PARALLEL A* ALGORITHM

CSE 633 Parallel Algorithm

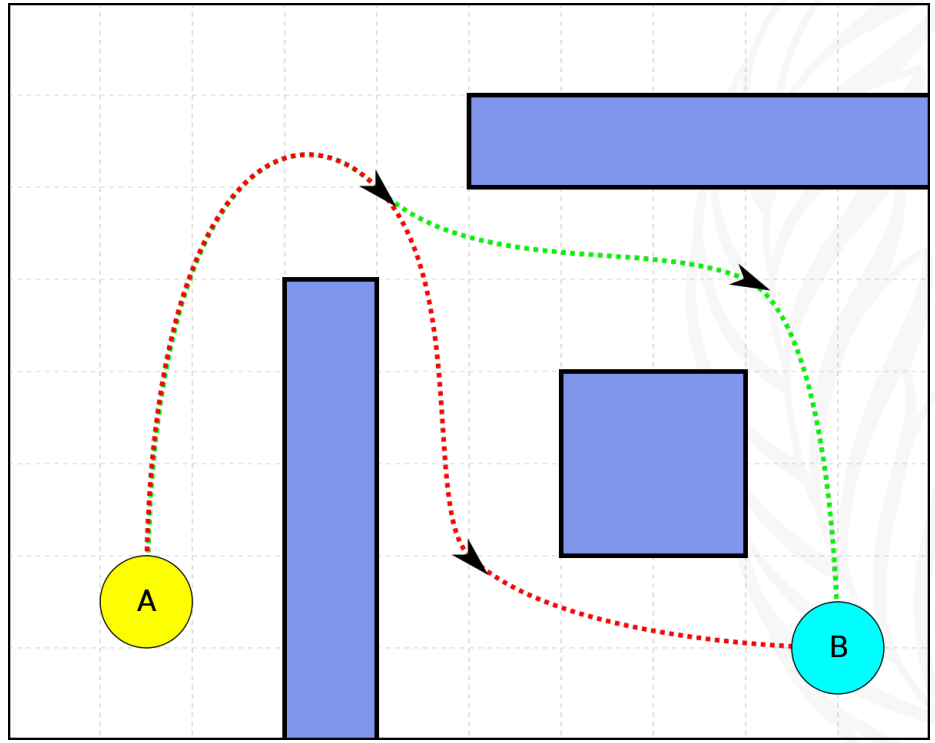
Weijin Zhu

Instructor: Dr. Russ Miller

University at Buffalo The State University of New York

Problem Statement

To find the shortest path between two points without run into the obstacles



What is A* Algorithm?

- A search algorithm used for path searching and path traversal
- It considers all adjacent cells and picks the cell with lowest cost
- It expands paths based on function f(n)
- It plots a walkable path between multiple points on the graph

How does A* Algorithm explores?

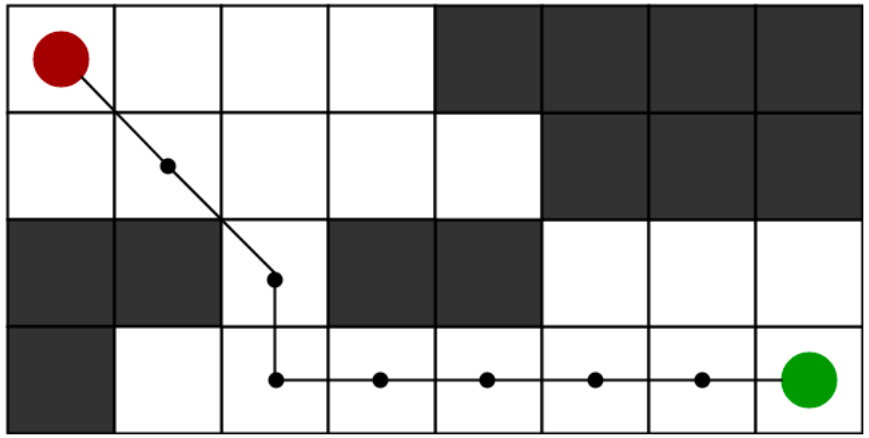
- Given a start node and a target node

- Each step picks next landing position according to f value

- f = g + h

  g: the cost to move from start to a given node

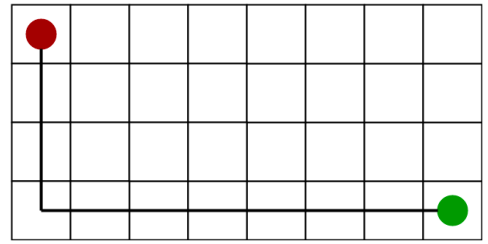
  h(heuristic): the cost from a given node to destination



Heuristic (h(n))

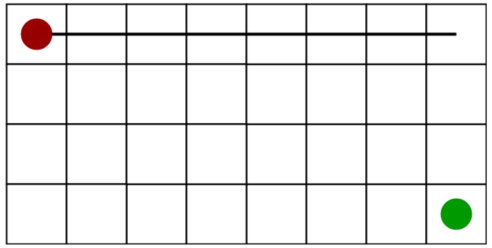
- Manhattan Distance

h(n) = abs (current_cell.x – goal.x) + abs (current_cell.y – goal.y)



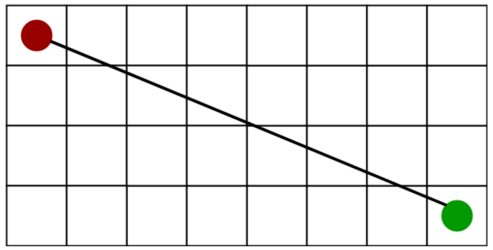
- Diagonal Distance

h(n) = max { abs(current_cell.x – goal.x), abs(current_cell.y – goal.y) }



- Euclidean Distance

h(n) = sqrt ((current_cell.x – goal.x)^2 + (current_cell.y – goal.y)^2)



