Proximity Groups for Mobile Ad Hoc Networks

Kulpreet Singh, Siobhán Clarke, Andronikos Nedos, Vinny Cahill

Distributed Systems Group,

Department of Computer Science,

Trinity College Dublin, Ireland

1 INTRODUCTION

With the developments in the field of wireless ad-hoc networks combined with low-cost communication and computational devices, pervasive environments [11] are closer to reality than ever. As such environments become more prevalent, they will simply be "invisible" to the user, and as a result, he will not distinguish wireless, ad-hoc infrastructures with wired, fixed ones. From the user perspective, the fact that the network now supports mobility means that those group-based activities enjoyed with the traditional PC-based paradigm should, of course, be available when roaming with a mobile device. In the not-so-distant future, teleconference meetings, online augmented reality gaming and other "collaborative work" [9] activities with mobile participants will seem perfectly reasonable.

To run applications in pervasive environments a need exists for communication between mobile devices or a mixture of mobile and stationary devices. This enables a user to access shared resources in a pervasive environment, and/or to run applications that involve a number of other mobile or static devices present in the environment. Online augmented reality games and teleconference meetings are example applications.

What such applications have in common is the need for short range, localized exchange of messages between participants that could be either mobile or stationary.

Our motivation in this paper is to support reliable group communication between mobile devices. Traditional, one-to-many communication systems running on fixed wired infrastructures offer ordering and reliability guarantees for group communications [3]. We envisage that many ubiquitous application areas will benefit from a wireless, ad hoc infrastructure offering the same kind of guarantees. A wireless network in ad hoc mode enables communication without the need for a fixed infrastructure, increasing the coverage possibilities for ubiquitous applications. Difficulties exist, however, in providing communication guarantees in such an environment. These difficulties relate to the high degree of mobility, and power/bandwidth limitations. The unpredictability of the environment, together with potentially unconstrained resource demands, make providing communication guarantees more challenging. For many of the pervasive application areas we consider, communicating within a relatively close region is appropriate. We therefore use proximity as the basis for increasing the guarantees that are possible with mobile ad hoc networks.

Our research is based on providing acceptable group communication guarantees for such an environment. There are two major elements to any group communications service. First, there is a "membership" part – i.e., what users/devices are in the group. The other part is delivery of messages to the members of the group. In the next sections, we discuss the requirements for a group communication service in a mobile ad hoc network, and provide a description of the proximity-based membership service subsystem that is the focus of our research. The other major element of any group communication system, message delivery, will be briefly mentioned but will not be discussed as it is an area where much research has happened and solutions to known problems have appeared [14, 15, 16].

2 PROXIMITY GROUPS FOR MOBILE ENVIRONMENTS

Part of the requirements for Proximity Groups is to provide a location aware membership service to enable group communication in a wireless ad hoc network. Such a membership service maintains the list of active group participants and notifies these participants about changes in the list (e.g., new members joining and existing members failing or being disconnected). Once a membership service that provides the guarantees/functionalities described in this document is available, it can be utilised to develop a delivery service that transmits application messages. This delivery service may take advantage of the guarantees that the membership service provides. The delivery service may, therefore, be implemented without worrying about the details of how the membership view is obtained, or its accuracy.

Before we discuss Proximity Groups in detail, we first describe some challenges of mobile ad hoc networks that place additional requirements on the membership service that are not present for traditional ones. Such challenges are:

- frequent switches to and from power saving mode, resulting in a high number of membership change messages,
- high degree of mobility, resulting in frequent partitions in the group memberships,
- power/bandwidth limitations that require a robust, modular and lightweight design with built-in power awareness features.

Even within such an environment, the group communication service described here should include a membership model that:

- is dynamic,
- supports multiple groups,
- is reliable,
- supports group mobility,
- is partitionable.

A *dynamic* model for group membership supports a node (user or device) leaving or joining a group at any point in the group's lifecycle. This supports, for example, players leaving a game, or meeting attendees arriving late. It also includes handling nodes that fail within a group, and therefore are no longer in a position to communicate.

It should be possible for a node to be a member of *multiple groups* at the same time. For example, a user may be currently involved in a mobile game, but is also a member of a different group that collaborates for creative writing in a public display.

