Postman Problems on Mixed Graphs

by

Francisco Javier Zaragoza Martínez

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2003

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Francisco Javier Zaragoza Martínez

Postman Problems on Mixed Graphs

Francisco Javier Zaragoza Martínez

Abstract

The *mixed postman problem* consists of finding a minimum cost tour of a mixed graph $M = (V, E, A)$ traversing all its edges and arcs at least once. We prove that two well-known linear programming relaxations of this problem are equivalent. The *extra cost* of a mixed postman tour T is the cost of T minus the cost of the edges and arcs of M. We prove that it is $\mathcal{NP}$-hard to approximate the minimum extra cost of a mixed postman tour.

A related problem, known as the *windy postman problem*, consists of finding a minimum cost tour of an undirected graph $G = (V, E)$ traversing all its edges at least once, where the cost of an edge depends on the direction of traversal. We say that G is *windy postman perfect* if a certain *windy postman polyhedron* $\mathcal{O}(G)$ is integral. We prove that series-parallel undirected graphs are windy postman perfect, therefore solving a conjecture of Win.

Given a mixed graph $M = (V, E, A)$ and a subset $R \subseteq E \cup A$, we say that a mixed postman tour of M is *restricted* if it traverses the elements of R exactly once. The *restricted mixed postman problem* consists of finding a minimum cost restricted tour. We prove that this problem is $\mathcal{NP}$-hard even if $R = A$ and we restrict M to be planar, hence solving a conjecture of Veerasamy. We also prove that it is $\mathcal{NP}$-complete to decide whether there exists a restricted tour even if $R = E$ and we restrict M to be planar.

The *edges postman problem* is the special case of the restricted mixed postman problem when $R = A$. We give a new class of valid inequalities for this problem. We introduce a relaxation of this problem, called the *b-join problem*, that can be solved in polynomial time. We give an algorithm which is simultaneously a $\frac{4}{3}$-approximation algorithm for the edges postman problem, and a 2-approximation algorithm for the extra cost of a tour.

The *arcs postman problem* is the special case of the restricted mixed postman problem when $R = E$. We introduce a class of necessary conditions for M to have an arcs postman tour, and we give a polynomial-time algorithm to decide whether one of these conditions holds. We give linear programming formulations of this problem for mixed graphs arising from windy postman perfect graphs, and mixed graphs whose arcs form a forest.

Acknowledgements

C'est triste d'oublier un ami. Tout le monde n'a pas eu un ami.

Le Petit Prince, Antoine de Saint-Exupéry

To my wonderful and beloved wife *Gabrijela*, for all the love she gives me, for all the care she takes of me, for being so patient with me, and for understanding me so well.

To my mother María de la Luz, from whom I learned about life and many other things. To my dear sisters Yesica, Fabiola, and Claudia, whom supported me in all possible ways. To my father Francisco. To my aunt Lucrecia, my uncle Enrique, and my cousins Laura and Alejandro. To Marietta, Jürgen, and Oliver, whom gifted me with a second family.

To Bill Cunningham for his infinite patience, wise advise, and unconditional support. To Joseph Cheriyan, Bertrand Guenin, Dan Brown, and Tom McCormick, for their excellent suggestions to improve this thesis. To Sylvia Boyd, Jim Geelen, Ian Goulden, Penny Haxell, Alfred Menezes, Bruce Richter, and Levent Tunçel, for their continuous encouragement.

To my dearest friends in México, whom I have not forgotten: Ada, Alejandro, Aurora, Carlos, Carlos Enrique, Clara, César, Edgar, Enrique, Erick, Ernesto, Fabio, Feliú, Gabriel, Gabriela, Gonzalo, Icela, Isidoro, Jaime, Jorge, José, José Luis, Juan Antonio, Luis, Luis Carlos, Luis Fernando, Manuel, Martha Irma, Martha Lucía, Moisés, Olienka, Rafael, Raúl, Rodrigo, Rogelio, Rosa, Xicoténcatl, and Yuri. To those that made my time in Waterloo a great experience: Ada, Alex, Antoine, Barry, Berenice, Carlos, Charles, Claudia, Daniel, Donna, Eduardo, Enrique, Fernando, Gabriela, Germán, Gina, Guillermo, John, Katrin, Kerri, Kevin, Leroy, Lisa, Lise, Luis, Luis Fernando, Marcelino, Marina, Miguel, Mike, Neal, Parveen, Reyna, Roberto, Rodolfo, Rodrigo, Rubén, Shashi, Vinko, and YoungBin.

I am in debt with Pastora Revuelta of Universidad de Sevilla, and Ángel Corberán of Universidad de Valencia, for their invitations and hospitality, with Martin Grötschel of Konrad Zuse ZIB for sending me a copy of Zaw Win's doctoral thesis, and with Gabriela Aceves of gasdesign for her freehand rendition of the bridges of Königsberg.

I am grateful for the financial support of the Consejo Nacional de Ciencia y Tecnología (Conacyt 69234), the National Sciences and Engineering Research Council of Canada, and various sources at the University of Waterloo, as well as for the academic leave granted by Universidad Autónoma Metropolitana Unidad Azcapotzalco.

Mojoj najdražoj Gabrijeli

A mi madre, María de la Luz

Contents

List of Figures

Chapter 1

Introduction

El mundo era tan reciente, que muchas cosas carecían de nombre,
y para mencionarlas había que señalarlas con el dedo.

Cien años de soledad, Gabriel García Márquez

In this thesis we study a class of problems collectively known as *postman* problems. As the name indicates, these are the problems faced by a postman who needs to deliver mail to all streets in a city, starting and ending his labour at the city's post office, and minimizing the length of his walk. Some other applications of postman problems include frequent activities such as garbage collection, street sweeping, snow plowing, and bus routing, but also less frequent activities such as museum planning, graph drawing, and circuit embedding.

In graph theoretical terms, a postman problem consists of finding a minimum cost tour of a graph traversing all its arcs (one-way streets) and edges (two-way streets) at least once. The postman problem when all streets are one-way, known as the *directed* postman problem, can be solved in polynomial time by a network flow algorithm. The postman problem when all streets are two-way, known as the *undirected* postman problem, can be solved in polynomial time using Edmonds' matching algorithm, as shown by Edmonds and Johnson. However, Papadimitriou showed that the postman problem becomes $\mathcal{NP}$-hard when both kinds of streets exist. This problem, known as the *mixed* postman problem, and some of its variants, is the central topic of this thesis.

We study some properties of the linear programming relaxations of two well-known integer programming formulations for the mixed postman problem. We prove that these linear programming relaxations are equivalent. In particular, we show that the polyhedron defined by one of them is essentially a projection of the other. We also give new, simpler proofs of the half-integrality of one of these two polyhedra for general graphs, and the integrality of the same polyhedron for graphs with vertices of even degree.

A problem closely related to the mixed postman problem is the *windy* postman problem, where all streets are two-way, but the cost of traversing a street depends on the direction of traversal. An integer programming formulation for the windy postman problem is similar to one for the mixed postman problem, and their linear programming relaxations define polyhedra with similar properties. We say that an undirected graph G is *windy postman perfect* if the windy postman polyhedron is integral. We can see that if G is windy postman perfect then, for every mixed graph M with *underlying* graph G, the mixed postman polyhedron of M is also integral. In his doctoral thesis, Win proved that *even* graphs are windy postman perfect, and conjectured that *series-parallel* graphs are also windy postman perfect. We prove a statement stronger than this conjecture, namely, that a generalization of the windy postman problem has linear programming relaxations whose corresponding polyhedra are always integral if and only if G is series-parallel. We study this problem in terms of *grafts*: We show that another generalization of the windy postman problem has linear programming relaxations whose corresponding polyhedra are always integral only if the graft defined by G and its vertices of odd degree does not contain certain graft minors.

In the remainder of this thesis, we study a variant of the mixed postman problem, called the *restricted* mixed postman problem, in which we require that a tour of a mixed graph traverses exactly once some of its edges and arcs, called *restricted*. These restrictions appear in real-life situations due to bad roads, safety concerns, noise limitation, traffic impediments, etc. We focus on two special cases of the restricted mixed postman problem: when all arcs are restricted, and when all edges are restricted.

The first special case of the restricted mixed postman problem that we study, when all arcs are restricted, is called the *edges* postman problem. It is not difficult to see that we can eliminate the arcs from a reformulation of the edges postman problem, replacing them by *demands* on the vertices. Hence, the edges postman problem can be seen as a version of the flow problem on an undirected graph G. In his doctoral thesis, Veerasamy conjectured that this problem was $\mathcal{NP}$-hard, and gave an approximation algorithm for it. We prove that the edges postman problem is $\mathcal{NP}$-hard, even if we restrict G to be planar. We correct a small flaw in Veerasamy's analysis of his algorithm, and give two other approximation algorithms for the edges postman problem with better approximation guarantees. The best of our algorithms has an approximation guarantee of $\frac{4}{3}$, which is better than the currently best known guarantee for the mixed postman problem, namely $\frac{3}{2}$.

For most practical applications of postman problems, we consider that the cost of traversing each edge or arc for the first time is irrelevant, since we are obliged to traverse them at least once. In fact, a computationally equivalent way of formulating any postman problem would be to find a tour minimizing the *extra* cost due to the additional traversals. However, it is possible that these formulations have different approximability properties. As an example, we mentioned in the last paragraph that the mixed postman problem has an approximation algorithm with constant guarantee for the optimal length of a tour, but we prove that it is impossible to approximate its extra cost within any given factor, unless $\mathcal{P} = \mathcal{NP}$. As a positive result, we prove that our best approximation algorithm for the edges postman problem has also a guarantee of 2 for the extra cost.

Given that the edges postman problem is $\mathcal{NP}$-hard, we cannot expect to obtain a complete linear programming formulation for it. However, we are able to give a linear programming formulation for a relaxation of the edges postman problem that we call the *b-join* problem. This problem is related to some well-known problems in matching theory.

The second special case of the restricted mixed postman problem, when all edges are restricted, is called the *arcs* postman problem. Although it was known that this problem is $\mathcal{NP}$-hard, nothing was known about the complexity of the problem of finding a feasible solution. We prove that finding a feasible solution is $\mathcal{NP}$-complete, even if we restrict the input to be planar. We introduce an infinite class of necessary conditions for feasibility, and we exhibit an infinite family of mixed graphs to show that no finite subset of our necessary conditions is sufficient. We give a polynomial-time algorithm to decide whether one of these conditions holds. Finally, we give linear programming formulations of the arcs postman problem for the class of mixed graphs arising from windy postman perfect graphs, and the class of mixed graphs whose arcs form a forest.

1.1 Outline of the Thesis

This thesis is divided into six chapters. In the remainder of Chapter 1, we define our basic terminology, and introduce some basic results that we will use later. Chapter 2 is a survey on the Eulerian tour problem, and on polynomial-time solvable postman problems. Each of the last four chapters is devoted to one of the hard postman problems mentioned before. In Chapter 3, we study linear programming relaxations of the mixed postman problem, and we introduce its bounded and restricted versions. Chapter 4 deals with the windy postman problem and the integrality properties of the windy postman polyhedron. In Chapter 5, we study the approximability of the edges postman problem, and the solvability of one of its relaxations. In Chapter 6, we study the complexity of the arcs postman problem, we give necessary conditions for feasibility, and we discuss some solvable cases.

Except for very small details, the last three chapters can be read independently of each other, although they all depend on Chapter 3. We recommend that time-conscious readers work quickly through Chapters 1 and 2, paying special attention to Section 1.3 on mixed graphs, and then move on to Chapter 3, before going into any of Chapters 4, 5, or 6.

1.2 Mathematical Preliminaries

We denote by $\mathbb{N}$ the set of *natural* numbers $\{1, 2, \ldots\}$, by $\mathbb{Z}$ the set of *integer* numbers $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$, by $\mathbb{Q}$ the set of *rational* numbers, and by $\mathbb{R}$ the set of *real* numbers. Moreover, we denote by $\mathbb{Z}_+$, $\mathbb{Q}_+$ and $\mathbb{R}_+$ the sets of nonnegative integer, rational, and real numbers, respectively. Given a real number a, we denote by $\lfloor a \rfloor$ the *integer part* of a, that is, the largest integer b such that $a \geq b$, and by $\{a\}$ the *fractional part* of a, that is, $a - \lfloor a \rfloor$.

Given a nonempty set S, an element x of $\mathbb{R}^S$ is called a *real vector* on S. Similarly, an element of $\mathbb{Z}^S$ is called an *integer* vector on S, an element of $\mathbb{Q}^S$ is called a *rational* vector on S, etc. For $s \in S$, we denote $x(s)$ also by x_s, and we call it the s-th *entry* of x. We say that the entries of x are *indexed* by S. For $T \subseteq S$, we denote $\sum_{t \in T} x_t$ by $x(T)$. For $S' \subseteq S$, the *restriction* of x to S' is the vector $x[S'] \in \mathbb{R}^{S'}$ such that $x[S'](s) = x(s)$ for all $s \in S'$. If $S = \{1, \ldots, n\}$, we denote $\mathbb{R}^S$ by $\mathbb{R}^n$, $\mathbb{Q}^S$ by $\mathbb{Q}^n$, $\mathbb{Z}^S$ by $\mathbb{Z}^n$, etc., and we say that $x \in \mathbb{R}^n$ is an n-dimensional real vector. We extend all these notations to *real matrices*.

If $T \subseteq S$, the *characteristic* vector χ^T of T with respect to S is defined by the entries $\chi^T(t) = 1$ if $t \in T$, and $\chi^T(t) = 0$ otherwise. If $T = S$ we write $\mathbf{1}_S$ or $\mathbf{1}$ instead of χ^S, if T consists of only one element t we write $\mathbf{1}_t$ instead of $\chi^{\{t\}}$, and if T is empty we write $\mathbf{0}_S$ or $\mathbf{0}$ instead of $\chi^{\emptyset}$. If $x \in \mathbb{R}^n$, the *support* of x is the vector $y \in \mathbb{R}^n$ such that $y_i = 1$ if $x_i \neq 0$, and $y_i = 0$ otherwise, and it is denoted by $\mathrm{supp}(x)$. The *positive* support of x is the vector $y \in \mathbb{R}^n$ such that $y_i = 1$ if $x_i > 0$, and $y_i = 0$ otherwise, and it is denoted by $\mathrm{supp}_+(x)$. The *characteristic* vector $\chi^{\mathcal{F}}$ of an ordered tuple $\mathcal{F}$ on S has entries $\chi^{\mathcal{F}}(s)$ equal to the number of times that s appears in $\mathcal{F}$, for every $s \in S$.

1.3 Graph Theory

There are no standard conventions for names and notations in this field, and we have decided to follow Bondy and Murty [10] to a certain extent. However, we aim to define as many terms as possible in such a way that they apply to all kinds of graphs.

1.3.1 Undirected, Directed and Mixed Graphs

We define three kinds of *graphs*: undirected, directed, and mixed. An *undirected graph* G is an ordered triple $(V(G), E(G), \psi_G)$ consisting of two disjoint sets $V(G)$ and $E(G)$ of *vertices* and *edges*, respectively, and an *incidence* function $\psi_G : E(G) \to \{\{u, v\} : u, v \in V(G)\}$. A *directed graph* D is an ordered triple $(V(D), A(D), \psi_D)$ consisting of two disjoint sets $V(D)$ and $A(D)$ of *vertices* and *arcs*, respectively, and an *incidence* function $\psi_D : A(D) \to \{(u, v) : u, v \in V(D)\}$. A *mixed graph* M is an ordered quintuple $(V(M), E(M), A(M), \psi_M^E, \psi_M^A)$ consisting of three mutually disjoint sets $V(M)$, $E(M)$, and $A(M)$ of *vertices*, *edges*, and *arcs*, respectively, and two *incidence* functions $\psi_M^E : E(M) \to \{\{u, v\} : u, v \in V(M)\}$ and $\psi_M^A : A(M) \to \{(u, v) : u, v \in V(M)\}$. See Figure 1.1 for some examples. In this thesis, all graphs are finite.

To avoid repetition, we see the undirected graph $G = (V(G), E(G), \psi_G)$ as the mixed graph $G = (V(G), E(G), \emptyset, \psi_G, \emptyset)$, and the directed graph $D = (V(D), A(D), \psi_D)$ as the mixed graph $D = (V(D), \emptyset, A(D), \emptyset, \psi_D)$. When they are clear from the context, we drop the incidence functions and write $G = (V(G), E(G))$, $D = (V(D), A(D))$ and $M = (V(M), E(M), A(M))$, or simply $G = (V, E)$, $D = (V, A)$ and $M = (V, E, A)$, respectively.

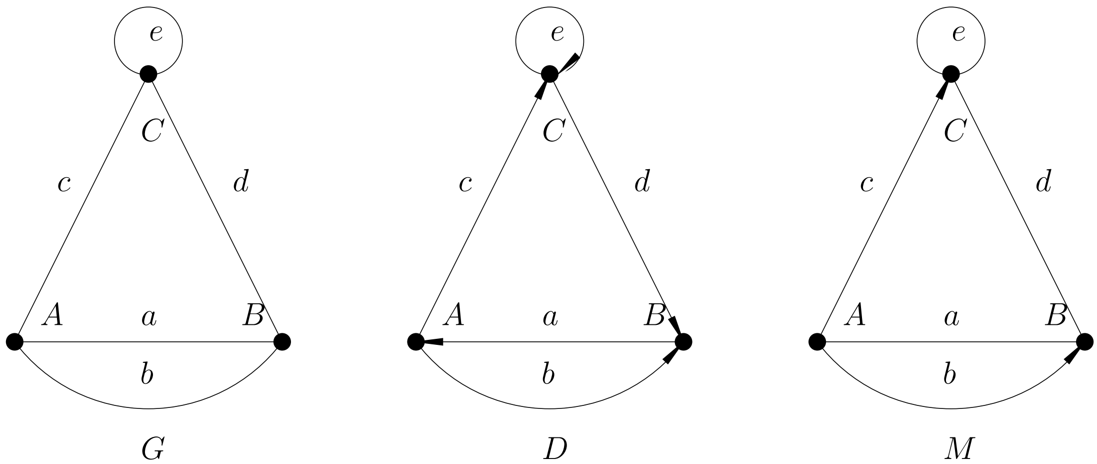

Figure 1.1: An undirected graph G, a directed graph D, and a mixed graph M.

Let $M = (V(M), E(M), A(M), \psi_M^E, \psi_M^A)$ be a mixed graph. Let $u, v \in V(M)$, and let $e, f \in E(M) \cup A(M)$. If $\psi_M^E(e) = \{u, v\}$ or $\psi_M^A(e) = (u, v)$ we say that u and v are the *ends* of e, and that e *joins* u and v. If e is an arc, we also say that e is *oriented* from u to v, that u is the *tail* of e, and that v is the *head* of e. If there exists $e \in E(M) \cup A(M)$ such that e joins u and v, we say that u and v are *adjacent*, that e is *incident* with u and v, and vice versa. If $u = v$ then e is called a *loop*. Two distinct edges e and f are *parallel* if $\psi_M^E(e) = \psi_M^E(f)$. Two distinct arcs e and f are *parallel* if $\psi_M^A(e) = \psi_M^A(f)$. An edge e and an arc f are *parallel* if $\psi_M^E(e) = \{u, v\}$ and $\psi_M^A(f) = (u, v)$ for some $u, v \in V(M)$. The mixed graph M is called *simple* if it has no loops, and no parallel edges or arcs. In this case, we usually denote an edge e with ends u and v as $\{u, v\}$ or uv, and an arc a with tail u and head v as (u, v) or uv. In this thesis, graphs are not necessarily simple, unless we explicitly say so. In Figure 1.1, we draw an edge as a curve joining its ends, and an arc as an arrow from its tail to its head. In G, edges a and b are parallel, while edge e is a loop.

Given a mixed graph M with no loops, and no parallel edges or arcs, its *adjacency matrix* $\mathcal{A}(M)$ is the integer matrix on $V(M) \times V(M)$ whose entries are defined as follows:

$$\mathcal{A}(M)_{u,v} = \begin{cases} 1 & \text{if there is an edge or arc from } u \text{ to } v \\ 0 & \text{otherwise.} \end{cases} \tag{1.1}$$

If we allow parallel edges or arcs, the *incidence matrix* $\mathcal{I}(M)$ of M is the integer matrix on $V(M) \times (E(M) \cup A(M))$ whose entries are defined as follows:

$$\mathcal{I}(M)_{v,e} = \begin{cases} 1 & \text{if } e \in E(M) \text{ and } v \text{ is an end of } e \\ 1 & \text{if } e \in A(M) \text{ and } v \text{ is the head of } e \\ -1 & \text{if } e \in A(M) \text{ and } v \text{ is the tail of } e \\ 0 & \text{otherwise.} \end{cases} \tag{1.2}$$

We say that the directed graph $M^* = (V(M), E(M) \cup A(M), \psi^*_M)$ is an *orientation* of M if, $\psi^*_M(a) = \psi^A_M(a)$ for all $a \in A(M)$, and, for all $e \in E(M)$, $\psi^E_M(e) = \{u, v\}$ implies that $\psi^*_M(e)$ is one of (u, v) or (v, u). Two orientations M^+ and M^- of M are said to be *opposite* if, for all non-loops $e \in E(M)$, $\psi^+_M(e) \neq \psi^-_M(e)$. The *associated* directed graph $\vec{M}$ of M is given by $(V(M), \vec{E}(M) \cup A(M), \vec{\psi}_M)$, where $\vec{E}(M) = \{e^+, e^- : e \in E(M)\}$, for every $e \in E(M)$, if $\psi_M(e) = \{u, v\}$ then $\vec{\psi}_M(e^+) = (u, v)$ and $\vec{\psi}_M(e^-) = (v, u)$, or vice versa, and $\vec{\psi}_M(a) = \psi^A_M(a)$ for all $a \in A(M)$. The *underlying* undirected graph $\bar{M}$ of M is given by $(V(M), E(M) \cup A(M), \bar{\psi}_M)$, where, for every $a \in A(M)$, if $\psi_M(a) = (u, v)$ then $\bar{\psi}_M(a) = \{u, v\}$, and $\bar{\psi}_M(e) = \psi_M(e)$ for every $e \in E(M)$. In Figure 1.1, D is an orientation of both G and M, and G is the underlying undirected graph of both D and M, while in Figure 1.2, D^- is an orientation of G opposite to D, $\vec{G}$ is the associated directed graph of G, and $\vec{M}$ is the associated directed graph of M.

1.3.2 Subgraphs and Isomorphism

Let $M = (V(M), E(M), A(M), \psi^E_M, \psi^A_M)$ and $N = (V(N), E(N), A(N), \psi^E_N, \psi^A_N)$ be two mixed graphs. We say that M is a *subgraph* of N, and we denote this by $M \subseteq N$, if $V(M) \subseteq V(N)$, $E(M) \subseteq E(N)$, $A(M) \subseteq A(N)$, $\psi^E_M(e) = \psi^E_N(e)$ for all $e \in E(M)$, and $\psi^A_M(a) = \psi^A_N(a)$ for all $a \in A(M)$. If $V(M) = V(N)$ we say that M is *spanning*. If $V(M) \cup E(M) \cup A(M) \subset V(N) \cup E(N) \cup A(N)$, we say that M is *proper*. Two special subgraphs of M are the undirected graph $G_M = (V(M), E(M), \psi^E_M)$ and the directed graph $D_M = (V(M), A(M), \psi^A_M)$. For $S \subseteq V(M)$, the subgraph of M *induced* by S is the mixed graph $M[S] = (S, E_S, A_S, \psi^E_S, \psi^A_S)$ such that $E_S = \{e \in E(M) : \psi^E_M(e) \subseteq S\}$, $A_S = \{a \in A(M) : \psi^A_M(a) \in S^2\}$, $\psi^E_S(e) = \psi^E_M(e)$ for all $e \in E_S$, and $\psi^A_S(a) = \psi^A_M(a)$ for all $a \in A_S$. For $F \subseteq E(M)$ and $B \subseteq A(M)$, the subgraph of M *induced* by F and B is

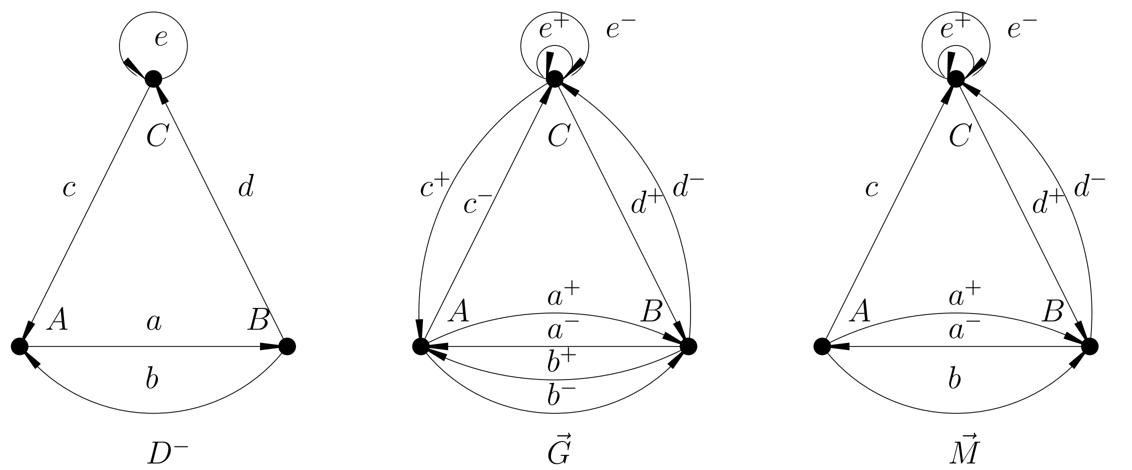

Figure 1.2: Another orientation of G, and the associated directed graphs of G and M.

the mixed graph $M[F, B] = (S, F, B, \psi_M^F, \psi_M^B)$ such that S is the set of ends of all edges in F and arcs in B, $\psi_M^F(e) = \psi_M^E(e)$ for all $e \in F$, and $\psi_M^B(a) = \psi_M^A(a)$ for all $a \in B$. For $v \in V(M)$, $M \setminus v = M[V(M) \setminus v]$ is the subgraph of M obtained by *deleting* v.

We say that M and N are *isomorphic*, and we denote this by $M \cong N$, if there are bijections $\phi_V : V(M) \to V(N)$, $\phi_E : E(M) \to E(N)$, and $\phi_A : A(M) \to A(N)$ such that, for all $u, v \in V(M)$, $e \in E(M)$ and $a \in A(M)$ we have that $\psi_M^E(e) = \{u, v\}$ if and only if $\psi_N^E(\phi_E(e)) = \{\phi_V(u), \phi_V(v)\}$ and $\psi_M^A(a) = (u, v)$ if and only if $\psi_N^A(\phi_A(a)) = (\phi_V(u), \phi_V(v))$. The ordered triple $\phi = (\phi_V, \phi_E, \phi_A)$ is called an *isomorphism* between M and N.

For $e \in E(M)$, the mixed graph obtained by *subdividing* e is $M_e = (V(M) \cup e, (E(M) \setminus e) \cup \{e_1, e_2\}, A(M), \psi_M^e, \psi_M^A))$, where $\{e_1, e_2\}$ is disjoint to $V(M) \cup E(M) \cup A(M)$, $\psi_M^E(e) = \{u, v\}$, $\psi_M^e(f) = \psi_M^E(f)$ for all $f \in E(M) \setminus e$, $\psi_M^e(e_1) = \{u, e\}$, and $\psi_M^e(e_2) = \{e, v\}$. Similarly, for $a \in A(M)$, the mixed graph obtained by *subdividing* a is $M_a = (V(M) \cup a, E(M), (A(M) \setminus a) \cup \{a_1, a_2\}, \psi_M^E, \psi_M^a))$, where $\{a_1, a_2\}$ is disjoint to $V(M) \cup E(M) \cup A(M)$, $\psi_M^A(a) = (u, v)$, $\psi_M^a(b) = \psi_M^A(b)$ for all $b \in A(M) \setminus a$, $\psi_M^a(a_1) = (u, e)$, and $\psi_M^a(a_2) = (e, v)$. If N is isomorphic to a mixed graph obtained from M by a (possibly empty) sequence of subdivisions of edges and arcs, then N is said to be a *subdivision* of M.

A simple undirected graph in which each pair of distinct vertices is joined by an edge is called *complete*. For each $n \in \mathbb{N}$, up to isomorphism, there is only one complete undirected graph on n vertices, denoted by K_n. A directed graph isomorphic to $\vec{K}_n$ is called *complete*. An undirected graph is called *bipartite* if its vertex set can be partitioned into two subsets X and Y, so that each edge has one end in X and one end in Y. Such a partition (X, Y) is called a *bipartition* of the undirected graph, and each of X and Y is called a *side* of the bipartition. A *complete bipartite* undirected graph is a simple undirected graph with a bipartition (X, Y) in which each vertex in X is joined to each vertex of Y. Up to isomorphism, the only complete bipartite undirected graph with one side of size m and the other side of size n is denoted by $K_{m,n}$. A mixed graph M is *bipartite* if $\bar{M}$ is.

1.3.3 Cuts, Paths, Trees and Connectivity

Let $M = (V(M), E(M), A(M), \psi_M^E, \psi_M^A)$ be a mixed graph, and let $S \subseteq V$. We define three kinds of *cuts*. The *undirected cut* $\delta_E(S)$ determined by S is the set of edges with one end in S and the other end in $\bar{S} = V \setminus S$. The *directed cut* $\delta_A(S)$ determined by S is the set of arcs with tails in S and heads in $\bar{S}$. The *total cut* $\delta_M(S)$ determined by S is the set $\delta_E(S) \cup \delta_A(S) \cup \delta_A(\bar{S})$. For single vertices $v \in V(M)$ we write $\delta_E(v)$, $\delta_A(v)$, $\delta_M(v)$ instead of $\delta_E(\{v\})$, $\delta_A(\{v\})$, $\delta_M(\{v\})$, respectively. The cardinalities of cuts are called *degrees*, and we define four kinds of these: The *degree* $d_E(S) = |\delta_E(S)|$, the *outdegree* $d_A(S) = |\delta_A(S)|$, the *indegree* $d_A(\bar{S}) = |\delta_A(\bar{S})|$, and the *total* degree $d_M(S) = |\delta_M(S)|$, respectively. If every vertex of M has even total degree, we say that M is *even*. If all vertices of M have the same total degree r, we say that M is *regular* of total degree r. A vertex of total degree zero is called *isolated*. The *minimum* and *maximum* total degrees of a vertex of M are denoted by δ_M and Δ_M, respectively. A cut determined by a proper subset S is called *proper*, otherwise it is called *trivial*. Trivial cuts are always empty. If M has no empty, proper total cut, we say that M is *connected*; otherwise M is *disconnected*. If N is a maximal connected subgraph of M, we say that N is a *connected component* of M. If a cut consists of exactly one edge or arc e, we say that e is a *cut edge* or a *cut arc*, respectively.

For $S, T \subseteq V(M)$, let $E(S,T) = \{e \in E(M) : \psi_M^E(e) = \{u, v\} \text{ for some } u \in S, v \in T\}$, and let $A(S,T) = \{a \in A(M) : \psi_M^A(a) = (u, v) \text{ for some } u \in S, v \in T\}$. Note that $E(S, \bar{S}) = \delta_E(S)$, $A(S, \bar{S}) = \delta_A(S)$. Let $\gamma_E(S) = E(S,S)$, and let $\gamma_A(S) = A(S,S)$.

A *walk* from v_0 to v_n is an ordered tuple $W = (v_0, e_1, v_1, \ldots, v_{n-1}, e_n, v_n)$ on $V(M) \cup E(M) \cup A(M)$ such that $n \in \mathbb{Z}_+$, $v_i \in V(M)$ for all $0 \le i \le n$, and, for all $1 \le i \le n$, either $e_i \in E(M)$ and $\psi_M^E(e_i) = \{v_{i-1}, v_i\}$, or $e_i \in A(M)$ and $\psi_M^A(e_i) = (v_{i-1}, v_i)$. We say that W *traverses* all of $v_0, v_1, \ldots, v_n$ and $e_1, \ldots, e_n$, and we call n the *length* of W. A walk has the same *parity* as its length. A walk W may also be represented by the ordered tuple $(e_1, \ldots, e_n)$ on $E(M) \cup A(M)$ or, if M has no parallel edges or arcs, by the ordered tuple $(v_0, v_1, \ldots, v_n)$. If $e_1, \ldots, e_n$ are pairwise distinct, W is called a *trail*. If $v_0, v_1, \ldots, v_n$ are pairwise distinct, W is called a *path*. If W_1 and W_2 are two paths from u to v, they are said to be *edge-disjoint* if they do not have edges or arcs in common, and they are said to be *internally disjoint* if, besides u and v, they do not have vertices in common. If $v_0 = v_n$, W is said to be a *closed* walk, or a *circuit*. If W is closed and traverses all vertices of M, we call it a *tour*. If W is a closed trail, and $v_1, \ldots, v_n$ are pairwise distinct, we call it a *cycle*. An undirected graph is bipartite if and only if it contains no odd cycle.

A mixed graph is *acyclic* if it contains no cycles. A mixed graph is a *forest* if its underlying undirected graph is acyclic. Note that there are *directed acyclic graphs* that are not forests. A connected forest is a *tree*. Let $G = (V, E)$ be an undirected graph. Then, G is a tree if and only if G is connected and every edge of G is a cut edge. Alternatively, G is a tree if and only if, for all $u, v \in V$, there is a unique path from u to v.

If $u, v \in V(M)$ and there is a walk from u to v, we say that u is *connected* to v, and we denote this by $u \to v$. If for every two vertices u and v of M, $u \to v$ *and* $v \to u$, we say that M is *strongly connected*. Note that if M is strongly connected then it is connected. In fact, if M is an undirected graph, these two concepts are equivalent. If N is a maximal strongly connected subgraph of M, we say that N is a *strongly connected component* of M.

A vertex v of M is a *cut vertex* if $E(M)$ can be partitioned into E_1 and E_2, and $A(M)$ can be partitioned into A_1 and A_2, in such a way that both $E_1 \cup A_1$ and $E_2 \cup A_2$ are nonempty, and the mixed graphs $M[E_1, A_1]$ and $M[E_2, A_2]$ have only v in common. If M is connected and has no cut vertices it is called a *block*. We say that N is a *block* of M if N is a maximal subgraph of M that is a block. A *vertex cut* of M is a subset $S \subseteq V(M)$ such that $M[V(M) \setminus S]$ is disconnected. If S is a vertex cut of M and it has cardinality k, we call S a *k-vertex cut*. If M has at least two nonadjacent vertices, the *vertex connectivity* $\kappa(M)$ of M is the minimum k for which M has a k-vertex cut; otherwise $\kappa(M) = |V(M)| - 1$. We say that M is *k-vertex-connected* if $\kappa(M) \geq k$. Blocks with at least 3 vertices are 2-vertex-connected. If $S \subset V(M)$ and $|\delta_M(S)| = k$, we say that $\delta_M(S)$ is a *k-edge cut*. If M

has at least two vertices, the *edge connectivity* $\kappa'(M)$ of M is the minimum k for which M has a k-edge cut, otherwise $\kappa'(M) = 0$. We say that M is *k-edge-connected* if $\kappa'(M) \geq k$. Note that $\kappa(M) \leq \kappa'(M) \leq \delta_M$ for all mixed graphs M. Menger proved several results relating the connectivity of an undirected graph with its disjoint paths [66]. Among them:

Theorem 1.1 (Menger) *Let G be an undirected graph with at least $k+1$ vertices. Then G is k-vertex-connected if and only if any two vertices of G are joined by at least k internally disjoint paths.*

1.3.4 Graph Minors

Let $G = (V(G), E(G), \psi_G)$ be an undirected graph, and let $e \in E(G)$ with ends $x, y \in V(G)$. The *deletion* of e is the graph $G \setminus e = (V(G), E(G) \setminus e, \psi_G^{\setminus})$ where $\psi_G^{\setminus}(f) = \psi_G(f)$ for all $f \in E(G) \setminus e$. If e is a loop, the *contraction* of e is the graph $G \,/\, e = G \setminus e$. If e is not a loop, the *contraction* of e is the graph $G \,/\, e = ((V(G) \cup e) \setminus \{u, v\}, E(G) \setminus e, \psi_G^{/})$ where $\psi_G^{/}(f) = \{\theta(u), \theta(v)\}$ for all $f \in E(G) \setminus e$ with ends u and v, and $\theta(w) = w$ if $w \notin \{x, y\}$, otherwise $\theta(w) = e$. Informally, $G \setminus e$ is obtained by deleting e from G, whereas $G \,/\, e$ is obtained by deleting e from G and identifying its ends. Note that edge deletion and edge contraction commute, that is, if e and f are two distinct edges of G then $(G \setminus e) \,/\, f \cong (G \,/\, f) \setminus e$, so we will simply write $G \setminus e \,/\, f$ or $G \,/\, f \setminus e$.

