On Paths and Trails in Edge-Colored Graphs and Digraphs

A thesis presented

by

Adria Ramos de Lyra

to the

Programa de Pós-graduação em Computação

in partial fulfillment of the requirements

for the degree of

Doctor in Computing

in the subject of

Combinatorial Optimization and Artificial Intelligence

Universidade Federal Fluminense

Niterói, Rio de Janeiro

Brazil

October 2009

On Paths and Trails in Edge-Colored Graphs and Digraphs

Adria Ramos de Lyra

Doctoral thesis submitted to the Programa de Pós-graduação em Computação of the Universidade Federal Fluminense in partial fulfillment of the requirements for the degree of Doctor in Computing.

Approved by:

D.Sc. Carlos Alberto de J. Martinhon / IC-UFF (Advisor)

D.Sc. Loana Tito Nogueira / IC-UFF

D.Sc. Fabio Protti / IC-UFF

D.Sc. Celina M. Herrera de Figueiredo / COPPE-UFRJ

D.Sc. Luerbio Faria / FFP-UERJ

Niterói, October 2009

Thesis advisor

Carlos Alberto de Jesus Martinhon

Author

Adria Ramos de Lyra

On Paths and Trails in Edge-Colored Graphs and Digraphs

Abstract

We deal with different algorithmic questions regarding properly edge-colored s-t paths/trails in edge-colored graphs and digraphs. Given a c-edge-colored graph G^c with no properly edge-colored closed trails, we present a polynomial time procedure for the determination of properly edge-colored s-t trails visiting all vertices of G^c a predefined number of times. As an immediate consequence, we polynomially solve the Hamiltonian path (resp., Eulerian trail) problem for this particular class of graphs. In addition, we prove that to check whether G^c contains 2 properly edge-colored s-t paths/trails with length at most $L > 0$ is **NP**-complete in the strong sense. Besides, we also show that if G^c is a general c-edge-colored graph, to find 2 monochromatic vertex disjoint s-t paths with different colors is **NP**-complete.

Regarding c-edge-colored digraphs, we show that the determination of a directed properly edge-colored s-t path is **NP**-complete in digraphs with $c = \Omega(n^2)$ colors. If the digraph is a c-edge-colored tournament, we show that deciding whether it contains a properly edge-colored circuit passing through a given vertex v (resp., directed s-t Hamiltonian path) is **NP**-complete. As a consequence, we solve a weak version of an open problem posed in [30]. In addition, we show that several problems are polynomial if we deal with directed properly edge-colored s-t trails instead of directed properly edge-colored s-t paths.

We also consider s-t paths, trails and walks with reload costs over c-edge-colored graphs. Each time a vertex is crossed by a walk there is an associated non-negative *reload* cost $r_{i,j}$, where i and j denote, respectively, the colors of successive edges in this walk. The goal is to find a route whose total reload cost is minimized. Polynomial algorithms and proofs of **NP**-hardness are given for particular cases: when the triangle inequality is satisfied or not, when reload costs are symmetric (*i.e.*, $r_{i,j} = r_{j,i}$) or asymmetric. We also investigate bounded degree graphs and planar graphs.

Keywords: Edge-colored graphs and digraphs; properly edge-colored paths/trails; monochromatic paths; edge-colored tournaments; reload optimization;

Sobre caminhos e trilhas em grafos e digrafos com cores nas arestas

Resumo

Neste trabalho, estuda-se diferentes questões sobre s-t caminhos e trilhas propriamente coloridos em grafos e digrafos com cores nas aretas. Dado G^c um grafo com c cores nas aretas sem trilhas fechadas propriamente coloridas, apresenta-se um procedimento polinomial para determinação de s-t trilhas propriamente coloridas que visitam todos os vértices de G^c um determinado número de vezes. Como consequência imediata, resolve-se polinomialmente o problema do caminho Hamiltoniano e Euleriano para esta classe particular de grafos. Além disso, prova-se que encontrar dois caminhos propriamente coloridos disjuntos por vértices ou arestas em G^c contendo no máximo $L > 0$ arestas é **NP**-completo forte. Também, mostra-se que achar dois caminhos monocromáticos disjuntos por vértices, com cores diferentes, em um grafo G^c qualquer é **NP**-completo.

Considerando digrafos com cores nas arestas, mostra-se que determinar um s-t caminho direcionado propriamente colorido é **NP**-completo mesmo para $c = \Omega(n^2)$. Se o digrafo for um torneio com cores nas arestas, mostra-se que decidir se este contém um circuito propriamente colorido passando por um vértice v (ou um caminho Hamiltoniano direcionado) é **NP**-completo. Como consequência, resolve-se uma versão mais fraca de um problema proposto em [30]. Além disso, considerando-se trilhas ao invés de caminhos, mostra-se que alguns problemas são polinomiais para s-t trilhas direcionadas propriamente coloridas.

Considera-se também s-t caminhos, trilhas e passeios em grafos coloridos com custos de conexão entre as aretas. Sempre que se muda de uma cor para outra em um passeio tem-se um custo de conexão $r_{i,j}$ associado, onde i e j são as cores das sucessivas arestas do passeio. O objetivo é encontrar uma rota cujo custo total de conexão seja minimizado. Algoritmos polinomiais e provas de **NP**-dificuldade são apresentados para casos particulares: quando a desigualdade triangular é satifeita ou não, quando os custos de conexões são simétricos (*i.e.*, $r_{i,j} = r_{j,i}$) ou assimétricos. Também são investigados instâncias com grau máximo limitado e grafos planares.

Palavras-chave: Grafos e digrafos com cores nas arestas; caminhos e trilhas monocromáticos e propriamente coloridos; Torneios com cores nas arestas; Otimização em grafos com custo de conexão entre as cores;

Contents

List of Figures

Acknowledgments

I first thank my Ph.D. advisor, Professor Carlos Martinhon, for supporting me during these past five years, for his warm encouragement and thoughtful guidance. Martinhon is a kind of person who is always in a good mood and ready to help. As a professional and disciplined researcher he thought me different ways to approach a scientific problem and the need to be persistent to accomplish any goal.

I also thank the Professors Jérôme Monnot and Laurent Gourvès for their insightful ideas and significant addition to this work, as well as Professor Fabio Protti for all his contributions, support and encouragement.

I am also in great debt with former professors: Professor Jayme Luiz Szwarcfiter, whose magic in classroom made graph theory a fascinating subject; Professor Luerbio Faria for all the stimulating discussions on graph theory; and Professor Luiz Satoru Ochi, who introduced me to the scientific world;

During the course of my studies many people on the staff of UFF encouraged me in various ways. I am especially grateful to Angela Regina Dias, Carlos Correa, Carlos Eduardo Cabral, Maria Freitas, Rafael Abreu, Teresa Cristina Cancela and Vera Elena da Silva.

I must also thank to the board of examiners: Professors Luerbio Faria, Celina Figueiredo, Loana Nogueira, Fabio Protti, and of course Professor Carlos Martinhon who made lots of comments and suggestions that improved the final thesis.

I thank CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) for their financial support during this course at UFF.

To the best friends ever: Luciana Brugiolo and Renatha Capua, many special thanks. Without them this past five years wouldn't be as pleasent as it was: the everyday lunches, the afternoon snacks, the occasional dinner, all the talking and the friendship helped me not to lose my mind and to keep the focus. I also have good friends that were of vital importance in this Ph.D. journey: Anand Subramanian, Diego Brandão, Fatima Marques, Jacques Silva, Janine Kniess, Jonivan Lisboa, Luidi Simonetti, Rosa Figueiredo and Stenio Soares. I'll never forget them. Either by listening or by helping with latex, linux, xfig and sometimes even English grammar doubts, all these friends supported me in one way or the other.

I am also very grateful to Eduardo Cirto and Rosana Costa, my coworkers and friends from CEFET-RJ, for being there every time I needed.

Last but not least, I thank my family: my parents, Maura and Carlos, for educating me, for all the encouragement to pursue my interests, and especially for their unconditional love. My brother Carlos for listening to my complaints and frustrations and for always believing in me, even when I doubted myself. Finally, I owe my warmest thanks to Carlos Eduardo Mello, my dear Kadu, that came all his way from Paris to Rio de Janeiro just to be by my side in the final stretch of my Ph.D.. Thanks for being with me in the best moments I've ever had (like the days we spent in Paris and the day of my Ph.D. defense) and for supporting me on the worst day I've had so far (the day I lost my grandmother Jamile). I hope we share more of these marvelous moments. And wish no more hard times, although I admit it is much more comfortable to face them whenever you are by my side.

Dedicated to my grandmother Jamile Lyra.

"We are what we repeatedly do. Excellence, then, is not an act, but a habit."

Aristotle

Chapter 1

Introduction

In the last few years a great number of applications have been modelled as problems in edge-colored graphs and digraphs. To solve them, we can explore some interesting connections between edge-colored graphs and the theory of cycles, paths and trails in directed and undirected graphs, matching theory, and other branches of graph theory [5]. For instance, problems in molecular biology correspond to extracting Hamiltonian or Eulerian paths or cycles colored in specified pattern [14, 15, 34, 35], transportation and connectivity problems where reload costs are associated with pair of colors at adjacent edges [19, 28, 41], social sciences [12], among others. Despite of their large application, a great number of works are restricted to 2-edge-colored graphs and digraphs, or other particular cases such as c-edge-colored complete graphs (for $c \geq 2$) [6, 9, 12, 13, 8] and c-edge-colored tournaments [30].

1.1 Notation and terminology

Let $I_c = \{1, 2, \ldots, c\}$ be a given set of colors with $c \geq 2$. In this work, G^c denotes a simple, *i.e.*, loopless and with no parallel edges, connected, non-oriented edge-colored graph containing two particular vertices s and t, where each edge has a color of I_c. In such case G^c is said to be a c-edge-colored graph.

We recall here some standard graph terminology: the vertex and edge sets of G^c are denoted by $V(G^c)$ and $E(G^c)$, respectively. The *order* of G^c is the number n of its vertices and the *size* of G^c is the number m of its edges. For c-edge-colored complete graphs of size n we write K_n^c instead of G^c. For a given color i, $E^i(G^c)$ denotes the set of edges of G^c colored by i. We denote by $N_{G^c}(x)$ the set of all neighbors of x in G^c, and by $N^i_{G^c}(x)$, the set of vertices of G^c, linked to x with edges colored by i. The degree of x in G^c is $d_{G^c}(x) = |N_{G^c}(x)|$ and the maximum degree of G^c, denoted by $\Delta(G^c)$, is $\Delta(G^c) = \max\{d_{G^c}(x) : x \in V(G^c)\}$. A non-oriented edge between two vertices x and y is denoted by xy while its color is denoted by $c(xy)$.

Similarly, given a c-edge-colored digraph D^c and two vertices $u, v \in V(D^c)$, we denote by $\vec{uv}$ an oriented edge or arc of $E(D^c)$ and its color by $c(\vec{xy})$. In addition, we define $N^+_{D^c}(x) = \{y \in V(D^c) : \vec{xy} \in E(D^c)\}$ the *out-neighborhood* of x in D^c ($d^+_{D^c}(x) = |N^+_{D^c}(x)|$ is the *out-degree* of x in D^c), $N^-_{D^c}(x) = \{y \in V(D^c) : \vec{yx} \in E(D^c)\}$ the *in-neighborhood* of x in D^c ($d^-_{D^c}(x) = |N^-_{D^c}(x)|$ is the *in-degree* of x in D^c) and $N_{D^c}(x) = N^+_{D^c}(x) \cup N^-_{D^c}(x)$ the *neighborhood* of $x \in V(D^c)$. We say that, T^c_n defines a *c-edge-colored tournament* with n vertices if it is obtained from K^c_n by choosing a direction for each colored edge.

Given a (non necessarily edge-colored) graph $G = (V, E)$, a *walk* ρ from s to t in G

(called s-t walk) is a sequence $\rho = (v_0, e_0, v_1, e_1, \ldots, e_k, v_{k+1})$ where $v_0 = s$, $v_{k+1} = t$ and $e_i = v_i v_{i+1}$ for $i = 0, \ldots, k$. A *trail* from s to t in G (called s-t trail) is a walk $\rho = (v_0, e_0, v_1, e_1, \ldots, e_k, v_{k+1})$ from s to t where $e_i \neq e_j$ for $i \neq j$. A *path* from s to t in G (called s-t path) is a trail $\rho = (v_0, e_0, v_1, e_1, \ldots, e_k, v_{k+1})$ from s to t where $v_i \neq v_j$ for $i \neq j$.

We will also recall the concept of contraction for non-oriented graphs. Given an induced subgraph Q of a non-colored graph G, a contraction of Q in G consists in replacing Q by a new vertex, say z_Q, so that each vertex x in $G - Q$ is connected to z_Q by an edge, if and only if, there exists an edge xy in G for some vertex y in Q.

Consider a $c \times c$ matrix $R = [r_{i,j}]$ (for $i, j \in I_c$) whose entries define *reload costs* (or *connection costs*) when going from an edge colored i to another edge colored j. It is assumed that each entry $r_{i,j}$ of R is a non-negative integer (*i.e.*, $r_{i,j} \in \mathbb{N}$). Here, we will both consider symmetric and asymmetric matrices. We say that a matrix R satisfies the *triangle inequality*, if and only if, for all edges $e_i, e_j, e_k \in E(G^c)$ which are adjacent to a common vertex, we have $r_{c(e_i),c(e_j)} \leq r_{c(e_i),c(e_k)} + r_{c(e_k),c(e_j)}$ (see [19, 41]). Given a path/trail/walk $\rho = (v_0, e_0, v_1, e_1, \ldots, e_k, v_{k+1})$ between vertices s and t, we define the reload cost of ρ as:

$$r(\rho) = \sum_{j=0}^{k-1} r_{c(e_j),c(e_{j+1})} \tag{1.1}$$

The *length* of the path, trail or walk ρ in G^c (resp., D^c), denoted by $|\rho|$, is the number of its edges (resp., arcs).

An instance of the *minimum reload s-t path/trail/walk* problem consists of a simple connected c-edge-colored graph G^c, a pair $s, t \in V(G^c)$ and a $c \times c$ matrix $R =$

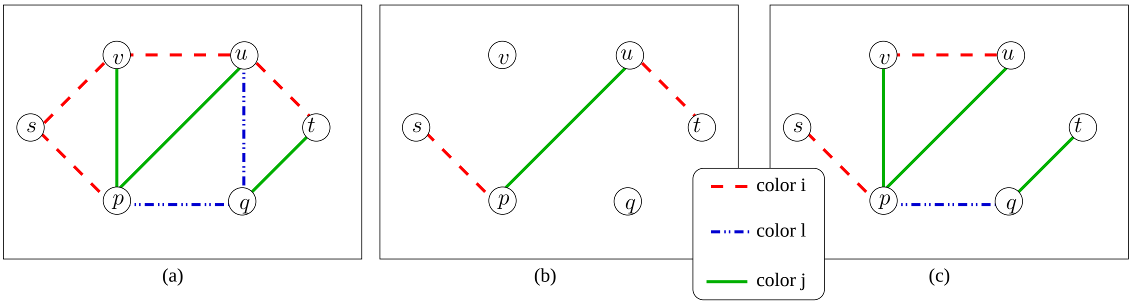


Figure 1.1: (a) 3-colored graph. (b) A PEC path and (c) a PEC trail associated.

$[r_{i,j}]$ associating a non-negative cost to each pair of colors. The objective is to find a path/trail/walk ρ from s to t with minimum reload cost. For instance, in the *Minimum Toll Cost s-t Path* problem, $r_{i,j} = r_j$ for $i, j \in I_c$ with $i \neq j$ and $r_{i,i} = 0$. Where the edges represent roads and every r_j is a non-negative integer that represents a cost that must be paid each time we change from a road to another. The objective is to find a path minimizing the cost of going from a source to a destination. Finally, notice that if $c = 1$ (*i.e.*, there is only one color in G^c), these problems are equivalent to finding an s-t path of minimum length in G^c. Thus, we will assume $c \geq 2$.

From now on, we write PEC instead of *properly edge-colored*. A PEC *path* (resp., PEC *trail*) is a path (resp., trail) such that any two consecutive edges have different colors, see Figure1.1. A PEC path or trail in G^c is *closed* if its end-vertices coincide and its first and last edges differ in color. They are also referred, respectively, as PEC *cycles* and PEC *closed trails*. However, if we deal with edge-colored digraphs, they are denoted, respectively, by PEC *circuits* and *directed* PEC *closed trails*. In the same way, a *monochromatic path/trail* is the path/trail whose all edges have the same color. We say that two or more s-t paths/trails are pairwise vertex (resp., edges) disjoint if they do not have a vertex (resp., edge) in common.

1.1.1 The gap reduction technique

We will also deal with some inapproximability results for the reload problems presented here. For that, we use the *gap reduction* technique. The idea is to use a problem ϕ and its gap version $\phi_{g(n)}$ to prove that if $\phi_{g(n)}$ is **NP**-hard, then it is **NP**-hard to obtain a worst-case approximation ratio for the optimization problem ϕ. Without loss of generality, suppose that ϕ is a minimization problem. The following definition can be made for maximization problems, as well. Formally:

For a (minimization) problem ϕ its gap version problem $\phi_{g(n)}$ and some function $h(n)$, we have:

- The YES instances are instances I of ϕ such that $OPT(I) \leq h(n)$ and
- The NO instances are instances I of ϕ such that $OPT(I) \geq g(n)h(n)$

The function $g(n) \geq 1$ is called *gap*. Now, suppose there is a polynomial time reduction from a **NP**-complete decision problem ϕ' to $\phi_{g(n)}$ such that the YES (resp. NO) instances of ϕ' are mapped to YES (resp. NO) instances of $\phi_{g(n)}$. Then $g(n)$-approximation algorithm for ϕ, if exists, can be used to decide the **NP**-complete decision problem ϕ' in polynomial time. It follows that it is NP-hard to approximate ϕ within a factor $g(n)$. This is typical way of proving inapproximability results.

As an example, to illustrate the previous definition, there is a very simple application of the gap technique. In the polynomial reduction from an instance G of the Hamiltonian Cycle problem (HC) to an instance G' of the Travelling Salesman Problem (TSP), one can set all the edges of G with weights 1, and the missing edges with weights 2 to construct G'. Observe that G' is a complete weighted graph with costs 1 and 2. Any valid Hamiltonian cycle for G in G' has cost n. An invalid tour will have at least a cost $n+1$. So it is **NP**-complete to distinguish between $OPT = n$ and $OPT = n+1$. In the same way, one can increase the size of the gap by replacing the distances of 2 by some exponential, *e.g.*, $n2^n$. Then, tours that do not come from a valid HC in the graph G have cost at least $n2^n + n - 1$ for the TSP in G'. So there is no polynomial time algorithm with a worst-case approximation ratio of 2^n.

1.2 Some related work

The determination of PEC s-t paths was polynomially solved for general graphs by Edmonds for two colors (see Lemma 1.1 in [32]) and then extended by Szeider[38] to include any number of colors. In Abouelaoualim *et al.*[1], the authors also deal with PEC trails and present polynomial time procedures for several versions of the s-t path/trail problem, such as the shortest PEC s-t path/trail on general c-edge-colored graphs and the longest PEC s-t path (resp., trail) for graphs with no PEC cycles (resp., closed trails). A characterization of c-edge-colored graphs containing PEC cycles was first presented by Yeo [42] and generalized in [1] for PEC closed trails. In addition, Abouelaoualim *et al.* in [1] prove that deciding whether there exist k pairwise vertex/edge disjoint PEC s-t paths/trails in a c-edge-colored graph G^c is **NP**-complete even for $k = 2$ and $c = \Omega(n^2)$ (for $c \geq 2$). Moreover, they prove that these problems remain **NP**-complete for c-edge-colored graphs containing no PEC closed trails and $c = \Omega(n)$. They describe a greedy procedure for the *Maximum Properly Edge Disjoint s-t Trail* - MPEDT (resp., *Maximum Properly Vertex Disjoint s-t Path* - MPVDP) problem, whose objective is to maximize the number of edge disjoint (resp., vertex disjoint) PEC trails (resp., paths) between s and t. They prove a $O(\frac{1}{\sqrt{m}})$ (resp., $O(\frac{1}{\sqrt{n}})$) performance ratio for the MPEDT problem (resp., MPVDP problem). Finally, they show how to polynomially solve the MPEDT problem (resp., MPVDP problem) over c-edge-colored graphs with no PEC closed trails or almost PEC closed trails (resp., PEC cycles or almost PEC cycles) [1] through s or t. We say that a closed trail (resp., cycle) with vertices $c_x = xa_1...a_jx$ and with $x \neq a_i$ for $i = 1, ..., j$ is an almost PEC closed trail (resp. cycle) through x to in G^c, if and only if, $c(xa_1) = c(xa_j)$ and both trails (resp., paths) from x to a_1 and x to a_j are PEC .

In Abouelaoualim *et al.* [2], the authors give sufficient degree conditions for the existence of PEC cycle and paths in edge-colored graphs, multigraphs and random graphs. In particular, they prove that an edge-colored multigraph of order n with at least 3 colors and with minimum color degree greater or equal to $\lceil \frac{n+1}{2} \rceil$ has PEC cycles of all possible lengths, including Hamiltonian cycles.

Concerning monochromatic results, they were exploited in c-edge-colored digraphs or bipartite tournaments. In [21], Sánchez and Monroy proved that if D^c is an c-colored bipartite tournament such that every directed cycle of length 4 is monochromatic, then D^c has a kernel by monochromatic paths. Besides, in [22], they present a method to construct a large variety of c-colored digraphs D^c with (resp. without a kernel) kernel by monochromatic paths; starting with a given c-colored digraph D_0^c.

A set $N \subseteq V(D^c)$ is said to be a kernel by monochromatic paths if it satisfies the following two conditions:

1. For every pair of different vertices $u, v \in N$ there is no monochromatic directed

[1] We say that we have an almost PEC closed trail (resp., almost PEC cycle) through a vertex x if both edges adjacent to x in this closed trail (resp., cycle) have the same color.

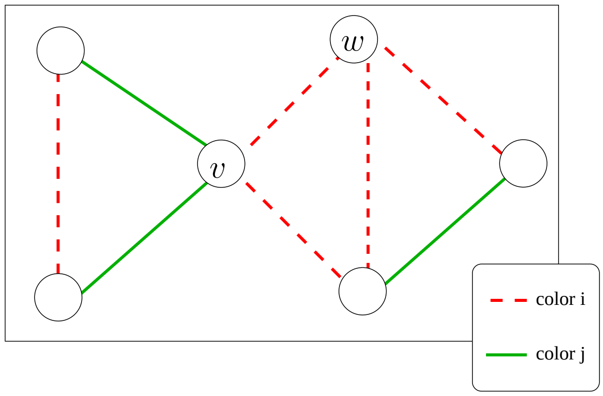


Figure 1.2: Vertex v and w are cut-vertices separating colors.

path between them.

2. For every vertex $x \in (V(D^c)\backslash N)$ there is a vertex $y \in N$ such that there is an x-y monochromatic directed path.

From our knowledge, most of the studies deal with kernel by monochromatic paths, see [33, 21, 22, 20].

The following results from the literature concerning c-edge-colored graphs will be used in this work. Firstly, by using the concept of cut-vertex separating colors we have the following result of Yeo [42] that allows us to decide whether a undirected c-edge-colored graph contains or not a PEC cycle. We say that a vertex v of G^c is a *cut-vertex separating colors*, if and only if, no component of $G^c - v$ is joined to v by at least two edges in different colors (See Figure 1.2). This theorem was generalized by Abouelaoualim *et al.*[1] to deal with the existence and search of PEC closed trails.

Theorem 1. *(Yeo, 1997) Let G^c be a c-edge-colored graph, $c \geq 2$, such that every vertex of G^c is incident with at least two edges colored differently. Then either G^c has a* PEC *cycle or G^c has a cut-vertex separating colors.*

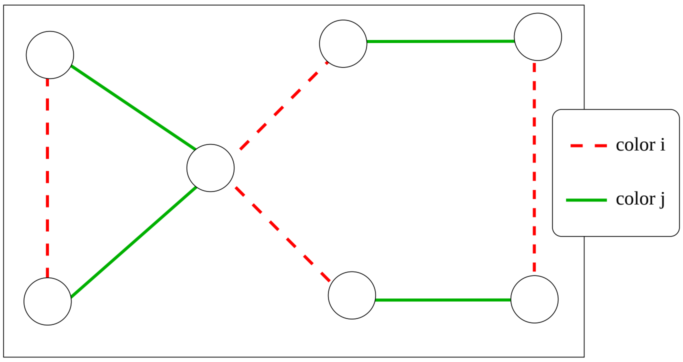


Figure 1.3: The graph G^c above does not contain a PEC cycle, however it contains a PEC closed trail.

Theorem 2. *(Abouelaoualim et al., 2008) Let G^c be a c-edge-colored graph, such that every vertex of G^c is incident with at least two edges colored differently. Then either G^c has a bridge or G^c has a* PEC *closed trail.*

As an immediate consequence of the Theorem 1 (resp., 2), the existence of a PEC cycle (resp., closed trail) in G^c may be checked in polynomial time. To see that it suffices to delete all cut-vertex separating colors (resp., bridges and vertices incident to edges of the same color in G^c). If the resulting set of edges is non-empty, then G^c contains a PEC cycle (resp., PEC closed trail). Recall that a *bridge* is an edge whose deletion increases the number of connected components of the original graph (See the example of Figure 1.3). Note that all such edges and vertices may be deleted without destroying any PEC cycle (resp., closed trail).

We will also use the following theorem from Szeider [38] for determining a PEC s-t path in G^c (if any).

Theorem 3. *(Szeider, 2003) Let s and t be two vertices in a c-edge-colored graph G^c, $c \geq 2$. Then, either we can find a* PEC *s-t path or else decide that such a path*

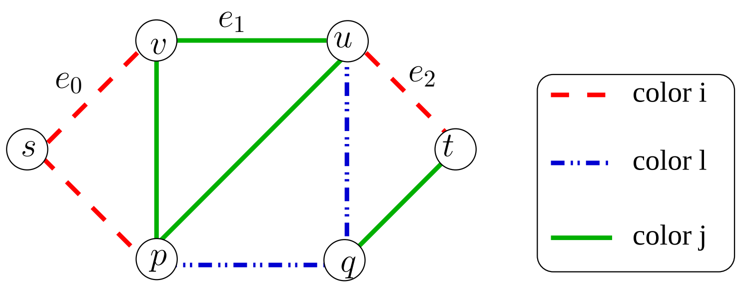


Figure 1.4: A 3-edge-colored graph we used as example for Szeider's Algorithm.

does not exist in G^c in linear time on the size of the graph.

Prior to explain this algorithm, let us first consider the following definitions. Given a graph $G = (V, E)$, a *matching* M in G is a set of pairwise non-adjacent edges, so that, no two edges share a common vertex. We say that M is *perfect*, when it matches all vertices of the graph. A *maximum matching* is a matching that contains the largest possible number of edges.

Essentially, the idea in the Szeider's algorithm is to reduce the PEC s-t path problem in G^c to a matching problem in a non-colored graph G defined as follows. Given G^c, $s, t \in V(G^c)$, set $W = V(G^c) \setminus \{s, t\}$. For every $x \in W$ define a subgraph G_x, where,

$V(G_x) = \bigcup_{i \in I_c} \{x_i, x'_i \mid N^i_{G^c}(x) \neq \emptyset\} \cup \{x''_a, x''_b\}$ and

$E(G_x) = \{x''_a x''_b\} \cup (\bigcup_{\{i \in I_c | x'_i \in V(G_x)\}} (\{x_i x'_i\} \cup (\bigcup_{j=a,b} \{x'_i x''_j\})))$.

