Travelling Salesman Problem

—

Siddharth Tarey
Pavan Bhat

Agenda

Summary for TSP problem

Our Approach - Branch & Bound

Design and Operation of B&B in TSP

- Sequential
- Parallel

Strong Scaling

Weak Scaling

Future Work

Learning

What is TSP?

➥ Given a complete weighted undirected graph G = (V {1,...n}, E) and a cost matrix C, a tour is a circle in G which visits each vertex exactly once.

Image References:
http://makeagif.com/i/4-ew7H

Our Approach

Branch and Bound Algorithm

- **Select:** A node is selected based on a search criterion
- **Branch:** The selected node is then **subdivided** into its child nodes
- **Bound:** Some of the nodes that are created are then **pruned**
- **Repeat the first 3 steps**

Design and Operation

Branch and Bound Algorithm

For each node in **queue**:

child → **getChildren(**node)

if(child is leaf):

-Trace path to check for TSP

-Save the **path** and **cost**

Else:

Update cost of child

Put the child on **queue** if cost is less than TSP cost

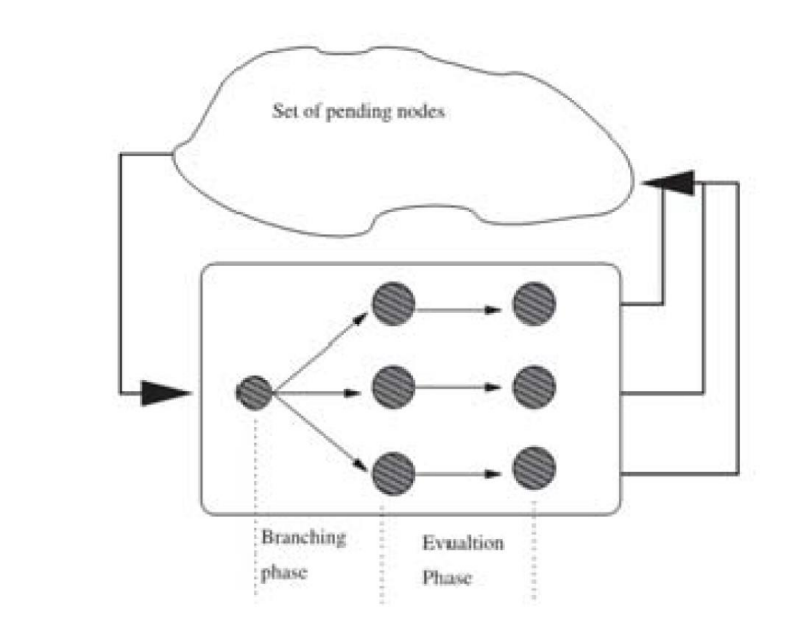
Parallelize the For loop using parallelFor from the pj2 library over a Work Queue

Source of Parallelism here:

