# Project 2

**By:** Robert Robillard & William B. Anderson

## Overview

- Using the GenSever functionality of Elixir we constructed a network supervisor which maintained child nodes. The network is responsible for starting up the nodes and initializing the registry to be able to reference each node in a programmatic way.

- In order for the Network to startup a Node it does the following:
    1) Finds all the neighbors of each node based on the topology given
    2) Initialize a node with its neighbors, the algorithm used, and an initial state (based on the algorithm)

- Once all are initialized, nodes are able to send messages to each other. Then, when the algorithm is started by the network the nodes will do so, according to the type of algorithm, until termination.

- The time was measured using System.monotonic_time(), which is a safe way to calculate differences within a program. The time starts directly before the initial message is sent and stops when the system terminates. The time is stored in the :native unit and displayed in seconds.

# Results Obtained

- Interesting Findings
    - In the Full Topology when using the gossip algorithm, the percentage of nodes reached drops dramatically at points, while the time to completion doesn't.

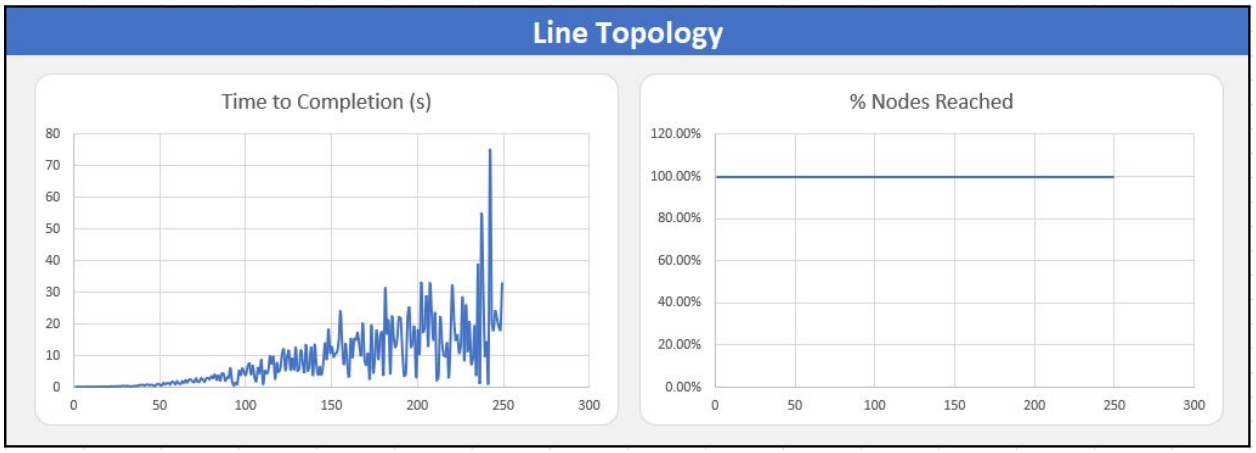
    - There was almost 100% of nodes reached in all topologies when using the push-sum algorithm.

    - In the Random 2D topology, only a small percentage of the nodes are reached up until 210 nodes are used. Then the number becomes 100% (for as many nodes as we tested)

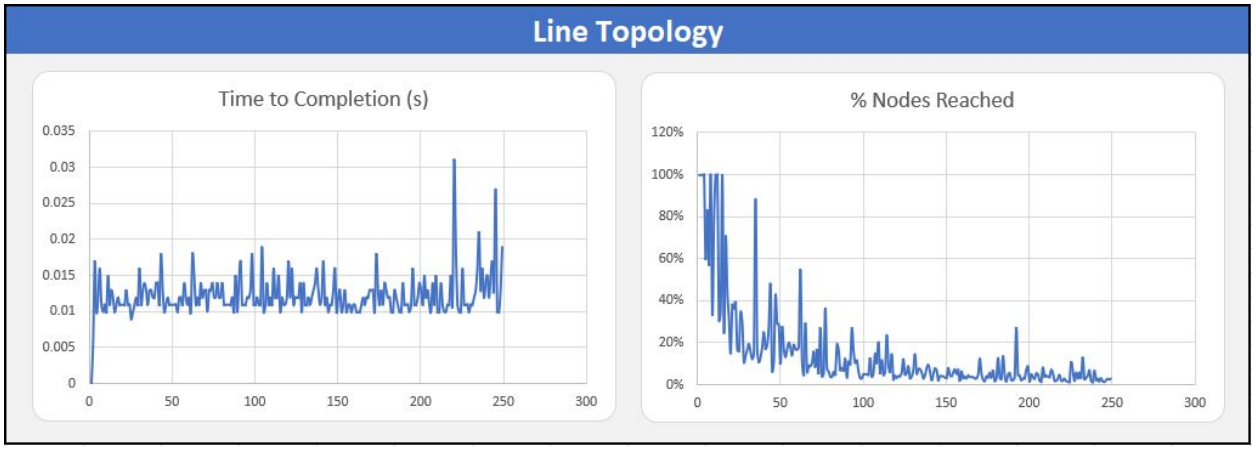
    - In the Random Honeycomb topology, the time to completion fluctuates drastically and often, yet the percentage of nodes reached remains relatively constant for push-sum algorithm. Whereas in the gossip algorithm the time to complete is relatively constantly increasing, and the percentage of nodes reached varies substantially.

