Report of the visit to Lisbon University*

Carl Ellis

June 30, 2008

1 Introduction

This report documents my visit to the University of Lisbon, between the 23rd and the 28th of June 2008.

The visit had the goal of determining if the Power Aware Message Propagation Protocol (PAMPA), which is used in Mobile Ad hoc Networks (MANETs), could determine the "direction" of message propagation within a network, to compute whether a nodes retransmission of a message would be beneficial to further message propagation.

During the visit I had meetings with several researchers of the University of Lisbon, namely: Professor Hugo Miranda and his Masters students: João Matos and Pedro Fonseca. These meetings included introductions into everyone's fields of research and resulted in some beneficial ideas concerning the evolution of the protocol.

1.1 Overview Of Report

A number of varying ideas were uncovered, ranging from the protocol tracking the history of message route, to using Received Signal Strength Indicator (RSSI) values in pre-determined thresholds for the protocol to determine its course of action. The following sections of this report present the principal conclusions for idea addressed during the visit and a small program which was created to easily determine the results of a protocol change through visual media. The report terminates with some comments and conclusions in section 9.

2 Background and Previous Work

2.1 Broadcast Algorithms in MANETs

Mobile Ad Hoc Networks are wireless mesh networks involving many mobile nodes and no centralised infrastructure.

Due to the lack of physical connections, messages are passed between mobile nodes through broadcasts. The fluid nature of a Mobile Ad Hoc Network also impresses a very temporary nature on routes between mobile nodes, so a reliable

*This visit was supported by the Middleware for Network Eccentric and Mobile Applications(MiNEMA) programme of the European Science Foundation.

and intelligent broadcast protocol is essential to allow communication between a single node and the rest of the mobile network.

A simple, but resource wasteful solution is to use flooding, where if a node receives a message it has not yet received, it logs the message and broadcasts it. This allows maximum broadcast coverage, but also the maximum use of battery power and bandwidth.

2.2 Background of PAMPA

PAMPA allows maximum coverage of a broadcast algorithm, but with a much lower use of total battery power within the network as a whole. Battery power is saved by preventing nodes broadcasting where the message coverage gained would be unnecessary or minimal compared to previous broadcasts.

PAMPA does this by taking into account the RSSI of a received message. From this metric a relationship to distance can be retrieved, so for a larger RSSI the closer a node must be to the source of the message it has received.

Using a similar algorithm paradigm as the GOSSIP1(p) [2], where a broadcast of a message is held off for a randomly defined interval, wherein the node listens for another broadcast of the same message before deciding whether to broadcast itself.

PAMPA defines its delay by using the RSSI of a message. This allows for a delay time proportional to a node's distance to a transmitter.

PAMPA also uses a message threshold similar to the HCOUNT(p) [2] algorithm, where the ratio of message coverage over number of messages can be improved by tweaking the message threshold.

2.3 The issue with the current implementation of PAMPA

The behavior of nodes within a Mobile ad-hoc network using the Pampa protocol was not reflecting what the theoretical results predicted. An order of magnitude more messages were being observed in the NS-2 [4] simulations than was needed or should have been transmitted.

There were certain scenarios where messages would be received by two different nodes which were isolated by each other, and both would transmit. Message propagation would split into two independent paths and then both be received by another node within its given waiting time.

This would not allow propagation to continue, as it would result in the key node receiving more messages than its threshold value, failing to transmit and halting the message propagating to the rest of the network, or at least that section (See Figure 1).

3 Proposed Ideas for the evolution of PAMPA

Work at the University of Lisbon involved brainstorming meetings with Hugo Miranda and his students, discussing an evolution of the protocol which would allow a node to determine the directionality of a message as it propagates through the network, and thus allow the scenario highlighted above to be resolved.

