Efficient Parallel Algorithm for Shortest Path Updates in Dynamic Networks at Scale

Arindam Khanda[1], Sajal K. Das (Advisor)[2]
Email: [1]*akkcm@mst.edu*, [2]*sdas@mst.edu*
Missouri University of Science and Technology, Rolla, MO 65401

Introduction

- In dynamic networks the structure of the graph changes with time.
- The Single Source Shortest Path (SSSP) problem computes the shortest paths from a source vertex to all other vertices.
- SSSP computation in a large-scale dynamic network is challenging.

Motivational Scenario — Dynamic Route Selection for Drone Based Delivery System in Varying Wind Condition

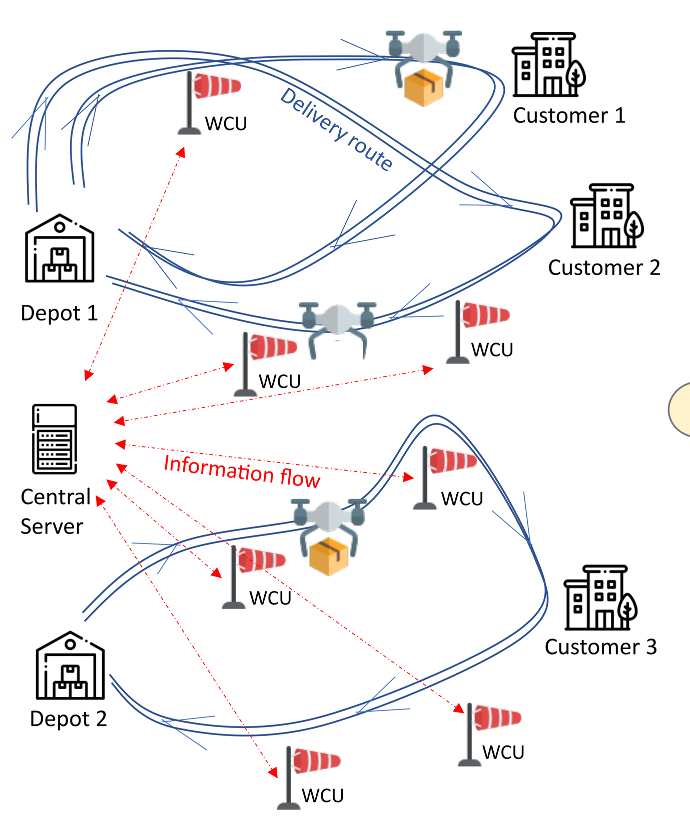

Figure 1: A drone based delivery scenario.

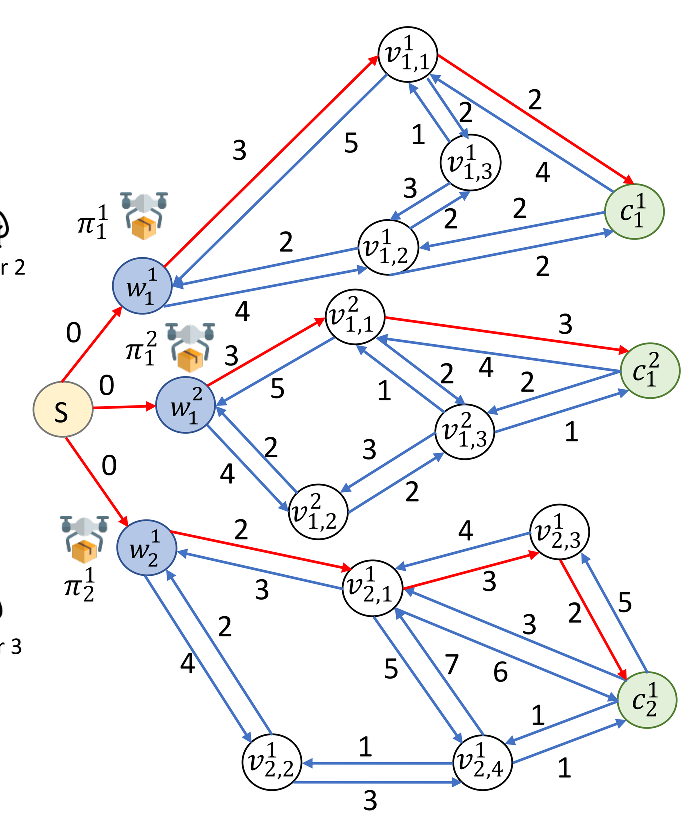

Figure 2: Delivery graph model for DBDS.

