COP5615 – Project2

Priyam Selugar UFID: 9423-1353

Maharishi Doshi, UFID:0533-9287

Email: priyamselugar@ufl.edu, maharshi.doshi@ufl.edu

Oct 1, 2019

1. Introduction

The goal is to determine the convergence of such algorithms through a simulator based on actors written in Elixir along with failure models.

In this project we manage to use two algorithms, Gossip algorithm, on 6 different topologies: Honey Comb, Random Honey Comb, Full Network, 3D Torus, Random 2D, Line.

• Full Network: Every actor is a neighbor of all other actors. That is, every actor can talk directly to any other actor.

• Line: Actors are arranged in a line. Each actor has only 2 neighbors (one left and one right, unless you are the first or last actor).

• Random 2D Grid: Actors are randomly position at x, y coordinates on a [0-1.0] x [0-1.0] square. Two actors are connected if they are within .1 distance to other actors.

• 3D torus Grid: Actors form a 3D grid. The actors can only talk to the grid neighbors. And, the actors on outer surface are connected to other actors on opposite side, such that degree of each actor is 6.

• Honeycomb: Actors are arranged in form of hexagons. Two actors are connected if they are connected to each other. Each actor has maximum degree 3.

• Honeycomb with a random neighbor: Actors are arranged in form of hexagons (Similar to Honeycomb). The only difference is that every node has one extra connection to a random node in the entire network.

Assumptions made:

- For 3DTorus, we assume that the number of nodes in the grid should be a perfect cube number.
- For honeycomb and random honeycomb, we take perfect square only.
- For Random 2D topology, we generate numbers up to 3 decimal digits, so only 1000 nodes will have distinct co-ordinates. If more than 1000 nodes are created, then topology won't

be formed. Also, random number generation and calculating distance takes $0(n^2)$ time, so the largest network for Random 2D grid topology is 1000.

- Here when we say a node fails, in practical scenario the outgoing connections fail.

Failure Models Implemented:(We have implemented failure model on Gossip algorithm only)

1. 10% Failure Model
2. 20% Failure Model
3. 40% Failure Model
4. 90% Failure Model

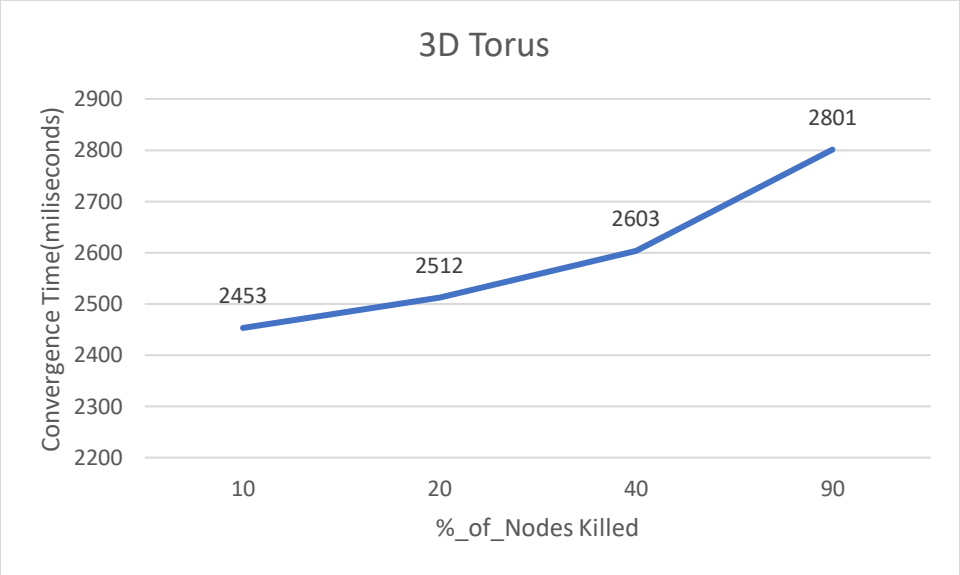
How to run code:

- Open Command Prompt in proj2_v1 directory
- Type in command: mix run my_program.exs (number of nodes) (topology) (algorithm)(no of nodes to fail)
- Example: mix run my_program.exs 100 line gossip 10