Reference: https://www.geeksforgeeks.org/a-search-algorithm/

A* Algorithm Pseudocode

Initialize the open and closed list & put the starting node on the open list
While the open list is not empty
- a) find the node with the least f on the open list, call it 'q'
- b) pop q off the open list
- c) generate q's 8 successors and set their parents to q
- d) for each successor
  - i) if successor is the goal, stop search
    - successor.g = q.g + dist(successor, q)
    - successor.h = dist(goal, successor)
    - successor.f = successor.g + successor.h
  - ii) if a node with the same position as successor is in the Open list which has a lower f than successor, skip this successor
  - iii) if a node with the same position as successor is in the Closed list which has a lower f than successor, skip this successor & otherwise add the node to the open list
- end (for loop)
- e) push q on the closed list

end (while loop)

PARALLEL A* ALGORITHM

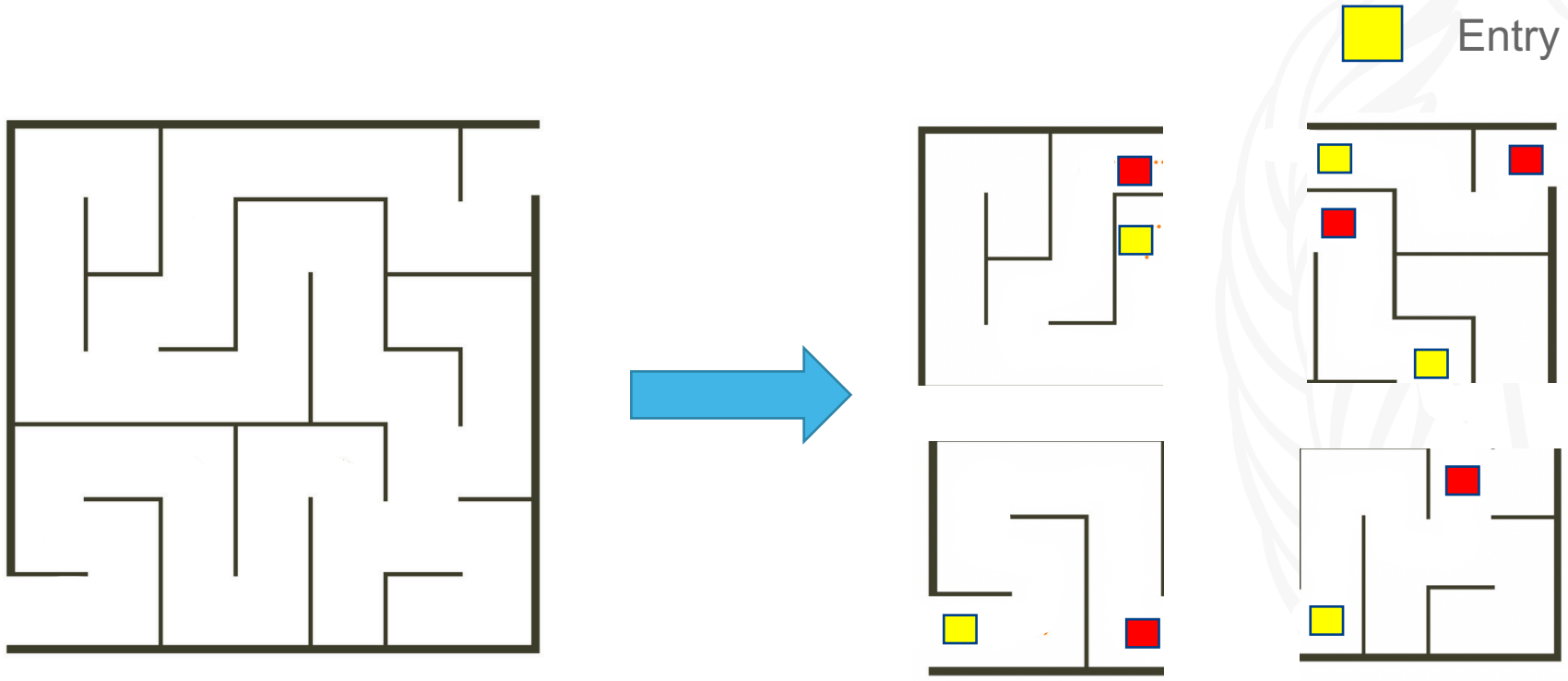
University at Buffalo The State University of New York

Parallel A* Algorithm

- Randomly Generate a graph of size n by n
- Split the graph into equal size subgraphs and each subgraph contains entry & exit points
- Distribute subgraphs to different processors
- Each subgraph constructs the path from its entry point to its exit point
- Each processor passes its path to the adjacent processor