The former graph will be called *Edmonds-Szeider graph* as in [1] and is constructed as follows:

$V(G) = \{s', t'\} \cup (\bigcup_{x \in W} V(G_x))$

$E(G) = \{\bigcup_{i \in I_c} \{s' x_i \mid sx \in E^i(G^c)\} \cup \{x_i t' \mid xt \in E^i(G^c)\} \cup \{x_i y_i \mid xy \in E^i(G^c)\}\} \cup \{\bigcup_{x \in W} E(G_x)\}$.

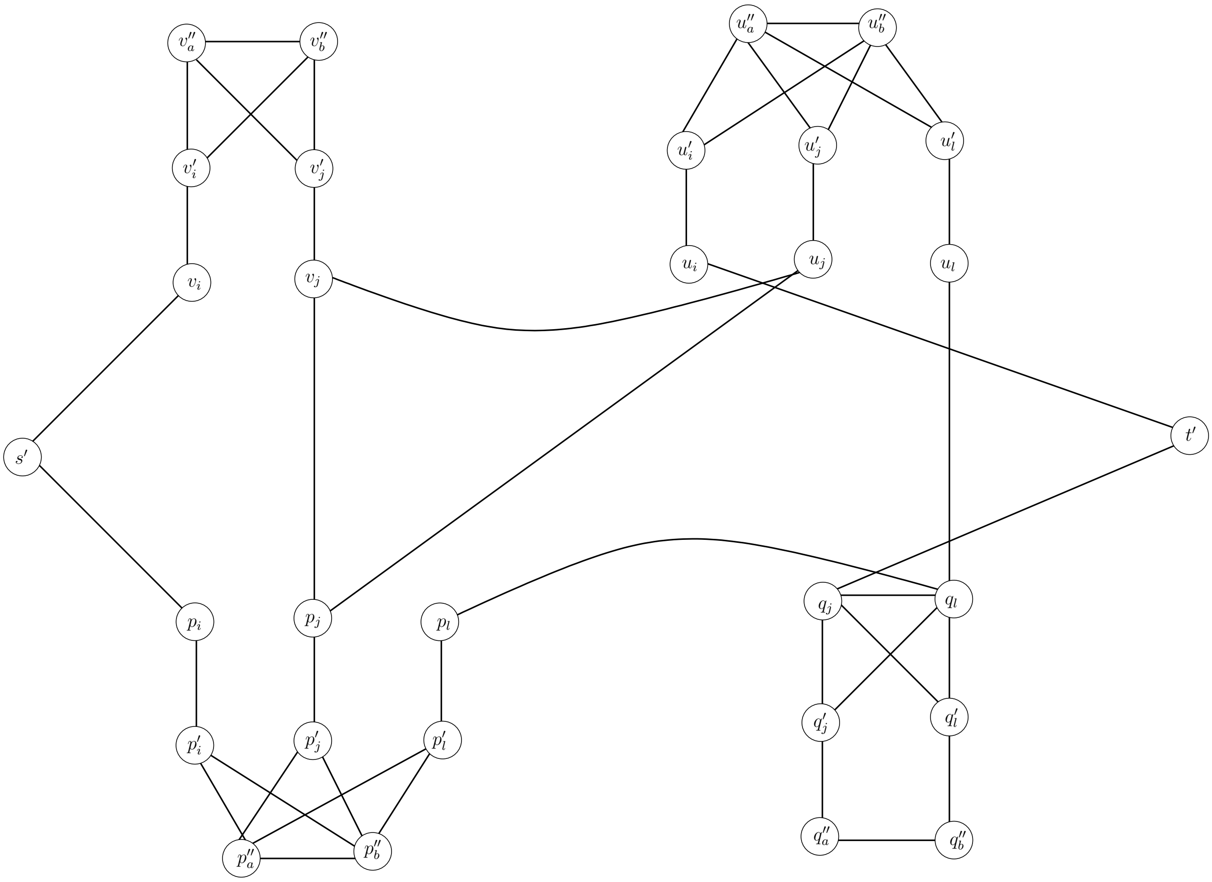


Figure 1.5: A non-colored graph G associated with the graph of Figure 1.4

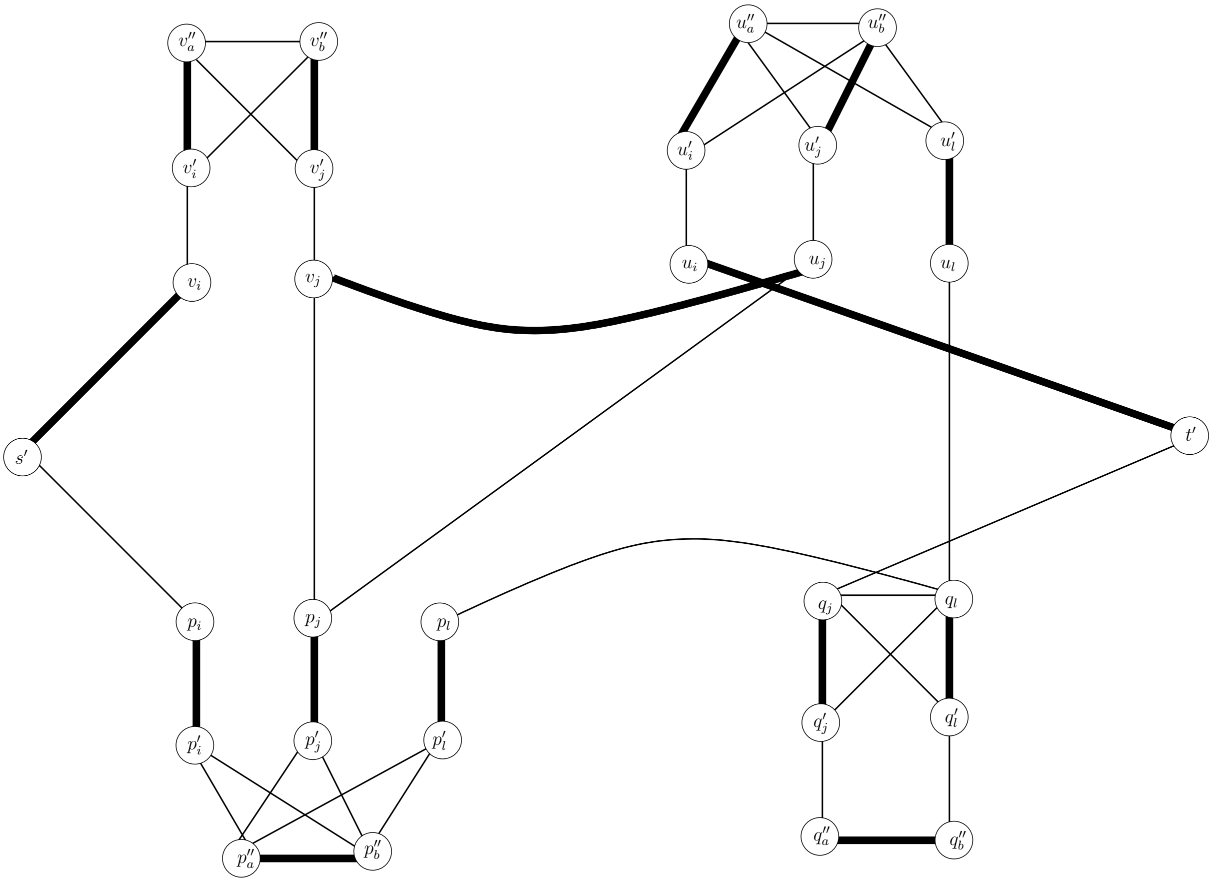


Figure 1.6: A perfect matching M in G

See Figure 1.5 for the non-colored graph associated with the edge-colored graph of Figure 1.4. In Figure 1.6 the bold edges correspond to the edges of a perfect matching M. The path ρ in G^c associated with M is $\rho = (s, e_0, v, e_1, u, e_2, t)$ (See Figure 1.4). Note, for instance, that for all vertices $x \in V(G^c)$ not belonging to the PEC path ρ, we have $x''_a x''_b \in M$ and reciprocally, whenever $x''_a x''_b \notin M$ at some gadget G_x in G, we have x belonging to the edge-colored path ρ in G^c. Further, for every $uv \in E^i(G^c)$ in a edge-colored path ρ, we have $u_i v_i \in M$ in $E(G)$.

Given a perfect matching M in $G - \{s', t'\}$ a PEC s-t path exists in G^c, if and only if, there is an augmenting path P associated with M between s' and t' in G. Note that a path P is augmenting with respect to a given matching M if for any pair of adjacent edges in P, exactly one of them is in M, with the further condition that the first and the last edges of P are not in M. Observe that augmenting paths in G can be found in $O(|E(G)|)$ linear time, see Tarjan's book [39]. The path ρ in G^c was obtained after contracting all subgraphs G_x in G, for every $x \in W$.

We will also deal with an important definition introduced in Abouelaoualim *et al.* [1]. Given an edge-colored graph G^c and an integer $p \geq 2$, a new edge-colored graph denoted by $p-H^c$ (called *trail-path graph*) is obtained from G^c as follows. Each vertex x of G^c will be replaced by p new vertices $x_1, x_2, ..., x_p$. Moreover, for any edge xy of G^c colored by j, for instance, add two new vertices v_{xy} and u_{xy}, add the edges $x_i v_{xy}$, $u_{xy} y_i$, for $i = 1, 2, ..., p$ all of them colored by j, and finally add the edge $v_{xy} u_{xy}$ in a new unused color $j' \in \{1, 2, ..., c\}$ with $j' \neq j$. The edge-colored subgraph of $p - H^c$ induced by the vertices x_i, v_{xy}, u_{xy}, y_i is associated with the edge xy of G^c and is denoted by H^c_{xy}. If $p = 2$, the subgraph $p - H^c$ is represented simply by H^c,

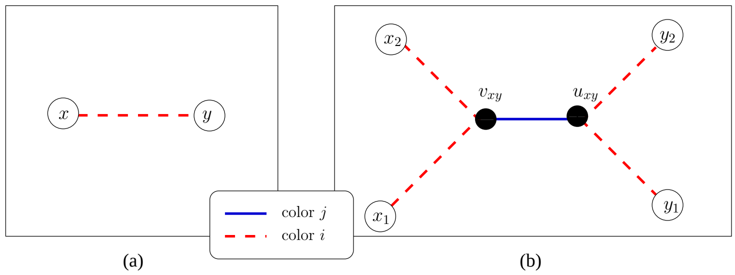


Figure 1.7: Edge $xy \in E^i(G^c)$ (a). Subgraph H^c_{xy} associated with $xy \in E^i(G^c)$ (b).

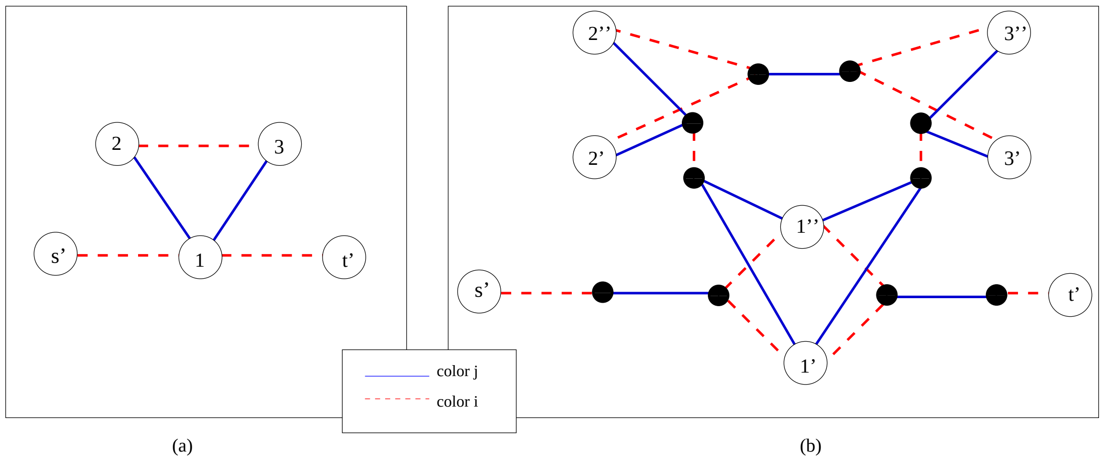


Figure 1.8: Transformation of the s-t trail problem into the s-t path problem.

see Figure 1.7.

Using the concept of trail-path graph, the authors in [1] extend Szeider's Algorithm to deal with s-t trails in G^c. The authors show that finding PEC s-t paths in p-H^c (for some p) is equivalent to find PEC $s-t$ trails in G^c.

See Figures 1.8.(a) and (b) for an example of a 2-colored graph G^c which contains a unique s-t trail and its associated trail-path graph H^c. Note that s-t trails in G^c are associated with s'-t' paths in H^c and vice verse. In order to use the Szeider's Algorithm to find a PEC trail in G^c, Abouelaoualim *et al* [1] first construct the associated trail-path graph p-H^c for $p = \lfloor \frac{(n-1)}{2} \rfloor$ (maximum possible number of visits at $x \in V(G^c) \backslash \{s, t\}$ at an arbitrary s-t trail). Now by using p-H^c, they construct its associated non-colored Edmonds-Szeider graph G and find a perfect matching M in G (if any). Thus, the problem of finding a PEC s-t trail in G^c (provided that one exists) can be solved in polynomial time. In [1], if we are looking for a shortest PEC s-t trail, it suffices to fix $p = 2$.

Concerning reload cost optimization, to the best of our knowledge, it has been mainly studied in the context of spanning trees [19, 23, 24, 41], but also very recently for some variants of paths, tours and flow problems [3]. In [41], the authors consider the problem of finding a spanning tree of minimum diameter with respect to the reload costs and they propose inapproximability results for graphs of maximum degree 5 and polynomial results for graphs of maximum degree 3. In [19], the author discusses inapproximability results for the same problem when restricted to graphs with maximum degree 4. In [23, 24], the authors give several formulations with computational results to solve the reload cost spanning tree problem.

Despite the importance in telecommunications and transportation industry, reload costs have not been extensively studied in the literature. In [41, 19], each color is viewed as a subnetwork and is used to model a cargo transportation network which uses different means of transportation or data transmission costs arising in large communication networks. In all these models, the transportation or communication costs between the subnetworks usually dominate the costs within individual subnetworks. Some applications in satellite networks are also discussed in [23] where the various subnetworks may represent different products offered by the commercial satellite service providers. In [23], terrestrial satellite dishes are required to first capture the radio signals and then special electric-to-fiber converters are required to transform the electric signals from the satellite dishes to optical pulses that can be sent over optical fibers. These interface costs are referred to as reload costs and depend on the technologies being connected. As another example, imagine a road network with many tolls. A fee (reload cost) must be paid each time we change from one road to another. One may be interested in paying as little as possible to travel from a source to a destination. We call this problem the *minimum toll cost s-t* path problem.

Amaldi *et al* [3] consider several models for paths, tours and flow problems with reload costs. As discussed above, consider a scenario in which a transportation network is divided in subnetworks, such that transportation costs are negligible within each subnetwork, but are significant when moving from one subnetwork to another. This scenario fits networks which use different means of transportation, like overlay networks, *i.e.*, networks where there is a change of technology used, or peer-to-peer telecommunication networks, and in general complex telecommunication networks. For instance, in overlay networks the costs may be related to the change of technology, in a cargo transportation network to unloading and reloading goods at different junctions, in large communication networks to data conversion at interchange points, etc. In this scenario the costs at the interchange points between the subnetworks usually dominate the costs within individual subnetworks. In particular, Amaldi *et al* [3] study the following model: given a directed edge-colored graph $D^c = (V, \overrightarrow{E})$ where each arc (or edge) $e \in \overrightarrow{E}$ has a non-negative cost $w(e)$ and a color $c(e) \in I_c$, and given a non-negative integer reload cost matrix $R = [r_{i,j}]$ for

$i, j \in I_c$, they want to find an oriented s-t trail $\rho = (s, e_1, v_1, e_2, \ldots, e_k, t)$ of D^c minimizing $\sum_{i=1}^{k} w(e_i) + \sum_{i=1}^{k-1} r_{c(e_i),c(e_{i+1})}$. In [3], they prove that this problem, called *the minimum reload+weight directed s-t trail problem*, is solvable in polynomial time.

The minimum reload s-t path (resp., trail) problem is also related to the problem of deciding whether a simple connected edge-colored graph G^c has a PEC s-t path (resp., s-t trail) or a monochromatic s-t path. For instance, if we set for the reload cost $r_{i,i} = 1$ and $r_{i,j} = 0$ for $i, j \in I_c$ with $i \neq j$, then there exists an s-t path (resp., s-t trail) with reload cost 0 in G^c, if and only if, G^c has a PEC s-t path (resp., trail). Analogously, if we are looking for monochromatic s-t paths in G^c, it suffices to set $r_{i,i} = 0$ and $r_{i,j} = 1$ for $i, j \in I_c$ with $i \neq j$.

1.3 Our Contributions

In Chapter 2 we study c-edge-colored graphs G^c with no PEC closed trails. We prove that checking whether G^c (with no PEC closed trails) contains two vertex/edge disjoint PEC s-t paths, each having at most $L > 0$ edges, is **NP**-complete in the strong sense. We conclude the section by presenting a polynomial time procedure for the determination of a PEC s-t trail (if one exists) visiting all vertices of G^c a predefined number of times. Using this result, we polynomially solve the PEC Hamiltonian path and PEC Eulerian trail problems for this particular class of graphs. Recall that, given a graph $G = (V, E)$, a Hamiltonian (resp., Eulerian) path is a path which visits each vertex of V (resp., edge of E) exactly once [11]. We conclude the chapter by studying polynomial and **NP**-completeness results regarding s-t paths and trails c-edge-colored graphs with no PEC cycles (note in this case that PEC closed trails are allowed)

In Chapter 3, we deal with monochromatic s-t paths in edge-colored graphs. We show that the problem of finding 2 vertex disjoint monochromatic paths with different colors between s and t is **NP**-complete. The **NP**-completeness of the directed monochromatic case follows as an immediate consequence.

In Chapter 4, we deal with c-edge-colored digraphs. We show that determining a directed PEC s-t path is **NP**-complete even if D^c is a planar c-edge-colored digraph with no PEC circuits or if D^c defines a 2-edge-colored tournament. We also prove that deciding whether a c-edge-colored tournament has a directed PEC Hamiltonian s-t path is **NP**-complete. Notice that there is no reduction between deciding whether a c-edge-colored tournament possesses a directed PEC s-t path and a directed PEC Hamiltonian s-t path, although finding a directed PEC s-t path seems to be an easier task than finding a directed PEC Hamiltonian path. As a consequence, we also show that deciding whether a 2-edge-colored tournament contains a PEC circuit passing through a given vertex v is **NP**-complete (this solves a weak version of an open problem initially posed by Gutin, Sudakov and Yeo [30]), which can be formulated as follows: does there exist a polynomial algorithm to check whether a 2-edge-colored tournament has a PEC cycle? In addition, we prove that the problem of maximizing the number of directed edge disjoint PEC s-t trails can be solved within polynomial time.

In Chapter 5, we present Reload Cost Problems. This chapter is organized as follows. In Section 5.1, we discuss the case of finding a minimum reload s-t walk, either with symmetric or asymmetric reload cost matrix. In Section 5.2 we deal with paths and trails when reload costs are symmetric. We prove that the minimum reload s-t trail problem can be solved in polynomial time for every $c \geq 2$. In addition, we show that the minimum reload s-t path problem is polynomially solvable either if $c = 2$ and the triangle inequality holds (here R is not necessarily a symmetric matrix) or if G^c has a maximum degree 3. However it is **NP**-hard when $c \geq 3$, even for graphs of maximum degree 4 and reload cost matrix satisfying the triangle inequality. We conclude by showing that, if $c \geq 4$ and the triangle inequality is satisfied, the minimum symmetric reload s-t path problem remains **NP**-hard even for planar graphs with maximum degree 4. In Subsection 5.2.1, we investigate a version of the traveling salesman problem with reload costs. In particular we show that the problem is **NP**-hard and no non-trivial approximation is likely to exist. Note that, given a graph $G = (V, E)$, with distances associated with the edges of E the goal of the Traveling Salesman Problem is to find the shortest tour that visits all the vertices of V exactly once [4]. Recall that a tour is a path that starts and ends with the same vertex. Finally, in Section 5.3 we deal with asymmetric reload costs. For a reload cost matrix satisfying the triangle inequality, we construct a polynomial time procedure for the minimum reload s-t trail problem and we prove that the minimum asymmetric reload s-t trail problem is **NP**-hard even for graphs with 3 colors and maximum degree equal to 3.

At the end of each chapter, we present some related open problems. Finally, some concluding remarks and future directions are given in Chapter 6.

Until now, this work generated the following publications: results presented in Chapters 2 and 3 were accepted for presentation at LAGOS 2009 [29]. This work was a joint collaboration with Professors Jérôme Monnot, Laurent Gourvès and Fabio

Protti. The results of Chapter 4, regarding c-edge-colored digraphs were published in a technical report [27] and is yet to be submitted for publication in some international journal. Chapter 5 were presented at SOFSEM 2009 [28] and submitted for publication in Discrete Applied Mathematics. These previous works were a joint collaboration with Professors Jérôme Monnot and Laurent Gourvès.

Chapter 2

Paths and trails in edge-colored graphs with no PEC closed trails

In this chapter, we deal with several questions regarding c-edge-colored (undirected) graphs G^c with no PEC closed trails and $c \geq 2$. Initially, we show that deciding whether or not G^c contains two vertex/edge disjoint PEC s-t paths with bounded length is **NP**-complete in the strong sense. In addition, when restricted to this particular class of graphs, we show that the determination of a PEC s-t trail visiting vertices a predefined number of times can be solved in polynomial time. We also deal with s-t paths and trails in graphs with no PEC cycles (note in this case that PEC closed trails are allowed). We conclude the chapter by proposing some related open problems and future directions.

2.1 Finding two vertex/edge disjoint PEC s-t paths with bounded length in graphs with no PEC closed trails

It is proved in Abouelaoualim *et al.* [1] that deciding whether an arbitrary c-edge-colored graph on n vertices (even with $\Omega(n^2)$ colors) contains two vertex/edge disjoint PEC s-t paths is **NP**-complete. However the complexity of this problem for graphs with no PEC closed trails is an open problem raised in this same work. Here, we propose and solve a weaker version of this problem. Given a graph G^c ($c \geq 2$) with no PEC closed trails and a constant $L > 0$, we prove that deciding whether G^c contains two vertex/edge disjoint PEC s-t paths, each having at most L edges is **NP**-complete.

This problem is interesting because it models a problem of telecommunication networks. As discussed in Itai *et al.* [31], bounding the length of a longest path ensures that the noise interference is under control. They show that the weighted 2 edge disjoint directed s-t paths problem is (weakly) **NP**-complete [31] (actually, their proof can be easily modified to handle directed acyclic graphs). However, as pointed out by Tragoudas and Varol [40], the authors in [31] consider a more general graph instance where the edge lengths are not polynomially bounded in the input size. In Tragoudas and Varol [40], they show how to solve this problem and present a proof (for the undirected case) where the edges weights are polynomially bounded in the input size.

We studied the result presented in Theorem 4 and show that the problem we deal with in this section is strong **NP**-complete. Thus, we have the following result:

Theorem 4. *Let G^c be a 2-edge-colored graph with no* PEC *closed trails and a constant $L > 0$. The problem of finding 2 vertex/edge disjoint* PEC *s-t paths, each having at most L edges in G^c is **NP**-complete, even for graphs with maximum vertex degree equal to 4.*

Proof: Suppose that $I_c = \{1, 2\}$. The vertex-disjoint case follows immediately from the edge-disjoint case so its proof is omitted. We prove that $(3, B2)$-SAT, called the 2-balanced 3-SAT, can be polynomially reduced to our problem. An instance $\mathcal{I}$ of $(3, B2)$-SAT consists of n variables $\mathcal{X} = \{x_1, \ldots, x_n\}$ and m clauses $\mathcal{C} = \{c_1, \ldots, c_m\}$. Each clause has exactly three literals. Each variable appears four times, twice negated and twice unnegated. Deciding whether $\mathcal{I}$ is satisfiable is **NP**-complete [10].

We say that c_j is the h-th clause of x_i, if and only if, x_i appears in c_j and x_i appears in exactly $h - 1$ other clauses $c_{j'}$ with $j' < j$. We say that x_i is the ℓ-th variable of c_j, if and only if, x_i and exactly $\ell - 1$ other variables $x_{i'}$ with $i' < i$ appear in c_j.

Let us show how to build a 2-edge-colored graph G^c with no PEC closed trail upon $\mathcal{I}$. For each $x_i \in \mathcal{X}$ (resp., $c_j \in \mathcal{C}$) we build a gadget G_{x_i} (resp., G_{c_j}) as depicted on the left (resp. right) of Figure 2.1. The gadget of a variable x_i has 18 vertices. It consists of a right part (vertices $t^k_{i_a}, t^k_{i_b}$ for $k = 0, \ldots, 3$ and edges $t^0_{i_b}t^1_{i_a}$, $t^1_{i_b}t^2_{i_a}$, $t^2_{i_b}t^3_{i_a}$) a left part (vertices $f^k_{i_a}, f^k_{i_b}$ for $k = 0, \ldots, 3$ and edges $f^0_{i_b}f^1_{i_a}$, $f^1_{i_b}f^2_{i_a}$, $f^2_{i_b}f^3_{i_a}$), an entrance a_i, an exit b_i and edges $a_it^0_{i_a}$, $a_if^0_{i_a}$, $t^3_{i_b}b_i$, $f^3_{i_b}b_i$. The left (resp., right) part of this gadget corresponds to the case where x_i is set to *false* (resp., *true*). Note that each edge of G_{x_i} has color 2 (see Figure 2.1(a)). The gadget of a clause c_j consists of an entrance

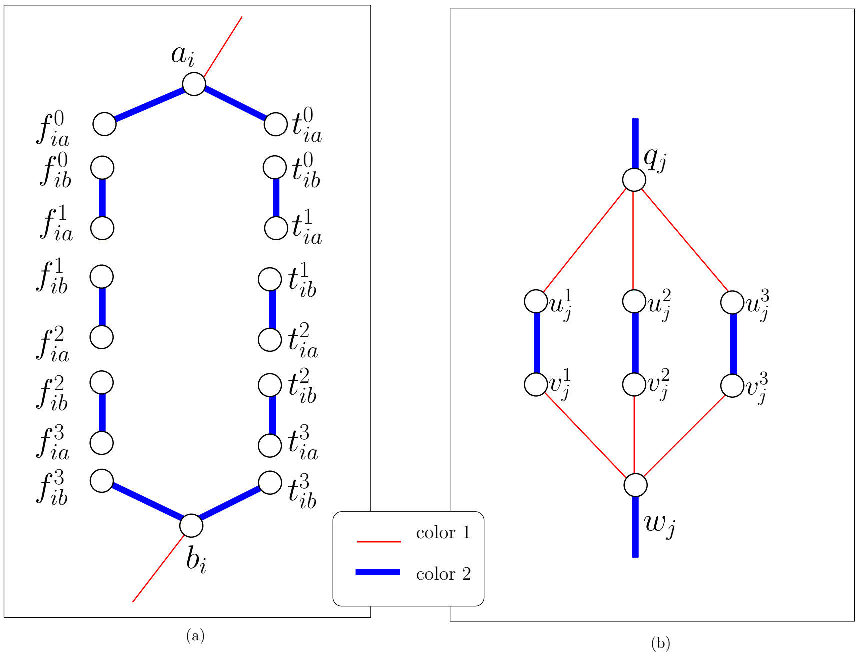


Figure 2.1: Gadgets for a variable x_i (left) and a clause c_j (right).

q_j, an exit w_j and three edges $u^1_j v^1_j$, $u^2_j v^2_j$, and $u^3_j v^3_j$ (all with color 2) corresponding to the first, second and third variables of c_j, respectively. Finally, we have 6 edges $q_j u^k_j$ for $k = 1, 2, 3$ and $v^k_j w_j$ for $k = 1, 2, 3$, all with color 1 (see Figure 2.1(b)).

