Project Report - Bonus

Dhruv Mahajan, UFID: 42111994
Ashvini Patel, UFID: 47949297

1) Gossip with failure

Implementation Details:

- **Failure model and parameter:** We use a permanent failure model where a node dies permanently, i.e, any node in the network/ topology can fail with a probability P. **Note:** At any given point we do not control how many nodes in a network fail, only that each node in the network can fail with a probability P.

- **What does failure mean:** We tried to imagine how failure in a true sensor network may occur and then modelled that. Particularly:
  - Failure is completely random and permanent for each gossip round.
  - A node is not aware which of the other nodes in the network have failed.
  - Active Nodes will transmit normally, completely unaware of the failed nodes.
  - Failed nodes will not propagate any messages.

- **Network Topologies Tested:** We tested our failure model for all the topologies but only in two of the topologies, **full and imp2D** message propagation could be achieved with any reliability. The reasons for this are apparent because in both line and 2D it is easy to completely break the network with random failure.

- **Our Failure model is only feasible in gossip. If,** A node is not aware which of the other nodes in the network have failed, failed nodes can be sent messages which will lead to weight loss and the true SUM/AVG can't be computed.
- **Time, Entry point and convergence strategies** same as regular gossip.

All the time values in the table and graph are the middle values of 3 trial runs. This is a good representative of the time values observed.

Testing

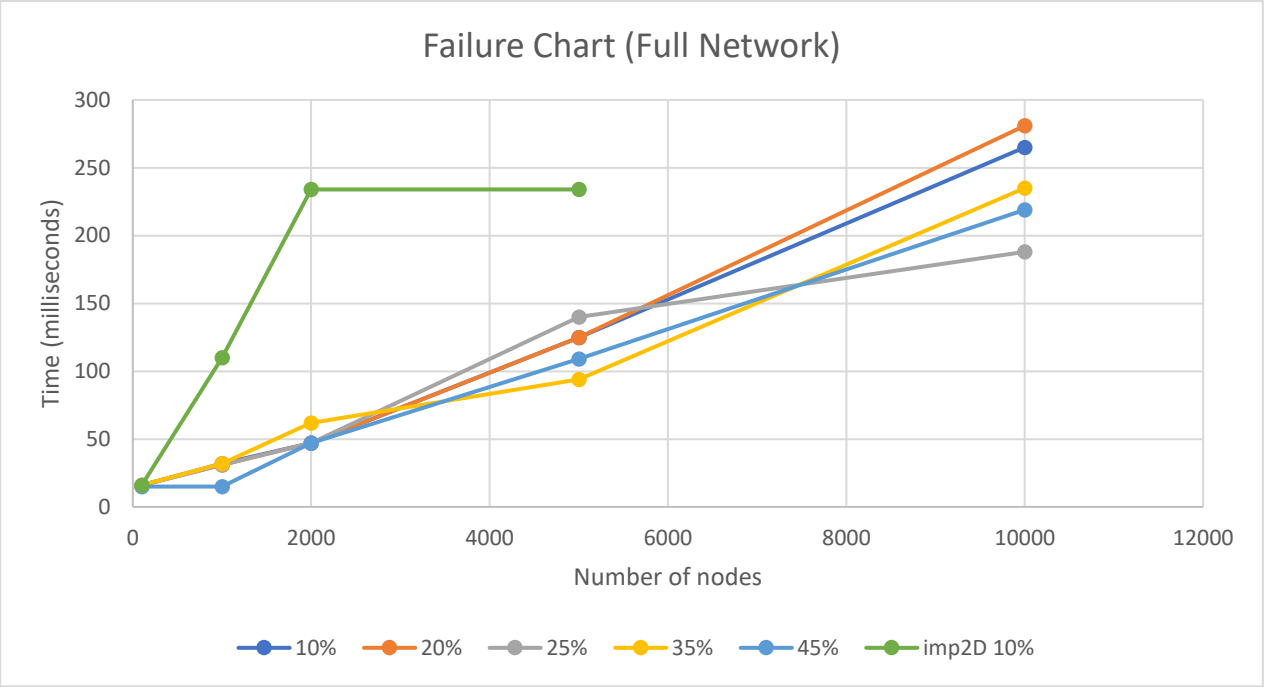
To test, we look at the full and imp2D control the parameter P which is the probability of an individual node failing

This needs to be inputed as the 4^{th} parameter to the code i.e.

./project2_bonus numNodes topology algorithm P

Note: Run project2_bonus and DO NOT add percentage symbol after P. P can take any value from 1 to 100. **Algo. for failure is always gossip.**

Graph



Convergence time in milliseconds for various values of P, **Full and imp2D networks**

Findings

- **Increase in P does not seem to significantly increase in the time taken for rumor propagation/ network termination.** This is in complete contrast to what we expected, the expectation was that increase in P will cause a significantly increase in the time taken for rumor propagation
- For really large **Full** networks, increasing P value, seems to decrease the time for convergence, **Note: For networks with 10,000 nodes, time to convergence is highest for P = 10% and 20%.**

Hypothesis to explain findings

- Since gossip propagation in full networks are truly random, increase in P just results in less number of nodes that have to be told the message/rumor for network termination and doesn't cause any slowdown in information propagation.

Table (P vs No. of nodes for Full Network). Time in Milliseconds.

Number of nodes	10%	20%	25%	35%	45%	imp2D 10%
100	16	16	16	16	15	16
1000	32	31	31	32	15	110
2000	47	47	47	62	47	234
5000	125	125	140	94	109	234
10000	265	281	188	235	219	