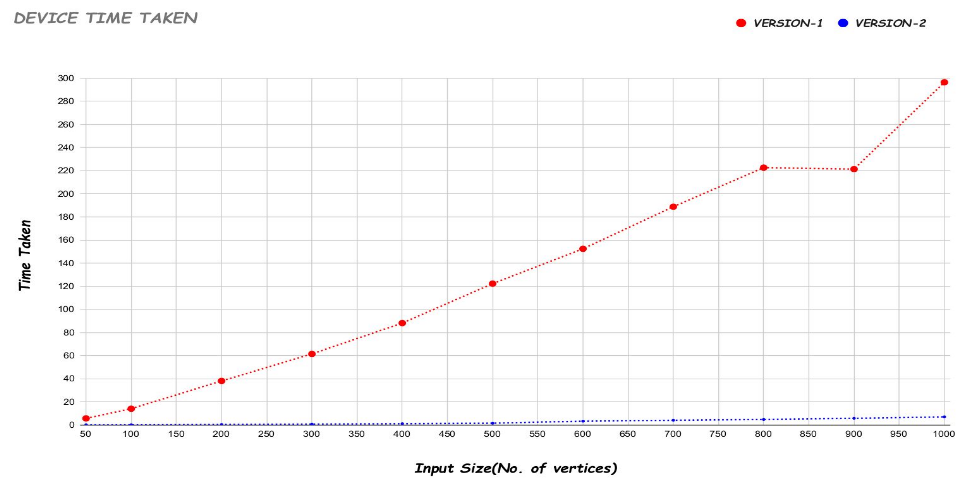
Betweenness Centrality of weighted-graphs

SHREYAS JAIN
CS20M063

Betweenness centrality for a node v is defined in terms of the proportion of shortest paths that go through v . Naive implementation of BC calculation runs in $O(n^3)$ time(using Floyd Warshall algo) then Brandes proposed an algo which runs in O(nm) and $O(nm + n^2 \log n)$ in time for unweighted and weighted networks, respectively, where n is the number of vertices and m is the number of edges . It is easier to implement the unweighted version because it involves parallelizing the BFS algorithm but for the weighted version , it’s difficult to parallelize the Dijkstra’ algo because In the sequential Dijkstra algorithm, the fact that one frontier node is selected in each iteration makes parallelization a difficult task . However, this restriction can be relaxed, which means that several nodes can be settled at once to form the frontier set, allowing them to be inspected simultaneously . Following these two research papers , link & link for BC calculation of weighted graphs .

Results : Since the graph traversal and shortest path accumulations of each root are independent, we can process all roots in parallel. We take advantage of this coarse-grained parallelism by assigning roots to each Streaming Multiprocessor (SM) of the GPU. Additionally, we can leverage fine-grained parallelism by having threads process the shortest path calculations and dependency accumulations cooperatively .

Version-1 implements only fine-grained parallelism and Version-2 implements both coarse-grained as well as fine-grained parallelism . Time comparisons between the two version can be seen in the graph.



Challenges Faced and how it is resolved :

1) Understand the working of Brandes Algorithm , So I first coded the serial version for the unweighted graph just to get the essence of how exactly things are done.
2) Then extending to idea to the weighted Graph , read 2 research papers and first implemented the serial version of the algorithm in use .

Overall , experience from Assignments 3 and 4 was very helpful in dealing with CSR representation of graph and also in implementation of two different kinds of parallelism .