Reliability of communications is important for the semantics of many group-based applications. Suppose that a Player A joins a shoot-em-up style mobile multiplayer game. For the game to function, it is important that the order in which Player A shoots and some Player B ducks is communicated precisely to all players. This has an obvious impact on whether Player B is deemed to have been shot by Player A. Other applications might require more relaxed reliability semantics. Thus, in order to obtain the flexibility to cope with differing reliability requirements, the Group Communication System should support the primitives for communication with FIFO, Causal or Total Order semantics [3]. These properties are part of the message delivery subsystem that complements the membership subsystem and will not be discussed further here.

A group may be *mobile* or *static*. In other words, the geographical region that the group relates to may be mobile (e.g., all nodes in proximity of a moving ambulance) or static (e.g., all nodes in the proximity of a particular playing area may be part of a game).

A group *partitions* if a subset of its members move away from the proximity of the group. In such a situation, partitioned members of the group may want to keep interacting with the application even if all the members of the group are not reachable. Anticipating such partitions is an important element of partition management, as an elegant, reliable application response to a partitioned node may then be possible. Members may be informed of the partition, and partitioned members may continue communicating with each other. By returning to the proximity of other partitions in the group, those partitions are merged. Without prior anticipation, a node is deemed to have failed if it moves away from the proximity of the group, or actually fails. In this case, such a node is no longer deemed to be communicating within the group.

2.1 WHAT IS A PROXIMITY GROUP?

A proximity group (PG) is a set of communicating nodes whose membership is determined by their geographical proximity, the membership policy of the group, and the matching of interests of the node with the group. When we refer to a PG we refer collectively to all the nodes participating in it. Once a PG is formed, the members of the group start communicating by exchanging messages with each other. Properties of a group are shown in Table 1.

2.2 PROXIMITY GROUP MEMBERSHIP SERVICE RESPONSIBILITIES

The Proximity Group Membership Service (PGMS) described in this paper manages membership of mobile nodes. In other words, it reliably informs a client group communication service of the currently active list of members of the group to which messages should be sent. Therefore, the PGMS has the following responsibilities:

1. Membership maintenance: Support for individual nodes discovering, joining, creating and leaving groups.
2. Mobile group membership: Where the navel of a group is mobile, ensure that the membership of the group is consistent with the moving geographical space.
3. Failure and recovery detection: Ensure the active members list of the group takes into account failed and recovered nodes.
4. Consistent view: Ensure that all members of the group have a consistent view of the membership.
5. Partition management: Ensure that the members of a group are notified of changes in membership as a result of both partitions and of merging of partitions, and that these membership change messages are ordered appropriately.
6. Agreement: If a message is delivered to any arbitrary member of a group, it is eventually delivered to all other members.

These responsibilities are common to many membership services in group communication systems [14, 15, 3, 6]. However, this research is focussed on extra challenges posed by the nature of mobile ad hoc networks. In the following sections we elaborate on each of the above responsibilities.

Property	**Description**
Group Identity	Each group has an identity associated with it which is used by nodes to identify the group.
Policies	A group has a set of policies it may implement to allow membership to the group. For example, one policy may dictate that all members have to be within a pre-specified region of the *navel* (see below).
Interests	A group has one or more interests. Nodes join goups based on common interests. For example, a PG might only allow nodes to join if the nodes represent tourists located in the proximity of a coffee place.
Active Members	Each group has a list of active members in the group.
Proximity	Each Proximity Group is associated with a region within which new nodes can join the group.
Navel	Proximity Groups have a fixed or moving point in space as the navel of the group. The PG service determines group membership based on geographic proximity to the navel regardless of whether it is mobile or static.

Table 1: Proximity Group Properties

2.3 Membership Maintenance

This section explains how the PGMS enables individual nodes to discover, join, create and leave communication groups.

2.3.1 Discovering Groups

A node can enter an ad hoc network either by physically moving into the geographical region where connectivity is available or by coming back online from a power saving mode or a previous failure.

When a node enters an environment where some groups already exist, the node first discovers the existing groups. The membership service of the groups present in the environment responds by letting the new node know what are the *group ids* of these groups and what *interests* they represent (see [1]).

