Jagiellonian University
Department of Theoretical Computer Science

Adrian Siwiec

Perfect Graph Recognition and Coloring

Master Thesis

Supervisor: dr inż. Krzysztof Turowski

September 2020

Abstract

TODO

Contents

1 Perfect Graphs

All graphs in this paper are finite, undirected and have no loops or parallel edges. We denote the chromatic number of graph G by $\chi(G)$ and the cardinality of the largest clique of G by $\omega(G)$. *Coloring* of a graph means assigning every node of a graph a color. A coloring is *valid* iff every two nodes sharing an edge have different colors. An *optimal* coloring (if exists) is a valid coloring using only $\omega(G)$ colors.

Given a graph $G = (V, E)$ and a set $X \subseteq V$ by $G[X]$ we will denote a graph induced on X. A graph $G = (V, E)$ is *perfect* iff for all $X \subseteq V$ we have $\chi(G[X]) = \omega(G[X])$.

Give some examples why are these interesting, some subclasses, and problems that are solvable for perfect graphs, including recognition and coloring

Given a graph G, its *complement* $\overline{G}$ is a graph with the same vertex set and in which two distinct nodes u, v are connected in $\overline{G}$ iff they are not connected in G. For example a clique in a graph becomes an independent set in its complement. A perfect graph theorem, first conjured by Berge in 1961 [Ber61] and then proven by Lovász in 1972 [Lov72] states that a graph is perfect iff its complement graph is also perfect.

Should we give some proof of that here?

1.1 Strong Perfect Graph Theorem

A *hole* is an induced chordless cycle of length at least 4. An *antihole* is an induced subgraph whose complement is a hole. A *Berge* graph is a graph with no holes or antiholes of odd length.

In 1961 Berge conjured that a graph is perfect iff it is Berge in what has become known as a strong perfect graph conjecture. In 2001 Chudnovsky et al. have proven it and published the proof in an over 150 pages long paper "The strong perfect graph theorem" [Chu+06].

2 Recognizing Berge Graphs

Cite the paper.

2.1 Recognition algorithm Overview

Recognizing simple structures (Diamonds, Jewels, T1, T2, T3).

Finding and Using Half-Cleaners.

Overview of proof of why algorithm using Half-Cleaners is correct.

2.2 Implementation

Anything interesting about algo/data structure?

Optimizations - Bottlenecks in performance (next path, are vectors distinct etc).

Validity tests - unit tests, tests of bigger parts, testing vs known answer and vs naive.

2.3 Parallelism with CUDA (?)

TODO

2.4 Experiments

Naive algorithm - brief description, bottlenecks optimizations (makes huge difference).

Description of tests used.

Results and Corollary - almost usable algorithm.

3 Coloring Berge Graphs

3.1 Ellipsoid method

Desctiption.

Implementation.

Experiments and results.

3.2 Combinatorial Method

Cite the paper.

On its complexity - point to appendix for pseudo-code.

Appendices

A Perfect Graph Coloring algorithm

TODO

References

[Ber61] C. Berge. "Färbung von Graphen, deren sämtliche beziehungsweise deren ungerade Kreise starr sind". In: Wissenschaftliche Zeitschrift der Martin-Luther-Universität Halle-Wittenberg, Mathematisch-naturwissenschaftliche Reihe, 1961, p. 114.

[Chu+06] Maria Chudnovsky et al. "The strong perfect graph theorem". In: *Annals of Mathematics* 164.1 (July 2006), pp. 51–229. DOI: `10.4007/annals.2006.164.51`. URL: `https://doi.org/10.4007/annals.2006.164.51`.

[Lov72] L. Lovász. "Normal hypergraphs and the perfect graph conjecture". In: *Discrete Mathematics* 2.3 (June 1972), pp. 253–267. DOI: `10.1016/0012-365x(72)90006-4`. URL: `https://doi.org/10.1016/0012-365x(72)90006-4`.