Tailoring Group Membership Consistency for Mobile Networks

Sebastian Gutierrez-Nolasco,
Nalini Venkatasubramanian
Dep. of Computer Science
University of California Irvine
Irvine, CA 92697, USA
seguti@ics.uci.edu,
nalini@ics.uci.edu

Carolyn Talcott,
Mark-Oliver Stehr
Computer Science Laboratory
SRI International
Menlo-Park, CA 94025, USA
clt@cs.stanford.edu,
stehr@csl.sri.com

ABSTRACT

Mobility and intermittent connectivity inject inaccuracy in determining group membership and exacerbate the time required to agree on the current group membership. In this paper, we present a group membership service based on partial member connectivity that allows members to agree on a shared approximation of the group membership based on local neighborhood connectivity. In particular, the consistency needs provided by the application determine the degree of consistency of the membership service and allow the membership service to tailor the neighborhood service in terms of fidelity ratio and time detection period.

KEYWORDS: Group Communication, Mobility.

1. INTRODUCTION

Many collaborative and groupware applications require messages (and data) to be sent to multiple destinations (nodes) in an efficient and consistent manner. These applications ideally run on top of a group communication system (GCS), which provides membership service and message delivery semantics. The membership service maintains information about the nodes that belong to the group and delivers the message to them. In addition, message delivery semantics may require members to communicate in an ordered manner. With advances in wireless networking technologies, mobile ad-hoc networks (MANETs) and delay-tolerant networks (DTNs) have been an active research area. These networks are formed by a collection of mobile wireless nodes that cooperate in order to achieve a global task. Communication links form and disappear as nodes come into and go out of each other's communication range. Most of the work on these networks have focused on routing and medium access control protocols and mechanisms, due to their impact on performance [1,2]. Much less work has been done on the development of distributed services, such as group communication. Group communication is an important building block for applications that involve collaborations among a group of people, since it allows the application to be shielded from the dynamic network environment. However, some mobile applications, such as search and rescue, wildlife tracking and military scenarios, require both order and location of occurrence of events to be taken into account for accurate decision making. We believe that the dichotomy between application transparency and context awareness can be bridged only if the GCS is capable of (i) dynamically adapting its behavior to fit the changing needs of mobile applications, and (ii) handling transparently unpredictable environment changes.

Providing group communication becomes especially difficult in mobile networks wherein (1) intermittent link availability and (2) node mobility produce frequent network topology changes, and (3) the lack of transitive closure of the neighbor relation induces asymmetric perceptions of network connectivity. These aspects hinder the three key features of group communication: consistent group membership, message delivery guarantees and high-level communication services. In the case of group membership, different members may have contradicting and asymmetric perceptions of the group's membership according to their relative location. This injects inaccuracy in determining the group membership and exacerbates the time required to agree on the current membership. Similarly, preserving message delivery ordering guarantees is challenging due to constant topology changes, high message loss, unpredictable message delay and the asymmetric perceptions of the group's membership.

Previous approaches to provide group communication in mobile networks have extended traditional GCS to deal with mobility by constraining the environment and imposing strong synchronization guarantees to predict failures and ensure reliable communication [3–5]. Consequently, they (1) rigidly enforce a well-defined and consistent group membership at every member, (2) delay membership join/leave operations while other messages are being propagated, so they can be ordered with respect to messages from other members (serialized transactions) and (3) deliver a message only to those members the sender thought were part of the group when the message was sent (virtual synchrony). This degree of synchronization limits the performance and scalability of the GCS and force applications that could otherwise use weaker synchronization and ordering guarantees to wait unnecessarily for message delivery. In addition, intermittent and short-lived connectivity fluctuations cause extra messages to be sent and received in order to agree on a consistent group membership, further delaying message delivery to the application. Fortunately, many collaborative applications do not require stringent synchronization guarantees, but they still need membership information to accomplish their task. Examples of such applications are location services and uniform quorums [6], service discovery using DHT [7] and random overlay constructions [8]. Since these applications currently use a random (flat) approach to membership, we believe that they can benefit from a relaxed notion of group membership to locally approximate their surrounding connectivity, use that information to divide members in two categories and provide different levels of message delivery semantics for each category. This enables the application to keep local consistent information with a subset of (nearby) members and provide probabilistic dissemination of this information to the remaining (remote) members. In this paper, we relax the assumption that members are fully connected and define a membership service based on partial member connectivity. We then define an application sensitive neighborhood service that allows members to locally approximate the surrounding neighborhood connectivity. This approximation is specified in terms of distance from the member and faithfulness of the information. Finally, a hierarchical membership based on partial member connectivity that allows members to agree on an approximate yet consistent snapshot of the group membership.

