Parallel and Distributed Computing

Name	Roll No	Section
Haider Zia	22i-1196	CS-F
Muhammad Iqrash Qureshi	22i-1174	CS-F
Hammad Shabbir	22i-1140	CS-F

Phase 1 - Presentation: Research Paper Summary and Parallelization Strategy

Selected Paper:

Title: A Parallel Algorithm for Updating a Multi-objective Shortest Path in Large Dynamic Networks

Authors: Rakesh Misra, David A. Bader

1. In-Depth Study of the Paper

This paper addresses the computational challenge of updating multi-objective shortest paths (MOSP) in large, dynamic networks. Unlike traditional shortest path problems which optimize a single metric (e.g., distance), MOSP aims to consider multiple conflicting objectives such as time and energy. Real-world examples include transportation systems, sensor networks, and logistics planning where dynamic changes (like traffic or node failure) require efficient path re-evaluation.

Key challenges include:

- The computational complexity of computing Pareto-optimal paths in a changing network.
- The inefficiency of recalculating paths from scratch upon each change.
- Limited research on parallel MOSP update algorithms.

The paper introduces:

- A parallel algorithm for Single-Objective Shortest Path (SOSP) updates.
- A heuristic method to construct MOSP paths by combining SOSP updates.
- A shared-memory implementation using OpenMP, achieving significant speedups compared to sequential recalculations.

2. Key Contributions

1. **Novel Parallel SOSP Update Algorithm:**

   a. Efficiently updates shortest path trees after edge insertions without full re-computation.
   b. Parallel threads handle affected vertices, minimizing race conditions using structured queues.
2. **Heuristic for Multi-Objective Path Construction:**
   a. Individual SOSP trees are computed for each objective.
   b. A weighted union of these trees forms a meta-graph for deriving near-Pareto-optimal paths.
3. **Scalability & Performance:**
   a. Implemented in shared-memory (OpenMP) environments.
   b. Tested on large synthetic and real-world graphs, demonstrating up to 15x speedups.
4. **Framework Reusability:**
   a. Modular approach that allows reuse across different systems and objectives.

3. Proposed Parallelization Strategy

To implement and extend this work in a distributed setting, I propose the following strategy:

A. MPI (Inter-node Communication)

- Partition the graph into subgraphs and assign each subgraph to a node.
- MPI handles communication of updated distances, affected vertices, and frontier exchanges.
- Synchronize global updates after local processing to ensure consistency of MOSP paths across partitions.

B. OpenMP (Intra-node Parallelism)

- Each node (or partition) uses OpenMP to process updates in parallel:
  - Update distance values for affected vertices.
  - Propagate changes in parallel through local queues.
  - Avoid locks by iterative convergence, as in the original paper.

C. METIS (Graph Partitioning)

- Use METIS to partition the large dynamic graph into balanced subgraphs:
  - Minimizes inter-node edge cuts.
  - Improves data locality.
  - Enhances scalability of both MPI and OpenMP phases.

4. Conclusion

The selected paper provides a strong foundation for MOSP updates in dynamic networks. The proposed parallelization strategy using MPI for distributed processing, OpenMP for shared-memory acceleration, and METIS for partitioning will allow the algorithm to scale across modern high-performance computing architectures. This approach is promising for real-world applications in logistics, networking, and transportation systems.