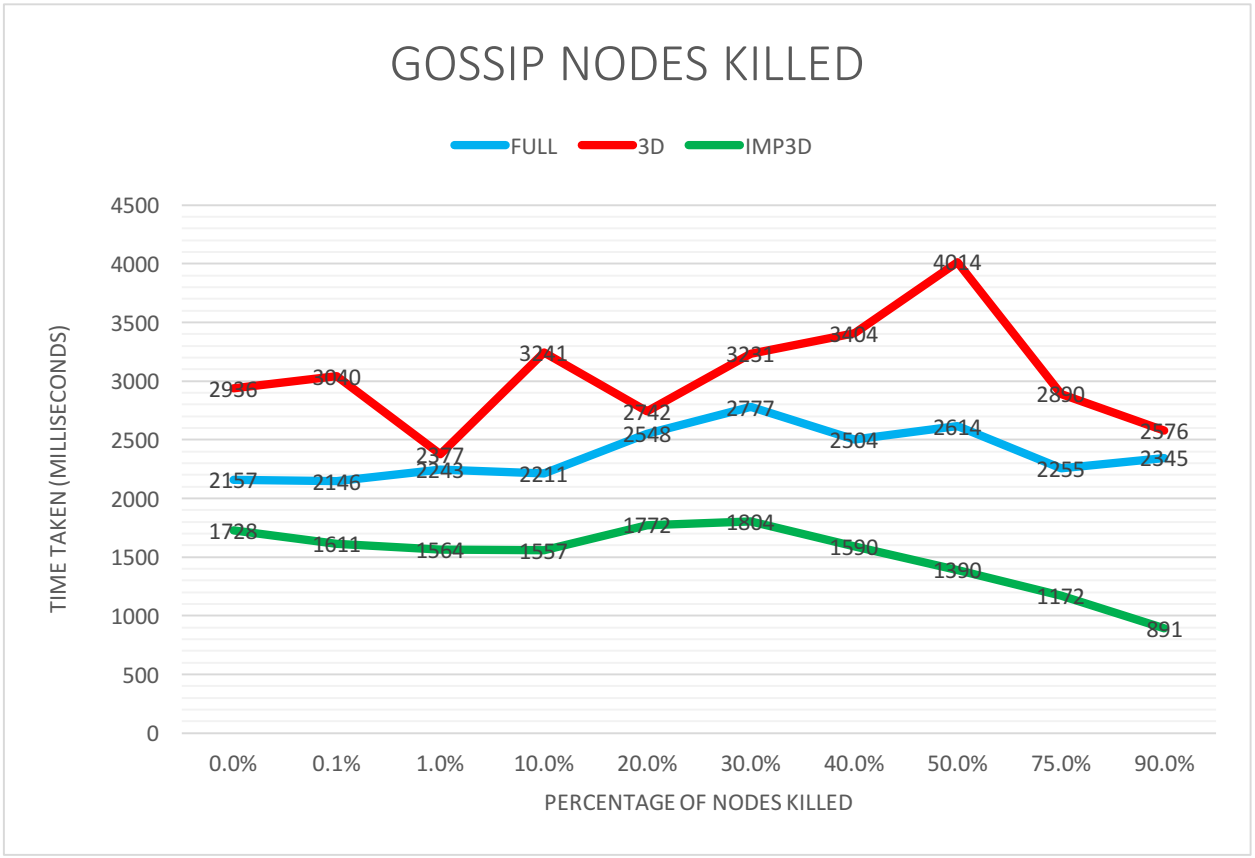
Project 2 Bonus Report

Robustness of Gossip

I implemented a failure model in which I randomly deleted a certain number of nodes (taken as input from user) and saw how much time the remaining number of nodes took to converge. Convergence in this case was achieved when more than 90% of the remaining nodes received the gossip message at least once. The network had 1000 nodes initially.

- I first started my testing on the line topology and the results were disastrous. After removing even one node from the topology the network fails to reach convergence. I tested line by force stopping after a certain interval of time and checking the percentage of nodes that converged.
- After removing one node from the network in line, it breaks into 2 different networks.
- E.g.: Let us assume there are 100 nodes in the network and the randomly chosen node to start the gossip is 3. The randomly chosen node to kill is node 50. In such a scenario all the nodes from 1 to 49 will converge and the rest will not receive any gossip message.
- If one node can cause such havoc, I could only imagine what killing 10-20% of the nodes would do.