We add four vertices s, t, s_a and t_a and we link the gadgets as follows (see Figure 2.2(Left)):

- sa_1, b_1a_2, $b_2a_3, \ldots, b_{n-1}a_n$ and b_nt_a, all of them with color 1 (thin);
- s_aq_1, w_1q_2, $w_2q_3, \ldots, w_{m-1}q_m$, w_mt, all of them with color 2 (bold);
- ss_a and t_at with colors 1 and 2, respectively.

For each pair x_i, c_j such that x_i is the ℓ-th variable of c_j and c_j is the h-th clause of x_i we proceed as follows. If x_i appears negated in c_j then add edges $t^{h-1}_{i_a} v^\ell_j$, $t^{h-1}_{i_b} u^\ell_j$

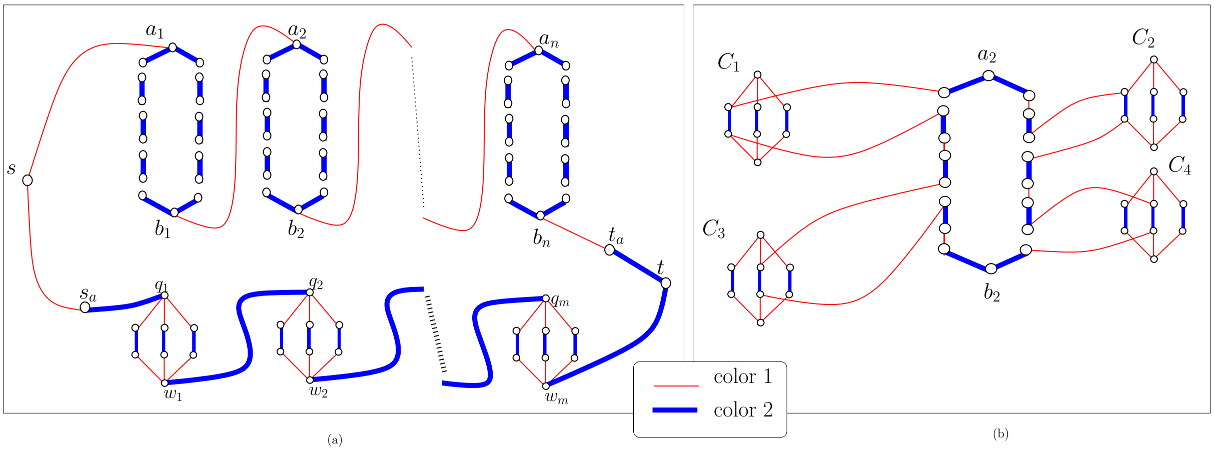


Figure 2.2: (Left) Linking gadgets G_{x_i} and G_{c_j}, respectively. (Right) x_2 appears in the clauses $c_1 = (x_2 \vee \bar{x}_3 \vee x_5)$, $c_2 = (\bar{x}_2 \vee x_3 \vee x_6)$, $c_3 = (\bar{x}_1 \vee x_2 \vee x_4)$ and $c_4 = (x_1 \vee \bar{x}_2 \vee x_5)$.

and $f_{i_a}^{h-1} f_{i_b}^{h-1}$, all colored 1 (thin). If x_i appears unnegated in c_j then add $f_{i_a}^{h-1} v_j^{\ell}$, $f_{i_b}^{h-1} u_j^{\ell}$ and $t_{i_a}^{h-1} t_{i_b}^{h-1}$, all colored 1 (thin). See the example of Figure 2.2.

Each vertex's degree is at most 4 and every edge incident to vertices a_i and b_i (resp., q_j and w_j), in G_{x_i} (resp., G_{c_j}) has color 2 (resp., color 1). In addition, every edge incident to s (resp., t) has color 1 (resp., color 2). In this way, it is easy to see that G^c contains no PEC closed trails.

In order to simplify the proof, we deal with the version where the edges have an odd and polynomially bounded length. Then, we can replace each edge e of length $\ell(e)$ by a PEC path ρ_e made of $\ell(e)$ edges (initial and terminal edges of ρ_e have color $c(e)$). We complete the construction of G^c by assigning a length $L_c = 14n - 1$ to the edges $w_1 q_2$, $w_2 q_3, \ldots, w_{m-1} q_m$, $w_m t$, and a length $L_v = 14mn + 2m - 14n + 1$ to $s a_1$. The remaining edges of G^c have length 1 (see Figure 2.2(Left)).

The graph contains $18n + 8m + 4$ vertices: 18 per variable gadget, 8 per clause gadget, s, s_a, t_a and t. Its construction is clearly done within polynomial time. An instance $\mathcal{I}'$ of our problem is to find two edge disjoint PEC s-t paths of total length at most $L = 14nm + 2m + 2$. We claim that a truth assignment for $\mathcal{I}$, instance of $(3, B2)$-SAT, corresponds to two edge disjoint PEC s-t paths in $\mathcal{I}'$, each of total length $L = 14mn + 2m + 2$ and vice-verse.

An s-t path with first edge sa_1 and last edge $t_a t$ is called a *variable path* and it is denoted by P_v. An s-t path with first edge ss_a and last edge $w_m t$ is called a *clause path* and it is denoted by P_c.

Suppose that we have two paths P_v and P_c, solution to $\mathcal{I}'$. If P_v uses an edge of length L_c then its total length exceeds L. Therefore it never passes through a vertex q_j or w_j $(1 \leq j \leq m)$. Since P_v is an s-t path, it must visit each variable gadget G_{x_i}. Thus, each vertex a_i is visited by P_v. Since P_v and P_c are edge-disjoint, P_c cannot go through a_i, $i = 1, \ldots, n$. Then, P_c must visit each clause gadget G_{c_j} to reach t. We can deduce a truth assignment for $\mathcal{I}$: if P_v uses the left (resp. right) part of G_{x_i} then $x_i =$ *false* (resp. $x_i =$ *true*). When P_c passes through the edge $u_j^k v_j^k$, it means that the k-th literal of c_j is true and c_j is satisfied. Since G_{c_j} reaches t, each clause has (at least) one true literal.

Suppose that we have a truth assignment, solution to $\mathcal{I}$. To build a variable path P_v, we take the right (resp., left) part, if and only if, x_i is *true* (resp., *false*) (see Figure 2.2(Right)). Then the total length of P_v is $L_v + 14n + 1 = L$. Each clause c_j is satisfied so there is an edge $u_j^k v_j^k$ of G_{c_j} not used by P_v. The clause path can use it to reach t. In this case P_c has total length $m(L_c + 3) + 2 = L$.

Observe in Figure 2.3 a complete example of the reduction used above and in Figure 2.4 the solution associated. One possible satisfying assignment to the instance

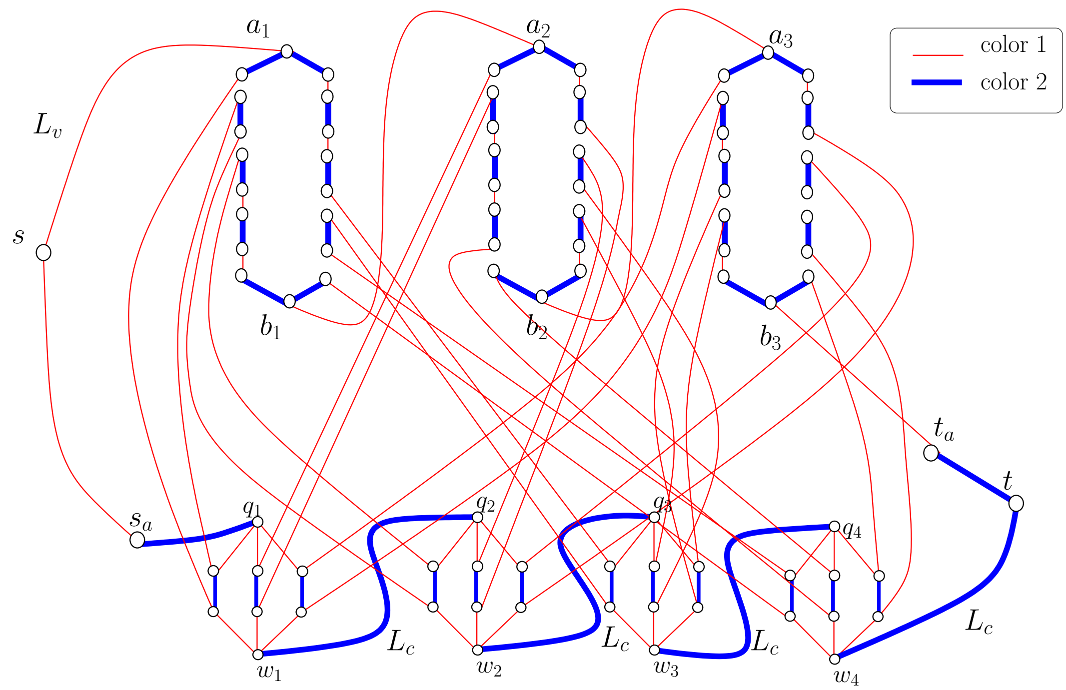


Figure 2.3: An example of the c-edge-colored graph G^c associated with the instance $\mathcal{I} = \{(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x_2} \vee \bar{x_3}) \wedge (\bar{x_1} \vee \bar{x_2} \vee x_3) \wedge (\bar{x_1} \vee x_2 \vee \bar{x_3})\}$ of the $(3, B2)$-SAT. With $L_c = 41$, $L_v = 135$ and $L = 178$.

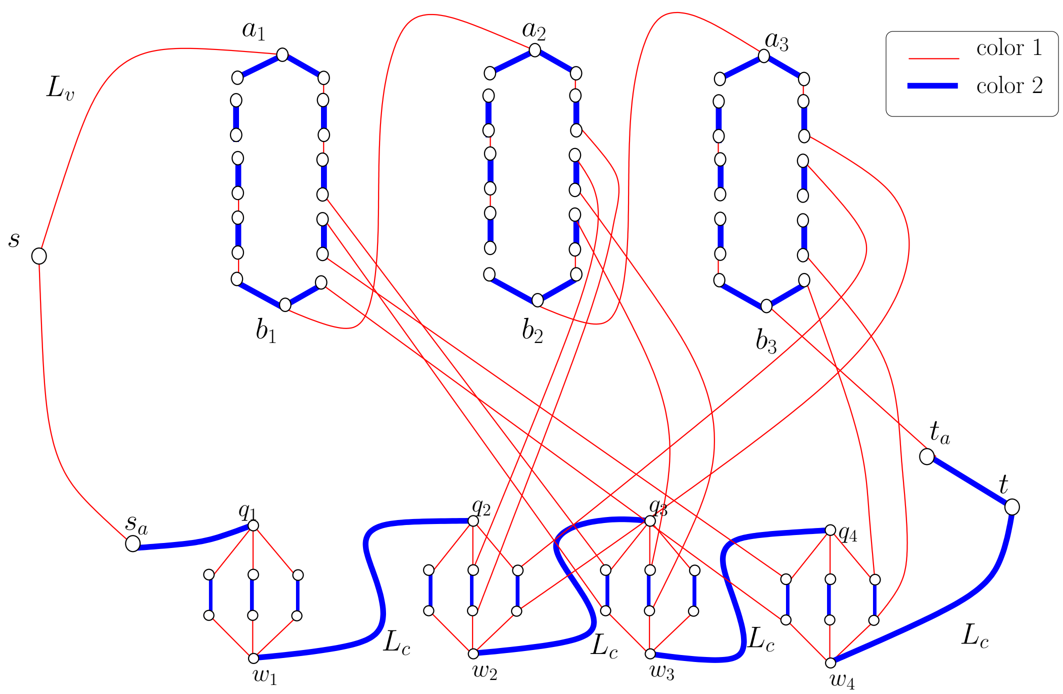


Figure 2.4: A subgraph of Figure 2.3 corresponding to the solution of the problem of finding 2 vertex disjoint PEC s-t paths with bounded length.

$\mathcal{I} = \{(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x_2} \vee \bar{x_3}) \wedge (\bar{x_1} \vee \bar{x_2} \vee x_3) \wedge (\bar{x_1} \vee x_2 \vee \bar{x_3})\}$ of the $(3, B2)$-SAT problem is the one that sets all the variables to true. Associated with the instance $\mathcal{I}$ with have the graph G^c of the Figure 2.3. In this case, where all the variables are set to true, the variable path P_v will use only the edges on the right part of G_{x_i}.

$\square$

Observe in the proof of Theorem 4 that we can reduce the maximum vertex degree from 4 to 3 if we change the gadget presented in Figure 2.1.(b) by the one in Figure 2.5.

As a final comment, we can prove Theorem 4 above in another way (if we remove the maximum vertex degree constraint) by using the 2 vertex/edge disjoint path with bounded length problem over directed acyclic graphs with arbitrary non-negative arc

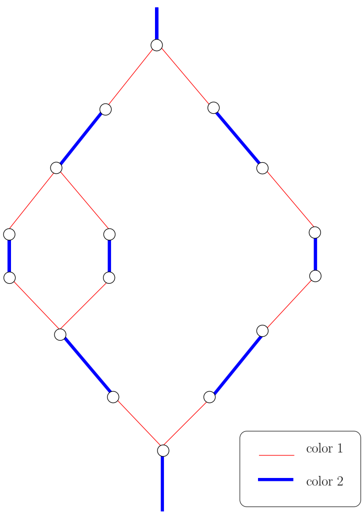


Figure 2.5: Clause Gadget used to reduce the maximum vertex degree in Theorem 4

weights (see [31, 40]). To see that, it suffices to change arcs $\vec{xy}$ with cost $w(\vec{xy})$ by edges xz, zy with colors 1 and 2, resp., and assign edge costs $w(\vec{xz}) = w(\vec{zy}) = \frac{w(\vec{xy})}{2}$. However, the arc weights are not polynomially bounded, which only give us a **NP**-completeness result in the normal sense.

2.2 The determination of PEC s-t trails visiting vertices a predefined number of times

In the work of Das and Rao [13], they characterize those 2-edge-colored complete graphs K_n^c which contain a PEC closed trail visiting each vertex x of $V(K_n^c)$ exactly $f(x) > 0$ times. Generalizing this theorem Bang-Jensen and Gutin [6] solved the problem of determining the length of a longest closed PEC trail visiting each vertex x in 2-edge-colored complete multigraphs at most $f(x) > 0$ times.

If G^c is a c-edge-colored graph containing no PEC closed trails, we propose a more

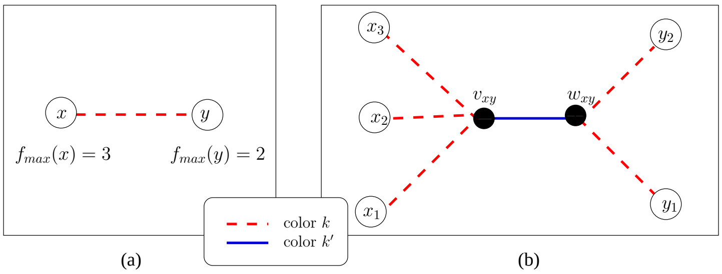


Figure 2.6: Construction of the modified trail-path graph $\bar{p}$-H^c.

general version of these problems and we show how to polynomially find, provided that one exists, a PEC s-t trail visiting all vertices of G^c a predefined number of times (defined by an interval associated with each vertex). Formally, given two integer non-negative functions f_{min} and f_{max} from $V(G^c)$ to $\mathbb{N}$ such that $0 \leq f_{min}(x) \leq f_{max}(x) \leq \lfloor \frac{d_{G^c}(x)}{2} \rfloor$, we show how to construct, if any, a PEC trail between vertices s and t, and visiting all vertices of $W = V(G^c) \backslash \{s, t\}$ exactly $f(x)$ times, for $x \in W$ and some $f(x) \in \{f_{min}(x), ..., f_{max}(x)\}$. Recall by the Theorem 2 that deciding whether or not G^c contains a PEC closed trail can be solved in polynomial time.

Thus, using both concepts of trail-path graph [1] and Edmonds-Szeider graph [38] (see Chapter 1), we can prove the following result:

Theorem 5. *Let G^c be a c-edge-colored graph with no* PEC *closed trails and $s, t \in V(G^c)$. Then we can find within polynomial time, if one exists, a* PEC *s-t trail visiting all vertices $x \in W$ exactly $f(x)$ times with $f_{min}(x) \leq f(x) \leq f_{max}(x)$.*

Proof: Basically, the idea is to construct both trail-path graph and Edmonds-Szeider graph in a modified manner in order to reduce PEC s-t trails (satisfying the constraints above) into perfect matchings over non-colored graphs.

Let $G^c = (V, E)$ be a c-edge-colored graph with no PEC closed trails and $s, t \in V$.

Without loss of generality, assume that $d_{G^c}(s) = d_{G^c}(t) = 1$ and then $f_{max}(x) = f_{min}(x) = 1$ for $x \in \{s,t\}$. Actually, by adding two dummy vertices s', t' and edges $s's$ and $t't$ with a new color in G^c, there is a PEC s'-t' trail, if and only if, there is a PEC s-t trail. Initially, construct a *modified trail-path* graph associated with G^c, denoted here by $\bar{p}$-H^c, by replacing each vertex x by a subset $S_x = \{x_1, \ldots, x_{\alpha_x}\}$ of vertices with $\alpha_x = f_{max}(x)$. To simplify our notation consider $x \in V(G^c)$ and $f_{max}(x) = f_{min}(x) = 1$, for $x = s, t$. Therefore, s_1 and t_1 are source and destination in $\bar{p}$-H^c. Thus, for any edge xy of G^c colored, say by k, we add two new vertices v_{xy} and u_{xy} and add edges $x_i v_{xy}$, $u_{xy} y_j$, for $i = 1, \ldots, \alpha_x$ and $j = 1, \ldots, \alpha_y$, all of them colored by k. Finally, we add edge $v_{xy}u_{xy}$ with a new unused color $k' \in \{1, \ldots, c\}$ with $k' \neq k$. Denote by $\bar{V} = \{v_{xy}, u_{xy} | xy \in E(G^c)\}$ this new set of auxiliary vertices. (See Figure 2.6)

Now define, randomly, a subset $S'_x = \{x_{i_1}, \ldots, x_{i_{\rho_x}}\}$ of S_x with $\rho_x = f_{min}(x)$. Thus, given $\bar{p}$-H^c as above, we construct the (non-colored) Edmonds-Szeider graph, say H, associated with $\bar{p}$-H^c (see Subsection 1.2). Note that for every $y \in \bar{W}$ for $\bar{W} = V(\bar{p}\text{-}H^c) \setminus (\{s_1, t_1\} \cup \bar{V})$, we obtain an associated (non-colored) subgraph H_y of H. Now, for every H_y associated with $y \in S'_x$, delete edges $y''_a y''_b$ (see Figure 2.7) and relabel by H'_y all these subgraphs. The resulting non-colored graph obtained in this way, denoted by H_m, will be called *modified Edmonds-Szeider graph.* The idea, provided that one PEC s-t trail in G^c exists (and satisfying both $f_{max}(x)$ and $f_{min}(x)$, $\forall x \in W$), is to find an associated PEC s_1-t_1 path in $\bar{p}$-H^c in order to force the visit (exactly once) of all vertices $y \in S'_x$ (the remaining vertices $y \in \bar{W} \setminus (\cup_{x \in W} S'_x)$ may be visited or not in this path). This may be accomplished by solving a perfect matching

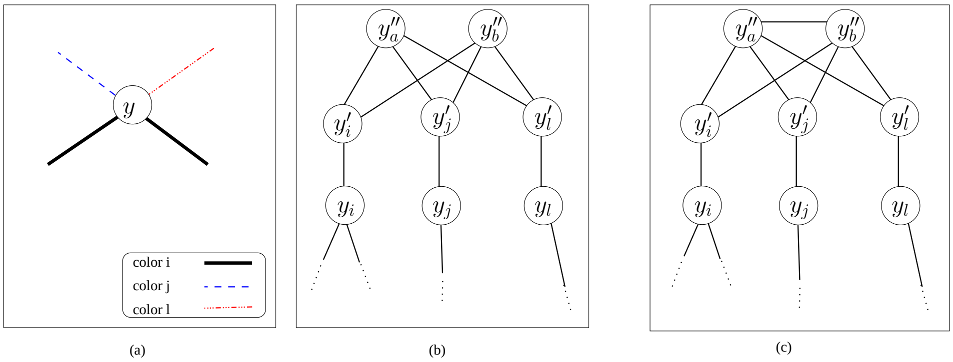


Figure 2.7: (a) Vertex $y \in \bar{W}$; (b) Subgraph H'_y associated with $y \in S'(x)$; (c) Subgraph H_y associated with $y \in S(x)$.

problem in H_m.

Thus, compute a perfect matching M in the modified Edmonds-Szeider graph H_m, if one exists. Given M, to determine the associated PEC s-t trail in G^c we first construct a non-colored graph $\bar{H}'$ by contracting subgraphs H_y and H'_y into a single vertex y and by contracting edges $v_{xy}u_{xy} \in E(\bar{V})$ into vertices P_{xy}. Let M' be the resulting non-contracted edges of M obtained in this way. It is easy to see that $\bar{H}'$ will contain a (non-colored) s_1-t_1 path (represented by P), cycles and isolated vertices associated, respectively, to a PEC s-t trail (represented by T), PEC closed trails and isolated vertices in G^c. However, by hypothesis, G^c does not contain PEC closed trails. Therefore, each pair of edges in M' will be associated with an edge in the path P and vice-verse. In this way, non-colored s_1-t_1 paths in $\bar{H}'$ will be associated with a PEC s-t trails in G^c.

Finally, by construction of $\bar{p}-H^c$ and the (non-colored) modified Edmonds-Szeider graph H_m, notice that every vertex y is visited exactly once in $\bar{p}$-H^c if $y \in S'_x$, and at most once for the remaining vertices of $S_x \setminus S'_x$. Since $|S'_x| = f_{min}(x)$ and $|S_x| = f_{max}(x)$, vertex $x \in W$ is visited exactly $f(x)$ times in G^c for some $f(x) \in$

$\{f_{min}(x), \ldots, f_{max}(x)\}$. □

Corollary 1. *Consider G^c an edge-colored graph with no* PEC *closed trails and two vertices $s, t \in V(G^c)$. Then, we can find in polynomial time (if any) a properly edge-colored Hamiltonian s-t path.*

Proof: It suffices to set $f_{min}(x) = f_{max}(x) = 1$, for every vertex $x \in W$ in Theorem5. □

Corollary 2. *Let G^c be a c-edge-colored graph with no* PEC *closed trails. Then, we can find within polynomial time, a shortest (resp., a longest)* PEC *s-t trail visiting vertices x of $V(G^c)$ at least $f_{min}(x)$ times (resp., at most $f_{max}(x)$ times).*

Proof: After the construction of the modified Edmonds-Szeider graph H_m (see the proof of Theorem 5), it suffices to assign costs $cost(pq) = 0$ for all edges pq of $E(H_y)$ and $E(H'_y)$ respectively, for every $y \in \bar{W}$. For the remaining edges of H_m we assign $cost(pq) = 1$. Now, to find a shortest (resp., a longest) PEC s-t trail visiting vertices x of G^c at least $f_{min}(x)$ times (resp., a most $f_{max}(x)$ times), compute, if possible, a minimum perfect matching (resp., maximum perfect matching) M in H_m. Note that a PEC s-t path P of $p - H^c$ with cost $cost(P)$ will be associated with a PEC s-t trail, say T, in G^c with cost $cost(T) = \frac{cost(P)}{3}$. In addition, in the case of the maximum perfect matching (if one exists), we always obtain a longest PEC s-t trail since G^c has no PEC closed trails. □

Now, we extend Theorem 5 by forcing the visit of a subset E' of edges.

Theorem 6. *Let G^c be a c-edge-colored graph with no* PEC *closed trails and let $E' \subseteq E(G^c)$. Then we can find within polynomial time, provided that one exists, a* PEC *s-t trail visiting all edges of E'.*

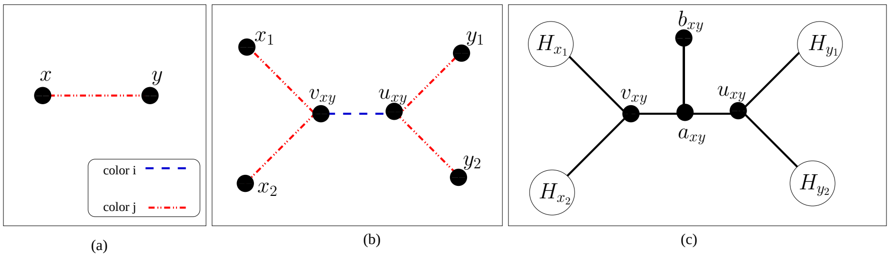


Figure 2.8: (a) Forcing the visit of edge $xy \in E'$; (b) Subgraph H_{xy} of $p-H^c$ associated with xy; (c) Non-colored subgraph of the Edmonds-Szeider graph H, associated with H_{xy}.

Proof: In order to force the presence of an edge $e = xy \in E'$ (colored, say i) at some PEC s-t trail of G^c, we first construct the trail-path graph $p - H^c$ and then the associated non-colored Edmonds-Szeider graph H in this order.

Note by the construction of $p - H^c$ and H, that we have two vertices v_{xy}, u_{xy} associated with edge $xy \in E^i(G^c)$. Now, for every pair v_{xy}, u_{xy} of H and with $xy \in E'$, we add two vertices a_{xy}, b_{xy} and change edge $v_{xy}u_{xy}$ by three new edges: $v_{xy}a_{xy}$, $a_{xy}b_{xy}$ and $a_{xy}u_{xy}$ respectively (as illustrated in the Figure 2.8). Let H' be this new non-colored graph.

Now, by using the same arguments as in the proof of Theorem 5, we can show that a perfect matching M of H', if one exists, will be associated with a PEC s-t trail of G^c visiting all edges of E', and vice-verse. $\square$

Note that Theorem 6 also allows to find a PEC Eulerian s-t trail in c-edge-colored graph with no PEC closed trails. Formally:

Corollary 3. *Let G^c be an edge-colored graph with no PEC closed trails and $s, t \in V(G^c)$. Then we can find in polynomial time, a properly edge-colored Eulerian trail or else decide it doesn't exist.*

Proof: By the Theorem 6 one can find an s-t trail, if any, by visiting all the edges of G^c, *i.e.*, we just set $E' = E(G^c)$. $\square$

The result presented in Corollary 3 is not very interesting since we recall that a polynomial algorithm is already known for finding PEC Eulerian trail (if one exists) in general c-edge-colored graphs [9].

