FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

A Hybrid Algorithm for the Partition Coloring Problem

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieurin

im Rahmen des Studiums

Computational Intelligence

eingereicht von

Gilbert Fritz
Matrikelnummer 0827276

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: Univ.Ass. Dipl.-Ing. Dr.techn. Dr. Bin Hu
Mitwirkung: Univ.-Prof. Dipl.-Ing. Dr.techn. Günther Raidl

Wien, 21.Oct.2013

(Unterschrift Verfasserin) (Unterschrift Betreuung)

Erklärung zur Verfassung der Arbeit

Gilbert Fritz
Schlosshofer Straße 49/18

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift Verfasserin)

Danksagung

Ich danke meinen Betreuern, ao. Univ.-Prof. Dipl.-Ing. Dr.techn. Günther Raidl und Univ.Ass. Dipl.-Ing. Dr.techn. Bin Hu für ihre Unterstützung bei der Erstellung dieser Arbeit durch ihr konstruktives Feedback und ihre Ideen, welche es mir ermöglicht haben, immer neue Aspekte der Problemstellung zu erkennen.

Besonderer Dank gilt meinen Eltern, Franz und Justine Fritz, sowie meiner Partnerin Odnoo und meinen Freunden für Ihre Unterstützung.

Abstract

Todo

Kurzfassung

Todo

Contents

CHAPTER 1

Preliminaries

This chapter introduces theoretical fundamentals like definitions, terms and methods, that are necessary for analysing the Partition Coloring Problem. The presented notations will be used consistently in the thesis. The reader is referenced to [] for a more detailed introduction into graph theory.

1.1 Graph Theory Definitions

Definition 1 (Graph) *A graph is a tuple $G = (V, E)$, where V denotes the set of nodes and $E \subseteq V \times V$ denotes the set of edges. An edge from node i to j is denoted by $\{i, j\}$. We call a graph simple, if it does not contain multiple edges, i.e. more than one edge between the same nodes, or loops, i.e. edges $\{i, i\}$.*

Definition 2 (Directed graph) *A directed graph or digraph is a tuple $D = (V, A)$, where V denotes the set of nodes and $A \subseteq V \times V$ denotes the set of arcs or directed edges. An arc from node i to j is denoted by (i, j). We call a directed graph simple, if it does not contain multiple arcs, i.e. more than one arc between the same nodes, or loops, i.e. arcs (i, i).*

COPYPASTE: When in the following it is clear from the context if we mean directed or undirected graphs or if it makes no difference, we will for simplicity just speak of graphs. Furthermore we consider only simple graphs, unless explicitly stated otherwise.

Definition 3 (Directed graph) *Given a graph $G = (V, E)$, $G' = (V', E')$ is called a subgraph if $V' \subseteq V$, $E' \subseteq E$ and $E' \subseteq V' \times V'$. If $V' = V$ we call G a spanning subgraph or factor.*

Definition 4 (Deletion of a node) *Given a graph $G = (V, E)$, $G - v = (V \setminus v, E \setminus \{e \mid v \in e\})$.*

Definition 5 (Adjacency and incidence) *Two nodes x and y are called adjacent, if they share an edge e, i.e. $\exists e = \{x, y\} \in E$. Two edges e and f are called adjacent, if they share a node x, i.e. $e \cap f = x$. A node v is called incident to an edge e, if $v \in e$.*

Definition 6 (Node degree) *The degree of a node v in an undirected graph G, denoted by $d(v)$, is the number of edges, that are incident to the node v, i.e. in E there exists an edge $\{v, x\}$. The number of outgoing arcs (v, x) from a node v in a directed graph D is called out-degree and is denoted by $d^+(v)$, the number of ingoing arcs (x, v) to a node v is called in-degree and is denoted by $d^-(v)$.*

Lemma 1 (Handshaking Lemma)

$$\sum_{v \in V} d(v) = 2\,|E|$$

Proof. As every edge $\{i, j\}$ is incident to exactly two nodes, namely i and j, it is counted one time at $d(i)$ and one time at $d(j)$. So the sum over all node degrees is exactly two times the number of edges. □

