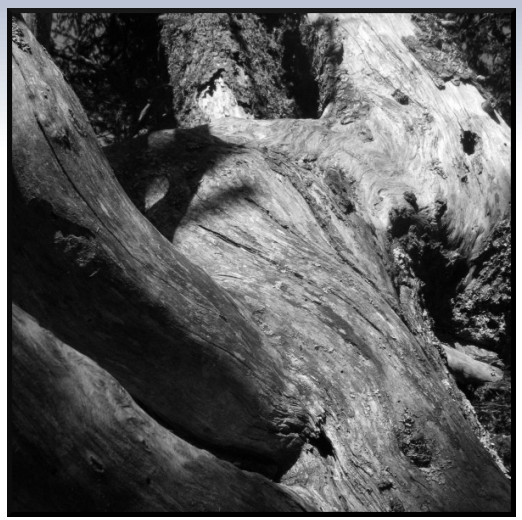


Bi-directional A* Algorithm and it's applications in finding the shortest path in road networks

---

Project Code: BPV02

Supervisor: Viswanath P

Group Members
P. Sonia (S20150010038)

D. Swathi Reddy (S20150010015)

Objectives

- Finding the shortest path in a dynamically varying Graph.
- We propose to explore the Bi-directional A* algorithm and its variants [1].
- Task-1: To implement BFS, Uniform cost search, A* search and its bi-directional variants.
- Task-2: To study the effect of nodes being added or deleted.
- Task-3: To study the effect of edges being added or deleted.
- Task-4: To study the effect of updating the edge weights.

[1] Holte, Robert C., et al. "MM: A bidirectional search algorithm that is guaranteed to meet in the middle." Artificial Intelligence 252 (2017): 232-266.

Scope of the proposed work

- Artificial Intelligence (AI) techniques.
- Graph Algorithms.

