UNIVERSITY OF WASHINGTON, TACOMA
TCSS 543: Advanced Algorithms
Empirical Study Project
Due: Saturday, November 30, 2019, 1:30pm

Autumn 2019 November 7, 2019

Overview

In this project, you will be conducting an empirical study to see which of the network flow algorithms is better than the others. It is unlikely that you will discover that one algorithm is better than the others for all input. In fact, our goal in this project is to see if you can figure out what kinds of graphs one algorithm does better than the others.

The algorithms you will implement and run experiments on are: the Ford-Fulkerson algorithm, the scaling Ford-Fulkerson algorithm, and (for bonus) the preflow-push algorithm.

Code

Graph code and some code to generate graphs will be provided. You should not change code to the graph code unless we as a class decide to change it. The reason for this is so that the underlying graph code is all the same, which allows us to make fair comparisons among different groups.

Input Graphs

You might consider the following when generating input graphs:
1. Random graphs (each edge is generated with probability p).
2. Mesh graphs.
3. Bipartite graphs.
4. The range of capacities on the edges.

Group Report

In addition to your code for your empirical study, you are expected to write a report on your work. The report is not quite an academic paper, but it follows roughly the same format. Your report should be divided into sections as follows:

Abstract: A description of the problem (network flow), what the goals of the study are, and a brief description of the results.

Methodology: A description of the code you used/borrowed, what code you wrote yourself, what algorithms you implemented, implementation difficulties, etc. You should also describe how you tested your code, how you generated your input instances (e.g., how you generated your graphs), etc. For example, for network flow, how did you choose your s and t?

You should also say something about how many times you ran your algorithm on input instances, what range of graph sizes (number of vertices and number of edges) you used, and how you systematically investigated the algorithms.

Results: What did you discover about your algorithms? Which was faster? Was one or more algorithm more sensitive to the number of edges in the graph, the number of vertices in the graph (or in network flow, the value C)?

Future work: If you were to do the experiments over again, what would you do differently? What future experiments can you think of that might be interesting, given what you learned from this project?

Individual Report

You will also turn in an brief individual report (at most one page), which will:
1. Describe the parts of the project you worked on (which algorithm(s) you implemented, which input graphs you tested, who you worked with on what parts, etc.) **and** what parts of the project your teammates worked on.
2. What you learned from the project.

We will discuss the details of this project in class.