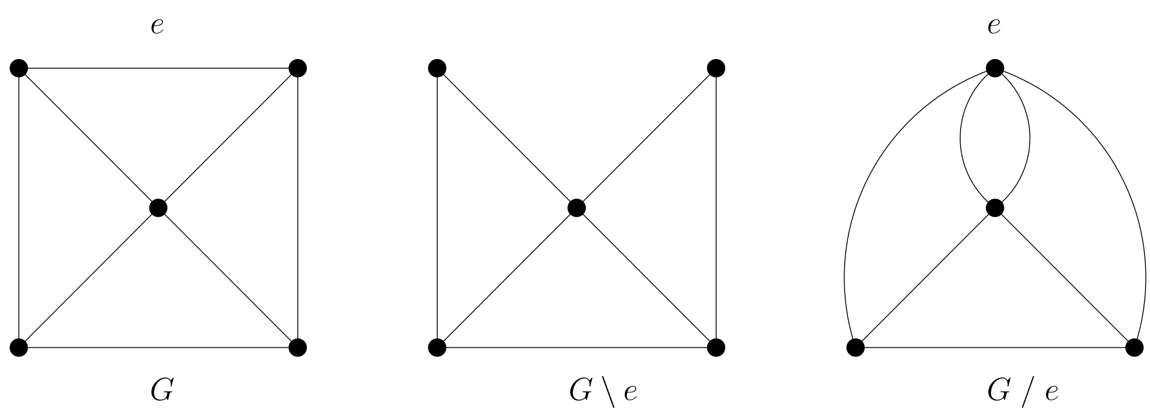

Figure 1.3: An undirected graph G, and the deletion and contraction of e.

Let H be an undirected graph. If H is isomorphic to an undirected graph obtained from G after a sequence of edge deletions, edge contractions, and isolated vertex deletions, we say that H is a *minor* of G, and we denote this by $H \preceq G$. Furthermore, if the sequence is not empty, we say that H is a *proper* minor of G, and we denote this by $H \prec G$. A *class* of undirected graphs $\mathcal{G}$ is a possibly infinite set of undirected graphs such that, for all distinct $G, H \in \mathcal{G}$, G and H are not isomorphic. We say that the class $\mathcal{G}$ is *closed under taking minors* if, for all $G \in \mathcal{G}$, and for all $H \preceq G$, there exists $H' \in \mathcal{G}$ such that $H \cong H'$. For example, the class of undirected forests is closed under taking minors, but neither the class of complete undirected graphs, nor the class of complete bipartite undirected graphs, are closed under taking minors. Given a class $\mathcal{H}$ of undirected graphs, we can define another class of undirected graphs $\mathrm{ex}(\mathcal{H})$ by *excluding* minors in $\mathcal{H}$, that is, we define $\mathrm{ex}(\mathcal{H})$ to be the largest class of undirected graphs such that none of its elements has a minor in $\mathcal{H}$. For example, the class of undirected forests is obtained by excluding loops. Note that $\mathrm{ex}(\mathcal{H})$ is closed under taking minors. One of the deepest results in graph theory, due to Robertson and Seymour in their series of papers *Graph Minors*, is that every class of undirected graphs closed under taking minors can be obtained by excluding finitely many minors.

1.3.5 Planarity and Duality

We usually represent undirected graphs by *drawings* on the real plane $\mathbb{R}^2$. Informally, we identify the vertices of an undirected graph with distinct points, and each edge with a curve joining its ends. It is always possible to draw an undirected graph on the plane in such a way that the only vertices met by any given edge are its ends, although edges may intersect each other. An undirected graph is said to be *planar* if it can be drawn on the plane so that its edges do not intersect. Such a drawing is called a *planar embedding*. Note that the property of being planar is closed under taking minors or subdivisions. It is easy to see that K_5 and $K_{3,3}$ are not planar. Kuratowski proved that an undirected graph is planar if and only if it does not contain a subdivision of K_5 or $K_{3,3}$ [64], and Wagner proved that $\mathrm{ex}(K_5, K_{3,3})$ is precisely the class of planar graphs [86].

A planar embedding of G partitions $\mathbb{R}^2 \setminus G$ into a number of *connected* regions called the *faces* of G. We denote by $F(G)$ the set of faces of G, and we call the pair $(G, F(G))$ a *plane* graph. Each plane graph has exactly one unbounded face, called the *exterior* face.

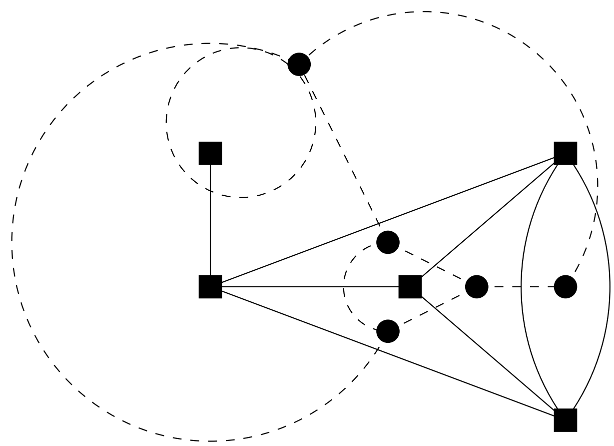

Figure 1.4: A plane graph and its dual.

A face f is said to be *incident* with the vertices and edges in its *boundary*. There are two faces incident with each edge e of G, except if e is a cut edge, when just one face is incident with e. The *degree* of a face f is the number of edges incident to it, with cut edges counted twice. Given a plane graph $(G, F(G))$ with $G = (V(G), E(G), \psi_G)$, its *dual* graph is the plane graph $(G^*, V(G))$, where $G^* = (F(G), E(G), \psi_G^*)$ and, for every $e \in E(G)$, $\psi_G^*(e)$ is the set of faces of G incident with e. In Figure 1.4 we show a plane graph (with square vertices and continuous edges) and its dual (with round vertices and dashed edges). Note that G^* is also a planar graph. In fact, one can find a planar embedding of G^* where each vertex of G^* is a point in the corresponding face of G. Also note that, if G is connected, $(G^*)^* \cong G$, and that a loop of G becomes a cut edge of G^* and vice versa. We should also note that isomorphic plane graphs may have nonisomorphic duals. We say that a mixed graph is *planar* if its underlying undirected graph is planar. Planar mixed graphs also have duals, but they are constructed in a different way, which we introduce later.

A subset of the class of planar graphs that is of considerable theoretical interest is the class $\mathcal{S}$ of *series-parallel* graphs. We define the class $\mathcal{S}$ recursively as follows:

1. All undirected forests are in $\mathcal{S}$.

2. If $G \in \mathcal{S}$, the graphs obtained by adding a loop to G, by subdividing an edge of G, or by adding an edge parallel to an edge of G are in $\mathcal{S}$.

Observe that every connected series-parallel graph with at least three vertices contains two edges in series or two edges in parallel. Also note that $\mathcal{S}$ is closed under taking minors. Furthermore, Duffin proved that $\mathcal{S} = \mathrm{ex}(K_4)$ [23].

1.4 Computational Complexity Theory

Earlier in this chapter we have made reference to *easy* and *hard* problems. Our purpose is to informally describe here what we mean by those two terms, and many others. We recommend the books by Hopcroft and Ullman [55] and Papadimitriou [71].

1.4.1 Problems, Algorithms and Running Time

A *decision problem* P is a pair $(Y(P), N(P))$ of disjoint subsets $Y(P)$ and $N(P)$ of $\mathbb{Z}_+$. An *instance* i of P is an element of $I(P) = Y(P) \cup N(P)$, and it is called a *yes* instance if $i \in Y(P)$, and a *no* instance if $i \in N(P)$. More often than not, we define a decision problem P by describing its set of instances and one of its set of yes instances or its set of no instances. For example, the decision problem PRIMES has set of instances $\mathbb{N}$ and set of yes instances $\{p \in \mathbb{N} : p \text{ is prime}\}$. Equivalently, we say that the decision problem P is the question: "Given an i in $I(P)$, is i in $Y(P)$?" Hence, we can reformulate the problem PRIMES as the question: "Given a natural number i, is i prime?"

Very often we are concerned with decision questions regarding objects that are not necessarily nonnegative integers, for example, rationals, sets, matrices, graphs, tuples, etc. In order to deal with these questions, we introduce the notion of a *good encoding*. In general, a good encoding is an injective function from the set of objects of interest to the set of natural numbers. For example, if the object of interest is a set of cardinality at most n, we can easily *encode* it as a nonnegative integer less than 2^n using binary representations. Another property of a good encoding is that we do not use numbers that are excessively large. For example, note that a set of cardinality n has 2^n subsets, and hence any encoding of sets of cardinality at most n will use a number at least as large as $2^n - 1$. Good encodings also exist for rationals, matrices, graphs, tuples, and many other objects. It is generally accepted that the incidence and adjacency matrices of a mixed graph are good encodings for it. Given a class of objects I with a good encoding, and a subset $Y \subseteq I$, we also say that the pair $(Y, I \setminus Y)$ is a decision problem, as well as the question: "Given an i in I, is i in Y?" The *size* $\ell(i)$ of a nonnegative number i is the length of its binary representation, that is, $\ell(i) = \lceil \log_2(i+1) \rceil$. If g is a good encoding for I, then the *size* $\ell(i)$ of object i is $\ell(g(i))$.

Informally, an *algorithm* $\mathcal{A}$ is a list of well-defined and *elementary* instructions that takes some *input* from the set I and eventually produces some *output* on the set O. The input and the output should be nonnegative integers, but any objects with a good encoding will do. If we *run* $\mathcal{A}$ with input $i \in I$ and it produces output $o \in O$, we denote this by $\mathcal{A}(i) = o$. The *running time* of $\mathcal{A}$ is a function $f_{\mathcal{A}} : \mathbb{N} \to \mathbb{N}$, where $f_{\mathcal{A}}(i)$ is the maximum number of elementary instructions needed to produce an output with an input of size at most i. These functions are usually difficult to compute, so we resort to the $O(\cdot)$ notation. If $f, g : \mathbb{N} \to \mathbb{N}$, we say that f is *order* of g, and denote this by $f = O(g)$, if there exists a positive real constant C, and an integer constant N such that $f(n) \leq C \cdot g(n)$ for all integers $n \geq N$. If $f_{\mathcal{A}} = O(g)$ we say that the running time of $\mathcal{A}$ is $O(g)$. If g is a linear, polynomial, or exponential function, we say that $\mathcal{A}$ is a *linear-*, *polynomial-*, or *exponential-time algorithm*, respectively. Edmonds recognized the importance of polynomial-time algorithms, which he called *good* algorithms, as a measure of efficiency [27].

Given a decision problem P, we say that $\mathcal{A}$ is a *decision algorithm* for P if the input for $\mathcal{A}$ is taken from the set of instances of P, the output of $\mathcal{A}$ is either 1 or 0 (or any other two distinct alternatives), and for all instances i of P, $\mathcal{A}(i) = 1$ if i is a yes instance, and $\mathcal{A}(i) = 0$ if i is a no instance. Very often we are confronted with problems were the objective is to count the number of objects satisfying the given property (a *counting* problem), or to construct an object satisfying the given property (a *construction* problem). Algorithms that find the correct count in the former case, or that construct a correct object in the latter case, are called *counting* and *construction algorithms*, respectively. Furthermore, if the given property is of the form "maximize" or "minimize" a certain function, we say that our problem is a *maximization* or *minimization problem*, respectively, and algorithms that find these *maxima* or *minima* are called *maximization* or *minimization algorithms*, respectively. A maximization or minimization problem is also called an *optimization problem*, and an algorithm that finds an *optimum* solution is called an *optimization algorithm*.

1.4.2 The Classes $\mathcal{P}$ and $\mathcal{NP}$

We say that a problem P is *solvable in polynomial time* if there exists an algorithm $\mathcal{A}$ for P whose running time is polynomial. We denote by $\mathcal{P}$ the set of all *decision* problems solvable in polynomial time. We define the set $\mathcal{NP}$ of decision problems *solvable in nondeterministic polynomial time* as follows: a decision problem P is in $\mathcal{NP}$ if and only if there exists a decision problem $P' \in \mathcal{P}$ and a polynomial function p such that, for every instance $i \in I(P)$, we have that $i \in Y(P)$ if and only if there exists a *certificate* j with $\ell(j) \leq p(\ell(i))$ such that $(i, j) \in Y(P')$. Naturally, $\mathcal{P} \subseteq \mathcal{NP}$. One of the most important questions in computer science is whether $\mathcal{P} = \mathcal{NP}$ or not, and most people believe that the answer is *no*.

We define co-$\mathcal{NP}$ as the set of all decision problems $P = (Y(P), N(P))$ such that their *complements* $P^c = (N(P), Y(P))$ are in $\mathcal{NP}$. Note that $\mathcal{P} \subseteq \mathcal{NP} \cap$ co-$\mathcal{NP}$; moreover, since a problem $P \in \mathcal{NP} \cap$ co-$\mathcal{NP}$ has both *yes* certificates and *no* certificates, we say that it has a *good characterization*. There are very few decision problems known to have good characterizations but not known to be in $\mathcal{P}$. In fact, it usually happens that one proves that a problem has a good characterization and then that it is in $\mathcal{P}$.

Given two decision problems P and Q, we say that P is *reducible* to Q if there exists a polynomial-time algorithm $\mathcal{A}$ with input in $I(P)$ and output in $I(Q)$ such that $i \in Y(P)$ if and only if $\mathcal{A}(i) \in Y(Q)$. Note that if P is reducible to Q and $Q \in \mathcal{P}$ then $P \in \mathcal{P}$ and, similarly, if P is reducible to Q and $Q \in \mathcal{NP}$ then $P \in \mathcal{NP}$. We extend in a natural way the notion of *reducibility* to pairs of problems that are not necessarily of decision type. A problem P is said to be $\mathcal{NP}$*-hard* if each problem in $\mathcal{NP}$ is reducible to P. If in addition $P \in \mathcal{NP}$ then P is said to be $\mathcal{NP}$*-complete*. Many important problems are known to be $\mathcal{NP}$-complete or $\mathcal{NP}$-hard. We refer the reader to the book by Garey and Johnson [42].

Since optimization problems are not decision problems, they cannot be in $\mathcal{P}$ nor $\mathcal{NP}$. However, they can be *transformed* into decision problems. For example, the minimization problem P "minimize $f(x)$ over $x \in X$" (where f is a rational function and X is a set of objects derived from the input to P) can be transformed into the decision problem Q "given an $r \in \mathbb{Q}$, is there an $x \in X$ such that $f(x) \leq r$?" We can do a similar transformation for maximization problems. In both cases, we call Q the *decision version* of P.

1.4.3 Satisfiability and Variants

The main tool to prove that a given problem P is $\mathcal{NP}$-complete is to prove that a known $\mathcal{NP}$-complete problem is reducible to P. In 1971, Cook exhibited the first $\mathcal{NP}$-complete problem, namely SATISFIABILITY, which we describe next [15]. A *variable* x has two literals, a *positive literal* x and a *negative literal* $\neg x$. We can *assign* to x either value *true* or *false*. If x is true, then $\neg x$ is false, and vice versa. A *clause* C is a set of literals, and it is said to be *satisfiable* if we can assign values to the variables in such a way that at least one literal in C is true. A set $\mathcal{C}$ of clauses is *satisfiable* if there exists an assignment of the variables that satisfies all its clauses.

Problem: SATISFIABILITY.
Input: A set $\mathcal{X}$ of variables and a set $\mathcal{C}$ of clauses.
Output: Is $\mathcal{C}$ satisfiable?

For fixed $k \in \mathbb{N}$, the special case of SATISFIABILITY where all clauses contain *exactly* k distinct literals is called k-SATISFIABILITY. Both 1-SATISFIABILITY and 2-SATISFIABILITY can be solved in polynomial time [19].

Theorem 1.2 (Cook) 3-SATISFIABILITY *is* $\mathcal{NP}$*-complete.*

Let $\mathcal{X}$ be a set of variables, and let $\mathcal{C}$ be a set of clauses. Consider the bipartite undirected graph G with vertex set $\mathcal{X} \cup \mathcal{C}$ and edges joining $x \in \mathcal{X}$ and $C \in \mathcal{C}$ if and only if the clause C contains at least one of x or $\neg x$. The special case of SATISFIABILITY where G is restricted to be planar is called PLANAR SATISFIABILITY. In Figure 1.5 we show an instance of this problem with set of variables $\{x_1, x_2, x_3, x_4\}$ and clauses $C_1 = \{x_1, x_2, \neg x_4\}$, $C_2 = \{x_2, \neg x_3, x_4\}$, $C_3 = \{\neg x_1, x_2, x_3\}$, and $C_4 = \{\neg x_1, x_3, \neg x_4\}$. The vertices corresponding to variables are represented with circles, and those corresponding to clauses are represented with squares. We can use the following result of Lichtenstein to prove that certain graph problems remain $\mathcal{NP}$-complete even when we restrict the input to be a planar graph [65]:

Theorem 1.3 (Lichtenstein) PLANAR 3-SATISFIABILITY *is* $\mathcal{NP}$*-complete.*

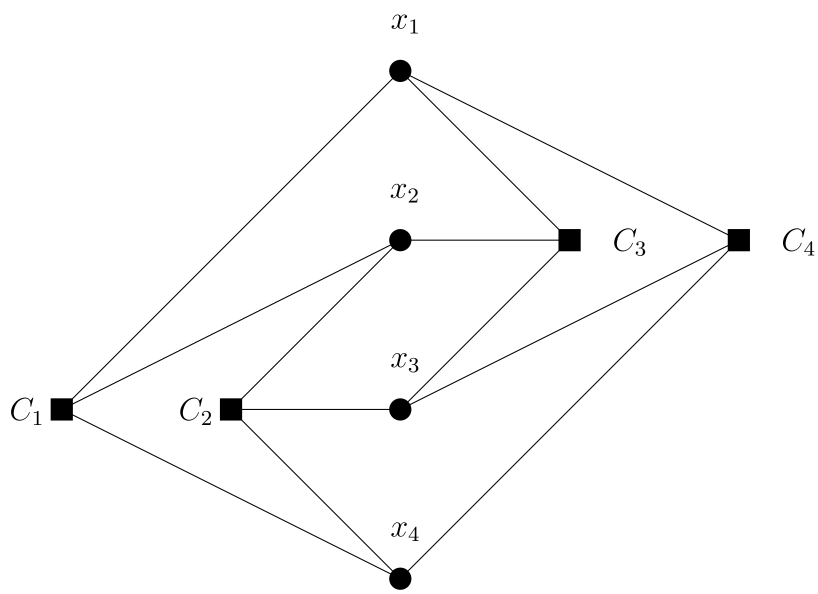

Figure 1.5: A planar instance of SATISFIABILITY.

We can also modify when a clause is said to be satisfiable by giving a different *truth table*. Schaefer obtained a complete characterization of those tables with three literals that give $\mathcal{NP}$-complete problems [79]. In 1-IN-3 SATISFIABILITY, a clause is true if and only if exactly one of its literals is true. In NOT ALL EQUAL SATISFIABILITY, a clause is true if and only if not all its literals have the same value.

Theorem 1.4 (Schaefer) 1-IN-3 SATISFIABILITY *is $\mathcal{NP}$-complete.*

Theorem 1.5 (Schaefer) NOT ALL EQUAL SATISFIABILITY *is $\mathcal{NP}$-complete.*

Similarly as before, we can consider the planar versions of 1-IN-3 SATISFIABILITY and NOT ALL EQUAL SATISFIABILITY, called PLANAR 1-IN-3 SATISFIABILITY and PLANAR NOT ALL EQUAL SATISFIABILITY, respectively. The first of these two problems was shown to be $\mathcal{NP}$-complete by Dyer and Frieze [24], while the second was recently shown to be solvable in polynomial time by Kratochvíl and Tuza [63].

Theorem 1.6 (Dyer and Frieze) PLANAR 1-IN-3 SATISFIABILITY *is $\mathcal{NP}$-complete.*

1.5 Combinatorial Optimization

Combinatorial optimization may be thought of as a branch of mathematics whose main goal is to find *good* algorithms for decision, construction, and optimization problems whose objects of interest are taken from a *discrete* set. When this is not possible, for example, when the problem of interest is $\mathcal{NP}$-hard, combinatorial optimization is also concerned with methods to find *optimal* solutions (even if we cannot guarantee that they run in polynomial time), and with polynomial-time algorithms to find guaranteed *near optimal* solutions. We refer the reader to the book by Cook, Cunningham, Pulleyblank, and Schrijver [16].

1.5.1 Polytopes and Polyhedra

Starting with the work of Edmonds [26], one of the main driving forces for combinatorial optimization has been the study of polyhedra and polytopes. Let $x_1, \ldots, x_m \in \mathbb{R}^n$, and let $\alpha_1, \ldots, \alpha_m \in \mathbb{R}$. We say that $y = \alpha_1 x_1 + \cdots + \alpha_m x_m$ is a *linear combination* of $x_1, \ldots, x_m$. We say that $x_1, \ldots, x_m$ are *linearly independent* if $y = \mathbf{0}_n$ implies that $\alpha_1 = \cdots = \alpha_m = 0$. If $\alpha_1, \ldots, \alpha_m \geq 0$ we say that y is a *nonnegative combination* of $x_1, \ldots, x_m$, and if in addition $\alpha_1 + \cdots + \alpha_m = 1$ we say that y is a *convex combination* of $x_1, \ldots, x_m$. For any set $S \subseteq \mathbb{R}^n$, the *convex hull* of S, denoted by $\mathrm{convhull}(S)$, is the set of all convex combinations of elements of S. A simple, but important result is that for any finite $S \subseteq \mathbb{R}^n$, and any $c \in \mathbb{R}^n$, we have that $\min\{c^\top x : x \in S\} = \min\{c^\top x : x \in \mathrm{convhull}(S)\}$.

Given a matrix $A \in \mathbb{R}^{n \times m}$ and a vector $b \in \mathbb{R}^n$, the *polyhedron* determined by A and b is the set $P(A, b) = \{x \in \mathbb{R}^m : Ax \leq b\}$. Given a vector $c \in \mathbb{R}^m$ and a real $d \in \mathbb{R}$, the *inequality* $c^\top x \leq d$ is *valid* for a polyhedron P if it holds for all $x \in P$. If there exists $l, u \in \mathbb{R}$ such that, for all $1 \leq i \leq m$, the inequalities $x_i \leq u$ and $x_i \geq l$ are valid for P, we say that P is *bounded* and we call it a *polytope*. If $c \neq \mathbf{0}_m$, the polyhedron $H(c, d) = \{x \in \mathbb{R}^m : c^\top x = d\}$ is called the *hyperplane* determined by c and d. We say that $H(c, d)$ is a *supporting* hyperplane of P if $c^\top x \leq d$ is valid for P and $P \cap H(c, d) \neq \emptyset$. The intersection of a polyhedron P and one of its supporting hyperplanes, as well as P and $\emptyset$, are called *faces*. A vector $x \in P$ is called an *extreme point* of P if $\{x\}$ is a face of P. Equivalently, x is an extreme point of P if and only if x is not a convex combination of vectors in $P \setminus \{x\}$. We say that P is *pointed* if it has at least one extreme point. If P is pointed then every minimal nonempty face of P is an extreme point. Nonempty polytopes are pointed. A polytope is equal to the convex hull of its extreme points. Furthermore, a set is a polytope if and only if it is equal to the convex hull of a finite set.

If $A \in \mathbb{Q}^{n\times m}$ and $b \in \mathbb{Q}^n$ then $P(A,b)$ is said to be a *rational* polyhedron. A rational polyhedron is said to be *integral* if each of its nonempty faces contains an integral vector. A pointed rational polyhedron is integral if and only if all its extreme points are integral.

Let P be a polyhedron on the variables $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^p$. The *projection of P onto $y = \mathbf{0}$* is the set $\mathrm{proj}_x(P) = \{x \in \mathbb{R}^n : (x,y) \in P \text{ for some } y \in \mathbb{R}^p\}$. The projection of a polyhedron is a polyhedron. A description of $\mathrm{proj}_x(P)$ is given by the following theorem.

Theorem 1.7 *Let $P = \{(x,y) \in \mathbb{R}^n \times \mathbb{R}^p : Ax + By \le c\}$, then*

$$proj_x(P) = \{x \in \mathbb{R}^n : z^\top(c - Ax) \ge 0 \text{ for all } z \in \mathbb{R}^m_+ \text{ such that } z^\top B = \mathbf{0}\}. \tag{1.3}$$

1.5.2 Linear Programming, Optimization and Separation

Given a rational polyhedron $P \subseteq \mathbb{R}^m$, and a rational vector $c \in \mathbb{Q}^m$, we are often interested in the problem of finding a vector $x^* \in P$ such that it minimizes the *linear function* $c^\top x$ over P, that is $c^\top x^* \le c^\top x$ for all $x \in P$. More precisely, we consider:

Problem: LINEAR PROGRAMMING MINIMIZATION.
Input: Numbers $n, m \in \mathbb{N}$, a matrix $A \in \mathbb{Q}^{n\times m}$, and vectors $b \in \mathbb{Q}^n$, $c \in \mathbb{Q}^m$.
Output: A vector $x^* \in P(A,b)$ minimizing $c^\top x$.

If instead we are interested in finding a vector $x^* \in P$ such that $c^\top x^* \ge c^\top x$ for all $x \in P$, we call this problem LINEAR PROGRAMMING MAXIMIZATION. More generally, we call LINEAR PROGRAMMING the problem of optimizing a linear function over a polyhedron. Linear programming optimization problems are also called *linear programs*. The polyhedron P is also called the *feasible region*, and a vector $x \in P$ is called a *feasible solution*. If $P \neq \emptyset$ the linear program is called *feasible*, and *infeasible* otherwise. If x^* is a feasible solution such that $c^\top x^*$ is *optimal*, we say that x^* is an *optimal* solution. A feasible linear program is called *bounded* if it has *optimal* solutions, and *unbounded* otherwise.

The first practical method to solve LINEAR PROGRAMMING was the *simplex method* designed by Dantzig around 1947 [18]; however, no variant of this method is known to be a polynomial-time algorithm. The first such algorithm (known as the *ellipsoid algorithm*) was developed in 1979 by Khachiyan [61]; nevertheless, this algorithm turned out to be very slow in practice. In 1984, Karmarkar proposed a fast (both theoretically and in practice) *interior point* algorithm for LINEAR PROGRAMMING [58].

Let $P \subseteq \mathbb{R}^m$ and $x \in \mathbb{Q}^m$. The problem of deciding whether $x \in P$ or not is called SEPARATION. An important result of Grötschel, Lovász, and Schrijver implies that, in a sense, LINEAR PROGRAMMING and SEPARATION are equivalent [46]. Their result applies only to *proper* classes of polyhedra. We do not define what *proper* means in this context; however, almost all classes of interesting polyhedra are proper.

Theorem 1.8 (Grötschel, Lovász and Schrijver) *For any proper class of polyhedra,* LINEAR PROGRAMMING *is solvable in polynomial time if and only if* SEPARATION *is solvable in polynomial time.*

1.5.3 Integer Programming

More often than not, we are only interested in the *integral* solutions of linear programs.

Problem: INTEGER PROGRAMMING MINIMIZATION.
Input: Numbers $n, m \in \mathbb{N}$, a matrix $A \in \mathbb{Q}^{n\times m}$, and vectors $b \in \mathbb{Q}^n$, $c \in \mathbb{Q}^m$.
Output: An integer vector $x^* \in P(A, b)$ minimizing $c^\top x$.

We define the problems INTEGER PROGRAMMING MAXIMIZATION and INTEGER PROGRAMMING in a similar way as before. Unfortunately, the decision version of INTEGER PROGRAMMING is $\mathcal{NP}$-complete (see [42, page 245]). An interesting aspect of INTEGER PROGRAMMING is that most combinatorial optimization problems can be easily formulated as *integer programs*. For example, in the instance of SATISFIABILITY introduced before, we can write the condition that variable x_1 must be true or false with the constraints $x_1 \geq 0$ and $x_1 \leq 1$, and the condition that clause $C_1 = \{x_1, x_2, \neg x_4\}$ must be satisfied with the constraint $x_1 + x_2 + (1 - x_4) \geq 1$. In this way, an instance of SATISFIABILITY with n variables and m clauses can be described by a system of $2n + m$ linear constraints plus the integrality condition on the variables. More generally, given a finite set S, a possibly infinite set $\mathcal{F}$ of *multisets* on S, and a vector $c \in \mathbb{Q}^S$, we are often interested in the problem of finding $F^* \in \mathcal{F}$ such that $c(F^*) = \max\{c(F) : F \in \mathcal{F}\}$. One can obtain a *linear programming formulation* of this problem by obtaining the convex hull of the set $\mathcal{X}$ of incidence vectors of multisets in $\mathcal{F}$. Note that $\min\{c(F) : F \in \mathcal{F}\} = \min\{c^\top x : x \in \text{convhull}(\mathcal{X})\}$.

Very often, to obtain a description of the polyhedron convhull($\mathcal{X}$) is extremely difficult. In those cases that it is possible, it is a good idea to start with an *integer programming formulation* of the problem (that is, an instance of INTEGER PROGRAMMING whose set of solutions is precisely $\mathcal{X}$), and then to ignore the integrality condition on the variables to obtain a *linear programming relaxation*. More often than not (in particular for $\mathcal{NP}$-hard problems), we will not have yet a description of the polyhedron convhull($\mathcal{X}$), but in this case, we can always add valid inequalities for convhull($\mathcal{X}$) to our linear program.

1.5.4 Approximation Algorithms

As we mentioned in the opening of this section, when we consider an $\mathcal{NP}$-hard problem, sometimes we settle for a polynomial-time algorithm that outputs a feasible solution to our problem which is not far from optimal. We are particularly interested on those algorithms for which we can give a bound on how far from optimal is the solution obtained.

Let P be a minimization problem with set of instances $I(P)$, let $\alpha : I(P) \to \mathbb{R}_+$, and let $\mathcal{A}$ be a polynomial-time algorithm. Then $\mathcal{A}$ is said to be an *α-approximation algorithm* for P if, for every instance i of P with minimal solution $P(i)$, $\mathcal{A}$ outputs a feasible solution $\mathcal{A}(i)$ such that $\mathcal{A}(i) \leq \alpha(i) \cdot P(i)$. We call α the *approximation guarantee* of $\mathcal{A}$, and we say that $\mathcal{A}$ *approximates* P *within a factor* of α. Note that $\alpha(i) \geq 1$ for all instances i. Let $\mathcal{A} = \{\mathcal{A}_\epsilon : \epsilon > 0\}$ be a set of algorithms for P. If for each fixed $\epsilon > 0$ we have that $\mathcal{A}_\epsilon$ is a $(1+\epsilon)$-approximation algorithm for P then we say that $\mathcal{A}$ is a *polynomial approximation scheme*. If in addition each $\mathcal{A}_\epsilon$ runs in polynomial time with respect to the size of the input and $\frac{1}{\epsilon}$, then $\mathcal{A}$ is said to be a *fully polynomial approximation scheme*. Similar definitions can be given for maximization problems. Papadimitriou and Yannakakis introduced the notion of *hardness of approximation* [72]. We say that a problem P is *Max-$\mathcal{SNP}$ hard* if, for some fixed $\epsilon > 0$, there is no $(1+\epsilon)$-approximation algorithm for P, unless $\mathcal{P} = \mathcal{NP}$.

The area of approximation algorithms is very active. We refer the reader to the books by Ausiello et al. [3], Hochbaum [53], and Vazirani [83].

Chapter 2

The Eulerian Tour Problem and the Chinese Postman Problem

Ach, alles ereignet sich einmal nur, aber einmal muß alles geschehen.
Über Berg und Tal, über Feld und Flur werd' ich vergehen, verwehen...
Die unendliche Geschichte, Michael Ende

We introduce the EULERIAN TOUR problems for undirected, directed and mixed graphs. In each case, we give necessary and sufficient conditions for the existence of an Eulerian tour. Later we introduce the CHINESE POSTMAN problems for undirected and directed graphs as generalizations of the corresponding EULERIAN TOUR problems.

2.1 The Eulerian Tour Problem

In 1736, the great Swiss mathematician Leonhard Euler wrote [33]:

> *The problem, which I am told is widely known, is as follows: in Königsberg in Prussia, there is an island A, called the Kneiphof; the river which surrounds it is divided into two branches, as can be seen in [Figure 2.1], and these branches are crossed by seven bridges a, b, c, d, e, f and g. Concerning these bridges, it was asked whether anyone could arrange a route in such a way that he would cross*

> *each bridge once and only once. I was told that some people asserted that this was impossible, while others were in doubt; but nobody would actually assert that it could be done. From this, I have formulated the general problem: whatever be the arrangement and division of the river into branches, and however many bridges there be, can one find out whether or not it is possible to cross each bridge exactly once?* [9, page 3]

It is widely recognized that Euler's paper gave birth to what he called the *geometry of position*, that is, to *graph theory*. Moreover, the problem he formulated became the first in a class of problems that we call *arc routing* problems. In this chapter, we will concentrate in Euler's original question, as well as in some variants and generalizations. For more on Eulerian graphs see the two books by Fleischner [35, 36], and for more on arc routing problems see the book edited by Dror [21], or the surveys by Ahr [1], Assad and Golden [2], Eiselt, Gendreau, and Laporte [31, 32], and Guan [50].

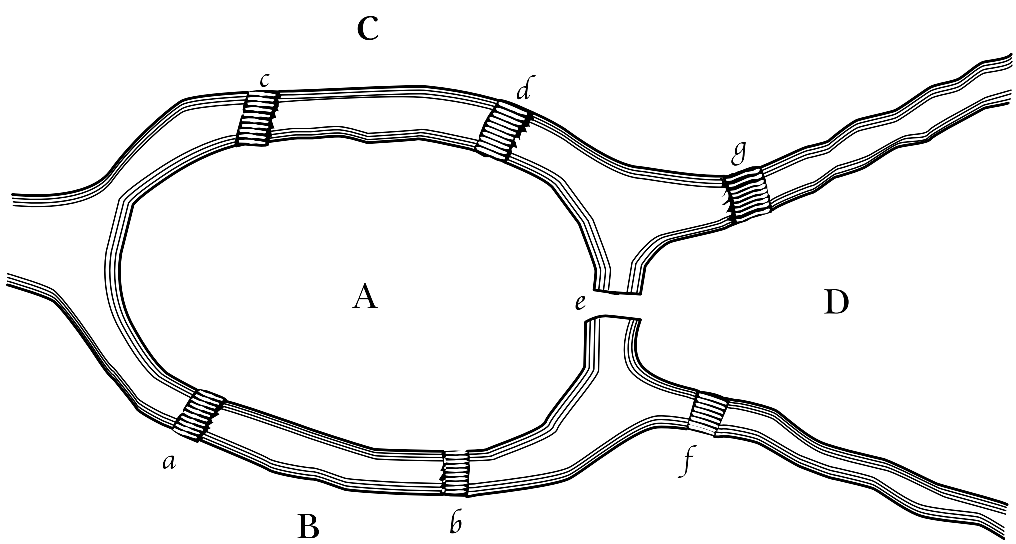

Figure 2.1: The seven bridges of Königsberg.

2.1.1 Undirected Eulerian Tours

Let $G = (V, E)$ be an undirected graph. A tour of G is *Eulerian* if it contains each edge of G exactly once. We can reformulate Euler's question as follows:

Problem: UNDIRECTED EULERIAN TOUR.
Input: An undirected graph $G = (V, E)$.
Output: Does G have an Eulerian tour?

We say that an undirected graph is *Eulerian* if it has an Eulerian tour. Euler gave necessary conditions for an undirected graph to be Eulerian [33]. A proof that they were also sufficient came more than a century later, in a posthumous paper by Hierholzer, in 1873 [52]. A second characterization was given in 1912 by Veblen [84].

Theorem 2.1 (Euler, Hierholzer) *Let G be a connected, undirected graph. Then G is Eulerian if and only if every vertex of G has even degree.*

Theorem 2.2 (Veblen) *Let G be a connected, undirected graph. Then G is Eulerian if and only if G is the disjoint union of some cycles.*

In Figure 2.2 we show on the left-hand side the undirected graph corresponding with the seven bridges of Königsberg. Note that this undirected graph is connected but has vertices with odd degree, hence it is not Eulerian. On the right-hand side, we show a connected undirected graph all whose vertices have even degree, hence it is Eulerian. An Eulerian tour of this undirected graph is

$$(A, a, B, b, A, c, C, d, A, e, D, f, B, i, C, g, D, h, A). \tag{2.1}$$

As Euler pointed out, once we determined that an undirected graph G has an Eulerian tour, we still need to find it. We can reformulate this question as follows:

Problem: UNDIRECTED EULERIAN TOUR CONSTRUCTION.
Input: An undirected graph $G = (V, E)$.
Output: An Eulerian tour T of G, if it exists.

Apparently, Euler did not consider this question as being important, and he did not give explicit details regarding the construction of an Eulerian tour:

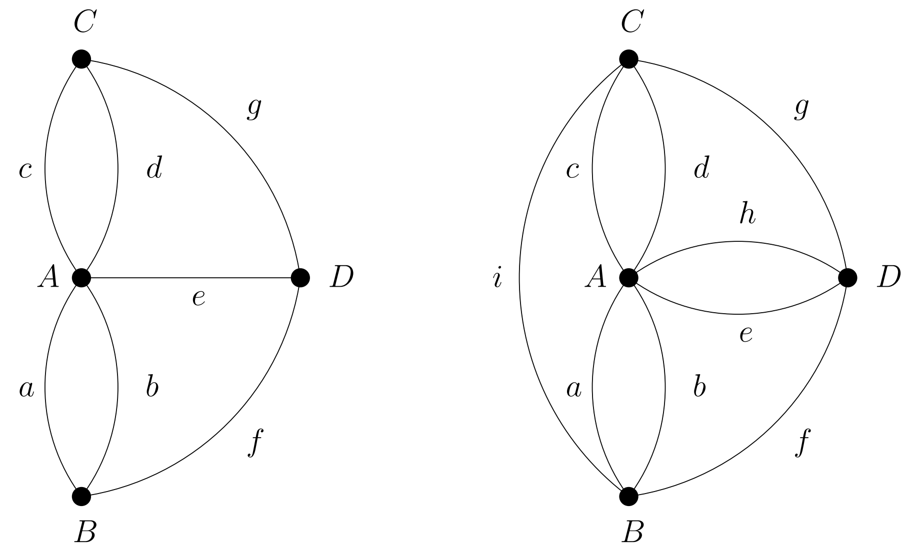

Figure 2.2: A non-Eulerian undirected graph and an undirected Eulerian graph.