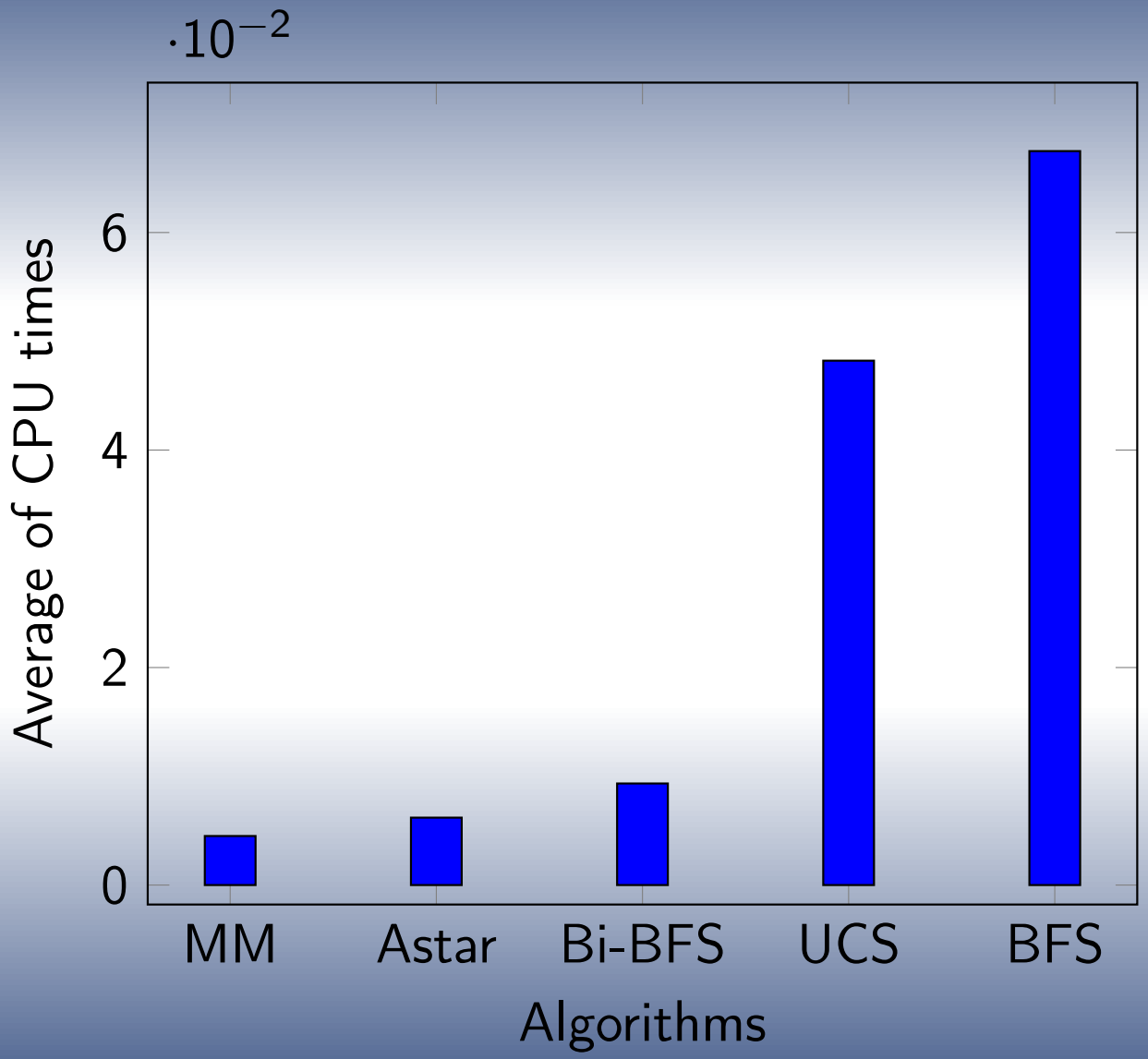
Workdone so far

- Uni-BFS (Breadth First Search)
- UCS (Uniform Cost search)
- Bi-BFS (the Bi-directional BFS algorithm)
- A* algorithm.
- MM (Modified Bi-directional A* algorithm)
- Studied the effect of deletion of a node from a graph.
- Studied the effect of insertion of a node in a graph.

Analysis of CPU times of all Algorithms

Analysis of CPU times of all Algorithms	
Algorithm	Average CPU time(sec)
MM	0.00450845454545
Astar	0.0061984
Bi-BFS	0.00934036363636
UCS	0.0482154
BFS	0.0674858181818