Now, we have the following result regarding c-edge-colored graphs G^c with no PEC cycles. Note that PEC closed trails are allowed in this case.

Corollary 4. *Let G^c be a c-edge-colored graph with no* PEC *cycles, $s, t \in V(G^c)$ and a subset $A = \{v_1, ..., v_k\}$ of $V(G^c)\backslash\{s, t\}$. Then, the problem of finding a* PEC *s-t path visiting all vertices of A can be solved in polynomial time.*

Proof: Given G^c , we construct the associated Edmonds-Szeider graph, except that for the vertices $v_i \in A$, for $i = 1, ..., k$ we remove edge $v''_{i_a}v''_{i_b}$ (see Figure 2.7.(b)) in order to force the visit of all vertices of A. $\square$

In Theorem 7, we are interested in finding a PEC s-t trail passing by a given vertex v in G^c with no PEC cycles. Again, PEC closed trails are allowed. Surprisingly, we show that this problem is **NP**-complete if we are restricted to this particular class of graphs.

Theorem 7. *Let G^c be a c-edge-colored graph with no* PEC *cycles, vertices $s, t, v \in V(G^c)$. Then, the problem of finding a* PEC *s-t trail passing by v is **NP**-complete.*

Proof: Clearly, our problem belongs to NP. To prove that it is **NP**-complete, we use a reduction from the Path-Finding Problem (PFP), whose the objective is to find a s-t path through a vertex v in a (non-colored) digraph D [17]. Without

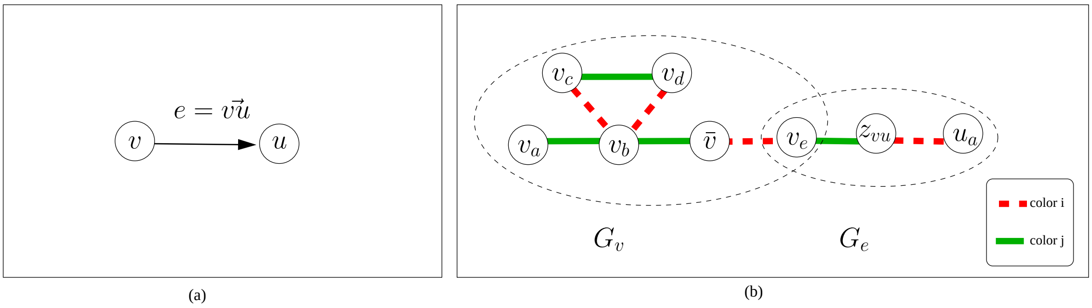


Figure 2.9: (a) Arc $\vec{vu} \in V(D)$. (b) Gadget G_v associated with vertex v and gadget G_e associated with edge e.

loss of generality, there is no incoming arcs at s and no outgoing arcs at t. Given a (non-colored) digraph $D = (V, A)$, instance of the PFP, we will show how to construct in polynomial time a 2-edge-colored graph G^c with no PEC cycles. For each vertex $v \in D$, create the following gadget G_v, with vertices $V(G_v) = \{v_a, v_b, v_c, v_d, v_e, \bar{v}\}$ and edges $v_a v_b$, $v_b \bar{v}$, $v_c v_d$ all colored with color j and edges $v_b v_c$, $v_b v_d$, $\bar{v} v_e$ with color i. All arcs $e = \vec{vu}$ in D are changed by edges $v_e z_{vu}$ and $z_{vu} u_a$ (gadget G_e) colored with i and j, respectively. See the example of Figure 2.9.

Observe that this transformation does not lead to a graph G^c with PEC cycle. So, if there is a directed s-t path a through a vertex v in D, there will be a PEC s-t trail through a vertex $\bar{v}$ in G^c. Conversely, if there is a PEC s-t trail through a vertex $\bar{v}$ in G^c, then we can easily find a directed s-t path a through a vertex v in D. □

Note in the Theorem 7 above that the set of all graphs containing no PEC closed trails is a subset of all graphs containing no PEC cycles.

Next, we present some open problems and future directions regarding c-edge-colored graphs with no PEC closed trails (or cycles).

Open Problem 1. *Consider a non-oriented c-edge-colored graph G^c with no* PEC *closed trails, an integer k and a sequence $p = (v_1, \ldots, v_k)$ of vertices in $V(G^c)$. Is it possible to find in polynomial time a* PEC *s-t path/trail visiting all vertices of p in this order?*

Open Problem 2. *Consider G^c a non-oriented c-edge-colored graph, an integer k and a sequence $C = (c_1, \ldots, c_k)$ of colors. Find a* PEC *s-t path/trail (if any) only visiting the sequence of C in this order. Is this problem polynomial for graphs with no* PEC *cycles?*

Open Problem 3. *Let L be the size of a minimum shortest* PEC *s-t path. Consider the problem of deciding whether a graph G^c (with no* PEC *closed trails) has k or more, edge disjoint* PEC *paths between nodes s and t, each having at most $L + 1$ edges. Is this problem **NP**-complete?*

In Tragoudas and Varol [40], the authors show that Problem 3 above is **NP**-complete for arbitrary non-colored graphs. We conclude the chapter by recalling an open problem posed by Abouelaoualim *et al.* [1]:

Open Problem 4. *Given a 2-edge-colored graph G^c with no* PEC *cycles, two vertices $s, t \in V(G^c)$ and a fixed constant $k \geq 2$. Does G^c contains k* PEC *vertex/edge disjoint paths between s and t? Is this problem **NP**-complete?*

Chapter 3

Monochromatic s-t paths in edge-colored graphs

Here, we deal with monochromatic s-t paths in c-edge-colored graphs G^c. We show that finding k vertex disjoint monochromatic s-t paths with different colors is **NP**-complete even if G^c has maximun vertex degree 4 and $k = 2$. As an immediate consequence, we show that the same problem over c-edge-colored digraphs is also **NP**-complete. We emphasize the fact that the paths have different colors because finding paths with the same color can be easily solved in polynomial time (it suffices to choose a color, one at a time, and remove all the other edges with different colors). In the monochromatic resultant graph find two paths as if the graph wasn't colored. This can be done in polynomial time [37]. Formally, we have the following result:

Theorem 8. *Let G^c be a c-edge-colored graph with $s, t \in V(G^c)$ with $c \geq 2$ and maximum vertex degree equal to 4. The problem of finding two vertex disjoint monochromatic s-t paths with different colors in G^c is **NP**-complete.*

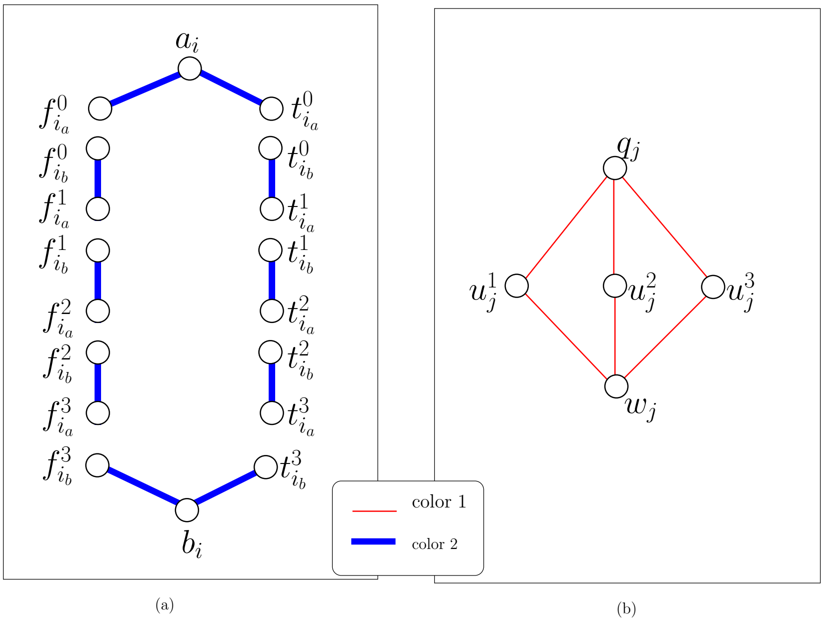


Figure 3.1: Gadgets for a variable x_i (left) and a clause c_j (right).

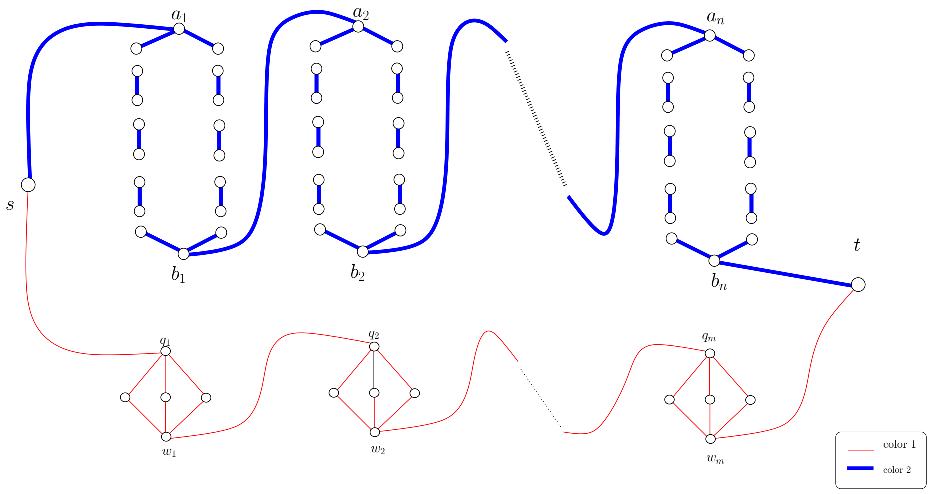


Figure 3.2: Linking components G_{x_i} and G_{c_j}, respectively.

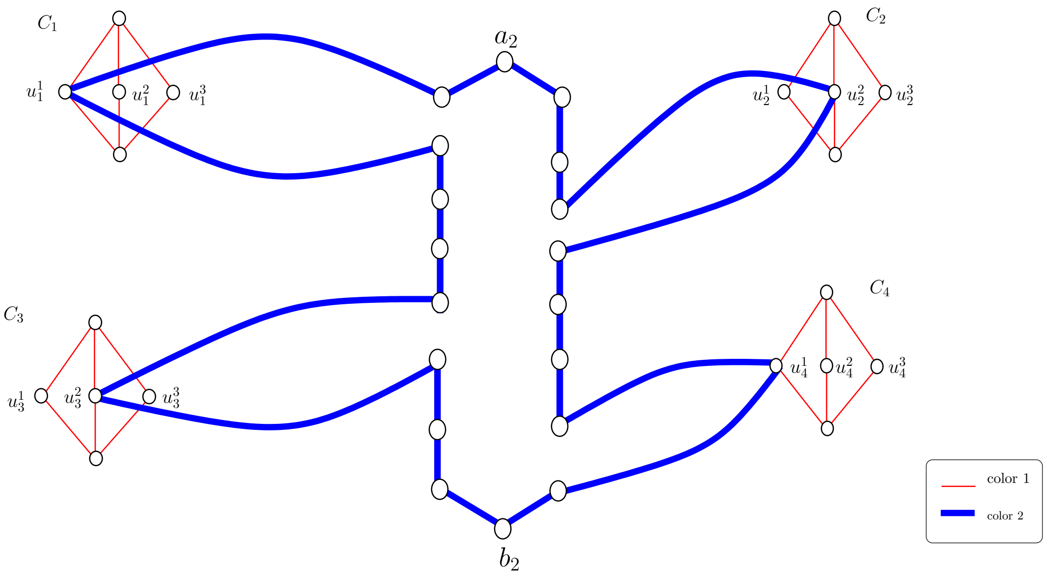


Figure 3.3: Variable x_2 appears in the clauses $c_1 = (x_2 \vee \bar{x}_3 \vee x_5)$, $c_2 = (\bar{x}_1 \vee \bar{x}_2 \vee x_6)$, $c_3 = (\bar{x}_1 \vee x_2 \vee x_4)$ and $c_4 = (\bar{x}_2 \vee x_3 \vee x_6)$.

Proof: This proof uses a similar idea of Theorem 4, *i.e.*, we reduce an instance $\mathcal{I}$ of the $(3, B2)$-SAT to the existence of 2 monochromatic $s-t$ paths with different colors in G^c for $c = 2$. We use the same notation and only describe how G^c is built upon $\mathcal{I}$.

The graph G^c will be composed by clause components G_{c_j} (for $j = 1, \ldots, m$) and variable components G_{x_i} (for $i = 1, \ldots, n$). For each $x_i \in \mathcal{X}$ we build a gadget as depicted on the left of Figure 3.1. Similarly to Theorem 4, the right (resp., left) part of this gadget corresponds to the case where x_i is set to *true* (resp., *false*). The gadget of a clause c_j consists of an entrance q_j, an exit w_j and 3 vertices u_j^1, u_j^2, and u_j^3 corresponding to the first, second and third variables of c_j, respectively. We conclude the construction of G_{c_j} by adding 6 edges $q_j u_j^k$ for $k = 1, 2, 3$ and $u_j^k w_j$ for $k = 1, 2, 3$, all of them with color 1 (thin). See Figure 3.1 for an example of the construction of the clause component and the variable component.

Now, we add vertices s, t and link all gadgets G_{x_i} (resp., G_{c_j}) by adding the following edges as described in the Figure 3.2:

- sa_1, b_1a_2, $b_2a_3, \ldots, b_{n-1}a_n$ and b_nt, all of them with color 2 (bold);
- sq_1, w_1q_2, $w_2q_3, \ldots, w_{m-1}q_m$, w_mt, all of them with color 1 (thin).

For each pair x_i, c_j such that x_i is the ℓ-th variable of c_j and c_j is the h-th clause of x_i we proceed as follows. If x_i appears negated in c_j then add edges $t_{i_a}^{h-1}u_j^\ell$, $t_{i_b}^{h-1}u_j^\ell$ and $f_{i_a}^{h-1}f_{i_b}^{h-1}$, all colored 2 (bold). If x_i appears unnegated in c_j then add $f_{i_a}^{h-1}u_j^\ell$, $f_{i_b}^{h-1}u_j^\ell$ and $t_{i_a}^{h-1}t_{i_b}^{h-1}$, all colored 2 (bold). Clearly, the construction of G^c can be done in polynomial time in the size of $\mathcal{X}$ and $\mathcal{C}$. Further, note that G^c has maximum vertex degree equal to 4.

Now, observe that truth assignments for an instance $\mathcal{I}$ of the $(3, B2)$-SAT problem are associated with 2 vertex disjoint monochromatic s-t paths of colors 1 and 2, respectively. To construct the path with color 2 (bold), whenever a variable x_i is *true* (resp., *false*), we take the sub-path between vertices a_i and b_i by using the right (resp., left) side of G_{x_i} (see the example of Figure 3.3). The unvisited vertices u_j^ℓ of c_j can be used at random, to construct the path colored 1 (thin) between s and t. Conversely, if we have 2 vertex disjoint monochromatic s-t paths of colors 1 and 2 then we have a truth assignment for $\mathcal{I}$. For instance, if a vertex u_j^ℓ of the component G_{c_j} is visited by some path colored 1 and variable x_i (appearing in the ℓ-th position of c_j) is in the negated form (resp., unnegated form) then variable x_i must be *false* (resp., *true*) and clause c_j will be *true* in the assignment. Therefore, by using both monochromatic s-t paths with colors 1 and 2 we can uniquely determine a truth assignment for $\mathcal{I}$, which completes the proof for $c = 2$.

The generalization of our proof for graphs containing $c \geq 3$ colors is identical to Theorem 4 above and will be omitted here. □

Theorem 8 above can be easily generalized for c-edge-colored digraphs:

Corollary 5. *Let D^c be a c-edge-colored digraph with maximum in- and out-degree equal to* 3 *and $s, t \in V(D^c)$. Then, the problem of finding two directed monochromatic s-t paths with different colors in D^c is **NP**-complete.*

Proof: In the non-oriented c-edge-colored graph G^c with maximum vertex degree 4 (see Theorem 8), whenever we have a path ρ with all edges colored k (for $k = 1, 2$) from s to t and passing by some edge xy, colored k, and y not belonging to the subpath from s to x in ρ, we change xy by $\vec{xy}$. Note that the maximum in-degree and out-degree in the resulting digraph D^c is 3. □

Finally, we conclude by noting that finding 2 monochromatic edge disjoint s-t paths in G^c can be easily done in polynomial time (it suffices to take all combinations of graphs with 2 colors).

Now, we conclude with the following related open problems:

Open Problem 5. *Is the problem of finding* 2 *monochromatic (vertex disjoint) s-t paths with different colors in planar c-edge-colored graphs **NP**-complete?*

Chapter 4

Paths, trails and circuits in edge-colored digraphs

Finding PEC paths, PEC trails, PEC cycles or PEC closed trails in undirected c-edge-colored graphs is polynomial [1, 38]. However finding directed PEC paths or PEC circuits in c-edge-colored digraphs seems harder. For example, Gutin, Sudakov and Yeo in [30] proved that deciding whether a 2-edge-colored digraph contains a PEC circuit is **NP**-complete. Nevertheless, this problem remains open if we are restricted to 2-edge-colored tournaments [30].

Here, we show that the problem of maximizing the number of edge disjoint PEC s-t trails can be solved in polynomial time on arbitrary edge-colored graphs. Surprisingly, we prove that the determination of one PEC s-t path is **NP**-complete. In addition, we show that finding a directed PEC closed trail in general c-edge-colored digraphs is polynomial time solvable (recall that finding PEC circuits is **NP**-complete [30]). We also prove that if the digraph is an edge-colored tournament deciding if it contains a PEC circuit passing through a given vertex v is **NP**-complete. As a consequence, we solve a weaker version of the open problem cited in [30] (*i.e.* whether or not a 2-edge-colored tournament contains a PEC circuit).

We conclude this chapter by proving that it is **NP**-complete to decide whether a 2-edge-colored tournament T_n^c contains a Hamiltonian and a directed PEC s-t path.

4.1 General c-edge-colored digraphs

Prior to deal with general c-edge-colored digraphs D^c we begin by the following simple case: when D^c has no circuits at all.

Lemma 1. *If D^c is a c-edge-colored acyclic digraph and s, t are two vertices of D^c then finding a directed* PEC *path from s to t is polynomial time solvable.*

Proof: We use an algorithm (see Algorithm 1) which maintains a set of labels $\mathcal{L}(v)$ for each vertex v (indicating the color of the last arc of each directed PEC path from s to v). At the beginning of the algorithm, $\mathcal{L}(v) = \emptyset$ for all v. The level of a vertex v, denoted by $\ell(v)$, is the length (*i.e.*, number of arcs) of the longest path between s and v. Therefore $\ell(s) = 0$, $\ell(v) \leq n-1$ for all v and $\ell(u) < \ell(v)$ for all arc $\vec{uv}$. For acyclic (non-colored) digraphs, a longest path can be found in time $\mathcal{O}(n+m)$ [7].

A label in $\mathcal{L}(v)$ indicates the color of the last arc of each directed PEC path from s to v. Thus, it is easy to see that D^c admits a directed PEC s-t path, if and only if, $\mathcal{L}(t) \neq \emptyset$.

$\square$

Algorithm 1 Polynomial algorithm for finding a PEC s-t path in c-edge-colored acyclic digraphs.

for all vertex v **do**
    $\mathcal{L}(v) \leftarrow \emptyset$
end for
for $j = 1$ to $n - 1$ **do**
    for $i = 0$ to $j - 1$ **do**
        for all arc $\vec{uv}$ such that $\ell(u) = i$ and $\ell(v) = j$ **do**
            if $u = s$ or $\mathcal{L}(u) \setminus \{c(\vec{uv})\} \neq \emptyset$ **then**
                $\mathcal{L}(v) \leftarrow \mathcal{L}(v) \cup \{c(\vec{uv})\}$;
            end if
        end for
    end for
end for

In Theorem 9, we discuss the same question for c-edge-colored digraphs with no PEC circuits. Note that consecutive arcs in the circuits may have the same color in this case.

Theorem 9. *Deciding whether or not a 2-edge-colored digraph D^c with no* PEC *circuits contains a directed* PEC *path from s to t is **NP**-complete.*

Proof: We use a reduction from the Path with Forbidden Pairs Problem (PFPP, in short). In PFPP, we are given a (non-colored) digraph $D = (V, A)$, two vertices $v, w \in V$ and a collection $C = \{(a_1, b_1), \ldots, (a_q, b_q)\}$ of pairs of vertices (with $a_i \neq b_i$) from $V \setminus \{v, w\}$. The objective is to determine whether there exists a directed path connecting v to w and passing through at most one vertex from each pair of C. The PFPP was shown **NP**-complete [18] even if D is acyclic and all pairs of C are required to be disjoint (see problem [GT54] page 203 in [25]).

Let $D = (V, A)$ be an acyclic digraph containing $v, w \in V$ and a subset C of disjoint pairs of vertices. Without loss of generality, assume that $d^-_D(v) = d^+_D(w) = 0$. The construction of D^c is done in two steps. We first build a (non-colored) digraph D' and then we build D^c from D'. The digraph $D' = (V', A')$ is such that $V' = V \cup \{s\}$, $A' = A \cup A'_1 \cup A'_2$ with $A'_1 := \{\vec{sa_1}, \vec{sb_1}, \vec{a_qu}, \vec{b_qu}\}$ and $A'_2 := \{\vec{a_ia_{i+1}}, \vec{a_ib_{i+1}}, \vec{b_ia_{i+1}}, \vec{b_ib_{i+1}} : i = 1, \ldots, q-1\}$ and vertices v and w are replaced by u and t, respectively. For the moment, two arcs connecting the same pair of vertices may exist.

We build D^c as follows: for arcs in A'_1, $\vec{sa_1}$ and $\vec{sb_1}$ are colored *blue* (color 2), while arcs $\vec{a_qu}$ and $\vec{b_qu}$ are colored *red* (color 1). Next, we apply the following transformation: each arc $e = \vec{xy}$ of $A \cup A'_2$ is replaced by a directed path of length two, that is $\vec{xv_e}, \vec{v_ey}$,

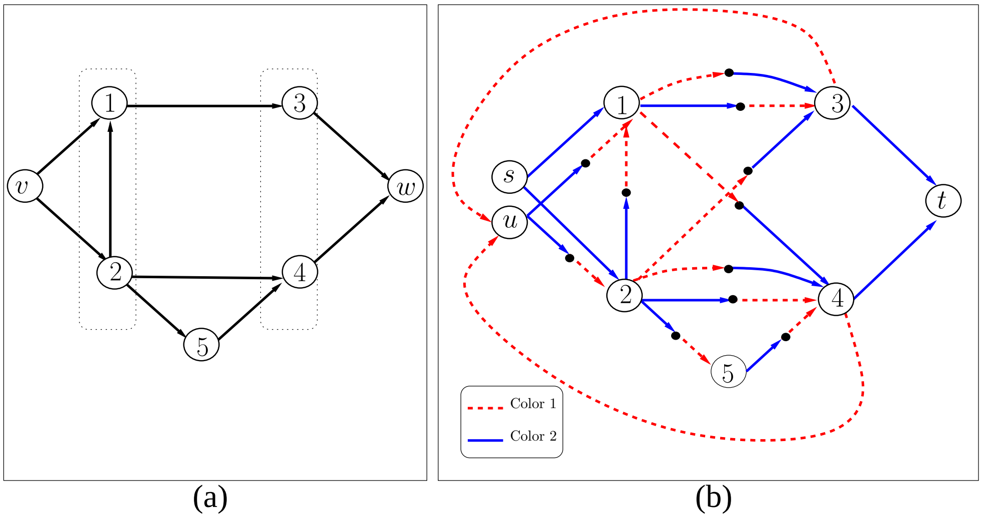


Figure 4.1: Reduction from the PFPP with $C = \{\{1,2\},\{3,4\}\}$ to the *directed* PEC *s-t path* problem. Color 1 (resp., 2) corresponds to *red* (resp., *blue*).

except for arcs incident to t. If $e = \vec{xy} \in A$, then $\vec{xv_e}$ is colored in *blue* and $\vec{v_ey}$ is colored in *red* (if $e = \vec{xt}$, then e is colored in *blue*). By extension, arcs $\vec{xv_e}, \vec{v_ey}$ are in A. If $e = \vec{xy} \in A'_2$ then $\vec{xv_e}$ is colored in *red* and $\vec{v_ey}$ is colored in *blue*. By extension, arcs $\vec{xv_e}, \vec{v_ey}$ are in this case in A'_2. The construction of D^c is completed (an example is given in Figure 4.1). This construction is clearly done within polynomial time and D^c is a 2-edge-colored digraph.

Now we give an intermediate property that will help us in the proof:

Property 1. *Any directed* PEC *path of D^c cannot use two consecutive arcs $\vec{xy}$ and $\vec{yz}$ such that $\vec{xy} \in A$ (resp., $\vec{xy} \in A'_1 \cup A'_2$) and $\vec{yz} \in A'_1 \cup A'_2$ (resp., $\vec{yz} \in A$) except if $y = u$.*

Proof: By inspection. If $\vec{xy} \in A$ (resp., $\vec{xy} \in A'_2$) then $\vec{v_{e_1}y} \in V(D^c)$ is *red* (resp., *blue*) and if $\vec{yz} \in A'_2$ (resp., $\vec{yz} \in A$) then $\vec{yv_{e_2}} \in V(D^c)$ is *red* (resp., *blue*). Thus, move from A (resp., A'_2) to A'_2 (resp. A) is not possible. Consider $y \neq u$, for

$\vec{xy} \in A_1'$, the arcs $\vec{xy} = s\vec{a}_1$ or $\vec{xy} = s\vec{b}_1$ in $V(D^c)$ are colored blue and if $\vec{yz} \in A$ the arc $y\vec{v}_{e_2} \in V(D^c)$ also have the color blue, or if $\vec{xy} \in A$ (consider the arcs $v_{e_1}\vec{a}_q, v_{e_2}\vec{b}_q \in V(D^c)$ colored red) and $\vec{yz} \in A_1'$ then for $\vec{yz} = a_q\vec{u}$ or $\vec{yz} = b_q\vec{u}$ in $V(D^c)$ also have color red. Then, going from A (resp., A_1') to A_1' (resp. A) is not possible either. Now, consider $y = u$, if $\vec{xy} \in A_1'$ then the arcs $a_q\vec{u}, b_q\vec{u} \in V(D^c)$ have color red and if $\vec{yz} \in A$ then the arcs $u\vec{v}_{e_1}, u\vec{v}_{e_2} \in V(D^c)$ have color blue. So, if $y = u$ it is possible to move from A (resp., A_1') to A_1' (resp. A). •

From Property 1, we deduce that any directed PEC path of D^c from s to t first uses some arcs in $A_1' \cup A_2'$ and after it uses some arcs in A (after passing through u).

Let us show that D^c contains no PEC circuit. Since (V', A) (by hypothesis) and $(V', A_1' \cup A_2')$ (by construction) have no circuits, if D^c has, it must contain two consecutive arcs such that the first arc is in A (resp., $A_1' \cup A_2'$) and the second arc is in $A_1' \cup A_2'$ (resp., A). Using Property 1, the circuit is not PEC.