> *When it has been determined that such a journey can be made, one still has to find how it should be arranged [...] [I]t is [...] an easy task to construct the required route [...] I do not therefore think it worthwhile to give any further details concerning the finding of the routes.* [9, page 8]

These details were given for the first time by Hierholzer [52].

Theorem 2.3 (Hierholzer) *There exists a linear-time algorithm that, given an undirected graph G, either finds an Eulerian tour of G, or shows that G is not Eulerian.*

2.1.2 Directed Eulerian Tours

We can consider a first variation of Euler's question as follows: Let $D = (V, A)$ be a directed graph. A tour of D is *Eulerian* if it contains each arc of D exactly once.

Problem: DIRECTED EULERIAN TOUR.
Input: A directed graph $D = (V, A)$.
Output: Does D have an Eulerian tour?

We say that a directed graph is *Eulerian* if it has an Eulerian tour. In his 1936 book, Kőnig gave a characterization of directed Eulerian graphs [62, page 29].

Theorem 2.4 (Kőnig) *Let D be a connected, directed graph. Then D is Eulerian if and only if, for every vertex of D, its indegree and outdegree are equal.*

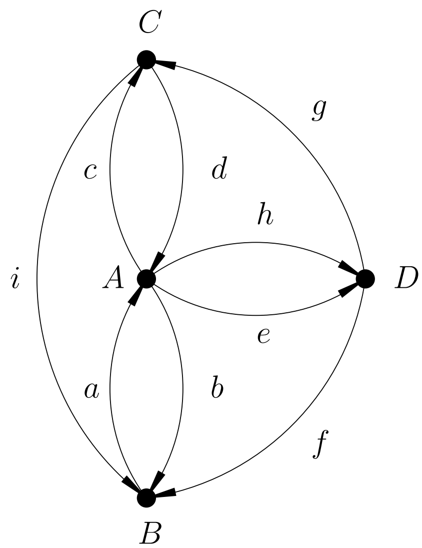

Figure 2.3: A non-Eulerian directed graph and a directed Eulerian graph.

On the left-hand side of Figure 2.3 we show a connected directed graph having vertices whose indegree and outdegree are not equal, hence it is not Eulerian. On the right-hand side, we show a connected directed graph each of whose vertices have its indegree equal to its outdegree, hence it is Eulerian. An Eulerian tour of this directed graph is

$$(A, c, C, i, B, a, A, e, D, g, C, d, A, h, D, f, B, b, A). \tag{2.2}$$

As before, we have the problem of constructing Eulerian tours of directed graphs.

Problem: DIRECTED EULERIAN TOUR CONSTRUCTION.
Input: A directed graph $D = (V, A)$.
Output: An Eulerian tour T of D, if it exists.

The algorithm mentioned in the previous section can be adapted to solve this problem.

2.1.3 Mixed Eulerian Tours

A second variation of Euler's question is as follows: Let $M = (V, E, A)$ be a mixed graph. A tour of M is *Eulerian* if it contains each edge and arc of M exactly once.

Problem: MIXED EULERIAN TOUR.
Input: A mixed graph $M = (V, E, A)$.
Output: Does M have an Eulerian tour?

We say that a mixed graph is *Eulerian* if it has an Eulerian tour. In their 1962 book *Flows in Networks*, Ford and Fulkerson gave a characterization of mixed Eulerian graphs [38, page 60] based on Hoffman's circulation theorem [54].

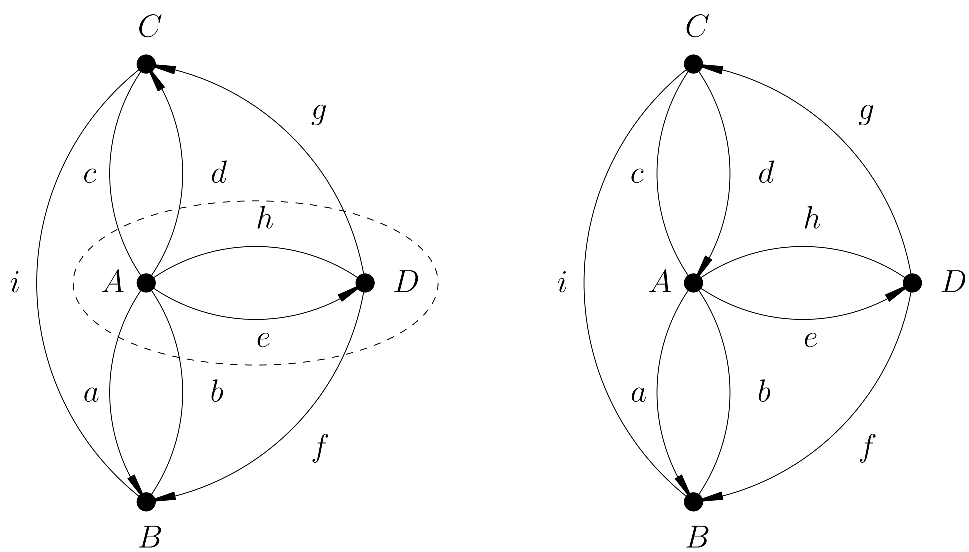

Figure 2.4: A non-Eulerian mixed graph and a mixed Eulerian graph.

Theorem 2.5 (Ford and Fulkerson) *Let M be a connected, mixed graph. Then M is Eulerian if and only if, for every subset S of vertices of M, the number of arcs and edges from $\bar{S}$ to S minus the number of arcs from S to $\bar{S}$ is a nonnegative even number.*

We observe that Veblen's characterization of undirected Eulerian graphs in terms of cycle decompositions applies also to directed and mixed graphs.

On the left-hand side of Figure 2.4 we show a connected mixed graph for which $S = \{A, D\}$ fails the condition of Theorem 2.5, and hence it is not Eulerian. On the right-hand side, we show a mixed Eulerian graph, one of whose Eulerian tours is

$$(A, a, B, i, C, d, A, e, D, f, B, b, A, h, D, g, C, c, A). \tag{2.3}$$

Consider the problem of constructing Eulerian tours of mixed graphs.

Problem: Mixed Eulerian Tour Construction.
Input: A mixed graph $M = (V, E, A)$.
Output: An Eulerian tour T of M, if it exists.

Ford and Fulkerson's proof of Theorem 2.5 also gives an algorithm, based on network flow techniques, to solve Mixed Eulerian Tour Construction. The much simpler algorithms for undirected and directed graphs do not work in this case.

Theorem 2.6 (Ford and Fulkerson) *There exists a polynomial-time algorithm that, given a mixed graph M, either finds an Eulerian tour of M, or shows that M is not Eulerian.*

2.2 The Undirected Postman Problem

In 1960, the Chinese mathematician Mei Gu Guan proposed the following:

> *When the author was plotting a diagram for a postman's route, he discovered the following problem: "A postman has to cover his assigned segment before returning to the post office. The problem is to find the shortest walking distance for the postman."* [48]

Guan's problem became widely known as the *Chinese postman* problem after a talk by Edmonds in 1965 [25]. However, due to the enormous variety of similarly posed questions, throughout this thesis we will refer collectively to all of them simply as *postman problems*, dropping the adjective *Chinese*, and adding more descriptive adjectives. For example, we will refer to Guan's original problem as Minimum Undirected Postman Tour because the objective is to *minimize* the length of a postman's route in an *undirected* graph.

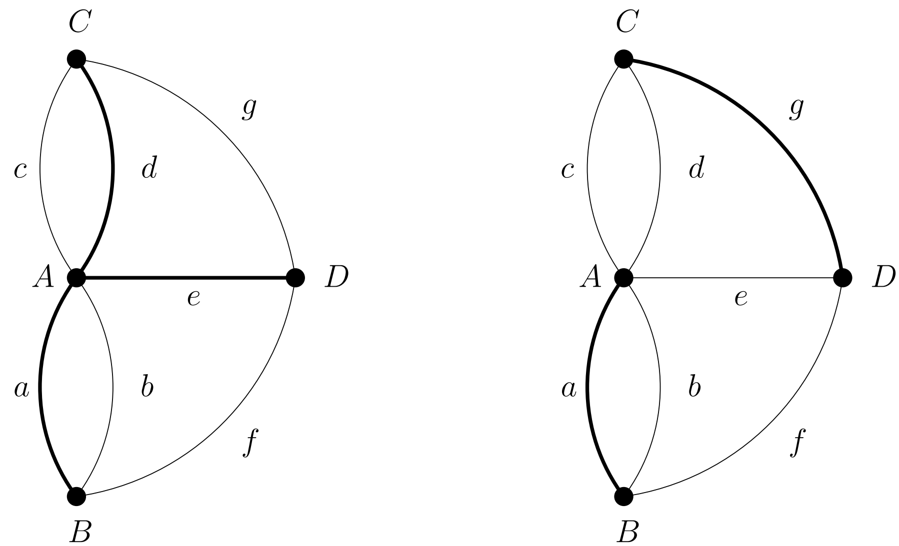

Figure 2.5: A non-Eulerian undirected graph and two postman sets.

Naturally, if the postman's route corresponds with an Eulerian graph the answer is given by the length of one of its Eulerian tours. More generally, given an undirected graph G, a tour of G is a *postman* tour if it contains each edge of G at least once. A postman tour of the non-Eulerian undirected graph shown on Figure 2.5 is

$$(A, a, B, b, A, a, B, f, D, e, A, e, D, g, C, d, A, c, C, d, A). \tag{2.4}$$

Since the graph obtained from a connected undirected graph G by duplicating each of its edges is Eulerian, both the problem of deciding whether G has a postman tour or not, and the problem of finding one such tour are as easy as the corresponding problems for Eulerian tours of undirected graphs.

The first interesting problem is, as Guan asked, to find the length of a shortest postman tour. Given a connected undirected graph $G = (V, E)$, for each edge $e \in E$ a nonnegative cost c_e, and a postman tour T of G, the *cost* of T (denoted by $c(T)$) is the sum of the costs of the edges used by T, that is, if $T = (v_1, e_1, v_2, e_2, \ldots, v_k, e_k, v_1)$, then

$$c(T) \equiv \sum_{i=1}^{k} c(e_i). \tag{2.5}$$

Problem: Minimum Undirected Postman Tour.
Input: A connected undirected graph $G = (V, E)$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost $\mathrm{MUPP}(G, c)$ of a postman tour of G.

Assuming all its edges have cost one, a minimum cost postman tour of the non-Eulerian undirected graph shown on Figure 2.5 is

$$(A, a, B, b, A, a, B, f, D, g, C, c, A, d, C, g, D, e, A). \tag{2.6}$$

2.2.1 Formulations

Let T be a postman tour of G and, for every edge e, let x_e be the number of times that T uses e, that is, let x be the incidence vector of T. Euler's Theorem implies that x must satisfy that for every vertex $v \in V$ the number $x(\delta(v))$ is even, and for every edge $e \in E$ the number x_e is a positive integer. Moreover, it also follows that any vector x satisfying these two conditions is the incidence vector of a postman tour of G. Hence, we obtain the following integer programming formulation for Minimum Undirected Postman Tour

$$\begin{aligned} \mathrm{MUPP}(G, c) &= \min c^\top x && (2.7)\\ \text{subject to} & && \\ x(\delta(v)) &\equiv 0 \pmod 2 \text{ for all } v \in V && (2.8)\\ x_e &\geq 1 \text{ and integer for all } e \in E. && (2.9) \end{aligned}$$

An easy consequence is that Minimum Undirected Postman Tour has optimal solutions with $x_e \leq 2$ for every edge e: If $x_e \geq 3$ for some edge e, then decreasing this component by 2 does affect feasibility, and does not increase the objective value.

An equivalent way of looking at Guan's problem is then to find the minimum length of the edges that are traversed twice by a postman tour. We say that a subset $F \subseteq E$ is a *postman set* of G if the graph obtained from G by duplicating the edges in F is Eulerian.

Problem: Minimum Undirected Postman Set.
Input: A connected undirected graph $G = (V, E)$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost $\mathrm{MUPSP}(G, c)$ of a postman set of G.

Figure 2.5 shows in bold the edges of two different postman sets of an undirected graph. Assuming all the edges have cost one, $\{a, g\}$ is a minimum cost postman set of the graph shown, corresponding with the minimum cost postman tour mentioned before.

Let $F \subseteq E$ and let x be its incidence vector. It follows from Euler's Theorem that F is a postman set if and only if $x(\delta(v)) + d(v)$ is even for every $v \in V$, and x_e is a nonnegative integer for every $e \in E$. Therefore, the following is an integer programming formulation for Minimum Undirected Postman Set:

$$\begin{aligned} \mathrm{MUPSP}(G, c) &= \min c^\top x && (2.10) \\ \text{subject to} & && \\ x(\delta(v)) &\equiv d(v) \pmod 2 \text{ for all } v \in V && (2.11) \\ x_e &\geq 0 \text{ and integer for all } e \in E. && (2.12) \end{aligned}$$

We say that a vertex v is *odd* if its degree is odd, and we say that v is *even* otherwise. Similarly, we say that a subset $S \subseteq V$ is *odd* if it contains an odd number of odd vertices.

Let $\mathcal{P}_{\mathrm{UPS}}(G)$ be the convex hull of all feasible solutions x to the integer program above. Consider the linear program

$$\begin{aligned} \mathrm{LMUPSP}(G, c) &= \min c^\top x && (2.13) \\ \text{subject to} & && \\ x(\delta(S)) &\geq 1 \text{ for all odd } S \subseteq V && (2.14) \\ x_e &\geq 0 \text{ for all } e \in E, && (2.15) \end{aligned}$$

and let $\mathcal{Q}_{\mathrm{UPS}}(G)$ be the set of its feasible solutions. Note that in this linear program the parity constraints (2.11) have been replaced by the *odd-cut* constraints (2.14) obtained from the following observation: if $S \subseteq V$ is odd, then any postman tour of G must use at least twice some edge in $\delta_G(S)$. Equivalently, if $S \subseteq V$ is odd, then any postman set of G must contain at least one edge in $\delta_G(S)$.

2.2.2 Algorithms

In 1965, Edmonds presented an algorithm for Minimum Undirected Postman Tour [25] based on his algorithm for maximum matching [26], as well as on the all-shortest paths algorithm of Floyd [37] and Warshall [88].

Theorem 2.7 (Edmonds) *There exists a polynomial-time algorithm that, given a connected, undirected graph G, and nonnegative rational costs on its edges, computes a minimum cost postman tour of G.*

In 1973, Edmonds and Johnson strengthened this result by showing that the integer programming formulation of Minimum Undirected Postman Set and the linear programming relaxation LMUPSP(G, c) are equivalent [29]. Their main result is:

Theorem 2.8 (Edmonds and Johnson) *Let G be a connected, undirected graph. Then $\mathcal{P}_{\text{UPS}}(G) = \mathcal{Q}_{\text{UPS}}(G)$. Moreover, there exists a polynomial-time algorithm that, given nonnegative rational costs c on the edges of G, solves the linear program LMUPSP(G, c), and hence computes the minimum cost of a postman tour of G.*

Using that Minimum Cut can be solved in polynomial time, Padberg and Rao proved that Minimum Odd Cut can also be solved in polynomial time [69].

Theorem 2.9 (Padberg and Rao) *Let $G = (V, E)$ be a connected, undirected graph, let $T \subseteq V$ have even cardinality, and let $x \in \mathbb{Q}_+^E$. There exists a polynomial-time algorithm to find a subset $S \subseteq V$, with $|S \cap T|$ odd, minimizing $x(\delta(S))$.*

Due to the equivalence between optimization and separation, Theorem 1.8, we can use this result as follows: Let T be the set of vertices of G with odd degree. Note that a subset $S \subseteq V$ has $|S \cap T|$ odd if and only if S is odd. Hence, if we want to decide whether a vector $x \in \mathbb{Q}_+^E$ belongs to $\mathcal{Q}_{\text{UPS}}(G)$, it is enough to find an odd subset $S^* \subseteq V$ minimizing $x(\delta(S))$, since $x \in \mathcal{Q}_{\text{UPS}}(G)$ if and only if $x(\delta(S^*)) \geq 1$. This observation, together with the ellipsoid algorithm for linear programming, gives a polynomial-time algorithm to solve Minimum Undirected Postman Set.

2.2.3 T-joins

We consider a generalization of Minimum Undirected Postman Set. Let $G = (V, E)$ be an undirected graph, and let $T \subseteq V$ with $|T|$ even. We say that $J \subseteq E$ is a *T-join* if, for every $v \in V$, $d_J(v)$ is odd if and only if $v \in T$.

Problem: MINIMUM T-JOIN.
Input: An undirected graph $G = (V, E)$, a subset $T \subseteq V$, with $|T|$ even, and a vector $c \in \mathbb{Q}^E$.
Output: The minimum cost $\mathrm{MTJ}(G, c)$ of a T-join of G.

Edmonds proved that MINIMUM T-JOIN can be solved in polynomial time [25].

It is not difficult to verify that the postman sets of G are precisely the T-joins of G where T is the set of vertices of G with odd degree, and the $\emptyset$-joins of G are precisely the even subgraphs of G. Also, it is not hard to see that a T-join is *minimal* if and only if it is the union of $\frac{1}{2}|T|$ edge disjoint paths whose set of endpoints is precisely T.

Edmonds and Johnson introduced two linear programming formulations for MINIMUM T-JOIN, one for nonnegative costs, and another for arbitrary costs [29]. We say that $S \subseteq V$ is T-*odd* if $S \cap T$ is odd. If $J \subseteq E$ is a T-join and $S \subseteq V$ is T-odd then the subgraph $G_S = (S, J \cap \gamma(S))$ has an even number of vertices of odd degree, but the subgraph $G_J = (V, J)$ has an odd number of vertices in S of odd degree. Hence $|J \cap \delta(S)|$ must be odd. In particular, if $x \in \mathbb{Z}_+^E$ is the incidence vector of J then $x(\delta(S)) \geq 1$. Let $\mathcal{Q}_{TJ}^1(G, T)$ be the set of solutions of the linear program

$$\begin{aligned} \mathrm{LMTJ1}(G, T, c) &= \min c^\top x && (2.16)\\ \text{subject to} & && \\ x(\delta(S)) &\geq 1 \text{ for all } T\text{-odd } S \subseteq V && (2.17)\\ x_e &\geq 0 \text{ for all } e \in E, && (2.18)\end{aligned}$$

and let $\mathcal{P}_{TJ}^1(G, T)$ be the convex hull of its integral solutions. Edmonds and Johnson proved that $\mathcal{Q}_{TJ}^1(G, T) = \mathcal{P}_{TJ}^1(G, T)$, and hence if $c \in \mathbb{Q}_+^E$ then $\mathrm{LMTJ1}(G, T, c)$ is equal to the minimum cost of a T-join.

Now we describe the second formulation. If $J \subseteq E$ is a T-join and $S \subseteq V$ then $|J \cap \delta(S)|$ and $|S \cap T|$ have the same parity. Hence, if $F \subseteq \delta(S)$ and $|F| + |S \cap T|$ is odd, then F and $J \cap \delta(S)$ cannot be equal and, in particular, their symmetric difference contains at least one edge. It follows that the characteristic vector x of J satisfies

$$x(\delta(S) \setminus F) + (|F| - x(F)) \geq 1. \qquad (2.19)$$

Let $\mathcal{P}^2_{TJ}(G,T)$ be the convex hull of the characteristic vectors of T-joins, and let $\mathcal{Q}^2_{TJ}(G,T)$ be the set of solutions of the linear program

$$\begin{aligned} \mathrm{LMTJ2}(G,T,c) &= \min c^\top x && (2.20)\\ \text{subject to} & && \\ x(\delta(S)\setminus F) - x(F) &\geq 1 - |F| \text{ for all } S \subseteq V \text{ and all } F \subseteq \delta(S) && (2.21)\\ & \quad \text{such that } |F| + |S \cap T| \text{ is odd} && \\ 1 \geq x_e &\geq 0 \text{ for all } e \in E. && (2.22) \end{aligned}$$

Again, Edmonds and Johnson proved that $\mathcal{Q}^2_{TJ}(G,T) = \mathcal{P}^2_{TJ}(G,T)$, and hence for every $c \in \mathbb{Q}^E$, LMTJ2(G,T,c) is equal to the minimum cost of a T-join.

The combinatorial structure of T-joins has been studied extensively. The cuts induced by T-odd sets are called *T-cuts*. Frank wrote a survey on T-joins and T-cuts [39].

2.2.4 Applications

In addition to its practical applications to vehicle routing, Minimum Undirected Postman Tour has various other theoretical applications. We mention very briefly only some of them here. We direct the interested reader to the survey by Barahona [6].

Spin glasses are magnetic alloys with interesting physical properties. One of the main problems of interest is the determination of configurations of minimum energy. There exists a mathematical model for configurations of spin glasses for which the problem of finding a minimum energy state can be reduced to solving a Minimum Undirected Postman Tour [4]. In the design of a two layer VLSI circuit, one may need to add interconnections (called *vias*) between its two layers. The via minimization problem consists of minimizing the number of such interconnections. This problem can also be reduced to solving a Minimum Undirected Postman Tour [7]. In fact, it can be seen that the via minimization problem is a special case of Maximum Cut on planar graphs, which can be solved in polynomial time, as shown by Barahona [8] and Hadlock [51]. Moreover, it is known that Maximum Cut can be solved in polynomial time for graphs not contractible to K_5, by solving a sequence of Minimum Undirected Postman Tour instances [5]. Recall that Maximum Cut is $\mathcal{NP}$-complete for general graphs [59].

2.3 The Directed Postman Problem

A first variation of Guan's question is to consider directed graphs instead of undirected graphs. Given a directed graph D, a tour of D is a *postman* tour if it contains each arc of D at least once. A postman tour of the directed graph shown on Figure 2.6 is

$$(A, c, C, i, B, a, A, d, C, i, B, a, A, e, D, f, B, a, A, e, D, g, C, i, B, b, A, e, D, h, A). \tag{2.23}$$

Note that some arcs are used more than twice, and that this cannot be avoided.

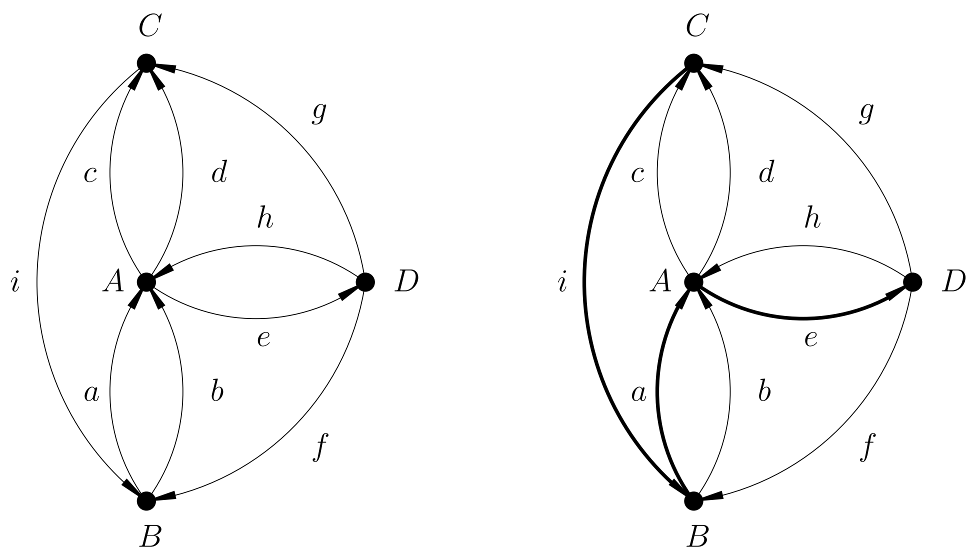

Figure 2.6: A non-Eulerian directed graph and a postman tour.

Since a directed graph has a postman tour if and only if it is strongly connected, it is easy to decide whether a directed graph has a postman tour or not. A more interesting problem is to obtain a postman tour of a directed graph.

Problem: DIRECTED POSTMAN TOUR CONSTRUCTION.
Input: A strongly connected directed graph $D = (V, A)$.
Output: A postman tour T of D.

For the moment, we are not going to discuss here how to do this. Instead, we are going to consider the problem of finding the length of a shortest postman tour. Given a strongly connected directed graph $D = (V, A)$, for each arc $a \in A$ a nonnegative rational cost c_a, and a postman tour T of D, the *cost* of T (denoted by $c(T)$) is the sum of the costs of the arcs used by T, that is, if $T = (v_1, a_1, v_2, a_2, \ldots, v_k, a_k, v_1)$ then $c(T) \equiv \sum_{i=1}^{k} c(e_i)$.

Problem: Minimum Directed Postman Tour.
Input: A strongly connected directed graph $D = (V, A)$, and a vector $c \in \mathbb{Q}_+^A$.
Output: The minimum cost $\mathrm{MDPP}(D, c)$ of a postman tour of D.

2.3.1 Formulation and Algorithms

Let T be a postman tour of D and, for every arc a, let x_a be the number of times that T uses a, that is, let x be the incidence vector of T. It follows from Kőnig's Theorem 2.4 that x must satisfy the conditions: For every vertex $v \in V$ the quantities $x(\delta(v))$ and $x(\delta(\bar{v}))$ are equal, and for every edge $e \in E$ the number x_e is a positive integer. Moreover, it also follows that any vector x satisfying these two conditions is the incidence vector of a postman tour of D. Hence, we obtain the following integer programming formulation for Minimum Directed Postman Tour:

$$
\begin{aligned}
\mathrm{MDPP}(D, c) &= \min c^\top x && (2.24)\\
\text{subject to} & && \\
x(\delta(\bar{v})) - x(\delta(v)) &= 0 \text{ for all } v \in V && (2.25)\\
x_a &\geq 1 \text{ and integer for all } a \in A. && (2.26)
\end{aligned}
$$

Given a directed graph $D = (V, A)$, a vector $l \in \mathbb{R}^A$ of *lower bounds* on the arcs, a vector $u \in \mathbb{R}^A$ of *upper bounds* on the arcs, and a vector $b \in \mathbb{R}^V$ of *demands* on the vertices, we say that a vector $x \in \mathbb{R}^A$ is a *(feasible) flow* if $l_a \leq x_a \leq u_a$ for all $a \in A$, and $x(\delta(\bar{v})) - x(\delta(v)) = b_v$ for all $v \in V$. The following result of Gale [41] gives necessary and sufficient conditions for the existence of feasible flows:

Theorem 2.10 (Gale) *Given a directed graph $D = (V, A)$, and vectors $l, u \in \mathbb{R}^A$, and $b \in \mathbb{R}^V$, there exists a flow $x \in \mathbb{R}^A$ if and only if $l \leq u$, $b(V) = 0$, and every $S \subseteq V$ satisfies*

$$u(\delta(\bar{S})) \geq b(S) + l(\delta(S)). \tag{2.27}$$

Moreover, if l, u, and b are integral, there exists an integral flow x if and only if the same conditions hold.

If $b_v = 0$ for all $v \in V$, a feasible flow is called a *circulation*. Note that the incidence vectors of postman tours of a directed graph are circulations. Hoffman's circulation theorem is a special case of the above [54].

Theorem 2.11 (Hoffman) *Given a directed graph $D = (V, A)$, and vectors $l, u \in \mathbb{R}^A$, there exists a circulation $x \in \mathbb{R}^A$ if and only if $l \leq u$, and every $S \subseteq V$ satisfies*

$$u(\delta(\bar{S})) \geq l(\delta(S)). \tag{2.28}$$

Moreover, if l and u are integral, there exists an integral circulation x if and only if the same conditions hold.

If we assign costs to the arcs, we are interested in the problem of finding a minimum cost feasible flow, or in the problem of finding a minimum cost circulation. These two problems, as well as MINIMUM DIRECTED POSTMAN TOUR, are special cases of:

Problem: MINIMUM COST FEASIBLE FLOW.
Input: A directed graph $D = (V, A)$, vectors $c \in \mathbb{Q}_+^A$, $l, u \in \mathbb{Z}^A$ (with $l \leq u$), and a vector $b \in \mathbb{Z}^V$ with $b(V) = 0$.
Output: The minimum cost $\mathrm{MCFF}(D, c, l, u, b)$ of a feasible flow x.

A polynomial-time algorithm to solve this problem is due to Edmonds and Karp [30], although faster algorithms exist (see [80, Chapter 12]). Edmonds and Johnson [29] used this algorithm to solve MINIMUM DIRECTED POSTMAN TOUR.

Theorem 2.12 (Edmonds and Karp) *Let $D = (V, A)$ be a directed graph, and let $c \in \mathbb{Q}_+^A$, $l, u \in \mathbb{Z}^A$ (with $l \leq u$), and $b \in \mathbb{Z}^V$ (with $b(V) = 0$). There exists a polynomial-time algorithm that finds a minimum cost feasible flow of (D, c, l, u, b).*

Theorem 2.13 (Edmonds and Johnson) *Let $D = (V, A)$ be a strongly connected directed graph, and let $c \in \mathbb{Q}_+^A$ be a vector of costs on its arcs. There exists a polynomial-time algorithm that finds a minimum cost postman tour of (D, c).*

Chapter 3

The Mixed Postman Problem

Doublethink means the power of holding two contradictory beliefs in one's mind simultaneously, and accepting both of them.

Nineteen Eighty-Four, George Orwell

We introduce a generalization of Mixed Eulerian Tour known as Minimum Mixed Postman Tour. We give integer programming formulations for this problem, and prove properties of their linear relaxations. We show that Minimum Mixed Postman Tour is $\mathcal{NP}$-hard under various assumptions, and describe approximation algorithms for it. Finally, we introduce a *bounded* version of the problem.

3.1 Introduction

In the previous chapter we introduced Mixed Eulerian Tour, and listed the conditions under which it has solutions. Just as we did for its undirected and directed counterparts, we relax now the condition that each edge and arc must be traversed exactly once. Given a mixed graph M, a tour of M is a *postman tour* if it contains each edge and arc of M at least once. A postman tour of the non-Eulerian mixed graph shown on Figure 3.1 is:

$$(A, a, B, b, A, d, C, c, A, e, D, f, b, A, h, D, g, i, B, b, A). \tag{3.1}$$

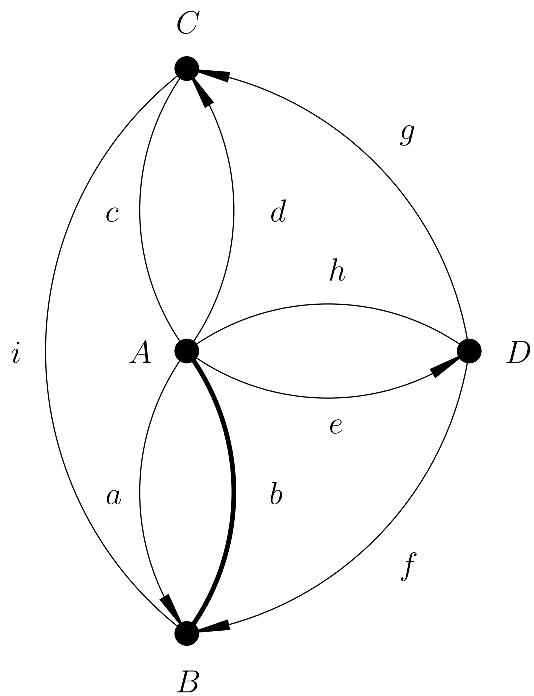

Figure 3.1: A non-Eulerian mixed graph and two postman tours.

As in the directed case, some edges or arcs are used more than twice, and this cannot be avoided. As in the undirected and directed cases, we consider the problems of deciding whether a mixed graph has a postman tour and, in that case, to construct one.

Problem:	MIXED POSTMAN TOUR.
Input:	A mixed graph $M = (V, E, A)$.
Output:	Does M have a postman tour?

Note that if M is not strongly connected then M cannot have a tour. Conversely, if M is strongly connected, define a mixed graph N with the vertex set of M and, for each edge e of M with ends u and v, add to N two parallel edges with ends u and v, and for each arc a of M with head u and tail v, add a to N together with a shortest mixed path of M from u to v. By Veblen's theorem, since N is the disjoint union of some cycles, it is Eulerian, and an Eulerian tour of N determines a postman tour of M. Hence M has a postman tour if and only if M is strongly connected. Moreover, note that N has at most $2|E| + |A| \cdot |V|$ edges and arcs, which is polynomial in the size of M. Hence, we can also use Hierholzer's algorithm to construct a postman tour T of M.

Given a strongly connected mixed graph $M = (V, E, A)$, for each edge or arc e of M a nonnegative rational cost c_e, and a postman tour T of G, the *cost* of T (denoted by $c(T)$) is the sum of the costs of the edges and arcs used by T. In this chapter, we concentrate on the problem of finding a postman tour of M with minimum cost.

Problem: MINIMUM MIXED POSTMAN TOUR.
Input: A strongly connected mixed graph $M = (V, E, A)$, and a vector $c \in \mathbb{Q}_+^{E \cup A}$.
Output: The minimum cost $\mathrm{MMPT}(M, c)$ of a postman tour of M.

Recall that, in the undirected case, we defined a postman set to be a subset of the edge set whose duplication produces an Eulerian graph. We did not define a similar concept for the directed version of the problem; however we define it now for the mixed version. We say that a family F of edges and arcs is a *postman set* of M if there exists a tour T of M using each edge and arc e of M one more than the number of times e appears in F.

Problem: MINIMUM MIXED POSTMAN SET.
Input: A strongly connected mixed graph $M = (V, E, A)$, and a vector $c \in \mathbb{Q}_+^{E \cup A}$.
Output: The minimum cost $\mathrm{MMPS}(M, c)$ of a postman set of M.

We note that the outputs of MINIMUM MIXED POSTMAN TOUR and MINIMUM MIXED POSTMAN SET are related by $\mathrm{MMPT}(M, c) = \mathrm{MMPS}(M, c) + c(E \cup A)$. Hence, from the optimization point of view, these two problems are equivalent.

Edmonds and Johnson introduced MINIMUM MIXED POSTMAN TOUR [29]. Fernandes, Lee, and Wakabayashi gave a polynomial-time algorithm for mixed graphs with *bounded tree width* [34]. A *branch and cut* algorithm for MINIMUM MIXED POSTMAN TOUR was given by Christofides et al. [13], whereas *branch and bound* algorithms were given by Nobert and Picard [68], and Yan and Thompson [91]. Corberán, Martí, and Sanchis [17], Edmonds and Johnson [29], Greistorfer [45], Jeworrek and Schulz [56], Wang [87], and Yaoyuenyong et al. [92] gave heuristics for MINIMUM MIXED POSTMAN TOUR. Pearn and Chou gave methods to improve solutions [73]. Surveys on MINIMUM MIXED POSTMAN TOUR include those by Brucker [11], Eiselt, Gendreau, and Laporte [31], Minieka [67], and Peng [75].

3.2 Computational Complexity

In contrast to MINIMUM UNDIRECTED POSTMAN TOUR and MINIMUM DIRECTED POSTMAN TOUR, which can be solved in polynomial time, Papadimitrou showed that the decision version of MINIMUM MIXED POSTMAN TOUR is $\mathcal{NP}$-complete, even if we restrict the input so that the mixed graph is planar, each of its vertices has total degree at most three, and all edge and arc costs are equal to one [70]. We introduce first a modification of Papadimitriou's construction, which will allow us to reproduce some of his results, namely, that the decision version of MINIMUM MIXED POSTMAN TOUR is $\mathcal{NP}$-complete even if all costs are equal to one. Our proof, however, will also give a negative result about the approximability of MINIMUM MIXED POSTMAN SET. We modify further our construction to prove that the decision version of MINIMUM MIXED POSTMAN TOUR is $\mathcal{NP}$-complete even if the costs arise from either the Euclidean or the Manhattan metric in $\mathbb{Q}^2$.

3.2.1 Assumptions on Satisfiability

Consider an instance I of 3-SATISFIABILITY with $n \in \mathbb{N}$ variables $x_1, x_2, \ldots, x_n$ and $m \in \mathbb{N}$ clauses $C_1, C_2, \ldots, C_m$, where each clause C_i contains exactly three literals y_i^1, y_i^2, y_i^3, each of which is either a positive variable (say x_j) or a negative variable (say $\neg x_j$). Note that $size(I)$ is $O(m \log n)$, which is $O(m \log m)$ since we can clearly assume that $n \leq 3m$. We will assume further the following:

1. No clause contains a variable and its negation: these clauses are trivially satisfied and can be removed from I. This condition can be verified in time $O(m)$.

2. No two clauses contain exactly the same literals: removing repetitions of clauses does not affect the satisfiability of I. We can verify this condition in time $O(m \log m)$ by sorting lexicographically the set of clauses.

3. Every variable appears in its positive and negative forms in the set of clauses: if a variable appeared only, say, in its positive form, we could set it to true and remove all clauses in which it appeared, without affecting the satisfiability of I. We can verify this condition in time $O(n + m) = O(m)$ by checking each clause and each variable.