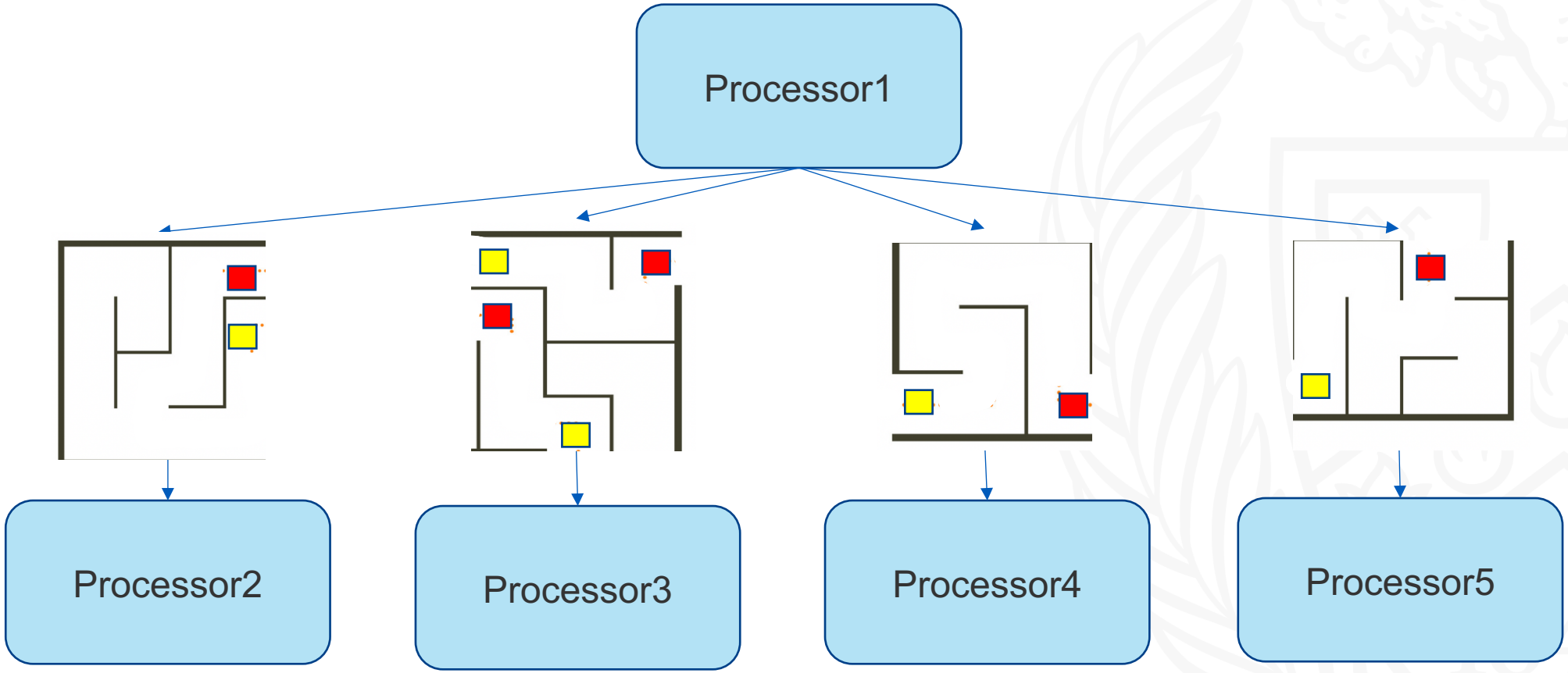
Parallel A* Algorithm

- First processor split the graph into equal size subgraphs and each subgraph contains entry & exit points

