COP5615: Fall 2015
PROJECT 2 - Bonus

Team Members:

Aakriti V Agrawal	14331642	aakritiv.agrawal@ufl.edu
Divya Awatramani	46599609	dawatram@ufl.edu

Failure Model:

The zipped folder includes a Gossip Algorithm for information propagation and Push Sum Algorithm for sum computation for four different topologies: Full, Line, 3D, Imperfect 3D; and a failure model for the Gossip Algorithm

Folder Structure:

```
| + src
  | +main
    | + scala
      | + GossipSimulation
        | + GossipSimulation.scala
| + build.sbt
```

To run:

```
> cd Aakriti_Divya_ Project2/Project2_Bonus
> sbt
> compile
> run numNodes topology algorithm
```

numNodes -> no of actors
topology -> full, line, 3D, Imp3D
algorithm -> Gossip or PushSum

What is the largest network you managed to deal with for each type of topology and algorithm?

Gossip Algorithm:

- Full -> 2500
- Line -> does not converge
- 3D -> 1000
- Imp3D -> 1000

Working modules:

- Gossip Failure for all the four mentioned topologies
  - Full
  - Line
  - 3D
  - Imperfect 3D

Implementation:

Gossip:

- Master randomly selects a node and triggers it to start the gossip and it works as mentioned before. The only difference is the master starts killing the nodes in parallel after some time (20 milliseconds after the gossip protocol has started) and the master continues killing the nodes until it has killed one fifth of the initial nodes.
- Also as here is no recovery model or fault tolerant model implemented hence this unexpected killing of nodes hamper the performance of some of the topologies and the convergence reaches faster and in some cases the system never converges.
  - Converges faster: if the node has received a message and gets killed after that.
  - Never converges: when the node has not received the message at all and gets killed before that than the system keeps waiting for all the nodes to hear the gossip.

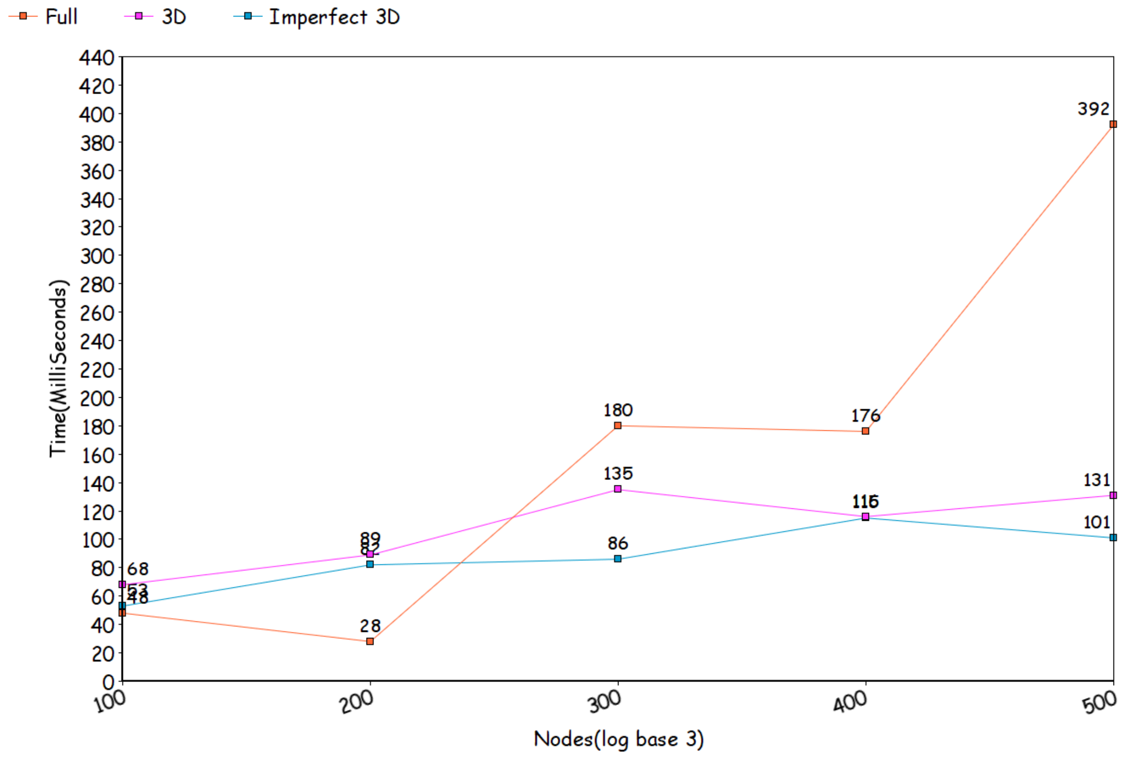
Observations:

- For full topology and 100 nodes: if one node dies in in the initial 20 milliseconds than the convergence was reached faster from 105 to 50 milliseconds.
- The network do not converge if number of nodes is greater than
  - 2500 for full topology
  - 1000 for 3D topology
  - 3000 for Imperfect 3D topology
- Another interesting observation was that the line topology did not converge even at 100 nodes.

Recovery/Fault tolerance logic:

- If a node has received the gossip 10 times than the node stops itself and also notifies all its neighbors. In this case all its neighbors will remove that node from their neighbor List. Hence there will no more message transmitted to the passive node and that node will be out of the system which increases the rate at which the gossip transmits.
- Also if node that is getting stopped/passive is the last neighbor of say node 'B' than we need to remove node B also from the system and will be stopped
- Also if a node dies abruptly or unexpectedly than without informing its neighbors than that scenario can be handled in a way, that whenever a node transmits the gossip it needs to check that status of the receiver node (node.isTerminate==true) if it is alive than only the message is send to that node otherwise some different neighbor is selected for gossiping.

Graph for Dependency of convergence time of each topology as a function of the size of the network



Convergence Time:

NumNodes	Time (in milliseconds)			
	Full	3D	Imperfect 3D	Line
100	48	68	53	--
200	28	89	82	--
300	180	135	86	--
400	176	116	115	--
500	392	131	101	--
1000	130	--	--	--
1500	2173	--	--	--
2000	3161	--	--	--
2500	--	--	--	--