Result

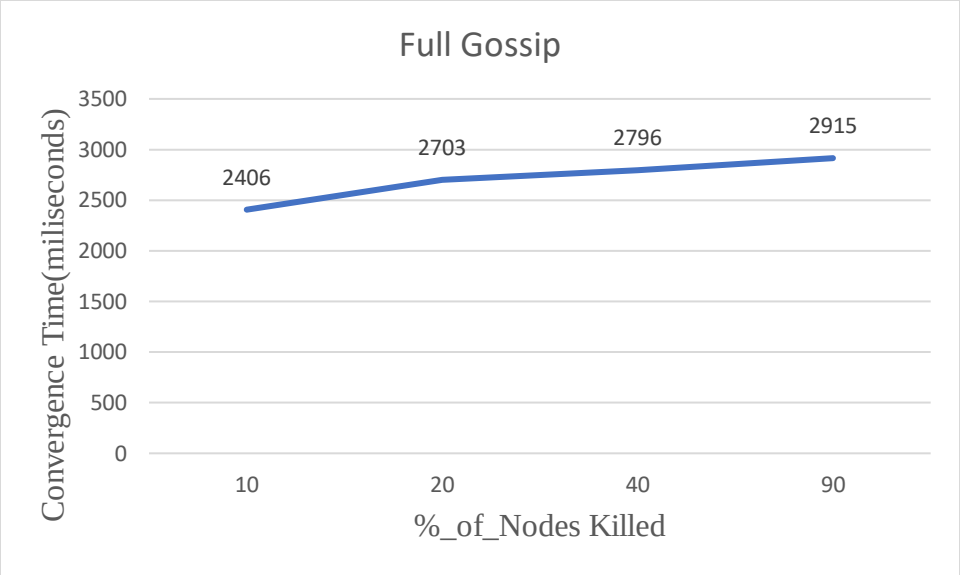
3D Torus

Kill Nodes Number	Convergence Time(ms)
10%	2453
20%	2312
40%	2203
90%	2500



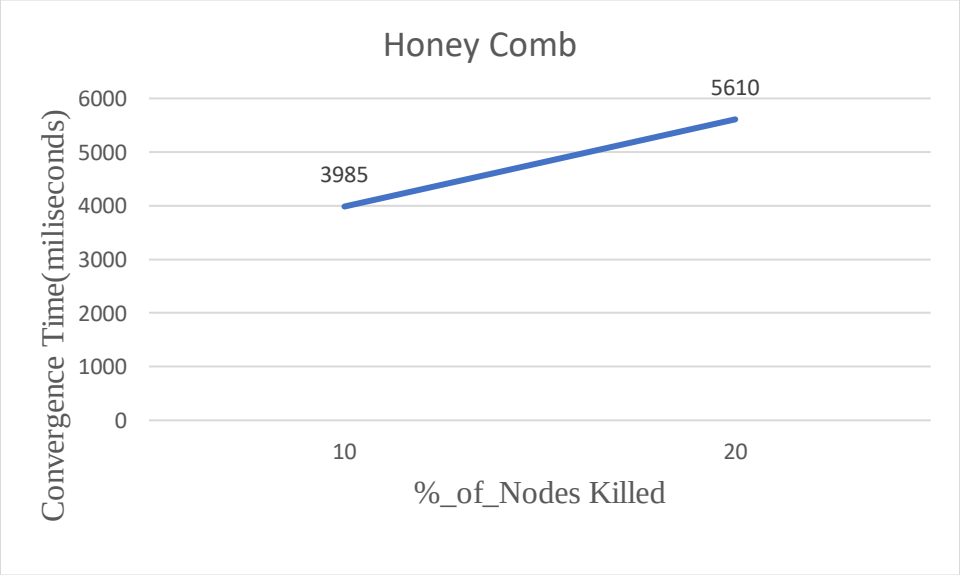
Full

Kill Nodes Number	Convergence Time(ms)
10%	2406
20%	2703
40%	2796
90%	2915



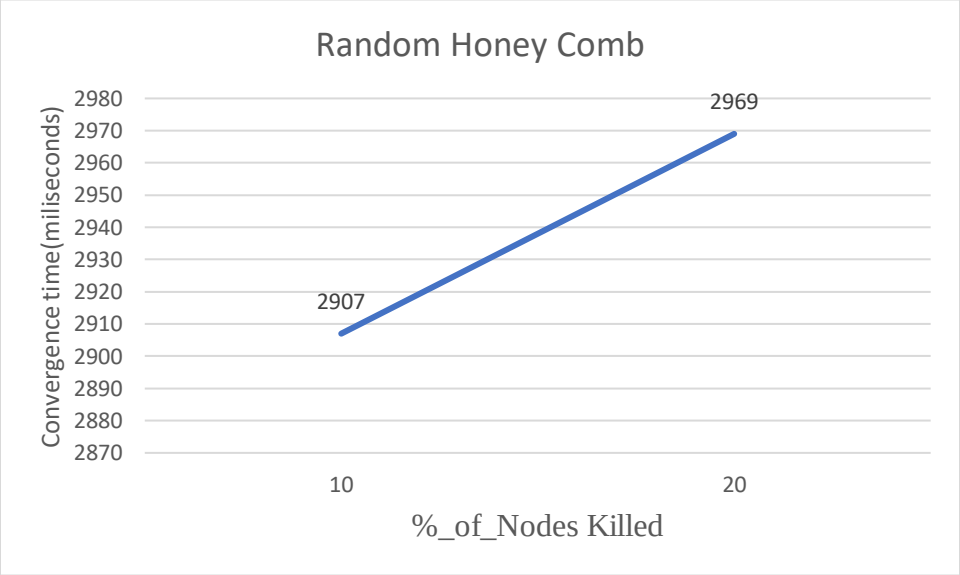
Honey Comb

Kill Nodes Number	Convergence Time(ms)
10%	3985
20%	5610
40%	-
90%	-



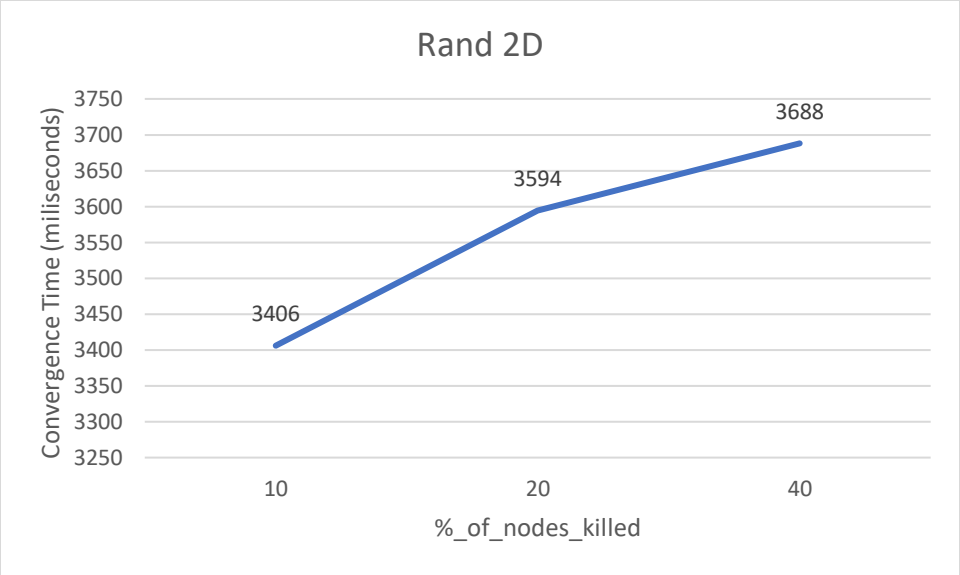
Random Honey Comb

Kill Nodes Number	Convergence Time(ms)
10%	2907
20%	2969
40%	-
90%	-



Rand 2D

Kill Nodes Number	Convergence Time(ms)
10%	3406
20%	3594
40%	3188
90%	-



Interesting finding for Gossip Protocol

1. Line Topology will never converge in any failure model.

2. For Full Topology, even if 90% of nodes fail, gossip protocol still converges.

3. For Torus Topology as well, even when 90% of nodes fail, gossip protocol still converges.

4. As the number of failure nodes increase, in some topologies, the convergence time decreases while in most topologies, it increases.