# Line Topology

Push - Sum

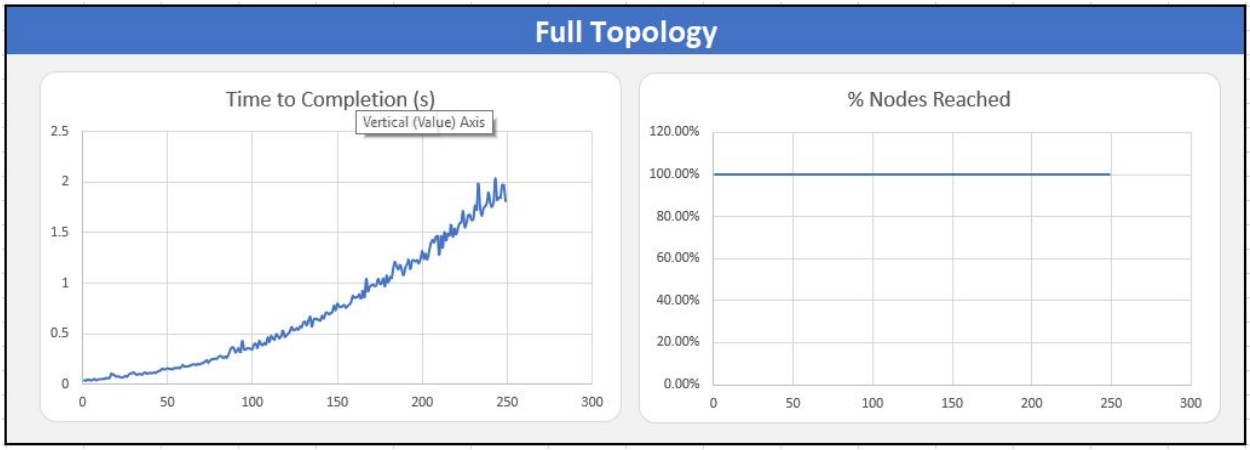



Gossip

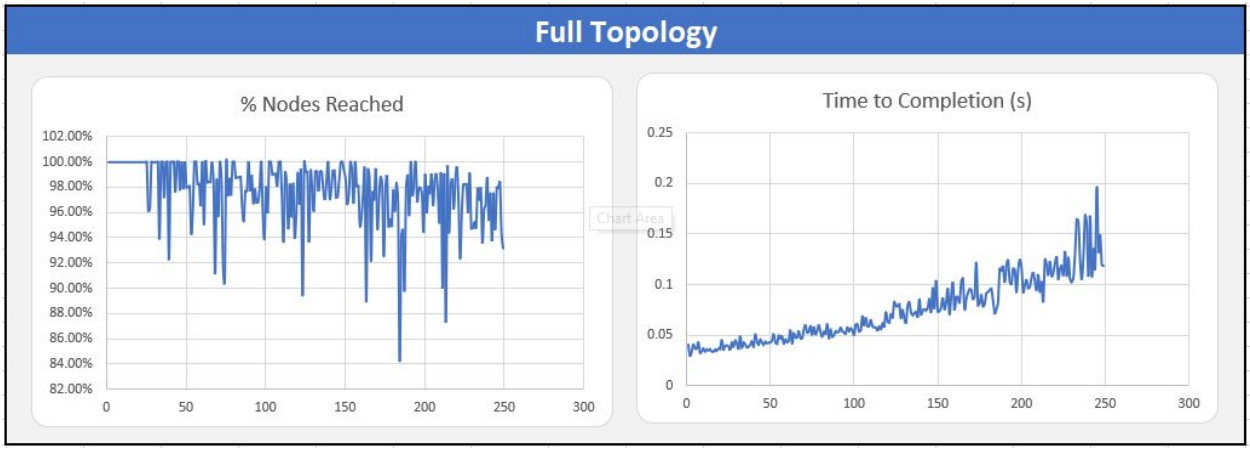


# Full Topology

Push - Sum