- Parallel Work Queue applied using a Tree-based Strategy

Best-First Search Algorithm

Approach



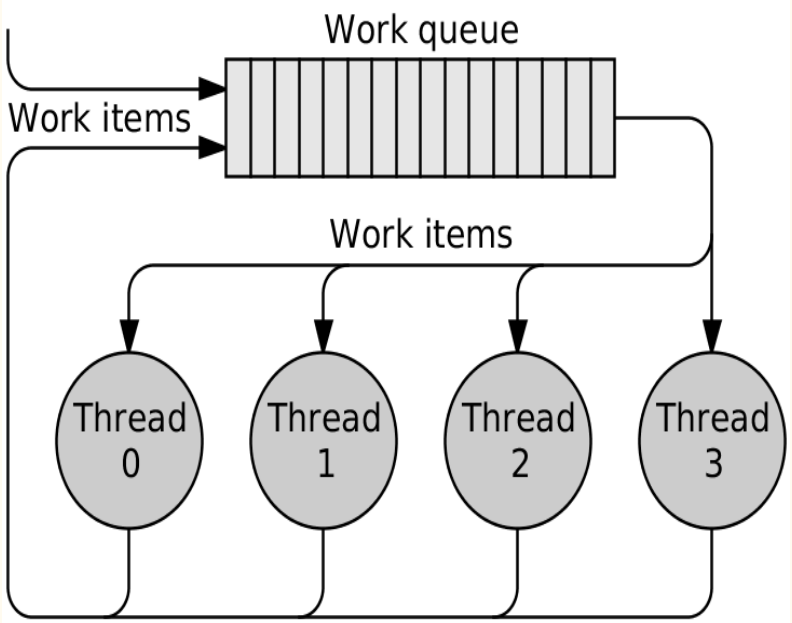
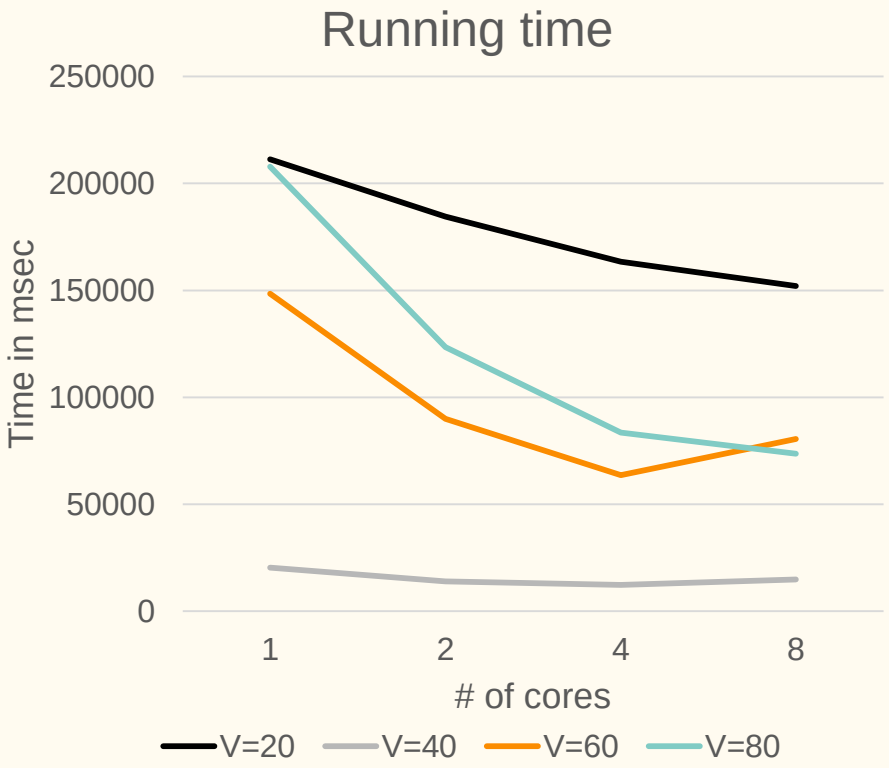
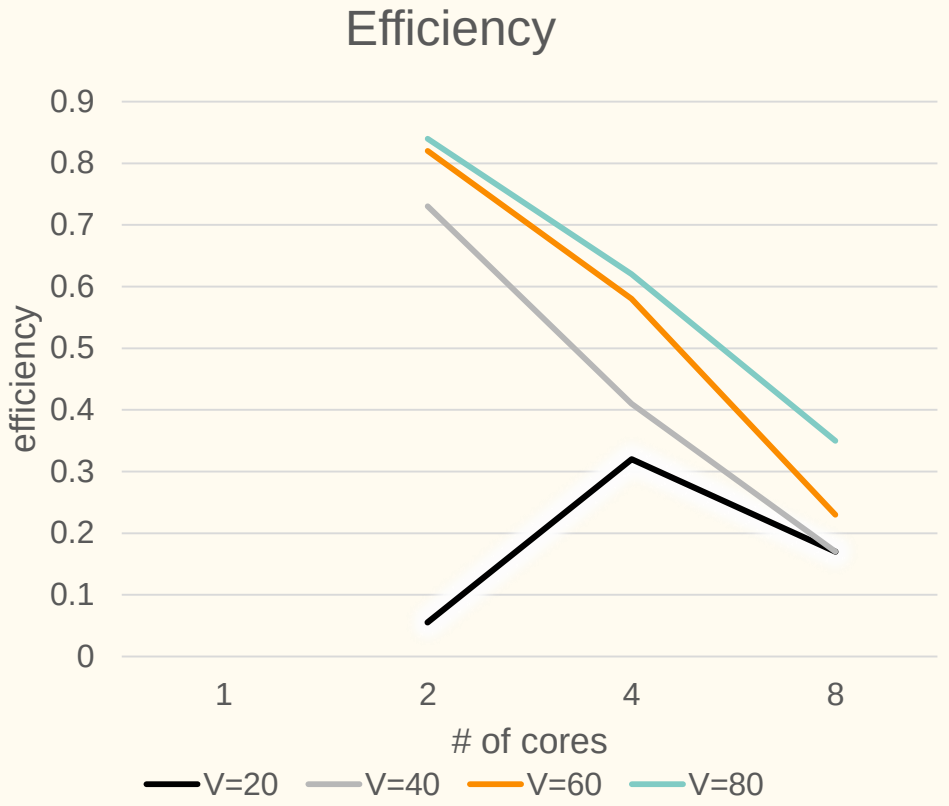


