Kth Shortest Path

This program calculates the kth shortest path between random pairs of nodes in a given graph. It reads the graph data from a CSV file, computes the shortest path using Dijkstra's algorithm, and then finds the kth shortest path utilizing parallel processing with OpenMP and MPI.

Compilation

To compile the program, ensure you have the necessary libraries installed (OpenMP and MPI), and then use the following command:

```
mpicc -o mpi FileName.c
```

Execution

To execute the program, use the following command:

```
mpiexec -n <num_process> ./mpi
```

Replace <num_processes> with the number of MPI processes you want to use.

Input Data

The program expects the graph data to be provided in a CSV file (Must Change the file name before Running). The CSV file should contain three columns: source node, target node, and edge weight.

Output

The program outputs the kth shortest paths for randomly selected source and destination nodes. It also displays the execution time of the program.

Code Structure

readNoLines: Reads the number of lines in the CSV file.

read_csv_and_transform: Reads the CSV file and transforms the data into edge structures.

displayCSVData: Displays the CSV data.

displayCityNodes: Displays unique city nodes.

dataTODistanceMatrix: Transforms the data into an adjacency matrix.

shortestPath: Calculates the shortest path using Dijkstra's algorithm.

shortestPathMemoized: Use the Previous Calculated Path to optimize the time.

findKthPath: Finds the kth shortest path using parallel processing.

main: Entry point of the program. Initializes MPI, reads data, broadcasts data to all processes, generates random source and destination nodes, computes kth shortest path, and outputs the result.

Parallel Processing

OpenMP is used for parallelizing the Dijkstra's algorithm. MPI is used for parallelizing the computation across multiple processes.

Performance

The performance of the program can be improved by adjusting the number of MPI processes and OpenMP threads according to the hardware specifications and the size of the graph.