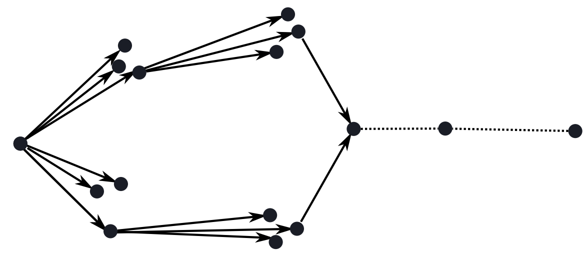

Figure 1: Split Propogation Scenario

This would mean that if a node were at the boundary of the message's current transmission coverage, it would be more likely to transmit.

Among the requirements for the solution, is the need to avoid exceptions strictly aimed to solve a particular case as they *a*) would reduce the general applicability of the algorithm; and *b*) unlikely cover all possible occurrences, due to the limited amount of information available to each node, whcih prevents them from recognizing the moment at which the problem will emerge. Ideally, the solution should not present a significant impact on the fine performance exhibited by the framework in other scenarios. In particular, trivial solutions like flooding should be avoided.

3.1 Common Parent

One method of determining if messages came from the same direction was to assume that the different messages had a common parent within its hop history. This then lead on to the problem of how far back do you look, and is the look-back generic enough or will it have to be tailored for each use. This was not an option which we wanted to take.

Another option was considered, which involved having to add some extra fields into the Pampa header:

- Last Hop
- Parent Node (Two Hops Back)

Using these extra fields we can determine if a message was retransmitted due to a common message from an orginating node two hops back. For example, these fields could be used as so:

1. Nodes A, B and C are in a small network.
2. Node A transmits and is received by B and C.
3. Node B transmits and is received by C and A.
4. Node C can determine that this was a retransmission as the Parent Node of Node B's message is the same as the Last Hop of Node A's message.

So using this technique we can only count messages that are received when they originate from the same parent.

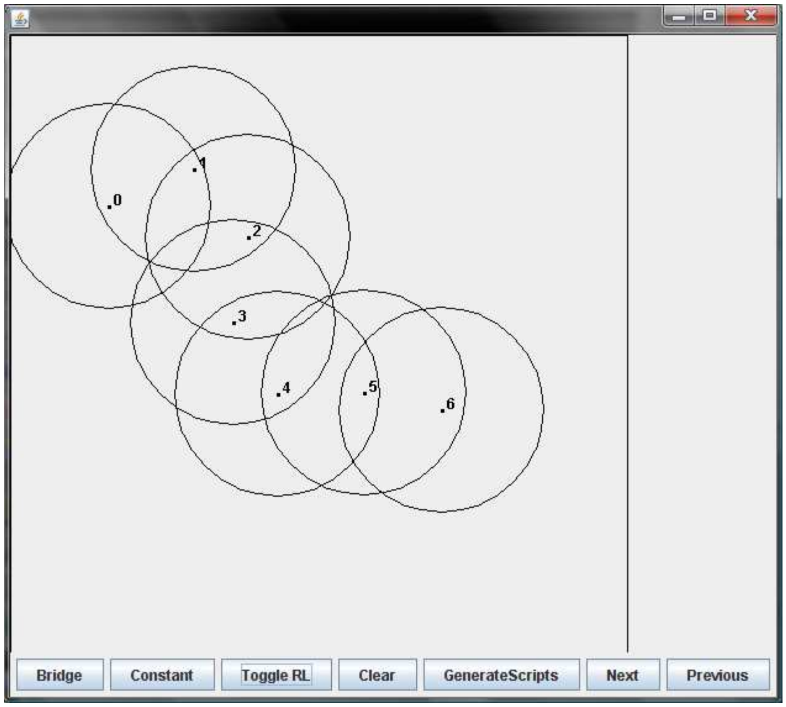

Figure 2: Topology Editor

3.2 Transmission Strength

Another proposed solution was to extend the idea of using a threshold on receiving signal strength, but instead of a fixed value, make it more dynamic by only accepting messages that have a lower receiving signal strength. This means that the only messages nodes accept are from nodes which are further away.

This is good in some node topologies, for example in highly dense node groups, but PAMPA generally works best in dense node populations. It is the sparsely populated node groups which the protocol needs to be able to determine "direction".