It could be necessary to perform the above verifications more than once, but as each time they fail they decrease the number of clauses or the number of variables, the number of verifications is $O(n+m) = O(m)$. Hence the total time spent in this process is $O(m^2 \log m)$, and therefore polynomial in $size(I)$. For each $1 \leq i \leq n$ we define p_i as the number of times the literal x_i appears in the clauses, q_i as the number of times the literal $\neg x_i$ appears in the clauses, and $t_i = \max\{p_i, q_i\}$. Under the above assumptions m and n satisfy the inequalities $n \leq \frac{3}{2}m$ and $m \leq (2n)^3$, and, for each $1 \leq i \leq n$, $p_i \geq 1$ and $q_i \geq 1$.

3.2.2 Unit Costs

We consider first the special case of MINIMUM MIXED POSTMAN TOUR in which the cost of each edge or arc is equal to one. The proof of the $\mathcal{NP}$-completeness result is based on three lemmas describing the behaviour of some mixed subgraphs.

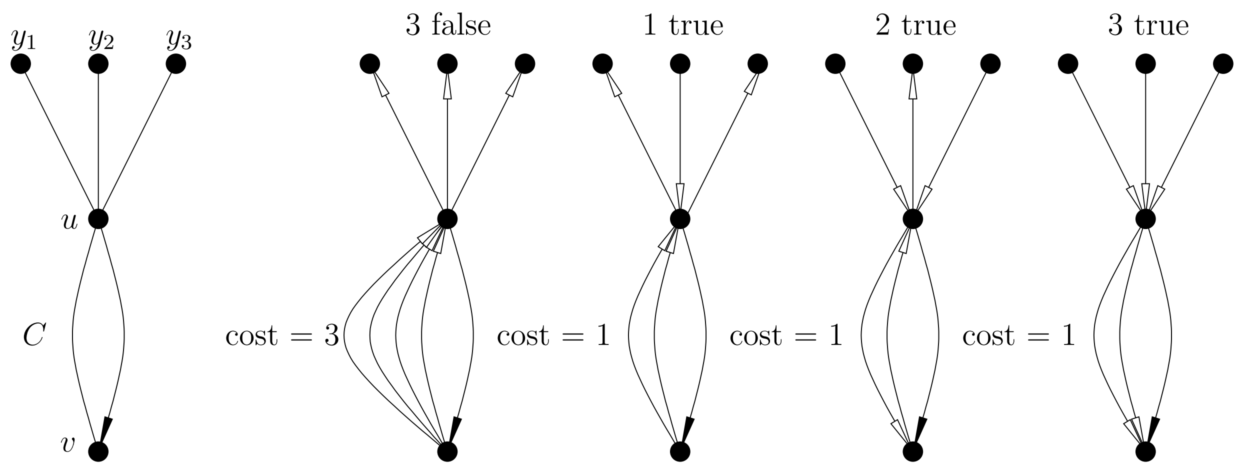

Figure 3.2: The clause subgraph and its optimal postman tours.

Lemma 3.1 *Consider the* clause *subgraph C in Figure 3.2 and assume it is induced by vertices u, v, y_1, y_2, y_3 on a mixed graph M where the shortest path between any two of v, y_1, y_2, y_3 has cost at least* 2*. Assume further that we specify the directions in which we traverse edges uy_1, uy_2, uy_3. Then the minimum cost of replicating edges and arcs in C to find a postman tour in M is* 3 *if the three edges were directed away of u, and* 1 *otherwise.*

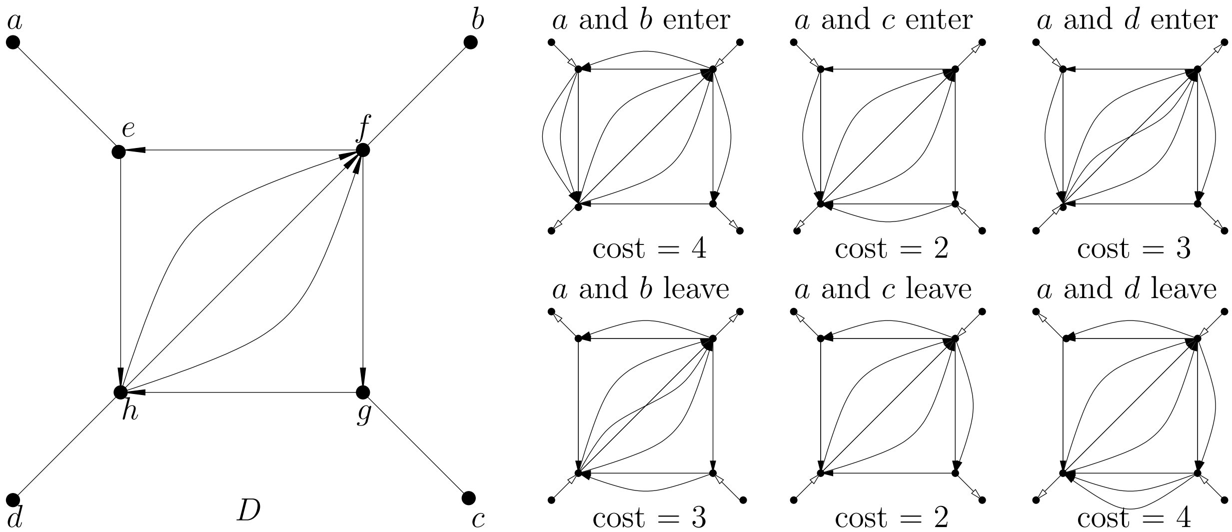

Figure 3.3: The comparator subgraph and its optimal postman tours.

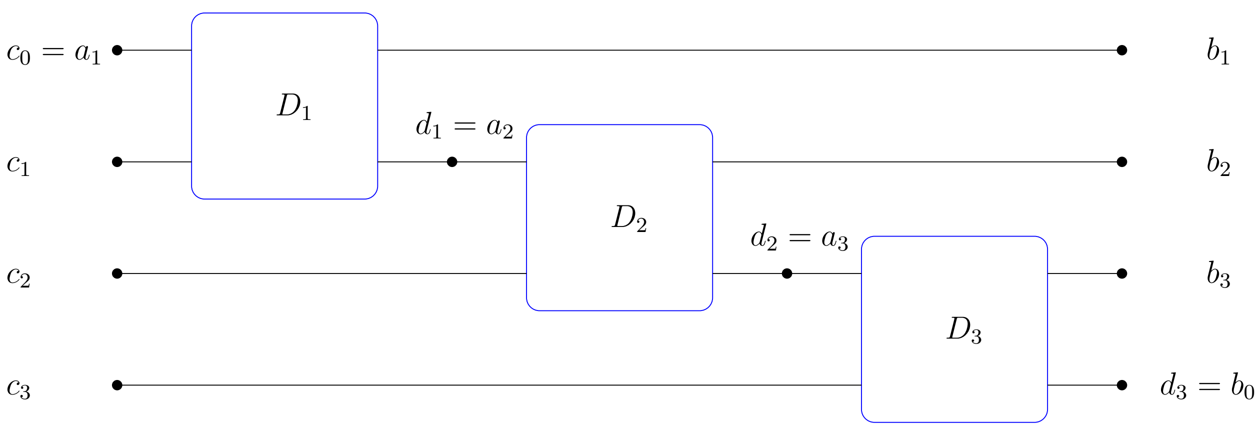

Figure 3.4: The 4-rail subgraph.

Proof. If the three edges uy_1, uy_2, uy_3 are directed away of u, then we have four arcs leaving u, so we need four arcs entering u. This can be done by replicating three times edge uv, and directing the four copies towards u. The other three cases are handled similarly. See Figure 3.2 for a case by case analysis. ■

Lemma 3.2 *Consider the* comparator *subgraph D in Figure 3.3 and assume it is induced by vertices a, b, c, d, e, f, g, h on a mixed graph M where the shortest path between any two of a, b, c, d has cost at least 2. Assume further that we specify the directions in which we traverse edges ae, bf, cg, dh (two entering D and two leaving D). Then the minimum cost of replicating edges and arcs in D to find a postman tour in M is 2 if both ae and cg have the same orientation, and at least 3 otherwise.*

Proof. Note that if we replicate any of the edges ae, bf, cg, dh then we need to replicate at least two, therefore incurring a cost of at least 3, if using a path in D two connect those edges, or at least 4, if using a path partially outside D. Hence we can assume that we only replicate arcs in D. See Figure 3.3 for a case by case analysis. ■

Given an integer $k \geq 1$ we define a $(k+1)$-*rail* mixed graph as follows: Take k copies $D_1, \ldots, D_k$ of the comparator graph, with endpoints a_i, b_i, c_i, d_i for $i \leq i \leq k$. For $1 \leq i < k$ identify vertices d_i and a_{i+1}. Rename vertex a_1 as c_0, and vertex d_k as b_0. See Figure 3.4 for an example of a 4-rail. A 1-*rail* is a path of length 2 with extreme points c_0 and b_0.

Lemma 3.3 *Let $k \in \mathbb{Z}_+$, and assume that a $(k+1)$-rail R is an induced subgraph (with endpoints $b_0, b_1, \ldots, b_k$ and $c_0, c_1, \ldots, c_k$) of a mixed graph M satisfying that the shortest path between any two of the endpoints of R has cost at least 2. Then the minimum cost of replicating edges and arcs in R to find a postman tour in M is $2k$ if we enter R through $c_0, c_1, \ldots, c_k$ and leave R through $b_0, b_1, \ldots, b_k$ or vice versa, and greater otherwise.*

Proof. R contains k induced copies of the comparator. Moreover, the length of the shortest path between any two extreme points of any copy of the comparator is at least 2. By Lemma 3.2 the minimum cost of replicating edges and arcs in R to find a postman tour in M is at least $2k$. This cost can be achieved if we enter R through $c_0, c_1, \ldots, c_k$ and leave R through $b_0, b_1, \ldots, b_k$ or conversely. If we fail to do this then the cost in one copy of the comparator will be at least 3. Hence the total cost will exceed $2k$. ■

Theorem 3.4 (Papadimitriou) *The decision version of* MINIMUM MIXED POSTMAN TOUR *with unit costs is* $\mathcal{NP}$*-complete.*

Proof. We are going to reduce 3-SATISFIABILITY to the decision version of MINIMUM MIXED POSTMAN TOUR with unit costs. Let I be any instance of 3-SATISFIABILITY that satisfies the assumptions on Section 3.2.1. Construct a mixed graph M as follows:

Take m copies of the clause graph, with common vertex v and vertices u_i, y_1^i, y_2^i, y_3^i for $1 \leq i \leq m$. For each $1 \leq j \leq n$, let R_j be a t_j-rail graph with endpoints $b_0^j, b_1^j, \ldots, b_{t_j-1}^j$ and $c_0^j, c_1^j, \ldots, c_{t_j-1}^j$. For each $1 \leq j \leq n$, and for each $1 \leq l \leq p_j$, identify c_{l-1}^j with y_z^i, where the l-th ocurrence of literal x_j is in position z of clause C_i. Similarly, for each $1 \leq j \leq n$, and for each $1 \leq l \leq q_j$, identify b_{l-1}^j with y_z^i, where the l-th ocurrence of literal $\neg x_j$ is in position z of clause C_i. Identify all the remaining b_l^j and c_l^j vertices with vertex v. Figure 3.5 shows the mixed graph obtained from the set of clauses $C_1 = \{x_1, x_2, x_3\}$, $C_2 = \{\neg x_1, \neg x_2, \neg x_3\}$, $C_3 = \{x_1, x_3, \neg x_4\}$, $C_4 = \{x_1, x_3, x_4\}$, and $C_5 = \{x_2, \neg x_3, \neg x_4\}$.

Let $w = m + 2\sum_{j=1}^{n}(t_j - 1)$. We claim that M has a postman set of cost at most w if and only if I is satisfiable. Assume that I is satisfiable by an assignment of truth values to variables $x_1, \ldots, x_n$. Then we can construct a postman tour of M as follows: For every $1 \leq j \leq n$, if x_j is true orient the edges in R_j entering through $b_0^j, b_1^j, \ldots, b_{t_j-1}^j$ and leaving through $c_0^j, c_1^j, \ldots, c_{t_j-1}^j$, otherwise orient the edges in R_j entering through $c_0^j, c_1^j, \ldots, c_{t_j-1}^j$ and leaving through $b_0^j, b_1^j, \ldots, b_{t_j-1}^j$. For each $1 \leq i \leq m$ direct edges y_z^i according to the orientations above. Figure 3.6 shows the orientation of M corresponding to the truth assignment $x_1 = x_2 = true$, $x_3 = x_4 = false$. The fact that the assignment satisfies I implies that at least one of the edges y_z^i will enter vertex u_i. By Lemmas 3.1 and 3.3, the above orientations imply there exists a postman set of M with cost w.

Now assume that there exists a postman set of M with cost w. By Lemmas 3.1 and 3.3 a cost of w cannot be avoided, since we have m copies of the clause subgraph, and n rail subgraphs, each with associated cost $2(t_j - 1)$. Moreover, cost w can only be achieved if at least one edge enters each clause subgraph, and all edges in each rail subgraph are properly oriented. This orientation defines an assignment of truth values that satisfies I.

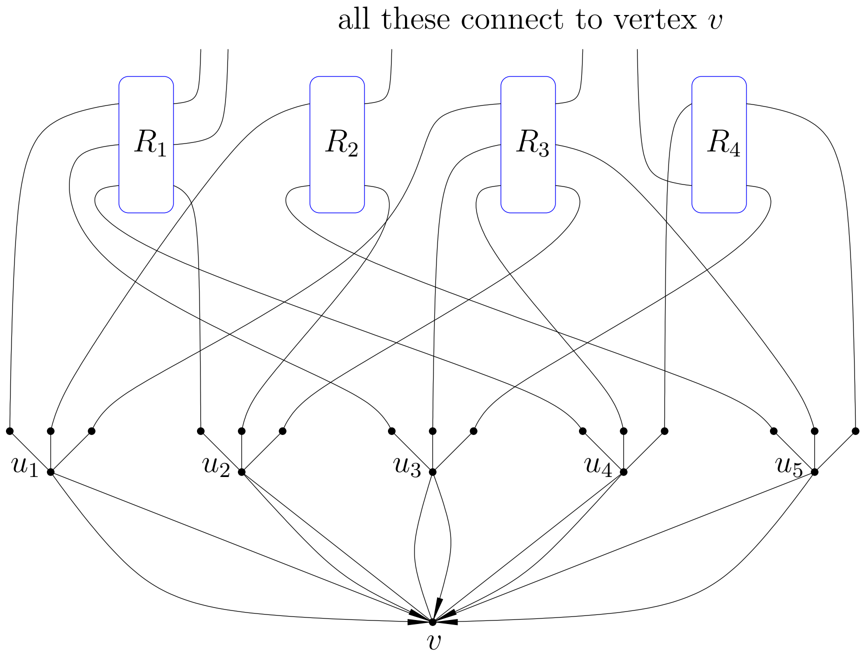

Figure 3.5: The mixed graph M in the proof of Theorem 3.4.

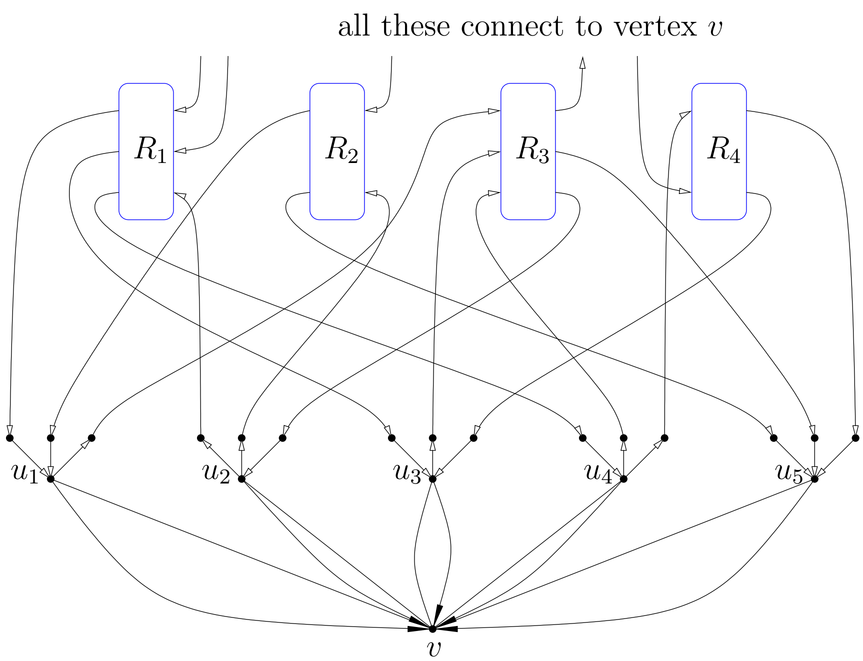

Figure 3.6: The orientation of M corresponding to $x_1 = x_2 = true$, $x_3 = x_4 = false$.

It only remains to show that M has size polynomial in the size of I. This is true since it has less than $2m + 8\sum_{j=1}^{n}(t_j - 1)$ vertices, and $5m + 11\sum_{j=1}^{n}(t_j - 1)$ edges and arcs. These two quantities are $O(m)$. ■

The mixed graph M constructed in the previous proof is definitely not planar (even if we restrict I to be an instance of PLANAR 3-SATISFIABILITY), and its maximum vertex total degree is greater than 3. However, by using techniques similar to those used by Papadimitriou, it is possible to modify our proof to obtain the stronger result in [70]. Moreover, we can use our proof to obtain a *weak* negative result regarding the approximability of MINIMUM MIXED POSTMAN SET. We give a stronger result in Chapter 6.

Theorem 3.5 *There is no fully polynomial approximation scheme for* MINIMUM MIXED POSTMAN SET, *even if all costs are* 1, *unless* $\mathcal{P} = \mathcal{NP}$.

Proof. Assume there exists a fully polynomial approximation scheme $\mathcal{A}$ such that, for all $\epsilon > 0$, $\mathcal{A}_\epsilon$ approximates MINIMUM MIXED POSTMAN SET within a factor of $(1+\epsilon)$. Recall that $\mathcal{A}_\epsilon$ must run in time polynomial in both $|V| + |E| + |A|$ and $\frac{1}{\epsilon}$. Let I be an instance of 3-SATISFIABILITY that satisfies the assumptions in Section 3.2.1, and let M be the mixed graph constructed in the proof of Theorem 3.4. Recall that I is satisfiable if and only if M has a postman set with cost at most w. Moreover, since

$$\sum_{j=1}^{n}(t_j - 1) < \sum_{j=1}^{n} t_j < \sum_{j=1}^{n}(p_j + q_j) = 3m \tag{3.2}$$

it follows that $m \le w < 7m$, that $2m \le |V| < 26m$, and that $5m \le |E| + |A| < 38m$. Choose $\epsilon = \frac{1}{7m}$ and let A be the algorithm $\mathcal{A}_\epsilon$. Note that A runs in time polynomial in $m = O(|V| + |E| + |A|)$ and satisfies:

1. If I is not satisfiable then $A(I) \ge OPT(I) \ge w + 1$.

2. If I is satisfiable then $A(I) \le (1+\epsilon)OPT(I) \le (1+\epsilon)w < w + 1$.

Therefore I is satisfiable if and only if $A(I) < w+1$, and we can decide 3-SATISFIABILITY in polynomial time. This is a contradiction, unless $\mathcal{P} = \mathcal{NP}$. ■

3.2.3 Metric Costs

We say that the vector of costs $c \in \mathbb{Q}_+^{E \cup A}$ is a *metric* if it satisfies the *triangle inequality*, that is, if for every edge e with ends u and v and for every mixed path P from u to v we have that $c_e \leq c(P)$, and for every arc a with tail u and head v and for every mixed path P from u to v we have that $c_a \leq c(P)$. Note that the vector of unit costs is a metric. It follows that the decision version of MINIMUM MIXED POSTMAN TOUR remains $\mathcal{NP}$-complete even if we restrict the costs to be metric, and there is no fully polynomial approximation scheme for MINIMUM MIXED POSTMAN SET, even if costs are metric, unless $\mathcal{P} = \mathcal{NP}$.

In the remainder of this section, we consider mixed graphs whose vertices are points in $\mathbb{Z}^2$. Given two points (p, q) and (r, s) in $\mathbb{Z}^2$, the *discrete Euclidean* distance between them is $\lfloor ((p-r)^2 + (q-s)^2)^{1/2} \rfloor$, and the *Manhattan* distance between them is $|p-r| + |q-s|$. It can be seen that each one of these two distances defines a metric on $\mathbb{Z}^2$. The vertices of an instance of MINIMUM MIXED POSTMAN TOUR with discrete Euclidean or Manhattan costs are described by their integer coordinates. As is usual when considering graphs whose vertices are points in $\mathbb{Z}^2$, we allow superimposed vertices.

Theorem 3.6 *The decision version of* MINIMUM MIXED POSTMAN TOUR *remains* $\mathcal{NP}$-*complete even if we restrict the costs to be Manhattan.*

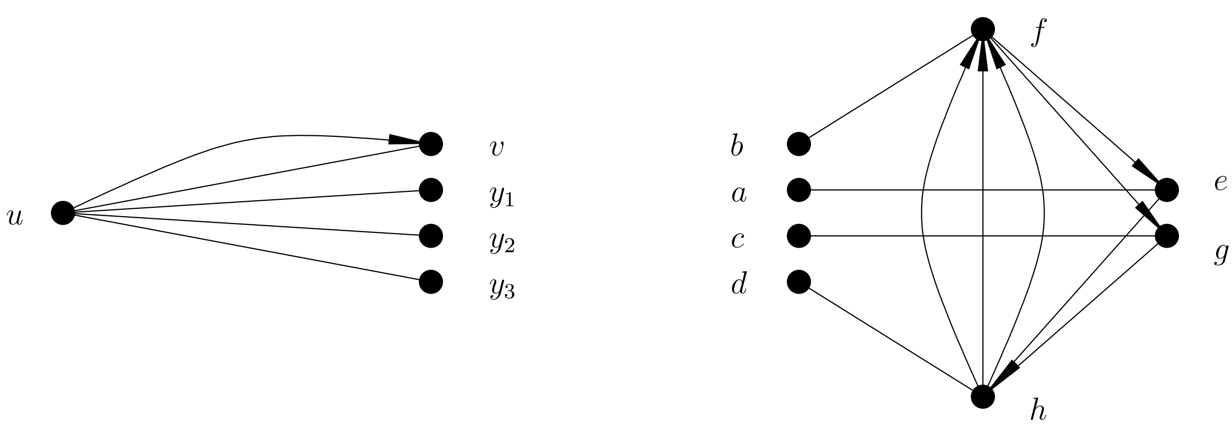

Figure 3.7: The clause and comparator subgraphs with Manhattan costs.

Proof. It is enough to show that we can construct a clause subgraph and a comparator subgraph on $\mathbb{Z}^2$ with Manhattan costs. We show those subgraphs in Figure 3.7. The coordinates for the points are as follows: $u = (0, 0)$, $v = y_1 = y_2 = y_3 = (2, 0)$, $a = b = c = d = (2, 0)$, $e = g = (4, 0)$, $f = (3, 1)$, and $h = (3, -1)$. Note that all costs have been doubled with respect to the costs of the *original* subgraphs. ■

Corollary 3.7 *The decision version of* MINIMUM MIXED POSTMAN TOUR *remains* $\mathcal{NP}$-*complete even if we restrict the costs to be discrete Euclidean.*

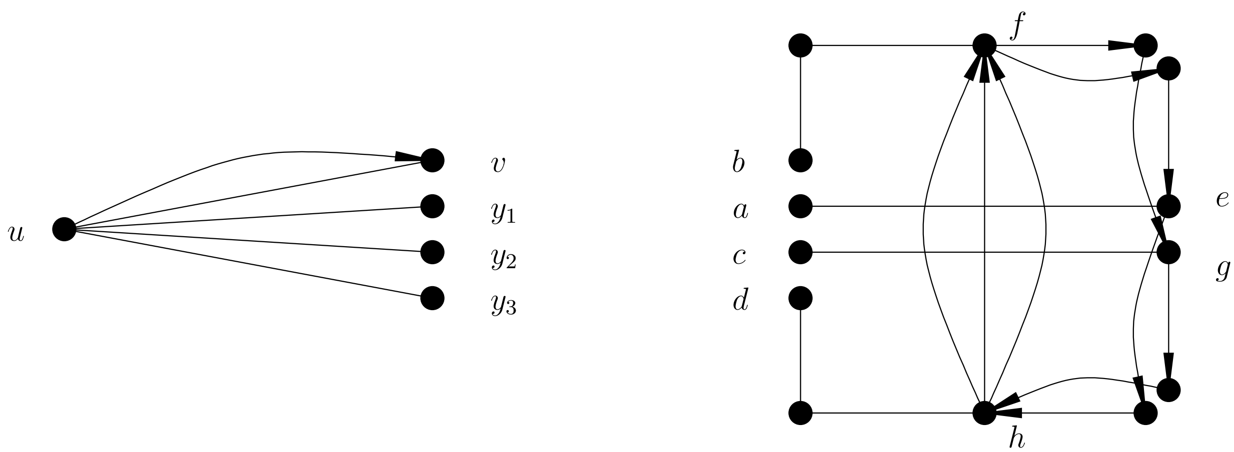

Figure 3.8: The clause and comparator subgraphs with discrete Euclidean costs.

Proof. A direct reduction from MINIMUM MIXED POSTMAN TOUR with Manhattan costs works: Replace every edge between points (a, b) and (c, d) with two edges, one between points (a, b) and (a, d), and the other between points (a, d) and (c, d). Similarly, replace every arc from (a, b) to (c, d) with two arcs, one from (a, b) to (a, d), and the other from (a, d) to (c, d). In Figure 3.8, we have applied this procedure to the subgraphs in Figure 3.7. The number of edges and arcs of the new mixed graph is at most doubled. The discrete Euclidean cost is now precisely the Manhattan cost. ■

3.3 Integer Programming Formulations

Let $M = (V, E, A)$ be a strongly connected mixed graph, and let $c \in \mathbb{Q}_+^{E \cup A}$. We present two different integer programming formulations of MINIMUM MIXED POSTMAN TOUR.

3.3.1 First Formulation

The first integer programming formulation we give is due to Kappauf and Koehler [57], and Christofides et al. [13]. It is based on Veblen's characterization of mixed Eulerian graphs and a flow formulation similar to the one for the directed case. Similar formulations were given by Edmonds and Johnson [29], Grötschel and Win [47], and Ralphs [78].

Let $\vec{M} = (V, A \cup E^+ \cup E^-)$ be the associated directed graph of M, and let $B = E^+ \cup E^-$. For every $e \in E$, let $c_{e^+} = c_{e^-} = c_e$. A nonnegative integer circulation x of $\vec{M}$ is the incidence vector of a postman tour of M if and only if $x_e \geq 1$ for all $e \in A$, and $x_{e^+} + x_{e^-} \geq 1$ for all $e \in E$. Therefore, we obtain the integer program:

$$
\begin{aligned}
\text{MMPT1}(M, c) &= \min c_A^\top x_A + c_E^\top x_E^+ + c_E^\top x_E^- && (3.3)\\
\text{subject to} & && \\
x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) &= 0 \text{ for all } v \in V && (3.4)\\
x_a &\geq 1 \text{ for all } a \in A && (3.5)\\
x_{e^+} + x_{e^-} &\geq 1 \text{ for all } e \in E && (3.6)\\
x_a &\geq 0 \text{ and integer for all } a \in A \cup E^+ \cup E^-. && (3.7)
\end{aligned}
$$

Let $\mathcal{P}^1_{MPT}(M)$ be the convex hull of the feasible solutions to the integer program above, and let $\mathcal{Q}^1_{MPT}(M)$ be the set of feasible solutions to its linear programming relaxation:

$$
\begin{aligned}
\text{LMMPT1}(M, c) &= \min c_A^\top x_A + c_E^\top x_E^+ + c_E^\top x_E^- && (3.8)\\
\text{subject to} & && \\
x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) &= 0 \text{ for all } v \in V && (3.9)\\
x_a &\geq 1 \text{ for all } a \in A && (3.10)\\
x_{e^+} + x_{e^-} &\geq 1 \text{ for all } e \in E && (3.11)\\
x_a &\geq 0 \text{ for all } a \in A \cup E^+ \cup E^-. && (3.12)
\end{aligned}
$$

3.3.2 Second Formulation

The second integer programming formulation we give is due to Nobert and Picard [68]. The approach they use is based on Ford and Fulkerson's characterization of mixed Eulerian graphs. The vector $x \in \mathbb{Z}_+^{E \cup A}$ is the incidence vector of a postman tour of G if and only if $x_e \geq 1$ for all $e \in E \cup A$, $x(\delta_{E\cup A}(v))$ is even for all $v \in V$, and $x(\delta_A(\bar{S})) + x(\delta_E(S)) \geq x(\delta_A(S))$ for all $S \subseteq V$. Therefore we obtain the integer program:

$$\begin{aligned} \text{MMPT2}(M,c) &= \min c^\top x && (3.13)\\ \text{subject to} & && \\ x(\delta_{E\cup A}(v)) &\equiv 0 \text{ (mod 2) for all } v \in V && (3.14)\\ x(\delta_A(\bar{S})) + x(\delta_E(S)) - x(\delta_A(S)) &\geq 0 \text{ for all } S \subseteq V && (3.15)\\ x_e &\geq 1 \text{ and integer for all } e \in E \cup A. && (3.16) \end{aligned}$$

Note that the parity constraints (3.14) are not in the required form for INTEGER PROGRAMMING MINIMIZATION; however, this can be easily solved by noting that, for all $v \in V$,

$$x(\delta_{E\cup A}(v)) \equiv x(\delta_A(\bar{v})) + x(\delta_E(v)) - x(\delta_A(v)) \text{ (mod 2)}, \tag{3.17}$$

and introducing a *slack* variable $s_v \in \mathbb{Z}_+$ to obtain the equivalent constraint

$$x(\delta_A(\bar{v})) + x(\delta_E(v)) - x(\delta_A(v)) - 2s_v = 0 \text{ for all } v \in V. \tag{3.18}$$

Let $\mathcal{P}^2_{MPT}(M)$ be the convex hull of the feasible solutions to the integer program above, and let $\mathcal{Q}^2_{MPT}(M)$ be the set of feasible solutions to its linear programming relaxation:

$$\begin{aligned} \text{LMMPT2}(M,c) &= \min c^\top x && (3.19)\\ \text{subject to} & && \\ x(\delta_A(\bar{S})) + x(\delta_E(S)) - x(\delta_A(S)) &\geq 0 \text{ for all } S \subseteq V && (3.20)\\ x_e &\geq 1 \text{ for all } e \in E \cup A. && (3.21) \end{aligned}$$

Note that the constraints (3.14) were relaxed to $x(\delta_{E\cup A}(v)) \geq 0$ for all $v \in V$, but these constraints are redundant in the linear program LMMPT2(M,c). We reach the same conclusion if we use the formulation with slacks and we discard them.

3.4 Linear Programming Relaxations

In the previous section we gave two integer programming formulations for Minimum Mixed Postman Tour, as well as their linear relaxations. One of the first questions we might ask is whether one of the relaxations is better than the other or they are in fact equivalent. Surprisingly, we were not able to find a single reference to this question in our literature search. We answer this question by showing in two rather different ways that the relaxations are equivalent. With this result in hand, we study some of the properties of the extreme points of the set $\mathcal{Q}^1_{MPT}(M)$ of solutions to our first formulation.

3.4.1 Equivalence

We give two proofs that LMMPT1(M, c) and LMMPT2(M, c) are essentially equivalent. Our first result says that solving both linear programs would give the same objective value.

Theorem 3.8 *For every $x^1 \in \mathcal{Q}^1_{MPT}(M)$ there exists $x^2 \in \mathcal{Q}^2_{MPT}(M)$ such that $c^\top x^1 = c^\top x^2$, and conversely, for every $x^2 \in \mathcal{Q}^2_{MPT}(M)$ there exists $x^1 \in \mathcal{Q}^1_{MPT}(M)$ such that $c^\top x^1 = c^\top x^2$. Moreover, in both cases we can ensure that $x^1_a = x^2_a$ for all $a \in A$, and $x^1_{e^+} + x^1_{e^+} = x^2_e$ for all $e \in E$.*

Proof: First note that $x^1_e = x^2_e$ for all $e \in A$, and $x^1_{e^+} + x^1_{e^-} = x^2_e$ for all $e \in E$ imply $c^\top x^1 = c^\top x^2$ for every vector of costs c. ($\Rightarrow$) Let $x^1 \in \mathcal{Q}^1_{MPT}(M)$ and define x^2 as above. It is clear that $x^2 \in \mathbb{R}^{E \cup A}_+$, so we only have to prove (3.20). Let $S \subseteq V$, then

$$
\begin{aligned}
0 &\leq 2x^1(\vec{\delta}_B(S)) && (3.22)\\
&= \sum_{v \in S} \Big(x^1(\vec{\delta}(\bar{v})) - x^1(\vec{\delta}(v)) \Big) + 2x^1(\vec{\delta}_B(S)) && (3.23)\\
&= x^1(\delta_A(\bar{S})) + x^1(\vec{\delta}_B(S)) + x^1(\vec{\delta}_B(\bar{S})) - x^1(\delta_A(S)) && (3.24)\\
&= x^2(\delta_A(\bar{S})) + x^2(\delta_E(S)) - x^2(\delta_A(S)). && (3.25)
\end{aligned}
$$

($\Leftarrow$) Let $x^2 \in \mathcal{Q}^2_{MPT}(M)$ and assume x^2 is rational. Let N be a positive integer such that each component of $x = Nx^2$ is an even integer. Consider the graph M^N that contains

x_e copies of each $e \in E \cup A$. Note that M^N is Eulerian, and $x_e \geq N$ for all $e \in E \cup A$. Hence we can direct some of the copies of $e \in E$ in one direction and the rest in the other (say x_{e^+} and x_{e^-}, respectively) to obtain an Eulerian tour of M^N. Therefore, $x \in \mathcal{Q}^1_{MPT}(M^N)$, $x_e \geq N$ for all $e \in A$, and $x_{e^+} + x_{e^-} \geq N$ for all $e \in E$, and hence $x^1 = \frac{1}{N} x \in \mathcal{Q}^1_{MPT}(M)$. Note that x^1 satisfies the properties in the statement. ■

Theorem 3.8 implies that, for every vector c, LMMPT1$(M, c) =$ LMMPT2(M, c), that is, it is equivalent to optimize over either polyhedron. Our second result goes a bit further: we show that $\mathcal{Q}^2_{MPT}(M)$ is essentially a projection of $\mathcal{Q}^1_{MPT}(M)$. Let $\mathcal{A}$ be the incidence matrix of the directed graph $D = (V, A)$, and let $\mathcal{D}$ be the incidence matrix of the directed graph $D^+ = (V, E^+)$. Let $\mathcal{Q}^3_{MPT}(M)$ be the set of solutions $x \in \mathbb{R}^{A \cup E \cup E^+ \cup E^-}$ of the system:

$$\begin{aligned} \mathcal{A} x_A + \mathcal{D}(x_{E^+} - x_{E^-}) &= \mathbf{0}_V && (3.26) \\ x_E - x_{E^+} - x_{E^-} &= \mathbf{0}_E && (3.27) \\ x_A &\geq \mathbf{1}_A && (3.28) \\ x_E &\geq \mathbf{1}_E && (3.29) \\ x_{E^+} &\geq \mathbf{0}_E && (3.30) \\ x_{E^-} &\geq \mathbf{0}_E && (3.31) \end{aligned}$$

Note that this system is a reformulation of (3.9)–(3.12) where all the constraints have been written in vector form, and we have included an additional variable x_e for each edge e. The following is a consequence of Theorem 3.8, but we give a different proof.

Theorem 3.9 *The projection of the polyhedron $\mathcal{Q}^3_{MPT}(M)$ onto $x_{E^+} = \mathbf{0}_E$ and $x_{E^-} = \mathbf{0}_E$ is $\mathcal{Q}^2_{MPT}(M)$.*

Proof: Let Q be the projection of $\mathcal{Q}^3_{MPT}(M)$ onto $x_{E^+} = \mathbf{0}_E$ and $x_{E^-} = \mathbf{0}_E$, that is

$$Q = \{x \in \mathbb{R}^{A \cup E} : (\mathcal{A}^\top z_V + z_A)^\top x_A + (z_B + z_E)^\top x_E \geq z_A^\top \mathbf{1}_A + z_E^\top \mathbf{1}_E \ \forall\, z \in R\}, \quad (3.32)$$

where

$$R = \{(z_V, z_B, z_A, z_E) \in \mathbb{R}^{V \cup E^+ \cup A \cup E} : z_A \geq \mathbf{0}_A, z_E \geq \mathbf{0}_E \text{ and } z_B \geq |\mathcal{D}^\top z_V|\}. \quad (3.33)$$

We verify first that (3.20) and (3.21) are valid inequalities for Q:

(3.20) Let $S \subseteq V$, and consider the element of R given by $z_V = \chi^S$, $z_B = \chi^{\delta_E(S)}$, $z_A = \mathbf{0}_A$, and $z_E = \mathbf{0}_E$. This implies the constraint $(\chi^S)^\top \mathcal{A} x_A + (\chi^{\delta_E(S)})^\top x_E \geq 0$, that is, $x(\delta_E(S)) + x(\delta_A(\bar{S})) - x(\delta_A(S)) \geq 0$.

