TransMAN: A Group Communication System for MANETs

Kulpreet Singh Andronikos Nedos Siobhán Clarke
Distributed Systems Group, Computer Science Department,
Trinity College Dublin, Ireland

1. Introduction

Mobile ad-hoc networks are characterised by frequent topology changes due to node mobility and node failures. These frequent topology changes increase the difficulty of providing reliability guarantees in MANETs. A group communication system [1] that provides all participating nodes with a consistent membership view while providing reliable and ordered communication will be useful while developing distributed applications in MANETs. This poster describes TransMAN, a group communication system designed for mobile ad hoc networks. TransMAN is designed to address frequent topology changes and to take advantage of the IEEE 802.11 MAC's broadcast based communication.

TransMAN provides a partitionable group communication service with FIFO reliable broadcast (and an optional total order semantics) and consistent membership view maintenance with virtual synchrony (VS). VS allows nodes that survive the same group view changes to deliver the same set of messages. This is useful in MANETs as nodes that remain in a group can continue uninterrupted while some other nodes join or leave the group. TransMAN provides a much needed service for developing distributed applications with reliability guarantees. The target applications for TransMAN are multiplayer mobile games and collaborative work in MANETs.

Initial performance results show the message delivery with FIFO guarantees remain below 500*ms* for networks of up to 5 hops. The result shows that TransMAN is highly usable for the class of applications it is targeted for.

2. Architecture

Figure 1 shows the TransMAN architecture diagram. The bottom layer implements a reliable broadcast protocol [2] that is designed to suit the needs of TransMAN. Messages delivered by the broadcast layer are buffered in the stability layer, where they are kept until they are known to be delivered to all other nodes in the network. The membership agreement and the failure detection layers are notified when a message is being delivered. The failure detection layer uses the lack of regular notifications to suspect a failed node, while the membership agreement layer utilises these notifications to determine changes in the group membership. The membership layer also uses suspicion information from the failure detection layer.

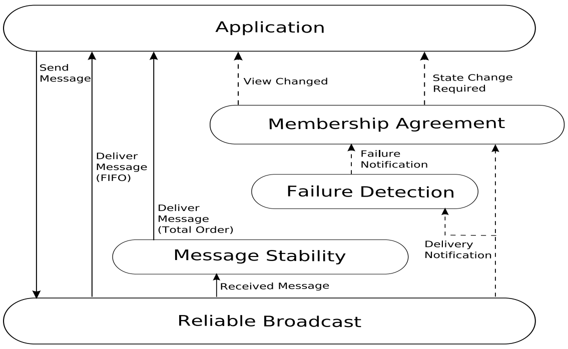

Figure 1. TransMAN architecture

2.1. Handling Mobility

One of the key novelties of TransMAN is the 2-phase approach to membership view changes. The 2-phase approach allows TransMAN to handle frequent topology changes without generating many extraneous messages. A node enters the first phase when it suspects that a view change is required. In the first phase a node determines the next possible view. Once a node has determined a tentative view it enters the second phase. During the second phase nodes run an agreement protocol to confirm the next view. Once an agreement is reached on a tentative view, nodes install the tentative view as the next membership view.

Figure 2 shows the state transitions for the 2-phase view change protocol. In the figure, *tviews* is the list of tentative views that can be installed next. The

