COP5615 Distributed Operating Systems Principles

Programming Assignment 3

Group 18

by
Team Members

Name	Email	UFID
Surya Teja Duppanapudi	**suryatej.duppana@ufl.edu**	**44084702**
Durga Sai Surya Ram Saladi	**dsaladi@ufl.edu**	**96891108**
Sushanth Reddy Kotha	**su.kotha@ufl.edu**	**67664160**
Prajay Yalamanchili	**yalamanchili.p@ufl.edu**	**16891669**

Index:

1. What is working?
2. How to run?
3. Executions & Results
4. Largest network managed to deal with each type of topology and algorithm?
5. Graphs for each type of topology and algorithm

1. What is working?

- Four different network topologies, namely line, full, 2D, and Imp3D, were implemented in the study.
- The gossip and push-sum algorithms were successfully executed for all four topologies.
- The gossip algorithm reached convergence when all nodes heard the rumor ten times, and the total time taken was printed by the supervisor upon achievement.
- We checked the running of the code for all topologies for the number of nodes upto 1500.

2. How to Run?

- To run the code, we need 3 inputs namely, the number of nodes, the topology (line/full/ 2D/imp3D), and the algorithm(gossip/push-sum).
- **Input:** dotnet run numberOfNodes topology algorithm
- **Ex:** dotnet run 20 line gossip

3. Execution & Results

```
C:\Users\sushanth\Downloads\gossip_project>dotnet run 20 line push-sum
Starting Push Sum Protocol for Line
------------------------------------------------------------
Convergence Time: 151.830400 ms

C:\Users\sushanth\Downloads\gossip_project>dotnet run 200 line gossip
Starting Protocol Gossip
------------------------------------------------------------
Convergence Time: 487.722700 ms

C:\Users\sushanth\Downloads\gossip_project>dotnet run 20 full gossip
Use Of Gossip Protocol
------------------------------------------------------------
Convergence Time: 18.300900 ms

C:\Users\sushanth\Downloads\gossip_project>dotnet run 39 full push-sum
Push Sum Started
------------------------------------------------------------
Convergence Time: 28.924600 ms

C:\Users\sushanth\Downloads\gossip_project>dotnet run 488 2D gossip
Use Of Gossip Protocol
------------------------------------------------------------
Convergence Time: 2879.487800 ms

C:\Users\sushanth\Downloads\gossip_project>dotnet run 399 2D push-sum
Push Sum Started
------------------------------------------------------------
Convergence Time: 26523.709600 ms
```

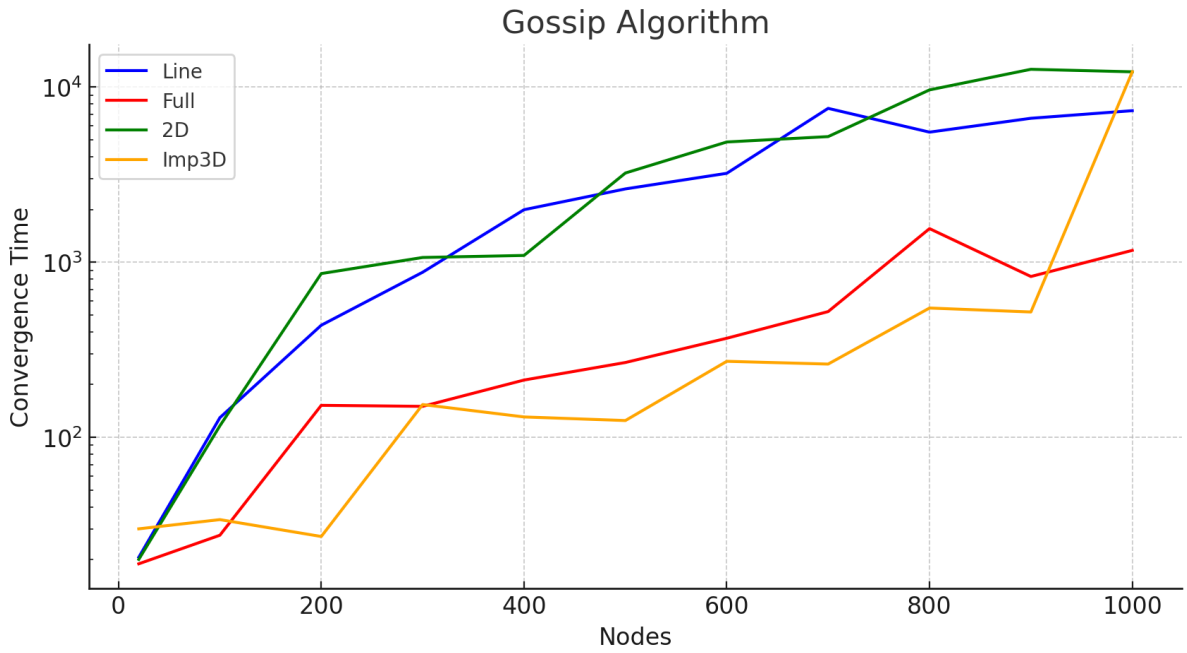
```
C:\Users\sushanth\Downloads\gossip_project>dotnet run 789 Imp3D push-sum
Push Sum Started
------------------------------------------------------------
Convergence Time: 1208.159400 ms

C:\Users\sushanth\Downloads\gossip_project>dotnet run 133 Imp3D gossip
Use Of Gossip Protocol
------------------------------------------------------------
Convergence Time: 61.121500 ms
```