(3.21) Let $a \in A$, and consider the element of R given by $z_V = \mathbf{0}_V$, $z_B = \mathbf{0}_E$, $z_A = \mathbf{1}_a$, and $z_E = \mathbf{0}_E$. This implies the constraint $\mathbf{1}_a^\top x_A \geq \mathbf{1}_a^\top \mathbf{1}_A$, that is, $x_a \geq 1$. Let $e \in E$, and consider the element of R given by $z_V = \mathbf{0}_V$, $z_B = \mathbf{0}_E$, $z_A = \mathbf{0}_A$, and $z_E = \mathbf{1}_e$. This implies the constraint $\mathbf{1}_e^\top x_E \geq \mathbf{1}_e^\top \mathbf{1}_E$, that is, $x_e \geq 1$.

Now we verify that every element of R can be written as a nonnegative linear combination of the following elements of R:

(S1) For $S \subseteq V$, let $z_V = \chi^S$, $z_B = \chi^{\delta_E(S)}$, $z_A = \mathbf{0}_A$, and $z_E = \mathbf{0}_E$.

(S2) For $S \subseteq V$, let $z_V = -\chi^S$, $z_B = \chi^{\delta_E(S)}$, $z_A = \mathbf{0}_A$, and $z_E = \mathbf{0}_E$.

(A) For $a \in A$, let $z_V = \mathbf{0}_V$, $z_B = \mathbf{0}_E$, $z_A = \mathbf{1}_a$, and $z_E = \mathbf{0}_E$.

(E1) For $e \in E$, let $z_V = \mathbf{0}_V$, $z_B = \mathbf{0}_E$, $z_A = \mathbf{0}_A$, and $z_E = \mathbf{1}_e$.

(E2) For $e \in E$, let $z_V = \mathbf{0}_V$, $z_B = \mathbf{1}_e$, $z_A = \mathbf{0}_A$, and $z_E = \mathbf{0}_E$.

If any component of z_A or z_E is positive, we can use **(A)** or **(E1)** to reduce it to zero, so we only consider the set of solutions of $z_B \geq |\mathcal{D}^\top z_V|$ with z_B and z_V free. Let $S_+ = \operatorname{supp}_+(z_V)$, and let $S_- = \operatorname{supp}_-(z_V)$. If both S_+ and S_- are empty, then we can reduce the components of z_B using **(E2)**. Otherwise, assume that S_+ is nonempty and that the minimal positive component of z_V is 1. For every edge $e \in \delta_E(S_+)$ with endpoints $u \in S_+$, $v \notin S_+$ we have

$$(z_B)_e \geq |(\mathcal{D}^\top z_V)_e| = |(z_V)_u - (z_V)_v| \geq |(z_V)_u| = (z_V)_u \geq 1. \tag{3.34}$$

Therefore, the vectors

$$z_B^* \equiv z_B - \chi^{\delta_E(S_+)} \text{ and } z_V^* \equiv z_V - \chi^{S_+} \tag{3.35}$$

satisfy $z_B^* \geq |\mathcal{D}^\top z_V^*|$ and have fewer nonzero components. So we can reduce (z_B, z_V) using **(S1)**. Similarly, if S_- is nonempty, we can reduce (z_B, z_V) using **(S2)**. ∎

3.4.2 Half-Integrality

Now we are going to explore the structure of the extreme points of $\mathcal{Q}^1_{MPT}(M)$. To start, we offer a simple proof of the following result due independently to Kappauf and Koehler [57], Ralphs [78], and Win [90]. We say that $e \in E$ is *tight* if $x_{e^+} + x_{e^-} = 1$.

Theorem 3.10 (Kappauf and Koehler, Ralphs, Win) *Every extreme point x of the polyhedron $\mathcal{Q}^1_{MPT}(M)$ has components whose values are either $\frac{1}{2}$ or a nonnegative integer. Moreover, fractional components occur only on tight edges.*

Proof: Let x be an extreme point of $\mathcal{Q}^1_{MPT}(M)$. We say that $a \in A$ is *fractional* if x_a is not an integer. Similarly, we say that $e \in E$ is *fractional* if at least one of x_{e^+} or x_{e^-} is not an integer. Let $F = \{e \in E \cup A : e \text{ is fractional}\}$. We will show that $F \subseteq E$, and that each $e \in F$ is tight. Assume that for some $v \in V$, $d_F(v) = 1$. Let e be the unique element of F incident to v. Since the total flow into v is integral the only possibility is that $e \in E$. Moreover, both x_{e^+} and x_{e^-} must be fractional. If e is not tight, the vectors x^1 and x^2 obtained from x replacing the entries in e^+ and e^- by

$$\begin{array}{rclcrcl} x^1_{e^+} & = & x_{e^+} + \epsilon & \quad & x^1_{e^-} & = & x_{e^-} + \epsilon \\ x^2_{e^+} & = & x_{e^+} - \epsilon & \quad & x^2_{e^-} & = & x_{e^-} - \epsilon \end{array} \tag{3.36}$$

(where $\epsilon = \min\{x_{e^+}, x_{e^-}, 2(x_{e^+} + x_{e^-} - 1)\} > 0$) would be feasible, with $x = \frac{1}{2}(x^1 + x^2)$, contradicting the choice of x. Hence e is a tight edge, and satisfies $x_{e^+} = x_{e^-} = \frac{1}{2}$. Delete e from F and repeat the above argument until F is empty or F induces an undirected graph with minimum degree 2. (Deletion of e does not alter the argument since it contributes 0 flow into both its ends.) Suppose F contains a cycle C. Assign an arbitrary orientation (say, positive) to C. We say that an arc in C is *forward* if it has the same orientation as C, and we call it *backward* otherwise. Partition C as follows:

$$C_A^+ = \{e \in C \cap A : e \text{ is forward}\} \tag{3.37}$$

$$C_A^- = \{e \in C \cap A : e \text{ is backward}\} \tag{3.38}$$

$$C_E^= = \{e \in C \cap E : e \text{ is tight}\} \tag{3.39}$$

$$C_E^> = \{e \in C \cap E : e \text{ is not tight}\} \tag{3.40}$$

and define

$$\begin{aligned}
\epsilon^+ &= \min_{e \in C_A^+} \lceil x_e \rceil - x_e & (3.41)\\
\epsilon^- &= \min_{e \in C_A^-} x_e - \lfloor x_e \rfloor & (3.42)\\
\epsilon^= &= \min_{e \in C_E^=} \{x_{e^+}, x_{e^-}\} & (3.43)\\
\epsilon^> &= \min_{e \in C_E^>} \{\lceil x_e \rceil - x_e, x_e - \lfloor x_e \rfloor\} & (3.44)\\
\epsilon^1 &= \min\{\epsilon^+, \epsilon^-, 2\epsilon^=, \epsilon^>\} & (3.45)
\end{aligned}$$

The choice of C implies $\epsilon^1 > 0$. Now we define a new vector x^1 as follows:

$$x_e^1 = \begin{cases} x_e + \epsilon^1 & \text{if } e \in C_A^+ \text{ or } e \text{ is forward in } C_E^> \\ x_e - \epsilon^1 & \text{if } e \in C_A^- \text{ or } e \text{ is backward in } C_E^> \\ x_e + \frac{1}{2}\epsilon^1 & \text{if } e \text{ is the forward copy of an edge in } C_E^= \\ x_e - \frac{1}{2}\epsilon^1 & \text{if } e \text{ is the backward copy of an edge in } C_E^= \\ x_e & \text{otherwise} \end{cases} \quad (3.46)$$

This is equivalent to pushing ϵ^1 units of flow in the positive direction of C, and therefore it is easy to verify that $x^1 \in \mathcal{Q}^1_{MPT}(M)$. Similarly, define ϵ^2 and a vector x^2 using the other (negative) orientation of C. But now x is a convex combination of x^1 and x^2 (in fact, by choosing $\epsilon = \min\{\epsilon^1, \epsilon^2\}$ and pushing ϵ units of flow in both directions we would have $x = \frac{1}{2}(x^1 + x^2)$) contradicting the choice of x. Therefore F is empty. ■

Note that F above is a forest [57]. A similar idea allows us to prove a sufficient condition, due to Edmonds and Johnson [29], for $\mathcal{Q}^1_{MPT}(M)$ to be integral. Recall that a mixed graph $M = (V, E, A)$ is *even* if the total degree $d_{E \cup A}(v)$ is even for every $v \in V$.

Theorem 3.11 (Edmonds and Johnson) *If M is even, then $\mathcal{Q}^1_{MPT}(M)$ is integral. Therefore* MINIMUM MIXED POSTMAN TOUR *can be solved in polynomial time for the class of even mixed graphs.*

Proof. Let x be an extreme point of $\mathcal{Q}^1_{MPT}(M)$. We say that $a \in A$ is *even* if x_a is even. We say that $e \in E$ is *even* if $x_{e^+} - x_{e^-}$ is even. For a contradiction, assume x is not integral, and define F as in the proof of Theorem 3.10. Let $N = \{e \in E \cup A : e \text{ is even}\}$. Note that by Theorem 3.10, $F \subseteq N$. Hence N is not empty. We show now that $M[N]$ has minimum degree 2, and hence contains a cycle C. Let $v \in V$. If $d_F(v) \geq 2$ then certainly $d_N(v) \geq 2$. If $d_F(v) = 1$ then

$$x(\vec{\delta}(v)) - x(\vec{\delta}(\bar{v})) = \sum_{a \in \delta_A(v) \cup \delta_A(\bar{v})} \pm x_a + \sum_{e \in \delta_E(v)} \pm(x_{e^+} - x_{e^-}) \tag{3.47}$$

is the sum of an even number of integer terms (one term per arc $a \in \delta_A(v) \cup \delta_A(\bar{v})$ and one term per edge $e \in \delta_E(v)$), and one of them is equal to zero (the one in $\delta_F(v)$); therefore another term must be even. The same argument works for a vertex v not in $V(F)$, that is, $d_F(v) = 0$, with at least one element of N incident to it, that is, $d_N(v) \geq 1$.

As before, assign an arbitrary (positive) orientation to C and partition it into the classes $C_A^+, C_A^-, C_E^=, C_E^>$. Note that all $e \in C \setminus C_E^=$ satisfy $x_e \geq 2$. Hence the vector x^1 defined as

$$x_e^1 = \begin{cases} x_e + 1 & \text{if } e \in C_A^+ \text{ or } e \text{ is forward in } C_E^> \\ x_e - 1 & \text{if } e \in C_A^- \text{ or } e \text{ is backward in } C_E^> \\ x_e + \frac{1}{2} & \text{if } e \text{ is the forward copy of an edge in } C_E^= \\ x_e - \frac{1}{2} & \text{if } e \text{ is the backward copy of an edge in } C_E^= \\ x_e & \text{otherwise,} \end{cases} \tag{3.48}$$

as well as the vector x^2 obtained from the negative orientation of C, belong to $\mathcal{Q}^1_{MPT}(M)$ and satisfy $x = \frac{1}{2}(x^1 + x^2)$. This contradiction implies that F must be empty. ■

Observe that the set N defined above is the complement of a postman set of $\bar{M}$.

3.4.3 Odd-Cut Constraints

A consequence of Theorems 3.9 and 3.10 is that $\mathcal{Q}^2_{MPT}(M)$ is integral, and hence it may have integral extreme points that are not incidence vectors of postman tours of M. By adding *odd-cut constraints* similar to those for MINIMUM UNDIRECTED POSTMAN TOUR, we can cut these extraneous integral extreme points from $\mathcal{Q}^2_{MPT}(M)$, as well as some fractional extreme points from $\mathcal{Q}^1_{MPT}(M)$. Let $S \subset V$ be such that $d_M(S)$ is odd. Then, in any postman tour of M, at least one element of $\delta_M(S)$ must be duplicated. Therefore, the inequality

$$x(\delta_A(S)) + x(\delta_A(\bar{S})) + x(\delta_E(S)) \geq d_A(S) + d_A(\bar{S}) + d_E(S) + 1 \tag{3.49}$$

is valid for $\mathcal{P}^2_{MPT}(M)$, and the inequality

$$x(\vec{\delta}(S)) + x(\vec{\delta}(\bar{S})) \geq \vec{d}(S) + \vec{d}(\bar{S}) + 1 \tag{3.50}$$

is valid for $\mathcal{P}^1_{MPT}(M)$. Note that both of these inequalities can be rewritten as:

$$x(\delta_M(S)) \geq d_M(S) + 1, \tag{3.51}$$

for all S such that $d_M(S)$ is odd.

Let $\mathcal{O}^1_{MPT}(M)$ be the subset of $\mathcal{Q}^1_{MPT}(M)$ that satisfies all the odd-cut constraints (3.50), and let $\mathcal{O}^2_{MPT}(M)$ be the subset of $\mathcal{Q}^2_{MPT}(M)$ that satisfies all the odd-cut constraints (3.49). We will study the properties of $\mathcal{O}^1_{MPT}(M)$ and $\mathcal{Q}^1_{MPT}(M)$. It is worth noting that the smallest graph for which $\mathcal{Q}^1_{MPT}(M)$ is not integral is K_2, and the smallest graph for which $\mathcal{O}^1_{MPT}(M)$ is not integral is K_4.

A consequence of Theorem 2.9 is that we can decide in polynomial time whether a vector $x \in \mathbb{Q}_+^{A \cup E^+ \cup E^-}$ satisfies all the odd-cut constraints (3.50) or not. Together with Theorem 1.8, this implies the following result of Grötschel and Win [47, 90]:

Theorem 3.12 (Grötschel and Win) *There exists a polynomial-time algorithm that, given a mixed graph $M = (V, E, A)$ and a vector $c \in \mathbb{Q}_+^{A \cup E}$, finds a vector $x \in \mathbb{Q}_+^{A \cup E^+ \cup E^-}$ minimizing $c^\top x$ over $\mathcal{O}^1_{MPT}(M)$. Hence,* Minimum Mixed Postman Tour *can be solved in polynomial time for the class of mixed graphs M with $\mathcal{O}^1_{MPT}(M)$ integral.*

3.5 Approximation Algorithms

In this section we sketch three approximation algorithms for Minimum Mixed Postman Tour. When we combine the first two, we obtain a $\frac{5}{3}$-approximation algorithm for Minimum Mixed Postman Tour, and when we combine the last two, we obtain a $\frac{3}{2}$-approximation algorithm for Minimum Mixed Postman Tour. In this section, T^* is a minimum cost postman tour of $M = (V, E, A)$ with costs $c \in \mathbb{Q}_+^{E \cup A}$.

3.5.1 A $\frac{5}{3}$-Approximation Algorithm

The first algorithm we describe (called MIXED1) is due to Edmonds and Johnson [29]. Let $\bar{M} = (V, A \cup E)$ be the underlying undirected graph of M. By Theorem 2.8, we can find a minimum cost postman set J_1 of $\bar{M}$ with costs c. Let $N = (V, F, B)$ be the mixed graph obtained from M by adding one copy of each element of J_1, with the same cost as the original. By Theorem 3.11, since N is even, we can find a minimum cost postman tour T_1 of N. Note that T_1 is also a postman tour of M. Frederickson proved that this is a 2-approximation algorithm for MINIMUM MIXED POSTMAN TOUR [40].

The second algorithm we describe (called MIXED2) is another 2-approximation algorithm for MINIMUM MIXED POSTMAN TOUR, due to Frederickson [40]. Let $D' = (V, A')$ be the directed graph with $A' = A \cup B_1 \cup B_2$, where each of B_1 and B_2, just as B, contains two oppositely directed arcs (e_1^+, e_1^- and e_2^+, e_2^-, respectively) for each edge e in E. Let x^* be an extreme point solution to the minimum feasible flow problem

$$\begin{aligned} \text{INOUT}(M, c) &= \min c_A^\top x_A + c_E^\top x^+ + c_E^\top x^- && (3.52)\\ \text{subject to} & && \\ x(\delta_{A'}(\bar{v})) - x(\delta_{A'}(v)) &= 0 \text{ for all } v \in V && (3.53)\\ x_a &\geq 1 \text{ for all } a \in A && (3.54)\\ x_b &\leq 1 \text{ for all } b \in B_2 && (3.55)\\ x_b &\geq 0 \text{ for all } b \in B_1 \cup B_2, && (3.56) \end{aligned}$$

and let U be the set of edges e such that $x_{e_2^+} + x_{e_2^-}$ is even. Let $T = \{v \in V : d_U(v) \text{ is odd}\}$, and let J_2 be a minimum cost T-join of $G = (V, E)$ with costs c. Note that U and J_2 form an even undirected graph. Hence adding J_2 to x gives a postman tour T_2 of M.

Frederickson [40] gave examples (M_1, c_1) and (M_2, c_2) to show that the above guarantees for MIXED1 and MIXED2 cannot be improved. He considered the algorithm MIXED12 that runs MIXED1 and MIXED2 with the same input, and outputs the better of the two solutions obtained, and proved that algorithm MIXED12 is a $\frac{5}{3}$-approximation algorithm for MINIMUM MIXED POSTMAN TOUR.

It is not known whether the factor $\frac{5}{3}$ is best possible for MIXED12. Frederickson gave a modification of MIXED12 that has a guarantee of $\frac{3}{2}$ for planar mixed graphs [40].

3.5.2 A $\frac{3}{2}$-Approximation Algorithm

The third algorithm we describe is due to Raghavachari and Veerasamy [76, 85]. This algorithm, called MIXED3, is a slight modification of MIXED1 that uses some intermediate output from MIXED2, namely U. Define a new vector of costs $c' \in \mathbb{Q}_+^{E \cup A}$ as $c'_e = c_e$ if $e \in U$, and $c'_e = 0$ otherwise, and let J_3 be a minimum cost postman set of $\bar{M}$ with costs c'. The purpose is to avoid the duplication of edges in U as much as possible. Let $N = (V, F, B)$ be the mixed graph obtained from M by adding one copy of each element of J_3, with its original cost, and find a minimum cost postman tour T_3 of N. Consider the algorithm MIXED23 that runs MIXED2 and MIXED3 with the same input, and outputs the best of the two solutions obtained. Raghavachari and Veerasamy proved that algorithm MIXED23 is a $\frac{3}{2}$-approximation algorithm for MINIMUM MIXED POSTMAN TOUR.

Frederickson's examples also show that the factor $\frac{3}{2}$ is the best possible for MIXED23. Currently, MIXED23 is the approximation algorithm with the best approximation guarantee for MINIMUM MIXED POSTMAN TOUR.

3.6 The Bounded Mixed Postman Problem

We can generalize MINIMUM MIXED POSTMAN TOUR by providing, for each edge and arc e, two integers $u_e \geq l_e \geq 0$, and requiring that e is used at least l_e and at most u_e times. We allow $u_e = \infty$ (that is, no upper bound), but l_e must be finite. We say that a family of circuits $\mathcal{C}$ is an (l, u)-*postman tour* (or simply, a *bounded postman tour*) if, for every edge and arc e, the total number of times that e is used by the elements of $\mathcal{C}$ is at least l_e and at most u_e. Note that a bounded postman tour is not necessarily a tour simply because the edges and arcs of the elements of $\mathcal{C}$ may induce a disconnected subgraph of M. However, if the spanning subgraph of M induced by the support of l is connected, then a bounded postman tour is a tour. Consider the feasibility and minimization questions:

Problem: BOUNDED MIXED POSTMAN TOUR.
Input: A mixed graph $M = (V, E, A)$, and vectors $l, u \in \mathbb{Z}_+^{E \cup A}$.
Output: Does M have a bounded postman tour?

Problem: MINIMUM BOUNDED MIXED POSTMAN TOUR.
Input: A strongly connected mixed graph $M = (V, E, A)$, vectors $l, u \in \mathbb{Z}_+^{E \cup A}$ with $l \leq u$, and a vector $c \in \mathbb{Q}_+^{E \cup A}$.
Output: The minimum cost MBMPT(M, l, u, c) of a bounded postman tour of M.

The study of these problems was originally suggested by Edmonds and Johnson [29]. Since the decision version of MINIMUM MIXED POSTMAN TOUR is $\mathcal{NP}$-complete, it follows that the decision version of MINIMUM BOUNDED MIXED POSTMAN TOUR is also $\mathcal{NP}$-complete. Tohyama and Adachi proved that the same is true even if all upper bounds are finite (even if $u_e = 2$ and $l_e = 1$ for all $e \in E \cup A$) [82]. It is easy to see that the following are integer programming formulations for MINIMUM BOUNDED MIXED POSTMAN TOUR:

$$\begin{aligned} \text{MBMPT1}(M, l, u, c) &= \min c_A^\top x_A + c_E^\top x_E^+ + c_E^\top x_E^- && (3.57)\\ \text{subject to} & && \\ x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) &= 0 \text{ for all } v \in V && (3.58)\\ u_a \;\geq\; x_a &\geq l_a \text{ for all } a \in A && (3.59)\\ u_e \;\geq\; x_{e^+} + x_{e^-} &\geq l_e \text{ for all } e \in E && (3.60)\\ x_e & \quad \text{integer for all } e \in A \cup E^+ \cup E^- && (3.61) \end{aligned}$$

and

$$\begin{aligned} \text{MBMPT2}(M, l, u, c) &= \min c^\top x && (3.62)\\ \text{subject to} & && \\ x(\delta_{E \cup A}(v)) &\equiv 0 \text{ (mod 2) for all } v \in V && (3.63)\\ x(\delta_A(\bar{S})) + x(\delta_E(S)) - x(\delta_A(S)) &\geq 0 \text{ for all } S \subseteq V && (3.64)\\ u_e \;\geq\; x_e &\geq l_e \text{ and integer for all } e \in E \cup A. && (3.65) \end{aligned}$$

Let $\mathcal{P}^1_{BMPT}(M, l, u)$ be the convex hull of the feasible solutions to the first integer program above, let $\mathcal{Q}^1_{BMPT}(M, l, u)$ be the set of feasible solutions to its linear programming relaxation, let $\mathcal{P}^2_{BMPT}(M, l, u)$ be the convex hull of the feasible solutions to the second integer program above, and let $\mathcal{Q}^2_{BMPT}(M, l, u)$ be the set of feasible solutions to its linear programming relaxation. Similarly to Theorem 3.10, we can prove that $\mathcal{Q}^1_{BMPT}(M, l, u)$ is half-integral. We say that $e \in E$ is *bound tight* if $x_{e^+} + x_{e^-} = u_e$ or $x_{e^+} + x_{e^-} = l_e$.

Theorem 3.13 *Every extreme point x of the polyhedron $\mathcal{Q}^1_{BMPT}(M,l,u)$ is half-integral. Moreover, fractional components occur only on bound tight edges.*

The following result says that testing feasibility of MINIMUM BOUNDED MIXED POSTMAN TOUR is equivalent to testing feasibility of a flow problem (Theorem 2.12) in the directed graph $D = (V, A \cup E^+)$. Recall that E^+ is an orientation of E.

Theorem 3.14 *Assume that $u_e > l_e$ for all $e \in E$. Then $\mathcal{P}^1_{BMPT}(M,l,u)$ is not empty if and only if there exists an integer solution x of*

$$\begin{aligned} x(\delta_{A\cup E^+}(\bar{v})) - x(\delta_{A\cup E^+}(v)) &= 0 \text{ for all } v \in V && (3.66)\\ u_a \;\geq\; x_a &\geq l_a \text{ for all } a \in A && (3.67)\\ u_e \;\geq\; x_{e^+} &\geq -u_e \text{ for all } e \in E && (3.68)\\ x_e \quad & \quad \text{integer for all } e \in A \cup E^+. && (3.69) \end{aligned}$$

Hence, if $u_E > l_E$, BOUNDED MIXED POSTMAN TOUR *is solvable in polynomial time.*

Proof. Let $\bar{x}$ be the incidence vector of a bounded postman tour of (M,l,u), and define $x \in \mathbb{Z}^{A\cup E^+}$ as $x_a = \bar{x}_a$ for all $a \in A$, and $x_{e^+} = \bar{x}_{e^+} - \bar{x}_{e^-}$ for all $e \in E$. Since $u_e \geq x_{e^+} \geq -u_e$ for all $e \in E$, it follows that x is a feasible solution to the system above. Conversely, if x is a feasible solution to the system above, define $\bar{x} \in \mathbb{Z}^{A\cup E^+\cup E^-}$ by $\bar{x}_a = x_a$ for all $a \in A$, and $\bar{x}_{e^+} = \lfloor \frac{1}{2}(u_e + x_{e^+}) \rfloor$ and $\bar{x}_{e^-} = \lfloor \frac{1}{2}(u_e - x_{e^+}) \rfloor$ for all $e \in E$. Since $x_{e^+} = \bar{x}_{e^+} - \bar{x}_{e^-}$, and $u_e \geq \bar{x}_{e^+} + \bar{x}_{e^-} \geq u_e - 1 \geq l_e$ for all $e \in E$, it follows that $\bar{x}$ is the incidence vector of a bounded postman tour of (M,l,u). ∎

Theorem 3.15 *If $u_E > l_E$, then there exists a 2-approximation algorithm for* MINIMUM BOUNDED MIXED POSTMAN TOUR.

Proof. By Theorem 3.14, we can assume that the given instance of MINIMUM BOUNDED MIXED POSTMAN TOUR is feasible. Let $c \in \mathbb{Q}_+^{E\cup A}$, let x^* be the incidence vector of an optimal solution to MBMPT1(M,l,u,c), and let x be an extreme point optimal solution to its linear programming relaxation. By Theorem 3.13, x is half-integral. Furthermore, if

$e \in E$ is a fractional bound tight edge, we can assume that $x_{e^+} + x_{e^-} = l_e$ since otherwise we can decrease both x_{e^+} and x_{e^-} without losing feasibility nor increasing the cost of the solution. In this case, we can also assume that $l_e \geq 1$, for otherwise $x_{e^+} = x_{e^-} = 0$. Define a vector $\bar{x} \in \mathbb{Z}_+^{A \cup E^+ \cup E^-}$ by $\bar{x}_{e^+} = x_{e^+} + \frac{1}{2}$ and $\bar{x}_{e^-} = x_{e^-} + \frac{1}{2}$ for all fractional bound tight edges e, and $\bar{x}_a = x_a$ otherwise. Note that $\bar{x}$ is a feasible solution to MBMPT1(M, l, u, c), and that the cost due to the fractional bound tight edges is at most doubled. Therefore $c^\top x \leq c^\top x^* \leq c^\top \bar{x} \leq 2c^\top x$. ■

3.6.1 The Restricted Mixed Postman Problem

Let $R \subseteq E \cup A$ be a set of *restricted* edges and arcs. We say that a postman tour of M is *restricted* if it uses exactly once each restricted edge and arc.

Problem: RESTRICTED MIXED POSTMAN TOUR.
Input: A mixed graph $M = (V, E, A)$, and a subset $R \subseteq E \cup A$.
Output: Does M have a restricted postman tour?

Problem: MINIMUM RESTRICTED MIXED POSTMAN TOUR.
Input: A strongly connected mixed graph $M = (V, E, A)$, a subset $R \subseteq E \cup A$, and a vector $c \in \mathbb{Q}_+^{E \cup A}$.
Output: The minimum cost MRMPT(M, R, c) of a restricted postman tour of M.

There are four special cases of this problem that seem to be natural. If $R = \emptyset$ then we obtain MINIMUM MIXED POSTMAN TOUR. If $R = E \cup A$ then we obtain MIXED EULERIAN TOUR. The special case when $R = A$ is the subject of Chapter 5, and the special case when $R = E$ is the subject of Chapter 6. These two special cases were studied before by Veerasamy in his doctoral thesis [85]. Although we show that both cases share some common properties, such as both problems being $\mathcal{NP}$-hard even when restricted to planar inputs, our research produced, in each case, some other very different kinds of results. For the case $R = A$ we give results about approximability, as well as about properties of linear relaxations of the problem. For the case $R = E$ we give results about necessary conditions for feasibility, as well as about some solvable cases of the problem.

Chapter 4

The Windy Postman Problem

Ich ging hin und her, Hände in den Hosentaschen,
einmal geschoben vom Wind, geradezu schwebend,
dann wieder gegen den Wind, dann mühsam...

Homo Faber, Max Frisch

We study an $\mathcal{NP}$-hard variant of Minimum Undirected Postman Tour, known as Minimum Windy Postman Tour, where the cost of an edge depends on the direction of traversal. We give an integer programming formulation for this problem, and study the integrality of the polyhedron defined by its linear programming relaxation. It was previously known that undirected even graphs have integral *windy postman* polyhedra. We prove that undirected series-parallel graphs also have integral windy postman polyhedra.

4.1 Introduction

When we considered Minimum Undirected Postman Tour, we assumed that the cost of traversing an edge was the same for either direction. While studying some methods to approach Minimum Mixed Postman Tour, Minieka argued that [67]:

> *This is hardly a good assumption when one direction might be uphill and the other downhill, when one direction might be with the wind and the other against the wind or when fares are different depending on direction.*

Let $G = (V, E)$ be an undirected, connected graph, and let $\vec{G} = (V, E^+ \cup E^-)$ be its associated directed graph. A tour T of $\vec{G}$ is a *windy postman tour* of G if, for every $e \in E$, T contains e^+ or e^- at least once. If $c \in \mathbb{Q}_+^{E^+ \cup E^-}$ is a vector of costs on the arcs of $\vec{G}$, then the cost of T, denoted by $c(T)$, is the sum of the costs of the arcs used by T.

Problem: MINIMUM WINDY POSTMAN TOUR.
Input: An undirected graph $G = (V, E)$, and a vector $c \in \mathbb{Q}_+^{E^+ \cup E^-}$.
Output: The minimum cost MWPT(G, c) of a windy postman tour of (G, c).

Just as for MINIMUM MIXED POSTMAN TOUR, a natural generalization of this problem is to find a minimum cost collection T of circuits of $\vec{G}$ such that, for each $e \in E$, T contains e^+ or e^- at least l_e and at most u_e times. We call T a *bounded windy postman tour* of G. In this case, both the feasibility and the optimization problems are interesting.

Problem: BOUNDED WINDY POSTMAN TOUR.
Input: An undirected graph $G = (V, E)$, and vectors $l, u \in \mathbb{Z}_+^E$.
Output: Does (G, l, u) have a bounded windy postman tour?

Problem: MINIMUM BOUNDED WINDY POSTMAN TOUR.
Input: An undirected graph $G = (V, E)$, vectors $l, u \in \mathbb{Z}_+^E$, and a vector $c \in \mathbb{Q}_+^{E^+ \cup E^-}$.
Output: The minimum cost MBWPT(G, c) of a windy postman tour of (G, l, u, c).

Guan proved that the decision version of MINIMUM WINDY POSTMAN TOUR is $\mathcal{NP}$-complete via a reduction from MINIMUM MIXED POSTMAN TOUR, and hence this remains true even if we restrict the input to be planar [49]. Win gave two distinct 2-approximation algorithms [89, 90], and Raghavachari and Veerasami gave a $\frac{3}{2}$-approximation algorithms for MINIMUM WINDY POSTMAN TOUR [77]. Currently, the latter is the best approximation algorithm for MINIMUM WINDY POSTMAN TOUR. Grötschel and Win gave a cutting plane algorithm to find optimal solutions for MINIMUM WINDY POSTMAN TOUR [47]. Pearn and Li gave some algorithms to improve solutions [74].

Du and Sun showed that the decision version of MINIMUM BOUNDED WINDY POSTMAN TOUR is $\mathcal{NP}$-complete even if all bounds are finite (even if $l_e = 1$ and $u_e = 2$ for all $e \in E$) [22]. A simple linear-time algorithm to decide whether an undirected graph $G = (V, E)$ has a bounded windy postman tour consists in verifying a parity condition on the connected components of the subgraph of G with edge set $F = \{e \in E : l_e < u_e\}$.

4.2 Integer Programming Formulation

An integer programming formulation for MINIMUM WINDY POSTMAN TOUR due to Win is as follows [89, 90]:

$$\begin{aligned} \mathrm{MWPT}(G, c) &= \min c^\top x && (4.1)\\ \text{subject to} & && \\ x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) &= 0 \text{ for all } v \in V && (4.2)\\ x_{e^+} + x_{e^-} &\geq 1 \text{ for all } e \in E && (4.3)\\ x_{e^+}, x_{e^-} &\geq 0 \text{ for all } e \in E && (4.4)\\ x_{e^+}, x_{e^-} &\quad \text{integral for all } e \in E. && (4.5) \end{aligned}$$

Let $\mathcal{P}_{WPT}(G)$ be the convex hull of the feasible solutions to the integer program above, and let $\mathcal{Q}_{WPT}(G)$ be the set of feasible solutions to its linear programming relaxation.

We can obtain a similar integer programming formulation for MINIMUM BOUNDED WINDY POSTMAN TOUR as follows:

$$\begin{aligned} \mathrm{MBWPT}(G, l, u, c) &= \min c^\top x && (4.6)\\ \text{subject to} & && \\ x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) &= 0 \text{ for all } v \in V && (4.7)\\ u_e \;\geq\; x_{e^+} + x_{e^-} &\geq l_e \text{ for all } e \in E && (4.8)\\ x_{e^+}, x_{e^-} &\geq 0 \text{ for all } e \in E && (4.9)\\ x_{e^+}, x_{e^-} &\quad \text{integral for all } e \in E. && (4.10) \end{aligned}$$

Let $\mathcal{P}_{WPT}(G, l, u)$ be the convex hull of the feasible solutions to the integer program above, and let $\mathcal{Q}_{WPT}(G, l, u)$ be the set of feasible solutions to its linear programming relaxation.

We observe that the integer programs MWPT(G, c) and MBWPT(G, l, u, c) are very similar to the integer programs MMPT1(M, c) and MBMPT1(M, l, u, c) for MINIMUM

MIXED POSTMAN TOUR. In fact, if G is the underlying undirected graph of the mixed graph M, then $\mathcal{Q}_{MPT1}(M)$ and $\mathcal{Q}_{MPT1}(M, l, u)$ are faces of $\mathcal{Q}_{WPT}(G)$ and $\mathcal{Q}_{WPT}(G, l, u)$, respectively. Hence, the integrality of the latter implies the integrality of the former.

Conversely, since $\mathcal{Q}_{WPT}(G) = \mathcal{Q}_{MPT1}(\vec{G})$ and $\mathcal{Q}_{WPT}(G, l, u) = \mathcal{Q}_{MPT1}(\vec{G}, l, u)$, the results about integrality and half-integrality (Theorems 3.10 and 3.13) in Chapter 3 apply to MINIMUM WINDY POSTMAN TOUR. Win proved the stronger statement [89, 90]:

Theorem 4.1 (Win) *Every extreme point x of the polyhedron $\mathcal{Q}_{WPT}(G)$ has components whose values are either $\frac{1}{2}$ or a nonnegative integer. Furthermore, $\mathcal{Q}_{WPT}(G)$ is integral if and only if G is even.*

This implies that MINIMUM WINDY POSTMAN TOUR can be solved in polynomial time for the class of undirected Eulerian graphs.

Corollary 4.2 *Every extreme point x of the polyhedron $\mathcal{Q}_{BWPT}(G, l, u)$ is half-integral.*

As before, we can strengthen the linear programming relaxation of MINIMUM WINDY POSTMAN TOUR by adding *odd-cut constraints*. Let $S \subset V$ be such that $d_G(S)$ is odd. Then, in any windy postman tour of G, at least one element of $\vec{\delta}(S) \cup \vec{\delta}(\bar{S})$ must be used more than once. Therefore, the inequality

$$x(\vec{\delta}(S)) + x(\vec{\delta}(S)) \geq d_G(S) + 1 \tag{4.11}$$

is valid for $\mathcal{P}_{WPT}(G)$. Let $\mathcal{O}_{WPT}(G)$ be the subset of $\mathcal{Q}_{WPT}(G)$ that satisfies the odd-cut constraints (4.11). We can also give some odd-cut constraints for the bounded case. Let $S \subset V$ be such that $l(\delta_G(S))$ is odd, and let $T \subset V$ be such that $u(\delta_G(T))$ is odd. Then any bounded windy postman tour of G uses at least $l(\delta_G(S)) + 1$ elements of $\vec{\delta}(S) \cup \vec{\delta}(\bar{S})$, and at most $u(\delta_G(T)) - 1$ elements of $\vec{\delta}(T) \cup \vec{\delta}(\bar{T})$. Therefore, the inequalities

$$x(\vec{\delta}(S)) + x(\vec{\delta}(S)) \geq l(\delta_G(S)) + 1 \tag{4.12}$$

and

$$x(\vec{\delta}(T)) + x(\vec{\delta}(T)) \leq u(\delta_G(T)) - 1 \tag{4.13}$$

are valid for $\mathcal{P}_{WPT}(G, l, u)$. Let $\mathcal{O}_{WPT}(G, l, u)$ be the subset of $\mathcal{Q}_{WPT}(G, l, u)$ that satisfies these odd-cut constraints. In the rest of this chapter, we assume that there are no upper bounds, and hence we refer to the polyhedra $\mathcal{P}_{WPT}(G, l)$, $\mathcal{Q}_{WPT}(G, l)$, and $\mathcal{O}_{WPT}(G, l)$.

4.3 Windy Postman Perfect Graphs

We say that an undirected graph G is *windy postman perfect* if the polyhedron $\mathcal{O}_{WPT}(G)$ is integral. Equivalently, an undirected graph G is windy postman perfect if $\mathcal{O}_{WPT}(G)$ is the convex hull $\mathcal{P}_{WPT}(G)$ of incidence vectors of windy postman tours of G.