4 Pampa Topology Generator

During my time at the University of Lisbon, I found that a quick and easy way to generate topologies would greatly increase my ability to test PAMPA in different scenarios.

So I developed a simple Java program which would allow me to click on an interactive canvas and position nodes at a specified (x, y) coordinate (See figure 2).

Then I made some simple additions, such as showing the maximum transmission range for all the nodes. This allowed me to further tailor my simulation by being able to determine if nodes were actually within another's transmission range. I also added the functionality to generate two different topologys for use with simulation.

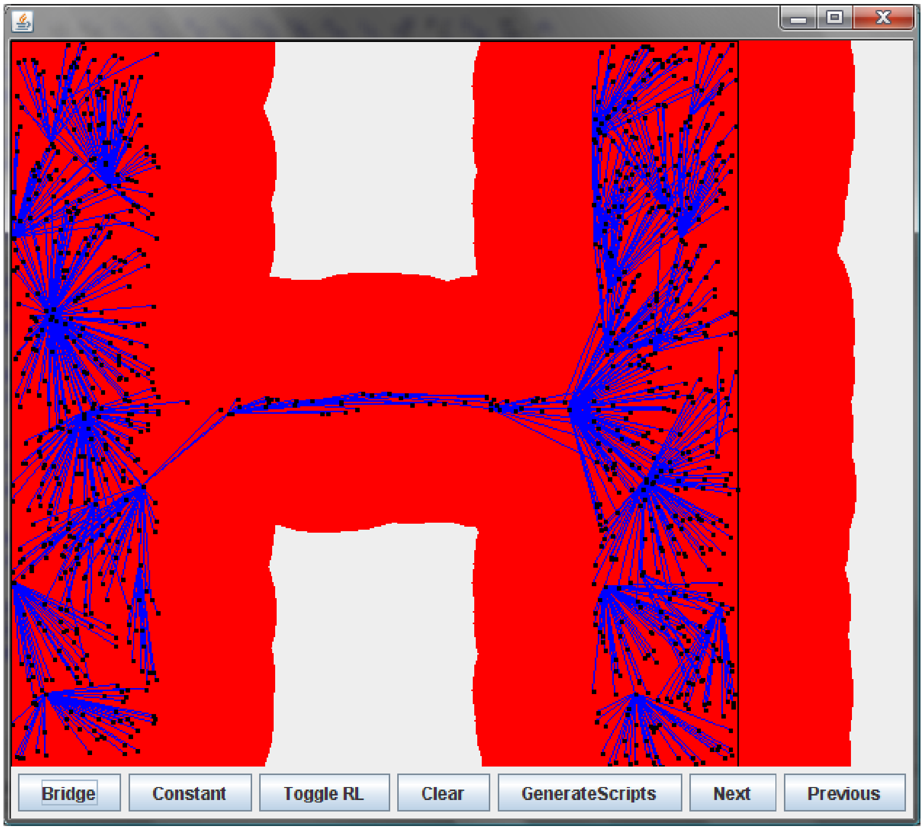

Figure 3: Topology Editor parsing a NS-2 tracefile

- Homogeneous Topology with a constant node density with in the network.
- Heterogeneous Topology with two node clusters on opposite sides of the simulation area, separated by a distance of over twice transmission range, with a small bridge of nodes linking the two sub-networks.

Once a topology is built the program generates traffic files, simulation files and shell scripts to run the simulation and compress the resulting trace files.

4.1 Program Extension

I developed a small extension to the visualisation program which reads the trace file generated by the NS-2 simulator and then display the positions of the nodes and the results for the first message propagation. This made the results of changes to the protocol apparent visually, and helped to determine when to examine the trace file more closely (See figure 3).