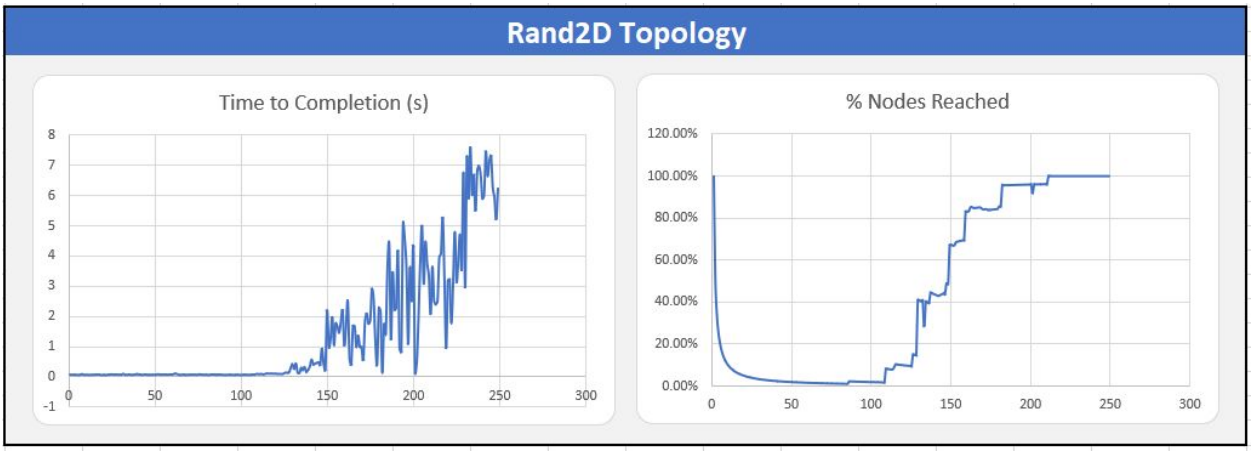



Gossip

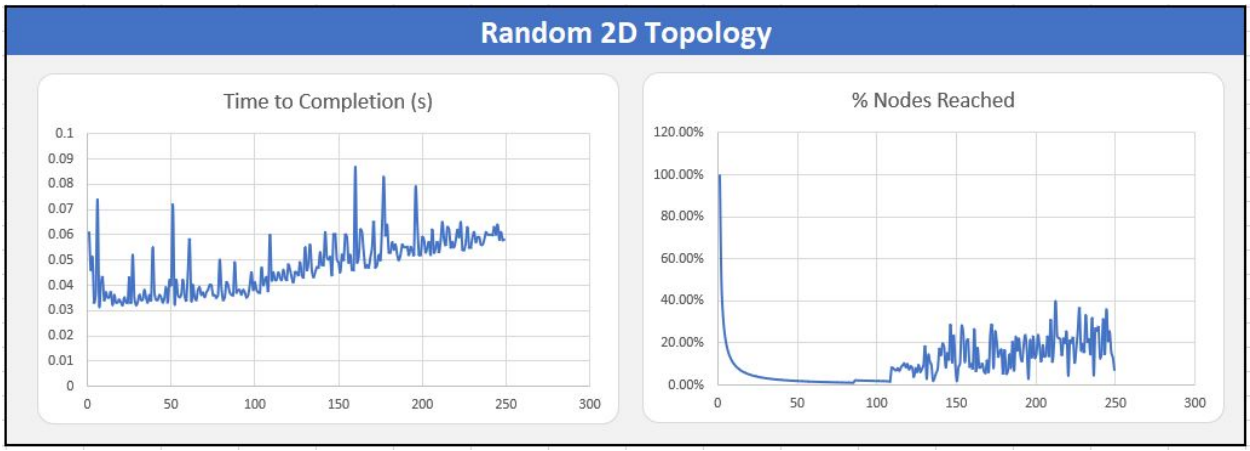


# Random 2D Topology

Push - Sum

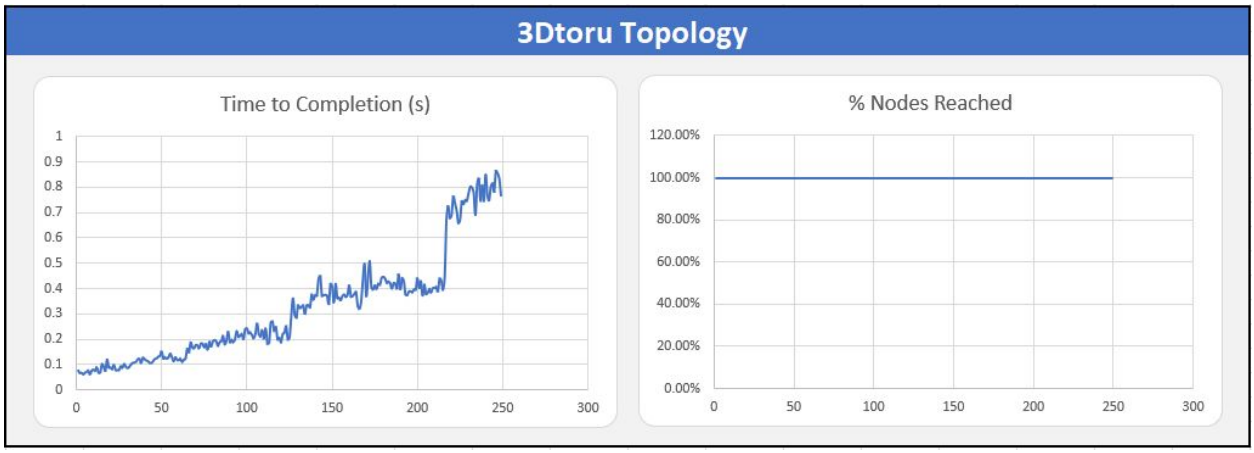