Finally, using Property 1, we claim that we have a directed path from v to w in D and visiting at most one vertex from each pair of C, if and only if, we have a directed PEC path from s to t in D^c. To see that, let K with $|K| \leq q$ be the subset of vertices belonging to C in the solution of the PFPP. As a consequence of that, we have a directed PEC path from u to t in D^c, say α, and visiting the same set K of vertices. Therefore, we can construct a PEC path from s to t in D^c by concatenating a PEC path from s to u containing no vertices of K (which always exist in this case) with the path α from u to t. Notice that, if we have a path from v to w in D visiting both vertices of an arbitrary pair of C, we do not have a PEC path from s to t in D^c.

Conversely, consider a PEC path from s to t in D^c. Note by construction of D^c, that we have a path from s to u containing exactly one vertex from each pair of C. As a consequence of that, the PEC path from u to t in D^c contains at most one vertex from each pair of C. Thus, if we repeat the same steps in the construction of D^c in the reverse order (*i.e.*, from D^c to D), we can easily construct a path from v to w in the associated (non-colored) acyclic digraph D and visiting each vertex of C at most once. Hence, the determination of one directed PEC s-t path in D^c with no PEC circuits is **NP**-complete. $\square$

Now, we show in Corollary 6 that the previous theorem holds even if the number of colors c of D^c is very large. Intuitively, this problem becomes easier when 3 colors or more are considered (an extreme case is when all arcs of D^c have different colors). As a consequence, an interesting question is to study the **NP**-completeness of these problems for digraphs with many colors. Thus, we have the following result:

Corollary 6. *Deciding if a c-edge-colored digraph with no* PEC *circuits D^c contains a directed* PEC *s-t path is **NP**-complete, even if $c = \Omega(|V(D^c)|^2)$.*

Proof: We extend Theorem 9 to construct digraphs with $2n$ vertices, $c = \Omega(n^2)$ colors and with no PEC circuits. To do so, we first construct a 2-edge-colored digraph $D_\varphi^{c'}$ with no PEC circuits, $c' = 2$ and with n vertices as done in Theorem 9. Next, we build a tournament T_n^c with n vertices and containing no circuits with colors $I_c \supseteq I_{c'}$. To do that, given a non-colored complete graph K_n, it suffices to choose arbitrary vertices of K_n and change all adjacent (non-oriented) edges by incoming arcs with arbitrary colors of I_c. Next, we choose two arbitrary vertices $v_1 \in V(D_\varphi^{c'})$ and $v_2 \in V(T_n^c)$ and add arc $v_1\vec{v}_2$ with an arbitrary color of I_c. The resulting digraph D^c has $2n$ vertices and at most $\frac{n(n-1)}{2}$ different colors (the colors inside T_n^c). Therefore, directed PEC s-t paths in 2-edge-colored digraphs (with no PEC circuits) correspond to directed PEC s-t paths in digraphs with $c = \Omega(n^2)$ colors (with no PEC circuits) and vice verse. □

Now, we have the following result regarding planar edge-colored digraphs:

Corollary 7. *Let D^c be a planar c-edge-colored digraph containing no* PEC *circuits, two vertices $s, t \in V(D^c)$ and $c = \Omega(n^2)$. Then, the problem of finding a directed* PEC *path between s and t in D^c is **NP**-complete.*

Proof: Basically, given D^c containing no PEC circuits, the idea is to conveniently change all intersections by new vertices in order to make it planar. Note that the number of intersections is polynomially bounded on the size of D^c.

Thus, whenever we have an intersection between 2 arcs $\vec{ab}$ and $\vec{cd}$, say colored *blue*, we add 3 new vertices f_1, f_2 and f_3 and replace arcs $\{\vec{ab}, \vec{cd}\}$ by 2 sets of arcs $\{\vec{af_1}, \vec{cf_2}, \vec{f_2b}, \vec{f_3d}\}$ and $\{\vec{f_1f_2}, \vec{f_2f_3}\}$, respectively colored *blue* and *red* (see Figure 4.2). However, if $\vec{ab}$ and $\vec{cd}$ have different colors (say *red* and *blue*), we add the vertices f_1, f_2, f_3 and change $\vec{ab}$ and $\vec{cd}$ by arcs $\{\vec{cf_2}, \vec{f_2f_1}, \vec{f_3d}\}$ all colored *blue*, and arcs $\{\vec{af_2}, \vec{f_2f_3}, \vec{f_1b}\}$ all colored *red* (see Figure 4.3). Obviously, the resulting digraph, denoted by D^c_P, is a planar c-edge-colored digraph and contains no PEC circuits. Therefore, if we have some path passing by $\vec{ab}$ (resp., $\vec{cd}$) in D^c, we have a path passing vertices a and b (resp., c and d) in D^c_P. □

It is natural to raise the same question asked in Theorem 9 for trails instead of for paths. Unfortunately, we cannot use the same arguments as in the proof of Theorem 9 (directed paths from v to w in D and visiting both vertices of an arbitrary pair of C may correspond to directed PEC s-t trails in D^c). Therefore it is interesting to study

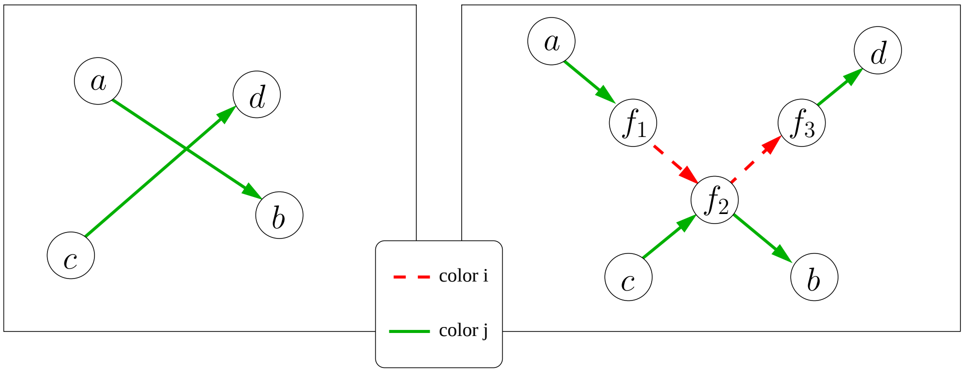


Figure 4.2: (a)Intersection of directed edges with the same color. (b)Making it planar.

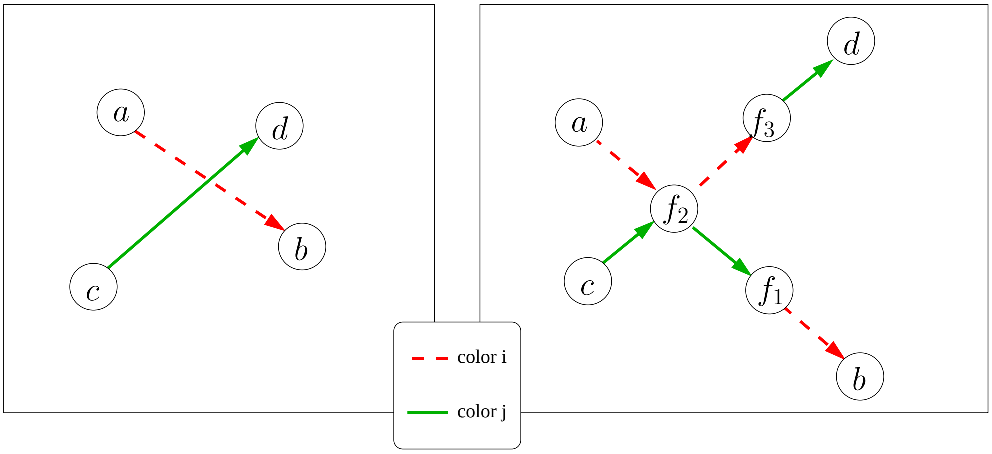


Figure 4.3: (a)Intersection of directed edges with different colors. (b)Making it planar.

the complexity of finding a directed PEC s-t trail in D^c. The problem turns out to be polynomial when using the notion of reload cost s-t trails [3, 28] (see Subsection 1.1 for the definition of reload costs).

Theorem 10. *Given an arbitrary c-edge-colored digraph D^c, finding a directed* PEC *s-t trail can be solved within polynomial time.*

Proof: A more general version of this problem was polynomially solved in [3]. Given reload costs $r_{i,j}$ associated with each pair of colors $i, j \in I_c$ (see Subsection 1.1 for the definition of reload costs), and costs $w(e)$ associated with each arc $e = \vec{xy}$, the objective is minimize

$$f(\rho) = \sum_{i=1}^{k} w(e_i) + \sum_{j=1}^{k-1} r_{c(e_j),c(e_{j+1})}$$

where $\rho = (v_1, e_1, \ldots, e_k, v_{k+1})$ with $v_1 = s$, $v_{k+1} = t$ and $e_i \neq e_j$ for $i \neq j$, is a sequence of arcs in a directed s-t trail (here, let us call this problem the *Minimum Reload+Weight Directed s-t Trail problem*). Basically, as described in [3], the idea is to apply a splitting procedure to all vertices v of $V(D^c) \setminus \{s, t\}$ (with $k_1(v)$ incoming arcs and $k_2(v)$ outgoing arcs) and construct a non-colored digraph $H(v)$ with unitary arc capacities as illustrated in the Figure 4.4. After repeating this process for each $v \in V(D^c) \setminus \{s, t\}$ we obtain a new uncolored digraph H.

Original arcs of D^c maintain their arc costs and unitary arc capacities in H and arcs $\vec{xy}$ of the complete bipartite digraphs of $H(v)$ (for each v) receive unitary arc capacities and appropriate reload costs $r_{i,j}$ where i and j are the colors of 2 arcs entering and leaving vertex v in D^c. Therefore, by applying a polynomial minimum cost flow algorithm to H to send one unit of flow between s and t we can polynomially solve the Minimum Reload+Weight Directed s-t Trail problem in D^c.

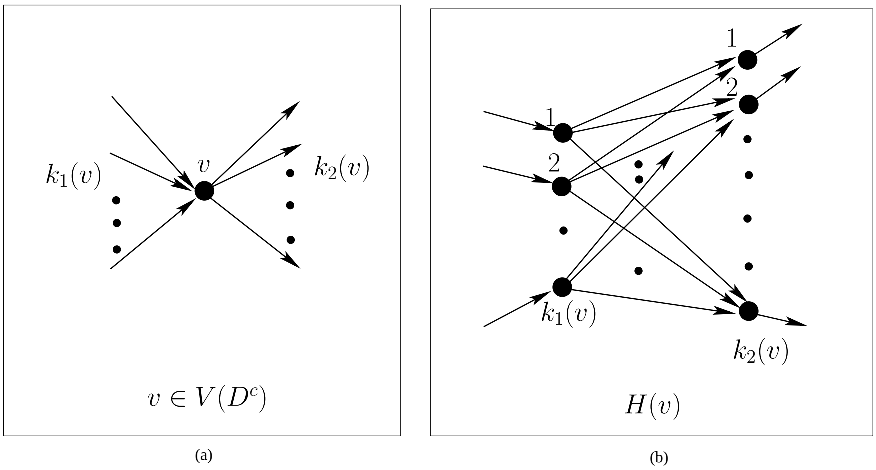


Figure 4.4: Splitting at vertex $v \in V(D^c)$ with $k_1(v)$ incoming arcs and $k_2(v)$ outgoing arcs.

Hence, in order to find a PEC s-t trail in D^c, it suffices to assume unitary arc capacities, to set $w(e) = 0$ for every arc $e = \vec{xy}$ of D^c and assign reload costs $r_{i,i} = 1$ and $r_{i,j} = r_{j,i} = 0$ for $i, j \in I_c$ with $i \neq j$. Thus, there exists a reload+weight directed s-t trail ρ with total cost $f(\rho) = 0$, if and only if, D^c has a directed PEC s-t trail. Therefore, we can find a directed PEC s-t trail within polynomial time (if one exists). □

In the work of Gutin, Sudakov and Yeo [30], they show that the determination of PEC circuits is **NP**-complete on arbitrary digraphs D^c for $c = 2$. However, as an immediate consequence of the Theorem 10, we can show that the determination of directed PEC closed trails can be done in polynomial time, provided that one exists. Formally:

Corollary 8. *Let D^c be a c-edge-colored digraph with $c \geq 2$. Then, the problem of finding a directed* PEC *closed trail in D^c (if any) can be solved in polynomial time.*

Proof: Our construction is done in two steps. Initially, for each vertex $x \in V(D^c)$

(one at a time), we apply the following procedure: we build a new graph, say D^c_x, by replacing x by two new vertices x_1, x_2 with $N^+_{D^c_x}(x_1) = N^+_{D^c}(x)$ and $N^-_{D^c_x}(x_2) = N^-_{D^c}(x)$ (all incoming and outgoing arcs are colored alike) and find, if one exists, a PEC trail from x_1 to x_2 in the new digraph D^c_x. Note that after finding a PEC trail between x_1 and x_2 in D^c_x the associated closed trail passing by x in D^c, say τ, may not be PEC (since both arcs of τ passing by x may have the same color). To avoid that we conclude with the following second step: for each color i with $N^i_{D^c}(x) \neq \emptyset$, delete all outgoing arcs of x_1, defined by $N^+_{D^c_x}(x_1)$ with color $j \neq i$ and delete all incoming arcs of x_2 colored i, defined by $N^-_{D^c_x}(x_2)$. Now, try to find a directed PEC trail from x_1 to x_2. Obviously, both steps are polynomially bounded. Thus, after finding a directed PEC x_1-x_2 trail in D^c_x, if any, we obtain in polynomial time a directed PEC closed trail passing by x in D^c. $\square$

Now, we can generalize Theorem 10 above to obtain the following stronger result:

Theorem 11. *Let D^c be a c-edge-colored digraph. The problem of maximizing the number of directed* PEC *s-t trails in D^c is polynomial time solvable.*

Proof: We construct a digraph H (associated with D^c) with the same reload costs, arc capacities and arc costs as in Theorem 10. Then it suffices to solve a sequence of minimum cost flow problems from s to t in H. The algorithm proceeds as follows: (1) Set $\theta \leftarrow n-2$; (2) Solve the minimum cost flow problem between s and t in H by sending θ units of flow and obtain ρ (if one exists); (3) If H contains a feasible flow ρ with $f(\rho) = 0$ then we are done (return ρ, θ and stop). Otherwise, set $\theta \leftarrow \theta - 1$ and go to step 2. We clearly get a polynomial time procedure to maximize the number of directed PEC trails from s to t since the minimum cost flow problem is polynomial time solvable. □

4.2 Tournaments

A tournament is a digraph which corresponds to a complete asymmetric binary relation. As indicated previously, one can build a tournament as follows: take a complete undirected graph and assign a direction to each edge. The problems of finding directed PEC s-t paths and PEC circuits in c-edge-colored tournaments are challenging. For example, the complexity of determining a PEC circuit in a 2-edge-colored tournament is evoked in [7, 30].

We begin with the problem of finding a directed PEC Hamiltonian s-t path. Dealing with uncolored tournaments, one of the earliest results is Rédei's theorem, which proves that every tournament has a directed Hamiltonian path (the endpoints are not specified) [36]. More recently, in [8] the authors gave a polynomial algorithm to find a directed Hamiltonian s-t path (if one exists) in a uncolored tournament. Given a general c-edge-colored digraph D^c, the problem of deciding if D^c contains a directed PEC Hamiltonian s-t path is **NP**-complete (since it generalizes the Directed Hamiltonian s-t path problem in general uncolored digraphs) [7]. However a nice characterization [16] shows that it is polynomial in undirected c-edge-colored complete graphs (with not specified endpoints). Here, if we fix a source s and a destination t, we prove that this result cannot be extended to the directed case.

Theorem 12. *Deciding whether a 2-edge-colored tournament T^c contains a directed* PEC *Hamiltonian s-t path is **NP**-complete.*

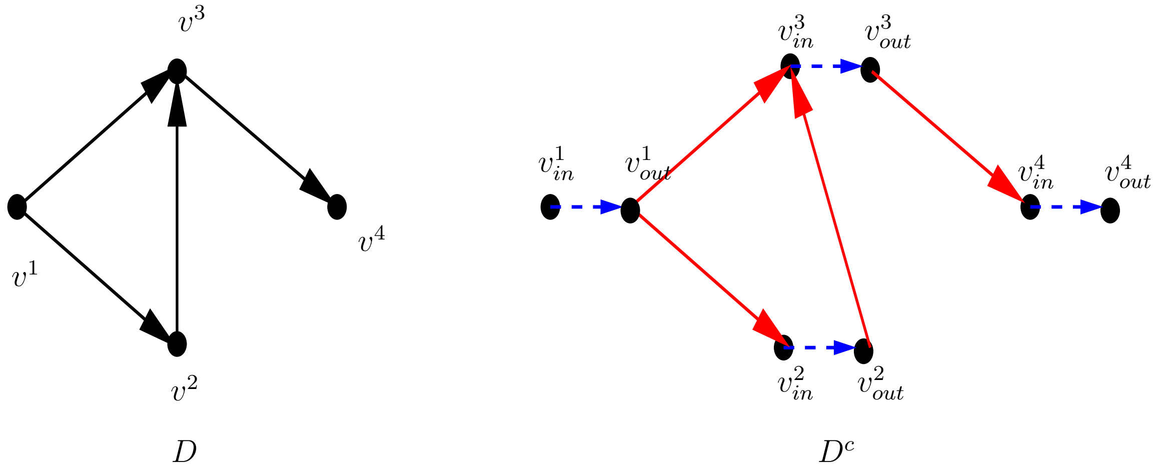


Figure 4.5: A digraph D and the 2-edge-colored digraph D^c. Dotted arcs are colored *blue* and rigid arcs are colored *red*.

Proof: We use a reduction from the directed Hamiltonian s'-t' path problem in general uncolored digraphs (DHPP in short). Given a digraph $D = (V, A)$ and two vertices s', t', DHPP asks whether a directed Hamiltonian s'-t' path exists. DHPP is **NP**-complete (see problem [GT39] page 199 in [25]).

Let $D = (V, A)$ be a digraph where $V = \{v^1, \dots, v^n\}$ and $v^1 = s'$, $v^n = t'$, instance of DHPP. Without loss of generality, assume that $d^-_D(v^1) = d^+_D(v^n) = 0$. The construction of the 2-edge-colored tournament T^c is done in two steps: we first build a 2-edge-colored digraph D^c and then we complete D^c into T^c.

The 2-edge-colored digraph $D^c = (V', A')$ is built in the following way: $V' = \{v^i_{in}, v^i_{out} : i = 1, \dots, n\}$ and $A' = A'_1 \cup A'_2$ where $A'_1 = \{\overrightarrow{v^i_{out} v^j_{in}} : \overrightarrow{v^i v^j} \in A\}$ and $A'_2 = \{\overrightarrow{v^i_{in} v^i_{out}} : i = 1, \dots, n\}$. Arcs in A'_1 are colored *red* while arcs in A'_2 are colored in *blue*. See Figure 4.5 for an illustration of D^c.

Next we build the tournament T^c from D^c as follows. For every missing arc in D^c, we apply the following procedure where $1 \leq i < j \leq n$ is assumed. If the endpoints

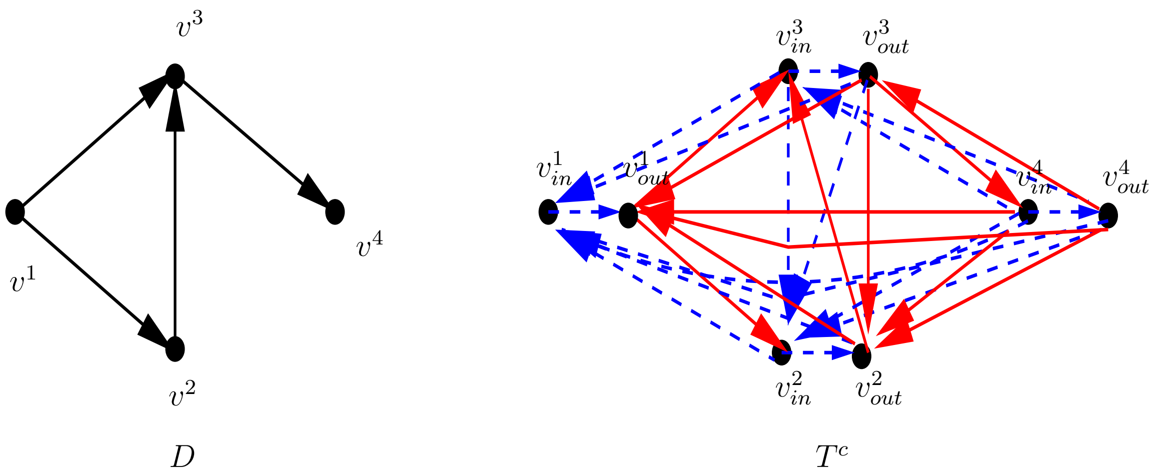


Figure 4.6: A digraph D and the 2-edge-colored tournament T^c. Dotted arcs are colored *blue* and rigid arcs are colored *red*.

of the missing arc are v_{in}^i and v_{in}^j (resp., v_{in}^i and v_{out}^j), add a *blue* arc $\overrightarrow{v_{in}^j v_{in}^i}$ (resp., $\overrightarrow{v_{out}^j v_{in}^i}$). If the endpoints of the missing arc are v_{out}^i and v_{in}^j (resp., v_{out}^i and v_{out}^j), add a *red* arc $\overrightarrow{v_{in}^j v_{out}^i}$ (resp., $\overrightarrow{v_{out}^j v_{out}^i}$). These new *blue* (resp., *red*) arcs define a set denoted by A_2'' (resp., A_1'').

The construction is completed (see Figure 4.6 for an illustration). It is clearly done within polynomial time. The resulting tournament is 2-edge-colored. Its *blue* arcs belong to $A_2' \cup A_2''$ while its *red* arcs belong to $A_1' \cup A_1''$. Let us give an intermediate property.

Property 2. *No directed* PEC *path from v_{in}^1 to v_{out}^n in T^c can use an arc of $A_1'' \cup A_2''$.*

Proof: By contradiction suppose that a directed PEC path $\rho = (v_0, e_0, v_1, e_1, \ldots, e_k, v_{k+1})$ linking $v_0 = v_{in}^1$ to $v_{k+1} = v_{out}^n$ uses some arcs of $A_1'' \cup A_2''$. Consider the last arc $e_p \in A_1'' \cup A_2''$ used by ρ (that is $e_q \notin A_1'' \cup A_2''$ for $q = p+1, \ldots, k+1$). If $e_p = \overrightarrow{v_{in}^j v_{in}^i}$ or $e_p = \overrightarrow{v_{out}^j v_{in}^i}$ $(i < j)$ then it belongs to A_2'' and it is *blue*. We have $v_{in}^i \neq v_{out}^n$ so the path must contain an arc going out of v^i_{in} which does not belong to $A''_1 \cup A''_2$. This arc $e_{p+1} = v^i_{in}\vec{v}^i_{out}$ is *blue*, contradiction. Otherwise, $e_p = v^j_{in}\vec{v}^i_{out}$ $(i \neq j)$ or $e_p = v^j_{out}\vec{v}^i_{out}$. Therefore $e_p \in A''_1$ and it is *red*. We have $v^i_{out} \neq v^n_{out}$ since $v^n_{in}\vec{v}^n_{out}$ is the unique arc coming into v^n_{out}. Then, the path must contain an arc $e_{p+1} \notin A''_1 \cup A''_2$ going out of v^i_{out} but all arcs of $A'_1 \cup A'_2$ going out of v^i_{out} are *red* since they belong to A'_1, contradiction.

•

We deduce from Property 2 that any directed PEC path from v^1_{in} to v^n_{out} in T^c only uses arcs of $A'_1 \cup A'_2$. Thus, D admits a directed Hamiltonian path from $s' = v^1$ to $v^n = t'$, if and only if, T^c has a directed PEC Hamiltonian path from $s = v^1_{in}$ to $t = v^n_{out}$. □

We now solve a weaker version of an open problem raised in [7, 30].

Theorem 13. *Deciding whether a 2-edge-colored tournament T^c contains a* PEC *circuit visiting a given vertex s of T^c is **NP**-complete.*

Proof: We start from the 2-edge-colored digraph $D^c = (V', A')$ built in Theorem 9 and we complete it in order to construct a tournament T^c. The idea is to get a tournament whose PEC circuits passing through s (if one exists) also visit vertex t. Then, directed paths from v to w in D (visiting at most one vertex from each pair of C), instance of the Path with Forbidden Pairs Problem, correspond to PEC circuits passing through s in T^c and vice-verse.

Recall that in the construction of D^c (see the proof of Theorem 9), we replace each arc $e \in A$ (resp., e from A'_2), except those which are incident to t, by a directed path of length two in A (resp., in A'_2) where the added vertex is denoted by v_e. If $e \in A$ (resp. $e \in A'_2$) then we suppose that $v_e \in V(A)$ (resp., $v_e \in V(A'_2)$).

Now, we show how to build the tournament T^c. The construction is done in four steps:

(1) Build a set of arcs A'_3 as follows. Add a *red* arc $\vec{ts}$ and a *blue* arc $\vec{us}$. Do $E(D^c) \leftarrow E(D^c) \cup A'_3$. Then, add a *blue* arc $\vec{tx}$ for each $x \notin N_{D^c}(t)$, a *blue* arc $\vec{xu}$ for each $x \notin N_{D^c}(u)$ and a *blue* arc $\vec{xs}$ for each $x \notin N_{D^c}(s)$. Do $E(D^c) \leftarrow E(D^c) \cup A'_3$.

(2) Build a set of arcs A'_4 as follows. Choose an arbitrary vertex v_e of $V(A)$ (resp., $V(A'_2)$) with an incoming *blue* (resp., *red*) arc $y\vec{v}_e$ (resp., $a_i\vec{v}_e$ or $\vec{b_iv_e}$), and add a *blue* (resp., *red*) arc $\vec{v_ex}$ for every $x \notin N_{D^c}(v_e)$. Let A'_4 be this new set of arcs and do $E(D^c) \leftarrow E(D^c) \cup A'_4$. Repeat the process for the remaining vertices v_e of $V(A)$ (resp., $V(A'_2)$) by following an arbitrary order.

(3) Build a set of *blue* arcs $A'_5 = \{\vec{a_qx} : \forall x \notin N_{D^c}(a_q)\} \cup \{\vec{b_qy} : \forall y \notin (N_{D^c}(b_q) \cup \{a_q\})\}$. Recall that (a_q, b_q) is the last pair of C. Set $E(D^c) \leftarrow E(D^c) \cup A'_5$.

(4) Build a set A'_6 of *blue* arcs with endpoints in $V(D^c) \setminus (\{s, u, t, a_q, b_q\} \cup \{v_e : v_e \in V(A) \cup V(A'_2)\})$ and arbitrary directions. Set $E(D^c) \leftarrow E(D^c) \cup A'_6$.

