Random Walk Gossip: A Manycast Algorithm for Disaster Area Networks

Mikael Asplund, Simin Nadjm-Tehrani
Department of Computer and Information Science, Linköping University
SE-581 83 Linköping, Sweden
{mikas,simin}@ida.liu.se

When the communication infrastructure is needed the most - in the event of a disaster - it is the most likely that it is not available. We suggest that rescue personnel working in such conditions can be supported by networking protocols which are tolerant to disconnectivity and unstructured topologies. Moreover, since communication devices will probably be battery driven and power is not easily available, protocols need to very restrictive in communication to save power. In particular, we are interested in reliable multicast operations in which a sender wants to send a message that can be relied upon to reach at least a portion of a certain group of receivers (i.e., manycast).

Epidemic algorithms in mobile networks can be broadly categorised as using localised gossipping [2] or anti-entropy [4]. Both mechanisms have drawbacks; while the former approach suffers from a complicated balance between wasting resources and the risk of messages not being propagated, the latter provides full coverage but generally results in slow propagation as well as a high bandwidth usage. Recently, Khelil et al. [3] proposed to use a combination of these approaches called hyper-gossipping to achieve best-effort broadcasts in partitioned networks.

We believe that cooperation in post-disaster areas requires a new kind of protocol which is efficient (short delay, low bandwidth), capable of dealing with disrupted communication, reliable, and which does not require knowledge about which nodes are currently operating in the network.

In this abstract we describe the basics of a protocol called Random Walk Gossip (RWG), described earlier and experimentally evaluated [1]. This protocol meets the need for energy-efficient reliable communication in an intermittently connected environment. The protocol relies heavily on hashing of node addresses instead of keeping track of all the nodes in the system. This way we take the middle way between best-effort algorithms requiring no knowledge at all and fully reliable protocols requiring full knowledge. RWG terminates when at least k nodes will be reached by the message.

The protocol has two modes: gossipping and waiting. During the gossipping phase, the message spread in the network. If a holder of a message (custodian) detects that the network is partitioned, it puts the message on hold. This will cause nodes to be silent when no new nodes can be reached, and thus reducing energy consumption. Eventually, the node will discover that uninformed nodes are nearby and resume propagation of the message.

Algorithm 1 outlines the basic behaviour during the gossipping phase of the algorithm. Each message has a bit vector (*informed*) whose role it is to keep track of the nodes that have received the message. The index of each node is decided by taking the hash of the node ID. For example, the vector $[0, 1, 0, 0, 1, 0]$ indicates that the nodes with hash 2 and 4 have received the message.

Algorithm 1 Random walk gossip

```
When a message m is heard by node i from neighbour j:
  wake up all messages not seen by j
  set m.informed[hash(i)]=1
  if forwarder:
    Send m to a random neighbour (if possible)

When k nodes have been reached (easily seen in m.informed):
  stop forwarding and propagate beSilent(m)
```

A message is made inactive if there is no neighbour that can forward it, thereby indicating a possible network partition. The decision to make a message inactive is specific to that message alone, since other messages might already have seen the parts of the network which are deemed to be currently unreachable. As soon as a node discovers a neighbouring node that has not seen the inactive message, it is reactivated, and the random walk starts anew.

The algorithm has been extensively evaluated in a simulation environment built on top of ns-3. Current work includes implementing the algorithm on a range of hand held devices (Linux-based tablets and laptops, as well as Android telephones).

Acknowledgements

This work was supported by the Swedish Civil Contingencies Agency (MSB) and the Swedish Research Council (VR).

References

[1] M. Asplund and S. Nadjm-Tehrani. A partition-tolerant manycast algorithm for disaster area networks. In *28th International Symposium on Reliable Distributed Systems*. IEEE, Sept. 2009.

[2] Z. Haas, J. Halpern, and L. Li. Gossip-based ad hoc routing. *IEEE/ACM Trans. Networking*, 14(3):479–491, June 2006.

[3] A. Khelil, P. J. Marrón, C. Becker, and K. Rothermelns. Hypergossiping: A generalized broadcast strategy for mobile ad hoc networks. *Ad Hoc Netw.*, 5(5):531–546, 2007.

[4] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical report, Duke University, 2000.