5 Optimising the Protocol Code

When I first started running simulations with high density of nodes, the simulations would take hours and the propagation of a single message would take

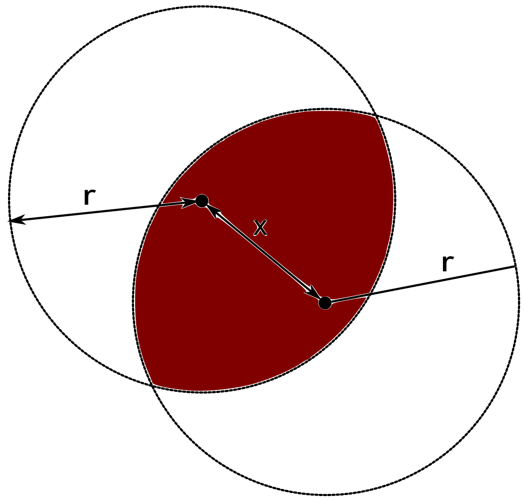

Figure 4: Common Parent Extention: Only messages from within the shaded area would be considered

seconds of simulation time.

I found the problem in the algorithm that inserts a newly received message into a "waiting" queue, where it is stored whilst waiting for a repeat transmission.

The list is ordered by the "expiry time" of the waiting message, and hense it does this by walking through the list and inserting it.

The problem was that the algorithm also walked through the messages that were already expired. Consequently as the node received more messages its processing time increased linearly with having to walk through all of the messages it had received.

This problem was fixed by adding an extra clause in the algorithm so that it stops walking through the linked-list after the next message in the list had an expiry time that was larger than the current message.

6 Results

6.1 Common Parent Extension

When the algorithm uses with the common parent check, I found that the total number of messages broadcast raises considerably. When the nodes were only taking into account the messages which are the direct consequence of the original message they received, the pool of nodes which could cause them to exceed their threshold decreases (See Figure 4).

This was a step towards making the protocol more like a broadcast storm, and not a direction we wished to take it.

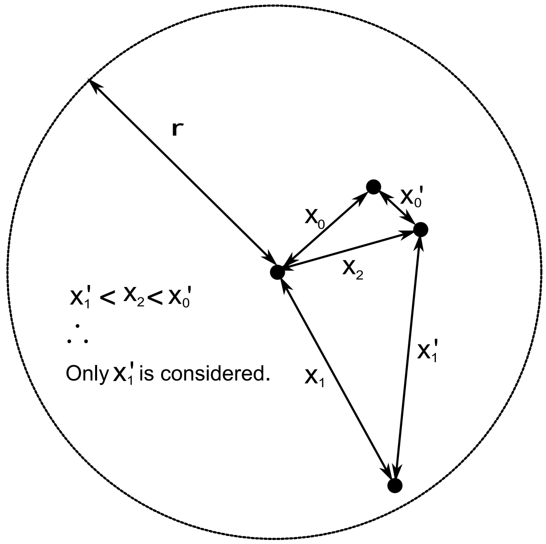

Figure 5: Signal Strength Extention

6.2 Signal Strength

When the algorithm uses the common parent extension, I found that the total number of messages broadcast increased. When the nodes were only taking the messages from nodes more distant into account, the nodes that were in sparesly populated areas of the topology broadcast more often than the unmodified protocol due to the only other nodes nearby did not pass the signal strength criterea (See Figure 5).

This was once again a step towards making the protocol more like a broadcast storm, and so was not desirable.

6.3 A Combination Of The Two

The best results I found was when I used a combination of the two. If the node receives a message that is from the same parent, it treats it as normal, and increments its message counter. Otherwise, it then checks the messages RSSI against its original message, and if it is smaller (hence the node is more distant) then it counts it in its message list.

This seems to allow two individual message propagation paths to combine and allow the message to continue. It does this as it does not accept the messages from one propagation path.

Exactly why this modification works so well is still under examination (See Figure 6).

