there is at least one mobile host in Ri that has not yet acknowledged that message). Thus, if j is such that $j < C_{SH}$, $\forall SH \in G_{Ri}$, the multicast message with sequence number j has been delivered to all members of the multicast group Ri.

We assume that $msg_{ilatest}$ is a new message that needs to be multicast to members of Ri.

1. Muticast this message reliably to all SHs in the host group G_{Ri} using SCE and increment latest.
2. When a SH has delivered msgij to all mobile hosts within its domain, it transmits a delivered(j) message to the sender, Upon receiving a delivered(j) message from S, the sender sets C_{SH} to j + 1.
3. Let $v = \min_{SH \in G_{Ri}}\{C_{SH}\} - 1$. v denotes the highest sequence number of the message that has been delivered to all mobile hosts in Ri. The sender sends discard(v) message to all SHs in G_{Ri} indicating that all messages up to and including v may be discarded from their buffers.
4. Upon receiving a rescind(k) message from Q, C_Q is set to k. The sender does not buffer any messages itself. It only keeps track of the sequence number of the messages delivered to mobile hosts.

B. SH

1. if (msgij is received from the sender of multicast group Ri) then {
   - msgij is buffered.
   - Acklist(i, j) is associated with msgij. This list contains a boolean entry for each m $\in$ multicast group Ri within SH's domain, initialized to false.
   - For every m $\in$ multicast group Ri within SH's domain, insert a reference to msgij in Ω_m.
   - Select the next message from Ω_m and send to the MSS module of the current MSS of m.

2. if (ackij is received for msgij from m)
   then {
   - delete the reference to it from Ω_m.
   - change the entry of m from false to true in Acklist(i, j).

   }
3. if (all entries of Acklist(i, j) are true)
   then send a delivered(j) message to the sender of group Ri.
4. if (a discard(j) message is received from the sender of Ri)
   then discard msgij from the local buffer.
5. When m has moved into the domain of a new SH and all information regarding m (such as undelivered messages and Ωm) have already been forwarded to the new SH.
   - if (an old(m) message is received from the Mobility Management Module)
     then remove all entries referring to m from the different Acklists and Ω_m.
   - if (an new(m, m_R, ω_m) message is received from the Mobility Management Module)
     then {
     - create a new entry for m in the Acklists indicated by ωm and a new Ω_m.
     - Q did not belong to all host groups of m_R.
       if (some messages indicated in Ω_m are not present locally)
       then fetch them from the previous SH.

   }

4.2.2. Mobility Management Module

When a mobile host changes its location, the static segment is required to transfer the necessary state information from its previous SH to its current SH to facilitate delivery of multicast messages at its current location. This module processes adds and drops to allow dynamic and distributed adds and drops to the multicast group.

In case of the sender of a multicast group, it would form a multicast group and destroy a multicast group.

We assume that m moves from MSS(S, a)'s cell into MSS(Q, b)'s cell.

1. if (Q = S)
   then {
   - m remains within the domain of the same SH.
     if (m enters the new cell)
     then {
     - m sends a greeting(m, MSS(S, a), S, (i, j)) message to its new MSS.
     - The greeting message gets forwarded to the Mobility Management Module at SH.
     - A deregister(m) message is sent to MSS(S, a).
     - (i, j) is treated as an ack and is used to remove messages from Ω_m.
     - It is possible that m moved after receiving messages from MSS(S, a) but before sending back an ack message. The Mobility Management Module updates a location table that indicates the current location of each mobile host within SH's domain.

     }

} else {

A. Protocol at new SH Q

- Q sends S a move(m, Q, (i, j)) message indicating that m has moved.
- Q receives m_R and ω_m from S.
- if (Q $\notin$ GRi)
  then {
  - Q sends join(Q, Ri) message to S.
  - Q waits for add_confirm(Q, Ri, (i, k)).

