We Want More Time and Less Work

Matthew Hudson Walter Xia

October 15, 2016

Abstract

Our paper is *Time-Work Tradeoffs for Parallel Algorithms, Spencer[2]*. The high-level idea that serves as the theme of *Spencer[2]* is the observation that there exists algorithms with the property that the amount of work they perform is inversely related to the amount of time they take. In particular, *Spencer[2]* discusses the following problems:

- Solving Triangular Systems of Linear Equations
- Topological Sort
- Breadth-First Search in Directed Graphs
- Strongly Connected Components
- Single Source Shortest Path

In particular, it presents a few new list data structures called *nearby lists*, *frontier lists*, *inverse nearby lists*, and *inverse frontier lists* that provide the necessary bookkeeping for the subsequent algorithms to be presented.
/* Note to Matt: I may have omitted stuff from your parts. For example, stuff related to probability. If you think we should add anything here, please do so. */

1 INTRODUCTION

Spencer[2] focuses on the problems for which there is no known efficient parallel algorithm that solves them. That is, there is no known parallel algorithm that is able to solve them in work that is within a logarithmic factor of the fastest sequential algorithm, where work is the product of the running time and the processor count. *Spencer[2]* develops parallel algorithms that solves the following problems presented in Karp-Ramachandran[1]:

- Directed Spanning Tree
- Breadth-First Search
- Topological Sort

- Cycle Detection for Directed Graphs
- Strong Connected Components
- Single Source Shortest Paths with Non-Negative Edges

His algorithms exhibit a time-work trade-off, that is, the longer they run, the less work they perform. This phenomenon exists due of the fact that certain speculative computations can be eliminated as time progresses, thus reducing work. One algorithm solves topological sort and the other algorithm solves breadth first search. It was mentioned in passing that the breadth first search algorithm can solve directed spanning trees and the topological sort algorithm can solve cycle detection for directed graphs. Furthermore, the breadth first search algorithm can be applied to strongly connected components and extended to single source shortest paths with non-negative edges.

/* Note: We may or may not need these definitions here. */
We next reproduce three definitions given in *Spencer[2]*:

Definition. A vertex v *reaches* a vertex u iff there is a path from v to u. Conversely, u is *reachable* from v iff there is path from v to u.

Definition. Let $G = (V, E)$ be a graph and let $R \subseteq V$. $G[R] = (R, E')$, where $E' \subseteq R \times R$.

Definition. Two vertices u, v are *identified* by replacing them with a single vertex uv. The outgoing edges of both u and v becomes the outgoing edges of uv and the incoming edges of both u and v become the incoming edges of uv.

2 Results and Techniques of Assigned paper

Highlight the major contributions and results of the assigned paper.

2.1 Techniques

Give an overview of the major techniques that were used in the paper to achieve these results.

3 Significant Related Results

Cite the three most significant related results. Here are some sample citations. Example: this paper [2].

Highlight the major contributions and results of these papers, and their relation to your assigned paper. Indicate which, if any, of these paperswere cited in your assigned paper. Also, you must cite the journal version of a conference paper, if the journal version has appeared.

Please note that while Google Scholar and other web tools are very useful in finding related results, they may not always give a pointer to the journal version or even the conference version. You need to locate and cite the latest version.

4 Presentation Topics

Here each team member should cite the paper from which they will present technical results in their presentation, and should indicate the exact theorem(s)/lemma(s) whose proof(s) they will present.

5 Further Research Directions

Give an overview of the current open problems and future research directions related to the work in the assigned paper.

References

[1] R. M. Karp and V. Ramachandran. A survey of parallel algorithm for shared-memory machines. *Handbook in Theoretical Computer Science*, pages 35–57, 1990.

[2] T. H. Spencer. Time-work tradeoffs for parallel algorithms. *J. ACM*, 44(5):742–778, Sept. 1997.