RAW DATA

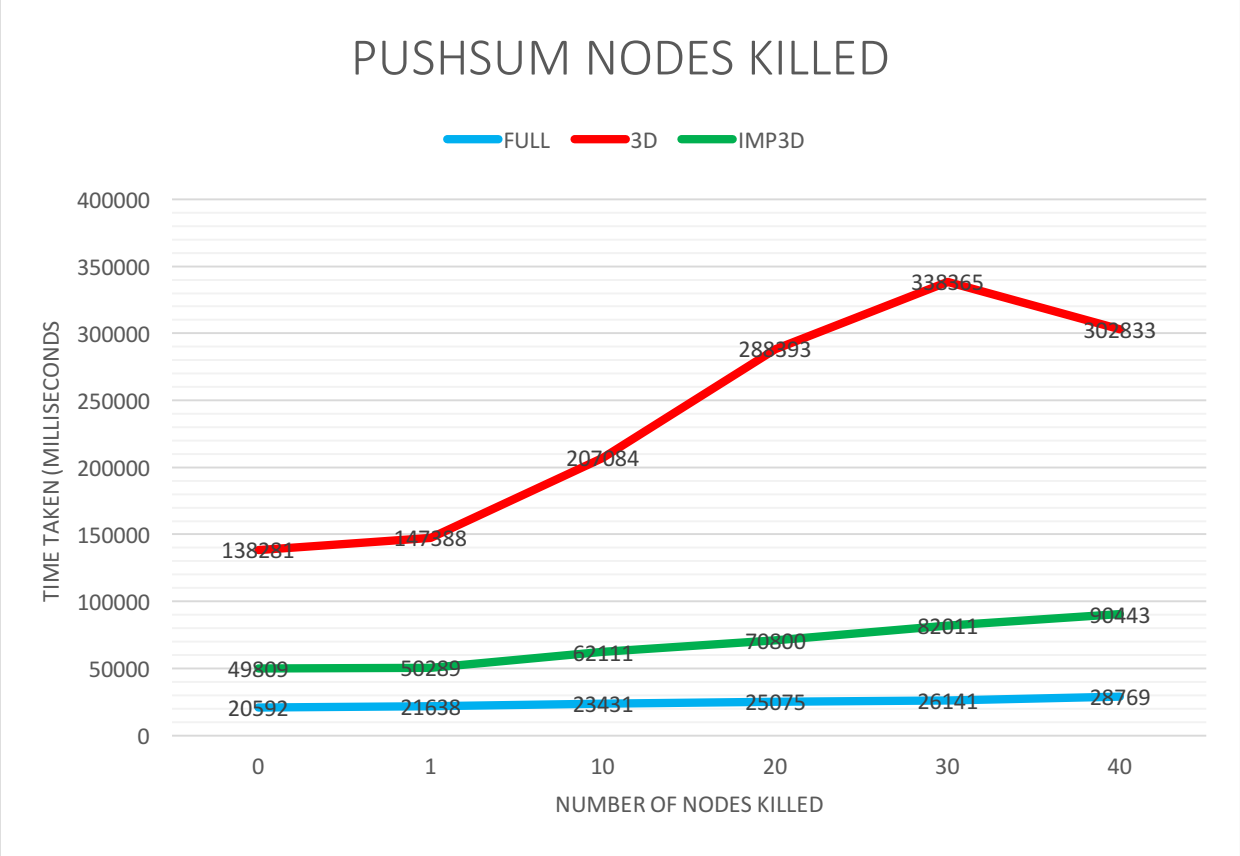
) OF NODES KILLED	FULL(in ms)	3D(in ms)	IMP3D(in ms)	
0.0%	2157	2936	1728	
0.1%	2146	3040	1611	
1.0%	2243	2377	1564	
10.0%	2211	3241	1557	
20.0%	2548	2742	1772	
30.0%	2777	3231	1804	
40.0%	2504	3404	1590	
50.0%	2614	4014	1390	
75.0%	2255	2890	1172	
90.0%	2345	2576	891	

- The full,3d and imperfect 3d topologies handled node failures extremely well and almost always converged no mater how many nodes I killed.
- Imperfect 3d was the fastest at converging, followed by full and then 3d.
- The shocking part was that even after deleting 90% of the nodes in the network the gossip message was received by more than 90% of the remaining nodes in each of these 3 networks. Moreover, the convergence occurred extremely quickly as shown in the graph.

Robustness of Pushsum

I implemented a failure model in which I randomly deleted a certain number of nodes (taken as input from user) and saw how much time the remaining number of nodes took to converge. Convergence in this case was achieved when more than 50% of the remaining nodes achieved the pushsum criteria. The network had 125 nodes initially.

- The same disaster occurs when we try killing nodes for line with pushsum algorithm. When one node is killed the network does not converge.
- The more the number of neighbors => denser the network => network is more fault tolerant
- Full topology is the most fault tolerant followed by imperfect 3d and then 3d. This can be easily seen in the data and line graph.
- Full did not converge when 100 nodes were killed. 3D did not converge for 50 nodes and imperfect 3d did not converge for 75 nodes.



RAW DATA

NUM. OF NODES KILLED	FULL(in ms)	3D(in ms)	IMP3D(in ms)
0	20592	138281	49809
1	21638	147388	50289
10	23431	207084	62111
20	25075	288393	70800
30	26141	338365	82011
40	28769	302833	90443
50	31329	NOT CONV	91100
75	46026	NOT CONV	NOT CONV
100	NO CONV	NOT CONV	NOT CONV

All entries in table are time taken to converge(in milliseconds)

NOT CONV: No convergence occurred.