The class of windy postman perfect graphs was introduced and studied extensively by Win [89, 90]. He observed that there exists a polynomial-time algorithm, based on the ellipsoid method, to solve Minimum Windy Postman Tour for the class of windy postman perfect graphs. This is a consequence of the equivalence of optimization and separation (Theorem 1.8) and the fact that the flow (4.2), lower bound (4.3), nonnegativity (4.4), and odd-cut constraints (4.11) can all be separated in polynomial time.

By Theorem 4.1, even undirected graphs are windy postman perfect. Win also proved that undirected forests and all undirected graphs with two vertices are windy postman perfect [89]. As with other graph properties, we might think that windy postman perfection is closed under taking graph minors. However, this is not true: K_5 is windy postman perfect, but $\mathcal{O}_{WPT}(K_4)$ has the fractional extreme point shown in Figure 4.1.

Nevertheless, Win proposed four operations that preserve windy postman perfection [89].

Theorem 4.3 (Win) *Let G, G_1, G_2 be windy postman perfect graphs.*

1. *Any subdivision of G is windy postman perfect.*
2. *If $e \in E(G)$, then $G \,/\, e$ is windy postman perfect.*
3. *If $e, f \in E(G)$ are parallel, then $G \setminus \{e, f\}$ is windy postman perfect.*
4. *If $v_1 \in V(G_1)$ and $v_2 \in V(G_2)$, then the undirected graph G_3 obtained by identifying the vertices v_1 and v_2 is windy postman perfect.*

We observe that the class of even undirected graphs is closed under each of these four operations, and that the same is true for the class of undirected forests. Another class of undirected graphs that has this property is the class of series-parallel undirected graphs. Win conjectured that these are also windy postman perfect [89]. In what follows, we prove a statement stronger than Win's conjecture.

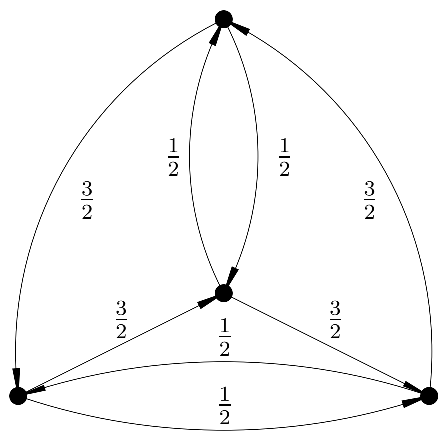

Figure 4.1: A fractional extreme point of $\mathcal{O}_{WPT}(K_4)$.

It is possible to verify, say using the polyhedral software package `PORTA` [12], that $K_{3,3}$ is also windy postman perfect. This example is interesting because $K_{3,3}$ is neither even nor series-parallel, and it cannot be obtained from these classes of graphs using the operations described in Theorem 4.3. Another interesting observation is that, although the polyhedron $\mathcal{Q}_{WPT}(G)$ is always half-integral, this is not true in general for the polyhedron $\mathcal{O}_{WPT}(G)$. In Figure 4.2, we show some undirected graphs G for which $\mathcal{O}_{WPT}(G)$ has both $\frac{1}{2}$-integral and $\frac{1}{3}$-integral extreme points. Using `PORTA`, we have verified that for all simple graphs G with at most 6 vertices, $\mathcal{O}_{WPT}(G)$ is $\frac{1}{6}$-integral.

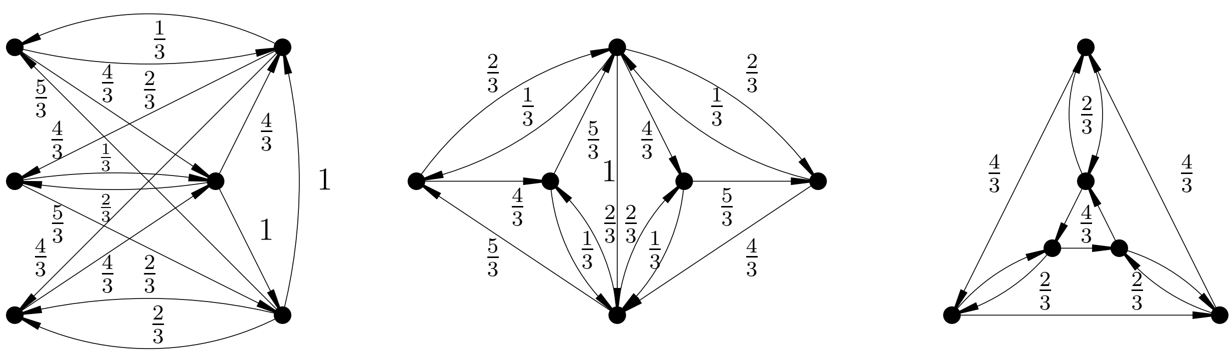

Figure 4.2: Some undirected graphs with $\frac{1}{3}$-integral extreme points.

4.3.1 Windy Postman Ideal Graphs

Let $G = (V, E)$ be an undirected graph, let $l \in \mathbb{Z}_+^E$, and let $b \in \mathbb{Z}^V$ with $b(V) = 0$. We say that $S \subset V$ is an *odd set* if $b(S) + l(\delta_E(S))$ is odd. Let $\vec{G} = (V, E^+ \cup E^-)$ be the associated directed graph of G, and let $\mathcal{O}(G, l, b)$ be the set of feasible solutions to the system

$$x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) = b_v \text{ for all } v \in V \tag{4.14}$$

$$x_{e^+} + x_{e^-} \geq l_e \text{ for all } e \in E \tag{4.15}$$

$$x(\vec{\delta}(S)) + x(\vec{\delta}(\bar{S})) \geq l(\delta_E(S)) + 1 \text{ for all odd } S \subset V \tag{4.16}$$

$$x_{e^+}, x_{e^-} \geq 0 \text{ for all } e \in E. \tag{4.17}$$

We say that G is *windy postman ideal* if the polyhedron $\mathcal{O}(G, l, b)$ is integral for all possible choices of l and b. Observe that windy postman ideal graphs are windy postman perfect. We prove that windy postman ideal graphs are precisely the series-parallel graphs, proving Win's conjecture as a consequence. In contrast to windy postman perfection, windy postman ideality is closed under taking graph minors.

Theorem 4.4 *Let $G = (V, E)$ be a windy postman ideal undirected graph, and let $e \in E$. Then $G \setminus e$ and $G \,/\, e$ are also windy postman ideal.*

Proof. Let u, v be the ends of e. Let $l' \in \mathbb{Z}_+^{E \setminus e}$, and let $b' \in \mathbb{Z}^{V(G \,/\, e)}$ with $b'(V(G \,/\, e)) = 0$. Let $l \in \mathbb{Z}_+^E$ and $b \in \mathbb{Z}^V$ be defined by $l_f = l'_f$ for all $f \in E \setminus e$ and $l_e = 0$, and $b_w = b'_w$ for all $w \in V \setminus \{u, v\}$, $b_u = b'_e$, and $b_v = 0$. Since G is windy postman ideal, $\mathcal{O}(G, l, b)$ is integral. Since $\mathcal{O}(G \,/\, e, l', b')$ is the projection of $\mathcal{O}(G, l, b)$ onto $x_{e^+} = 0$ and $x_{e^-} = 0$, it is also integral. Hence, $G \,/\, e$ is windy postman ideal.

Let $l' \in \mathbb{Z}_+^{E \setminus e}$, and let $b' \in \mathbb{Z}^V$ with $b'(V) = 0$. Define $l \in \mathbb{Z}_+^E$ by $l_f = l'_f$ for all $f \in E \setminus e$ and $l_e = 0$. Since G is windy postman ideal, $\mathcal{O}(G, l, b')$ is integral. Since $\mathcal{O}(G \setminus e, l', b')$ is a face of $\mathcal{O}(G, l, b')$, it is also integral. Hence, $G \setminus e$ is windy postman ideal. ∎

Let $x \in \mathcal{O}(G, l, b)$, and let $e \in E$. We say that e is *integral* if both x_{e^+} and x_{e^-} are integral, and we say that e is *fractional* otherwise. We say that e is *tight* if $x_{e^+} + x_{e^-} = l_e$.

Lemma 4.5 *Let $G = (V, E)$ be a minor minimal, non windy postman ideal undirected graph. Let $b \in \mathbb{Z}^V$ and $l \in \mathbb{Z}_+^E$ be such that $\mathcal{O}(G, l, b)$ is not integral, and let x be one of its fractional extreme points. If $e \in E$ is integral, then $x_{e^+} + x_{e^-} = l_e + 1$.*

Proof. Let e^+ be oriented from u to v, and e^- be oriented from v to u. Assume first that e is tight. Let $H = G \setminus e$. Define the vectors $x' \in \mathbb{Q}_+^{E^+ \cup E^- \setminus \{e^+, e^-\}}$, $l' \in \mathbb{Z}_+^{E \setminus e}$ and $b' \in \mathbb{Z}^V$ by $x'_a = x_a$ for all $a \in E^+ \cup E^- \setminus \{e^+, e^-\}$, $l'_h = l_h$ for all $h \in E \setminus e$, and

$$b'_w = \begin{cases} b_u - x_{e^+} + x_{e^-} & \text{if } w = u \\ b_v + x_{e^+} - x_{e^-} & \text{if } w = v \\ b_w & \text{otherwise.} \end{cases} \tag{4.18}$$

Observe that (G, l, b) and (H, l', b') have the same odd sets. Since H is windy postman ideal, x' is fractional, and $x' \in \mathcal{O}(H, l', b')$, there exist distinct vectors $y', z' \in \mathcal{O}(H, l', b')$ such that $x' = \frac{1}{2}(y' + z')$. Define the vectors $y, z \in \mathbb{Q}^{E^+ \cup E^-}$ by $y_a = y'_a$ and $z_a = z'_a$ for all $a \in E^+ \cup E^- \setminus \{e^+, e^-\}$, $y_{e^+} = z_{e^+} = x_{e^+}$ and $y_{e^-} = z_{e^-} = x_{e^-}$. Then $y \in \mathcal{O}(G, l, b)$, $z \in \mathcal{O}(G, l, b)$, and $x = \frac{1}{2}(y + z)$, a contradiction to the choice of x.

Now assume that $x_{e^+} + x_{e^-} \geq l_e + 2$. We can also assume that $x_{e^+} \geq 1$. Observe that edge e does not cross any *tight odd set*, that is, if $S \subset V$ is odd and $e \in \delta(S)$, then

$$x(\vec{\delta}(S)) + x(\vec{\delta}(\bar{S})) > l(\delta_E(S)) + 1. \tag{4.19}$$

Let $H = G \;/\; e$, and define the vectors $x' \in \mathbb{Q}_+^{E^+ \cup E^- \setminus \{e^+, e^-\}}$, $l' \in \mathbb{Z}_+^{E \setminus e}$ and $b' \in \mathbb{Z}^V(H)$ by $x'_a = x_a$ for all $a \in E^+ \cup E^- \setminus \{e^+, e^-\}$, $l'_h = l_h$ for all $h \in E \setminus e$, $b'_w = b_w$ for all $w \in V \setminus \{u, v\}$ and $b'_e = b_u + b_v$. Since H is windy postman ideal, x' is fractional, and $x' \in \mathcal{O}(H, l', b')$, there exist distinct vectors $y', z' \in \mathcal{O}(H, l', b')$ such that $x' = \frac{1}{2}(y' + z')$. We may choose y', z' so that $\|y' - z'\|$ is arbitrarily small. Define α by

$$2\alpha = y'(\vec{\delta}(\bar{u}) \setminus e^-) - y'(\vec{\delta}(u) \setminus e^+) - z'(\vec{\delta}(\bar{u}) \setminus e^-) + z'(\vec{\delta}(u) \setminus e^+), \tag{4.20}$$

and define the vectors $y, z \in \mathbb{Q}^{E^+ \cup E^-}$ by

$$y_a = \begin{cases} x_{e^+} + \alpha & \text{if } a = e^+ \\ x_{e^-} & \text{if } a = e^- \\ y'_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} x_{e^+} - \alpha & \text{if } a = e^+ \\ x_{e^-} & \text{if } a = e^- \\ z'_a & \text{otherwise.} \end{cases} \tag{4.21}$$

Using that $\|y' - z'\|$ is arbitrarily small, we can show that $y \in \mathcal{O}(G, l, b)$, $z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, a contradiction to the choice of x. ∎

The following two lemmas imply that we only need to consider 2-vertex-connected undirected graphs. We leave their straightforward proofs to the reader.

Lemma 4.6 *Let $G = (V, E)$ be an undirected graph, and let $G_1, \ldots, G_k$ be its connected components. Let $b \in \mathbb{Z}^V$ and $l \in \mathbb{Z}_+^E$. For every $1 \leq i \leq k$, let b^i and l^i be the restrictions of b and l to G_i. If $\mathcal{O}(G_i, l^i, b^i)$ is integral for all $1 \leq i \leq k$, then $\mathcal{O}(G, l, b)$ is also integral. Hence, G is windy postman ideal if and only if G_i is windy postman ideal for all $1 \leq i \leq k$.*

Lemma 4.7 *Let $G = (V, E)$ be an undirected graph with a cut vertex v, and let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be the partition of G induced by v. Let $b \in \mathbb{Z}^V$ and $l \in \mathbb{Z}_+^E$. For $i \in \{1, 2\}$, let b^i and l^i be the restrictions of b and l to G_i, except $b_v^1 = b(V_2)$ and $b_v^2 = b(V_1)$. Then $\mathcal{O}(G, l, b)$ is integral if and only if $\mathcal{O}(G_1, l^1, b^1)$ and $\mathcal{O}(G_2, l^2, b^2)$ are integral. Hence, G is windy postman ideal if and only if G_1 and G_2 are also windy postman ideal.*

Now we prove our characterization of windy postman ideal graphs.

Theorem 4.8 *Let $G = (V, E)$ be an undirected graph. Then G is windy postman ideal if and only if G is series-parallel.*

Proof. Since $\mathcal{O}_{WPT}(K_4)$ is not integral, it follows that windy postman ideal graphs must be series-parallel. Let $G = (V, E)$ be a minor minimal, non windy postman ideal series-parallel graph. By Lemmas 4.6 and 4.7, we can assume that G is 2-vertex-connected. We can verify that all series-parallel graphs with at most two vertices are windy postman ideal. Hence, we can assume that G has two edges in parallel or two edges in series. Let $\vec{G} = (V, E^+ \cup E^+)$ be the associated directed graph of G, let $l \in \mathbb{Z}_+^E$, and let $b \in \mathbb{Z}^V$ with $b(V) = 0$. For a contradiction, assume that x is a fractional extreme point of $\mathcal{O}(G, l, b)$.

Parallel case. Assume first that G has two *parallel* edges e and f, with ends u and v. Let $H = (V, F)$ be the undirected graph obtained from G by replacing edges e and f by a single edge g, and let $\vec{H} = (V, F^+ \cup F^-)$ be its associated directed graph. We can assume that e^+, f^+, and g^+ are oriented from u to v, that e^-, f^-, and g^- are oriented from v to u, and that all other arcs of $\vec{G}$ and $\vec{H}$ are oriented consistently. Define $l' \in \mathbb{Z}_+^F$ by $l'_h = l_h$ if $h \neq g$, and $l'_g = l_e + l_f$. Observe that (G, l, b) and (H, l', b) have the same odd sets.

Define the vector $x' \in \mathbb{Q}_+^{F^+ \cup F^-}$ by $x'_a = x_a$ if $a \notin \{g^+, g^-\}$, $x'_{g^+} = x'_{e^+} + x'_{f^+}$, and $x'_{g^-} = x'_{e^-} + x'_{f^-}$, and observe that $x' \in \mathcal{O}(H, l', b)$. Assume first that x' is integral. Then x_a is integral for all $a \notin \{e^+, e^-, f^+, f^-\}$, x_{e^+} is integral if and only if x_{f^+} is integral, and x_{e^-} is integral if and only if x_{f^-} is integral. Since x is fractional, we can assume without loss of generality that x_{e^+} and x_{f^+} are fractional. Hence $x_{e^+} > 0$ and $x_{f^+} > 0$.

Case P1: If x_{e^-} is integral, then x_{f^-} is integral. Since x_{e^+} and x_{f^+} are fractional, neither e nor f is tight. Let $\alpha = \min\{x_{e^+}, x_{f^+}, x_{e^+} + x_{e^-} - l_e, x_{f^+} + x_{f^-} - l_f\} > 0$, and define the vectors y and z by

$$y_a = \begin{cases} x_{e^+} + \alpha & \text{if } a = e^+ \\ x_{f^+} - \alpha & \text{if } a = f^+ \\ x_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} x_{e^+} - \alpha & \text{if } a = e^+ \\ x_{f^+} + \alpha & \text{if } a = f^+ \\ x_a & \text{otherwise.} \end{cases} \tag{4.22}$$

Observe that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, contradicting the choice of x.

Case P2: If x_{e^-} is fractional, then x_{f^-} is fractional. Hence $x_{e^-} > 0$ and $x_{f^-} > 0$. Let $\alpha = \min\{x_{e^+}, x_{e^-}, x_{f^+}, x_{f^-}\} > 0$, and define the vectors y and z by

$$y_a = \begin{cases} x_a + \alpha & \text{if } a \in \{e^+, f^-\} \\ x_a - \alpha & \text{if } a \in \{e^-, f^+\} \\ x_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} x_a - \alpha & \text{if } a \in \{e^+, f^-\} \\ x_a + \alpha & \text{if } a \in \{e^-, f^+\} \\ x_a & \text{otherwise.} \end{cases} \tag{4.23}$$

Observe that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, contradicting the choice of x.

Since we get contradictions in both cases, it follows that x' is fractional. Since H has fewer edges than G, $\mathcal{O}(H, l', b)$ is integral. Hence, there exist distinct vectors $y', z' \in \mathcal{O}(H, l', b)$ such that $x' = \frac{1}{2}(y' + z')$. We may choose y', z' so that $\|y' - z'\|$ is arbitrarily small.

Assume that neither e nor f is tight. If both x_{e^+} and x_{f^+} are positive, or both x_{e^-} and x_{f^-} are positive, we can obtain a contradiction in a similar way to Case P1. Hence, we may assume, by interchanging e and f if necessary, that $x_{e^+} > l_e$, $x_{f^-} > l_f$, $x_{e^-} = 0$, and $x_{f^+} = 0$. Define the vectors y and z by

$$y_a = \begin{cases} y'_{g^+} & \text{if } a = e^+ \\ y'_{g^-} & \text{if } a = f^- \\ 0 & \text{if } a \in \{e^-, f^+\} \\ y'_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} z'_{g^+} & \text{if } a = e^+ \\ z'_{g^-} & \text{if } a = f^- \\ 0 & \text{if } a \in \{e^-, f^+\} \\ z'_a & \text{otherwise.} \end{cases} \tag{4.24}$$

Using that $\|y' - z'\|$ is arbitrarily small, we can conclude that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, contradicting the choice of x.

Hence, we can assume without loss of generality that e is tight. If all of x_{e^+}, x_{e^-}, x_{f^+}, and x_{f^-} are positive, we can obtain a contradiction in a similar way to Case P2. Hence, we can assume that at least one of x_{e^+}, x_{e^-}, x_{f^+}, or x_{f^-} is zero. If $x_{e^-} = 0$, we contradict Lemma 4.5. Hence, we can assume that $x_{e^+} > 0$, $x_{e^-} > 0$, $x_{f^+} > l_f$, and $x_{f^-} = 0$. (If $x_{f^+} = l_f$, we interchange the roles of e and f.) Define the vectors y and z by

$$y_a = \begin{cases} l_e - y'_{g^-} & \text{if } a = e^+ \\ y'_{g^-} & \text{if } a = e^- \\ y'_{g^+} + y'_{g^-} - l_e & \text{if } a = f^+ \\ 0 & \text{if } a = f^- \\ y'_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} l_e - z'_{g^-} & \text{if } a = e^+ \\ z'_{g^-} & \text{if } a = e^- \\ z'_{g^+} + z'_{g^-} - l_e & \text{if } a = f^+ \\ 0 & \text{if } a = f^- \\ z'_a & \text{otherwise.} \end{cases} \tag{4.25}$$

Using that $\|y' - z'\|$ is arbitrarily small, it is not difficult to verify that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, a last contradiction to the choice of x.

Series case. Now assume that G has two edges e and f in *series*, with ends u and v, and v and w, respectively. Assume first that all edges in $E \setminus e$ are integral, and e is fractional. By Lemma 4.5, $x_{g^+} + x_{g^-} = l_g + 1$ for all $g \in E \setminus e$. Since e is fractional, but $x_{e^+} - x_{e^-} = b_v - x_{f^-} + x_{f^+}$ is integral, it follows that $x_{e^+} > 0$ and $x_{e^-} > 0$. Since G is 2-edge-connected, it follows that e cannot be an odd cut by itself, and hence it must be tight. Furthermore, our assumptions imply that

$$x_{e^+} = \tfrac{1}{2}(l_e + l_f + b_v + 1) - x_{f^-} \text{ and } x_{e^-} = \tfrac{1}{2}(l_e + l_f - b_v + 1) - x_{f^+}, \tag{4.26}$$

and hence $l_e + l_f + b_v$ must be even (otherwise e would be integral). Let $C \subseteq E$ be the edge set of a cycle containing e. We assume without loss of generality that, for each edge $g \in C \setminus e$, the arc g^+ satisfies $x_{g^+} \geq 1$. We say that $g^+ \in C^+ \setminus e^+$ is *forward* if it has the same orientation as e^+ along C; otherwise it is *backward*. Define the vectors y and z by

$$y_a = \begin{cases} x_{e^+} + \frac{1}{2} & \text{if } a = e^+ \\ x_{e^-} - \frac{1}{2} & \text{if } a = e^- \\ x_a + 1 & \text{if } a \text{ is forward} \\ x_a - 1 & \text{if } a \text{ is backward} \\ x_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} x_{e^+} - \frac{1}{2} & \text{if } a = e^+ \\ x_{e^-} + \frac{1}{2} & \text{if } a = e^- \\ x_a - 1 & \text{if } a \text{ is forward} \\ x_a + 1 & \text{if } a \text{ is backward} \\ x_a & \text{otherwise.} \end{cases} \tag{4.27}$$

Note that y and z satisfy the flow (4.14), lower bound (4.15), and nonnegativity (4.17) constraints. Since y and z are integral, they satisfy the odd-cut constraints (4.16). Hence, $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, contradicting the choice of x.

Therefore, there are fractional edges in $E \setminus e$ and in $E \setminus f$. In a similar way, we obtain a contradiction if we assume that the only two fractional edges are e and f. Hence, there are fractional edges in $E \setminus \{e, f\}$. Let $\mathcal{S}_e$ be the set of all odd sets crossed by e, and let

$$s_e = \min_{S \in \mathcal{S}_e} x(\vec{\delta}(S)) + x(\vec{\delta}(\bar{S})) - l(\delta(S)) - 1. \tag{4.28}$$

We define $\mathcal{S}_f$ and s_f in a similar way. We assume without loss of generality that $s_e \geq s_f$. The rest of the proof is divided into five main cases.

Case S1: If $s_e \geq s_f > 0$, then neither e nor f crosses a tight odd set. Let $H = (V \setminus \{u, w\}, F)$ be obtained from contracting e and f in G, and let $\vec{H}$ be its associated directed graph. Define the vectors $x' \in \mathbb{Q}_+^{F^+ \cup F^-}$, $l' \in \mathbb{Z}_+^F$ and $b' \in \mathbb{Z}^{V \setminus \{u,w\}}$ by $x'_a = x_a$ for all $a \in F^+ \cup F^-$, $l'_h = l_h$ for all $h \in F$, $b'_t = b_t$ for all $t \in V \setminus \{u, v, w\}$ and $b'_v = b_u + b_v + b_w$. Since H is windy postman ideal, x' is fractional, and $x' \in \mathcal{O}(H, l', b')$, there exist distinct vectors $y', z' \in \mathcal{O}(H, l', b')$ such that $x' = \frac{1}{2}(y' + z')$. We may choose y', z' so that $\|y' - z'\|$ is arbitrarily small. Define α by

$$2\alpha = y'(\vec{\delta}(\bar{u}) \setminus e^-) - y'(\vec{\delta}(u) \setminus e^+) - z'(\vec{\delta}(\bar{u}) \setminus e^-) + z'(\vec{\delta}(u) \setminus e^+). \tag{4.29}$$

Define the vectors $y, z \in \mathbb{Q}^{E^+ \cup E^-}$ as follows: For all $a \in F^+ \cup F^-$, let $y_a = y'_a$ and $z_a = z'_a$. If $x_{e^-} = 0$, then $y_{e^+} = x_{e^+} + \alpha$, $z_{e^+} = x_{e^+} - \alpha$, and $y_{e^-} = z_{e^-} = 0$. If $x_{e^+} = 0$, then $y_{e^-} = x_{e^-} - \alpha$, $z_{e^-} = z_{e^-} + \alpha$, and $y_{e^+} = z_{e^+} = 0$. Otherwise, $y_{e^+} = x_{e^+} + \frac{1}{2}\alpha$, $y_{e^-} = x_{e^-} - \frac{1}{2}\alpha$, $z_{e^+} = x_{e^+} - \frac{1}{2}\alpha$, and $z_{e^-} = x_{e^-} + \frac{1}{2}\alpha$. Define y_{f^+}, y_{f^-}, z_{f^+}, and z_{f^-} in a similar way. Using that $\|y' - z'\|$ is arbitrarily small, we can show that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, contradicting the choice of x.

Case S2: If $s_e > s_f = 0$, then e does not cross any tight odd set, but f does. This case can be handled in a similar way to Case S1, except that we use $H = G \,/\, e$.

Hence, we can assume that $s_e = s_f = 0$, that is, both e and f cross tight odd sets. We assume without loss of generality that $x_{e^+} + x_{e^-} - l_e \geq x_{f^+} + x_{f^-} - l_f$.

Case S3: Assume that $\{v\}$ is even. Since e crosses the odd set S if and only if f crosses the odd set $S \triangle v$, it follows that $x_{e^+} + x_{e^-} - l_e = x_{f^+} + x_{f^-} - l_f$. We assume without loss of generality that $\min\{x_{e^+}, x_{e^-}, x_{f^+}, x_{f^-}\} \in \{x_{f^+}, x_{f^-}\}$. Assume first that this minimum is positive. Let $H = G \,/\, e$, and let $\vec{H}$ be its associated directed graph. Define the vectors $x' \in \mathbb{Q}_+^{E^+ \cup E^- \setminus \{e^+, e^-\}}$, $l' \in \mathbb{Z}_+^{E \setminus e}$ and $b' \in \mathbb{Z}^{V(H)}$ by $x'_a = x_a$ for all $a \in E^+ \cup E^- \setminus \{e^+, e^-\}$, $l'_h = l_h$ for all $h \in E \setminus e$, $b'_e = b_u + b_v$, and $b'_t = b_t$ for all $t \in V(H) \setminus e$. Since H is windy postman ideal, x' is fractional, and $x' \in \mathcal{O}(H, l', b')$, there exist distinct vectors $y', z' \in \mathcal{O}(H, l', b')$ such that $x' = \frac{1}{2}(y' + z')$. We may choose y', z' so that $\|y' - z'\|$ is arbitrarily small. Define α and β by $2\alpha = y'_{f^+} - z'_{f^+}$ and $2\beta = y'_{f^-} - z'_{f^-}$, and define the vectors $y, z \in \mathbb{Q}^{E^+ \cup E^-}$ by

$$y_a = \begin{cases} x_{e^+} + \alpha & \text{if } a = e^+ \\ x_{e^-} + \beta & \text{if } a = e^- \\ y'_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} x_{e^+} - \alpha & \text{if } a = e^+ \\ x_{e^-} - \beta & \text{if } a = e^- \\ z'_a & \text{otherwise.} \end{cases} \tag{4.30}$$

Using that $\|y' - z'\|$ is arbitrarily small, we can show that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, contradicting the choice of x.

Without loss of generality, we can now assume that $x_{f^-} = 0$. If $x_{e^+} > 0$, then the above construction works (observe that $\beta = 0$). Hence, we can assume that $x_{e^+} = 0$. Since $x_{e^-} + x_{f^+} = -b_v$ and $x_{e^-} - x_{f^+} = l_e - l_f$, it follows that $x_{e^-} = \frac{1}{2}(l_e - l_f - b_v)$ and $x_{f^+} = \frac{1}{2}(l_f - l_e - b_v)$. Since $b_v + l_e + l_f$ is even, both x_{e^-} and x_{f^+} are integral. By Lemma 4.5, $x_{e^-} = l_e + 1$, $x_{f^+} = l_f + 1$, and $b_v = -(l_e + l_f + 2)$. But then the above construction works again (observe that $\alpha = \beta = 0$).

Hence, we can assume that $\{v\}$ is odd. Let $t_v = x_{e^+} + x_{e^-} + x_{f^+} + x_{f^-} - l_e - l_f - 1$. Let $\mathcal{T}_e$ be the set of all odd sets crossed by e, except for $\{v\}$ and its complement, let

$$t_e = \min_{T \in \mathcal{T}_e} x(\vec{\delta}(T)) + x(\vec{\delta}(\bar{T})) - l(\delta(T)) - 1, \tag{4.31}$$

and let $T_e \in \mathcal{T}_e$ achieve this minimum. Define $\mathcal{T}_f$, t_f, and T_f in a similar way. Since both e and f cross tight odd sets, it follows that either $t_v = 0$, or $t_v > 0$ and $t_e = t_f = 0$.

Case S4: Assume that $t_v = 0$. Let $H = G \,/\, e$, and let $\vec{H}$ be its associated directed graph. Define the vectors $x' \in \mathbb{Q}_+^{E^+ \cup E^- \setminus \{e^+, e^-\}}$, $l' \in \mathbb{Z}_+^{E \setminus e}$ and $b' \in \mathbb{Z}^{V(H)}$ by $x'_a = x_a$ for all $a \in E^+ \cup E^- \setminus \{e^+, e^-\}$, $l'_h = l_h$ for all $h \in E \setminus e$, $b'_e = b_u + b_v$, and $b'_t = b_t$ for all

$t \in V(H) \setminus e$. Since H is windy postman ideal, x' is fractional, and $x' \in \mathcal{O}(H, l', b')$, there exist distinct vectors $y', z' \in \mathcal{O}(H, l', b')$ such that $x' = \frac{1}{2}(y' + z')$. We may choose y', z' so that $\|y' - z'\|$ is arbitrarily small. Define α and β by $2\alpha = z'_{f^+} - y'_{f^+}$ and $2\beta = z'_{f^-} - y'_{f^-}$, and define the vectors $y, z \in \mathbb{Q}^{E^+ \cup E^-}$ by

$$y_a = \begin{cases} x_{e^+} + \alpha & \text{if } a = e^+ \\ x_{e^-} + \beta & \text{if } a = e^- \\ y'_a & \text{otherwise} \end{cases} \quad \text{and} \quad z_a = \begin{cases} x_{e^+} - \alpha & \text{if } a = e^+ \\ x_{e^-} - \beta & \text{if } a = e^- \\ z'_a & \text{otherwise.} \end{cases} \tag{4.32}$$

Using that $\|y' - z'\|$ is arbitrarily small, we can show that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, contradicting the choice of x.

Case S5: Assume that $t_v > 0$ and $t_e = t_f = 0$. Let $H = (V \setminus \{u, w\}, F)$ be obtained from contracting e and f in G, and let $\vec{H}$ be its associated directed graph. Define the vectors $x' \in \mathbb{Q}_+^{F^+ \cup F^-}$, $l' \in \mathbb{Z}_+^F$ and $b' \in \mathbb{Z}^{V \setminus \{u,w\}}$ by $x'_a = x_a$ for all $a \in F^+ \cup F^-$, $l'_h = l_h$ for all $h \in F$, $b'_t = b_t$ for all $t \in V \setminus \{u, v, w\}$ and $b'_v = b_u + b_v + b_w$. Since H is windy postman ideal, x' is fractional, and $x' \in \mathcal{O}(H, l', b')$, there exist distinct vectors $y', z' \in \mathcal{O}(H, l', b')$ such that $x' = \frac{1}{2}(y' + z')$. We may choose y', z' so that $\|y' - z'\|$ is arbitrarily small. Define the vectors $y, z \in \mathbb{Q}^{E^+ \cup E^-}$ by $y_a = y'_a$ and $z_a = z'_a$ for all $a \in F^+ \cup F^-$, and, for all $a \in \{e^+, e^-, f^+, f^-\}$, let y_a and z_a be the unique solutions to the system of linear equations

$$\begin{aligned}
y_{e^+} + z_{e^+} &= 2x_{e^+} && (4.33)\\
y_{e^-} + z_{e^-} &= 2x_{e^-} && (4.34)\\
y_{f^+} + z_{f^+} &= 2x_{f^+} && (4.35)\\
y_{f^-} + z_{f^-} &= 2x_{f^-} && (4.36)\\
y(\vec{\delta}(\bar{u})) - y(\vec{\delta}(u)) &= b_u && (4.37)\\
y(\vec{\delta}(\bar{w})) - y(\vec{\delta}(w)) &= b_w && (4.38)\\
y(\vec{\delta}(T_e)) + y(\vec{\delta}(\bar{T}_e)) &= l(\delta(T_e)) + 1 && (4.39)\\
y(\vec{\delta}(T_f)) + y(\vec{\delta}(\bar{T}_f)) &= l(\delta(T_f)) + 1. && (4.40)
\end{aligned}$$

Using that $\|y' - z'\|$ is arbitrarily small, we can show that $y, z \in \mathcal{O}(G, l, b)$, $y \neq z$, and $x = \frac{1}{2}(y + z)$, a last contradiction to the choice of x. ■

Recall that if $G = (V, E)$ is not series-parallel, then it contains a subdivision $K = (W, F)$ of K_4. Define the vector $l \in \mathbb{Z}_+^E$ by $l_e = 1$ if $e \in F$, and $l_e = 0$ otherwise. Since $\mathcal{O}_{WPT}(K_4)$ is not an integral polyhedron, it follows that $\mathcal{O}_{WPT}(G, l)$ is also not an integral polyhedron.

Corollary 4.9 *Let $G = (V, E)$ be an undirected graph. Then G is series-parallel if and only if the polyhedron $\mathcal{O}_{WPT}(G, l)$ is integral for all $l \in \{0, 1\}^E$.*

We also obtain Win's conjecture as an easy corollary.

Corollary 4.10 *If G is series-parallel, then G is windy postman perfect.*

Using Theorems 4.1, 4.3, and 4.8, we can extend the class of undirected graphs known to be windy postman perfect.

Theorem 4.11 *Let $\mathcal{F}$ be the class of undirected graphs constructed as follows:*

1. *All graphs whose connected components are even, series-parallel, or $K_{3,3}$ are in $\mathcal{F}$.*
2. *Any graph obtained from graphs in $\mathcal{F}$ by performing any of the operations described in the statement of Theorem 4.3 is in $\mathcal{F}$.*

Then every undirected graph in $\mathcal{F}$ is windy postman perfect.

4.3.2 Windy Postman Perfect Signed Graphs

Win's operations suggest the study of graphs for which we have associated a *parity* to each edge. A *signed graph* is a pair (G, Σ) where $G = (V, E)$ is an undirected graph, and $\Sigma \subseteq E$. If $e \in \Sigma$ we say that e is *odd*; otherwise we say that it is *even*. Similarly, we say that a subset $F \subseteq E$ is *odd* if $|F \cap \Sigma|$ is odd; otherwise we say that it is *even*. A vector $l \in \mathbb{Z}_+^E$ is said to be *valid* if for every $e \in E$, l_e is odd if and only if $e \in \Sigma$. We say that (G, Σ) is *windy postman perfect* if $\mathcal{O}_{WPT}(G, l)$ is integral for all valid l.

We say that Σ' is a *signature* for (G, Σ) if (G, Σ) and (G, Σ') have the same family of odd cuts. In particular, if $C \subseteq E$ is the edge set of a cycle in G then $\Sigma \triangle C$ is a signature for (G, Σ). We call $(G, \Sigma \triangle C)$ the *resigning* of (G, Σ) along C. For $F \subseteq E$ we define $(G, \Sigma) / F$ to be the signed graph $(G / F, \Sigma \setminus F)$. For $F \subseteq E \setminus \Sigma$ we define $(G, \Sigma) \setminus F$

to be the signed graph $(G \setminus F, \Sigma)$. We call these two the *contraction* and *deletion* of F, respectively. We say that (G', Σ') is a *minor* of (G, Σ) if (G', Σ') can be obtained from (G, Σ) after a sequence of resignings, contractions and deletions. Observe that our minor operations are essentially *dual* to the usual minor operations for signed graphs.

Windy postman perfection of signed graphs is closed under taking minors.

Theorem 4.12 *Let (G, Σ) be a windy postman perfect signed graph.*

1. *If $e \in E \setminus \Sigma$, then $(G, \Sigma) \setminus e$ is windy postman perfect.*
2. *If $e \in E$, then $(G, \Sigma) \ / \ e$ is windy postman perfect.*
3. *If $C \subseteq E$ is a cycle, then $(G, \Sigma \triangle C)$ is windy postman perfect.*

Proof: Let e be an even edge, let $G' = G \setminus e$, and let $l' \in \mathbb{Z}_+^{E'}$ be valid for (G', Σ). Define the vector $l \in \mathbb{Z}_+^E$ by $l_f = l'_f$ if $f \neq e$, and $l_e = 0$. Since (G, Σ) is windy postman perfect and l is valid for (G, Σ) we have that $\mathcal{O}_{WPT}(G, l)$ is integral, and so is its face defined by $x_{e^+} = 0$ and $x_{e^-} = 0$. This implies the integrality of $\mathcal{O}_{WPT}(G', l')$.