2. THE GROUP MEMBERSHIP PROBLEM FOR MOBILE NETWORKS

Traditionally, a group G is formed when one or more (application) clients agree on a group identifier and an initial snapshot of the group membership, called the initial view. Clients can join or leave the group at any time by issuing a membership operation to add/remove one or more clients to/from the group. Every issued membership operation, as well as changes in the network topology, modify the membership of the group. The task of a traditional membership service is to maintain a consistent group membership and report changes to every connected member via a new view. Usually, a view includes a view identifier and a list of currently active and mutually connected members. Thus, members that receive the same view are thought to have the same perception of the group's membership. Although a consistent membership service works well in wired environments, it is expensive to maintain in mobile networks due to its full member connectivity assumption. That is, it requires every member to keep track of every other member regardless of their spatial location, and frequent membership changes impose high message loads. Therefore, it seems natural to relax the full-connectivity assumption and define a membership service that takes advantage of partial member connectivity to minimize the communication overhead required to maintain the group membership.

We start by considering a mobile network where each node has a unique identifier. Changes in network topology are arbitrary, nodes may fail and recover, all transmissions are omni directional with the same transmission range and nodes can only communicate by exchanging messages over wireless links. Messages can be lost with a non-negligible probability *pl* and the delay of message delivery within the transmission range is bounded by some known value *hdel*. We further assume that nodes do not know their positions and do not use any geographic knowledge. This mobile network can be modeled as a graph $MN = (V,E)$, where V represents the set of mobile nodes and E represents the set of links. A link between nodes $i, j \in V$ means that i is within transmission range of j, and vice versa. Since not all nodes are within transmission range of each other, we define the concept of a (communication) path between two nodes as a sequence of nodes and links that connect them. If we limit the length of the path to some number d, then a node i can only communicate with the set of nodes within its d-hop radius. Let us call this set of nodes the d-hop neighborhood of i, denoted by Ni_d, and every node contained in this set a d-hop neighbor of i. For simplicity, we use the term neighborhood to refer to the d-hop neighborhood of a node. For sake of clarity, we start with a simplified one-to-one mapping of clients to nodes. For simplicity, we use the term node to refer to the client it serves and member to any node that belongs to a group. Thus, we let a set of nodes form a group $G \subset V$ where every member $i \in G$ has a current local view v_i and a sequence V_i of installed views. Initially, every member has the initial view v_0 containing the initial snapshot of the membership as the current view and $V_i = \{v_0\}$. In general, not all members can communicate directly, but any two members can communicate if there is a communication path of length at most d between them. Naturally, connected members may have different perceptions of the group membership due their spatial location and bounded neighborhood. Fortunately, for non-critical applications, an approximation to the current group membership suffices as long as connected members share that approximation. That is, a view of the group membership is consistent, if all members contained in the view share the same perception of the group membership.

In our approach, we decouple how to detect nodes located within the neighborhood from how to build the local perception of the membership. This allows us to encapsulate timing assumptions, neighborhood size and node detection accuracy in an oracle, called the neighborhood service. The membership service uses the consistency needs provided by the application to determine its degree of consistency and tailor the local neighborhood service by providing two tuning parameters: a time detection period T and application fidelity x. The time detection period determines how often an update on the neighborhood is expected by the membership service. The application fidelity specifies the required precision and faithfulness of the neighborhood information. Based on these tuning parameters, the neighborhood service adapts the neighborhood size d and heartbeat inter-sending interval t_{hb} and outputs the neighborhood set. The neighborhood set contains a set of detected nodes grouped by hop distance and the number of heartbeats received from each one of them in the last T time steps. The neighborhood service communicates with its membership service through a special send-only channel on which the membership service receives the neighborhood set. The membership service considers the last neighborhood set received as the current one. Based only on the information received by the neighborhood service, the membership service builds its local perception of the membership. Since the local perception of the membership changes independently and arbitrarily at each member, the membership service has to build a shared perception of the group membership among connected members. A conservative solution is to build a transitive closure of connected members and create a view containing all connected members. However, this approach easily becomes expensive in terms of time and communication overhead as the group grows in either size or diameter. Empirical observations showed that some members are better connected than others and those well-connected members tend to have a similar perception of the group membership. Furthermore, well-connected members are resilient to intermittent connectivity and play an important role in the group connectivity. We capture the intuition of well-connectedness by defining the degree of connectivity of a node as a weighted sum of the degree of connectivity of nodes it can communicate directly with.