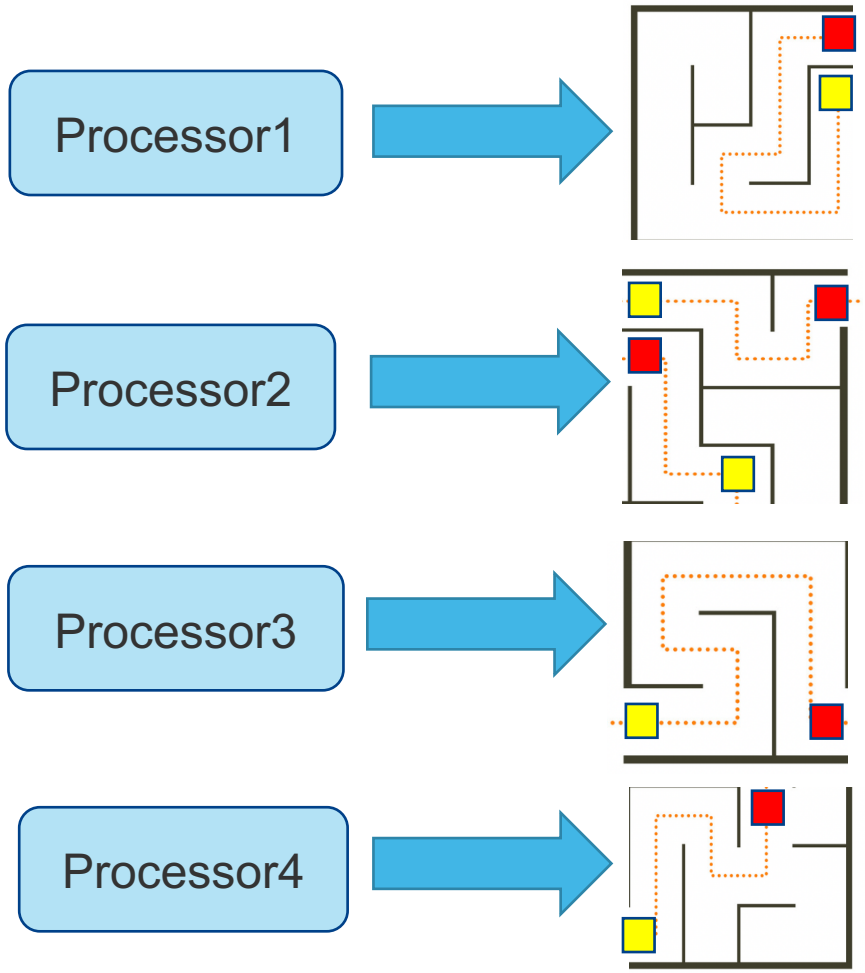


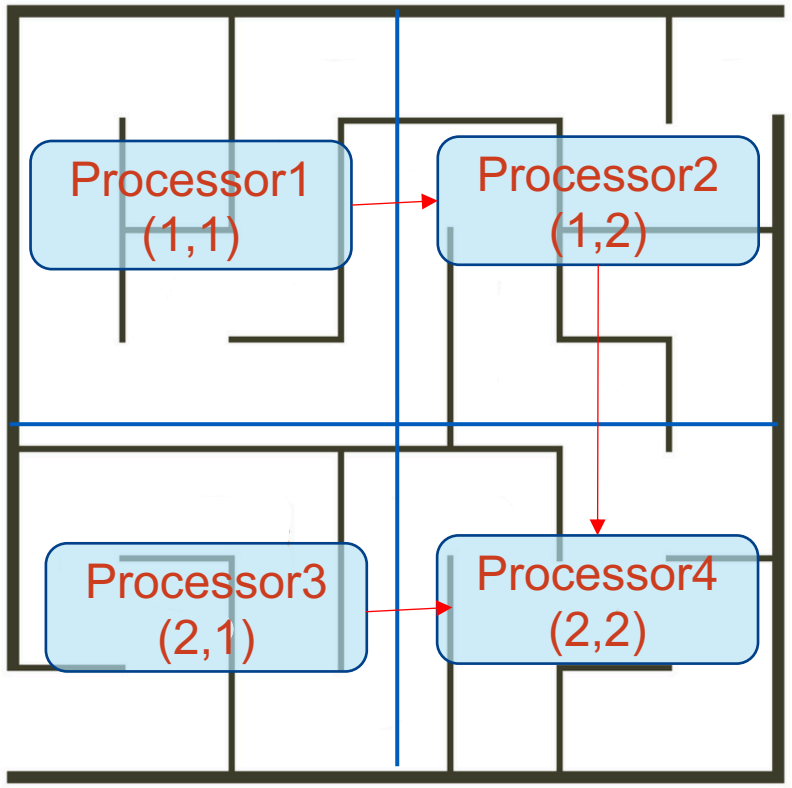
Parallel A* Algorithm

- First processor distributes subgraphs to different processors



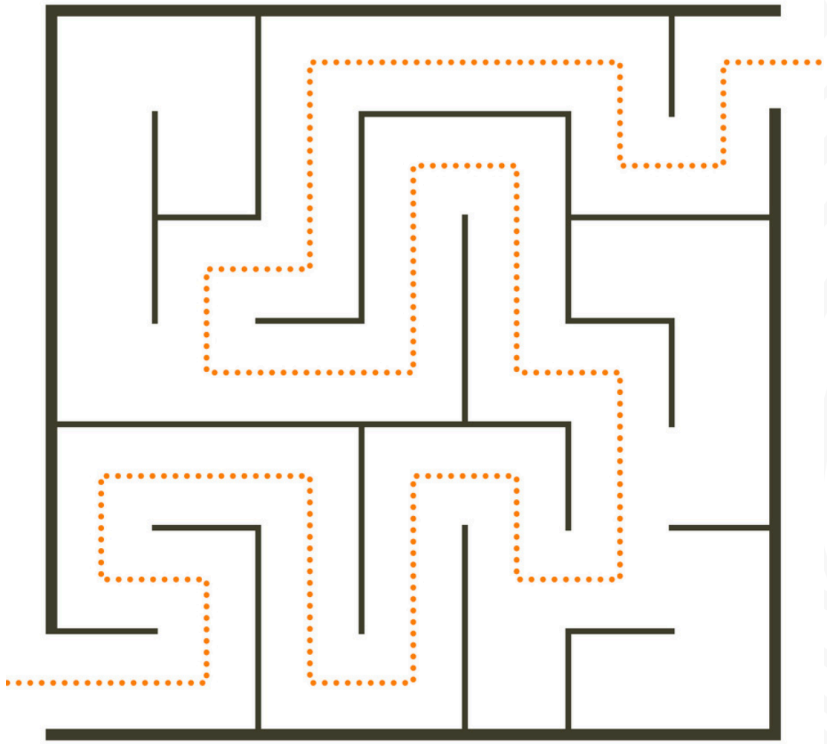
