MatrixBFS
Parallel BFS using Matrix Partitioning

Blake Appleby (aba2176)
Jonathan Hall (jah2328)
Ryan Wee (rw2800)

November 27 2023

1. Introduction

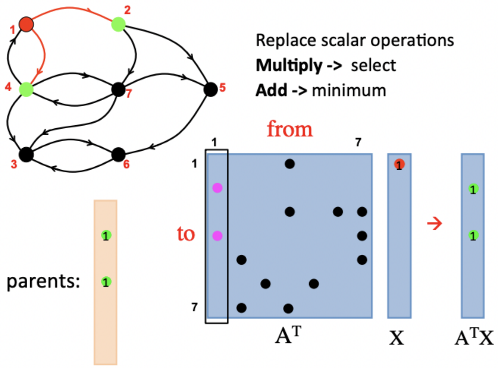
The theory of "six degrees of separation" states that any two individuals are separated by at most six interpersonal connections. (In popular culture, one of these individuals is usually Kevin Bacon, and the other is some Hollywood actor.) In this project, we apply this theory to the dataset of Wikipedia pages. In particular, we say that Wikipedia page A experiences *one degree of separation* from Wikipedia page B if and only if the contents of A contain a hyperlink to B. Given two arbitrary Wikipedia pages, our goal is thus to verify whether they are indeed connected by at most six degrees of separation.

2. Overview

It is quickly apparent that this is essentially a shortest-path problem on a directed graph with fixed-weight edges. In particular, we can think of each Wikipedia page as a node, and each hyperlink on page A to page B as an edge from A to B. To solve this problem, we can do a naïve breadth-first search (BFS) from A, and find the shortest path to B.

3. Opportunities for Parallelization

There is ample pre-existing material on how BFS can be parallelized using matrix partitioning.[1] The central idea is that BFS can be represented using matrix algebra. In particular, represent the adjacency matrix as an N by N matrix A^T, where $A^T_{i,j}$ is 1 if there is an edge from node j to node i and 0 otherwise. Represent the current frontier as an N by 1 matrix X, where $X_{i,0}$ is 1 if node i is in the current frontier and 0 otherwise. Then we can obtain the next frontier by taking $A^T X$, as seen below.[2]



This matrix multiplication can be parallelized among C cores by partitioning the adjacency matrix A^T into C submatrices, either using 1-dimensional or 2-dimensional decomposition. 2-dimensional decomposition can be further optimized with direction optimization. Our plan is to implement as many of these strategies as possible, following the order above.

4. Opportunities for Scaling

We can test our implementation on datasets of varying sizes by considering only a subset of all Wikipedia pages.

[1] See: `people.eecs.berkeley.edu/~demmel/cs267_Spr16/Lectures/CS267_March17_Buluc_2016_4pp.pdf`, `people.eecs.berkeley.edu/~aydin/sc11_bfs.pdf`, `en.wikipedia.org/wiki/Parallel_breadth-first_search`, and `www3.nd.edu/~zxu2/acms60212-40212/Lec-07-2.pdf`.

[2] Image taken from the Berkeley CS 267 PowerPoint linked above.