BlueNode

Ruben Stap, Duncan Kampert, Joey Lai, Julius Wagt

Netcentric Computing 2019, Universiteit van Amsterdam

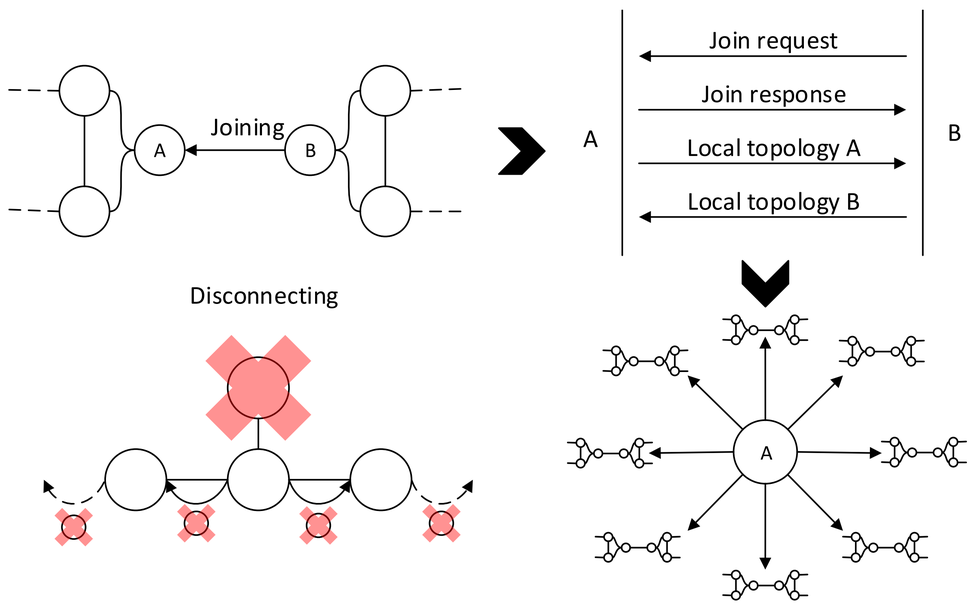
Introduction

Ad hoc networks are self organized networks without any centralized control. In these networks nodes aren't necessarily directly connected with all other nodes. Nodes which are not directly connected can then communicate with one another with the help of multi-hop routing. This means that data might pass multiple nodes before reaching the destination. The goal of our project is to create an application with which an ad hoc network can be formed. This network should allow for indirect communication. We implemented multiple routing algorithms with a chat application built on top.
We compared two network maintenance protocols and showed which of these are better for certain network topologies.

Heartbeat

We use a Heartbeat network to periodically check for inactive peers. When a peer is inactive it will be removed from the peer list. Removed peers can sever the network but by removing them manually we can restore the network after.

On change



Network performance

The first experiment checks the performance of our network compared to the theoretical maximum Bluetooth offers. This experiment was split into two sub-experiments, namely latency, and throughput.
For latency measurement we used a ping round trip time (RTT) measurement over multiple hops. We used the direct routing algorithm and measured the ping RTT over one, two, and three hops. Our amount of devices limited the measurement of more hops.

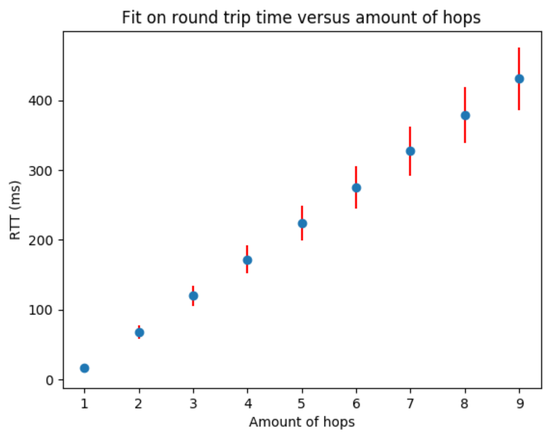
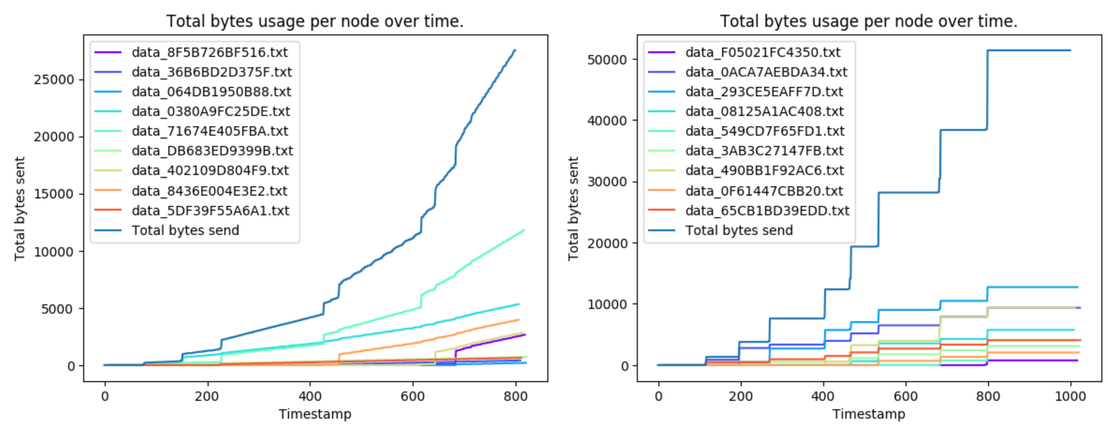


Figure: RTT data fit for more than three hops where 4-9 were linearly extrapolated.