Parallel A* Algorithm



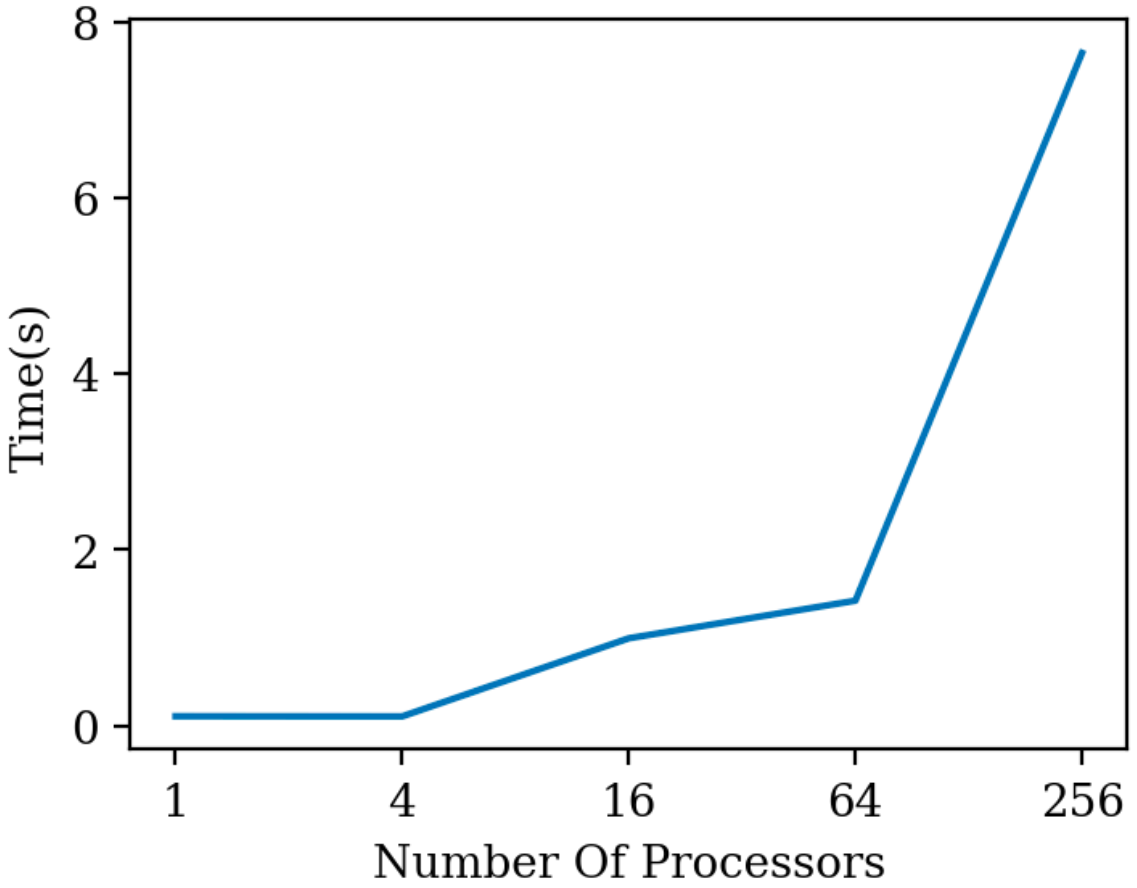


Parallel A* Algorithm

- Combine all the paths into a big graph

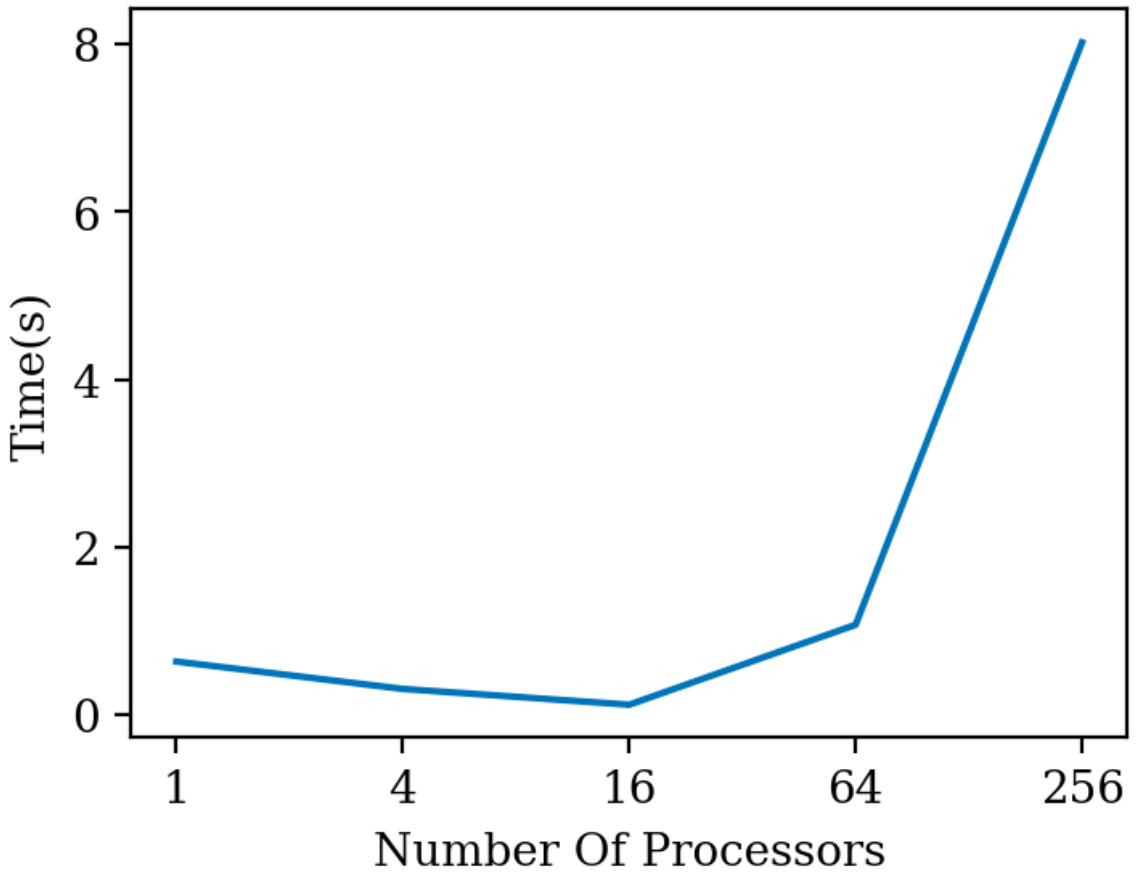


64x64 grid



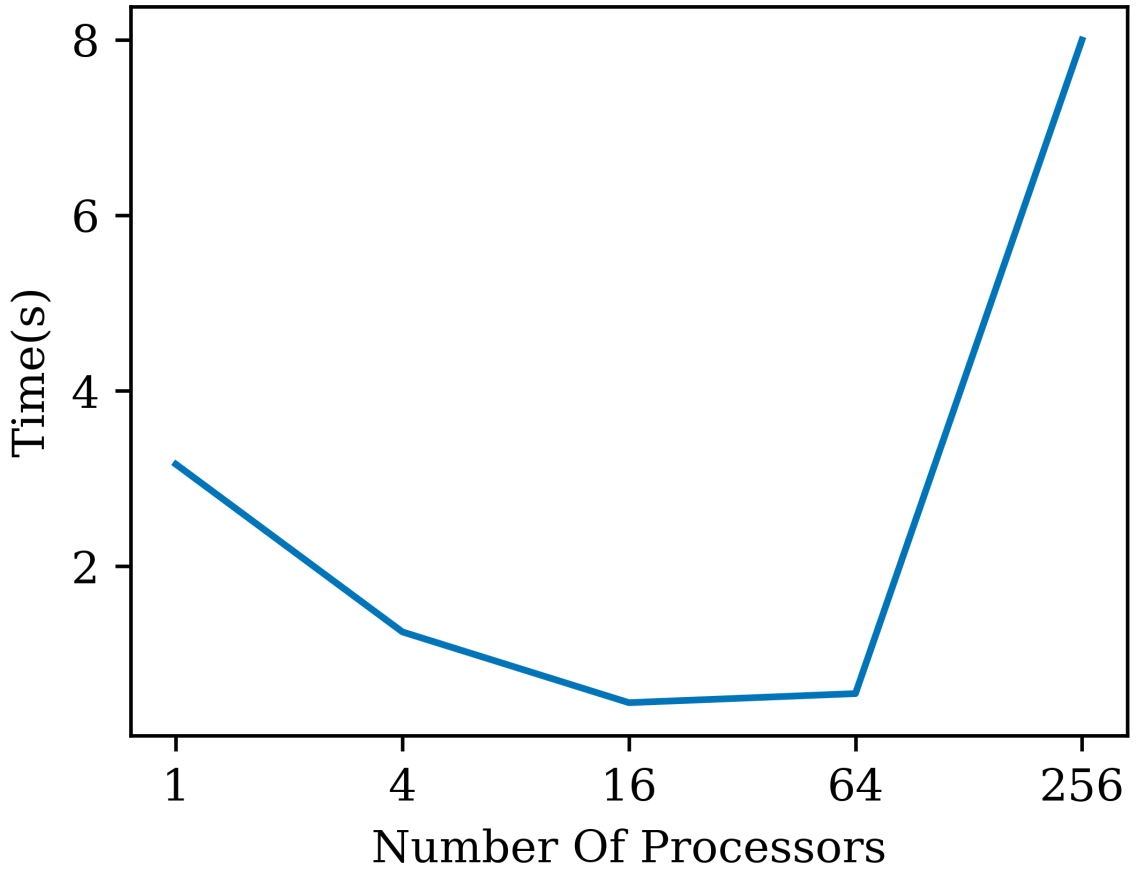
