COP 5615 – PROJECT 3

Chord Simulator (Bonus Readme)

Sourav Kumar Parmar (UFID: 86511933)
Priyanshu Pandey (UFID: 21081358)

Instructions to run the program Build system: Linux (Ubuntu):

- Unzip the **PriyanshuPandey_SouravParmar_Project3.zip**.
- Change directory to the following
  Path: *PriyanshuPandey_SouravParmar_Project3/Project3*
- At following path “Project3/src/main/scala” perform below two steps.
- Change the filename of project3_bonus.txt to project3_bonus.scala
- Change the filename of project3.scala to project3.txt
- Execute the following command:
  sbt "project project3" "run <numNodes> <numRequests>"

--numNodes = number of peers in the network.
--numRequests = number of key look-up requests sent by each node.

eg: ***sbt "project project3" "run 1000 2"***

Failure Model

Once the message passing has started, one fifth nodes are killed at random. A successor list is maintained to save the states of successors of current node.

InfectOthers: Method to select the nodes to be killed. A boolean is mainteined to indicate node live status. The respective node recieves “Die” where it updates its successor and predecessor before leaving.

notifyPredecessordied: This method inform succesor of the dying node to update its predecessor with predecessor of dying node.

notifySuccessordied: This method inform predecessor of dying node to update its successor and fingertable using information provided by dead node. Here stablize method is called to fix network.

Stabilize: This method check if the successor is alive or not. If it is alive it checks for successor predecessor say pred. If pred is not null and it lies in open interval of node and its successor the figertable entry of node is updated with pred. Here it also updates its successor list

Also a ping method is implemented: though not used in current implementation.
Here three things are performed:

a) Fix_finger: node periodically checks if its random finger is consistent. If it is not it updates it.
b) Stabilize: same as described above
c) Check_predecessor: Check if the predecessor of node is alive or not.

Observation:

Here the averagehop count was compared with the average in case of no failure model. It was observed that there was an increased delay.

No of nodes	Avg Hop count model	Avg Hop count failure model
10	1.5	1.8
50	4.28	5.32
100	4.81	5.4
200	6	6.1

Though delay was observed it was not that significant.