Gossip

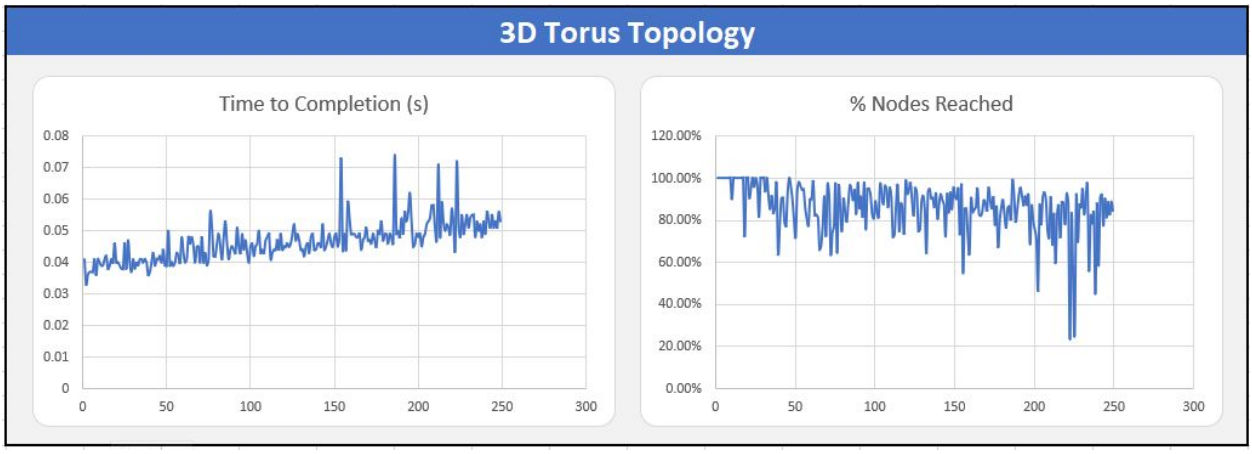


# 3D Torus Topology

Push - Sum

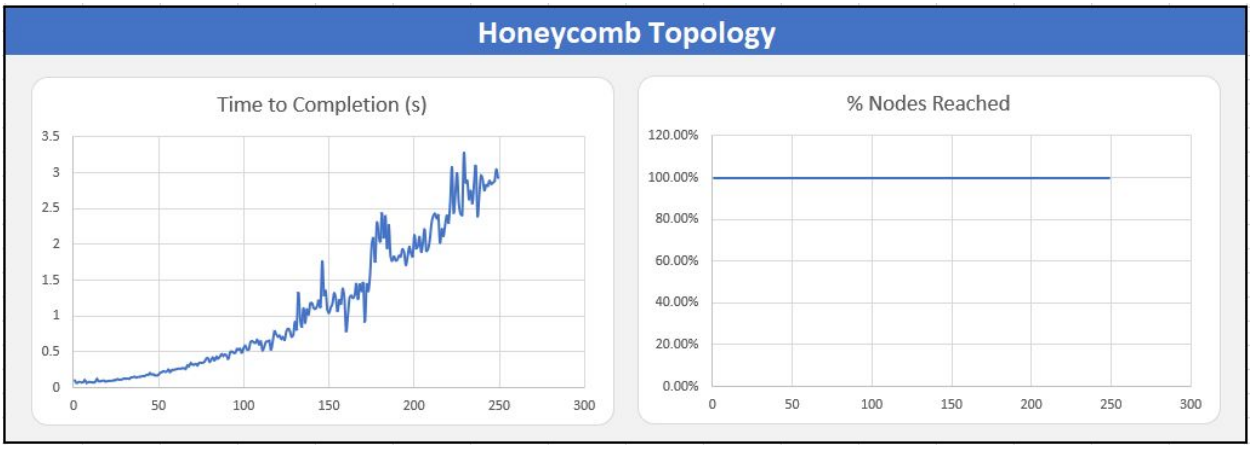



Gossip