Definition 1 (Degree of Connectivity) *Given a set of detected nodes and a stochastic matrix of link weights* $\boldsymbol{W}$*. The degree of connectivity of a node is a real number* $r_i > 0$*, such that* $r_i = \Sigma_j w_{ij} r_j$

By default, the weight of a link is $1/deg$, where deg is the degree of the node. However, the weight can be enhanced to include the end-to-end link quality, as long as $\boldsymbol{W}$ remains stochastic (a non-negative matrix, all of whose rows sum to 1).

3. THE NEIGHBORHOOD SERVICE

Given the time detection period T and application fidelity value x, the neighborhood service on a node determines the maximum size d of the neighborhood in terms of hop distances and the heartbeat inter-sending interval t_{hb} such that the following two constraints are satisfied:

$$t_{hb} + \left(d \times hdel\right) \leq T$$

$$\sum_{k=1}^{d} \left|N_k\right| \left(1 - \left(1 - pl\right)^k\right) \leq \sum_{k=1}^{d} \left|N_k\right| \left(1 - x\right)$$

Where $d \geq 1$, N_k is the set of nodes located k hops away from the node, $|N_k| \geq 1$ is the cardinality of N_k, and pl is the estimated message loss probability, which is recalculated every T time steps based on the average number of detected missed heartbeats. Intuitively, the probability of detecting a node within the d-hop neighborhood of the node depends on the message loss probability, the number of hops covered, the number of nodes in each hop distance range and the connectivity of them. For the analysis, we assume that the links between two neighbor nodes satisfies the following message independent property: The message loss probabilities of any two messages sent are independent. Therefore, every message sent has a probability pl of been lost and a probability $(1 - pl)$ of been successfully received by one-hop neighbors. Assuming that a node forwards a heartbeat only once and every heartbeat can only be forwarded $d-1$ times, in the worst case scenario there is only one path between any two non-adjacent nodes within d-hop distance from each other. We then use the message independence property in the worse case scenario to calculate the lower bound in terms of a successful end-to-end heartbeat reception as $(1 - pl)^d$. Thus, $1 - (1 - pl)^d$ bounds the probability of not detecting a node within d-hop distance because his only heartbeat was lost in the process. If we calculate this for each detected node at every hop covered and then divide it by the total number of nodes detected, we have a bound in terms of the number of nodes not detected within the neighborhood, which is the complement of the fidelity compliance. Since the distribution of nodes in the d-hop neighborhood of a node changes stochastically, the set of nodes within the d-hop neighborhood and the number of messages lost due to connectivity changes or network traffic fluctuates over time. Thus, at the end of *T* time steps, the cardinality of every hop covered ($|N_k|$) is calculated based on the number of received heartbeats and a new estimated message loss probability (pl_{new}) is calculated based on the average number of lost heartbeats detected within the d-hop neighborhood as follows:

$$pl_{new} = \frac{\sum_{k=1}^{d} \sum_{\forall j \in N_k} missed[j] \Big/ received[j]}{\left| N_k \right|}$$

Where *missed[j]* is the cumulative missed heartbeats from *j* and *received[j]* is the total number of received heartbeats from *j*. Then, the message loss probability (*pl*) is adjusted if the difference between pl_{new} and the current *pl* is greater than *pl/2*. Finally, the constraints are recomputed and the values of *d* and t_{hb} are adjusted, if needed.

4. THE MEMBERSHIP SERVICE

The membership service is responsible for extending the sequences of views based on changes in the local perception of the membership, and to do so consistently at different members. We divided the membership service in two phases. In the first phase, members build their local perception of the membership *M* independently, based on the information received by their local neighborhood service. Then, each member uses an application input parameter $k > 0$ to divide *M* in two disjoint sets: a local nearby-set containing *k* members, with the highest degree of connectivity, and a local remote-set containing the remaining members. In the second phase, members extend their sequences of views by reaching an agreement on a common nearby-set and thus committing to a new view with a consistent nearby-set that reflects a group membership approximation.