The construction is completed. It is clearly done within polynomial time, and T^c is a 2-edge-colored tournament. We now give some useful properties:

Property 3. *The following properties hold:*

(i) Any PEC *circuit passing through s (resp., u) in T^c uses $\vec{ts}$ and one arc among $\{s\vec{a}_1, s\vec{b}_1\}$ (resp., uses exactly one arc among $\{\vec{a_qu}, \vec{b_qu}\}$ and one arc $u\vec{v}_e \in A$).*

(ii) No PEC *circuit passing through s in T^c uses an arc of A'_4.*

(iii) No PEC *circuit passing through s in T^c uses an arc of $A'_5 \cup A'_6$.*

Proof: For (i). Due to step (1) of the above procedure, there is a unique *red* arc incident to s (resp., t) which is $\vec{ts}$. Thus, any PEC circuit passing through s also visits t. Moreover, vertex s only has two outgoing arcs ($x\vec{a}_1$ and $x\vec{b}_1$ which are colored *blue*. Concerning vertex u, $\vec{a_q u}$ and $\vec{b_q u}$ are the only *red* arcs incident to u. Thus, if a PEC circuit visits u then it contains one of these two arcs as incoming arc and one arc $u\vec{v}_e \in A$ as outgoing arc. Actually, vertex u has only arcs $u\vec{v}_e \in A$ and $\vec{us}$ like outgoing arcs and such a PEC circuit cannot use the *blue* arc $\vec{us}$ since all arcs going out of s are *blue*.

For (ii). By contradiction, assume that there is a PEC circuit passing through s, $\rho = (v_0, e_0, v_1, e_1, \dots, e_k, v_{k+1})$ with $v_1 = v_{k+1} = s$ and containing some arcs of A'_4. Consider the first arc $e_p \in A'_4$ met when we walk around ρ (*i.e.*, $e_q \notin A'_4$ for $q = 1, \dots, p-1$). By construction $e_p = \vec{v_e x}$ and from (i), we deduce $k > p > 1$ (*i.e.*, $x \notin \{s, t\}$). Since $e_{p-1} \notin A'_4$ and $e_{p-1} \notin A'_3$ from (i), arc $e_{p-1} = y\vec{v}_e \in A \cup A'_2$. Thus, e_{p-1} has the same color as e_p, which is a contradiction.

For (iii). By contradiction. Firstly assume that there is a PEC circuit passing through s, $\rho = (v_0, e_0, v_1, e_1, \dots, e_k, v_{k+1})$ with $v_1 = v_{k+1} = s$ and containing some arcs of A'_5. In the same way as before, consider the first arc $e_p \in A'_5$ of ρ (*i.e.*, $e_q \notin A'_5$ for $q = 1, \dots, p-1$). Without loss of generality, suppose $e_p = \vec{a_q x}$ (the same result holds for $e_p = \vec{b_q x}$; we get $x \neq u$ from (i). Then, $e_{p-1} = v_e\vec{a}_q \in A$ is colored in *red* and from (ii) we deduce that $e_{p-2} = y\vec{v}_e \in A$ and is colored in *blue*. Since all arcs in A'_6 are *blue* like e_{p-2}, by induction we deduce that $e_q \in A$ for $q = 1, \dots, p-1$. We obtain a contradiction since from (i) $e_1 \in A'_1$ (*i.e.*, $e_1 \in \{s\vec{a}_1, s\vec{b}_1\}$).

Now, suppose that a PEC circuit passing through s, $\rho = (v_0, e_0, v_1, e_1, \ldots, e_k, v_{k+1})$ with $v_1 = v_{k+1} = s$ contains some arcs in A'_6. Consider the last arc $e_p = \vec{xy} \in A'_6$ met when we walk around ρ (*i.e.*, $e_q \notin A'_6$ for $q = p+1, \ldots, k+1$). Since e_p is colored in *blue* and $y \neq t$ (from (i)), we deduce that e_{p+1} is colored in *red*. Then, we get $y = a_i$ or $y = b_i$ and $e_{p+1} = \vec{yv_e} \in A'_2$ since $e_{p+1} \notin A'_6$. Moreover, from (ii), $e_{p+2} = \vec{v_ez} \in A'_2$ is colored in *blue*. Now, since $e_k \in A$ (the arc of ρ incoming in vertex t) is also colored in *blue*, the directed PEC subpath of ρ from x to $t = v_k$ must contain arc $\vec{a_qu}$ or $\vec{b_qu}$ (using Property 1 of Theorem 9, it is the only way to flip arcs of A'_2 to arcs of A). Thus, this PEC circuit ρ can be decomposed into three directed PEC paths: ρ_1 from y to u, ρ_2 from u to s (and containing arc $e_{k+1} = \vec{ts}$) and ρ_3 from s to y. In particular, the directed PEC path ρ_3 begins with a *blue* arc (by (i)), only uses arcs in A'_2 and ends by a *blue* arc, which is impossible since ρ_3 does not contain u. Actually, directed path ρ_3 cannot use some arcs of A'_6. We have $e_2 = \vec{x_1v_e} \in A'_2$ with $x_1 \in \{a_1, b_1\}$ (since the arc must be colored in *red*) and using (ii), arc $e_3 = \vec{v_ex_2}$ with $x_2 \in \{a_2, b_2\}$ is colored in *blue*. Thus, $e_4 \notin A'_5 \cup A'_6$. Then, the result follows by induction. Notice that it may exist a PEC circuit containing one arc $e = \vec{xy} \in A'_6$ (but not passing through s). In this case, this PEC circuit is composed of two directed PEC paths ρ_1 from y to u and ρ_2 from u to y: ρ_1 only uses arcs of A'_2 from y to a_q (or b_q) and uses arc $\vec{a_qu} \in A'_1$ (or $\vec{b_qu} \in A'_1$) while ρ_2 only uses arcs of A from u to x and uses arc $e = \vec{xy} \in A'_6$. •

Using Properties 1 and 3, we can easily see that we have a path from u to w in D and visiting at most one vertex from each pair of C, if and only if, we have a PEC circuit passing through s in T^c. □

The Theorem 13 above also holds for an arbitrary number of colors. Thus, we have the following result:

Corollary 9. *Deciding whether a c-edge-colored tournament T^c contains a* PEC *circuit passing through s is* ***NP****-complete, even for $c = \Omega(|V(T^c)|^2)$.*

Proof: Construct a tournament $T_n^{c'}$ with n vertices and $c' = 2$ colors, as described in the proof of Theorem 13 (note that $s \in V(T_n^{c'})$). Now, we can easily define a new tournament $\bar{T}_n^{c'}$ with $I_c \supseteq I_{c'}$ by adding all arcs $\vec{xy}$ with $x \in V(\bar{T}_n^c)$, $y \in V(\bar{T}_n^{c'})$ and arbitrary colors $c(\vec{xy}) \in I_c$. Let $E(T, \bar{T})$ be this new set of arcs. In this way, the resulting tournament T_{2n}^c with vertices $V(T_{2n}^c) = V(T_n^{c'}) \cup V(\bar{T}_n^{c'})$ and arcs $E(T_{2n}^c) = E(T_n^{c'}) \cup E(\bar{T}_n^{c'}) \cup E(T, \bar{T})$ will have respectively, $2n$ vertices and at most $\frac{n(n-1)}{2}$ different arc colors. Therefore, the determination of a PEC circuit in $T_n^{c'}$ (for $c' = 2$) will correspond to the determination of a PEC circuit in T_{2n}^c with $c = \Omega(n^2)$ colors and vice verse. □

Dealing with paths instead of circuits, we get:

Corollary 10. *Deciding whether a 2-edge-colored tournament T^c contains a* PEC *path from s to t is* ***NP****-complete (the result also holds for $c = \Omega(|V(T^c)|^2)$ colors).*

Proof: In the proof of Theorem 13, we have a PEC circuit passing through s, if and only if, we have a directed PEC s-t path in T^c. □

Now, regarding Theorem 12 above, we have the following open problem:

Open Problem 6. *Given a 2-edge-colored tournament T^c. The problem of deciding if T^c contains a directed* PEC *Hamiltonian path (with no fixed extremities s and t) is* ***NP****-complete?*

We conclude by recalling the open problem posed by Gutin, Sudakov and Yeo [30]:

Open Problem 7. *Given a 2-edge-colored tournament T^c. To check whether T^c contains a* PEC *circuit is* ***NP****-complete?*

Chapter 5

Paths, trails and walks with reload costs

In this chapter we deal with paths, trails and walks problems. The goal is to find a path/trail/walk whose total reload cost is minimum.

We deal with the case of finding a minimum reload s-t walk, either with symmetric or asymmetric reload cost matrix. We prove that both cases are polynomial time solvable. Then, we discuss paths and trails with symmetric reload costs. We prove that the minimum reload s-t trail problem can be solved in polynomial time for every $c \geq 2$. Besides, we show that the minimum reload s-t path problem is polynomially solvable either if $c = 2$ and the triangle inequality holds and R is not necessarily a symmetric matrix or if G^c has a maximum degree 3. Although, it is **NP**-hard when $c \geq 3$, even for graphs of maximum degree 4 and reload cost matrix satisfying the triangle inequality, as well as if $c \geq 4$ and the triangle inequality is satisfied, the minimum symmetric reload s-t path problem remains **NP**-hard even for planar graphs with maximum degree 4. We also show that the TSP with reload costs is **NP**-hard and no non-trivial approximation is likely to exist, even if $c = 2$ the reload cost matrix is symmetric and satisfies the triangle inequality. Last, we deal with asymmetric reload costs. For a reload cost matrix satisfying the triangle inequality, we construct a polynomial time procedure for the minimum reload s-t trail problem and we prove that the minimum asymmetric reload s-t trail problem is **NP**-hard even for graphs with 3 colors and maximum degree equal to 3.

5.1 Walks with reload costs

Choosing a walk instead of a path can help in reducing the reload costs. For instance, Figure 5.1 illustrates two different s-t walks, $\rho_1 = (s, e_1, v_1, e_2, v_2, e_2, v_1, e_3, t)$ and $\rho_2 = (s, e_1, v_1, e_3, t)$, with reload costs $r_{i,j} = 1$ for $i, j \in \{1, 2, 3\}$ except for $r_{1,3} = r_{3,1} = 4$. The reload cost of ρ_2 is $r(\rho_2) = r_{1,3} = 4$ whereas the reload cost of ρ_1 is $r(\rho_1) = r_{1,2} + r_{2,2} + r_{2,3} = 3$. Notice that the minimum reload cost of an s-t walk is a lower bound on the minimum reload cost of an s-t trail which is a lower bound on the minimum reload cost of an s-t path since a path is a trail and a trail is a walk.

We already know that the minimum reload s-t walk problem is polynomial since there is a polynomial reduction from the minimum reload s-t walk problem to the minimum reload+weight directed s-t trail problem (see Subsection 1.2 for a description of this problem). Actually, from G^c, c, I_c and a reload cost matrix $R = [r_{i,j}]$, an instance of the minimum reload s-t walk problem, we build an instance D^c, c' I'_c, w and a reload cost matrix $R' = [r'_{i,j}]$ of the minimum reload+weight directed s-t trail problem as follows: $V(D^c) = V(G^c)$ and we replace each edge $e = v_i v_j$ of G^c by two arcs $e_1 = v_i\vec{v}_j$ and $e_2 = v_j\vec{v}_i$ with color $c'(e_1) = c'(e_2) = c(e)$. Thus, $I'_c = I_c$. Finally, $r'_{i,j} = r_{i,j}$ for $i, j \in I_c$ and $w(e) = 0$ for every arc $e \in \overrightarrow{E}(D^c)$. It is not difficult to see that any directed s-t trail ρ of D^c with reload+weight cost $r'(\rho) + w(\rho)$ corresponds to an s-t walk ρ_c of G^c with reload cost $r(\rho_c) = r'(\rho) + w(\rho)$. On the other side, any optimal s-t walk ρ_c^* of G^c using a minimum number of edges can be converted into a directed s-t trail ρ^* of D^c with reload+weight cost $r'(\rho^*) + w(\rho^*) = r(\rho_c^*)$.

$$R = \begin{pmatrix} 1 & 1 & 5 \\ 1 & 1 & 1 \\ 5 & 1 & 1 \end{pmatrix}$$

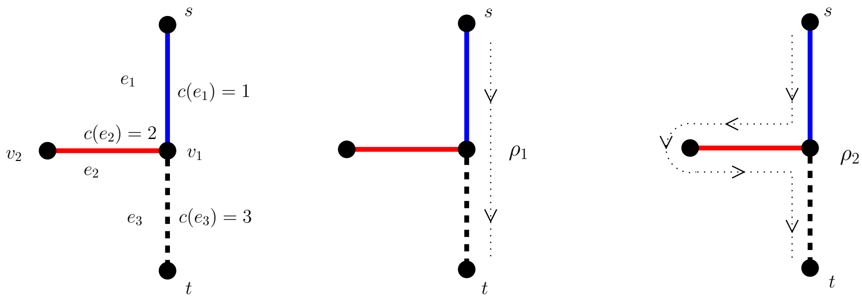


Figure 5.1: Two different reload s-t walks and the associated reload cost matrix R. Walk ρ_1 has reload cost 5 and ρ_2 has reload cost 3.

Here, we propose another polynomial method to solve the minimum reload s-t walk problem. Notice that the construction used differs from the one given in [3] for solving the minimum reload+weight directed s-t trail problem.

Let G^c with $V(G^c) = \{s, t\} \cup \{v_1, \ldots, v_n\}$ be a simple c-edge-colored connected graph. We reduce the minimum s-t walk problem to the computation of a shortest s_0-t_0 path in an auxiliary digraph $H = (V', \overrightarrow{E'})$ whose arcs are weighted by w. The digraph H contains $|I_c|$ directed subgraphs H_ℓ for $\ell \in I_c$. The vertex set of each subgraph H_ℓ is $\{v_1^\ell, \ldots, v_n^\ell\}$. There is an arc from v_i^ℓ to $v_k^{\ell'}$, if and only if, there is a walk $(v_j, e_1, v_i, e_2, v_k)$ in G^c such that $c(e_1) = \ell$ and $c(e_2) = \ell'$. This construction can be done within polynomial time. An example is given in Figure 5.2.

Formally, the digraph H is built as follows:

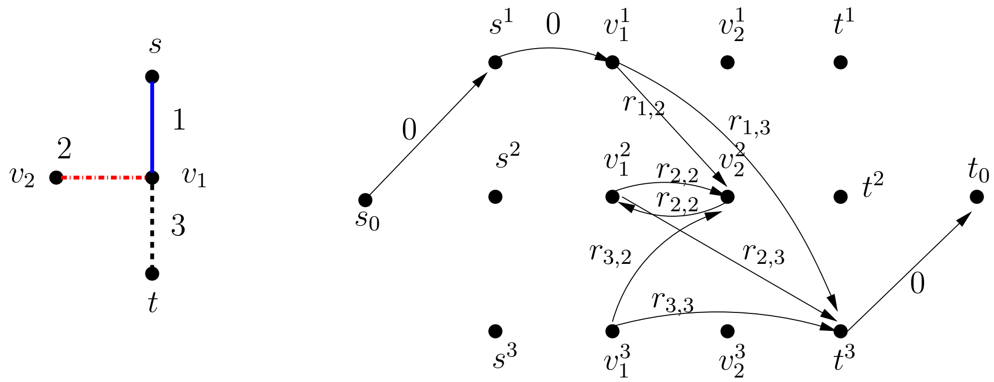


Figure 5.2: Transformation of G^c into a digraph H.

- $V' = \{s_0, t_0\} \cup \{s^\ell, v_1^\ell, \ldots, v_n^\ell, t^\ell : \ell \in I_c\}$

- For any pair of edges $v_j v_i \in E^\ell(G^c)$ and $v_i v_k \in E^{\ell'}(G^c)$, with $\ell, \ell' \in I_c$ and $v_i \in V(G^c)$ (possibly with $v_j = v_k$), add arcs $\overrightarrow{v_i^\ell v_k^{\ell'}}$ and $\overrightarrow{v_i^{\ell'} v_j^\ell}$ to $\overrightarrow{E'}$. Next update $\overrightarrow{E'}$ by deleting all incoming (resp., outgoing) arcs to s^ℓ (resp., to t^ℓ) for every $\ell \in I_c$. Moreover, add arc $\overrightarrow{s_0 s^\ell}$ to $\overrightarrow{E'}$ (resp., $\overrightarrow{t^\ell t_0}$ to $\overrightarrow{E'}$), if and only if, there exists $sv_i \in E^\ell(G^c)$ (resp., $v_i t \in E^\ell(G^c)$).

- If $v_i \neq s, t$ and $v_j \neq s, t$, then $w(\overrightarrow{v_i^{\ell'} v_j^\ell}) = r_{\ell',\ell}$ for arc $\overrightarrow{v_i^{\ell'} v_j^\ell} \in \overrightarrow{E'}$. If $v_i \in \{s, t\}$ or $v_j \in \{s, t\}$, then $w(\overrightarrow{v_i^{\ell'} v_j^\ell}) = 0$ for arc $\overrightarrow{v_i^{\ell'} v_j^\ell} \in \overrightarrow{E'}$. Finally, $w(\overrightarrow{s_0 s^\ell}) = 0$ for arc $\overrightarrow{s_0 s^\ell} \in \overrightarrow{E'}$ and $w(\overrightarrow{t^\ell t_0}) = 0$ for arc $\overrightarrow{t^\ell t_0} \in \overrightarrow{E'}$.

Theorem 14. *For any simple connected edge-colored graph G^c and any pair s, t of vertices of G^c, the minimum reload s-t walk problem can be solved in polynomial time.*

Proof: Let G^c with $V(G^c) = \{s, t\} \cup \{v_1, \ldots, v_n\}$ be a simple edge-colored connected graph with colors in I_c. We apply the transformation described above. Now, observe that any directed path ρ' from s_0 to t_0 in H with weight $w(\rho') = \sum_{e \in \rho'} w(e)$

corresponds in G^c to an s-t walk ρ_c with reload cost $r(\rho_c) = w(\rho')$. Symmetrically any minimum reload s-t walk ρ_c^* of G^c with reload cost $r(\rho_c^*)$ and using a minimum number of edges can be converted into a directed path ρ' from s_0 to t_0 in H such that $w(\rho') = r(\rho_c^*)$. Actually, in order to prove this claim we need to show that the directed path ρ' will not pass twice by vertex v^ℓ for each $v \in V(G^c)$ and $\ell \in I_c$. This latter property holds because we have:

Property 4. *If ρ_c^* is a minimum reload s-t walk of G^c using a minimum number of edges, then ρ_c^* does not contain a subsequence $(e_0, v, e_1, \ldots, e_k, v, e_{k+1})$ with $c(e_0) = c(e_k)$ or $c(e_1) = c(e_{k+1})$.*

Proof: We will show Property 4 by contradiction. Let ρ_c^* be a minimum reload s-t walk of G^c using a minimum number of edges and assume that ρ_c^* contains a subsequence $(e_0, v, e_1, \ldots, e_k, v, e_{k+1})$ with $c(e_0) = c(e_k)$ or $c(e_1) = c(e_{k+1})$. Let ρ_c' be the walk in which the subsequence $(e_0, v, e_1, \ldots, e_k, v, e_{k+1})$ is replaced by (e_0, v, e_{k+1}). In this case, the sequence ρ_c' is an s-t walk in G^c with reload cost $r(\rho_c') \leq r(\rho_c^*)$, contradiction with the definition of ρ_c^*. Thus, we deduce that ρ_c^* can be converted into an oriented path from s_0 to t_0 in H since this path will pass through vertices $v^{c(e_0)}$ and $v^{c(e_k)}$ which are different. Notice that Property 4 also implies that ρ_c^* contains at most twice the same edge and if an edge e appears twice in ρ_c^* then it is used in both directions (see for instance the walk ρ_1 in Figure 5.1. This figure illustrates two different s-t walks, $\rho_1 = (s, e_1, v_1, e_3, t)$ and $\rho_2 = (s, e_1, v_1, e_2, v_2, e_2, v_1, e_3, t)$, with reload costs $r_{i,j} = 1$ for $i, j \in \{1, 2, 3\}$ except for $r_{1,3} = r_{3,1} = 4$. The reload cost of ρ_1 is $r(\rho_1) = r_{1,3} = 4$ whereas the reload cost of ρ_2 is $r(\rho_2) = r_{1,2} + r_{2,2} + r_{2,3} = 3$). •

In conclusion, a shortest directed path from s_0 to t_0 in H corresponds to a minimum reload s-t walk in G^c and thus it can be computed within polynomial time. $\square$

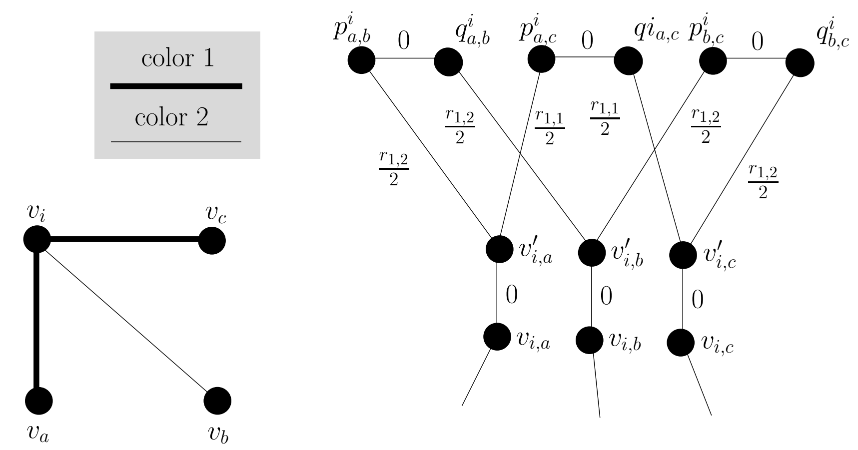
5.2 Paths and trails with symmetric reload costs

Let R be a symmetric matrix with non-negative integer reload costs. Here, we prove that the minimum reload s-t trail problem can be solved in polynomial time for every $c \geq 2$. In addition, we show that the minimum reload s-t path problem can be solved in polynomial time either if $c = 2$ and the triangle inequality holds (here R is not necessarily a symmetric matrix) or if G^c has a maximum degree 3. However the problem is **NP**-hard when $c \geq 3$ for graphs satisfying the triangle inequality and with maximum degree equal to 4. We conclude the section by showing that, if $c \geq 4$ and the triangle inequality is satisfied, the minimum reload s-t path problem remains **NP**-hard even for planar graphs with maximum degree 4.

In the sequel, we show how to turn the minimum reload s-t trail problem into a minimum perfect matching problem in a weighted non-colored graph G defined as follows.

Given two vertices s and t in $V(G^c) = \{v_1, \ldots, v_n\}$, set $W = V(G^c) \setminus \{s, t\}$. Now, for each $v_i \in W$, we define a subgraph G_i with vertex and edge sets as illustrated in Figure 5.3. Formally:

- $V(G_i) = \{v_{i,j}, v'_{i,j} : v_j \in N_{G^c}(v_i)\} \cup \{p^i_{j,k}, q^i_{j,k} : j < k \text{ and } v_j, v_k \in N_{G^c}(v_i)\}$
- $E(G_i) = \{v_{i,j}v'_{i,j} : v_j \in N_{G^c}(v_i)\} \cup \{v'_{i,j}p^i_{j,k}, p^i_{j,k}q^i_{j,k}, q^i_{j,k}v'_{i,k} : j < k \text{ and } v_j, v_k \in N_{G^c}(v_i)\}$



(a) Neighborhood of v_i in G^c (b) Weighted non-colored subgraph G_i

Figure 5.3: Reduction of the minimum reload s-t trail to a minimum perfect matching.

The non-colored graph $G = (V', E')$ edge weighted by w is constructed as follows:

- $V' = \{s', t'\} \cup (\bigcup_{v_i \in W} V(G_i))$, and
- $E' = \{v_{i,j}v_{x,y} : j = x \text{ and } i = y\} \cup \{s'v_{i,j} : v_j = s \text{ and } v_iv_j \in E(G^c)\} \cup \{v_{i,j}t' : v_j = t \text{ and } v_iv_j \in E(G^c)\}$
- $w(v'_{i,j}p^i_{j,k}) = \frac{1}{2}r_{c(v_iv_j),c(v_iv_k)}$, $w(v'_{i,k}q^i_{j,k}) = \frac{1}{2}r_{c(v_iv_k),c(v_iv_j)}$ and all remaining edges have a weight 0.

After G is constructed, we have to find a minimum weighted perfect matching M^* in G. The weight of matching M is $w(M) = \sum_{e \in M} w(e)$ (computing a minimum weighted perfect matching is polynomial, see [26] for a good reference on general matchings). We can prove that perfect matchings in G will be associated with reload s-t trails in G^c and vice-verse. Formally:

Theorem 15. *For any simple connected edge-colored graph G^c and any pair s, t of vertices of G^c, the minimum reload s-t trail problem can be solved in polynomial time.*

Proof: From G^c, an instance of the the minimum symmetric reload s-t trail problem, we polynomially build a weighted undirected graph $G = (V', E')$ as indicated above (see Figure 5.3). Let M be a weighted perfect matching in G with weight $w(M) = \sum_{e \in M} w(e)$. The associated reload s-t trail ρ_c in G^c can be obtained after the contraction of all subgraphs G_i in G and by associating the remaining non-colored edges with colored edges in G^c. Since the reload cost matrix is symmetric and $w(v'_{i,j}p^i_{j,k}) + w(v'_{i,k}q^i_{j,k}) = r_{c(v_iv_j),c(v_iv_k)}$ we can easily see that $w(M) = r(\rho_c)$.

Conversely, given an s-t trail ρ_c of G^c with reload cost $r(\rho_c)$, we construct the associated perfect matching M in the following manner: (a) for every vertex v_i of G^c, out of ρ_c, we choose all edges with weight 0 in G_i; and (b), for every vertex v_i of G^c, belonging to ρ_c, if ρ_c contains the subsequence (v_a, e, v_i, e', v_b) with $e \neq e'$, we choose edges $v'_{i,a}p^i_{a,b}$ and $q^i_{a,b}v'_{i,b}$ (we assume $a < b$) of G_i; and finally, (c) we choose all the remaining edges of G (with cost 0), in order to obtain a perfect matching of G. In this way, it is easy to see that $w(M) = r(\rho_c)$. Therefore, a minimum reload s-t trail corresponds in G to a minimum weighted perfect matching. Note that the complexity of the minimum reload s-t trail is dominated by the complexity of the minimum perfect matching problem in G. Since the construction of each G_i depends on the number of neighbors of v_i, we can say that a minimum reload s-t can be obtained in polynomial time in the size of G^c.

In Figure 5.4 we show a cubic edge-colored-graph and its associated non-colored graph.

