Group Assignment

CS 4990 Social Computing

Grade: 20 points.

* **This is a group project.**

**Due Date: 05/09 11:59PM (with presentation May 10-13)**

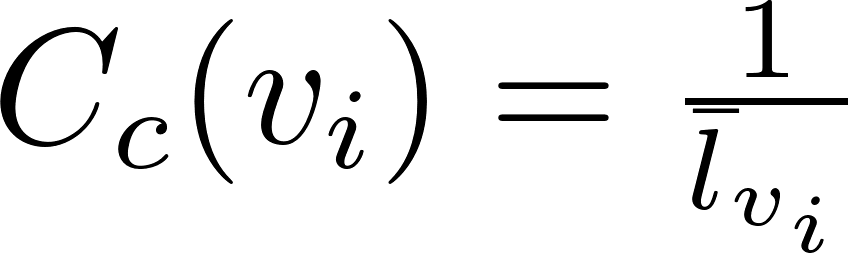
**Late submission (without penalty): 05/15 (with presentation May 16-20)**

**Computational resource:** You may use the cluster resource provided for this class and/or personal computer.

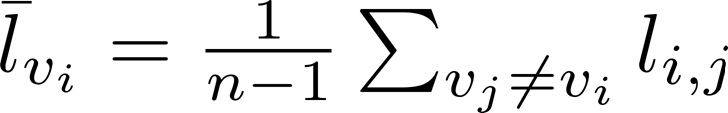
**PROJECT DESCRIPTION**

1. **Design and implement a parallel algorithm for closeness centrality.**

* **Problem statement**: Design a parallel algorithm that will process a graph of size *n* and calculates the closeness centrality for each node. You will implement this algorithm using Message Passage Interface (MPI) libraries for Python[[1]](#footnote-1). Note that the closeness centrality for a node is defined as:



Where,



**Input:** You will use the following two social networks datasets for your performance analysis (from the SNAP[[2]](#footnote-2) website). They are:

1. Facebook dataset. The dataset is provided on this page:

<https://snap.stanford.edu/data/ego-Facebook.html>

Specifically, use the file facebook\_combined.txt.gz <https://snap.stanford.edu/data/facebook_combined.txt.gz>

2. Twitter data. The twitter dataset is provided on this page:

<https://snap.stanford.edu/data/ego-Twitter.html> (Note the statistics about this dataset provided on this page) Specifically, you will use the file titled “twitter\_combined.txt.gz”:

<https://snap.stanford.edu/data/twitter_combined.txt.gz>

* **Output:** The first processor (ID=0), will print out to a local file called “output.txt”, the centrality measures for all the vertices. The processor will also print five nodes with the top centrality values (if there are more than five nodes with the centrality values, then print any five) and also the average of the centrality values of all nodes on screen.
* **Program design:** You may use any APSPs algorithm to calculate the shortest paths required in centrality calculation. The key aspect of the project is to come up with a good parallel design. Points will be given for efficient design.
* **Performance study:** Tabulate the running times of your implementation with varying number of processors = 4,8, 16,…... perform your study using your chosen social networks. Prepare plots showing the trends in the run time. Answer questions, such as
  + what is the parallel speedup for your implementation? Plot the speedup with P=4,8,16…
  + What is the cost of the implementation? Plot the cost with various values of P.
* **Report:** You will prepare a brief report that will describe your design, pseudocode, data structures, performance study results, and any other details required to evaluate your algorithm and implementation. You will also provide a time complexity and space complexity analysis of your algorithm.
* You will use the **MPI library for python** to implement the algorithm.
* You will conduct performance comparison of the serial and parallel versions of the centrality calculation algorithm.

1. **Deliverables:**

All submissions, unless otherwise specified, will be done through Blackboard using the link for “Project” under the “module and course schedule” folder.

Initial Report

**Deadline:** April. 25th

* You will submit an initial report on your plan for implementing the parallel algorithm. This should include the specific algorithm that you are using and your plans (pseudocode, performance metrics, experiment plan, etc.) for parallelizing it. This report be in the form of a ppt file with 5 slides at the most.
* A team will submit a single copy of the slides using the link for initial presentation under the “Group Assignment” folder.
* The instructor will schedule a 10 minute discussion with the team on April. 26th / 28th to go over the report.
* Non submission will lead to a reduction of 3 pts from the final score. Points will also be reduced if the team does not meet with the instructor.

Final Submission and presentation

**Deadline for submission: May 9th 11:59 PM**

* A PowerPoint file that provides a description of your design, analysis. You may use flowcharts, UML etc. to provide details of your implementation. You will also provide the tasks performed by each team member.
* The source code files.
* The compiled files
* Each member of the team will submit the slides (same copy) using the link for “final submission” under the “Group Assignment” folder.

**Each team will schedule a presentation with the instructor (May 10 - 13).** Your presentation will last about 20 minutes. The instructor will email you to set up the presentations.

**Anyone who misses the final presentation will not receive a grade for the project.**

1. <https://mpi4py.readthedocs.io/en/stable/> [↑](#footnote-ref-1)
2. <https://snap.stanford.edu/data/> [↑](#footnote-ref-2)