2.3.2 Joining Existing Groups

After a node that is new to the network has obtained the list of groups available in the environment, it sends a request to *join* specific groups based on the *interests* specified by the groups. When the group membership service of the group that the new node wants to join receives the request, it is able to determine whether the new node is within the *proximity* of the group, and conforms to the general policies of the group, and thus whether it can join.

If the membership service running on the nodes of a group determines that the node can be a part of the group, it allows the new node to join the group and send a membership change message to the new set of group participants [6]. This set of members now also contains the new node as a member.

The new node can then start communicating with the other members. How the application and the membership change messages are ordered is an important issue. We will discuss this point and the ordering between the two further in section 2.6.

A guarantee is needed that Join messages are eventually reported to all the members of the group. This is important so that all members of the group have a consistent view of the participants.

2.3.3 Creating New Groups

When individual nodes want to create a PG, they have to first arrive within each other's *proximity.* As the nodes arrive into each other's *proximity,* the group membership service components on each of these nodes initiate a startup protocol. The membership service components on these nodes start broadcasting a "create group" message. When the individual nodes that initiated the startup receive each other's "create group" messages they are merged by the membership service as though they were different partitions of a group. The merging of these nodes is handled in the same way as the merging of different partitions; this is further elaborated in section 2.7.

Once these nodes have been merged into one group, a consistent membership view of the group is delivered to each of them. No application messages are delivered until all the nodes of the group have received the new group membership view.

2.3.4 Leaving a Group

Leaving a group is quite straight forward. If a node wants to leave a group, it initiates a "leave group" message. This "leave group" message is delivered to all the members of the group. The other nodes are thus aware that this node has left the group, and the group can keep communicating aware of the node's departure.

A problem arises if a node crashes and is not able to send out a "leave group" message. In such a case the group membership service detects the failure and sends out a membership state change message to the rest of the group. See section 2.5 for more details.

2.4 Mobile Group Membership

One of the responsibilities of PGMS is to determine the physical geographic region around the group within which new nodes may join. Where the group is mobile, the geographic region will change as the group moves. We refer to this region as the group's "proximity". As the group moves, the PGMS dynamically changes its proximity and allows nodes from within the new area to join the group. The "navel" of the group, which may be mobile, is used to calculate the group's proximity.

On the other hand, if some members of the group move away from the navel, the partition anticipator detects this movement and the PGMS *partitions* the group. This now associates two distinct geographical areas with the group. In this case nodes from these different areas can join the different partitions of the group. This enables the two partitions to proceed independently and the partitions can later merge. Partition handling is explained later in section 2.7.

2.5 FAILURE AND RECOVERY

The PGMS should be able to detect if a node has failed. Once the service has detected a failure, it delivers a membership change message to the rest of the members of the group. This indicates to the rest of the group that the failed node is no longer in the group. The delivery of such a membership change message to all the other members of the group is guaranteed by the PGMS.

If a node has gone out of the communication range of the Proximity Group without the system anticipating such an occurrence, the proximity group membership service detects this as a failure and an appropriate membership change message is delivered.

When a node recovers from a failure it is synchronised with the rest of the group. To achieve this, the state of the group membership is transferred to the recovering node after the node has successfully reported its recovery. A membership change message is delivered to the rest of the group members only after the recovering node has obtained the present state of group membership. This lets the PGMS guarantee that the node recovery was complete.

2.6 CONSISTENT VIEW

The PGMS delivers the membership change messages such that the receiving node is guaranteed that all other nodes of the group have either received the message or are in a transition state just before receiving the message. This enables the receiving node to continue with its task, assured that other members have the same view of the group at that particular stage in the computation.

At the time of merging partitions, the PGMS guarantees the order between the application and membership change messages. This is discussed further in section 2.7, after we have elaborated on the partition handling by the PGMS service.

2.7 PARTITION HANDLING

The PGMS communicates with a component that anticipates partitions [7]. Such a “partition anticipator” *anticipates* that a partition is likely to occur because the members of the group are moving away from each other's proximity. The PGMS allows a partitioned group to simultaneously proceed in different partitions, as long as it has been warned by the partition anticipator about such partitions.

Partitions of groups may return to each other’s proximity, whereupon they are merged. The service then delivers a merge message to the group as a whole. The service further makes sure that the application and the membership change messages are delivered in a meaningful order. This order is especially important for some applications.