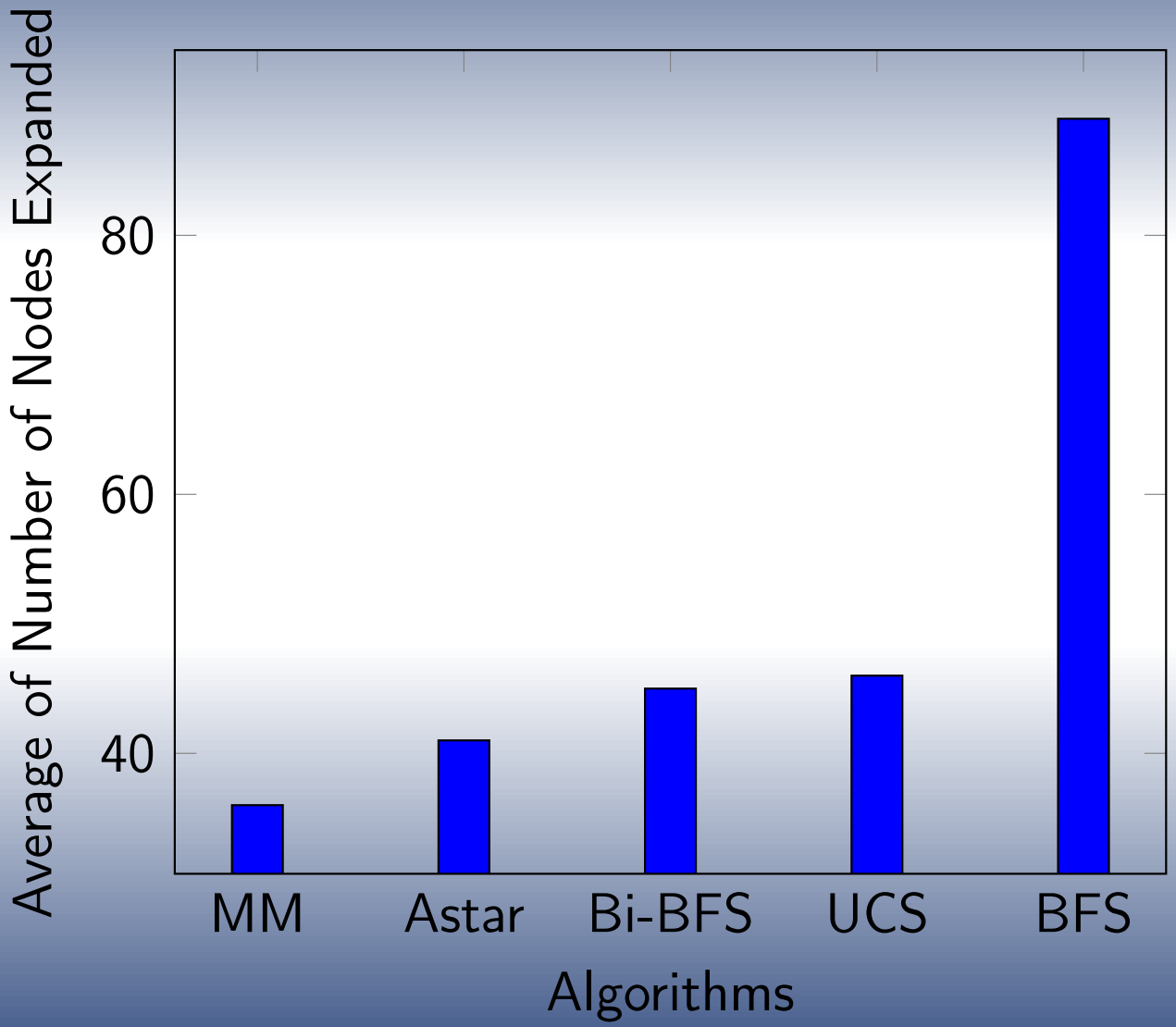
Graph of CPU times of all Algorithms



Analysis of Average Number of Nodes Expanded for all Algorithms

Analysis of Average Number of Nodes Expanded for all Algorithms	
Algorithm	Average Number of Nodes Expanded
MM	36
Astar	41
Bi-BFS	45
UCS	46
BFS	89

Graph of Average Number of Nodes Expanded for all Algorithms



Dynamic nature of graphs

Which node deletion effects the shortest path?

Contd..

- Deletion of a node from the solution path?
- Deletion of a node from the open list?
- Deletion of a node from the closed list?

Conclusion

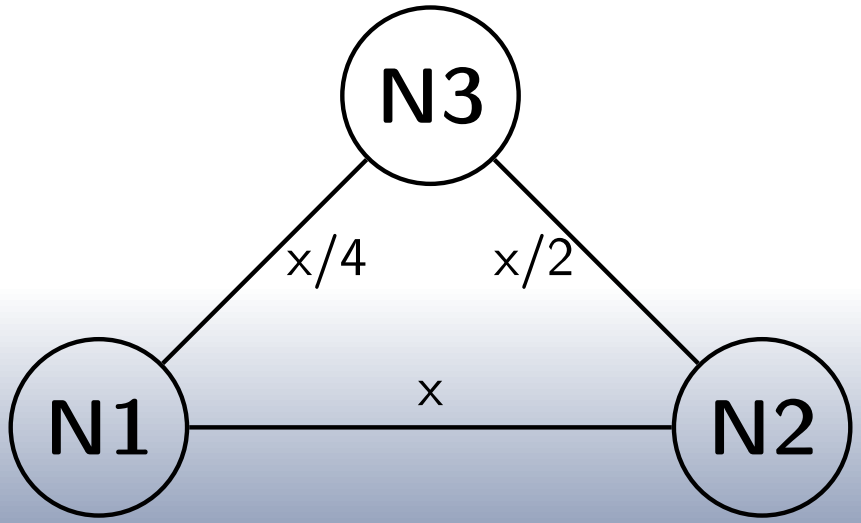
- From the results, we observed that the deletion of a node from the solution path effects the shortest path for a particular source and a destination.
- We also observed that the deletion of a node from the open queue as well as the closed queue does not effect the shortest path.

Effect of insertion of a node

Hypothesis: Let us consider two nodes N1 and N2 are the only nodes in the solution path. N1 being the source and N2 being the destination. Let us consider N3 is the newly added node.
The insertion of a node in a graph effects the shortest path of a particular source and goal iff sum of the distances between N1, N3 and N3, N2 is less than or equal to straight line distance between N1, N2 i.e.
dist(N1, N3) + dist(N3, N2) $\leq$ dist(N1, N2)