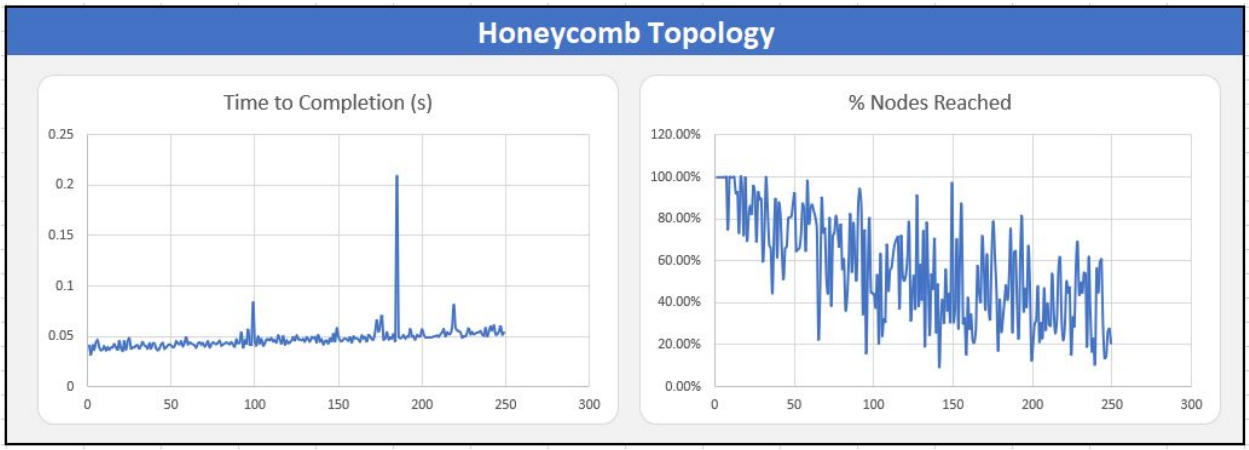


# Honeycomb

Push - Sum

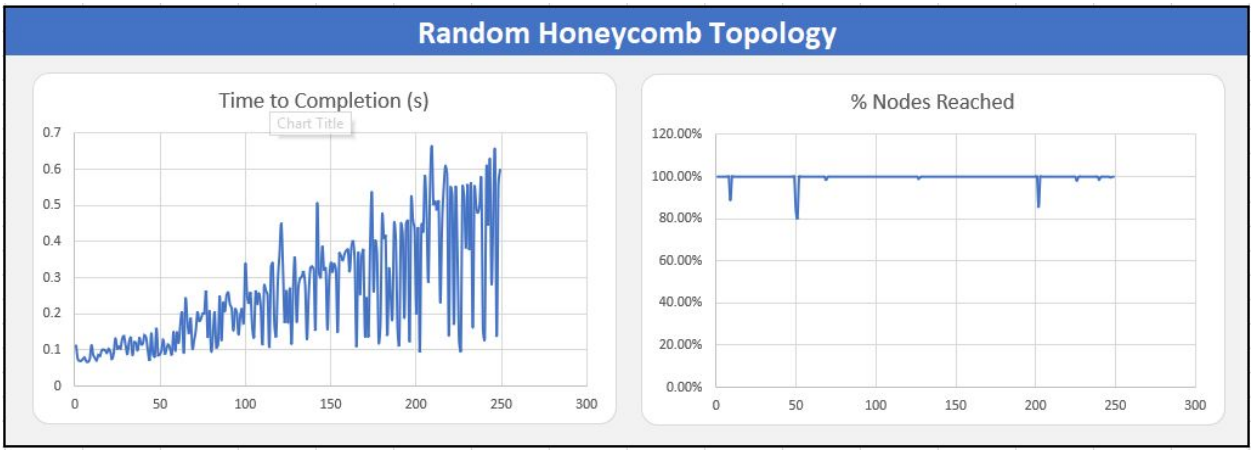



Gossip



# Random Honeycomb

Push - Sum




Gossip