4. Largest network for each type of topology and algorithm

Topology	Gossip	Push sum
Full Network	2000	2000
2D Grid	1100	1000
Line	1200	1000
Imperfect 3D Grid	2000	2000

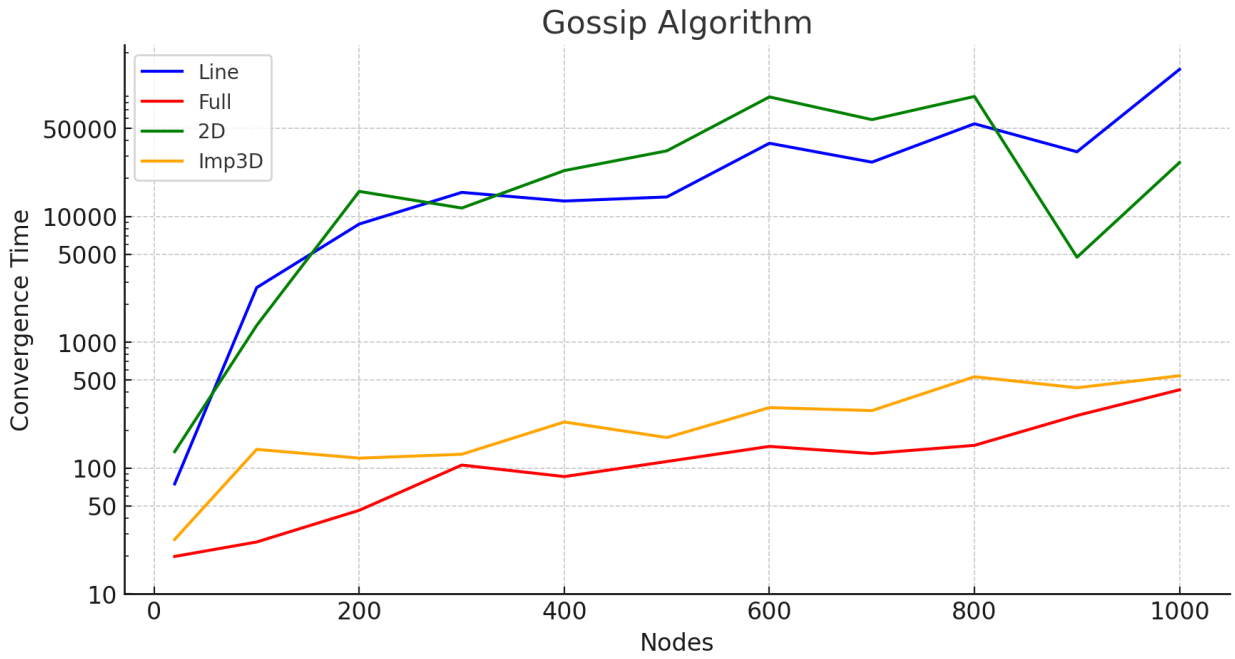
5. Convergence time vs size of the network Graph

Gossip Algorithm:



#Nodes	line	full	2D	Imp3D
20	20.68	18.97	20.11	30.04
100	129.49	27.61	116.36	33.91
200	436.40	152.29	860.62	27.16
300	875.73	150.24	1063.35	153.85
400	1992.27	212.32	1092.14	130.73
500	2618.29	267.38	3226.73	124.69
600	3214.54	367.72	4851.94	271.62
700	7548.45	522.16	5207.95	261.95
800	5522.17	1553.60	9621.80	547.16
900	6626.31	828.07	12614.34	519.38
1000	7322.90	1167.20	12203.49	12203.49

Push-Sum Algorithm:



#Nodes	line	full	2D	Imp3D
20	74.78	19.83	134.88	27.06
100	2717.23	25.84	1360.50	140.76
200	8695.51	46.13	15806.46	119.85
300	15517.12	105.55	11654.63	128.65
400	13251.76	85.54	23125.06	232.29
500	14271.60	112.69	33201.60	174.68
600	38139.77	148.56	89039.30	302.16
700	26987.17	130.43	58778.68	286.17
800	54484.09	151.46	89820.94	531.63
900	32632.50	261.58	4738.33	434.52
1000	147447.74	418.63	26818.37	541.43

Graph Analysis

- In both graphs, the 'line' algorithm seems to be the least scalable, as it shows the highest convergence times with increasing nodes.
- The 'full' algorithm appears to be the most stable and scalable across both datasets, showing minimal change in convergence time.
- The '2D' algorithm is the least predictable, with wide variations in performance at different node levels in the updated graph, while in the initial graph, it was only second to the 'line' algorithm in poor scalability.
- The 'Imp3D' shows moderate scalability in both graphs, but its performance worsens with a larger number of nodes.

These analyses suggest that the 'full' algorithm may be the most suitable for scenarios where the number of nodes is dynamic and can grow to a large scale, while the 'line' algorithm could be more appropriate for smaller, more controlled environments. The '2D' algorithm might require further investigation to understand the conditions under which it performs optimally or suboptimally.