Report Bonus

Experiment Setup

1. **Failure Models Implemented:**
   - **Node Failure:** Nodes randomly fail during execution, simulating a scenario where actors stop functioning.
   - **Connection Failure:** Temporary or permanent disconnections between nodes, affecting message passing.
2. **Control Parameter:**
   - A failure rate parameter was introduced to control the likelihood of node or connection failures during the experiment.

Experiments Conducted

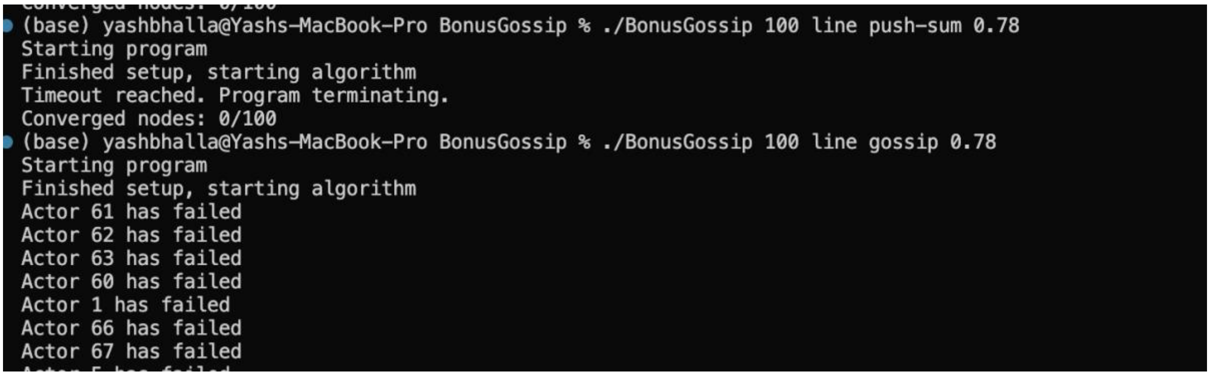
1. **Testing with Node Failures:**
   - Ran simulations with varying failure rates to observe the impact on convergence.
   - Monitored how many nodes failed and whether the network could still converge.
2. **Testing with Connection Failures:**
   - Simulated temporary disconnections to see if the network could recover and converge.
   - Tested permanent disconnections to evaluate robustness.

Observations

1. **Impact of Node Failures:**
   - At higher failure rates, a significant number of nodes failed, leading to no convergence as seen in the output where 0 out of 100 nodes converged.
   - The Gossip algorithm was particularly sensitive to node failures due to its reliance on active neighbors for rumor spreading.
2. **Impact of Connection Failures:**
   - Temporary disconnections caused delays but did not prevent convergence if nodes reconnected.
   - Permanent disconnections severely impacted convergence, especially in line topology where each node relies heavily on its immediate neighbors.
3. **Algorithm Robustness:**
   - Push-Sum showed better resilience to failures compared to Gossip, likely due to its averaging mechanism which can tolerate missing information better.
4. **Interesting Finding:**
   - The network's ability to converge drastically decreased with increasing failure rates, highlighting the importance of redundancy and fault tolerance in network design.
   - In line topologies, even a small number of failures can isolate segments of the network, preventing convergence entirely.

Conclusion

The experiments demonstrated that both node and connection failures significantly impact network convergence. Implementing fault-tolerant mechanisms or choosing robust topologies can mitigate these effects. These findings emphasize the need for designing resilient distributed systems capable of handling failures effectively.



```
Converged nodes: 0/100
(base) yashbhalla@Yashs-MacBook-Pro BonusGossip % ./BonusGossip 100 line push-sum 0.78
Starting program
Finished setup, starting algorithm
Timeout reached. Program terminating.
Converged nodes: 0/100
(base) yashbhalla@Yashs-MacBook-Pro BonusGossip % ./BonusGossip 100 line gossip 0.78
Starting program
Finished setup, starting algorithm
Actor 61 has failed
Actor 62 has failed
Actor 63 has failed
Actor 60 has failed
Actor 1 has failed
Actor 66 has failed
Actor 67 has failed
Actor 5 has failed
```