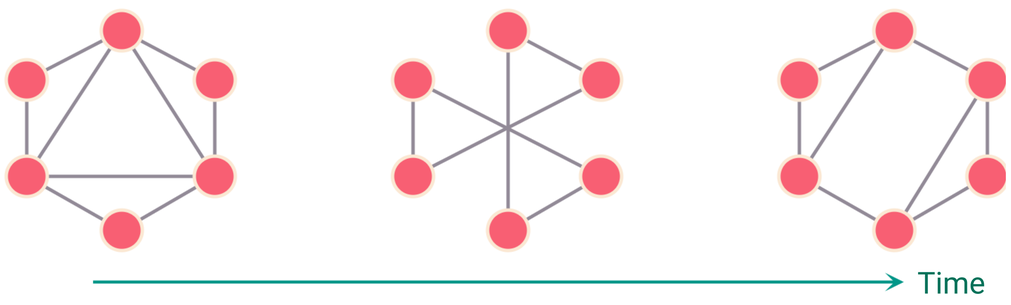
Path Planning for Dynamic Graphs using A* on GPU

Abstract

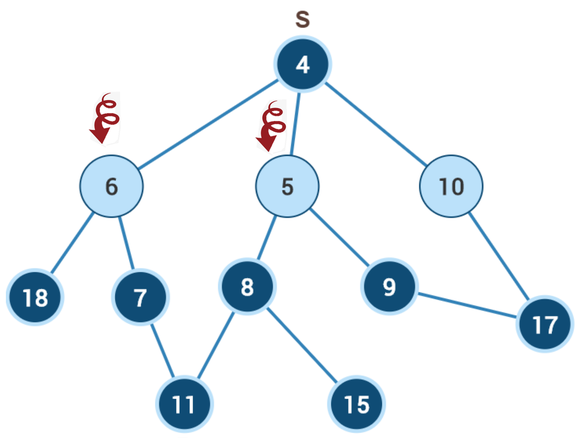
- A* is one of the widely used path planning algorithms applied in a diverse set of problems in robotics and video games.
- Here we present A* for dynamic graphs (dynamic A*) on GP-GPUs which achieves 5x-8x speedup than existing methods on the SNAP dataset [2].
- Our implementation is freely available on GitHub.

Dynamic Graphs



Static A* Algorithm on GPU [1]

- Find the optimal path from source to destination .
- Sequential in nature, minimum cost nodes are extracted from the priority queue and processed.
- Keep multiple priority queues (PQ) to extract many nodes in parallel to process.



PQ 1
6
10
-

PQ 2
5
-
-

Fig 1: Multiple threads executing A* on graph

Dynamic A*: Insertion of edges

- Newly added edges can alter the optimal path.
- To find the new optimal path instead of executing A* from scratch, we propagate the change to the affected nodes of the graph.
- Inserted edges are batched and sent to GPU for processing.

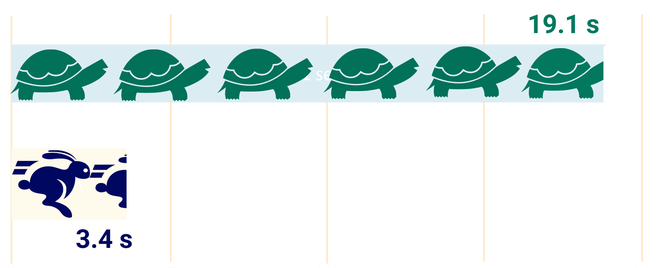
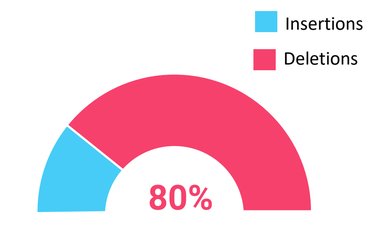


Fig 2: Execution time of static A* and dynamic A* on graph Wiki-Talk

Dynamic A*: Deletion of edges

- Deleting only that edge which belongs to the optimal path can create a new optimal path.
- For all such affected nodes recompute the cost and select the neighbour with the least cost.
- Propagate the updated cost to all the affected nodes.



80% of execution time used in processing deletions

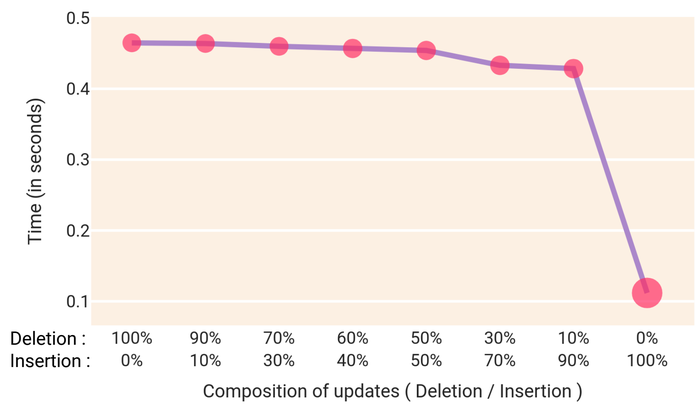


Fig 3: Execution time vs insertion / deletion

Dynamic A*: Insertion + Deletion

- The update contains both insertion and deletion of edges.
- Propagate insertions and deletions of edges separately.
- Performs better than re-executing static A* algorithm after each update.