  } else if (k < C_Q)
  then Q sends a rescind(k – 1) message to the sender of the multicast group.
- Send a new(m, m_R, ω_m) message to Reliability Management Module.
- The Reliability Management Module at Q adds m to AckList(i, j), for all (i, j) $\in$ ω_m. If msgij is not present locally, request it from S. After all msgij are received, Q sends S a remove(m) message.
- Q sends a register(m) message to MSS(Q, b)

B. Protocol at old SH S

- Upon receiving a move(m, Q, (i, j)) message from Q, S sends a deregister(m) message to MSS(S, a).
- S sends move_respond(m, m_R, ω_m) to Q after updating Ω_m and ω_m using (i, j).
- K is the sequence number of the oldest undelivered message for multicast group Ri. if (S receives a join(Q, Ri) message from Q)
  then S sends add(Q, Ri, (i, k)).
- When Q is a new member of G_{Ri}, if (S receives a request from Q for buffered msgij)
  then S sends msgij to Q.
- if (S receives a remove(m) message)
  then {
  - Remove all references to m.
  - The Reliability Management Module at S will remove m from all Acklists. Send an old(m) message to Reliability Management Module.

  }

- if (The Mobility Management Module at the sender of a multicast group Ri, upon receipt of add(Q, Ri, (i, k)))
  then {
  - The Reliability Management Module at S will remove m from all AckLists. Adds Q to Ri by making a call to SCE.
  - Messages starting at sequence number k get delivered to m exactly once (a discard(k) message has not yet been sent since S had not delivered msgik to m before m moved. The Reliability Management Module sets C_Q to k.
  - if (after the departure of m, no mobile hosts belonging to Ri lie within S's domain)
    then {
    - After the remove(m) message is received from Q, S sends a drop(S, Ri) message to the sender of multicast group Ri.
    - To remove S from the host group. The Mobility Management Module at the sender sends a delete(S) message to SCE.

    }

}

5. Simulations and Performance Evaluation

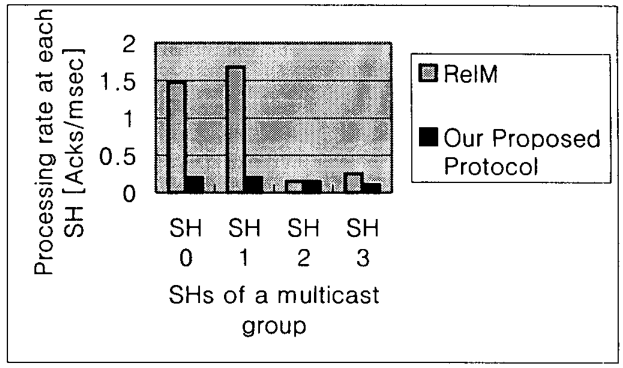
Our model is an event-driven simulation with objects; it models the behavior of the fixed sender, mobile hosts, MSSs, and SHs. The architecture of our system model consists of small system[4] (4 SHs and 16 MSSs) and a large system[5] (4 SHs and 36 MSSs). It is assumed that cells are square and the grid is connected at the top, bottom, and both sides. Cell bandwidth is assumed to be 2 Mbps, and there are always 10 mobile hosts in a cell, some of which may be multicast receivers. The TDMA slot size is 12 bytes. Table 2 summarizes the parameter settings in our simulated experiments.

Table 2. Simulated experiments

Parameter	Value
The number of mobile hosts per cell	Variable
Time between messages at sender	20000 μs
Average time a mobile host stays in a cell	20000 ms
Size of each multicast message	800 bits
Probability of loss on the wireless link	0.007968
Ratio of uplink/downlink channels	0.2/0.8
Length of time to simulate	60000 ms

[4] Each SH controls four MSSs in a small system.
[5] Each SH controls nine MSSs in a large system.

Figure 6 presents the comparison between the RelM protocol and our proposed protocol, with the processing rate of ack messages processed at each SH in a multicast group for a given period of time.



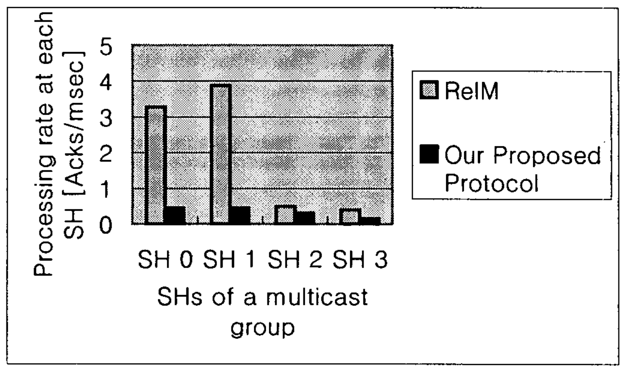


Figure 6. The processing rate of each SH in a small system (top) and a large system (bottom).

