**COP5615 Distributed Operating System**

**Project 2 Report**

**# Group Members**

1. Bhavya Kavdia, UFID : 7017-7277

2.Ishita Trivedi, UFID : 6893-6496

**Problem Statement -**

To calculate the convergence of gossip algorithms – gossip algorithm which is used for communication and push-sum algorithm which is used for computing the aggregation by simulating models for the two algorithms. The user will input parameters which indicate the different network topologies (Full, Line, 3D, Imperfect 3D) to be used and number of nodes in the network. The goal is to experiment both protocols with different network topologies using actor and AKKA model and find the convergence time.

**Implementation of Gossip Model –**

* We have tried implementing Gossip model using 2 approaches –

1. In the first approach we waited for all the actors to send messages and terminate the program when all the actors terminate. In this case we noticed that some actors never receive messages and the program does not terminate. Our line algorithm was working sequentially instead of working parallelly.
2. In the second approach we tried to solve the problem arising in first approach. We did so by making each actor maintain a list of its alive neighbors and sending messages only to alive neighbors. The logic behind this approach is that if the actor does not have any alive neighbors, then it will not send any further messages and will terminate itself and hence the program also converges once all the actors have received the rumor at least once.

**Implementation of Push- Sum Model –**

* We have tried implementing Push- Sum model using 2 approaches –

1. In the first model we developed the model in such a way that the program would converge when the s/w ratio present at each node does not change more than 10 ^ -10 even after three rounds. When we ran this approach, and few nodes did not converge and as a result the program did not terminate. We observed that the system was running into a loop where few actors were continuously sending message without stopping or few actors stopped sending messages and their neighbors were not receiving any messages. In this model since the actors were sending messages to random neighbors the performance was quite low.
2. In the second model we made each actor send rumors only to its alive neighbors, which resulted in better performance and s/w ratios. Since the actors are technically sending messages only to alive nodes, the rumor is meaningful, and the program converges.

One important thing to note is that we have set up an Actor Dispatcher which continuously sends messages at an interval of 1000 ms to the actor itself. Meanwhile, during the dispatch wait time, other actors are picked up and messages are sent to them to ensure that all actors are running concurrently.

**Graphs –**

1. There are two graphs one for each protocol – Gossip and Push- Sum which denotes the running time for each of the 4 different network topologies – Full, Line, 3D, Imperfect 3D for increasing no of nodes.
2. The x-axis denotes the no of nodes in the network, and the y-axis denotes the running time or time taken for the model to converge in ms.

**Gossip :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nodes** | **Full** | **Line** | **3D** | **Imperfect 3D** |
| 100 | 22 | 1040 | 13 | 15 |
| 250 | 45 | 5147 | 24 | 44 |
| 400 | 103 | 2086 | 88 | 72 |
| 500 | 171 | 5200 | 66 | 60 |
| 700 | 171 | 10260 | 66 | 79 |
| 800 | 258 | 14353 | 57 | 58 |
| 1000 | 399 | 14404 | 72 | 85 |
| 1500 | 1303 | 26683 | 198 | 157 |
| 2000 | 1157 |  | 359 | 216 |
| 3000 | 2349 |  | 450 | 254 |
| 5000 | 7007 |  | 710 | 516 |
| 7000 | 11795 |  | 1026 | 899 |
| 8000 | 15877 |  | 815 | 1050 |
| 10000 | 45122 |  | 1306 | 1516 |
| 15000 | 125635 |  | 5098 | 3179 |
| 20000 |  |  | 10165 | 7077 |
| 50000 |  |  | 43662 | 83183 |
| 100000 |  |  | 244153 | 362088 |

Chart, line chart

Description automatically generated

Observations: The results for Gossip implementation using second approach indicate that 3D and imperfect 3D had almost the same performance which is better than rest of the topologies. Line topology showed significantly higher running time and eventually stopped converging for larger no of nodes. One possible explanation for this could be that in line topology, each actor spreads rumors only to 2 of its neighbors, in the right and left and thus convergence takes time. The full topology works fine for lower values of nodes but shows exceptionally high running time for higher no of nodes which might be possible due to the large no of neighbors the rumor has to spread to.

**Push- Sum:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nodes** | **Full** | **Line** | **3D** | **Imperfect 3D** |
| 100 | 97 | 1068 | 1042 | 1048 |
| 250 | 1214 | 1413 | 1717 | 1074 |
| 400 | 3826 | 1394 | 3412 | 1237 |
| 500 | 5264 | 1456 | 3897 | 1158 |
| 700 | 18688 | 1735 | 4555 | 1373 |
| 800 | 21555 | 2671 | 4644 | 3364 |
| 1000 | 59680 | 3343 | 4445 | 3017 |
| 1500 | 216633 | 5921 | 8137 | 4837 |
| 2000 | 640904 | 4858 | 9987 | 6521 |
| 3000 | 3381180 | 6044 | 16839 | 10889 |
| 5000 |  | 15626 | 37182 | 21940 |
| 7000 |  | 16132 | 39087 | 36851 |
| 8000 |  | 21231 | 43306 | 39017 |
| 10000 |  | 19724 | 68032 | 57162 |
| 20000 |  | 54981 | 133665 | 181553 |
| 50000 |  |  | 476316 | 1177115 |

Chart, line chart

Description automatically generated

Observations : The results for Gossip implementation using second approach indicate that 3D and imperfect 3D had almost the same performance. Line topology showed significantly lower running time which is better than rest of the topologies. Full topology showed good performance for less no of nodes but eventually stopped converging due to large no of nodes which might be possible due to the large no of neighbors the rumor has to spread to as was the case in Gossip implementation.