Fundamentals, Significance and Practicality of Treewidth and Tree Decompositions

Taylor Cox
Department of Computer Science
University of Manitoba
Winnipeg, Manitoba
Email: coxt3@myumanitoba.ca

***Abstract*—In 1987, Arnborg proved that it is NP-Complete to determine whether the treewidth of a graph is upper-bounded by some value k. However, Bodlaender proved in 1992 that the treewidth problem is in P in cases where k is fixed. Despite the fact that the treewidth problem is fixed-parameter tractable, the implementation of a general treewidth algorithm has been largely unsuccessful in the graph theory literature. The absence of a comprehensive, practical algorithm for determining the treewidth of a graph leads to the conclusion that fixed-parameter tractibility may not be a sufficiently descriptive characterization of the treewidth problem and related problems.**

***Index Terms*—Treewidth, Tree Decomposition, Fixed Parameter Tractibility**

I. Introduction

A key result of Robertson and Seymour's Graph Minor Theory is the notion of treewidth [1]. Let $G = (V, E)$ be an arbitrary graph. Informally, the treewidth of G $tw(G)$ is an indicator of how similar G is to a tree. Formally, $tw(G)$ is the minimum width across all possible tree decompositions of G. A tree decomposition is an organization of V into a collection of bags, where each bag contains one or more vertices and the union of all bags is equal to V. The bags are connected such that they form a tree. The width of a tree decomposition is the cardinality of its largest bag minus one. Treewidth-related definitions are further explored in section V.

Many NP-Complete and NP-Hard graph problems are solvable in polynomial time for graphs of bounded treewidth [2]. Such problems include Three-Colorability, Max-Clique and Minor Containment. The substantial theoretical significance of the treewidth property has motivated extensive research into approaches for determining the treewidth of a graph. For a general graph G of nonzero genus, it is NP-Complete to determine whether $tw(G) \leq k$ for some k. When limited to the planar graphs, is not known whether the treewidth problem is in P or NP [3]. Approximation, Heuristic, Dynamic Programming and Fixed-Parameter approaches have all been developed for the purpose of finding the treewidth for general graphs, with varying degrees of success [4].

This report evaluates two treewidth algorithms in terms of their programmatic and computational complexity. The first algorithm is Bodlaender's 1992 fixed-paramater treewidth algorithm [5], and is ultimately determined to be impractical for usage outside of theoretical computer science. The second algorithm explored is Bodlaender's 2012 dynamic programming algorithm [6]. A C implementation of this algorithm is benchmarked against the implementation provided in Sage, a standard mathematical software package [7]. Comparative results indicate a need to employ further optimizations in order to decrease the algorithm's runtime. However, the potential for improvement to the algorithm is limited due to its exponential nature. The Sage benchmarks themselves do not run in graphs of more than 100 vertices.

The remainder of this report is structured as follows: Section II describes the background to the treewidth problem, including relevant literature. Sections III and IV outline the motivation for implementing algorithms for treewidth and capture a formal description of the treewidth problem itself. In sections V through VII, implementations of Bodlaender's 1992 and 2012 algorithms will be described and critically evaluated. Finally, section

VIII will deliver the key conclusions of this project, accompanied by possible directions of future work from theoretical and implementation perspectives.

[1]
[2]
[3]
[4]
[5]
[6]
[7]