4.1. Building a Local Perception

The membership service on a node, uses the set of detected nodes grouped by hop distance and heartbeats to generate a neighborhood connectivity graph, where detected nodes are the vertices of the graph and links are created using the node-disjoint path information contained in the received heartbeats. This information is stored in the form of an adjacency matrix $\boldsymbol{A} = (a_{ij})$, in which $a_{ij} = 1$, if members *i, j* are connected and $a_{ij} = 0$ otherwise. Once the adjacency matrix has been created and the degrees of each member in $\boldsymbol{A}$ calculated, the associated link weights matrix is generated as follows: $\boldsymbol{W} = (w_{ij})$, in which $w_{ij} = 1/deg(i)$, if $a_{ij} = 1$ and $w_{ij} = 0$ otherwise. Then, the local perception of the membership is generated by identifying the members contained in the set of detected nodes. Finally, the degree of connectivity of every node in the neighborhood connectivity graph is calculated using the matrix-vector form representation of Definition 1, where $\boldsymbol{r}$ is the vector containing the degree of connectivity of members in $\boldsymbol{A}$. This matrix-vector form representation leads to a set of linear equations for unknown $\boldsymbol{r}$ with known $\boldsymbol{W}$, which is the standard eigenvalue problem. Thus, Definition 1 can be rewritten as $\boldsymbol{\lambda r} = \boldsymbol{W^T r}$, where the degree of connectivity vector $\boldsymbol{r}$ is the eigenvector corresponding to $\lambda = 1$, the largest eigenvalue of $\boldsymbol{W^T}$, the transpose of $\boldsymbol{W}$. Given that $\boldsymbol{W}$ is a stochastic matrix, $\boldsymbol{W}$ and $\boldsymbol{W^T}$ have the same characteristic polynomial and thus the same eigenvalues, which are the roots of the polynomial. Furthermore, $\boldsymbol{W}$ and $\boldsymbol{W^T}$ are also non-negative square matrices with real-valued entries, thus the Perron-Frobenius theorem guarantees that the largest eigenvalue of $\boldsymbol{W^T}$ is always one and all eigenvectors can be computed efficiently. Therefore, the membership service on a node chooses its local nearby-set as follows:

1. Calculate the eigenvector $\boldsymbol{v_{pf}}$ associated to the largest eigenvalue, λ_{pf}, of $\boldsymbol{W^T}$.
2. Sort the members by decreasing order of the absolute values of their coordinates in $\boldsymbol{v_{pf}}$, where equalities are broken arbitrarily.
3. Select the first *k* members of the sorted list and call this set the local nearby-set of the member.
4. Select the remaining members in *M* which are not contained in the nearby-set and call this set the remote-set of the member.

4.2. Agreeing on a New View

Since members choose their local nearby-set and remote-set independently and arbitrarily, members have partial random views and a common shared view that reflects an approximation of the current group membership has to be agreed upon before any communication among members can take place. Our approach relaxes the consistency required to agree on a snapshot of a group membership by:

1. Defining an enhanced view where detected members are divided in two disjoint sets, a nearby-set and a remote-set.
2. Allowing only members contained in the (individual) local nearby-sets to agree on a common set of members as the nearby-set of a new enhanced view.
3. Requiring consistency of the enhanced view's nearby-set only. That is, a member is allowed to choose its own remote-set based on its local perception of the membership.

We first define an enhanced view as follows:

Definition 2 (Enhanced View) *Given a local perception of the membership M, an enhanced view is a triplet of the form* $ev = \langle id, N, R \rangle$*, where id is a view identifier,* $N \subseteq M$ *is a non-empty nearby-set and* $R \subset M$ *is the remote-set, such that* $M - N = R$.