- A centralized Drone Based Delivery System (DBDS) can have multiple depots, Wind Control Units (WCU) and central server.
- WCU measure wind speed, and direction.
- Energy requirement for travelling changes due to wind characteristics.
- Finding energy efficient delivery route dynamically is a SSSP problem.

Our Approach — Parallel Shortest Path Update

- Distance and immediate parent vertex in the shortest path are stored in a tree, called shortest path tree.
- Instead of working on the whole graph, algorithm uses the SSSP tree to update the shortest path.

Problem Statement

- Let $G_{k-1}(V_{k-1}, E_{k-1})$ is a graph at time step $k-1$, and one SSSP tree of G_{k-1} is T_{k-1}.
- Changes in G from $k-1$ to k is captured by a set of changed edges ΔE_k.
- *Goal:* Compute the SSSP tree T_k at time step k, based on the previous SSSP tree and ΔE_k.

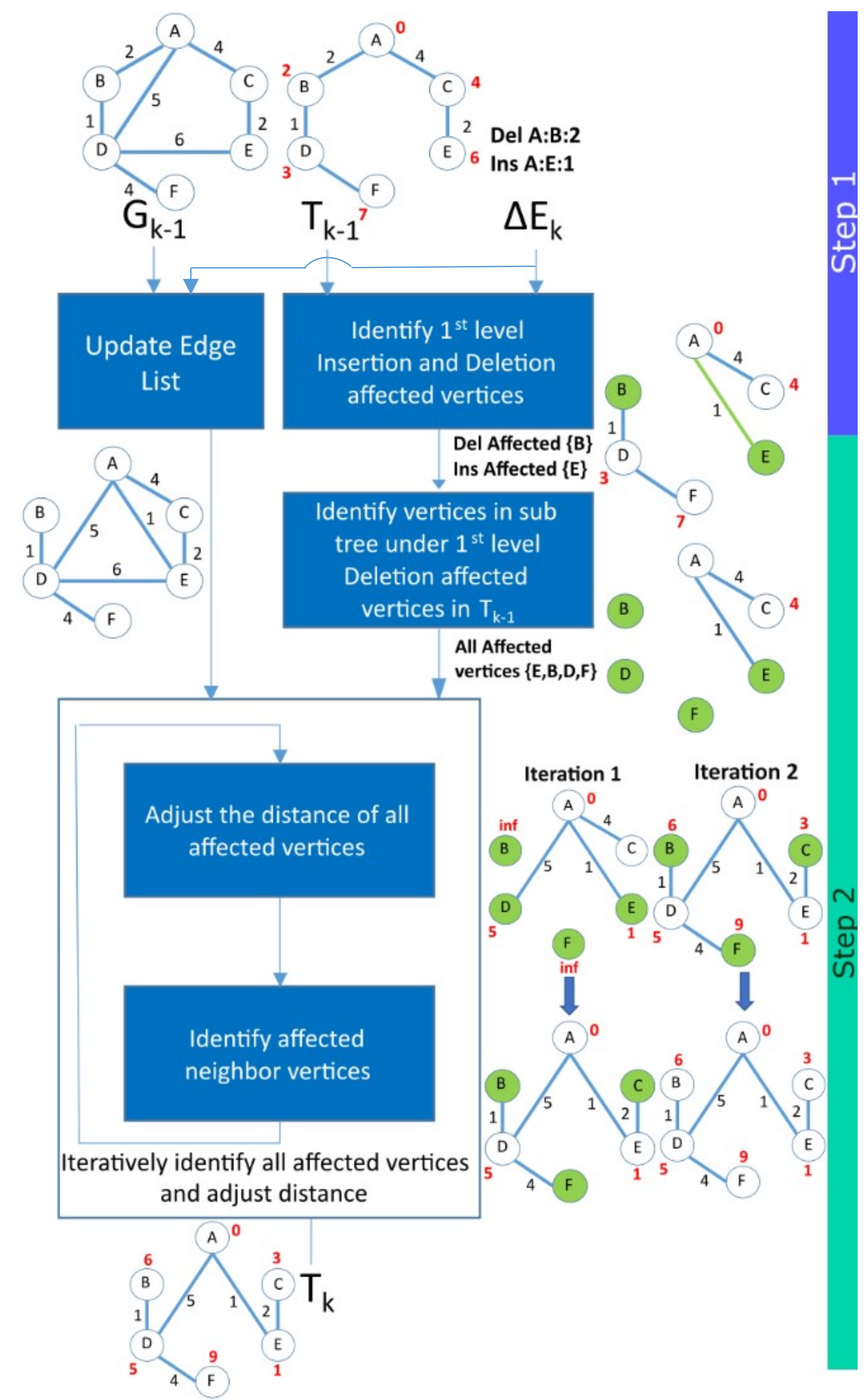

Figure 3: SSSP update example.