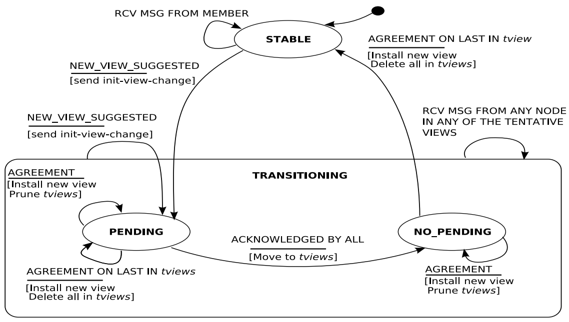

Figure 2. State transition diagram for membership protocol

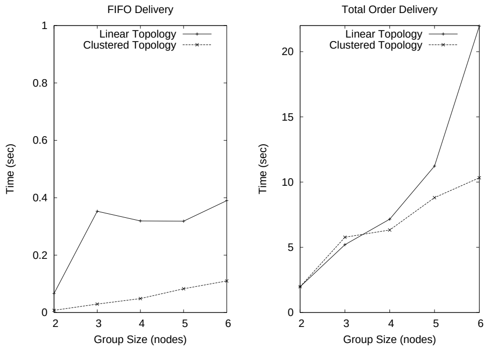

Figure 3. Delivery Times

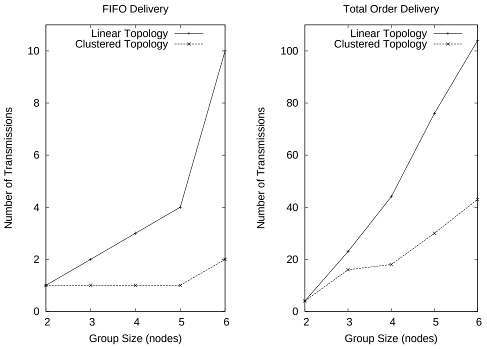

Figure 4. Delivery Xmits

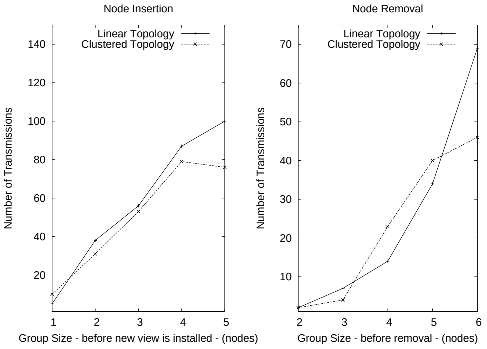

Figure 5. View Changes

STABLE state shows the node is not anticipating any view changes while the TRANSITIONING state shows that a node is aware of a tentative view. The TRANSITIONING state has two sub-states, PENDING and NO_PENDING, which correspond to the state of agreement for the tentative views.

3. Performance

Each laptop runs Linux 2.4.27 on a Pentium III 1GHz processor with 256MB memory. The system implementation is developed in the Ruby programming language. The broadcast layer at each node transmits a stream of messages separated by randomly chosen time periods between the range of 500*ms* to 1500*ms*. Each regular broadcast message is 200 Bytes in size.

Two network topologies are chosen, linear and clustered. For the linear case we set up a network of n nodes such that there are $n-1$ hops between the furthest two nodes. To achieve this topology, we remove external antennas from the Cisco (aironet 350) IEEE 802.11b cards. Once the external antennas are removed each node has a transmission range of about 2 feet. This allows us to set up linear networks inside our lab. Our most extreme network setup, a linear network with 6 nodes, has 5 hops. An IEEE 802.11b based MANET of 5 hops will extend to more than a Kilometre.

Figure 3 shows the delivery times for FIFO and Total order delivery for linear and clustered networks. The experiment shows FIFO delivery of messages within 500*ms* for networks of up to 5 hops. Total order delivery shows rapid deterioration with number of nodes. Figure 4 shows the same result as Figure 3 but in terms of number of message transmissions. Figure 5 captures the response times for membership view changes with varying network sizes. The experiment show only a linear increase of response times with increasing group size.

References

[1] G. V. Chockler, I. Keidar, and R. Vitenberg, “Group communication specifications: A comprehensive study,” *ACM Computing Surveys(CSUR)*, vol. 33, no. 4, pp. 427–469, 2001.

[2] K. Singh, A. Nedos, G. Gaertner, and S. Clarke, “Message stability and reliable broadcasts in mobile ad-hoc networks,” in *ADHOC-NOW*, V. Syrotiuk and E. Chávez, Eds. LNCS 3788, October 2005, pp. 297–310.