Number Of Processors	Time(s)
1	0.107
4	0.105
16	0.99
64	1.42
256	7.65

128x128 grid



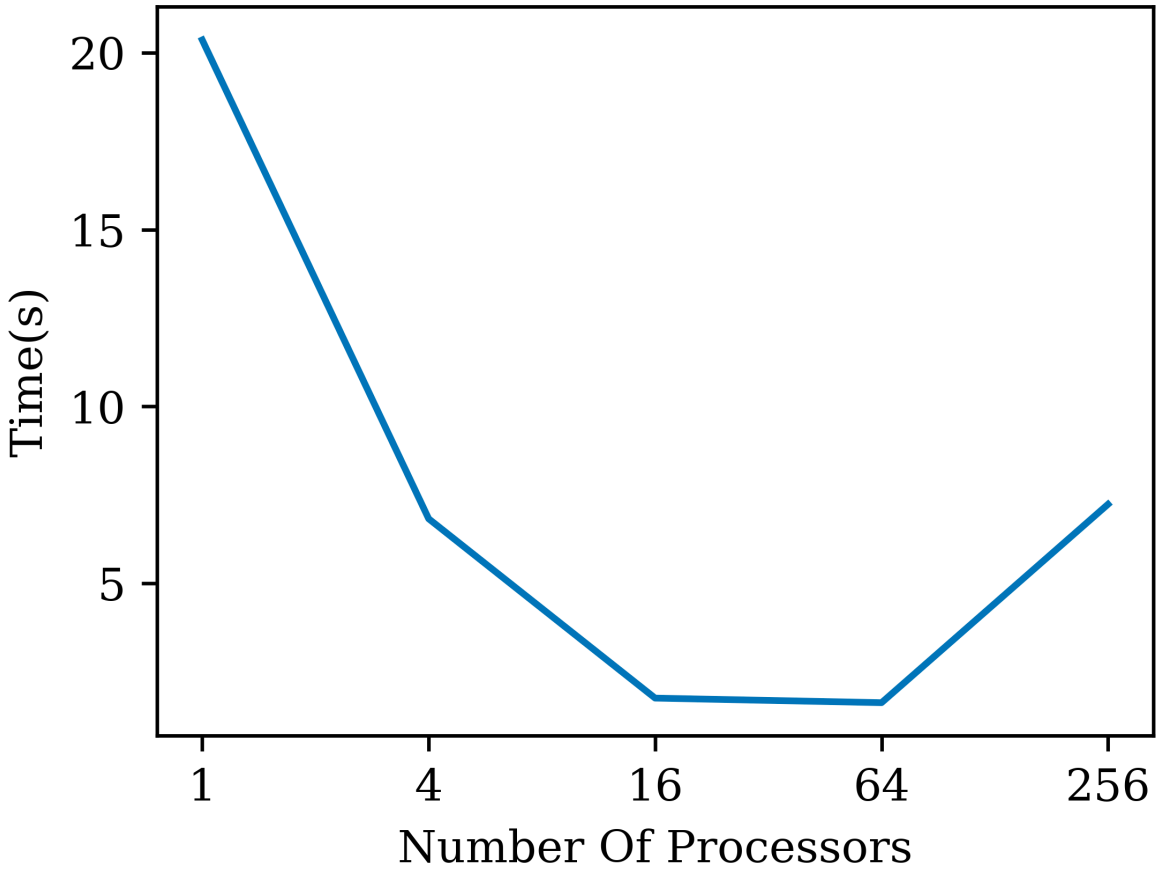
Number Of Processors	Time(s)
1	0.64
4	0.31
16	0.13
64	1.08
256	8.02