We measured both the throughput, and the latency of our network per hop. For the throughput we sent a large file (300 MB) of data through the network using direct routing. We observed a maximum throughput of approximately 700 KB/s, averaged over multiple seconds of communication.

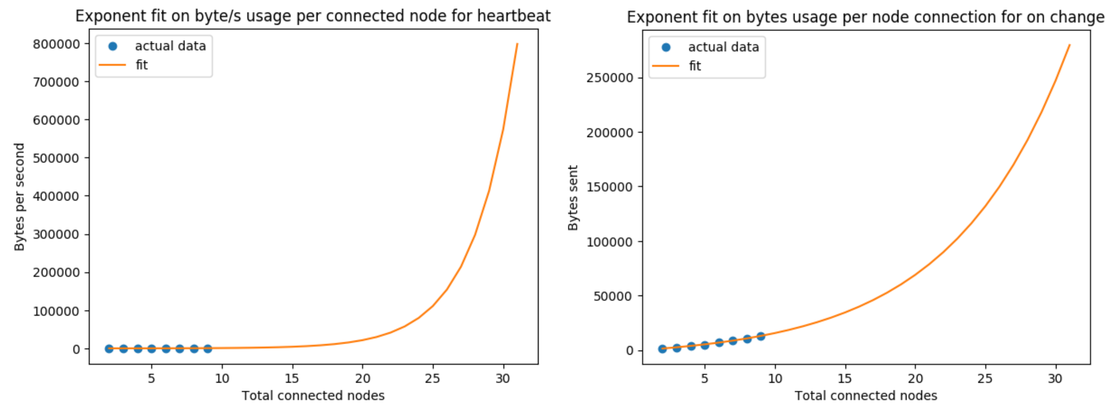
On-change vs Heartbeat

The second experiment we did was to check the performance of our implemented algorithms. We experimented to compare the performance between two modes. To ensure the results have as little differences between them as possible, we did all experiments on the same network topology.



(a) Heartbeat algorithm (b) On-change algorithm

After a quick glance we can see that the total bytes usage per node with the on-change algorithm is much higher than the Heartbeat algorithm in the first 800 timestamps.



(c) Heartbeat algorithm (d) On-change algorithm

Figure: Fitted data usage per network maintenance algorithm

We can see that the exponential growth of bytes of the Heartbeat algorithm is higher than the exponential growth of the total bytes send with on-change. This is because the Heartbeat algorithm floods the network with a exist message from every node periodically, while the on-change algorithm only floods the network with a updated topology from the joined node.

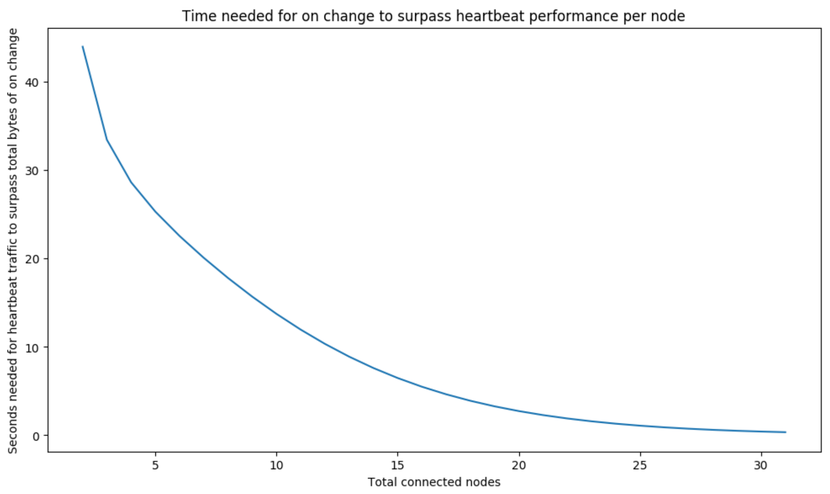


Figure: Performance comparison between Heartbeat and on change

Conclusion

Our final work shows it is possible to create a mobile and dynamic ad hoc network. Our implementation may not be perfect and leaves room for further improvement, but it proves that with only short range wireless technology such as Bluetooth we can replicate most of the functionality that we find in other communication protocols. Furthermore, we have concluded what the optimal network maintenance algorithms are for different networks. That is to say, Heartbeat for short small networks and on-change for long large networks.