5x Speedup by using dynamic A*

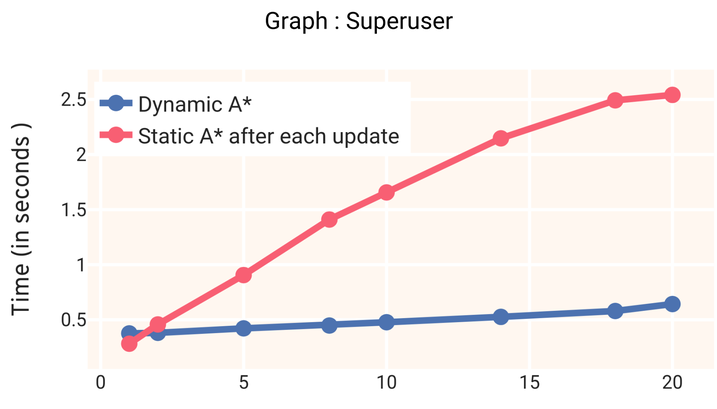


Fig 4 : Execution time vs number of updates In the graph

Results

The below table shows execution time (in seconds) and speedup of dynamic A* compared to re-executing static A* every time.

No.	Graph	Edges	Queries	Dynamic A*	Static A*	Speedup
1	Live Journal	34,681,189	10	6.01	33.93	**5x**
2	Wiki Talk	7,833,140	10	12.24	24.84	**2x**
3	Ask Ubuntu	964,437	10	0.25	1.31	**5x**
4	YouTube	2,987,624	10	0.81	5.78	**7x**
5	Math Overflow	506,550	10	0.09	0.67	**7x**
6	Live Journal	34,681,189	100	11.41	424.06	**37x**

Applications

1. We have applied dynamic A* on energy efficient routing protocol (EERP) and achieved 35x speedup from static A*.

35x Speedup by using dynamic A*

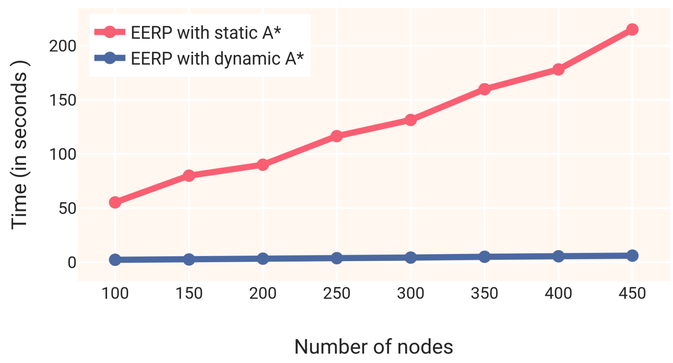
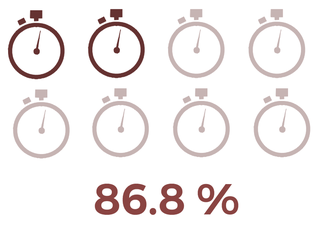


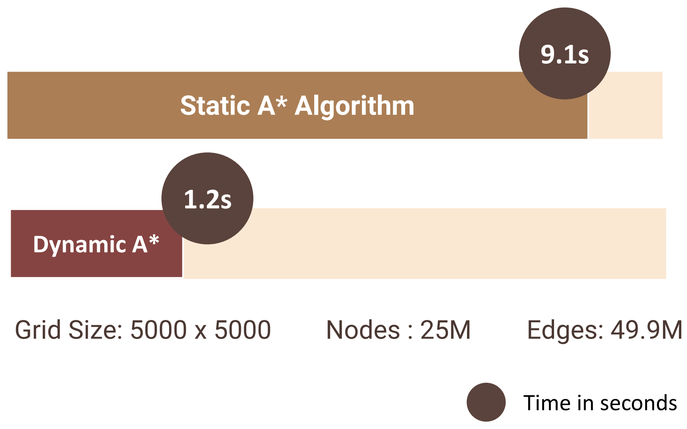
Fig 5: Comparison b/w static A* and dynamic A* on EERP algorithm

2. On applying dynamic A* for pathfinding in the maze, we achieved 8x speedup.



86.8 %

reduction in execution time by using Dynamic A* algorithm.



References

1. Yichao Zhou and Jianyang Zeng. “Massively Parallel A* Search on GPU”. In: Twenty-Ninth AAAI Conference on Artificial Intelligence (2015).
2. Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data. June 2014.