□

Corollary 11. *For any simple connected edge-colored graph G^c of maximum degree*

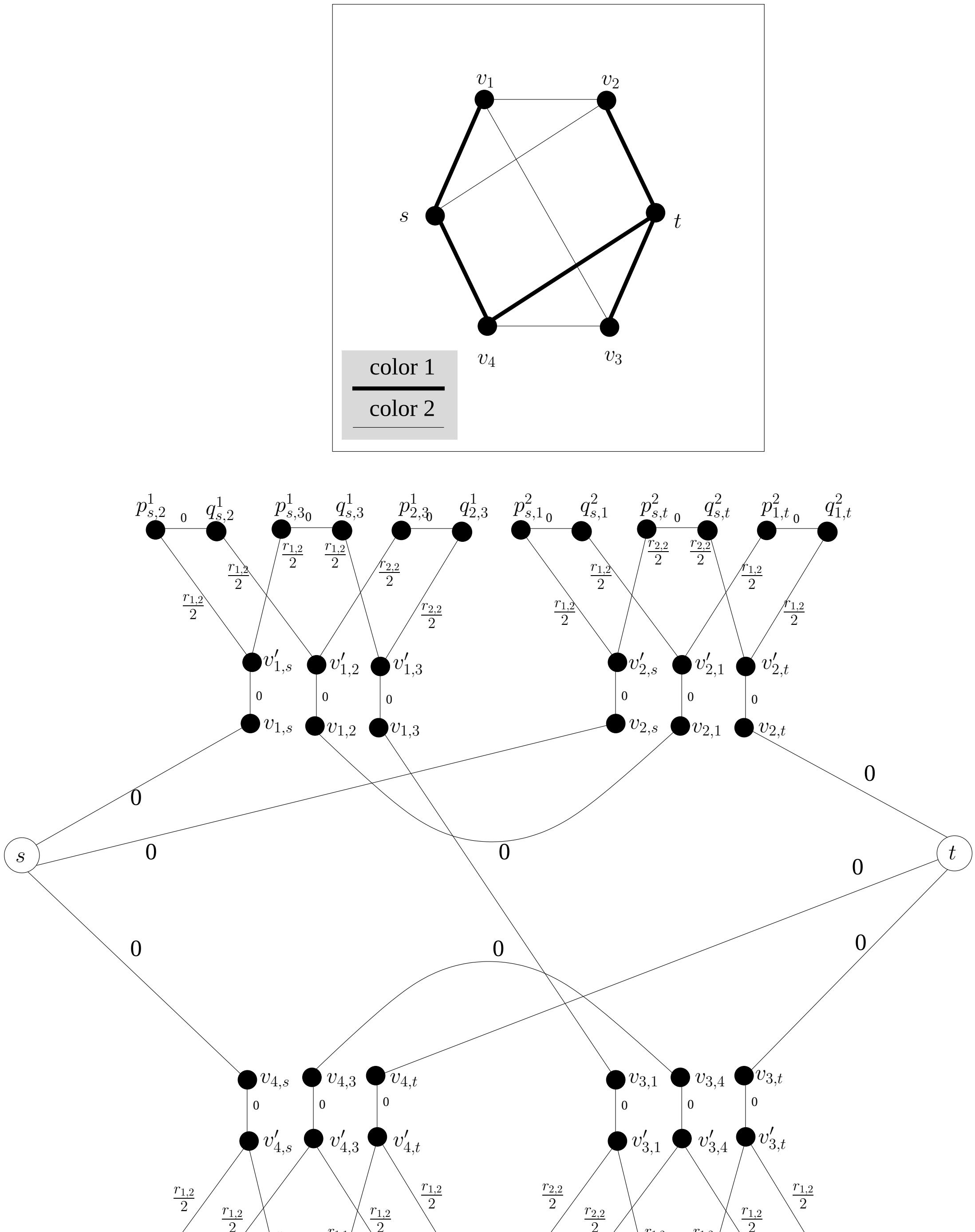


Figure 5.4: A 2-edge-colored graph G^c (top). Associated Weighted non-colored graph G (bottom).

3 and any pair s, t of vertices of G^c, the minimum symmetric reload s-t path problem can be solved in polynomial time.

Proof: The result is obvious, since in graphs of maximum degree 3, a minimum s-t trail is an s-t path. The reload cost matrix being symmetric, one can apply Theorem 15. □

Now, we deal with graphs G^c colored with two colors. We show that the minimum reload s-t path problem is polynomial if the reload cost matrix R satisfies the triangle inequality (R is not necessarily symmetric).

Theorem 16. *Let G^c be a simple connected edge-colored graph with $c = 2$ colors, such that the associated matrix R of reload costs satisfies the triangle inequality. For any pair s, t of vertices of G^c, the minimum reload s-t path problem can be solved in polynomial time.*

Proof: Let $G^c = (V, E)$ with $I_c = \{1, 2\}$ be an instance of the minimum reload s-t path problem. We also assume that the reload cost matrix $R = [r_{i,j}]$ satisfies the triangle inequality. Here, R is not necessarily symmetric. We first show that any minimum reload s-t walk of G^c using a minimum number of edges is an s-t path of G^c. Let ρ_c^* be a minimum reload s-t walk of G^c using a minimum number of edges and assume that ρ_c^* passes twice through some vertices. Consider the first vertex v visited twice by ρ_c^*. This means that ρ_c^* contains the subsequence $C = (v_0, e_0, v, e_1, \ldots, e_k, v, e_{k+1}, v_k)$ (see Figure 5.5 for an illustration). Let ρ_c' be the s-t walk in which the subsequence C is replaced by $(v_0, e_0, v, e_{k+1}, v_k)$. We show that $r(\rho_c') \leq r(\rho_c^*)$ which leads to a contradiction since $|\rho_c'| < |\rho_c^*|$. We consider two cases:

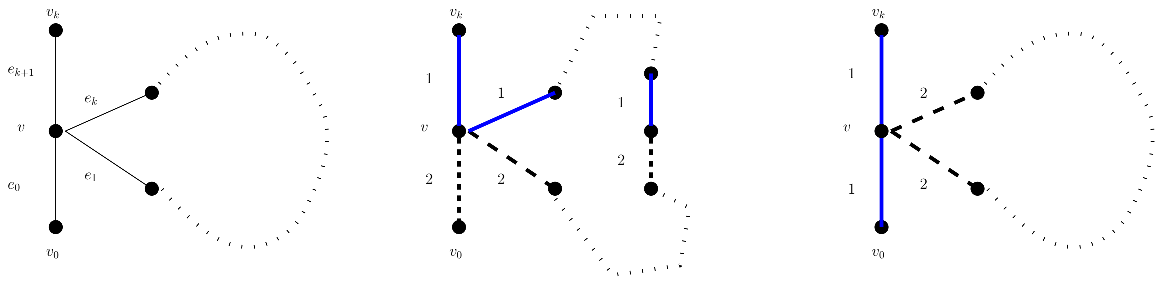


Figure 5.5: Some cases for the subsequence $C = (v_0, e_0, v, e_1, \ldots, e_k, v, e_{k+1}, v_k)$.

- $c(e_1) \neq c(e_k)$. If $c(e_0) = c(e_{k+1})$ then $r_{c(e_0),c(e_{k+1})} \leq r_{c(e_0),c(e_1)} + r_{c(e_k),c(e_{k+1})}$ (recall that $|I_c| = 2$); thus $r(\rho'_c) \leq r(\rho^*_c)$ and we get a contradiction. So, $c(e_0) \neq c(e_{k+1})$ and moreover $c(e_0) = c(e_1)$ for the same reasons. Now, since $|I_c| = 2$, there exists $i \in \{2, \ldots, k\}$ such that $c(e_1) = c(e_{i-1}) \neq c(e_i)$. We deduce that $r(\rho'_c) \leq r(\rho^*_c)$. See case A of Figure 5.5.

- Now, assume $c(e_1) = c(e_k)$. Since edges e_0, e_k, e_{k+1} are adjacent to a common vertex v, by applying the triangle inequality we obtain $r_{c(e_0),c(e_{k+1})} \leq r_{c(e_0),c(e_k)} + r_{c(e_k),c(e_{k+1})} = r_{c(e_0),c(e_1)} + r_{c(e_k),c(e_{k+1})}$. Thus, $r(\rho'_c) \leq r(\rho^*_c)$. See case B of Figure 5.5.

In conclusion, any optimal reload s-t walk of G^c using a minimal number of edges is an s-t path.

Finally, we apply the transformation made in Theorem 14 from instance G^c with $|I_c| = 2$ except that we replace $w(e)$ by $w'(e) = (2m + 1)w(e) + 1$ for each arc e of H. Let ρ' be a shortest directed $s_0 - t_0$ path in H with weight $w'(\rho')$. The path ρ' corresponds in G^c to an optimal reload s-t walk ρ'_c of G^c using a minimum number of edges. This conclude the proof. Otherwise, let ρ^*_c be an optimal reload

s-t walk of G^c using a minimum number of edges $|\rho_c^*| < |\rho_c'|$. We have $|\rho_c^*| \leq 2m$ since any edge of G^c is used at most twice (see Property 4 of Theorem 14). The sequence ρ_c^* corresponds to a directed path ρ^* in H with weight $w'(\rho^*) = (2m + 1)w(\rho^*) + |\rho_c^*| + 2 = (2m+1)r(\rho_c^*) + |\rho_c^*| + 2$. We deduce $r(\rho_c^*) = r(\rho_c')$ since otherwise $w(\rho^*) = r(\rho_c^*) \leq r(\rho_c') - 1 = w(\rho') - 1$ ($r_{i,j} \in \mathbb{N}$) and then $w'(\rho^*) \leq (2m+1)(w(\rho') - 1) + |\rho_c^*| + 2 < (2m+1)w(\rho') + |\rho_c'| + 2 = w'(\rho')$ (recall that $|\rho_c^*| \leq 2m$). Thus, $w'(\rho^*) = (2m+1)w(\rho^*) + |\rho_c^*| + 2 < (2m+1)w(\rho') + |\rho_c'| + 2 = w'(\rho')$, which is a contradiction since ρ' is assumed to be a shortest directed $s_0 - t_0$ path of H. $\square$

A possible application of Theorem 16 is the following. Consider a $(2,2)$-matrix R satisfying $r_{1,1} = r_{2,2} = 0$. It is easy to see that R satisfies the triangle inequality, and then one can apply Theorem 16 (on the other hand, this restriction becomes **NP**-hard for a $(3,3)$-matrix with $r_{i,i} = 0$, see the proof of item (i) of Corollary 12). We also deduce that the minimum toll cost s-t path problem (see Subsection 1.2) is polynomial for two colors since it is a subproblem of the case considered above. Notice that, the minimum toll cost s-t path problem for $r_j = 1\ \forall j \in I_c$, is equivalent to minimizing the number of color changes in an s-t path. Actually, the minimum toll cost s-t path problem is polynomially solvable (without constraints on the number of colors).

Theorem 17. *Let G^c be a simple connected edge-colored graph and s and t be any pair of vertices of $V(G^c)$. The minimum toll cost s-t path problem can be solved in polynomial time.*

Proof: The proof is quite identical to Theorem 16. Let $R = [r_{i,j}]$ be a reload cost matrix satisfying $r_{j,j} = 0$ and $r_{i,j} = r_j$ if $i \neq j$. We only show that any minimum reload s-t walk of G^c using a minimum number of edges is an s-t path of G^c. Let ρ_c^* be a minimum reload s-t walk of G^c using a minimum number of edges and assume that ρ_c^* contains the subsequence $C = (v_0, e_0, v, e_1, \ldots, e_k, v, e_{k+1}, v_k)$ (possibly with $e_1 = e_k$). Let ρ_c' be the s-t walk in which the subsequence C is replaced by $C' = (v_0, e_0, v, e_{k+1}, v_k)$. If $c(e_0) = c(e_{k+1})$, then $r(C') = 0 \leq r(C)$. If $c(e_0) \neq c(e_{k+1})$, then $r(C') = r_{c(e_0),c(e_{k+1})} = r_{c(e_{k+1})} = r_{c(e_k),c(e_{k+1})} \leq r(C)$. The rest of the proof is similar to proof of Theorem 16. □

Now, we show that the previous restrictions on G^c are almost the best ones to obtain polynomial cases for the minimum reload s-t path problem.

Theorem 18. *The minimum symmetric reload s-t path problem is* **NP***-hard if $c \geq 3$, the triangle inequality holds and the maximum degree of G^c is equal to 4.*

Proof: Given a set $\mathcal{C}$ of CNF clauses defined over a set $\mathcal{X}$ of boolean variables. An instance of the $(3, B2)$-SAT problem, called 2-balanced 3-SAT, is such that each clause has exactly 3 literals, each of them appearing exactly 4 times in the clauses, twice negated and twice unnegated. Deciding whether an instance of $(3, B2)$-SAT is satisfiable is **NP**-complete [10]. We are going to reduce $(3, B2)$-SAT to the existence of an s-t path with reload cost at most L. Let $\mathcal{I}$ be an instance of $(3, B2)$-SAT. We say that C_j is the h-th clause of x_i, if and only if, x_i appears in C_j and x_i appears in exactly $h-1$ other clauses $C_{j'}$ with $j' < j$. We say that x_i is the ℓ-th variable of C_j, if and only if, x_i and exactly $\ell - 1$ other variables $x_{i'}$ with $i' < i$ appear in C_j. We build G^c – instance of the s-t path with reload cost at most L – as follows. We have $I_c = \{1, 2, 3\}$ and $L = 11|\mathcal{X}| + 3|\mathcal{C}|$. The matrix R is defined as $r_{1,2} = r_{2,1} = M$ where $M > L$. The other entries of R are set to 1.

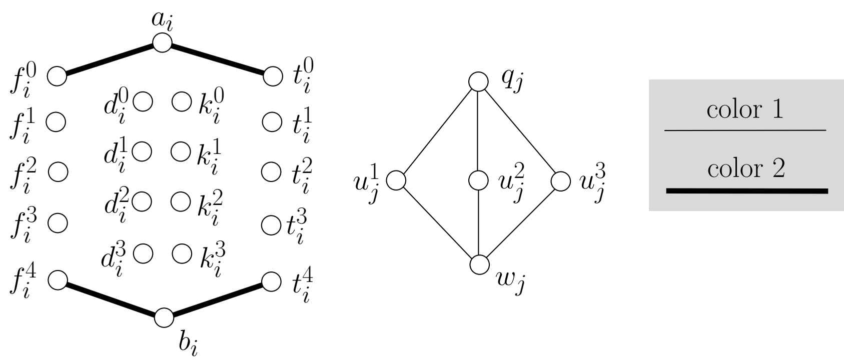


Figure 5.6: Gadgets for a variable x_i (left) and a clause C_j (right).

The graph G^c has a source vertex s and a sink vertex t. In addition, for each $x_i \in \mathcal{X}$ (resp. $\mathcal{C}_j \in \mathcal{C}$) we build a gadget as depicted on the left (resp. right) of Figure 5.6. The gadget of a variable x_i consists of a left part (vertices f_i^0 to f_i^4 and d_i^0 to d_i^3), a right part (vertices t_i^0 to t_i^4 and k_i^0 to k_i^3), an entrance a_i and an exit b_i. The left (resp. right) part corresponds to the case where x_i is set to FALSE (resp. TRUE). The gadget of a clause C_j consists of an entrance q_j, an exit w_j and three vertices u_j^1, u_j^2, and u_j^3 which correspond to the first, second and third variable of C_j respectively. We link the gadgets by adding the following edges (see Figure 5.7):

- sa_1,b_1a_2,b_2a_3,...,$b_{|\mathcal{X}|-1}a_{|\mathcal{X}|}$ with color 2 (bold) ;

- $b_{|\mathcal{X}|}q_1$ with color 3 (dashed) ;

- $w_1q_2]$,w_2q_3,...,$w_{|\mathcal{C}|-1}q_{|\mathcal{C}|}$, $w_{|\mathcal{C}|}t$ with color 1 (thin).

For each pair x_i, C_j such that x_i is the ℓ-th variable of C_j and C_j is the h-th clause of x_i we proceed as follows. If x_i appears negated in C_j then add $t_i^{h-1}u_j^\ell$, $t_i^h u_j^\ell$, $f_i^{h-1}d_i^{h-1}$ and $f_i^h d_i^{h-1}$ with color 2 (bold). If x_i appears unnegated in C_j then add

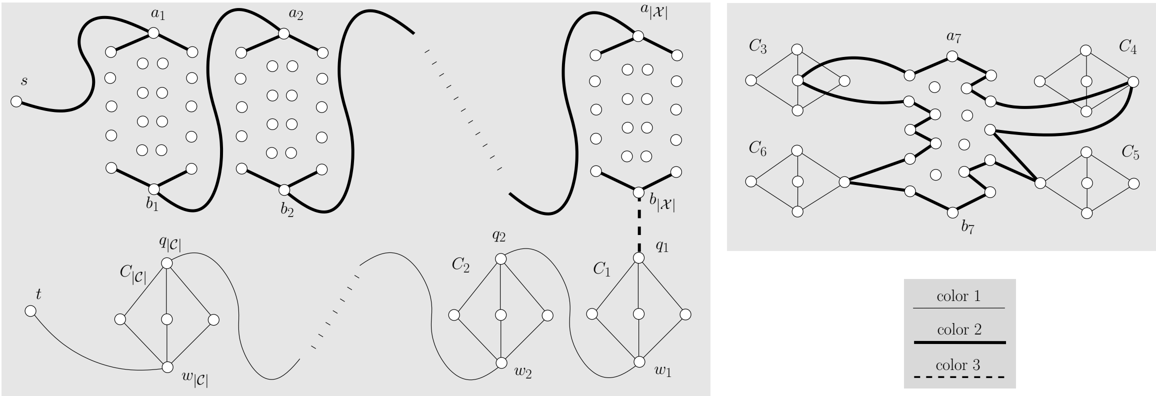


Figure 5.7: Left: How the gadgets are linked. Right: How to link the gadget of x_7 if it appears in $C_3 = (x_1 \vee x_7 \vee x_8)$, $C_4 = (\overline{x}_3 \vee x_5 \vee \overline{x}_7)$, $C_5 = (\overline{x}_7 \vee \overline{x}_8 \vee x_9)$ and $C_6 = (\overline{x}_1 \vee \overline{x}_6 \vee x_7)$.

$f_i^{h-1}u_j^\ell$, $f_i^h u_j^\ell$, $t_i^{h-1}k_i^{h-1}$ and $t_i^h k_i^{h-1}$ with color 2 (bold). It is not difficult to see that each vertex's degree is at most 4. Moreover the triangle inequality holds.

Since $r_{1,2} > L$ and $r_{2,1} > L$, any s-t path ρ_c with reload cost at most L starts at s, enters the gadget of x_1 and visits the variable-gadgets in turn. When $b_{|\mathcal{X}|}$ is reached, ρ_c uses $b_{|\mathcal{X}|}q_1$ and visits the clause-gadgets in turn. Finally t is reached from $w_{|\mathcal{C}|}$. Then exactly 11 (resp., 3) vertices are visited when passing through a variable-gadget (resp., a clause-gadget).

If a truth assignment τ satisfies $\mathcal{I}$ then G^c admits an s-t path ρ_c with reload cost $11|\mathcal{X}|+3|\mathcal{C}|$. Indeed, if x_i is FALSE (resp. TRUE) in τ then ρ_c goes across the left (resp. right) part of x_i's gadget. Since τ satisfies $\mathcal{I}$ we know that at least one literal per clause is true. If the ℓ-th literal of C_j is true (choose ℓ arbitrarily if it is not unique) then ρ_c passes through u_j^ℓ. Conversely an s-t path ρ_c with reload cost $11|\mathcal{X}| + 3|\mathcal{C}|$ induces a truth assignment that satisfies $\mathcal{I}$: set x_i to FALSE (resp. TRUE) if ρ_c passes through the left (resp. right) part of x_i's gadget. $\square$

Corollary 12. *The two following statements hold:*

(i) *In the general case, the minimum symmetric reload s-t path problem is not approximable at all if $c \geq 3$, the triangle inequality holds and the maximum degree of G^c is equal to 4.*

(ii) *If $r_{i,j} \geq 1$ for every $i, j \in I_c$, the minimum symmetric reload s-t path problem is not $O(2^{P(n)})$-approximable for every polynomial P if $c \geq 3$, the triangle inequality holds and the maximum degree of G^c is equal to 4.*

Proof: We show that the reduction built in Theorem 18 is a gap reduction. Let us denote by $OPT(G^c)$ the reload cost of an optimal solution of G^c, the instance built in Theorem 18.

For (i) we modify the reload costs as follows: $r_{2,3} = r_{3,2} = r_{3,1} = r_{1,3} = 0$, $r_{i,i} = 0$ for $i \in I_c = \{1, 2, 3\}$ and $r_{1,2} = r_{2,1} = M$. Notice that the reload cost matrix R is symmetric and satisfies the triangle inequality. We have $OPT(G^c) = 0$, if and only if, $\mathcal{I}$ is satisfiable. Thus, it is **NP**-complete to distinguish between $OPT(G^c) = 0$ and $OPT(G^c) \geq 1$.

For (ii), let P be a polynomial. Set $M = O(2^{P(n)})\, L$ where n is the number of vertices of G^c in the proof of Theorem 18. We deduce that it is **NP**-complete to distinguish between $OPT(G^c) \leq L$ and $OPT(G^c) \geq O(2^{P(n)})\, L$. □

See Subsection 1.2 for a better description of the gap reduction technique.

Corollary 13. *The minimum symmetric reload s-t path problem is* **NP***-hard if $c \geq 4$, the graph is planar, the triangle inequality holds and the maximum degree is equal to 4.*

Proof: We use the instance G^c in the proof of Theorem 18 and make it planar. To do so we use an additional color 4 such that $r_{3,4} = r_{4,3} = M$ and $r_{1,4} = r_{4,1} = r_{2,4} = r_{4,2} = 1$. Let G^c be an embedding of the graph built in Theorem 18. Here $M > n + 3p$ where n (resp. p) is the number of vertices (resp. intersections between two edges) in the graph of G^c. Note that p is polynomially bounded in the order of G^c.

One can suppose Without loss of generality, that $b_{|\mathcal{X}|}q_1$ (with color 3) is not intersected by another edge of G^c (see Figure 5.7). If some edge ab with color 1 intersects cd with color 2 in G^c we add a new vertex f and replace ab by $\{af, fb\}$ with color 1, and edge cd by $\{cf, fd\}$ with color 2. If ab with color 1 (resp., 2) intersects cd with color 1 (resp., 2), we add five new vertices $\{f, a', b', c', d'\}$ and replace ab by $\{aa', b'b\}$ with color 1 (resp., 2), replace cd by $\{cc', d'd\}$ with color 1 (resp., 2), add $\{a'f, fb'\}$ with color 3 and add $\{c'f, fd'\}$ with color 4. In this way, the graph of the resulting instance – denoted by $\bar{G}^c$ – is planar.

It is not difficult to see that $\mathcal{I}$ (the instance of $(3, B2)-$SAT from which G^c is built) is satisfiable iff there is an s-t path ρ_c in $\bar{G}^c$ such that $r(\rho_c) < M$. $\square$

5.2.1 Traveling salesman problem with reload costs

The reload traveling salesman problem is defined upon a complete graph K_n^c on vertices $\{1, \ldots, n\}$ where edges are colored in I_c. The goal is to find a vertex permutation π (*i.e.,* a Hamiltonian cycle) of K_n minimizing its reload cost $r(\pi) = \sum_{i=1}^{n} r_{c(e_i),c(e_{(i+1) mod\, n})}$ with $e_i = (\pi(i), \pi((i+1) mod\, n))$ for $i = 1, \ldots, n$.

Theorem 19. *The reload traveling salesman problem is* **NP**-*hard even if* $c = 2$*, the*

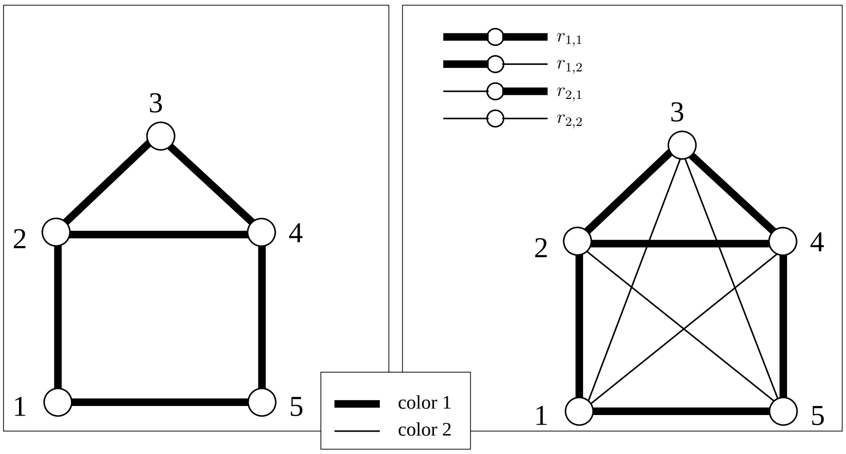


Figure 5.8: Instance of the Hamiltonian Cycle, where all edges are colored 1 (left). Complete graph, instance of the The reload traveling salesman problem (right).

reload cost is symmetric and satisfies the triangular inequality.

Proof: The reduction is very simple and it is done from the Hamiltonian cycle problem (HC in short). This latter problem consists in deciding wether a simple graph G contains an HC. HC is known to be **NP**-complete [25]. Starting from a graph $G = (V, E)$ on n vertices, instance of HC, we complete it into K_n^c where the initial edges (*i.e.*, edges of E) are colored 1 and added edges are colored 2 (see Figure 5.8). We set $r_{1,1} = 1$ and $r_{1,2} = r_{2,1} = r_{2,2} = M$ where $M > n$. Clearly, K_n^c is colored with two colors and the reload cost $r_{i,j}$ for $i, j \in I_c$ is symmetric and satisfies the triangular inequality.

It is clear that G has an HC, if and only, if there is an acyclic permutation π of $V(K_n^c)$ with reload cost $r(\pi) \leq n$.

□

From this theorem, we deduce the following results.

Corollary 14. *The two following statements hold:*

(*i*) *In the general case, the reload traveling salesman problem is not approximable at all even if* $c = 2$*, the reload cost matrix is symmetric and satisfies the triangular inequality.*

(*ii*) *If* $r_{i,j} \geq 1$ *for every* $i, j \in I_c$*, the reload traveling salesman problem is not* $O(2^{P(n)})$*-approximable for every polynomial* $P(n)$ *even if* $c = 2$*, the reload cost matrix is symmetric and satisfies the triangular inequality.*

Proof: The proofs are quite identical to the proof of Corollary 12. For (*i*), replace the entries of R equal to 1 (*i.e.*, $r_{1,1} = 1$) by 0, and for (*ii*) replace M by $M = O(2^{P(n)})\, n$. □

5.3 Paths and trails with asymmetric reload costs

We now deal with asymmetric reload costs. We mainly prove that the minimum reload s-t trail problem is **NP**-hard in this case.

Theorem 20. *The minimum asymmetric reload s-t trail problem is* **NP***-hard if* $c \geq 3$ *and the maximum degree of* G^c *is equal to* 3*.*

Proof: This proof is similar to the one of Theorem 18, *i.e.* we reduce $(3, B2)$-SAT to the existence of an s-t path with reload cost at most L. In what follows, we use the same notations and only describe how G^c is built upon $\mathcal{I}$. A trail must be a path in the graph of G^c since a vertex's maximum degree is 3. Hence we only deal with paths in this proof.