256x256 grid



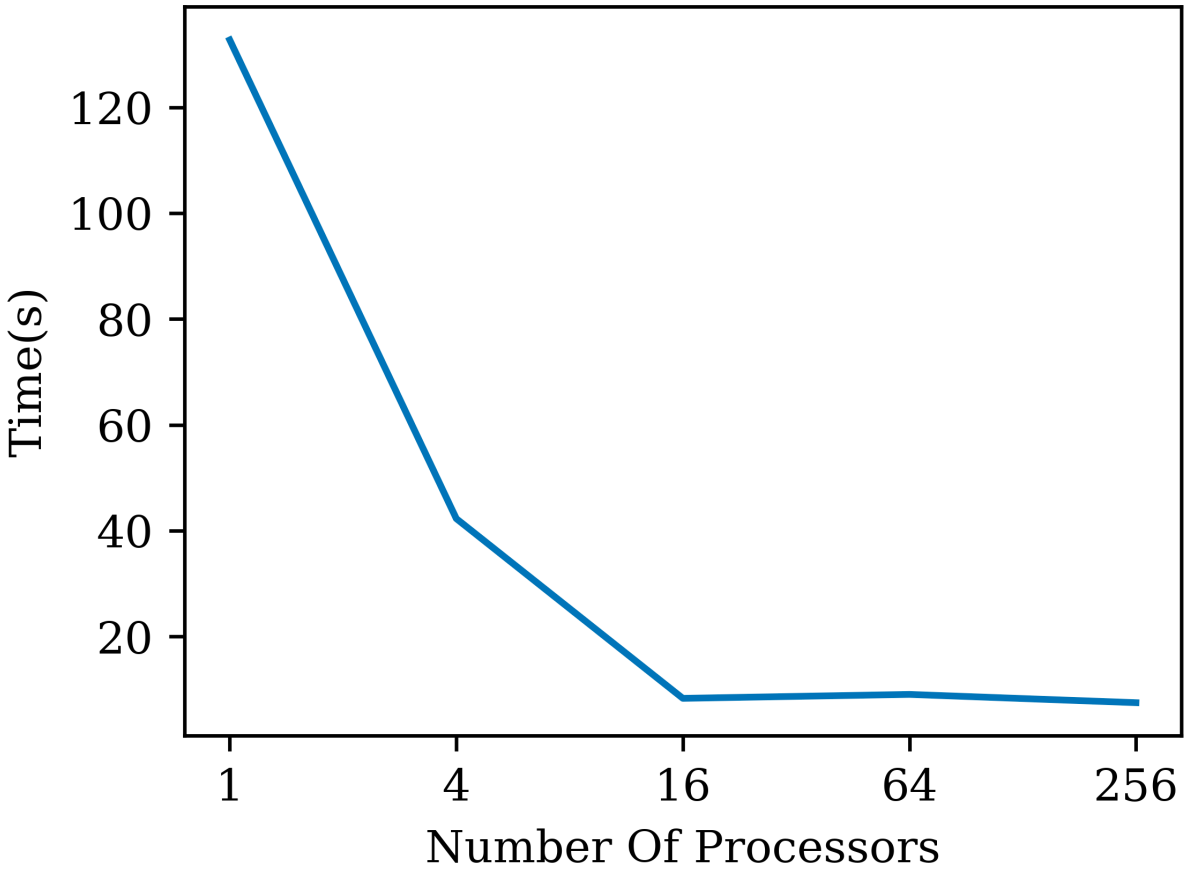
Number Of Processors	Time(s)
1	3.16
4	1.25
16	0.45
64	0.55
256	8.00

512x512 grid



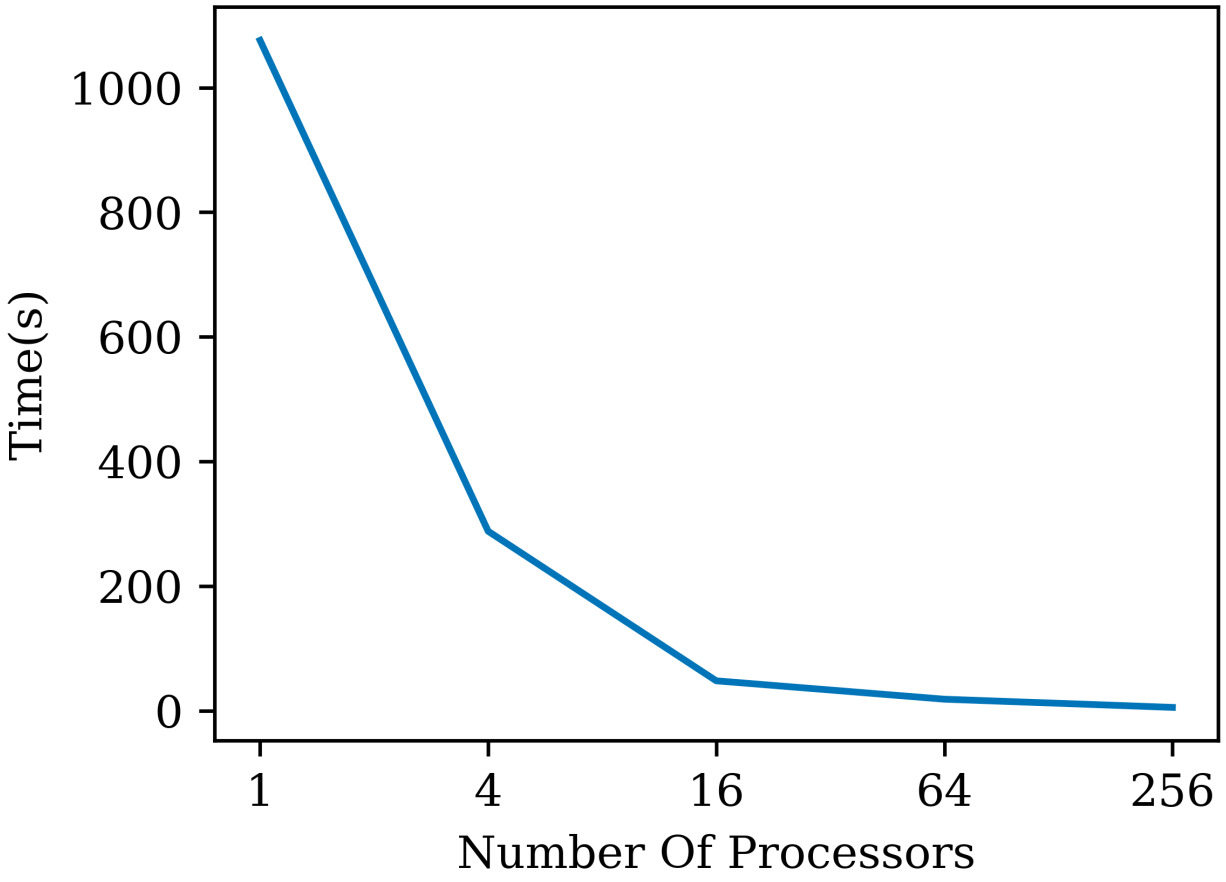
Number Of Processors	Time(s)
1	20.36
4	6.83
16	1.76
64	1.63
256	7.24

