**Convergence Analysis of Gossip and Push-Sum Algorithms in Various Network Topologies**

1. **Introduction**

The project implements Gossip and Push-Sum algorithms using the Pony programming language. These algorithms are used for information dissemination and distributed sum computation in various network topologies.

1. **Methodology**

Implemented topologies: Full Network, 3D Grid, Line, and Imperfect 3D Grid

Algorithms: Gossip and Push-Sum

The program allows specifying the number of nodes, topology, and algorithm via command-line arguments.

1. **Implementation Details**

**3.1 Gossip Algorithm**

Each actor starts with heard count of 0. Upon receiving a rumor, the heard count increases. An actor is considered converged when the node has listened to the rumor 10 times. Actors continue to spread rumors at regular intervals.

**3.2 Push-Sum Algorithm**

Each actor maintains two values: s (initially set to the actor's ID) and w (initially 1). Convergence is determined when the ratio s/w remains unchanged (within 10^-10) for 3 consecutive rounds. After convergence, actors continue participating for a short cooldown period.

**3.3 Network Topologies**

1. Full Network: Every actor is connected to every other actor
2. 3D Grid: Actors arranged in a three-dimensional grid
3. Line: Actors arranged linearly with two neighbors each (except endpoints)
4. Imperfect 3D Grid: 3D grid with an additional random connection for each actor
5. **Results and Analysis**

**4.1 Gossip Algorithm**

**Topology: Full**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 3000 | 17378 | 237 |
| 3500 | 26285 | 212 |
| 4000 | 30900 | 187 |
| 4500 | 36525 | 192 |
| 5000 | 44011 | 173 |
| 10000 | 61436 | 160 |
| 12000 | 69784 | 155 |
| 13000 | 97037 | 132 |

**Topology: Line**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 1000 | 2493 | 101 |
| 2000 | 2783 | 189 |
| 3000 | 3146 | 222 |
| 4000 | 3142 | 289 |
| 5500 | 5000 | 173 |
| 6000 | 5258 | 178 |
| 5000 | 3780 | 257 |

**Topology: 3D**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 3000 | 8668 | 150 |
| 5000 | 11955 | 168 |
| 6000 | 11058 | 171 |
| 8000 | 10124 | 230 |
| 10000 | 14813 | 227 |
| 12000 | 14789 | 231 |
| 14000 | 15516 | 285 |
| 16000 | 14133 | 304 |
| 25000 | 20339 | 350 |
| 50000 | 49644 | 384 |
| 75000 | 69621 | 374 |
| 100000 | 110297 | 395 |
| 200000 | 1040167 | 368 |
| 300000 | 3403735 | 276 |

**Topology: imp3D**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 50000 | 33063 | 388 |
| 100000 | 70485 | 401 |
| 150000 | 311478 | 254 |
| 200000 | 445340 | 278 |

The network size, which is at the bottom of the table is approximately the largest.

A graph with different colored lines

Description automatically generated

* 1. **Push-Sum Algorithm**

**Topology: Full**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 1000 | 54 | 373 |
| 5000 | 1225 | 409 |
| 10000 | 6039 | 324 |
| 15000 | 38952 | 164 |
| 15500 | 26627 | 177 |

**Topology: Line**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 100 | 2033 | 782 |
| 500 | 222562 | 778 |
| 600 | 345746 | 527 |

**Topology: 3D**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 1000 | 677 | 397 |
| 10000 | 43957 | 657 |
| 25000 | 225084 | 760 |
| 50000 | 683042 | 569 |

**Topology: imp3D**

|  |  |  |
| --- | --- | --- |
| No of Nodes | Time Taken (ms) | CPU % |
| 1000 | 125 | 358 |
| 5000 | 894 | 389 |
| 10000 | 2573 | 383 |
| 25000 | 5993 | 384 |
| 50000 | 13035 | 769 |
| 100000 | 27929 | 386 |
| 200000 | 52271 | 384 |

The network size, which is at the bottom of the table is approximately the largest.

A graph of different colored lines

Description automatically generated

**4.3 Topology Comparison**

1. Full Network: Fastest convergence expected due to high connectivity
2. 3D Grid: Moderate convergence speed, balanced connectivity
3. Line: Slowest convergence due to limited information flow
4. Imperfect 3D Grid: Improved convergence over regular 3D Grid due to additional random connections
5. **Interesting Findings**

The implementation uses a timeout mechanism (1000 seconds) to handle cases without convergence. Both algorithms continue to operate after individual actor convergence, which helps in achieving global convergence. The Imperfect 3D Grid showcases how a small change in topology (adding one random connection per actor) can potentially improve convergence significantly.

1. **Conclusion**

The implementation provides a flexible framework for studying Gossip and Push-Sum algorithms across various network topologies. The actor-based model in Pony allows for efficient parallel execution.

1. **Team Members**

Sashank Boppana, 4171-9973

Tejesh Boppana, 1234-0626