Then, we define a membership change as a change in the nearby-set of the current view. When the local perception of the membership changes, the membership service on a node chooses a new local nearby-set as described in the previous section. Then, it compares the new local nearby-set to its current local nearby-set. If they differ, a membership change has occurred and the membership service executes the following steps:

1. Updates its local nearby-set.
2. If the member is contained in its local nearby-set and it has the lowest node identifier in its local nearby-set, then the member sends a propose-view message to all members in its nearby-set proposing a new view with its updated local nearby-set.
3. Whenever the member receives a propose-view message, it compares its local nearby-set with the one contained in the proposed message and sends back an accept or retry message:
    a. An accept message is sent if the member agrees on the proposed view identifier and its local nearby-set is contained in the proposed nearby-set. Upon sending the accept message, the member reserves the corresponding position in its sequence of views, so no other proposal is accepted with that view identifier.
    b. A retry message containing the union of the proposed nearby-set and the member's nearby-set is sent, if the member's nearby-set is not contained in the proposed nearby-set or the member does not agree with the proposed view identifier.
4. Whenever a retry message is received, the member updates its proposed nearby-set with the nearby-set contained in the retry message and resends it with a possibly higher view identifier.
5. When the proposing member has collected accept messages from all the members in its proposed nearby-set, it sends a commit message.
6. Upon receiving a commit message, members send the proposed view to every member in their remote-set as an install-view message. Then they update the view's remote-set with their local remote-set, install it as the current view and extend their sequences of views accordingly.

Every (remote) member that receives an install-view message, updates the new view's remote-set with its local remote-set, installs it as the current view and extends their sequences of views accordingly. As a result, after a membership change, the membership service outputs a new view, such that every member contained in the new enhanced view extends its sequence of views in a consistent manner.

5. SIMULATIONRESULTS

We used the ns2 network simulator to evaluate our approach under three network densities and two commonly used mobility models. In all six scenarios, nodes have a fixed transmission range of 150m and 1Mb of available bandwidth. In order to obtain meaningful results, we repeated each scenario 50 times, aggregated and averaged the obtained results. The network density determines the number of nodes in the simulation and the network topology based on the distribution of the nodes in the simulation space. We use the following network densities: (1) *Sparse:* The network consists of 29 nodes, distributed in the central region of the simulation space. Initially all the nodes form a connected component, (2) *High-Low:* The network consist of 120 nodes, distributed in two high density regions containing 50 nodes each and two low density regions containing 10 nodes each. All regions form a connected component at the beginning of the simulation, and (3) *Dense:* The network consists of 250 nodes, distributed randomly in the simulation space. The following mobility models were used to generate initial node positions and schedule node movements for the simulation: (i) *Gauss-Markov:* scenarios were generated using a simulation area of 1024x1024 square meters. The node velocities were chosen from the interval 5-10 m/s and a speed standard deviation of 0.5. The nodes changed speed and direction every 2.5s and the standard deviation for the angle was $\pi/4$. (ii) *Random-Waypoint:* scenarios were generated using a simulation area of 1024x1024 square meters enhanced with 2 and 8 attraction points. Each point had an attraction intensity taken from the interval 5-10, with a standard deviation of 20 and a uniform probability of pausing at the attraction point of 0.75. The node velocities were chosen from the interval 5-10 m/s and a speed standard deviation of 0.5. Finally, we implemented the following two service strategies that characterize previous approaches to provide group communication in mobile environments and compare them to our approach: (I) *Global-DSDV strategy:* A traditional membership service running on top of the omega failure detector [9] and using the DSDV routing mechanism to allow members to track every other member in the group. (II) *Cluster Strategy:* A traditional membership service running on top of the cluster-based failure detector [10], where cluster-heads keep track of members within their transmission range and members communicate by sending a message to their corresponding cluster-head, and the cluster-head re-broadcast to its neighbors and other cluster-heads any received message. This failure detector was designed to work with an additional mechanism to create and maintain a connected dominating set in the network.

5.1. Neighborhood Service

For the purpose of comparison, we normalize the heartbeat interval of both failure detectors with our neighborhood service in all scenarios. Following [10], we initialize the message loss probability to 0.01 and one-hop message delay to 0.02. We let the initial parameters for the fidelity neighborhood service be $T = 0.5$ and $x = 0.8$. Since the omega failure detector assumes that every member has a direct communication link with every other member, it uses the underlying routing mechanism (DSDV) to send and receive heartbeat messages. This adds unpredictable queuing and routing delays for every message send, and increases the number of mistakes made. Thus, the global strategy does not scale. The performance of the cluster-based failure detector is dependent on the mobility model and the unstated assumption that the mobility of the cluster-heads is low. In general, the neighborhood service and the cluster strategy have a similar overhead in high-low and dense topologies, regardless of the selected mobility model. The cluster strategy has a slightly less message overhead in sparse topologies due to its hierarchical approach. This validates the intuition that for structured networks with a moderate node density, the cluster strategy has the best cost-performance. In terms of neighborhood coverage, the average number of detected nodes in the global strategy is always low due to the lack of locality and unpredictable message delays. Furthermore, unpredictable delays of far away nodes often trigger intermittent detection that causes the neighborhood fidelity to decrease considerably due to false suspicions. The neighborhood-driven (NS) strategy outperformed the cluster strategy in the Gauss-Markov scenarios due to the unstructured topology and random mobility. However, in the Random-Waypoint scenarios, the differences are marginal due to the attraction points and extended pause times. Figure 1 shows the neighborhood coverage and corresponding neighborhood fidelity of two selected scenarios. The observed sharp fluctuations of the cluster strategy is due to the cluster-head election process. In the cluster strategy, cluster-heads may change due to mobility or power balancing. During the cluster-head election process and the new cluster-head information gathering, members use stalled information, which decreases fidelity for a short period of time.