We have $I_c = \{1, 2, 3\}$ and $L = 15|\mathcal{X}| + 6|\mathcal{C}| + 1$. The matrix R is defined as $r_{1,2} = r_{2,3} = r_{3,1} = M$ where $M > L$. The other entries of R are set to 1. The graph

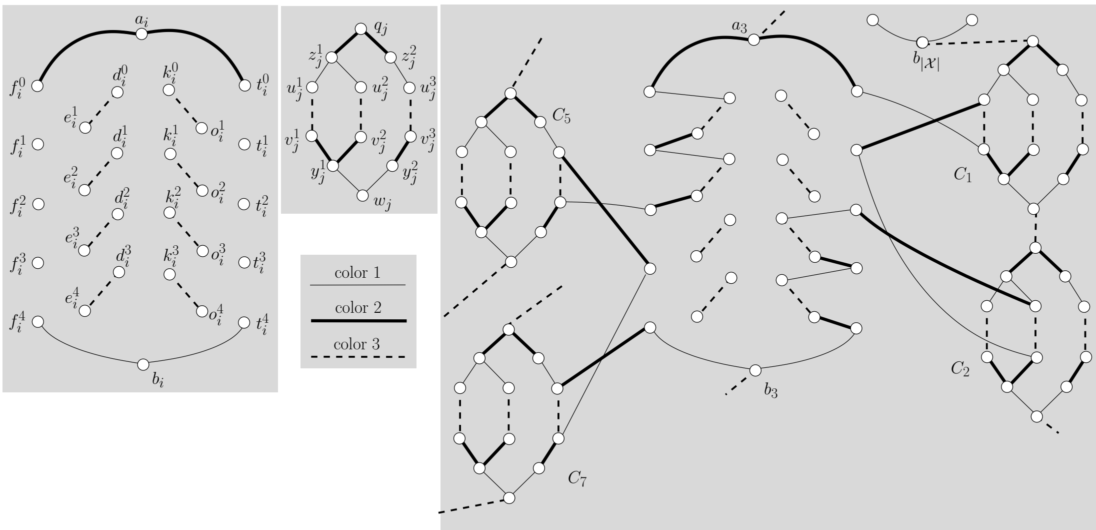


Figure 5.9: Left: Gadgets for a variable x_i. Middle: Gadget of a clause C_j. Right: x_3 appears in the four clauses $C_1 = (\overline{x}_3 \vee x_5 \vee \overline{x}_6)$, $C_2 = (\overline{x}_1 \vee \overline{x}_3 \vee x_4)$, $C_5 = (x_1 \vee x_2 \vee x_3)$ and $C_7 = (\overline{x}_1 \vee x_2 \vee x_3)$.

G^c has a source s and a sink t. In addition, for each $x_i \in \mathcal{X}$ (resp. $\mathcal{C}_j \in \mathcal{C}$) we build a gadget as depicted on the left (resp. middle) of Figure 5.9. The gadget of a variable x_i consists of a left part (vertices f_i, d_i and e_i), a right part (vertices t_i, k_i and o_i), an entrance a_i and an exit b_i. The left (resp. right) part corresponds to the case where x_i is set to FALSE (resp. TRUE). The gadget of a clause C_j consists of a left part (vertices u_j^1 and v_j^1), a middle part (vertices u_j^2 and v_j^2), a right part (vertices u_j^3 and v_j^3), an entrance q_j, an exit w_j and four intermediate vertices z_j^1, z_j^2, y_j^1 and y_j^2. The left, middle and right parts correspond to the first, second and third variable of C_j respectively.

We link the gadgets by adding the following edges with color 3 (dashed): sa_1, b_1a_2, b_2a_3, $\ldots$, $b_{|\mathcal{X}|-1}a_{|\mathcal{X}|}$; $b_{|\mathcal{X}|}q_1$; w_1q_2,w_2q_3, $\ldots$, $w_{|\mathcal{C}|-1}\ q_{|\mathcal{C}|}$, $w_{|\mathcal{C}|}t$ (this construction is similar to the one described in the left part of Figure 5.7 except for the colors of the edges). For each pair x_i, C_j such that x_i is the ℓ-th variable of C_j and C_j is the h-th

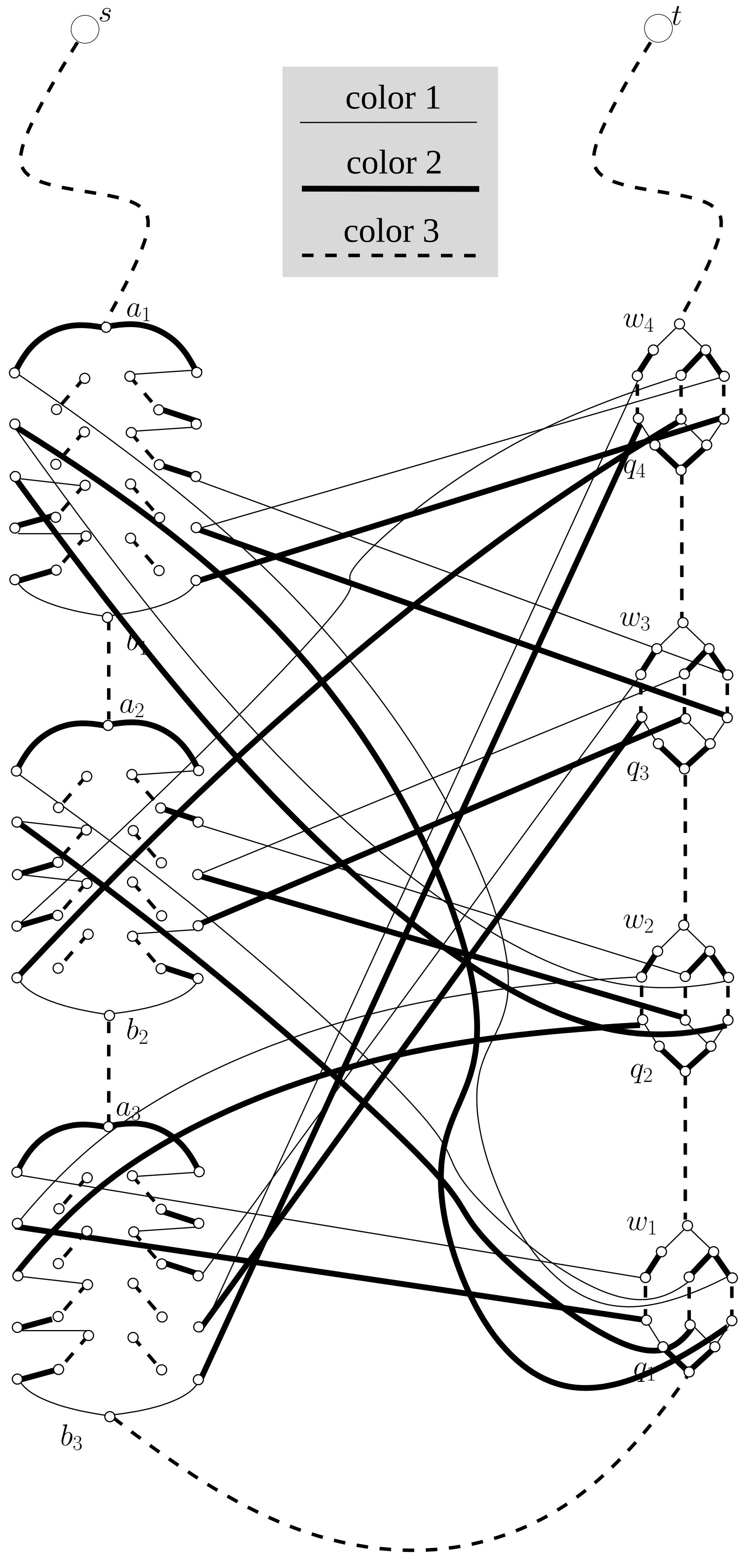


Figure 5.10: An example of the 3-edge-colored graph G^c associated with the instance $\mathcal{I} = \{(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x_2} \vee x_3) \wedge (\bar{x_1} \vee \bar{x_2} \vee \bar{x_3}) \wedge (\bar{x_1} \vee x_2 \vee \bar{x_3})\}$ of the $(3, B_2)$-SAT problem.

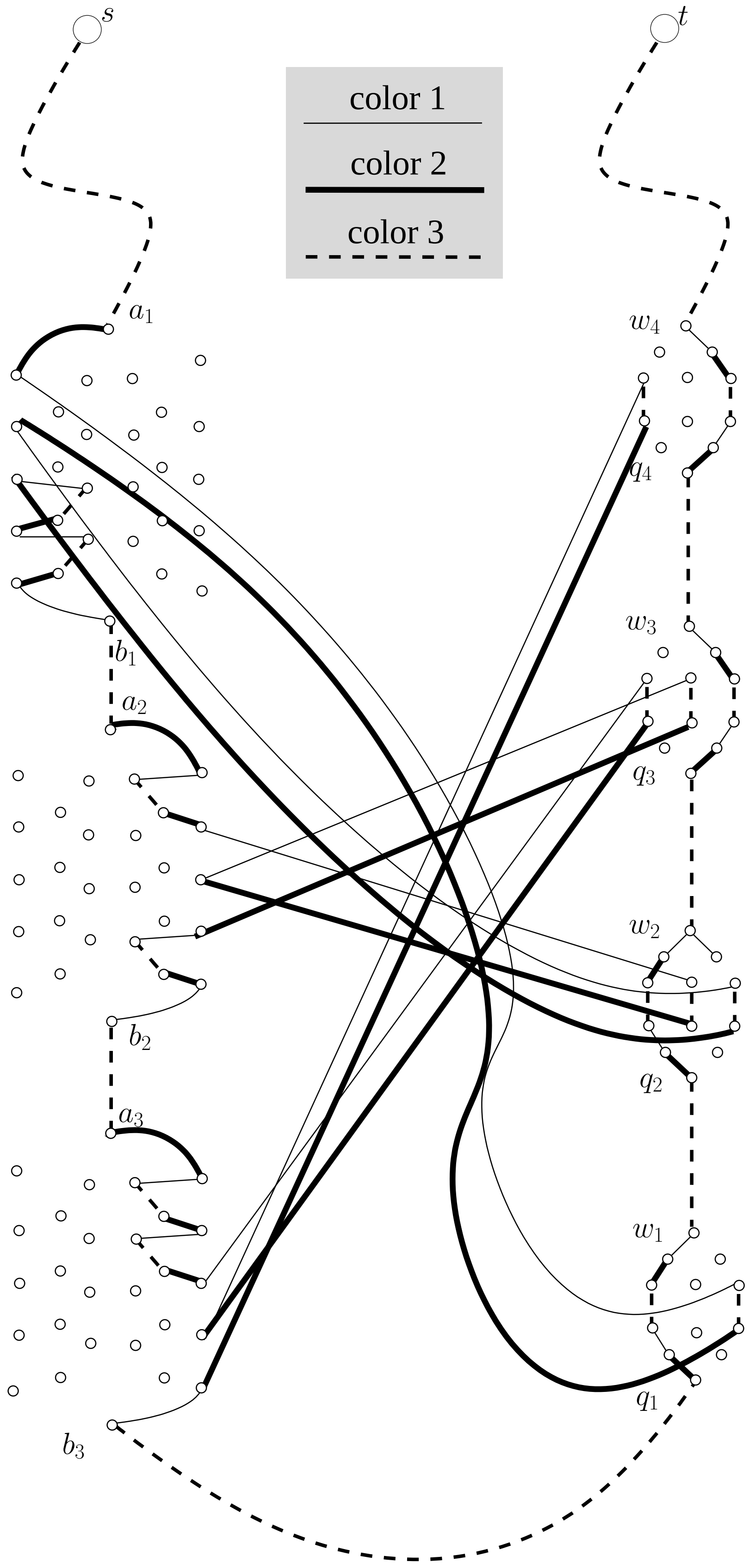


Figure 5.11: An example of a solution of the Figure 5.10, where the variable x_1 is set to false and the variables x_2 and x_3 are set to true. The reload costs are $r_{1,2} = r_{2,3} = r_{3,1} = M > L$, the others entries are set to 1 and $L = 70$.

clause of x_i we proceed as follows. If x_i appears negated in C_j then add $t_i^{h-1}v_j^\ell$ with color 1 (thin), $t_i^h u_j^\ell$ with color 2 (bold), $f_i^{h-1}d_i^{h-1}$ with color 1 and $f_i^h e_i^h$ with color 2. If x_i appears unnegated in C_j then add $f_i^{h-1}v_j^\ell$ with color 1, $f_i^h u_j^\ell$ with color 2, $t_i^{h-1}k_i^{h-1}$ with color 1 and $t_i^h o_i^h$ with color 2. Now G^c is fully described. An example is given on the right of Figure 5.9. It is not difficult to see that each vertex's degree of G^c is at most 3.

As in the proof of Theorem 18 it is not difficult to see that a truth assignment that satisfies $\mathcal{I}$ corresponds to an s-t path with reload cost $15|\mathcal{X}| + 6|\mathcal{C}|$ in G^c and vice-verse. See Figures 5.10 for an example and Figure 5.11 for its solution. $\square$

For graphs of maximum degree 3, trails and paths are identical. Thus, using Theorem 20, we deduce:

Corollary 15. *The minimum asymmetric reload s-t path problem is* **NP***-hard if $c \geq 3$ and the maximum degree of G^c is equal to 3.*

Corollary 16. *The two following statements hold:*

(*i*) *In the general case, the minimum asymmetric reload s-t trail/path problems are not approximable at all if $c \geq 3$ and the maximum degree of G^c is equal to 3.*

(*ii*) *If $r_{i,j} \geq 1$ for every $i, j \in I_c$, the minimum asymmetric reload s-t trail/path problems are not $O(2^{P(n)})$-approximable for every polynomial P if $c \geq 3$ and the maximum degree of G^c is equal to 3.*

Proof: The proofs are quite identical to the proof of Corollary 12. For (*i*) replace the entries of R equal to 1 by 0 and for (*ii*) replace M by $M = O(2^{P(n)})\,L$. $\square$

We know that the minimum symmetric reload s-t trail problem is polynomially solvable (see Theorem 15). We now prove that this result also holds with asymmetric reload costs if the triangle inequality is satisfied.

Theorem 21. *For any simple connected edge-colored graph G^c and any pair s, t of vertices of G^c, the minimum asymmetric reload s-t trail problem can be solved in polynomial time, if the triangle inequality holds.*

Proof: The proof is similar to the one presented in Theorem 16 except that we deal with trails instead of paths. In other words, we can prove that any minimum reload s-t walk ρ_c^* of G^c using a minimal number of edges is indeed an s-t trail of G^c. We recall that ρ_c^* contains the same edge at most twice (see Property 4 of Theorem 14). □

Open Problem 8. *When $c = 2$, it is not known the complexity of the minimum symmetric reload s-t path if the matrix of reload costs does not satisfy the triangle inequality.*

Open Problem 9. *When $c = 2$, it is not known the complexity of the minimum asymmetric reload s-t trail if the matrix of reload costs does not satisfy the triangle inequality.*

These open problems seem important to better understand the complexity of the properly edge-colored s-t trail/path problems when G^c does not have a properly edge-colored s-t trail/path. In this case, one could be interested in seeking an s-t trail/path with a minimum number of vertices for which the adjacent edges have the same color. As a future direction, one could be interested in finding heuristic or exact solutions for the minimum reload s-t path problem. In this case, the polynomial problems regarding s-t trails/walks could be used in the determination of good lower bounds for the value of the minimum reload s-t path problem. Notice that if we study *the min-max reload s-t walk/trail/path problems*, all the results presented here also hold. In this case, we replace the reload cost of a path/trail/walk $\rho = (v_1, e_1, v_2, e_2, \ldots, e_k, v_{k+1})$ between vertices s and t defined as in equation (1.1) by $r(\rho) = \max\{r_{c(e_j),c(e_{j+1})} : j = 1, \ldots, k-1\}$.

Chapter 6

Conclusions and Future Work

We have considered different questions regarding monochromatic and PEC s-t paths and trails on c-edge-colored graphs and digraphs. We also give a rather complete description of the complexity of the minimum reload s-t walk/trail/path problems. Note that, when dealing with reload costs we want to study the complexity of those problems for the smaller number of colors as possible. On the other hand, when studying graphs with no reload costs, finding PEC or monochromatic paths and trails seems easier the greater is the number of colors. In this way, we are interested to find out if the problems remain **NP**-complete when the set of colors is as great as possible.

Finally, in addition to the open problems proposed at the end of each chapter, an interesting question is to study the complexity of PEC s-t paths/trails when restricted to c-edge-colored planar graphs or series-parallel graphs. Next, we enumerate the list of open problems:

Open Problem 1. Consider a non-oriented c-edge-colored graph G^c with no PEC closed trails, an integer k and a sequence $p = (v_1, \ldots, v_k)$ of vertices in $V(G^c)$. Is it possible to find in polynomial time a PEC s-t path/trail visiting all vertices of p in this order?

Open Problem 2. Consider G^c a non-oriented c-edge-colored graph, an integer k and a sequence $C = (c_1, \ldots, c_k)$ of colors. Find a PEC s-t path/trail (if any) only visiting the sequence of C in this order. Is this problem polynomial for graphs with no PEC cycles?

Open Problem 3. Let L be the size of a minimum shortest PEC s-t path. Consider the problem of deciding whether a graph G^c (with no PEC closed trails) has k or more, edge disjoint PEC paths between nodes s and t, each having at most $L+1$ edges. Is this problem **NP**-complete?

Open Problem 4. Given a 2-edge-colored graph G^c with no PEC cycles, two vertices $s, t \in V(G^c)$ and a fixed constant $k \geq 2$. Does G^c contains k PEC vertex/edge disjoint paths between s and t? Is this problem **NP**-complete?

Open Problem 5. Is the problem of finding 2 monochromatic (vertex disjoint) s-t paths with different colors in planar c-edge-colored graphs **NP**-complete?

Open Problem 6. Given a 2-edge-colored tournament T^c. The problem of deciding if T^c contains a directed PEC Hamiltonian path (with no fixed extremities s and t) is **NP**-complete?

Open Problem 7. Given a 2-edge-colored tournament T^c. To check whether T^c contains a PEC circuit is **NP**-complete?

Open Problem 8. When $c = 2$, it is not known the complexity of the minimum symmetric reload s-t path if the matrix of reload costs does not satisfy the triangle inequality.

Open Problem 9. When $c = 2$, it is not known the complexity of the minimum asymmetric reload s-t trail if the matrix of reload costs does not satisfy the triangle inequality.

Tables 6.1, 6.2, 6.3 summarize the main results given in this work.

Table 6.1: Summary of main polynomial results of Chapters 2, 3 and 4.

	Polynomial time problems
c-edge-colored digraph	maximizing the number of PEC s-t trails
	finding a PEC closed trail
c-edge-colored graph with no PEC closed trail G^c	finding a s-t trail visiting all vertices of G^c a predefined number of times
	Hamiltonian path
	Eulerian path

Table 6.2: Summary of main **NP**-complete results of Chapters 2, 3 and 4.

	NP-complete problems
c-edge-colored digraphs D^c	a directed PEC s-t path, even if D^c is a planar 2-edge-colored digraph with no PEC circuits or if D^c is 2-edge-colored tournament or if D^c has $\Omega(\vert V(D^c)\vert)$ colors
	if D^c is a 2-edge-colored tournament, to find a directed PEC s-t Hamiltonian path
	if D^c is a 2-edge-colored tournament, to decide if D^c contains a PEC circuit passing through a given vertex
	2 vertex disjoint monochromatic s-t paths, for paths with different colors
G^c with no PEC closed trail	2 PEC vertex/edge disjoint with length at most $L > 0$
c-edge-colored graphs G^c	2 vertex disjoint monochromatic s-t paths, for paths with different colors

Table 6.3: Summary of main results of Chapter 5.

	Polynomial time problems	**NP-hard problems**
walk	all cases	
trail	(sym. R)	(asym. R) $\wedge$ ($\Delta(G^c) = 3$) $\wedge$ ($c = 3$)
	(asym. R) $\wedge$ (triangle ineq.)	
path	($c = 2$) $\wedge$ (triangle ineq.)	(sym. R) $\wedge$ ($\Delta(G^c) = 4$) $\wedge$ ($c \geq 3$) $\wedge$ (triangle ineq.)
	(sym. R) $\wedge$ ($\Delta(G^c) \leq 3$)	(sym. R) $\wedge$ (G^c is planar) $\wedge$ ($\Delta(G^c) = 4$) $\wedge$ ($c \geq 4$) $\wedge$ (triangle ineq.)

Bibliography

[1] A. Abouelaoualim, K. C. Das, L. Faria, Y. Manoussakis, C. A. Martinhon, and Rachid Saad. Paths and trails in edge-colored graphs. *Theoretical Computer Science*, 409(3):497–510, September 2008.

[2] A. Abouelaoualim, K. C. Das, M. Karpinski, Y. Manoussakis, C. A. Martinhon, Rachid Saad, and W. F. de la Vega. Cycles, paths and trails in edge-colored graphs with given degrees. *Accepted for publication, Journal of Graph Theory*, 2009.

[3] E. Amaldi, G. Galbiati, and F. Maffiol. On minimum reload cost paths, tours and flows. In *Proc. of the Cologne Twente Worshop on Graphs and Combinatorial Optimization*, pages 112–115, Gargnano, Italy, 2008.

[4] D. L. Applegate, R. E. Bixby, V. Chvtal, and W. J. Cook. *The Traveling Salesman Problem: A Computational Study*. Princeton Series in Applied Mathematics. Princeton University Press, 2006.

[5] J. Bang-Jensen and G. Gutin. Alternating cycles and paths in edge-coloured multigraphs: a survey. In *Proceedings of an international symposium on Graphs and combinatorics*, pages 39–60, Amsterdam, The Netherlands, 1997.

[6] J. Bang-Jensen and G. Gutin. Alternating cycles and trails in 2-edge-coloured complete multigraphs. *Discrete Math.*, 188(1-3):61–72, 1998.

[7] J. Bang-Jensen and G. Gutin. *Digraphs: Theory, Algorithms and Applications*. Springer Monographs in Mathematics. Springer-Verlag, 2nd edition edition, 2008.

[8] J. Bang-Jensen, Y. Manoussakis, and C. Thomassen. A polynomial algorithm for hamiltonian-connectedness in semicomplete digraphs. *J. Algorithms*, 13(1):114–127, 1992.

[9] A. Benkouar, Y. Manoussakis, V. Th. Paschos, and R. Saad. On the complexity of some hamiltonian and eulerian problems in edge-colored complete graphs. In *ISA '91: Proceedings of the 2nd International Symposium on Algorithms*, pages 190–198, London, UK, 1991.

[10] P. Berman, M. Karpinski, and A. D. Scott. Approximation hardness of short symmetric instances of max-3sat. *Electronic Colloquium on Computational Complexity*, 49, 2003.

[11] J.A. Bondy and U.S.R. Murty. *Graph Theory With Applications*. Elsevier Science Ltd, 1976.

[12] W.S. Chou, Y. Manoussakis, O. Megalakaki, M. Spyratos, and Z. Tuza. Paths through fixed vertices in edge-colored graphs. *Math. Inf. Sci. Hum.*, 127:49–58, 1994.

[13] P. Das and S. B. Rao. Alternating eulerian trails with prescribed degrees in two edge-colored complete graphs. *Discrete Mathematics*, 43(1):9–20, 1983.

[14] D. Dorninger. On permutations of chromosomes. *Contributions to General Algebra*, 5:95–103, 1987.

[15] D. Dorninger. Hamiltonian circuits determining the order of chromosomes. *Discrete Appl. Math.*, 50(2):159–168, 1994.

[16] J. Feng, H. Giesen, Y. Guo, G. Gutin, T. Jensen, and A. Rafiey. Characterization of edge-colored complete graphs with properly colored hamilton paths. *J. Graph Theory*, 53(4):333–346, 2006.

[17] S. Fortune, J. Hopcroft, and J. Wyllie. The directed subgraph homeomorphism problem. *Theor. Comput. Science*, 10:111–121, 1980.

[18] H. N. Gabow, Sh. N. Maheshwari, and L. J. Osterweil. On two problems in the generation of program test paths. *IEEE transactions on software engineering*, SE-2(3):227–231, 1976.

[19] G. Galbiati. The complexity of a minimum reload cost diameter problem. *Discrete Applied Mathematics*, 156(18):3494–3497, 2008.

[20] H. Galeana-Sánchez, L. R. Pastrana, and H. M. A. Rincn. Kernels in monochromatic path digraphs. *Discuss. Math. Graph Theory*, 25(3):407–417, 2005.

[21] H. Galeana-Sánchez and R. Rojas-Monroy. On monochromatic paths and monochromatic 4-cycles in edge coloured bipartite tournaments. *Discrete Mathematics*, 285(1-3):313–318, 2004.

[22] H. Galeana-Sánchez and R. Rojas-Monroy. Extensions of edge-coloured digraphs.

In *MATH'06: Proceedings of the 10th WSEAS International Conference on Applied Mathematics*, pages 395–400, Stevens Point, Wisconsin, USA, 2006.

[23] I. Gamvros. *Satellite network design, optimization and management*. Phd thesis, University of Maryland, 2006.

[24] I. Gamvros, L. Gouveia, and S. Raghavan. Reload cost trees and network design. In *Proc. International Network Optimization Conference*. INOC 2007, 2007.

[25] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.

[26] A. M. H. Gerards. Matching. *Handbooks in Operations Research and Management Science*, 7(05), 1995.

[27] L. Gourvès, A. Lyra, C. A. Martinhon, and J. Monnot. Complexity of paths, trails and circuits in arc-colored digraphs. Submitted, 2009.

[28] L. Gourvès, A. Lyra, C. A. Martinhon, and J. Monnot. The minimum reload s-t path/trail/walk problems. In *SOFSEM '09: Proceedings of the 35th Conference on Current Trends in Theory and Practice of Computer Science*, volume 5404 of *Lecture Notes in Computer Science*, pages 621–632. Springer-Verlag (Also accepted for publication in the Discrete Applied Mathematics journal), 2009.

[29] L. Gourvès, A. Lyra, C. A. Martinhon, J. Monnot, and F. Protti. On *s-t* paths and trails in edge-colored graphs. In *V Latin-American Algorithms, Graphs and Optimization Symposium (LAGOS 09)*, to appear in Electronic Notes in Discrete Applied Mathematics, Rio Grande do Sul, Brazil, 2009.

[30] G. Gutin, B. Sudakov, and A. Yeo. Note on alternating directed cycles. *Discrete Math.*, 191(1-3):101–107, 1998.

[31] A. Itai, Y. Perl, and Y. Shiloach. The complexity of finding maximum disjoint paths with length constraints. *Networks*, 13(3):277–286, 1982.

[32] Y. Manoussakis. Alternating paths in edge-colored complete graphs. *Discrete Appl. Math.*, 56(2-3):297–309, 1995.

[33] I. Włoch. On kernels by monochromatic paths in the corona of digraphs. *Central European Journal of Mathematics*, 6(4):537–542, 2008.

[34] P. Pevzner. *Computational Molecular Biology: An Algorithmic Approach*. The MIT Press, 2000.

[35] P. A. Pevzner. Dna physical mapping and alternating eulerian cycles in colored graphs. *Algorithmica*, 13(1/2):77–105, 1995.

[36] L. Rédei. Ein kombinatorischer satz. *Acta Litteraria Szeged*, 4:39–43, 1934.

[37] Y. Shiloach and Y. Perl. Finding two disjoint paths between two pairs of vertices in a graph. *J. ACM*, 25(1):1–9, 1978.

[38] S. Szeider. Finding paths in graphs avoiding forbidden transitions. *Discrete Appl. Math.*, 126(2-3):261–273, 2003.

[39] R. E. Tarjan. *Data Structures and Network Algorithms*. Society for Industrial Mathematics, Philadelphia, 1987.

[40] S. Tragoudas and Y. L. Varol. Computing disjoint path with lenght constraints. In *WG '96: Proceedings of the 22nd International Workshop on Graph-Theoretic Concepts in Computer Science*, pages 375–389, London, UK, 1997.

[41] H. Christoph Wirth and J. Steffan. Reload cost problems: minimum diameter spanning tree. *Discrete Appl. Math.*, 113(1):73–85, 2001.

[42] A. Yeo. A note on alternating cycles in edge-coloured graphs. *Journal of Combinatorial Theory*, Series B- 69(2):222–225, 1997.