SSSP Update Steps

- Step 1: Process Inserted and Deleted edges in parallel and identify affected vertices (1^{st} lvl. affected).
- Step 2A: For each 1^{st} lvl. deletion affected vertex in parallel disconnect the subtrees under them.
- Step 2B: For all affected vertex update distance and parent in parallel. Identify affected neighbors.
- Step 2C: Repeat 2B until there is no affected vertex left.

Results and Analysis

Networks Used for Experiments

Name	Num. of Vertices	Num. of Edges
BHJ2015-M87101049(BHJ)	1,827,148	193,540,306
soc-Orkut	2,997,166	106,349,209
LiveJournal	12,693,249	161,021,950
RMAT24_G	16,777,215	134,511,383

- Implemented using CUDA C++ and experimented on a NVIDIA Tesla V100 GPU.
- Randomly generated 100-million edge updates for each graph dataset.
- $p\%$ insertion implies $p\%$ of total changed edges are for insertion and rest of the edges are for deletion.

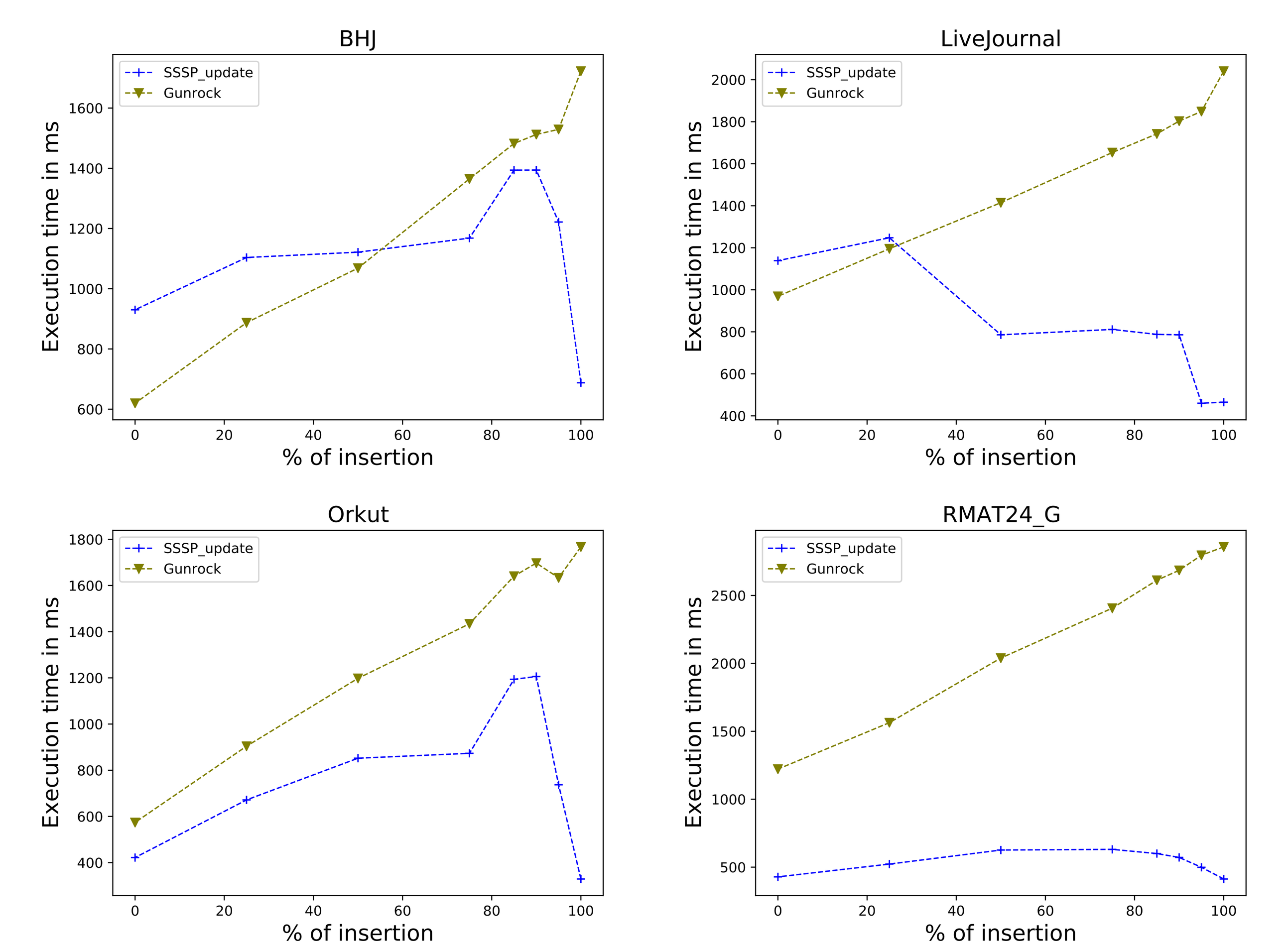

Figure 4: Execution time of our SSSP update algorithm and Gunrock. Experimented on 100 Million change edges

