CS 685 - Project 3 Report
Comparison between Flood and Trickle in Reliability

Ali Alnajjar

1 Introduction:

In this project we implemented two different retasking algorithms. The first algorithm is simple food and the second algorithm is trickle. There are many element we use to compare the two algorithms number of messages, power usage, Latency, and network overhead are some of them. From my point of view the most important thing is the algorithm's reliability.

2 Algorithm:

2.1 Flood:

In the flooding algorithm the base station send re task messages to the network. Each node that hear the re task message apply the new update then resend the message to the others nods in the network. If the node hears a messages that it already dealt with is should ignore these messages.

2.2 Trickle:

In trickle the base station send the retask messages to the network. Each node hears this re task apply the new update and resume working in the application without resending the re task messages. In trickle each node wait for a sometime the check if it didn't hear a metadata message simile to the metadata it has it will send a meta data message. If a node hears a meta data newer than the meta data it has it will send its meta data. If the node hears older mete data it will send a re task message to the network.

3 Experiment :

3.1 Setup:

3.1.1 Base Station:

The host application in the base station will send a retask message then wait for 20 seconds before sending another retask message. The base station will send 50 re task messages into the network. The base station will collect the temperature, log, and statistics from the spots.

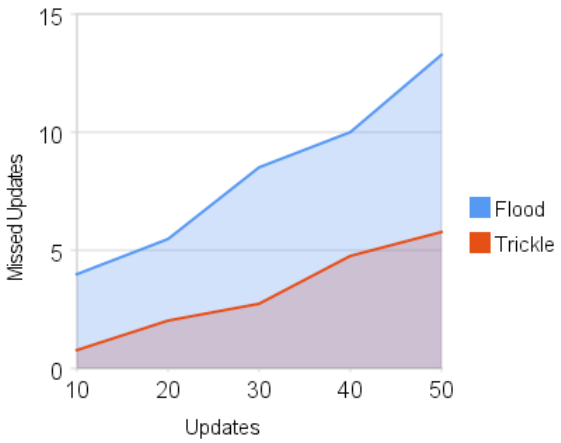
3.1.2 SPOTs:

3.1.2.1 Topology:

I arrange the spot to be in a multi-hop network contains 3 levels A,B and C. Level A spots can communicate directly with the base station and level B spots. The level B spots can communicate with level A and C spots. Finally level C spots can communicate only with level B spots.

3.1.2.2 Software:

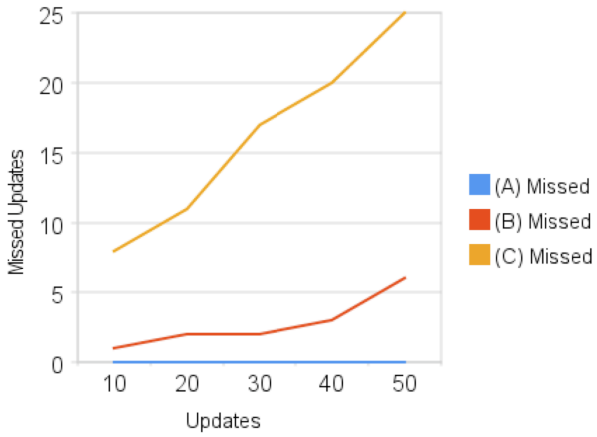
Each spot will collect informations about the received updates and send this data with the temperate to the base station. For trickle setup I set k=1, Tl=5, Th=80, and T=20.



3.2 Results:

As we can see from the above graph in flood the spots missed many messages which was expected. In flood algorithm there is no acknowledges sent to the base station and no resending procedure in the algorithm. On the other hand trickle missed some updates which wasn't expected. This happened inly in this experiment because trickle is slow. The spots far form the

base station may not get the update until the next frame or even more. Which means it missed one or more updates. It is clear from the graph that flood lost more updates than trickle.



From the above graph we can see where flood lose updates. We can see that the (A) level spots didn't missed any updates. With the (B) level spots missed up to 10% of the updates while the level (C) spots missed up to 50% of the updates. It is clear that when the spots are located near the source it loses less updates and the lost increase while we move further.

4 Conclusion:

Trickle may be slower than flood in implementing the updates and causing more network traffic. But from this comparison we can say that we can relay more on trickle to deliver the updates properly.