The order that the PGMS strives to achieve is as follows:

1. deliver messages sent before merge request,
2. deliver merge message,
3. deliver messages sent after merge request.

For anticipated partition, both partitions can proceed with their tasks and can be synchronised later. The synchronisation of the application states in the two partitions is left to be specified by the application and is not dealt with by the PGMS. The membership service simply provides a new membership view to all the nodes that have merged.

The above properties enable an implementation of *creating new groups* as stated in section 2.3.3. The individual nodes can be treated as different partitions that are merging, and the procedure followed to merge partitions can be applied for starting a group. Thus, in the absence of prior knowledge of the group membership, new nodes use this bootstrapping mechanism to create a new group.

In contrast to the above mechanism if a set of nodes move away from the group so that they lose communication with the group and the partition anticipator does not anticipate this move as a partition, the PGMS treats the nodes that moved away as having failed.

2.8 AGREEMENT

The PGMS guarantees that if one of the members receives a membership change message, all the other members receive the message too. This is a fundamental property called *agreement* and helps provide a consistent view of the group membership to all the members.

3 CURRENT WORK

In this paper we have given a description and a set of properties that must be fulfilled for reliable communication over highly dynamic and unpredictable ad hoc networks.

We are currently working to specify the detailed semantics and technical design for the Proximity Group Membership Service specified above. It will be implemented using IEEE's 802.11 hardware running in an ad-hoc mode. This service will be evaluated by comparison with existing Group Membership Services ported from a fixed and wired infrastructure to mobile ad hoc networks.

4 REFERENCES

[1] Kenneth P. Birman. Building Secure and Reliable Network Applications. Manning Publications Co., 1996.

[2] IEEE Standards Board, editor. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Number IEEE Std 802.11-1997. IEEE Computer Society, 1997.

[3] Gregory V. Chockler, Idit Keidar, and Roman Vitenberg. Group communication specifications: A comprehensive study. ACM Computing Surveys (CSUR), 33(4):427-469, 2001.

[4] Alan Fekete, Nancy Lynch, and Alex Shvartsman. Specifying and using a partitionable group communication service. ACM Transactions on Computer Systems (TOCS), 19(2):171-216, 2001.

[5] Gregor Gärtner. Group Communication for flare. Technical report, Distributed Systems Group, Trinity College Dublin, 2002.

[6] Matti A. Hiltunen and Richard D. Schlichting. Understanding membership. Technical Report TR95-07, Computer Science Department, The University of Arizona, 1995.

[7] R. Meier, M.-O. Killijian, R. Cunningham and V. Cahill. Towards proximity group communication. In Workshop on Middleware for Mobile Computing (IFIP/ACM Middleware), 2001.

[8] Niels Reijers and Vinny Cahill. A formal specification of a group communication service for flare. Technical Report TCD-CS-2001-46, Distributed Systems Group, Trinity College Dublin, 2001.

[9] Robert Simon, Robert Sclabassi, and Taieb Znati. Communication control in computer supported cooperative work systems. In *Proceedings of the conference on Computer supported cooperative work*, pages 311-321. ACM Press, 1994.

[11] M. Weiser. The computer for the 21st century. In *Scientific American*, volume 265, pages 66-75. 1991.

[12] J. Broch, D.A. Maltz, D.B.Johnson, YC Hu, J. Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. 1998.

[13] L. Blazevic_, L Buttyan, S. Capkun, S. Giordano, J.P. Hubaux and J.Y. Le Boudec. Self-Organization in Mobile Ad-Hoc Networks: the Approach of Terminodes. IEEE Communication Magazine, June 2001.

[14] E. M. Royer and C. E. Perkins, Multicast Operation of the Ad hoc On-Demand Distance Vector Routing Protocol, *Proceedings of MobiCom '99*, Seattle, WA, August 1999, pp. 207-218.

[15] D. Dolev, D. Malki. The Design of the Transis System. Technical Report 95-4, Institute of Computer Science, The Hebrew University of Jerusalem, 1995

[16] Robbert van Renesse, Kenneth P. Birman, Silvano Maffeis. Horus: A FlexibleGroup Communications System. Communications of the ACM, 39(4):76--83, April 1996.