- Execution time depends on the graph architecture and the position of the change
- Performance is compared between our algorithm and Gunrock, the state-of-the-art GPU implementation for SSSP.
- Our algorithm outperforms Gunrock most cases.
- Usually deletion takes more time than insertion.
- Multiple edge deletion affecting same sub graph, requires less time as implementation avoids overlapping works by finding affected vertices uniquely.

Results from Drone Based Delivery Scenario

- Modeled DBDS using a real network **roadNet-PA**.
- Depot and customer vertices are chosen randomly.
- Edges of the graph are changed to simulate a varying wind model.
- SSSP update algorithm was used for finding the energy efficient delivery path in varying wind condition.
- Applied Gunrock in the same scenario for comparison.
- Our algorithm always performs better than Gunrock in DBDS scenario.

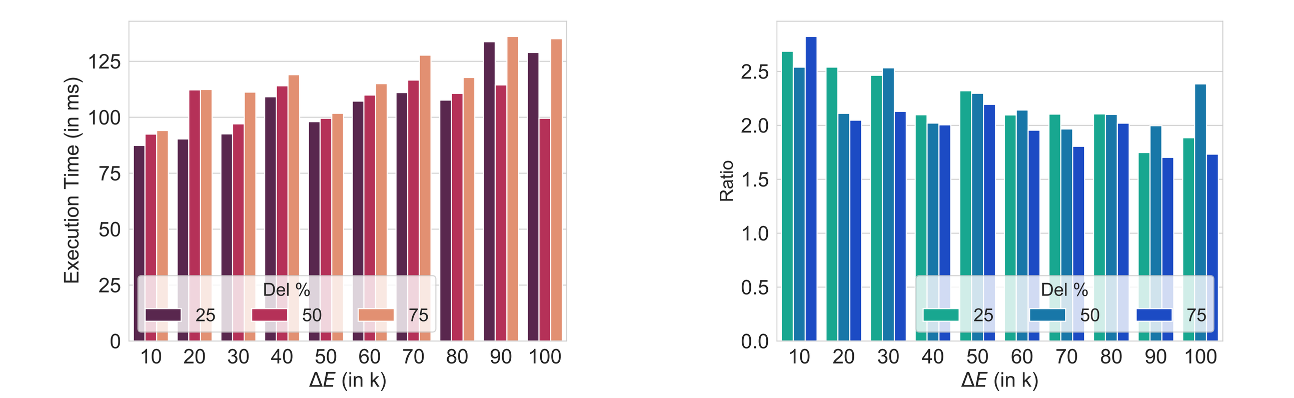

Figure 5: Left: Required time for computing energy efficient delivery route while varying the number of changed edges. Right: ratio of delivery route computing time by our algorithm and time taken by Gunrock

Conclusions and Future Work

- An efficient and scalable algorithm for updating shortest path in a large-scale dynamic network is presented.
- Proposed approach does not consider any domain constraints or dataset-specific optimization. However, in the future, we plan to use domain knowledge to improve the solution.
- Another future plan is to explore predictive algorithms for SSSP.

References

[1] A. Khanda, F. Coro, F. Betti Sorbelli, C. M. Pinotti, and S. K. Das, "Efficient route selection for drone-based delivery under time-varying dynamics," in *18th International Conference on Mobile Ad-Hoc and Smart Systems (MASS). IEEE*, 2021.

[2] F. B. Sorbelli, F. Corò, S. K. Das, and C. M. Pinotti, "Energy-constrained delivery of goods with drones under varying wind conditions," *IEEE Transactions on Intelligent Transportation Systems*, to appear, 2021.

[3] A. Khanda, S. Srinivasan, S. Bhowmick, B. Norris, and S. K. Das, "A parallel algorithm template for updating single-source shortest paths in large-scale dynamic networks," *IEEE Transactions on Parallel and Distributed Systems*, 2021.

[4] S. Srinivasan, S. Riazi, B. Norris, S. K. Das, and S. Bhowmick, "A shared-memory parallel algorithm for updating single-source shortest paths in large dynamic networks," in *IEEE 25th International Conference on High Performance Computing*, pp. 245–254, 2018.

Acknowledgment

This work is supported by the NSF OAC grants 2104078 and 1725755.