1024x1024 grid



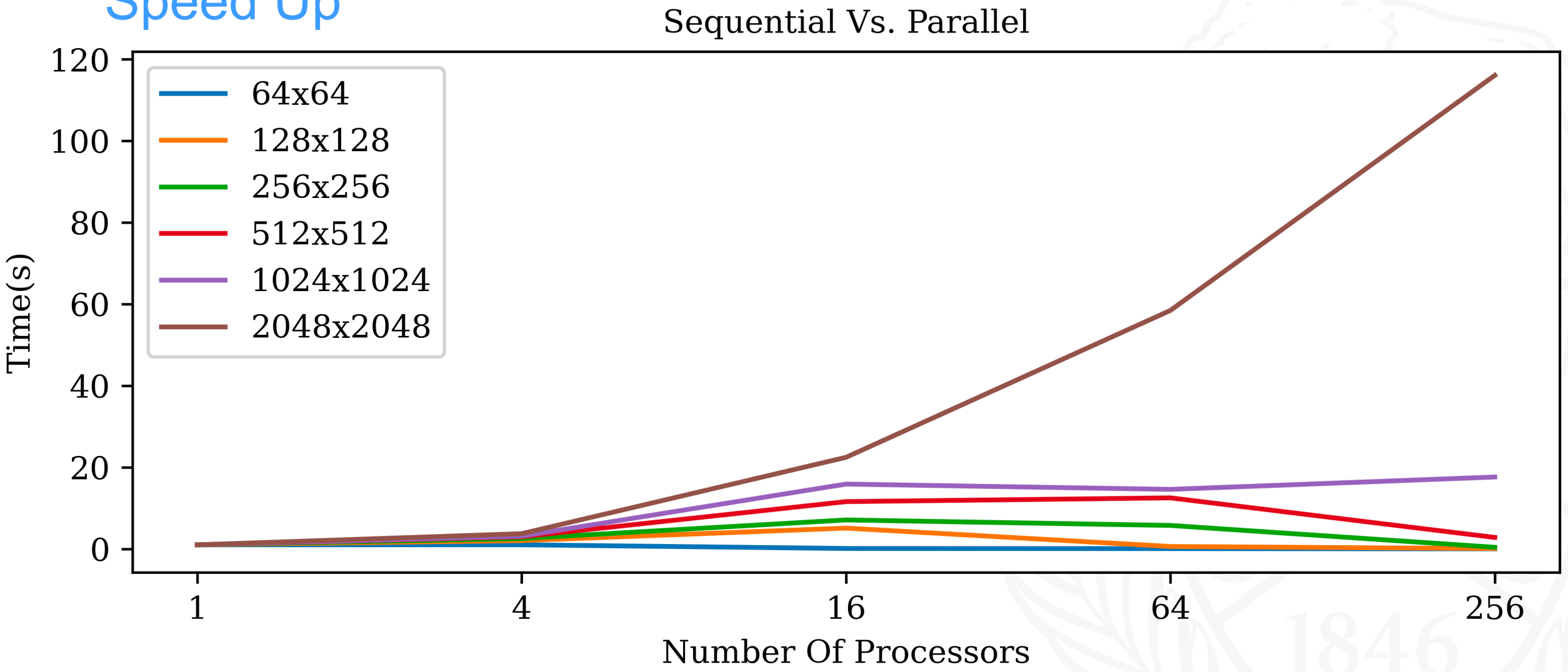
Number Of Processors	Time(s)
1	132.71
4	42.30
16	8.35
64	9.09
256	7.53

2048x2048 grid



Number Of Processors	Time(s)
1	1075.74
4	287.98
16	47.92
64	18.40
256	9.27

Speed Up


Sequential Vs. Parallel
64x64
128x128
256x256
512x512
1024x1024
2048x2048
Time(s)
0
20
40
60
80
100
120
1
4
16
64
256
Number Of Processors

References

Wikipedia contributors. "A* search algorithm." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 28 Apr. 2020. Web. 7 May. 2020.

A* Search Algorithm, GeeksforGeeks

“Algorithms, Sequential & Parallel, A Unified Approach”, Russ Miller and Laurence Boxer

“A* Search.” *Brilliant Math & Science Wiki*, brilliant.org/wiki/a-star-search/#references.

Swift, Nicholas. “Easy A* (Star) Pathfinding.” *Medium*, Medium, 1 Mar. 2017, medium.com/@nicholas.w.swift/easy-a-star-pathfinding-7e6689c7f7b2.

Questions?