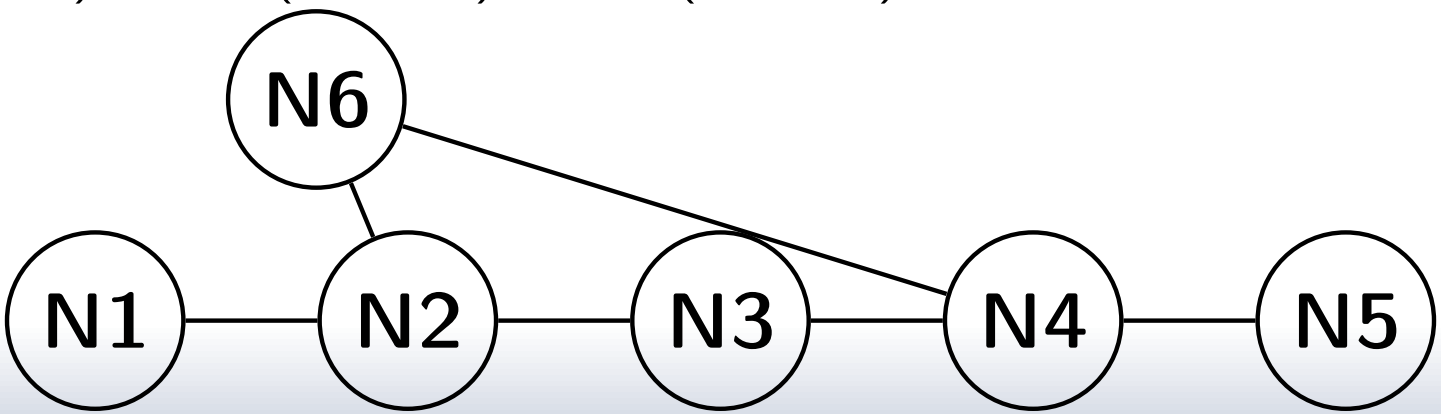
Proof

Basis: Let us consider two nodes N1 and N2 are the only nodes in the solution path. N1 being the source and N2 being the destination.
Let us consider N3 is the newly added node. The newly added node N3 effects the solution path if and only if
dist(N1, N3) + dist(N3, N2) $\leq$ dist(N1, N2)



Proof Contd...

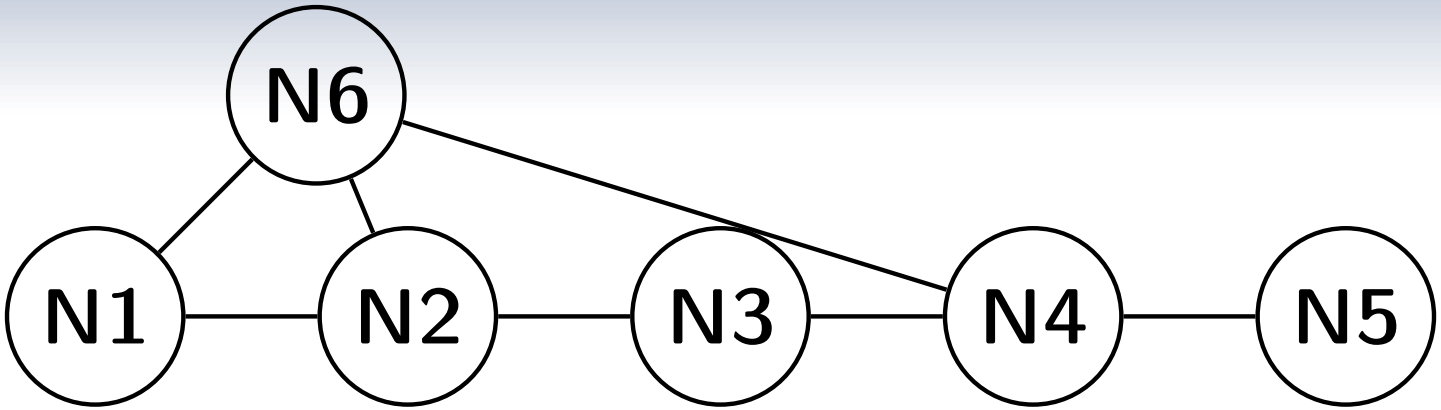
Assumption: Assuming that newly added node has k neighbours.The newly added node must be connected to atleast two nodes in a solution path either directly or indirectly. We consider N6 iff
dist(N2, N6) + dist(N6, N4) $\leq$ dist(N2, N4)



Where N6 is the newly added having K-neighbours.

Proof Contd...

Conclusion: Now, we have to prove if an added node has k+1 neighbours.



Where N6 is the newly added node having K+1-neighbours.
We will compute path from newly added node to all of it's neighbours (direct or indirect) in the solution path and take the minimum cost path among them.

Future Work

- Effect of deletion of an edge.
- Updating the weights of an edge.
- Effect of insertion of an edge or a node.

References

- Holte, Robert C., et al. "MM: A bidirectional search algorithm that is guaranteed to meet in the middle." Artificial Intelligence252 (2017): 232-266.
- Chen, Jingwei, et al. "Front-to-End Bidirectional Heuristic Search with Near-Optimal Node Expansions." arXiv preprint arXiv:1703.03868 (2017).
- Ding, Bolin, Jeffrey Xu Yu, and Lu Qin. "Finding time-dependent shortest paths over large graphs." Proceedings of the 11th international conference on Extending database technology: Advances in database technology. ACM, 2008.
- Dennis de Champeaux, Lenie Sint, An improved bidirectional heuristic search algorithm, J. ACM 24 (2) (1977) 177–191.