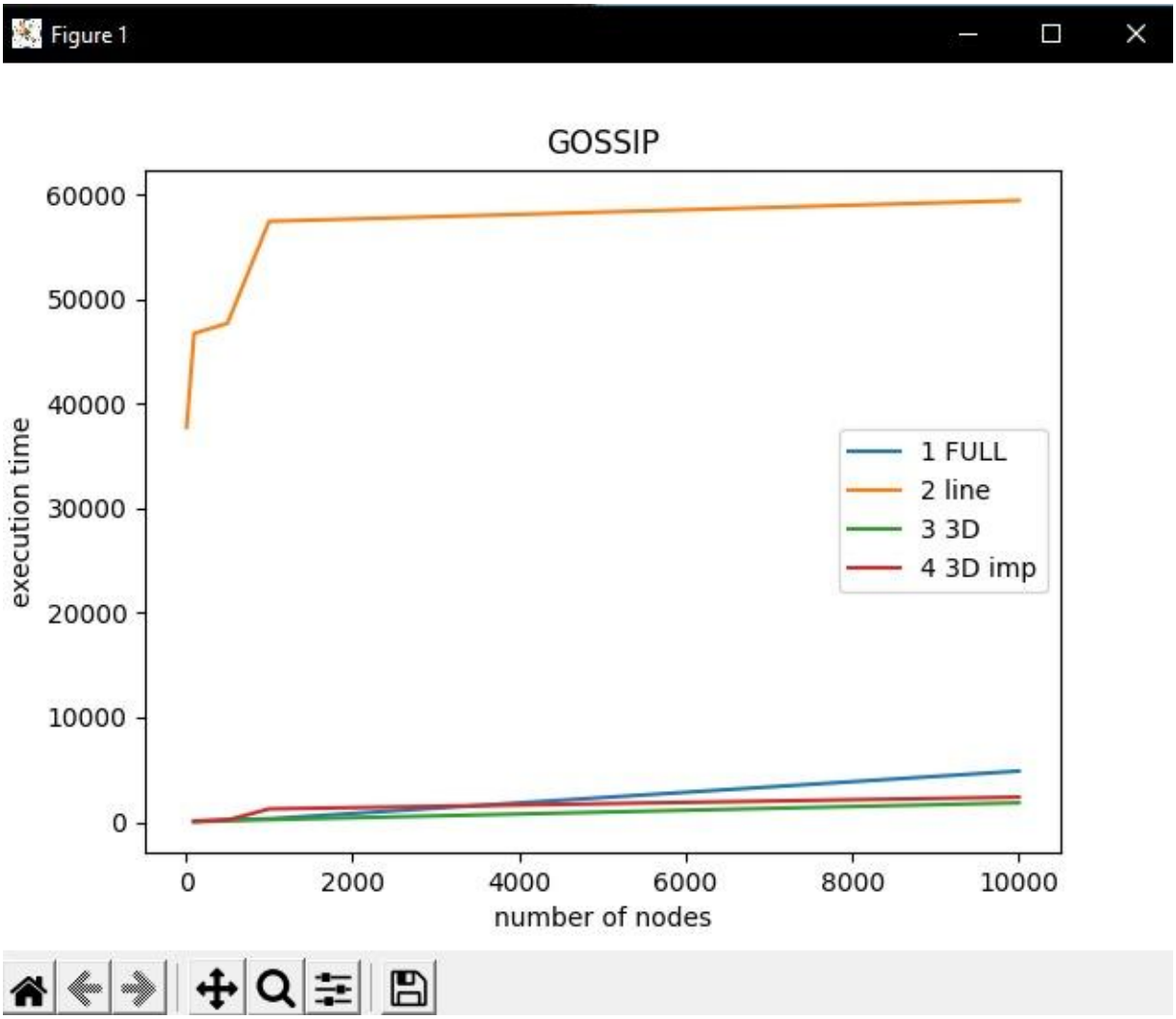
Report - Assignment 2

Gossip Algorithm

The gossip algorithm's speed relies on being able to send the message or "rumor" to as many nodes as possible in order for one of them to hear the "rumor" ten or more times. The topologies that have the fewest restrictions will therefore yield faster times. As you can see in the results on the right and in the graph below, the times are much faster when using the full topology versus any other topology due to all nodes being considered a neighbor.