Let $e \in E$ with ends u and v. Assume that $e^+ = (u, v)$ and $e^- = (v, u)$. Let $G' = G \ / \ e$, let $\Sigma' = \Sigma \setminus e$, and let $l' \in \mathbb{Z}_+^{E'}$ be valid for (G', Σ'). Define the vector $l \in \mathbb{Z}_+^E$ by $l_f = l'_f$ if $f \neq e$, and $l_e = 1$ if $e \in \Sigma$ or $l_e = 0$ if $e \notin \Sigma$. Let x' be an extreme point of $\mathcal{O}_{WPT}(G', l')$. For every $f \in E'$, let $x_{f^+} = x'_{f^+}$ and $x_{f^-} = x'_{f^-}$. In order to have $x \in \mathcal{O}_{WPT}(G, l)$, define x_{e^+} and x_{e^-} as follows: Let $z = x(\vec{\delta}(u) \setminus e^-) - x(\vec{\delta}(\bar{u}) \setminus e^+)$. Without loss of generality, we assume that $z \geq 0$. Let $\mathcal{W}$ be the set of odd cuts containing e. For $W \in \mathcal{W}$ let $y(W) = \sum_{f \in W \setminus e}(x_{f^+} + x_{f^-} - l_f)$. Let $y = \min_{W \in \mathcal{W}} y(W)$ if $\mathcal{W}$ is not empty; otherwise let $y = 1$. Assume first that $z \geq l_e$. If $y + z \geq 1 + l_e$ let $x_{e^+} = z$, $x_{e^-} = 0$; otherwise let $x_{e^+} = \frac{1}{2}(1 + l_e + z - y)$ and $x_{e^-} = \frac{1}{2}(1 + l_e - z - y)$. It is not difficult to show that x is an extreme point of $\mathcal{O}_{WPT}(G, l)$. Hence x and x' are integral. Now assume that $z < l_e$ (this can only happen if $l_e = 1$). If $y + z \geq 2$ let $x_{e^+} = \frac{1}{2}(1 + z)$ and $x_{e^-} = \frac{1}{2}(1 - z)$; otherwise let $x_{e^+} = \frac{1}{2}(2 + z - y)$ and $x_{e^-} = \frac{1}{2}(2 - z - y)$. Again, it is not difficult to show that x is an extreme point of $\mathcal{O}_{WPT}(G, l)$. Hence x and x' are integral.

Let $\Sigma' = \Sigma \triangle C$, let $l' \in \mathbb{Z}_+^E$ be valid for (G, Σ'), and let x' be an extreme point of $\mathcal{O}_{WPT}(G, l')$. Let C^+ be a cyclic orientation of C. We assume that, for every $e \in C$, e^+ is oriented as in C^+. Construct the vectors $l \in \mathbb{Z}_+^E$ and $x \in \mathbb{Q}_+^D$ as follows:

- If $e \notin C$ let $l_e = l'_e$, $x_{e^+} = x'_{e^+}$, and $x_{e^-} = x'_{e^-}$,
- else if $x'_{e^+} > 0$ let $l_e = l'_e + 1$, $x_{e^+} = x'_{e^+} + 1$, and $x_{e^-} = x'_{e^-}$,
- else if $l'_e \geq 1$ let $l_e = l'_e - 1$, $x_{e^+} = 0$ and $x_{e^-} = x'_{e^-} - 1$,
- else if $x'_{e^-} > 0$ let $l_e = 1$ and if $x'_{e^-} \geq 2$ let $x_{e^+} = 0$ and $x_{e^-} = x'_{e^-} - 1$; otherwise let $x_{e^+} = 1 - \frac{1}{2}x'_{e^-}$ and $x_{e^-} = \frac{1}{2}x'_{e^-}$,
- else let $l_e = 1$, $x_{e^+} = 1$ and $x_{e^-} = 0$.

It is not difficult to show that l is valid for (G, Σ) and that x is an extreme point of $\mathcal{O}_{WPT}(G, l)$. Hence x and x' are integral. ■

Since K_4 is not windy postman perfect, it follows that the signed graph $K_4^6 = (K_4, E(K_4))$ is also not windy postman perfect. Another signed graph that is not windy postman perfect is $K_4^1 = (K_4, \{e\})$, where e is an edge of K_4. We exhibit a corresponding fractional extreme point in Figure 4.3. The lower bounds are shown on the right-hand side.

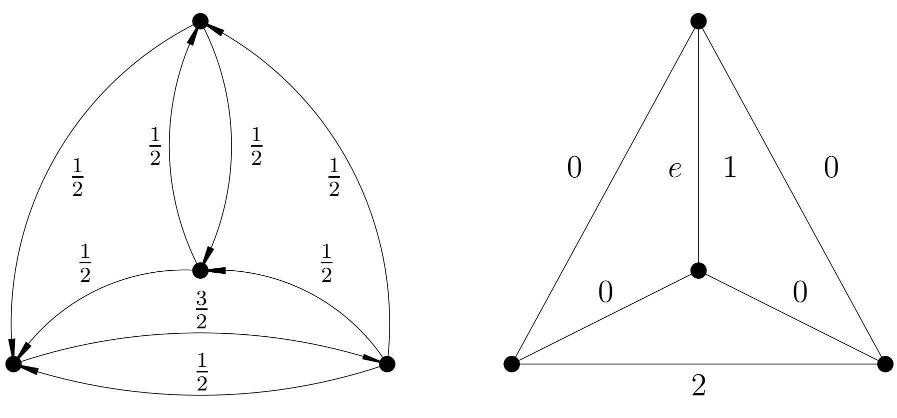

Figure 4.3: The signed graph K_4^1 and a fractional extreme point.

We believe that K_4^1 and K_4^6 are the only obstructions to windy postman perfection of signed graphs. The following would be a generalization of Theorem 4.8.

Conjecture 4.13 *A signed graph (G, Σ) is windy postman perfect if and only if (G, Σ) does not have a K_4^1 nor a K_4^6 minor.*

4.3.3 Windy Postman Perfect Grafts

A *graft* is a pair (G, T) where $G = (V, E)$ is an undirected graph, and $T \subseteq V$. If $v \in T$ we say that v is *odd*; otherwise we say that it is *even*. Similarly, we say that a subset $S \subseteq V$ is *odd* if $|S \cap T|$ is odd; otherwise we say that it is *even*. Grafts were introduced by Seymour [81]. Some reductions of grafts, called *splits*, were introduced by Gerards [43] in order to give an interpretation of Seymour's decomposition theorem for regular matroids.

A vector $l \in \mathbb{Z}_+^E$ is *valid* if for every $v \in V$, $l(\delta(v))$ is odd if and only if $v \in T$. We say that the graft (G, T) is *windy postman perfect* if $\mathcal{O}_{WPT}(G, l)$ is integral for each valid l.

Let (G, Σ) be a signed graph, let $T = \{v \in V : \delta(v) \text{ is odd}\}$, and consider the graft (G, T). Note that if $l \in \mathbb{Z}_+^E$ is valid for (G, Σ), then l is also valid for (G, T). Moreover, if Σ' is any signature for (G, Σ), then any valid $l \in \mathbb{Z}_+^E$ for (G, Σ') is valid for (G, T). The converse is also true: If $l \in \mathbb{Z}_+^E$ is valid for (G, T), then the set $\Sigma' = \{e \in E : l_e \text{ is odd}\}$ is a signature for (G, Σ). This implies a strong relationship between windy postman perfection of grafts and of signed graphs.

Lemma 4.14 *Let (G, Σ) be a signed graph and let (G, T) be the graft defined above. Then (G, Σ) is windy postman perfect if and only if (G, T) is windy postman perfect.*

For $e \in E$ we define $(G, T) \setminus e$ to be the graft $(G \setminus e, T)$, and $(G, T) \,/\, e$ to be the graft $(G \,/\, e, T_e)$, where T_e is defined as follows: Let u, v be the ends of e in G and let w be the newly created vertex in $G \,/\, e$. Then $T_e = (T \setminus \{u, v\}) \cup \{w\}$ if $|T \cap \{u, v\}|$ is odd, and $T_e = T \setminus \{u, v\}$ otherwise. We call these two the *deletion* and the *contraction* of e, respectively. We say that (G', T') is a *minor* of (G, T) if (G', T') can be obtained from (G, T) after a sequence of edge contractions and edge deletions. Theorem 4.12 and Lemma 4.14 imply that windy postman perfection of grafts is closed under taking minors.

Theorem 4.15 *Let (G, T) be a windy postman perfect graft, and let $e \in E(G)$. Then $(G, T) \setminus e$ and $(G, T) \,/\, e$ are also windy postman perfect.*

Since the signed graphs K_4^6 and K_4^1 are not windy postman perfect, Lemma 4.14 implies that the grafts $K_4^4 = (K_4, V(K_4))$ and $K_4^2 = (K_4, \{u, v\})$ (where u and v are two distinct vertices of K_4) are also not windy postman perfect. In fact, Lemma 4.14 and Theorem 4.15 imply that the following statement is equivalent to Conjecture 4.13.

Conjecture 4.16 *A graft (G, T) is windy postman perfect if and only if (G, T) does not have a K_4^2 nor a K_4^4 minor.*

Let (G, T) be a graft. If any component G' of G has $|V(G') \cap T|$ odd, there does not exist a valid $l \in \mathbb{Z}_+^E$ for (G, T). Therefore, (G, T) is windy postman perfect. In the rest of this section we assume that every component G' of G has $|V(G') \cap T|$ even.

If G is disconnected, with V_1 the vertex set of one of its components, then $(G_1, T_1) = (G[V_1], T \cap V_1)$ and $(G_2, V_2) = (G[V \setminus V_1], T \setminus T_1)$ form a 1-*split* of (G, T). If G is connected, with a cut vertex v, then (G_1, T_1) and (G_2, T_2) form a 1-*split* of (G, T), where $E(G_1)$ and $E(G_2)$ form a partition of $E(G)$ induced by v, T_1 is defined as $T \setminus V(G_2)$ if $|T \cap V(G_2)|$ is even, or $(T \setminus V(G_2)) \cup \{v\}$ otherwise. T_2 is defined similarly. (G_1, T_1) and (G_2, T_2) are the *parts* of the 1-split. The following can be proved in a similar way to Lemmas 4.6 and 4.7.

Lemma 4.17 *Let (G, T) have a 1-split with parts (G_1, T_1) and (G_2, T_2). Then (G, T) is windy postman perfect if and only if (G_1, T_1) and (G_2, T_2) are windy postman perfect.*

If G has a 2-vertex cut $\{u, v\}$ with sides G_1 and G_2, neither of which consists of a two edge path with middle vertex in T, then (G_1', T_1') and (G_2', T_2') form a 2-*split* of (G, T), where (G_1', T_1') is defined as follows: If $T \subseteq V(G_1)$, then $V(G_1') = V(G_1)$, $E(G_1') = E(G_1) \cup \{uv\}$, and $T_1' = T$. If $T \setminus V(G_1)$ is not empty, then $V(G_1') = V(G_1) \cup \{w\}$ (where w is a new vertex) and $E(G_1') = E(G_1) \cup \{uw, wv\}$. Moreover $T_1' = (T \cap V(G_1)) \cup \{w\}$ if $|T \setminus V(G_1)|$ is odd, and $T_1' = (T \cap V(G_1)) \triangle \{u, w\}$ otherwise. Observe that this definition guarantees that there will be an even number of vertices in T_1'. (G_2', T_2') is defined in a similar way. (G_1', T_1') and (G_2', T_2') are the *parts* of the 2-split. See Figure 4.4 for an example where T consists of the odd-degree vertices of G, and T_1', T_2' are obtained as described above. Vertices in T, T_1', T_2' are drawn as squares, whereas other vertices are drawn as circles.

The following property of splits is straightforward to verify.

Lemma 4.18 *Let (G, T) have a 1-split or a 2-split with parts (G_1', T_1') and (G_2', T_2'). Then (G, T) has a K_4^2 or a K_4^4 minor if and only if at least one of (G_1', T_1') and (G_2', T_2') does.*

We close this chapter by giving a characterization of grafts with no K_4^2 nor K_4^4 minor. This characterization depends on the following technical result.

Lemma 4.19 *Let $G = (V, E)$ be a connected undirected graph, and let $T \subseteq V$ with $|T|$ even. If (G, T) has neither a K_4^2 nor a K_4^4 minor, then one of the following holds:*

1. *T is empty.*

2. *G is series-parallel.*

3. *(G, T) has a* 1*-split or a* 2*-split.*

Proof: For a contradiction, we assume that none of the above three statements hold. Then G is 2-vertex-connected and has a K_4 minor, and $|T| \geq 2$. Furthermore, if $\{u, v\}$ is a 2-vertex cut with sides G_1 and G_2, then exactly one of them consists of a two edge path with middle vertex in T. Observe that contracting one of these two edges does not decrease the vertex connectivity, nor changes the property of having a K_4 minor. Choose one of these two edges, say e, such that it maximizes the number of odd vertices in $(G', T') \equiv (G, T) / e$. We claim that $|T'| \geq 2$: If one of u or v is not odd and we contract the edge incident to it, then $|T'| = |T| \geq 2$. Else, if both u and v are odd and we contract either edge, then $|T| \geq 4$ and $|T'| = |T| - 2 \geq 2$. If G' has two parallel edges, delete one of them from (G', T'). Redefine (G, T) to be (G', T'), and continue this process until G is 3-vertex-connected.

We claim that (G, T) has a K_4^2 or a K_4^4 minor. Since G has a K_4 minor, it has at least four vertices. Let u and v be two odd vertices. By Menger's Theorem 1.1, u and v are joined by three internally-disjoint paths P_1, P_2, and P_3. Since G has no parallel edges, two of these paths, say P_2 and P_3, have internal vertices r and s, respectively. Since $G \setminus \{u, v\}$ is connected, it contains a path P_4 joining r and s. Choose P_1, P_2, P_3, r and s in such a way that P_4 is as short as possible. It follows that the edges of P_1, P_2, P_3 and P_4 induce a subdivision $K = (V_K, E_K)$ of K_4. Let $T_K = T \setminus V_K$. For each $t \in T_K$, let P_t be a shortest path, not containing u nor v, that joins t to a vertex in V_K. P_t exists since there are three internally-disjoint paths from t to r, and at least one of them does not contain u nor v. Let $E_P = \cup_{t \in T_K} P_t$, and let $E_D = E \setminus (E_K \cup E_P)$. Let $(G', T') = (G, T) \setminus E_D / E_P$. Observe that G' is isomorphic to K, and that T' still contains u and v. Let (K_4, T'') be obtained from (G', T') by contracting all edges in E_K, except for the three edges incident to u, and one edge on each of the paths joining r to s, s to v, and v to r. See Figure 4.5, where we show (G, T) after deleting E_D, but before contracting the edges in E_P (bold edges), and the edges in E_K (dashed bold edges). Since $u \in T''$, it follows that (K_4, T'') is either a K_4^2 or a K_4^4 minor of (G, T). ■

Lemmas 4.18 and 4.19 imply the desired characterization.

Theorem 4.20 *Let (G, T) be a graft with G connected and $|T|$ even.*

1. *If (G, T) has neither a 1-split nor a 2-split, then (G, T) has neither a K_4^2 nor a K_4^4 minor if and only if G is series-parallel or T is empty.*

2. *If (G, T) has a 1-split or a 2-split, then (G, T) has neither a K_4^2 nor a K_4^4 minor if and only if none of the parts of the split does.*

We believe that a statement similar to Lemma 4.17 holds for 2-splits. Together with Theorem 4.20, this would immediately imply Conjectures 4.13 and 4.16.

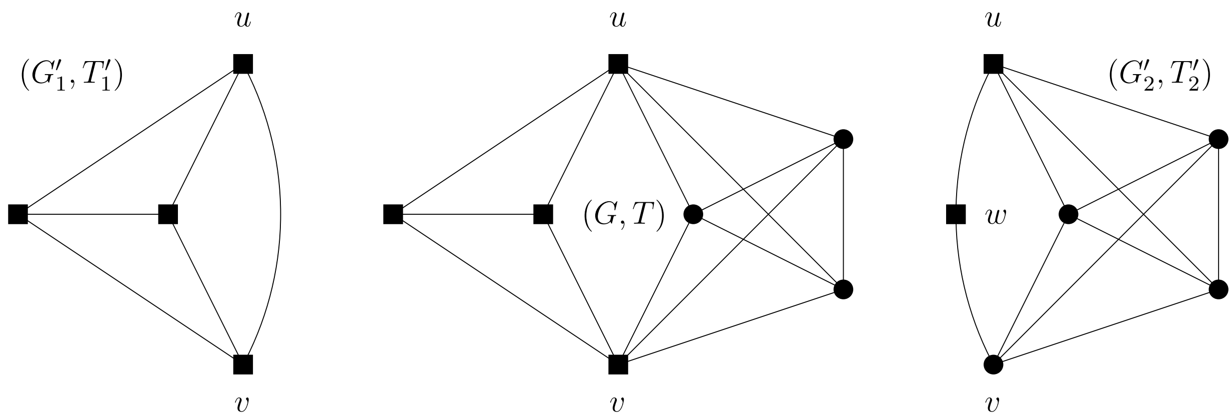

Figure 4.4: A 2-split and its parts.

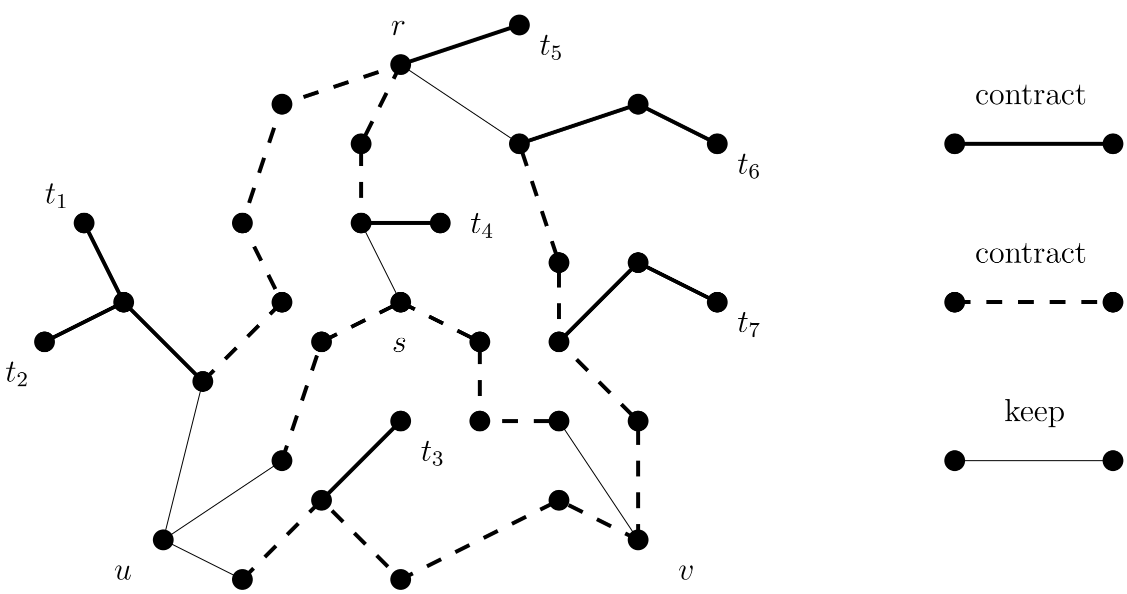

Figure 4.5: The graft (G, T) after deleting E_D.

Chapter 5

The Edges Postman Problem

Now, here, you see, it takes all the running you can do, to keep in the same place.
If you want to get somewhere else, you must run at least twice as fast as that!

Through the Looking Glass, Lewis Carroll

We introduce a special case of Minimum Restricted Mixed Postman Tour, called Minimum Edges Postman Tour. It was conjectured that this problem is $\mathcal{NP}$-hard, and we prove that this is the case, even when the input is restricted to be planar. We study a linear relaxation of Minimum Edges Postman Tour; in particular, we give a class of valid inequalities for its integral solutions, and a class of integral extreme points of the polyhedron it defines. We present four approximation algorithms for Minimum Edges Postman Tour, the best of which has a guarantee of $\frac{4}{3}$ for the cost of an optimal tour, and a guarantee of 2 for the cost of an optimal postman set.

5.1 Introduction

At the end of Chapter 3, we defined Minimum Restricted Mixed Postman Tour, and we proposed the study of two special cases. The first special case that we study is when the restricted set of the input mixed graph $M = (V, E, A)$ coincides with its arc set. We say that a postman tour of M is an *edges postman tour* if it uses each arc of M exactly once. We say that a family F of edges is an *edges postman set* of M if there exists an edges postman tour of M using each edge e of M once more than the number of times e appears in F. We show in Figure 5.1 a mixed graph and one of its edges postman tours.

$$(u, e, v, b, w, f, u, e, v, b, w, c, x, d, z, g, v, b, c, x, h, y, i, z, i, y, a, u). \tag{5.1}$$

Observe that edge b must be used more than twice and, in this tour, edge i has been traversed in two different directions. We define the problems that we study in this chapter.

Problem: EDGES POSTMAN TOUR.
Input: A mixed graph $M = (V, E, A)$.
Output: Does M have an edges postman tour?

Problem: MINIMUM EDGES POSTMAN TOUR.
Input: A strongly connected mixed graph $M = (V, E, A)$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost $\text{MEPT}(M, c)$ of an edges postman tour of M.

Problem: MINIMUM EDGES POSTMAN SET.
Input: A strongly connected mixed graph $M = (V, E, A)$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost $\text{MEPSP}(M, c)$ of an edges postman set of M.

Note that since arcs cannot be replicated, we do not assign them a cost (or consider their cost to be zero) in the minimization version of the problem. Furthermore, we can easily remove the arcs from its description. Let $G = (V, E)$, let $\vec{G} = (V, E^+ \cup E^-)$ be its associated directed graph and, for each $v \in V$, let $b_v = d_A(v) - d_A(\bar{v})$ be the *demand* at vertex v. Then the problem of finding an edges postman tour on the mixed graph M is equivalent to the problem of finding a feasible flow x on the directed graph $\vec{G}$, with vector of demands b and vector of lower bounds $\mathbf{0}$, such that $x_{e^+} + x_{e^-} \geq 1$ for all $e \in E$. See Figure 5.2 for an example. Hence, we can reformulate the above three problems as follows.

Problem: EDGES POSTMAN TOUR.
Input: An undirected graph $G = (V, E)$, and a vector $b \in \mathbb{Z}^V$.
Output: Does (G, b) have an edges postman tour?

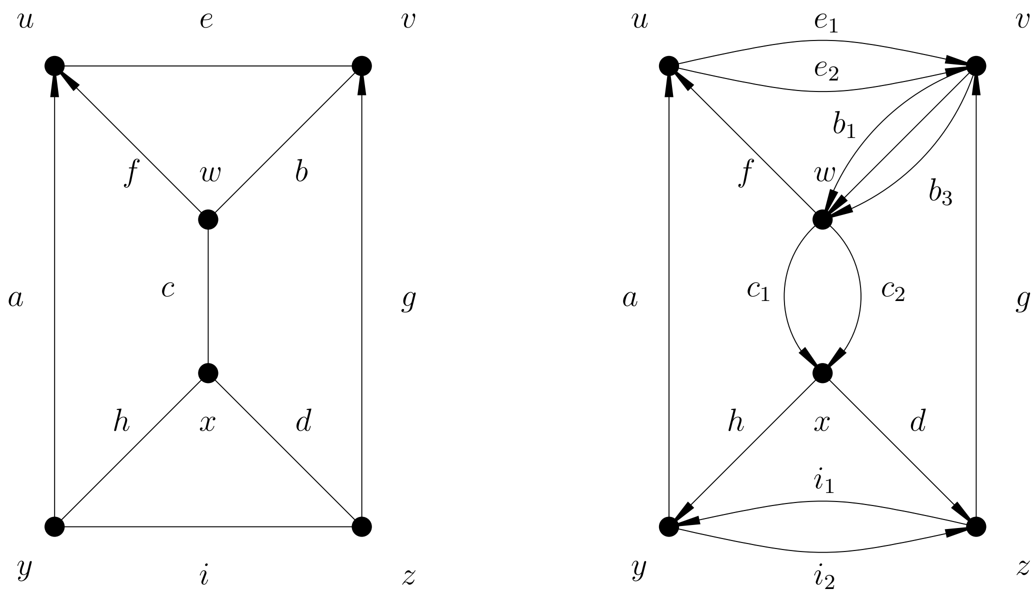

Figure 5.1: A mixed graph and an edges postman tour.

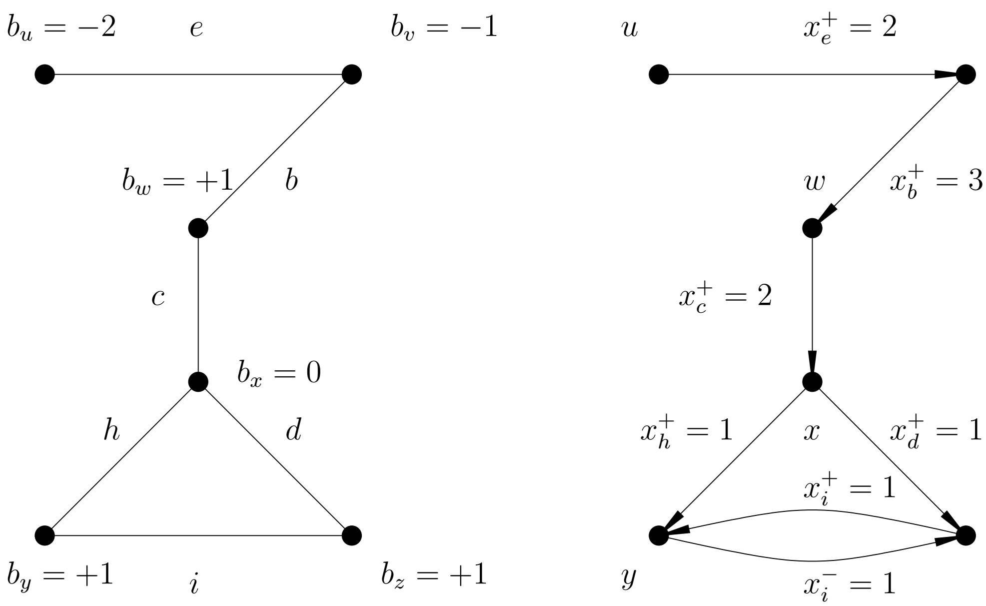

Figure 5.2: A feasible flow on an undirected graph.

Problem: MINIMUM EDGES POSTMAN TOUR.
Input: An undirected graph $G = (V, E)$, a vector $b \in \mathbb{Z}^V$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost MEPT(G, b, c) of an edges postman tour of (G, b).

Problem: MINIMUM EDGES POSTMAN SET.
Input: An undirected graph $G = (V, E)$, a vector $b \in \mathbb{Z}^V$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost MEPSP(G, b, c) of an edges postman set of (G, b).

From this alternate formulation, it is easy to see that (G, b) is a yes instance of EDGES POSTMAN TOUR if and only if the vertex set S of each connected component of G satisfies $b(S) = 0$. Equivalently, M is a yes instance of EDGES POSTMAN TOUR if and only if the vertex set S of each connected component of G satisfies $d_A(S) = d_A(\bar{S})$. A natural generalization of the above problems is to require that each edge $e \in E$ is traversed at least l_e and at most u_e times ($u_e \geq l_e \geq 0$). Such an edges postman tour is said to be *bounded*.

Problem: MINIMUM BOUNDED EDGES POSTMAN TOUR.
Input: An undirected graph $G = (V, E)$, vectors $l, u \in \mathbb{Z}_+^E$ with $l \leq u$, a vector $b \in \mathbb{Z}^V$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost MBEPT(G, l, u, b, c) of a bounded edges postman tour of (G, l, u, b).

5.2 Computational Complexity

In his doctoral thesis, Veerasamy conjectured that MINIMUM EDGES POSTMAN TOUR is $\mathcal{NP}$-hard [85]. In this section we prove that the decision version of MINIMUM BOUNDED EDGES POSTMAN TOUR is $\mathcal{NP}$-complete, and then we remove the upper bounds to prove that the decision version of MINIMUM EDGES POSTMAN TOUR is also $\mathcal{NP}$-complete.

Finally, we remove the non-planarity from our proof to show that MINIMUM EDGES POSTMAN TOUR remains $\mathcal{NP}$-complete even if we restrict the input to be planar. Our reductions are from the $\mathcal{NP}$-complete problems 1-IN-3 SATISFIABILITY and PLANAR 1-IN-3 SATISFIABILITY (Theorems 1.4 and 1.6). Recall that in 1-IN-3 SATISFIABILITY a clause is *satisfied* if and only if exactly one of its three literals is true.

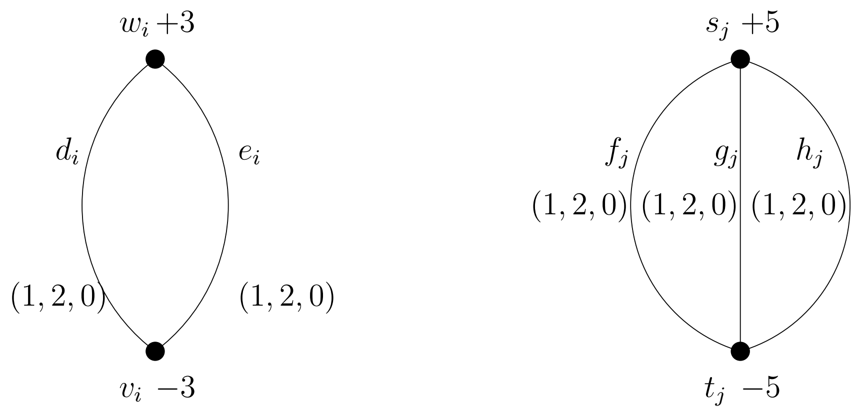

Figure 5.3: The variable and clause subgraphs.

For each variable x_i, consider the *variable* subgraph on the left-hand side of Figure 5.3, with vertices v_i and w_i, and edges d_i and e_i. (In this section, all figures indicate the demands at the vertices, and the bounds and costs of the edges as the triple (l, u, c).) The only two feasible solutions carry one unit of flow from v_i to w_i on d_i and two on e_i, or vice versa. For each clause C_j, consider the *clause* subgraph on the right-hand side of Figure 5.3, with vertices s_j and t_j, and edges f_j, g_j, and h_j. The only three feasible solutions carry one unit of flow from t_j to s_j on one of f_j, g_j, and h_j, and two units on the other two. We describe now a subgraph used to interconnect the variable to the clause subgraph.

Lemma 5.1 *Consider the* negator *subgraph N in Figure 5.4 and assume that the demands at vertices w, x, y, z satisfy $b_w, b_x \in \{1, 2\}$, $b_y, b_z \in \{-1, -2\}$, and $b_w + b_x + b_y + b_z = 0$. Then the minimum cost of an edges postman tour of (N, l, u, b, c) is 3 if $b_w = 1$, $b_x = 2$, $b_y = -1$, and $b_z = -2$, or if $b_w = 2$, $b_x = 1$, $b_y = -2$, and $b_z = -1$. For other values of the demands, the cost is greater, or the problem is infeasible.*

Proof. First we show that in any feasible solution of N, the demands at w, x, y, z must satisfy $b_y = -b_w$ and $b_z = -b_x$. By symmetry, it is enough to show the former. Since $b_p = 0$ and $l_{pq} = u_{pq} = 1$, it follows that in any feasible solution of N, one of *op* or *pw* carries one unit of flow, and the other carries two units of flow. Similarly, one of *op* and *oy* carries one unit of flow, and the other carries two units of flow. Hence, there are only two

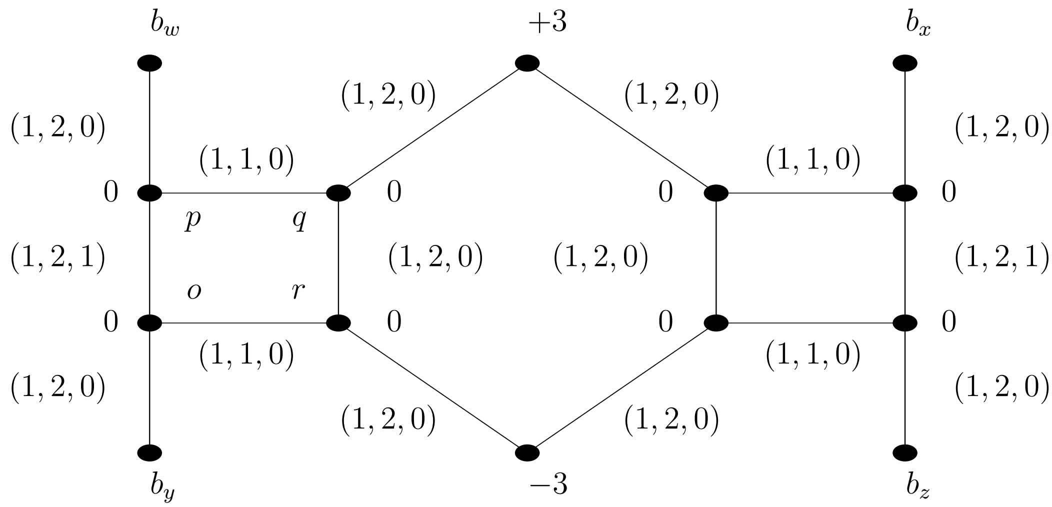

Figure 5.4: The negator subgraph.

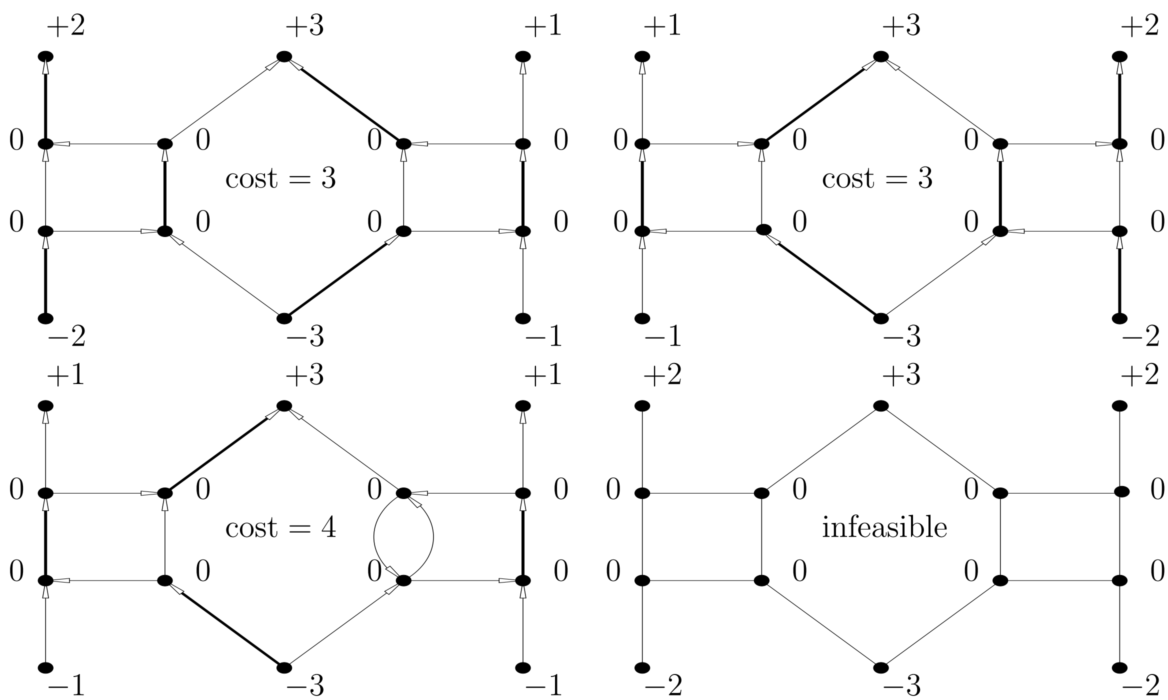

Figure 5.5: Edges postman tours of the negator subgraph.

possibilities: Either op carries one unit of flow, or both oy and pw carry one unit of flow. In the former case $b_w = -b_y = 2$, and in the latter case $b_w = -b_y = 1$.

To finish the proof, it is enough to find optimal solutions for the four cases that result from the restrictions on the demands at w, x, y, z. We show these solutions in Figure 5.5. Bold arcs carry two units of flow. The other arcs carry one unit of flow. ■

Theorem 5.2 *The decision version of* MINIMUM BOUNDED EDGES POSTMAN TOUR *is* $\mathcal{NP}$*-complete, even if all bounds are finite.*

Proof. We reduce the $\mathcal{NP}$-complete problem 1-IN-3 SATISFIABILITY to the decision version of MINIMUM BOUNDED EDGES POSTMAN TOUR. Let I be an instance of 1-IN-3 SATISFIABILITY with n variables $x_1, x_2, \ldots, x_n$ and m clauses $C_1, C_2, \ldots, C_m$, where each clause C_j contains three literals y_j^1, y_j^2, y_j^3. Construct an instance (G, l, u, b, c) of MINIMUM BOUNDED EDGES POSTMAN TOUR consisting of n copies of the variable subgraph and m copies of the clause subgraph, interconnected with negator subgraphs as follows.