Corollary 1 (Directed graph) *The number of nodes with odd node degree is even.*

Proof. This immediately follows from the handshaking lemma. □

Lemma 2

$$\sum_{v \in V} d^+(v) = \sum v \in V d^-(v) = 2\,|A|$$

Proof. As every arc (i, j) has exactly one "in-node" and one "out-node", it follows, that the sum of all out-degrees equals the sum of all in-degrees and hence the number of arcs. □

Definition 7 (Maximum and minimum node degree) $\triangle(G) = max\{d(v) \mid v \in V\}$ *denotes the maximum node degree in a graph.* $\delta(G) = min\{d(v) \mid v \in V\}$ *denotes the minimum node degree in a graph.*

Definition 8 (Neighborhood of a node) *The neighborhood of a node $v \in V$ is denoted by $N(v) = \{x \mid \{v, x\} \in E\}$. In the directed case the neighborhood consists of all nodes that are reachable from v, i.e. $N(v) = \{x \mid (v, x) \in A\}$.*

Definition 9 (Walk) *A sequence $v_0, e_1, v_1, e_2, \ldots, e_n, v_n$ with $n \geq 0$ is called a walk, if for all v_i with $i \neq 0$ exists an $e_i = \{v_{i-1}, v_i\} \in E$.*

Definition 10 (Directed walk) *A sequence $v_0, a_1, v_1, a_2, \ldots, a_n, v_n$ with $n \geq 0$ is called a directed walk, if for all v_i with $i \neq 0$ exists an $a_i = (v_{i-1}, v_i) \in A$.*

Definition 11 (Trail) *A sequence $v_0, e_1, v_1, e_2, \ldots, e_n, v_n$ with $n \geq 0$ is called a trail, if for all v_i with $i \neq 0$ exists an $e_i = \{v_{i-1}, v_i\} \in E$ and all e_i are distinct.*

Definition 12 (Directed Trail) *A sequence $v_0, a_1, v_1, a_2, \ldots, a_n, v_n$ with $n \geq 0$ is called a directed trail, if for all v_i with $i \neq 0$ exists an $a_i = (v_{i-1}, v_i) \in A$ and all a_i are distinct.*

Definition 13 (Path) *A sequence $v_0, e_1, v_1, e_2, \ldots, e_n, v_n$ with $n \geq 0$ is called a path, if for all v_i with $i \neq 0$ exists an $e_i = \{v_{i-1}, v_i\} \in E$ and all v_i are distinct.*

Definition 14 (Directed path) *A sequence $v_0, a_1, v_1, a_2, \ldots, a_n, v_n$ with $n \geq 0$ is called a directed path, if for all v_i with $i \neq 0$ exists an $a_i = (v_{i-1}, v_i) \in A$ and all v_i are distinct.*

COPYPAST
When in the following it is clear from the context if we mean directed orundirected walks/trails/paths or if it makes no difference, we will for simplicity just speak of walks/trails/paths.

Definition 15 (Length of a path) *Given a path $P = v_0, e_1, v_1, e_2, \ldots, e_n, v_n = P(v_0, v_n)$, the length is the number of edges and denoted by $l(P) = n$, analogously for the directed case.*

Definition 16 (Cycle) *A cycle is a path, where $v_0 = v_n$.*

Definition 17 (Acyclic graph) *A graph is called acyclic if it does not contain a cycle.*

Theorem 1 *Let $W = W(v_0, v_n)$ be a walk, then there is a subsequence $P = P(v_0, v_n) \subseteq W(v_0, v_n)$ such that P is a path.*

Proof. We know that for a path holds $v_i = v_j \forall i < j$. Suppose for W holds that $v_i = v_j$ for arbitrary $i < j$. Then $W = v_0, e_1, v1, \ldots, v_i, e_{j+1}, v_{j+1}, \ldots, v_n$ is a walk with $i - j$ less edges and W is a subsequence of W. Applying this until $\forall i, j : v_i = v_j$ yields a path from v_0 to v_n. This of course also holds for the directed case. □

From Theorem **??** follows that dealing with paths suffices when we want to draw conclusions about a connection between two nodes.