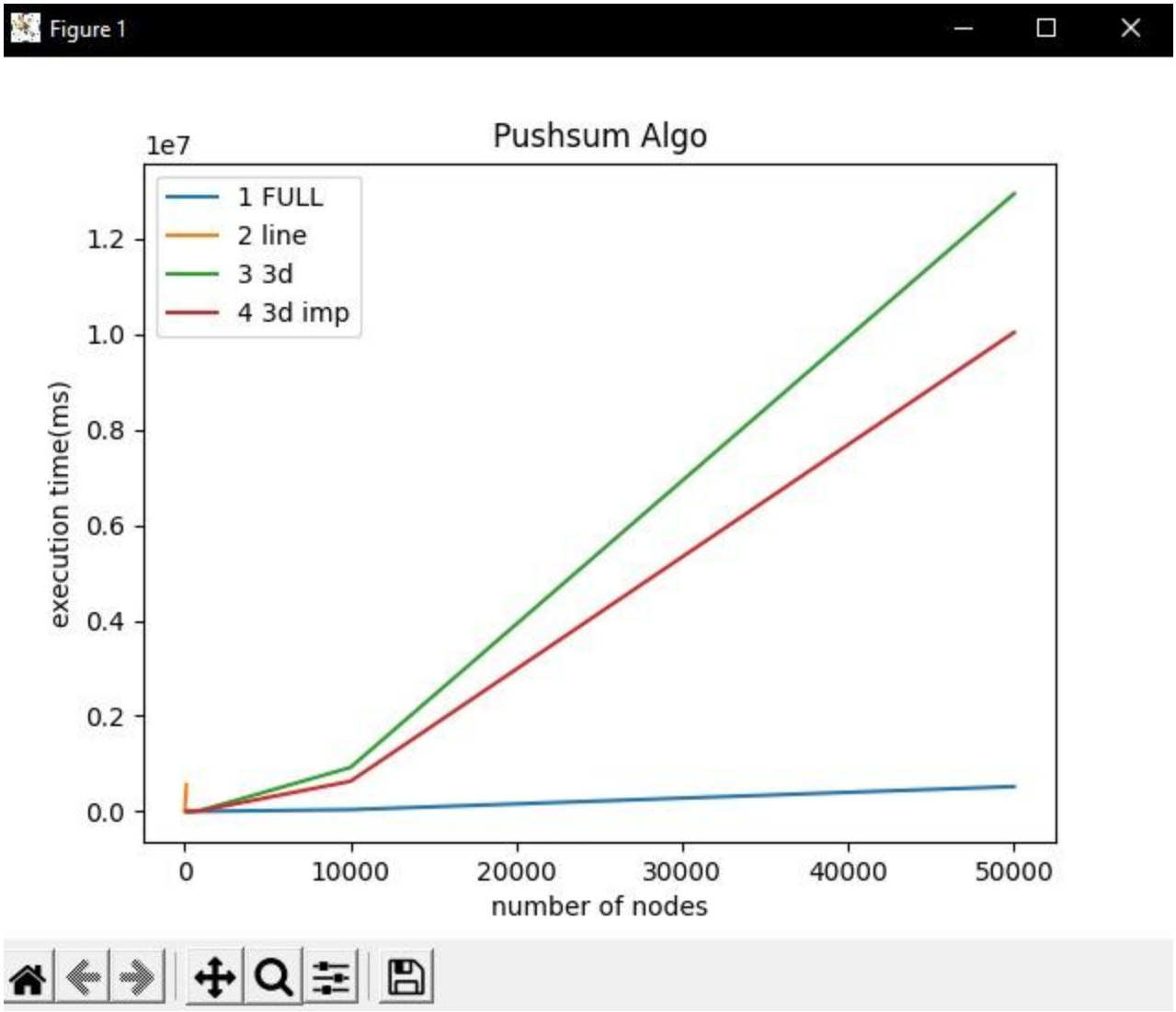
Another important find was that imperfect 3D performed faster than 3D, which makes sense with what we know from how full topology performs the fastest. Since imperfect 3D sends the message to a random node along with the usual neighbor from the 3D topology, the rumor can spread to a distant node that otherwise would not have been reached quickly when limited to only directly connected nodes. Once a rumor is spread to a different region, the distribution of the rumor becomes more even throughout all the nodes. This also prevents the chance for regional convergence where the rumor algorithm ends too early due to the rumor being spread repeatedly amongst a small group of neighboring nodes.

		gossip	
	cpu (ms)	real (ms)	nodes
full	46	36.08	100
	234	107.714	500
	312	166.735	1000
	4875	2336.6337	10000
	82859	56878.237	50000
	319062	264615.095	100000
line	37718	37707.4	10
	46703	46531.094	100
	47671	47337.926	500
	57421	57353.984	1000
	59406	59412.945	10000
3D	31	46.453	100
	125	71.85	500
	234	110.06	1000
	1843	679.757	10000
	7484	2596.174	50000
	14421	5134.256	100000
	200437	91008.287	1000000
3D imp	110	46.453	100
	186	86.004	500
	1255	500.1	1000
	2387	1004.95	10000
	12956	5456.66	50000
	16723	7345.222	100000



Gossip - Algorithm

		pushSum	
	cpu (ms)	real (ms)	nodes
full	171	73.44	100
	921	279.33	500
	1437	457.994	1000
	36859	16061.81	10000
	521547	241352.365	50000
line	9437	4333.22	10
	562254	34517.49	100
3D	406	178.262	100
	3328	1195.82	500
	24109	8596.06	1000
	922109	371156.31	10000
	12938529	5625447.8	50000
3D imp	334	153.45	100
	2826	924.68	500
	19245	1065.143	1000
	632419	32541.25	10000



The first interesting find for the push sum algorithm was noticing that it took a bit longer overall than the gossip algorithm. I believe this is because a node only needs to be called 10 times before it ends, while a node in push sum may never converge depending on how many nodes there are and the topology used.

Right off that bat, we realized that the line topology takes much longer than any other topology similar to the gossip algorithm, but with even more of a difference since push sum takes longer overall. This is because nodes in the line topology have the least likely chance of receiving a message since they have the fewest neighbors, especially on the two ends of the line which can only receive a message from one neighbor. For the full topology however, the time was much faster, as in the case for the gossip algorithm. The full topology allows messages to be spread the fastest since all nodes may be contacted by any other node. This leads to a faster convergence for push sum, which relies on sending and receiving sum/weight repeatedly in order for the difference in sum to weight ratio (between each iteration) to approach zero.

We noticed that line topology started doing better once adding a scheduler to our algorithm that sent continual messages to all available neighbors. However we think this is due to regionalized convergence and that it is not more efficient than 3D topologies. This is due to neighbors repeatedly calling one another. We could tell

this by changing the count of how many times the difference must be less than 10^-10. Line topology will take much longer to converge if this number is incre. The full topology however can scale well when adding nodes since the distribution of pushing sum/weight to each node stays somewhat consistent across all nodes. Another important find was that imperfect 3D performed significantly better than 3D without using a random message, which again, is similar to the results from the gossip algorithm. The logic works similarly in the case of push sum which also benefits from more messages being sent between nodes.