Image References:
References [1]
Referenced from text book: https://www.cs.rit.edu/~ark/bcbd/

Strong Scaling

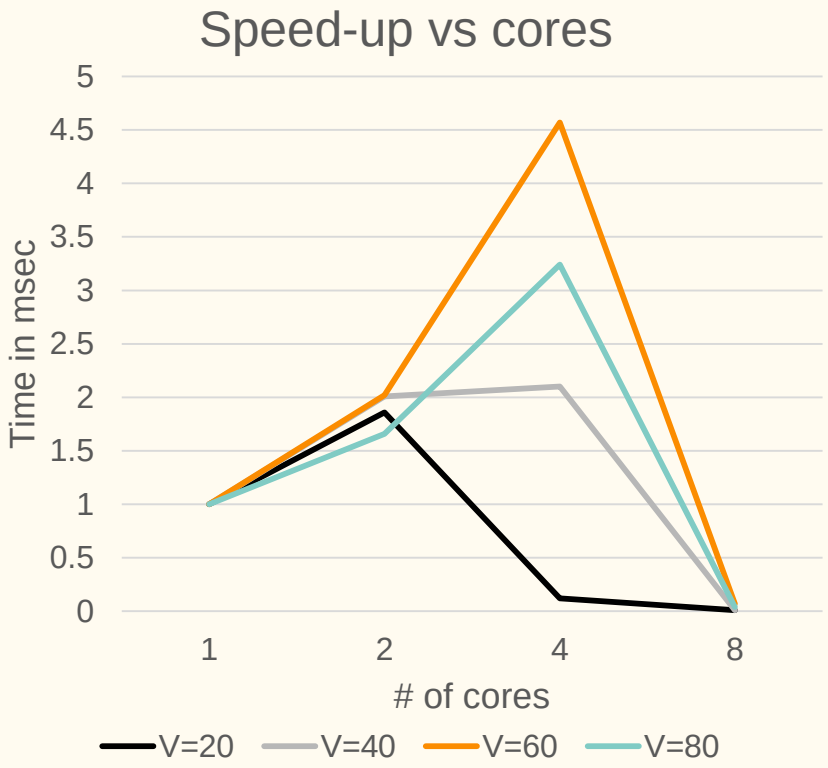
No of cities	No. of cores	Time (in msec)	Speed-up	Efficiency
20	1(seq)	211225		
	2	184467	0.11	0.055
	4	163392	1.29	0.32
	8	151996	1.39	0.17
40	1(seq)	20384		
	2	13928	1.46	0.73
	4	12331	1.65	0.41
	8	14869	1.37	0.17
60	1(seq)	148514		
	2	89919	1.65	0.82
	4	63605	2.33	0.58
	8	80575	1.84	0.23
80	1(seq)	207777		
	2	123531	1.68	0.84
	4	83590	2.48	0.62
	8	73590	2.80	0.35