7 Discussion

The results of the experiments show some interesting problems with this research question. Each node must follow simple rules for the group to exhibit

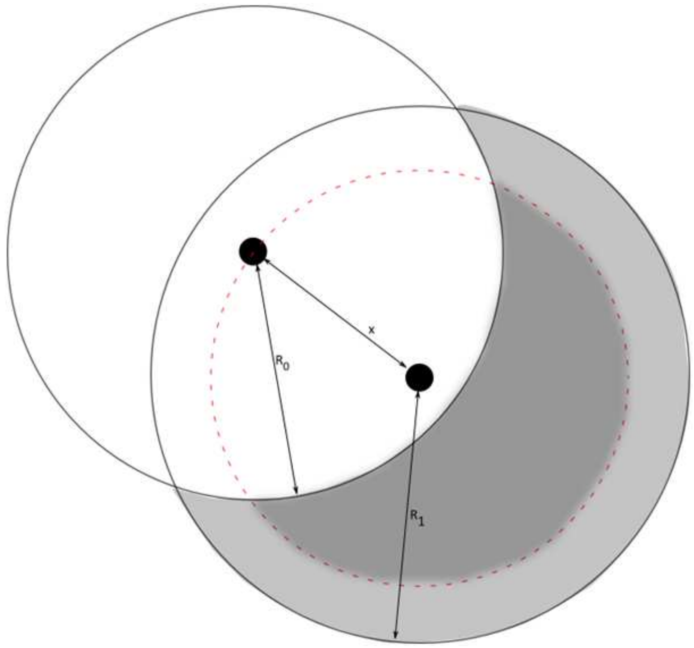

Figure 6: Combination of the two extentions. Nodes without a common parent are only accepted within the lighter shaded region.

a more complex behaviour. The problem is to change the simple behaviour incrementally and then observe the results of the group behaviour. The group behaviour that this research question needs however, is for nodes to broadcast messages only when it is beneficial for the group. Also individual nodes need to be aware of message direction, to allow for messages to continue propagating within the group.

The Combination Extention shows the most promise and does add an element of non-determinism. The area where nodes from a non-common parent may be accepted by a node circles around the range of the listening node. However the nodes that would be transmitting within the time that a node would be listening are much more likely to be at the edges of the acceptable area (See Figure 6), and so more from the same direction.

The problem of nodes being aware of direction was the focus of my visit to the University of Lisbon, and some interesting extentions to the protocol have been considered and can now be researched further.

8 Future Work

In the immediate future, we plan to continue testing the proposed extensions so that their behavior can be fully understood. The extensions, together with extensive simulations that must yet be performed will then be submitted for publication to a relevant conference on the field.

Future research will look more into some more of the problems that PAMPA is currently having, namely making the nodes determine direction more accurately.

Also there is a problem where nodes within the outer reaches of another nodes range fall within the same RSSI value, then they broadcast at the same time, which means that any nodes which receive their broadcast have a number of messages in there buffers straight away. The problem is if there are more nodes than the protocol threshold in the RSSI band, hen the receiving nodes will never broadcast, as their conditions have been met.

The protocol also has to be tested with movement, and with more clusters of nodes to simulate urban environments. A radio propagation model that models buildings needs to be developed to help more accurately simulate this protocol in an environment where it is most likely to be used [1].

A movement model which takes into account the buildings could be integrated with this to provide a full urban simulation model and would improve the accuracy of the simulations

9 Conclusion

To conclude, my visit at the University of Lisbon brought to light some interesting directions for the protocol to take. At current the protocol has been improved by extentions which allow it to take the direction of message propagation into account within its protocol logic.

With further research the protocol could become adapted to many different topologies to give on average a much better performance of broadcast algorithms. This could be used by routing algorithms such as DSR to retain the same capabilities, but at a much lower cost of resources.

References

[1] H. Miranda, S. Leggio, L. Rodrigues, K. Raatikainen: A Power-Aware Broadcasting Algorithm. In *Proc. of The 17th Annual IEEE International Symp. on Personal, Indoor and Mobile Radio Communications (PIMRC'06)*, (2006)

[2] Z. J. Haas, J. Y. Halpern, and L. Li. Gossip-based ad hoc routing. In *Procs. of the 21st Annual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM 2002)*, pages *1707–1716*, 2002.

[3] IEEE Computer Society LAN MAN Standards Committee: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (1999)

[4] Network Simulator 2: *http://www.isi.edu/nsnam/ns/*