5.2. Group Communication Service

In our simulations, we assume that only one group is active in the network and that the initial group size is set to the number of nodes in the sparse topology. Initially, members are randomly distributed in the network topology. We model join and leave events using a negative exponential distribution with parameter $\lambda = 10$. Similarly, we use a Poisson distribution with parameter $\eta = 2$ to model message requests from the application and we model node failures using a Weilbull failure rate function with parameters $\alpha = 1$ and $\beta = 2$.

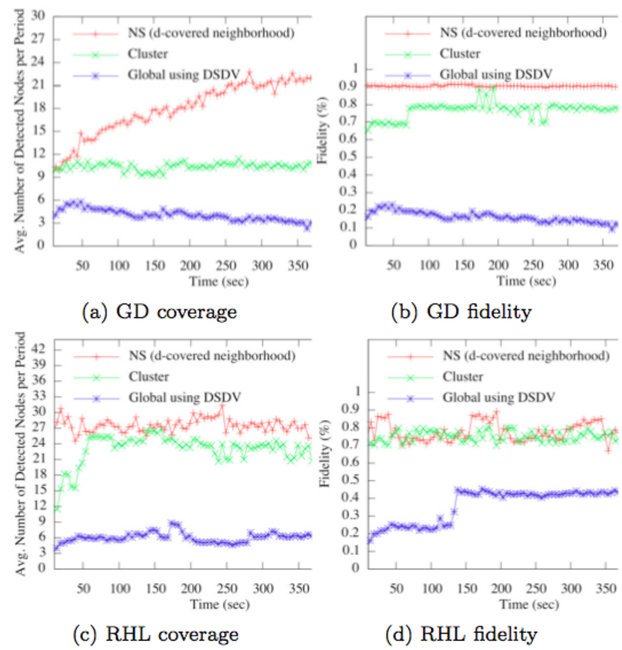
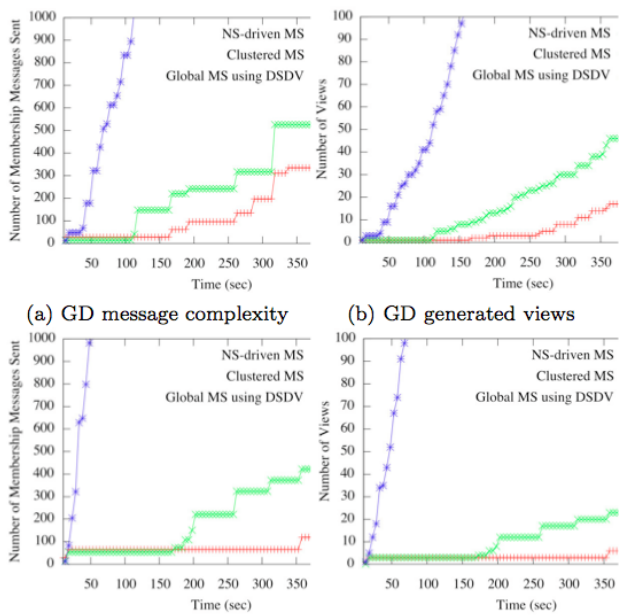


Figure 1. Neighborhood coverage and fidelity of a Gauss-Markov scenario (GD), and a Random-Waypoint High-Low scenario (RHL).

When a partition or merge occurs, members have to exchange several messages in a short period of time in order to agree on a new view. Since the cluster and global strategies use traditional membership services, the number of messages exchanged is $O(n^2)$, where n is the number of reachable members within the partition. This produces the observed staircase effect in the message complexity graph in Figure 2. Since, the NS-driven MS requires only members in the nearby-set to agree on a new view, the staircase effect only occurs if the partition disconnected members in the nearby-set.

