

# **COP5615: Distributed Operating Systems**

## **Project -2 - Bonus**

Submitted by:

Aarthi Kashikar, UFID: 0366-8968

Sri Kanth Juluru, UFID: 9279-1918

Contents of zip folder:

1. Main project folder (project2)
2. Report.pdf

We can run the project by going inside the project2 folder, and run this command

```
dotnet fsi --langversion:preview proj2Bonus.fsx numNodes topology algorithm nodesToKill
```

In place of numNodes, we will pass number of nodes. Possible values of topology are “Line”, “2D”, “FullNetwork”, and “Imperfect2D”. For example, dotnet fsi --langversion:preview proj2Bonus.fsx 1000 2D Gossip 100

### **Experiment:**

We are implementing the failure model on Gossip algorithm where some nodes are dead in the topology and analysing the performance on the same. Here, we give the program beforehand the number of nodes that we intend to be dead. We let gossip algorithms run and check how dead nodes are impacting the algorithm, how many nodes are reaching the convergence for all the topologies and make some observations.

### **Implementation details:**

The implementation follows as below:

- 1) The commands number of nodes, topology, the algorithm to be performed and number of nodes to be dead are received from the command prompt.
- 2) From all the nodes, the given number of nodes are randomly killed using poisonpill.
- 3) Here we have implemented the Gossip algorithm for our experiment.
- 4) The master actor selects a random alive actor and shares the gossip.
- 5) The gossip is then shared to the neighbour. In case the neighbour is alive it receives the information and if the neighbour is dead, we can see deadLetters info in the terminal indication that the receiving actor is dead.
- 6) This is continued and till all the alive actors are converged i.e., receives the information 10 times. In case, all the alive actors could not be converged, the program terminates after timeout(3 minutes in our case).
- 7) A graph is plotted to know the behaviour of all the topologies in failure model. Number of nodes initiated is plotted on x-axis and number of nodes converged is plotted on y-axis.

**Findings :**

Among all the topologies, we can clearly see that line topology has the least number of converged nodes. This is because if any node is killed in this topology it breaks the connection to its neighbour actors which forms independent clusters in the network making it difficult for the actors to communicate. Full network topology has the highest number of converged nodes. Performance of this topology is better because each node is talking to the rest of the nodes and hence the probability of message reaching the node is high.

Another interesting finding is that, for full network and 2D, when we gave initial nodes as 400 and number of nodes to be killed as 100, we saw that 316 nodes were converged. This is because we are killing the actors using the poisonpill command. Poisonpill performs some actions like termination child node etc., before it kills the actor. Some actors got converged before they could be killed and included in the converged node count by master node. Hence, we are seeing more than 300 nodes to be converged.

	Convergent nodes			
Number of nodes	Line	2D	Full Network	Imperfect2D
400	10	316	316	319
800	21	708	708	703
1000	26	907	908	994
2000	53	960	1902	1794
4000	107	1697	3901	2710
10000	233	3634	9900	5165

## Failure in Gossip

