A Message Stability Tracking Protocol for Partitionable "ad-hoc" Mobile Networks

Roberto Baldoni, Carlo Marchetti
Dipartimento di Informatica e Sistemistica
Universita' di Roma "La Sapienza"
Via Salaria 113, I-00198 Roma, Italy
www.dis.uniroma1.it/~baldoni

Augusto Ciuffoletti
Dipartimento di Informatica
Universitá di Pisa
Corso Italia 40, 56125 Pisa, Italy
augusto@di.unipi.it

In an ad-hoc network hosts are mobile devices and there is no base station that can coordinate the activity of subsets of nodes. Therefore, in order to route and deliver messages, all the nodes have to cooperate and collectively make decisions. The mobility of the nodes adds extra complexity to the routing protocol, since a node's neighborhood changes with time, and is not predictable. As nodes can communicate only if they are within the wireless communication range, their mobility dynamically changes the connectivity, possibly inducing network partitions.

Literature on routing in ad-hoc networks (see [4] for a survey) generally assumes that a *phisical* (wireless) communication path exists between each pair of mobile hosts, even though these paths may change over time. i.e., if an host h_i sends a message m to h_j at time t, there will exists a phisical communication path between h_i and h_j at time t. This path consists of a sequence of phisical communication links, active at time t, connecting neighbour hosts [1].

This makes it possible to route a message from any pair of mobile hosts without buffering the message anywhere within the network, that is clearly not prone to the partitioning phenomenon. However, there are cases where this assumption is too strong, since it excludes the partitioning of the network.

In this investigation we weaken the former system model by assuming that a *logical* communication path between each pair of mobile hosts is formed infinitely often in an infinite run of the system. A logical communication path connecting two hosts h_i and h_j consists of a finite sequence of phisical communication paths $c_1, \ldots c_k$ (with $k \geq 1$) established at different times $t_1, \ldots t_k$ with $(t_i \leq t_{i+1})$.

As a consequence a phisical communication path is a particular logical communication path with k equal to 1. We call such a model *logically connected ad-hoc network.*

A logically connected ad-hoc network is composed by a set of hosts $\{h_1, h_2, \ldots h_n\}$. Due to the mobility of hosts, the system can be partitioned. A *partition* is a non-empty subset of hosts $H \subseteq \{h_1, h_2, \ldots, h_n\}$ such that for each pair of hosts (h_i, h_j) in H, there exists a physical communication path.

As a consequence each host h_i is associated to a partition H_j that contains at least h_i.

In a logically connected ad-hoc network, the routing protocol alone cannot always route a message between any two hosts; if they belong to distinct partitions the message needs to be *buffered* in the sender's partition, in order to be eventually delivered to the final destination through a logical communication path.

A *message stability tracking* protocol [2] is responsible for managing the host buffers. This activity consists in deciding when to deposit a message in the host buffer, and when to discard it. In figure 1 the protocol stack at each host is depicted.

---

[1] Phisical communication links between hosts whose distance is less than D meters are provided by MAC layers such as IEEE 802.11 ([1]).

We are currently designing a *message stability tracking* protocol that guarantees FIFO and reliable communication between mobile hosts in a logically connected ad-hoc network, by exploiting message buffering.

As an example, let host $h_i \in H_k$ send at time t a message m to host $h_j \in H_l$ which does not belong to h_i's partition. Each host in H_k will buffer m. When H_k and H_l merge creating a new partition H due to the mobility, a merging protocol will be executed whose aims is to update the buffer at each hosts so that it will contain all not-yet-delivered messages coming from buffers of the merging partitions. This result is reached by starting appropriate *sessions* between the first two hosts belonging to the merging partitions that detect the merging event.

In order to reduce the number of opened sessions, the protocol exploits the underlying network layer, which runs a routing protocol suitable for ad-hoc networks. In particular, the network layer allows $h_i \in H_i$ to communicate with hosts belonging to H_i (intra-partition communication) and provides information about host's partition to the protocol layer (host partition information).