In order to capture the impact of node failures and long-term disconnections in a uniform manner, we model disconnections as temporal failures without state or information loss. We assume that nodes fail or disconnect independently. Figure 2 shows the message complexity and the corresponding generated views of a scenario where nodes can fail/disconnect and (as a result of it) network partitions and merges. We observed that disconnection of multiple nodes in a short period of time is not a rare event in a dense topology scenario and that the cluster strategy is highly sensitive to cluster-head failures.



(a) GD message complexity (b) GD generated views

(c) RHL message complexity (d) RHL generated views

Figure 2. Message complexity and number of generated views of a Gauss-Markov scenario (GD), and a Random Waypoint High-Low scenario (RHL).

6. CONCLUDING REMARKS

In this paper, we presented a group membership service based on partial member connectivity that allows members to agree on a shared approximation of the group membership. In particular, our membership protocol uses the degree of connectivity to divide members in two disjoint sets: the nearby-set and the remote-set. Members agree on a common nearby-set and view identifier, but they can disagree on the remote-set. Furthermore, only members contained in the nearby-set participate in the agreement of a new view and membership changes are restricted to changes in the agreed nearby-set. We also developed interaction parameters that allow us to tailor dynamically the behavior of both services to specific application requirements. First, the application consistency needs are translated into three values: the input parameter k, a time detection period T and application fidelity x. Then, the membership service takes the value of k as the initial size of the local nearby-set and uses T and x to tailor the behavior of the neighborhood service in terms of how often the neighborhood information should be updated and how faithful the information should be, respectively. Our preliminary studies show that our neighborhood service outperforms previous approaches in terms of neighborhood coverage and captures the neighborhood connectivity with a high degree of accuracy. Similarly, our hierarchical membership service generates less views than traditional ones, since membership changes are triggered only if the agreed subset of members (nearby-set) changes. Furthermore, when a membership change is triggered, only the nearby-set participates in the agreement of a new view, decreasing also the message complexity of the membership protocol. Our on-going work include an in-depth analysis of probabilistic message delivery semantics in our model, including the impact of the input parameter k in the performance of the membership service and its consequence on data delivery accuracy. Another direction in which we are extending our work is the integration of group communication services in delay tolerant networks.

REFERENCES

[1] S. S. Basagni, M. Conti and I. Stojmenovic, MOBILE AD HOC NETWORKING, 1st ed., Wiley-IEEE Press, 2004.

[2] A. Balasubramanian, B.N. Levine and A. Venkataramani, "DTN Routing as a Resource Allocation Problem," ACM SIGCOMM, 2007.

[3] R. Prakash and R. Baldoni, "Architecture for Group Communication in Mobile Systems," Symposium on Reliable Distributed Systems, 1998.

[4] A. G. Anastasi and F.L. Luccio, "Fault-Tolerant Support for Reliable Multicast in Mobile Wireless Systems: Design and Evaluation," *Wireless Networks*, Vol. 10, Num 3, pp. 259-269, 2004.

[5] Q. Huang, C. Julien, and G.-C. Roman, "Relying on Safe Distance to Achieve Strong Partitionable Group Membership in Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, Vol. 3, Num 2, pp. 192-205, 2004.

[6] Z. Haas and B. Liang, "Ad Hoc Mobility Management with Randomized Database Groups," IEEE International Conference on Communications, 1999.

[7] S. H. Pucha and Y.C. Hu, "Ekta: An Efficient DHT Substrate for Distributed Applications in Mobile Ad Hoc Networks," IEEE Workshop on Mobile Computing Systems and Applications, 2004.

[8] R. Melamed and I. Keidar, "Araneola: A Scalable Reliable Multicast System for Dynamic Environments," IEEE International Symposium on Network Computing and Applications, 2004.

[9] H. M.K. Aguilera, C. Delporte-Gallet and S. Toueg, "On Implementing Omega with Weak Reliability and Synchrony Assumptions," ACM Symposium of Principles of Distributed Computing, 2003.

[10] K. A.T. Tai and W.H. Sanders, "Cluster-Based Detection Service for Large-Scale Ad-hoc Wireless Network Applications," International Conference on Dependable Systems and Networks, 2004.