Definition 18 (Network) *A network $N = (G, c)$ consists of a graph $G = (V, E)$ and a cost function $c : E(G) \longrightarrow \mathbb{R} \geq 0$, which assigns each edge e a nonnegative value c_e . Networks are also called weighted graphs.*

Definition 19 (Costs of a graph) *The cost c_G of a graph G is the sum of its edge costs, i.e. $c_G = \sum_{e \in E} c_e$.*

Definition 20 (Coloring) *A coloring is a mapping $v \longrightarrow \mathbb{R}, \forall v \in V$.*

Definition 21 (chromatic number) *Let $G = (V, E)$ be a graph. We state that c is a (proper) k-coloring of G if all the vertices in V are colored using k colors such that no two adjacent vertices have the same color. The chromatic number is defined as the minimum k for which there exists a (proper) k-coloring of G.*

1.2 Metaheuristics

Bibliography

[1] R. Andersen. Finding large and small dense subgraphs. *CoRR*, 0707032, 2007.

[2] M. Templ B. Meindl. Analysis of commercial and free and open source solvers for linear optimization problems. 2012.

[3] M.Gurusamy C. S. Ram Murthy. Wdm optical networks - concepts. *Design and Algorithms*, 2002.

[4] M. Charikar. Greedy approximation algorithms for finding dense components in a graph. *APPROX*, (84-95), 2000.

[5] C.Volko. Selective graph coloring problem. 2012.

[6] Yuri Frota, Nelson Maculan, Thiago F. Noronha, and Celso C. Ribeiro. A branch-and-cut algorithm for the partition coloring problem. *Networks*, 55(3):194–204, 2010.

[7] S.Pirkwieser G.Fritz, G.Raidl. Heuristic methods for the hop constrained survivable network design problem. 2011.

[8] R. M. Krishnaswamy and K. N. Sivarajan. Algorithms for routing and wavelength assignment based on solutions of lp-relaxations. *IEEE Communication Letters*, 5(10), 2001.

[9] E.L. Lawler. Combinatorial optimization: Networks and matroids. 1976.

[10] Guangzhi Li and Rahul Simha. The partition coloring problem and its application to wavelength routing and assignment. In *1st Workshop on Optical Networks*, 2000.

[11] B.Hu M.Leitner, G.Raidl. Solving two generalized network design problems with exact and heuristic methods. 2006.

[12] Thiago F. Noronha and Celso C. Ribeiro. Routing and wavelength assignment by partition colouring. *European Journal of Operational Research*, 171(3):797–810, 2006.

[13] S.Pirkwieser P.Gebhard, G.Raidl. The vehicle routing problem with compartments based on solutions of lp-relaxations. 2012.

[14] Petrica C. Pop, Bin Hu, and Günther R. Raidl. A memetic algorithm for the partition graph coloring problem. In *Extended Abstracts of the 14th International Conference on Computer Aided Systems Theory*, pages 167–169, Gran Canaria, Spain, 2013.

[15] B. Saha S. Khuller. On finding dense subgraphs. 2009.

[16] K. Jansen T. Erlebach. The complexity of path coloring and call scheduling. *Theoretical Computer Science*, 255, 2001.

[17] G. Kortsarz U. Feige and D. Peleg. The dense k-subgraph problem. *Algorithmica*, 29(410-421), 1997.

[18] H. Tamakic T. Tokuyamad Y. Asahiroa, K. Iwamab. Greedily finding a dense subgraph. *Journal of Algorithms*, 34, 2000.

[19] K.Iwama Y.Asahiro, R.Hassin. Complexity of ÿnding dense subgraphs. *Discrete Applied Mathematics*, 121(15–26), 2002.