Intra-Partition Communication The network layer offers interfaces to send messages to all the hosts belonging to a partition H by invoking the `bcast(m) within` H `primitive`. The network layer may also invoke an up-call that causes the `receipt(m)` guard in the host's message stability layer to become valid, indicating that the message m has been delivered to the local host. These features are implemented extending an existing routing protocol for ad-hoc networks (e.g. [6, 3, 5] etc.).

Host Partition Information The network layer informs the protocol layer about changes in the partition membership.

We assume that, when h_i's partition changes, the network layer provides the message stability layer of h_i with the updated H_i membership. This can be implemented in the network layer by invoking an up-call. that causes the `partition_modification`(H_i) guard in h_i's message stability layer to become valid, where H_i lists the updated membership of h_i's partition.

Regarding the information available about the membership, we distinguish two cases, generating two distinct system models, that differ for the quality of the information supplied by the network layer, which in turn depends on the complexity of the underlying routing protocol. We named the two models the *complete information model* and *partial information model*:

- **Partial Information Model:** the network layer can supply only information about the number of hops of the communication paths connecting the host with another host in the same partition. Therefore, the h_k's message stability protocol layer can invoke a network layer function $\#\texttt{hops}_\texttt{k}$(i), $i \in H_k$, that returns the number of hops of the path connecting h_k with h_i.
- **Complete Information Model:** the network layer allows each node to *exactly* track *all* the topology changes inside its partition: the generic host h_k can invoke the $\#\texttt{hops}_\texttt{k}$(i,j) network layer function, that returns the number of hops of the communication path existing between h_i and h_j, with $h_i, h_j \in H_k$.

Figure 1 illustrates the relationship between host h_i's MAC, network and message stability protocol layers.

The solution we propose is open to a number of refinements, and one of the main questions to answer is the tradeoff between the *partial information model* and the *complete information model.* The former has an inexpensive and known implementation in the network layer, but the incomplete information it provides may bring to the activation of redundant sessions. The latter has an expensive implementation, since each node must collect and keep updated a more complete information about the connectivity of the network, but avoids the generation of redundant sessions. We are currently simulating both environments to quantify the tradeoff.

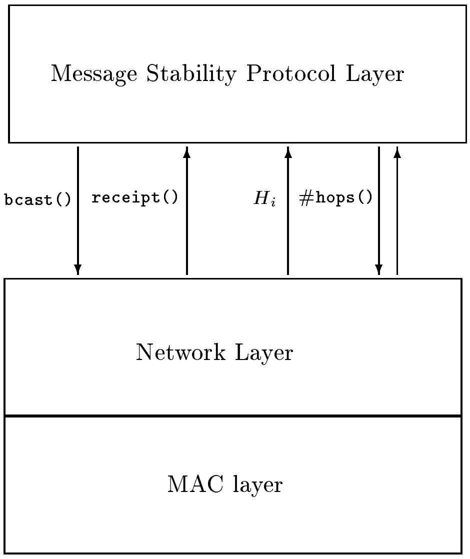

Figure 1: Interactions between the Network and the Message Stability Protocol Layers

References

[1] IEEE Computer Society LAN MAN Standard Committee. Wireless lan medium access control (mac) and physical layer (phy) specification. Technical Report IEEE Std 802.11-1997, The Institute of Electrical and Electronics Engineering, New York, 1997.

[2] K. Guo, R. van Renesse, W. Vogels, and K. Birman. Hierarchical message stability tracking protocols. Technical report, Dept Computer Science, Cornell University, 1997.

[3] D.B. Johnson and D.A. Maltz. Dynamic source routing in ad-hoc wireless networks. In T. Iemielinski and H.Korth, editors, *Mobile Computing*, pages 153–181. Kluwer Academic Publishers, 1996.

[4] E.R. Royer nad C-K Toh. A review of current routing protocols for ah-hoc mobile wireless networks. *IEEE Personal Communication Magazine*, April 1999.

[5] C.E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance vector routing (dsdv) for mobile computer. *Computer Communication Review*, pages 234–244, October 1994.

[6] M.R. Pearlman Z.J. Haas. The zone routing protocol (zrp) for ad-hoc networks. Technical report, Internet Draft RFC, November 1997.