**Phase 2 Report:   
Parallel Algorithm Implementation and Demonstration**

**Muhammad Tahir**

**Sameed**

Contents

**No table of contents entries found.**

# 1. Introduction

* Brief overview of the algorithm (e.g., PSAIIM) and its purpose.
* Motivation for parallelization (e.g., large-scale social graphs).
* Tools and technologies used (MPI, OpenMP/OpenCL, METIS).

# 2. Implementation Details

# 2.1 Parallelization Strategy

* Explain the core idea behind parallelization.
* Outline how MPI is used for inter-process communication.
* Justify the choice between OpenMP and OpenCL (or both).

# 2.2 Graph Partitioning

* Describe how METIS was used to partition the input graph.
* Mention type of partitioning (e.g., k-way, edge-cut).
* Show a sample graph and its partitioning (figure recommended).

# 2.3 Dataset Handling

* List datasets used (e.g., SNAP, Facebook, Twitter social networks).
* Size and characteristics of each dataset.
* Preprocessing steps (e.g., removing isolated nodes, reindexing).

# 3. Experimental Setup

## 3.1 Hardware/Environment

* Number and specifications of machines used.
* Operating system and MPI version.
* Compilation commands and flags.

## 3.2 Execution Parameters

* Number of processes and threads used.
* OpenMP schedule or OpenCL kernel config (if applicable).
* METIS parameters.

# 4. Results and Analysis

## 4.1 Performance Metrics

* Execution time
* Speedup
* Efficiency
* Scalability (weak/strong scaling)

## 4.2 Visualizations

* Line/bar charts comparing:
  + Sequential vs. MPI vs. MPI+OpenMP/OpenCL.
  + Execution time vs. number of processors.
  + Speedup vs. processors.
  + Partitioning quality (edge cut, load balance)

## 4.3 Scalability Discussion

* **Weak scaling**: Keep workload per processor constant.
* **Strong scaling**: Fixed total workload, increase processors.

# 5. Discussion

* Challenges encountered (e.g., communication overhead, load imbalance).
* How METIS improved load balancing.
* Effect of hybrid MPI+OpenMP/OpenCL approach.
* Accuracy of results vs. performance gain.

# 6. Conclusion and Future Work

* Summary of improvements due to parallelization.
* Best configuration based on results.
* Future improvements (e.g., dynamic partitioning, GPU acceleration).
* Portability and limitations.

# 7. References

* Research paper(s) implemented.
* Libraries: METIS, MPICH, OpenMP/OpenCL docs.
* Dataset sources (e.g., SNAP dataset citation).

**8. Appendix**

* Code snippets (e.g., MPI communication, METIS partition integration).
* Compilation/run scripts.
* Raw data (optional, or link to GitHub).

Would you like this in a downloadable .docx or LaTeX format as well?