# PDC Project Phase 1: Parallel Algorithm for Updating SSSP in Dynamic Networks

Ayishah(22i-0957) Mohib Ullah(22i-1044) Maryam Farooq(22i-1217)

April 20, 2025

## Contents

# 1 Introduction

The paper "A Parallel Algorithm Template for Updating Single-Source Shortest Paths in Large-Scale Dynamic Networks" by Khanda et al. (IEEE TPDS, Vol. 33, No. 4, 2022) proposes a parallel framework for updating Single-Source Shortest Paths (SSSP) in dynamic networks with edge insertions/deletions. Unlike static graph algorithms, this framework efficiently handles structural changes in large networks (e.g., transportation, social networks). This document summarizes the paper's key findings, contributions, and proposes a parallelization strategy using METIS, MPI, and OpenMP for distributed systems.

# 2 Key Findings and Contributions

The paper presents a scalable parallel algorithm for updating SSSP in dynamic networks, focusing on edge changes.

## 2.1 Problem Context

- **Dynamic Networks**: Real-world networks change frequently, making static SSSP algorithms (e.g., Dijkstra's) inefficient due to recomputation.
- **SSSP Relevance**: Essential for network analysis (e.g., centrality measures).

## 2.2 Parallel Framework

- **Two-Step Approach**:
  1. **Identify Affected Subgraphs**: Parallel processing of edge changes to mark affected vertices, without synchronization.
  2. **Update Affected Subgraphs**: Iteratively update distances in affected subgraphs using a rooted tree, with asynchronous updates to reduce synchronization.
- **Platform Independence**: Applicable to shared-memory CPUs and GPUs.

## 2.3 Implementation

- **Shared-Memory**: Uses OpenMP, processes edge changes in batches, and employs asynchronous updates for scalability.
- **GPU**: Uses CUDA with Vertex Marking Functional Blocks (VMFB) and CSR format, parallelizing edge/vertex processing.

## 2.4 Experimental Results

- **GPU**: Up to 8.5x speedup (50M edges) and 5.6x (100M edges) vs. Gunrock when insertions more than 25%. Degrades for more than 75% deletions.
- **Shared-Memory**: Up to 5x speedup vs. Galois for 100M edge changes. Less effective if more than 75–85% nodes affected.

- **Scalability**: Improves with threads and batch processing; asynchronous updates reduce execution time.

### 2.5 Contributions

- Platform-independent parallel SSSP update framework.
- Rooted tree data structure for load balancing and synchronization.
- Batch processing and asynchronous updates for scalability.
- VMFB approach for efficient GPU computation.

## 3 Parallelization Strategy

To scale the algorithm on distributed systems, we propose using METIS, MPI, and OpenMP.

### 3.1 METIS for Graph Partitioning

- Partitions graph into balanced subgraphs, minimizing edge cuts.
- Localizes edge changes to detect affected subgraphs, reducing update scope.

### 3.2 MPI for Inter-Node Communication

- Distributes subgraphs and broadcasts edge changes.
- Synchronizes affected vertex lists and boundary vertex updates across partitions.

### 3.3 OpenMP for Intra-Node Parallelism

- Parallelizes edge change processing (Step 1) and distance updates (Step 2) within subgraphs.
- Uses dynamic scheduling to handle load imbalance.

### 3.4 Workflow

1. Partition graph with METIS.
2. Distribute subgraphs via MPI; broadcast edge changes.
3. Step 1: Use OpenMP to identify affected vertices; MPI synchronizes cross-partition effects.
4. Step 2: Update distances with OpenMP; MPI exchanges boundary updates.
5. Combine local SSSP trees via MPI.

### 3.5 Benefits and Challenges

- **Benefits**: Scalability, localized updates, efficient intra-node parallelism.
- **Challenges**: Communication overhead, load imbalance, synchronization complexity.

## 4 Conclusion

The paper's parallel framework efficiently updates SSSP in dynamic networks, outperforming recomputation (Gunrock, Galois) for insertion-heavy changes. The proposed METIS-MPI-OpenMP strategy extends this to distributed systems, enhancing scalability but requiring careful management of communication and load balancing.

## References

[1] A. Khanda et al., "A Parallel Algorithm Template for Updating Single-Source Shortest Paths in Large-Scale Dynamic Networks," *IEEE TPDS*, vol. 33, no. 4, pp. 929–940, 2022.