Figure 6 shows the imbalanced processing overhead among SHs, in which all the mobile hosts in a multicast group lie. We assume that SH 0 and SH 1 are located in the downtown areas, while SH2 and SH 3 are in the suburb areas. The mobile hosts are continuously crowded around SH 0's domain or SH 1's due to the regional property. In this case, the RelM protocol shows that the processing overhead of SH 0 and SH 1 is higher than SH 2 and SH 3, the rest of the multicast group. In reliable multicast communication, the sender must send the next message to all the SHs, after collecting ack messages related to the previous message from all the SH. Therefore, as the overhead of these SH 0 and SH 1, which process a number of ack messages, is overly increased, as shown in above Figure 6, the multicast group communication is limited by the imbalanced processing overhead of SH 0 and 1. Our proposed protocol, however, shows that the processing overhead of all the SHs remains almost the same, regardless of the regional property. We expect that the multicast group is smoothly communicated by the balanced processing overhead of all the SHs.

6. Conclusion

We have proposed a protocol to provide reliable 1×N multicast to mobile hosts. This protocol has a three-tier architecture, in which an ack implosion of these SHs can occur due to a number of ack messages from mobile hosts and multicast group communication is limited by the imbalanced processing overhead among SHs in multicast group, if the mobile hosts within the domain of these particular SHs are continuously crowded according to the regional property such as downtowns and suburbs.

We propose an efficient protocol to solve this problem, in which both the MSS and the SH process ack messages from mobile hosts. In conclusion, the processing overhead of the SH is almost the same, regardless of the regional property with the same cost of handoff messages as the RelM protocol.

Reference

[1] A. Acharya and R. Badrinath, "Delivering Multicast Messages in Networks with Mobile Hosts", 13th Intl. Conf. on Distributed Computing Systems, 1993.

[2] A. Acharya and B. Badrinath, "Framework for delivering multicast messages in networks with mobile hosts", *ACM/Baltzer Journal of Wireless Networks*.

[3] K. Brown and S. Singh, "RelM: Reliable multicast for mobile networks", *Computer Communications Vol. 21 No. 16*, 1998, pp. 1379-1400.

[4] R. Talpade and M.H. Ammar, "Single Connection Emulation: An Architecture for Providing a Reliable Multicast Transport Service", in Proceedings of the 15th IEEE Intl. Conf. on Distributed Computing Systems, Vancouver, 1995.

[5] K. Brown and S. Singh, Multicasting in Mobile Networks, Technical Report, Department of Computer Science, University of South Carolina, Columbia, SC 29208, 1995.

[6] G.S. Poo and A.M. Goscinski, "Performance Comparision of Sender-based and Receiver-based Reliable Multicast Protocols", *Computer Communications Vol. 21 No. 7*, 1998.

[7] D. Towsley, J. Kurose, and S. Pingali, "A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols, *IEEE Journal on Selected Areas in Communications Vol. 15 No. 3*, 1997.

[8] J. Wei, A Survey on Reliable Multicast Protocols for Internet, Technical Report, Hughes Research Laboratory, Malibu, CA, 1997.

[9] A. Acharya, A. Bakre, and B.R. Badrinath, "IP Multicast Extensions for Mobile Internetworking", Proc. of the IEEE Infocom 96, San Francisco, 1996, pp. 67-74.

[10] S. Deering, Host Extensions for IP Multicasting, RFC 1112, 1989.

[11] H. Eriksson, "MBONE: The multicast backbone", *Communications of the ACM Vol. 37 No. 8*, 1994 pp. 54-60.

[12] R. Ghai and S. Singh, "An Architecture and Communication Protocol for Picocellular Networks", *IEEE Personal Communications Magazine Third Quarter*, 1994, pp. 36-46.

[13] S. Armstrong et al., Multicast Transport Protocol, RFC 1301, 1992.

[14] B. Badrinath, A. Acharya, and T. Imielinski, "Structuring Distributed Algorithms for Mobile Hosts", 14th Intl. Conf. on Distributed Computing Systems, 1994.

[15] S. Deering and D. Cheriton, Host Groups: A Multicast Extension to the Internet Protocol, RFC 966, 1985.

[16] B.N. Levine and J.J. Garcia-Luna-Aceves, "A Comparison of Reliable Multicast Protocols", *Multimeida Systems Vol. 6 No. 5*, 1998, pp. 334-348.