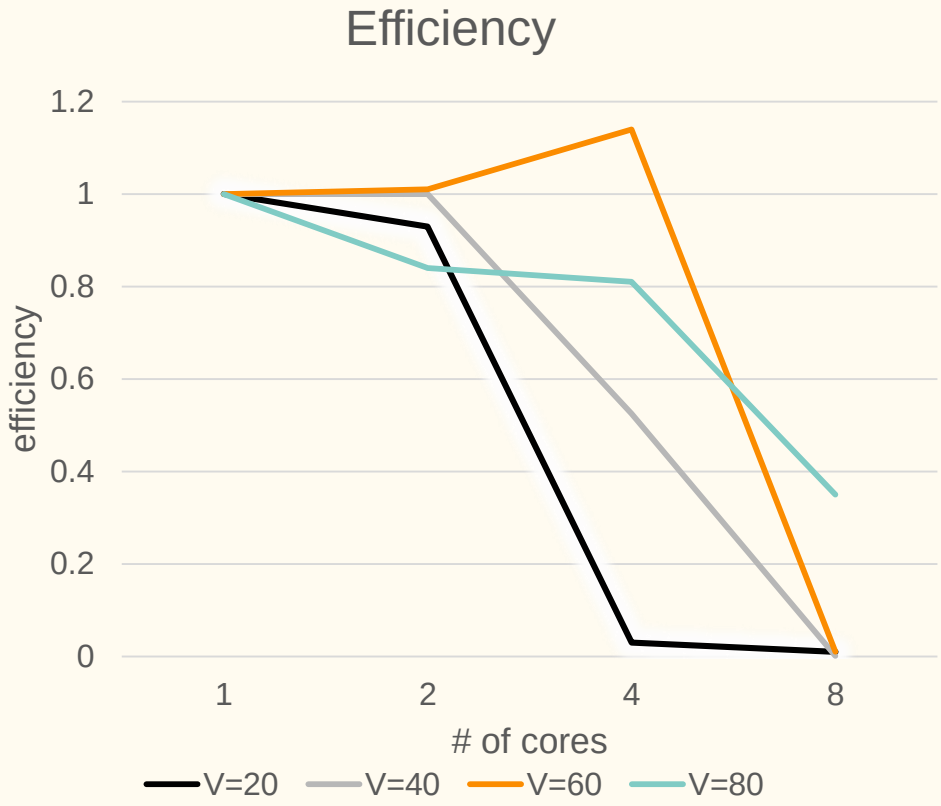

Running time
250000
200000
150000
100000
50000
0
Time in msec
1
2
4
8
# of cores
V=20
V=40
V=60
V=80


Efficiency
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
efficiency
1
2
4
8
# of cores
V=20
V=40
V=60
V=80

Weak Scaling

No of cities	No. of cores	Problem size	Time (in msec)	Speed-up	Efficiency
20	1(seq)	200	95		
	2	400	102	1.86	0.93
	4	800	3142	0.12	0.03
	8	1600	774996	0.01	0.01
40	1(seq)	400	110		
	2	800	109	2.01	1.00
	4	1600	209	2.10	0.525
	8	3200	900869	0.01	0.001
60	1(seq)	525	112		
	2	1050	110	2.02	1.01
	4	2100	98	4.57	1.14
	8	4200	12397	0.07	0.01
80	1(seq)	880	98		
	2	1760	118	1.66	0.84
	4	3520	119	3.24	0.81
	8	7040	17724	0.04	0.005


Speed-up vs cores
Time in msec
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
1
2
4
8
# of cores
V=20
V=40
V=60
V=80


Efficiency
efficiency
1.2
1
0.8
0.6
0.4
0.2
0
1
2
4
8
# of cores
V=20
V=40
V=60
V=80

Non Ideal Scaling

- Is caused because the bounding function is realized only once the exploration is completed

Future Work

Future Work

- Incorporate Hybridization schemes where the Branch and Bound Algorithm is used in conjunction with Local Search Algorithms such as Variable Neighborhood Search, Random Neighborhood Search etc. for pruning nodes and thereby avoiding unnecessary computations.

- Implement a GPU based approach in order to achieve a better speedup and to attain an expected strong and weak scaling.

Learning

What we Learnt?

- During exploration of different strategies for optimizing the computation of the TSP problem we came across several techniques and algorithms that would provide a speedup to the traditional implementation of the Branch and Bound Algorithm.
- While scaling the output we noticed behavior of different graphs which introduced anomalies with the increase in the number of cores.

References

- [1] T. Carneiro, A. E. Muritiba, M. Negreiros and G. A. Lima de Campos, "**A New Parallel Schema for Branch-and-Bound Algorithms Using GPGPU,**" Computer Architecture and High Performance Computing (SBAC-PAD), 2011 23rd International Symposium on, Vitoria, Espirito Santo, 2011, pp. 41-47.

- **“Solving the Traveling Salesman Problem with a Parallel Branch-and-Bound Algorithm on a 1024 Processor Network”** by S. Tschoke, M. Racke, R. Luling, B. Monien, Department of Mathematics and Computer Science, University of Paderborn, Germany.

- Ten-Hwang Lai and Sartaj Sahni. 1984. Anomalies in parallel branch-and-bound algorithms. *Commun. ACM* 27, 6 (June 1984), 594-602. DOI=http://dx.doi.org/10.1145/358080.358103

Questions?