Ankit Goyal
ankitgoyal@utexas.edu

Software Multicore Processors
Homework 2

In this report different algorithms for Single Source Shortest Path algorithm have been implemented and analyzed for performance. All the programs were ran on the same hardware with 16 cores.

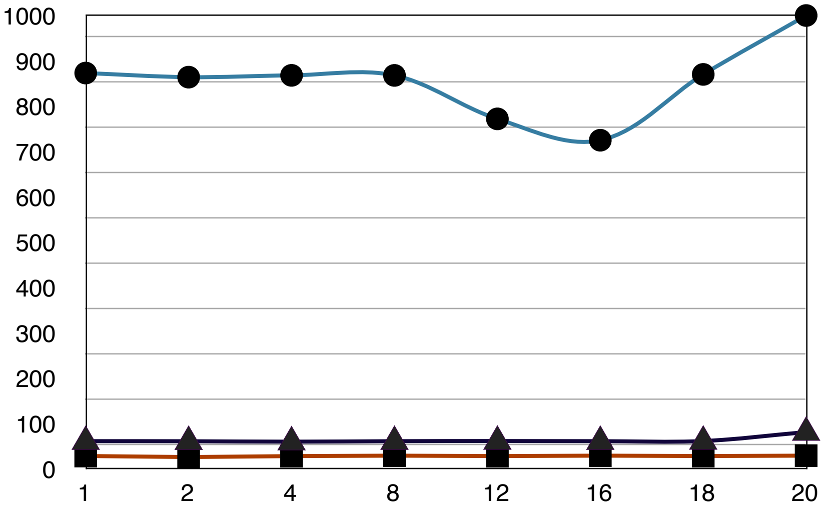
Bellman Ford Algorithm

In this algorithm we iteratively go through the edges and relax each edge until no more edges can be relaxed. Data structure used to implement sequential bellman ford are:
A vector is used to store the edges was made. Edges are a struct type and in the algorithm edges were iterated.

The algorithm ends when no edges are relaxed. Following table shows different values for sequential and parallel Bellman Ford Algorithm.

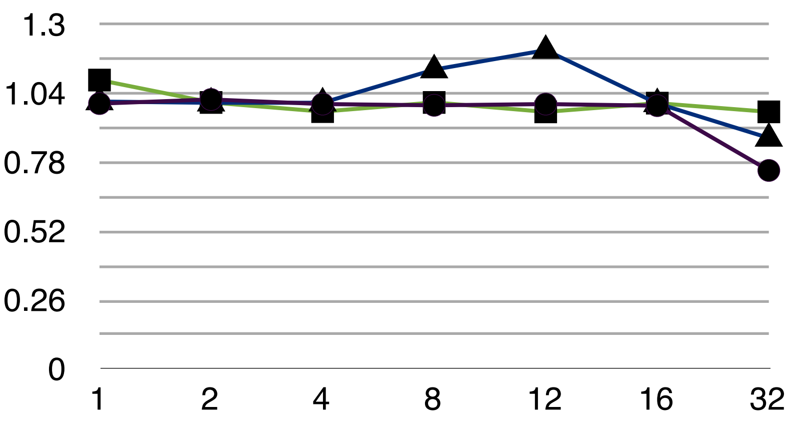
n	***Seq.***	*1*	2	4	8	12	16	32
$T_n\,(nodes = 33554432\,)$	60.3007	60.1312	59.2142	60.235	60.523	60.2424	60.604	80.352
$T_n\,(nodes = 23947347\,)$	872.316	863.252	867.232	867.124	771.24	724.142	104.93	999.23
$T_n\,(nodes = 16777210\,)$	27.548	25.2352	27.3535	28.325	27.353	28.3544	27.426	28.352

To implement parallel bellman ford similar data structure was used. However a conflict free schedule was created which was then iterated upon.

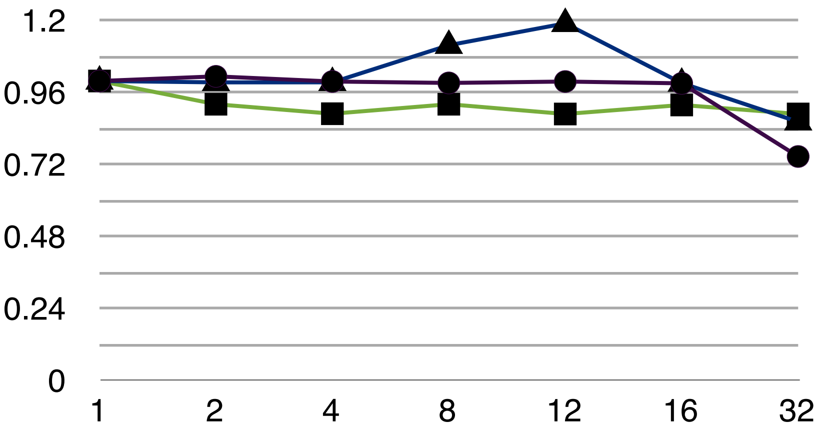


Runtime vs Number of threads

Since Openmp was used to parallelize the overhead of the threads exceeded the actual throughput and hence no gain in performance is observed.



Speedup Plot



Scalability Plot

Chaotic Relaxation

To implement the work-set, unordered set in c++ was used. Since the nodes are removed at random, we always choose the first element which makes the read of constant order.
Following results were obtained for the different graphs.

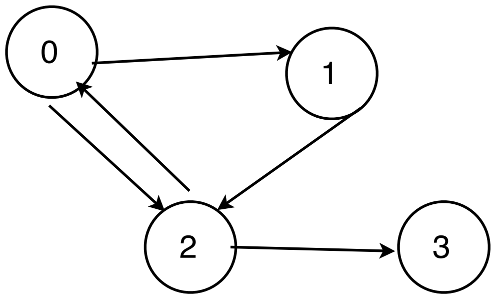
Number of Nodes	Time taken
23947346	313.242
33554432	40.2424
16777216	45.2525

To implement the parallel chaotic relaxation delta stepping technique was used. To create the work-set an unordered set was used. Algorithm works without pthreads but I was unable to parallelize it.

Ankit Goyal
ankitgoyal@utexas.edu

Dijkshtra Algorithm

For the Dijkshtra algorithm priority queue provided by C++ Library was used. Graph was stored in a vector as shown below.



graph[0]:	2	1

Wights were also stored in a vector in a similar manner.

In the actual run Dijkshtra's algotrithm outperformed both bellman ford and chaotic relaxation, due to the amount of work done.

Number of Nodes	Time taken
23947346	5.83196
33554432	28.1426
16777216	12.9161