For each clause C_j, identify its literals y_j^1, y_j^2, y_j^3 with the edges f_j, g_j, h_j, respectively. If literal y_j^1 is a positive variable (say x_i), then identify it with edge d_i. Otherwise, if literal y_j^1 is a negative variable (say $\neg x_i$), then identify it with edge e_i. Split each of the two edges identified with literal y_j^1 into three parts, and connect these two paths of length three with a negator subgraph. Repeat this procedure with the other literals of C_j. Note that in this process, the edges of the variable subgraphs may be split many times. See Figure 5.6.

We claim that I is satisfiable if and only if (G, l, u, b, c) has a bounded edges postman tour of cost at most $9m$. Assume that I is satisfiable, and let $x_1^*, x_2^*, \ldots, x_n^*$ be an assignment of the variables satisfying I. For all $1 \leq i \leq n$, if x_i^* is true, send a unit of flow from v_i to w_i through d_i, otherwise, send it through e_i. For all $1 \leq j \leq m$, send a unit of flow from t_j to s_j through the edges identified with the false literals in C_j. By Lemma 5.1, it is possible to send flow through each negator subgraph with a cost of 3, for a total cost of $9m$. Conversely, if we have a feasible solution for (G, l, u, b, c) with cost at most $9m$, by Lemma 5.1, it must have cost exactly $9m$ because each negator subgraph involves a cost of 3. Moreover, if we set x_i^* to true if and only if a unit of flow is sent from v_i to w_i through d_i, we obtain an assignment of the variables that satisfies I. ■

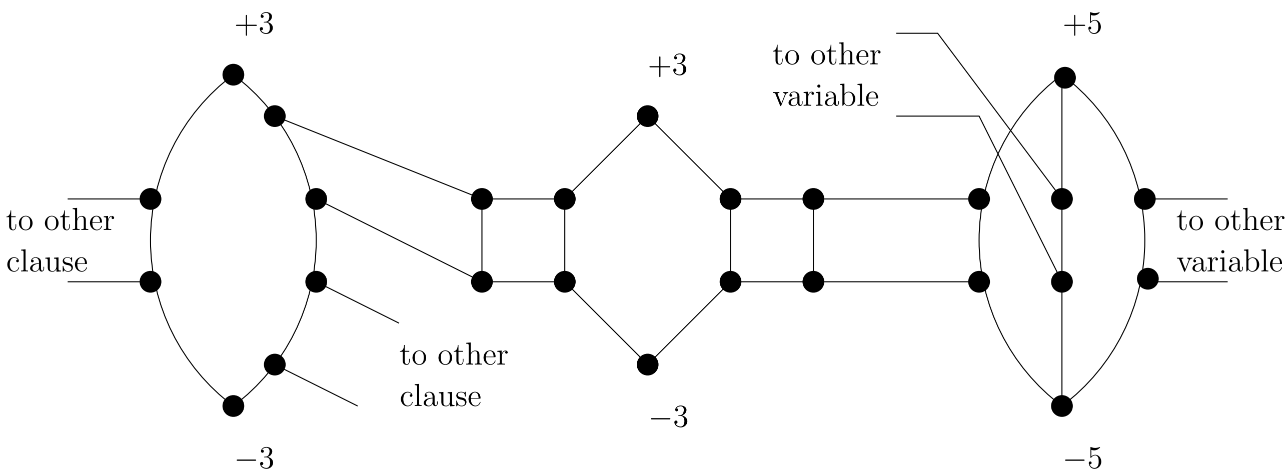

Figure 5.6: Connecting a variable to a clause with a negator.

Our next step is to remove the upper bounds.

Corollary 5.3 *The decision version of* MINIMUM EDGES POSTMAN TOUR *is* $\mathcal{NP}$*-complete.*

Proof. Consider the instance of MINIMUM BOUNDED EDGES POSTMAN TOUR used in the proof of Theorem 5.2. Remove the upper bounds, and set the cost of the edges to zero everywhere, except on the two *squares* of each negator subgraph, where we set the cost of the edges to one. The variable subgraph and the clause subgraph satisfy the same properties as before. Subject to the conditions of Lemma 5.1, each negator subgraph can be traversed with cost at most 10 if and only if $b_w = 1$, $b_x = 2$, $b_y = -1$, and $b_z = -2$ or $b_w = 2$, $b_x = 1$, $b_y = -2$, and $b_z = -1$. For other values of the demands, the cost is at least 11. The corresponding optimal solutions are the same as in Figure 5.5, except that the infeasible case becomes feasible, with a cost of 11. The same argument as in Theorem 5.2, with $9m$ replaced by $30m$, proves the result. ■

The following result can be proved in a similar way to Theorem 3.5, with $w = 30m$ for MINIMUM EDGES POSTMAN TOUR, and $w = 6m$ for MINIMUM EDGES POSTMAN SET.

Corollary 5.4 *There is no fully polynomial approximation scheme for* MINIMUM EDGES POSTMAN TOUR*, nor for* MINIMUM EDGES POSTMAN SET*, unless* $\mathcal{P} = \mathcal{NP}$*.*

In order to give hardness results for the planar case of MINIMUM EDGES POSTMAN TOUR, we need a slight modification of the variable subgraph.

Theorem 5.5 *The decision version of* MINIMUM BOUNDED EDGES POSTMAN TOUR *remains $\mathcal{NP}$-complete even if G is restricted to be planar and all bounds are finite.*

Proof. We reduce PLANAR 1-IN-3 SATISFIABILITY to the decision version of MINIMUM BOUNDED EDGES POSTMAN TOUR. Let I be an instance of PLANAR 1-IN-3 SATISFIABILITY with n variables $x_1, x_2, \ldots, x_n$ and m clauses $C_1, C_2, \ldots, C_m$, where each clause C_j contains three literals y_j^1, y_j^2, y_j^3. For each $1 \leq i \leq n$, let p_i and q_i be the number of times the literals x_i and $\neg x_i$ appear in the clauses, respectively, and let $t_i = \max\{p_i, q_i\}$.

Let H be the bipartite undirected graph associated with I (see Figure 1.5 and the explanation that precedes it). Since H is planar, we can find a planar embedding of H in polynomial time. Using this embedding, we construct an undirected graph G, essentially replacing each variable vertex of H with a *new* variable subgraph, each clause vertex of H with a clause subgraph, and each edge of H with a negator subgraph.

For each variable x_i, the new variable subgraph is a cycle X_i of length $4t_i$, whose edges have $l = 1$, $u = 2$, and $c = 0$, and whose vertices have demands either $+1$ or -1, alternating around X_i. Label the edges of X_i from e_1 to e_{4t_i} around X_i, and associate the odd edges with the literal x_i, and the even edges with the literal $\neg x_i$. Observe that each X_i has only two feasible solutions: All even edges carrying two units of flow from a vertex with demand $+1$ to a vertex with demand -1 and all odd edges carrying one unit of flow from a vertex with demand -1 to a vertex with demand $+1$, and vice versa.

Following the embedding of H, connect these variable subgraphs to the clause subgraphs as in the proof of Theorem 5.2. Observe that, in this process, many edges of the variable subgraphs are left unused. If two consecutive edges are unused we may contract them. See Figure 5.7, where the bold edges with square ends represent the negator subgraphs. Note that some crossings have appeared, but only between edges with $l = 1$, $u = 2$, and edges with $l = u = 1$. These crossings can be removed simply by transforming them into vertices. The rest of the proof is similar to that of Theorem 5.2. ■

Corollary 5.6 *The decision version of* MINIMUM EDGES POSTMAN TOUR *remains $\mathcal{NP}$-complete even if G is restricted to be planar.*

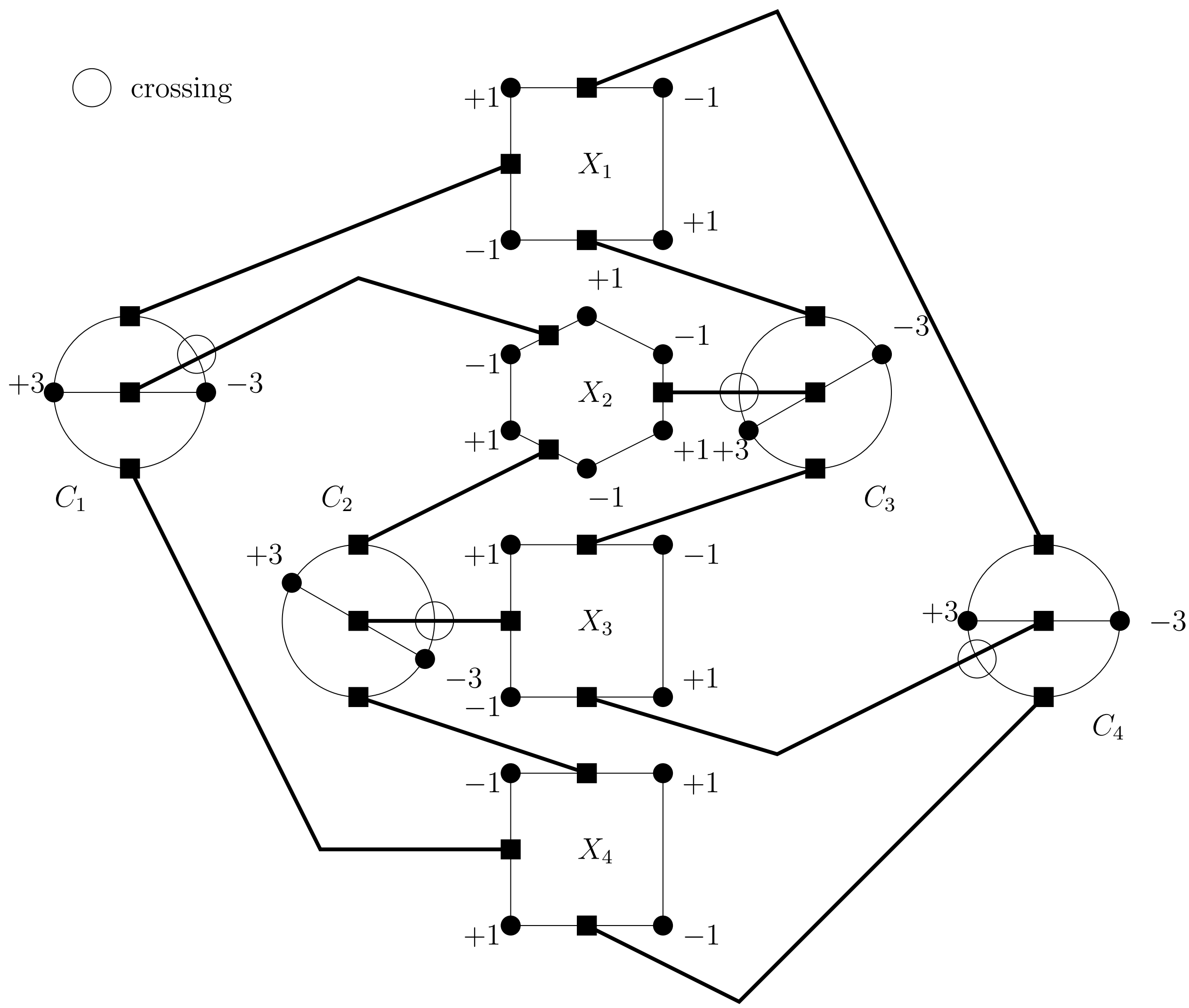

Figure 5.7: A planar instance of MINIMUM EDGES POSTMAN TOUR.

5.3 Integer Programming Formulations

We give two integer programming formulations for MINIMUM EDGES POSTMAN TOUR and MINIMUM BOUNDED EDGES POSTMAN TOUR based on those we gave for MINIMUM MIXED POSTMAN TOUR in Section 3.3, as well as their linear programming relaxations. Let $M = (V, E, A)$ be a strongly connected mixed graph, let $c \in \mathbb{Q}_+^E$, let $G = (V, E)$, and let $b_v = d_A(v) - d_A(\bar{v})$ for every $v \in V$.

5.3.1 First Formulation

For the first integer programming formulation, let $\vec{G} = (V, E^+ \cup E^-)$ be the associated directed graph of G. As before, we obtain the following integer program for MINIMUM EDGES POSTMAN TOUR.

$$\begin{aligned} \text{MEPT1}(G, b, c) &= \min c^\top x_{E^+} + c^\top x_{E^-} && (5.2) \\ \text{subject to} & && \\ x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) &= b_v \text{ for all } v \in V && (5.3) \\ x_{e^+} + x_{e^-} &\geq 1 \text{ for all } e \in E && (5.4) \\ x_e &\geq 0 \text{ and integer for all } e \in E^+ \cup E^-. && (5.5) \end{aligned}$$

Denote its linear programming relaxation by LMEPT1(G, b, c).

We can obtain an integer programming formulation MBEPT1(G, l, u, b, c) for MINIMUM BOUNDED EDGES POSTMAN TOUR, as well as its linear programming relaxation LMBEPT1(G, l, u, b, c), replacing the constraint (5.4) by

$$u_e \geq x_{e^+} + x_{e^-} \geq l_e \text{ for all } e \in E. \qquad (5.6)$$

Let $\mathcal{P}_{EPT}^1(G, b)$ be the convex hull of the feasible solutions to the integer program MEPT1(G, b, c), and let $\mathcal{Q}_{EPT}^1(G, b)$ be the set of feasible solutions to its linear programming relaxation LMEPT1(G, b, c). Similarly, let $\mathcal{P}_{BEPT}^1(G, l, u, b)$ be the convex hull of the feasible solutions to MBEPT1(G, l, u, b, c), and let $\mathcal{Q}_{BEPT}^1(G, l, u, b)$ be the set of feasible solutions to LMBEPT1(G, l, u, b, c). By Theorem 3.13, both $\mathcal{Q}_{EPT}^1(G, l, u, b)$ and $\mathcal{Q}_{BEPT}^1(G, b)$ are half-integral polyhedra.

5.3.2 Second Formulation

The second integer programming formulation we give is again based on Ford and Fulkerson's characterization of mixed Eulerian graphs.

$$\begin{aligned} \text{MEPT2}(G,b,c) &= \min c^\top x && (5.7)\\ \text{subject to} & && \\ x(\delta(S)) &\geq b(S) \text{ for all } S \subseteq V && (5.8)\\ x(\delta(v)) &\equiv b_v \pmod 2 \text{ for all } v \in V && (5.9)\\ x_e &\geq 1 \text{ for all } e \in E && (5.10)\\ x_e & \quad \text{integral for all } e \in E. && (5.11) \end{aligned}$$

Its linear programming relaxation is obtained by deleting the parity constraints (5.9) and the integrality constraints (5.11), and it is denoted by LMEPT2(G,b,c).

We can obtain an integer programming formulation MBEPT2(G,l,u,b,c) for Minimum Bounded Edges Postman Tour, as well as its linear programming relaxation LMBEPT2(G,l,u,b,c), replacing the constraints (5.10) by $u_e \geq x_e \geq l_e$ for all $e \in E$.

Let $\mathcal{P}^2_{EPT}(G,b)$ be the convex hull of the feasible solutions to the integer program MEPT2(G,b,c), and let $\mathcal{Q}^2_{EPT}(G,b)$ be the set of feasible solutions to its linear programming relaxation LMEPT2(G,b,c). Similarly, let $\mathcal{P}^2_{BEPT}(G,l,u,b)$ be the convex hull of the feasible solutions to MBEPT2(G,l,u,b,c), and let $\mathcal{Q}^2_{BEPT}(G,l,u,b)$ be the set of feasible solutions to LMBEPT2(G,l,u,b,c).

5.4 Linear Programming Relaxations

In this section, we study some properties of the linear programming relaxations of Minimum Edges Postman Tour and Minimum Bounded Edges Postman Tour. We give some valid inequalities for the sets of feasible solutions of these two problems. We also introduce a relaxation of Minimum Bounded Edges Postman Tour, that we call Minimum b-Join, for which we give a linear programming formulation. In this section, we assume that $u_e = \infty$ for all $e \in E$, and hence we refer to MBEPT1(G,l,b,c) instead of MBEPT1(G,l,u,b,c), $\mathcal{Q}^1_{BEPT}(G,l,b)$ instead of $\mathcal{Q}^1_{BEPT}(G,l,u,b)$, etc.

5.4.1 Odd-Cut Constraints

We can easily see that the polyhedra $\mathcal{Q}^1_{BEPT}(G, l, b)$ and $\mathcal{Q}^2_{BEPT}(G, l, b)$ have extreme points that do not correspond to feasible solutions of MINIMUM BOUNDED EDGES POSTMAN TOUR, for example when G consists of only one edge e, $l_e = 1$ and $b = \mathbf{0}$.

We say that $S \subseteq V$ is an *odd set*, and that $\delta(S)$ is an *odd cut* if $b(S) + l(\delta(S))$ is odd. We also say that $v \in V$ is *odd* if the set $\{v\}$ is odd. It is easy to see that an integral solution x to MBEPT1(G, l, b, c) must satisfy the *odd-cut* constraints

$$x(\vec{\delta}(S)) + x(\vec{\delta}(\bar{S})) \geq l(\delta(S)) + 1 \text{ for each odd set } S \subseteq V, \tag{5.12}$$

and an integral solution x to MBEPT2(G, l, b, c) must satisfy the odd-cut constraints

$$x(\delta(S)) \geq l(\delta(S)) + 1 \text{ for each odd set } S \subseteq V. \tag{5.13}$$

Let $\mathcal{O}^1_{BEPT}(G, l, b)$ be the subset of $\mathcal{Q}^1_{BEPT}(G, l, b)$ that satisfies the odd-cut constraints (5.12), and let $\mathcal{O}^2_{BEPT}(G, l, b)$ be the subset of $\mathcal{Q}^2_{BEPT}(G, l, b)$ that satisfies the odd-cut constraints (5.13). Since MINIMUM EDGES POSTMAN TOUR is $\mathcal{NP}$-hard, we cannot expect that $\mathcal{O}^2_{BEPT}(G, l, b) = \mathcal{P}^2_{BEPT}(G, l, b)$. In Figure 5.8 we show one of the smallest examples we know of an undirected graph G, with vector of demands b, and vector of lower bounds l, together with a fractional extreme point x of $\mathcal{O}^2_{BEPT}(G, l, b)$.

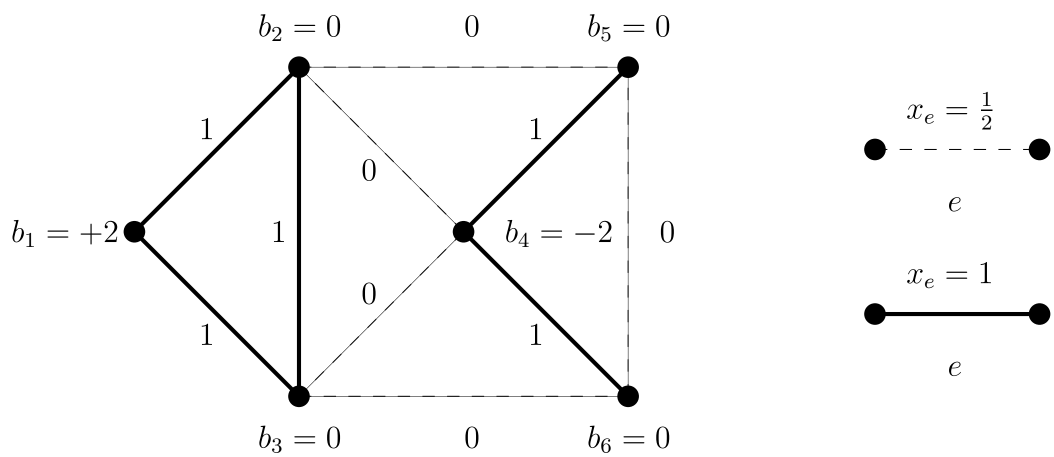

Figure 5.8: A fractional extreme point of $\mathcal{O}^2_{BEPT}(G, l, b)$.

5.4.2 b-Joins

Let $G = (V, E)$ be an undirected graph, and let $T \subseteq V$ with $|T|$ even. A *(generalized) T-join* of G is a vector $x \in \mathbb{Z}_+^E$ such that for each $v \in V$, $x(\delta(v))$ is odd if and only if $v \in T$. For $S \subseteq V$, we say that S is *T-odd* and that $\delta(S)$ is a *T-cut* if $|S \cap T|$ is odd. Let $b \in \mathbb{Z}^V$ be a vector with $b(V)$ even, and let $T = \{v \in V : b_v \text{ is odd}\}$. Note that $|T|$ is even. We say that $x \in \mathbb{Z}_+^E$ is a *b-join* of G if x is a T-join of G, and $x(\delta(v)) \geq b_v$ for all $v \in V$.

Problem: Minimum b-Join.
Input: An undirected graph $G = (V, E)$, a vector $b \in \mathbb{Z}^V$, and a vector $c \in \mathbb{Q}_+^E$.
Output: The minimum cost $\mathrm{MBJ}(M, c)$ of a b-join of G.

We can slightly generalize b-joins by giving a vector $l \in \mathbb{Z}^E$ of lower bounds, and requiring that $x \geq l$. Let $\mathcal{P}_{BJ}(G, l, b) \subseteq \mathbb{R}^E$ be the polyhedron defined by

$$x(\delta(v)) \geq b_v \text{ for all } v \in V \tag{5.14}$$

$$x(\delta(S)) \geq l(\delta(S)) + 1 \text{ for all odd } S \subseteq V \tag{5.15}$$

$$x_e \geq l_e \text{ for all } e \in E, \tag{5.16}$$

where $S \subseteq V$ is odd if $b(S) + l(\delta(S))$ is odd. Let $\mathcal{P}_{BJ}(G, b)$ be the polyhedron $\mathcal{P}_{BJ}(G, \mathbf{0}, b)$. We prove first that the integrality of $\mathcal{P}_{BJ}(G, b)$ implies the integrality of $\mathcal{P}_{BJ}(G, l, b)$.

Lemma 5.7 *If the polyhedron $\mathcal{P}_{BJ}(G, b)$ is integral for all choices of b, then the polyhedron $\mathcal{P}_{BJ}(G, l, b)$ is integral for all choices of l and b.*

Proof. Note that $\mathcal{P}_{BJ}(G, l, b)$ is the translate of $\mathcal{P}_{BJ}(G, b')$ obtained from $y = x - l$, where, for all $v \in V$, $b'_v = b_v - l(\delta(v))$. It is enough to verify that parity is preserved:

$$b'(S) = b(S) - \sum_{v \in S} l(\delta(v)) = b(S) + l(\delta(S)) - 2l(\delta(S)) - 2l(\gamma(S)). \tag{5.17}$$

Hence $b'(S)$ is odd if and only if $b(S) + l(\delta(S))$ is odd. ■

Now we prove that $\mathcal{P}_{BJ}(G, b)$ is indeed integral. Our proof is very similar to one given by Cook et al. [16, Section 6.1] for a description of the *b-factor polytope*.

Theorem 5.8 *The polyhedron $\mathcal{P}_{BJ}(G, b)$ is integral.*

Proof. It is easy to see that if x is a b-join, then $x \in \mathcal{P}_{BJ}(G, b)$. We are going to show that if $x \in \mathcal{P}_{BJ}(G, b)$ then x is a convex combination of some b-joins.

Let y be a T-join of G. For each $v \in V$, define the *violation* $f(y, v)$ of y at v as

$$f(y, v) = \max(0, b_v - y(\delta(v))). \tag{5.18}$$

Define the *total violation* $f(y)$ of y as $f(y) = \sum_{v \in V} f(y, v)$.

Since $\mathcal{P}_{BJ}(G, b)$ is a subset of the T-join polyhedron for (G, T), we can write x as a convex combination of some T-joins $x = \sum_{i=1}^{k} \lambda_i x^i$. Choose this convex combination so that the *weighted violation* $\alpha = \sum_{i=1}^{k} \lambda_i f(x^i)$ is as small as possible. If $\alpha = 0$ then each x^i is a b-join, and we are done. Otherwise, we can assume that $f(x^1, u) > 0$ for some vertex u, that is, $x^1(\delta(u)) < b_u$. Since $x(\delta(u)) \geq b_u$, we can also assume that $x^2(\delta(u)) > b_u$.

Let $\tilde{x} \in \mathbb{Z}_+^E$ be defined by $\tilde{x}_e = |x_e^1 - x_e^2|$ for $e \in E$. Since x^1, x^2 are T-joins, $\tilde{x}$ is an $\emptyset$-join, that is, $\tilde{x}(\delta(v))$ is even for all $v \in V$. Let $\tilde{G}$ be an undirected graph with vertex set V and edge set $E_1 \cup E_2$, where E_1 contains $\tilde{x}_e$ copies of edge e whenever $x_e^1 > x_e^2$, and E_2 contains $\tilde{x}_e$ copies of edge e otherwise. Construct an edge-simple closed path P as follows.

- Start at u. The first edge of P is in E_2.
- If P enters $v \in V$ on an edge in E_1, it leaves if possible on an edge in E_2.
- If P enters $v \neq u$ on an edge in E_2, it leaves if possible on an edge in E_1.
- If P enters u on an edge in E_2, it terminates.

This path exists since $\tilde{G}$ is an even undirected graph, and $|E_2 \cap \tilde{\delta}(u)| > |E_1 \cap \tilde{\delta}(u)|$. For each $e \in E$, let $\bar{x}_e$ be the number of copies of e used by P. Note that $\bar{x}$ is again a $\emptyset$-join. Define the vectors y^1, y^2 by

$$y_e^1 = \begin{cases} x_e^1 - \bar{x}_e & \text{if } x_e^1 > x_e^2 \\ x_e^1 + \bar{x}_e & \text{if } x_e^1 < x_e^2 \\ x_e^1 & \text{otherwise} \end{cases} \quad \text{and} \quad y_e^2 = \begin{cases} x_e^2 - \bar{x}_e & \text{if } x_e^2 > x_e^1 \\ x_e^2 + \bar{x}_e & \text{if } x_e^2 < x_e^1 \\ x_e^2 & \text{otherwise.} \end{cases} \tag{5.19}$$

Since x^1, x^2 are T-joins, and $\bar{x}$ is a $\emptyset$-join, we have that y^1, y^2 are T-joins. Moreover, it is not difficult to see that $y^1 + y^2 = x^1 + x^2$ and $f(y^1) + f(y^2) < f(x^1) + f(x^2)$. Define

z^i to be x^i for $1 \leq i \leq k$, and $z^{k+1} = y^1$, $z^{k+2} = y^2$. Let $\epsilon = \min\{\lambda_1, \lambda_2\}$, and define $\mu_1 = \lambda_1 - \epsilon$, $\mu_2 = \lambda_2 - \epsilon$, $\mu_i = \lambda_i$ for $3 \leq i \leq k$, and $\mu_{k+1} = \mu_{k+2} = \epsilon$. Then $\sum_{i=1}^{k+2} \mu_i z^i$ is an expression for x as a convex combination of T-joins for which the weighted violation satisfies $\sum_{i=1}^{k+2} \mu_i f(z^i) < \alpha$, a contradiction to the choice of the x^i. ■

Since $\mathcal{P}_{BJ}(G, l, b)$ is integral, and since we can separate in polynomial time all its defining constraints, the equivalence between separation and optimization implies that we can optimize over $\mathcal{P}_{BJ}(G, l, b)$ in polynomial time, that is, we can solve MINIMUM b-JOIN in polynomial time, even in the presence of lower bounds. An interesting question is whether we can give such an algorithm that does not depend on the ellipsoid method.

A vector $x \in \mathbb{Z}_+^E$ is a *perfect b-matching* of G if $x(\delta(v)) = b_v$ for all $v \in V$. Observe that perfect b-matchings are precisely the integral vectors in the set of solutions $\mathcal{P}_{PBM}(G, b)$ of the system

$$x(\delta(v)) = b_v \text{ for all } v \in V \tag{5.20}$$

$$x(\delta(S)) \geq 1 \text{ for all odd } S \subseteq V \tag{5.21}$$

$$x_e \geq 0 \text{ for all } e \in E. \tag{5.22}$$

Furthermore, since $\mathcal{P}_{PBM}(G, b)$ is a face of $\mathcal{P}_{BJ}(G, b)$, it follows from Theorem 5.8 that the polytope $\mathcal{P}_{PBM}(G, b)$ is integral, and hence $\mathcal{P}_{PBM}(G, b)$ is the convex hull of perfect b-matchings of (G, b), a result of Edmonds and Johnson [28].

5.5 Other Valid Inequalities

In this section we describe a new class of valid inequalities for $\mathcal{P}^1_{BEPT}(G, l, b)$ obtained via Gomory-Chvátal cutting plane proofs [14, 44]. This class of constraints is unusual in that it arises from congruency modulo 4, instead of modulo 2 as the classic odd-cut constraints.

Let $T_0, T_1, \ldots, T_k$ be $k+1$ subsets of V such that

1. $b(T_0) \equiv l(\delta(T_0)) + k + 2 \pmod 4$,
2. $T_1, \ldots, T_k$ are odd, and
3. each $e \in E$ appears an even number of times in $\delta(T_0), \delta(T_1), \ldots, \delta(T_k)$.

For $e \in E$, let t_e be the number of times that e appears in $\delta(T_1), \ldots, \delta(T_k)$. Note that t_e is odd if and only if $e \in \delta(T_0)$. Define the subset $F \subseteq E$ by $e \in F$ if and only if either $e \in \delta(T_0)$ and $t_e \equiv 3 \pmod 4$, or $e \notin \delta(T_0)$ and $t_e \equiv 2 \pmod 4$.

5.5.1 First Rounding

We will consider the two valid inequalities

$$\tfrac{1}{4}(x(\delta(T_0)) + 3\sum_{i=1}^{k} x(\delta(T_i)) + 2x(F)) \geq \tfrac{1}{4}(b(T_0) + 3\sum_{i=1}^{k}(l(\delta(T_i)) + 1) + 2l(F)) \tag{5.23}$$

and

$$\tfrac{1}{4}(3x(\delta(T_0)) + \sum_{i=1}^{k} x(\delta(T_i)) + 6x(F)) \geq \tfrac{1}{4}(3b(T_0) + \sum_{i=1}^{k}(l(\delta(T_i)) + 1) + 6l(F)) \tag{5.24}$$

obtained from adding multiples of the valid inequalities $x(\delta(T_0)) \geq b(T_0)$, $x(F) \geq l(F)$, and $x(\delta(T_i)) \geq l(\delta(T_i)) + 1$ for all $1 \leq i \leq k$.

First, we verify that all the coefficients on the left-hand sides are integers, then we show that the right-hand sides are non-integral multiples of $\frac{1}{2}$. This will allow us to add $\frac{1}{2}$ to both right-hand sides to obtain the valid inequalities

$$\tfrac{1}{4}(x(\delta(T_0)) + 3\sum_{i=1}^{k} x(\delta(T_i)) + 2x(F)) \geq \tfrac{1}{4}(b(T_0) + 3\sum_{i=1}^{k}(l(\delta(T_i)) + 1) + 2l(F) + 2) \tag{5.25}$$

and

$$\tfrac{1}{4}(3x(\delta(T_0)) + \sum_{i=1}^{k} x(\delta(T_i)) + 6x(F)) \geq \tfrac{1}{4}(3b(T_0) + \sum_{i=1}^{k}(l(\delta(T_i)) + 1) + 6l(F) + 2). \tag{5.26}$$

For $e \in E$, let c_e and d_e be the coefficients of x_e in the left-hand sides of (5.23) and (5.24), respectively. Observe that: If $e \in \delta(T_0)$, then $c_e = \frac{3}{4}(t_e + 1)$ and $d_e = \frac{1}{4}(t_e + 9)$ if $e \in F$, and $c_e = \frac{1}{4}(3t_e + 1)$ and $d_e = \frac{1}{4}(t_e + 3)$ if $e \notin F$. If $e \notin \delta(T_0)$, then $c_e = \frac{1}{4}(3t_e + 2)$ and $d_e = \frac{1}{4}(t_e + 6)$ if $e \in F$, and $c_e = \frac{3}{4}t_e$ and $d_e = \frac{1}{4}t_e$ if $e \notin F$. Else, if $e \notin \cup_{i=0}^{k}\delta(T_i)$, then $c_e = d_e = 0$.

Note that in every case, c_e and d_e are integers. To verify our second claim, observe that the right-hand sides of (5.23) and (5.24) have the same fractional parts as

$$\tfrac{1}{4}(l(\delta(T_0)) + 3\sum_{i=1}^{k} l(\delta(T_i)) + 2l(F) + 4k + 2) \tag{5.27}$$

and

$$\tfrac{1}{4}(3l(\delta(T_0)) + \sum_{i=1}^{k} l(\delta(T_i)) + 6l(F) + 4k + 6), \tag{5.28}$$

respectively. Now note that the coefficients of l_e in the above expressions are c_e and d_e, that the constant terms are $\frac{1}{2}$ and $\frac{3}{2}$, respectively, and that the coefficients of k are both 1.

5.5.2 Second Rounding

Add inequalities (5.25) and (5.26), and multiply the result by $\frac{1}{2}$. After some simplification, we obtain the valid inequality

$$\tfrac{1}{2}(x(\delta(T_0)) + \sum_{i=1}^{k} x(\delta(T_i)) + 2x(F)) \geq \tfrac{1}{2}(b(T_0) + \sum_{i=1}^{k}(l(\delta(T_i)) + 1) + 2l(F) + 1). \tag{5.29}$$

For $e \in E$, let a_e be the coefficient of x_e on the left-hand side of (5.29). We claim that each a_e is integral, and that the right-hand side has a fractional part of $\frac{1}{2}$. As before, if $e \notin \cup_{i=0}^{k}\delta(T_i)$, then $a_e = 0$. If $e \in \delta(T_0)$, then $a_e = \frac{1}{2}(t_e + 3)$ if $e \in F$, and $a_e = \frac{1}{2}(t_e + 1)$ if $e \notin F$. Else, if $e \notin \delta(T_0)$, then $a_e = \frac{1}{2}(t_e + 2)$ if $e \in F$, and $a_e = \frac{1}{2}t_e$ if $e \notin F$.

To verify our second claim, note that the right-hand side of (5.29) has the same fractional part as

$$\tfrac{1}{2}(l(\delta(T_0)) + \sum_{i=1}^{k} l(\delta(T_i)) + 2l(F) + 2k + 1). \tag{5.30}$$

Note that the coefficient of l_e in the above expression is a_e, that the coefficient of k is 1, and that the constant term is $\frac{1}{2}$. Hence we can add $\frac{1}{2}$ to the right-hand side of (5.29) to obtain the valid inequality

$$\tfrac{1}{2}(x(\delta(T_0)) + \sum_{i=1}^{k} x(\delta(T_i)) + 2x(F)) \geq \tfrac{1}{2}(b(T_0) + \sum_{i=1}^{k}(l(\delta(T_i)) + 1) + 2l(F) + 2), \tag{5.31}$$

or simply

$$x(\delta(T_0)) + \sum_{i=1}^{k} x(\delta(T_i)) + 2x(F) \geq b(T_0) + \sum_{i=1}^{k} (l(\delta(T_i)) + 1) + 2l(F) + 2. \tag{5.32}$$

5.5.3 Small Values of k are Redundant

Assume that x is a vector that satisfies all the flow, odd-cut and lower bound constraints. We show that if $k \leq 3$, then x satisfies all the constraints (5.32).

If $k = 1$, it must be that $b(T_0) \equiv l(\delta(T_0)) + 3 \pmod 4$, $\delta(T_0) = \delta(T_1)$, and F is empty. We can further assume that $T = T_0 = T_1$. Hence, the inequality

$$x(\delta(T_0)) + x(\delta(T_1)) + 2x(F) \geq b(T_0) + l(\delta(T_1)) + 2l(F) + 3 \tag{5.33}$$

is equivalent to

$$2x(\delta(T)) \geq b(T) + l(\delta(T)) + 3. \tag{5.34}$$

To see that this inequality is redundant, note that either

1. $b(T) \geq l(\delta(T)) + 3$, and hence $2x(\delta(T)) \geq 2b(T) \geq b(T) + l(\delta(T)) + 3$, or
2. $b(T) \leq l(\delta(T)) - 1$, and hence $2x(\delta(T)) \geq 2(l(\delta(T)) + 1) \geq b(T) + l(\delta(T)) + 3$.

If $k = 2$, our assumptions imply that $b(T_0) \equiv l(\delta(T_0)) \pmod 4$. With a bit more work we see that all possible configurations of T_0, T_1, T_2 can be obtained as follows: Let (A, B, C, D) be a partition of the vertex set such that $T_0 = A \cup B$, $T_1 = A \cup C$, and $T_2 = B \cup C$. The set F will consist of the edges between A and B, and the edges between C and D. Moreover, we can easily check that

$$x(\delta(T_0)) + 2x(F) = x(\delta(T_1)) + x(\delta(T_2)) \tag{5.35}$$

and

$$l(\delta(T_0)) + 2l(F) = l(\delta(T_1)) + l(\delta(T_2)). \tag{5.36}$$

Hence, the inequality

$$x(\delta(T_0)) + x(\delta(T_1)) + x(\delta(T_2)) + 2x(F) \geq b(T_0) + l(\delta(T_1)) + l(\delta(T_2)) + 2l(F) + 4 \tag{5.37}$$

is equivalent to

$$2x(\delta(T_0)) + 4x(F) \geq b(T_0) + l(\delta(T_0)) + 4l(F) + 4. \tag{5.38}$$

To see that this inequality is redundant, note that one of the following is true:

1. $b(T_0) \geq l(\delta(T_0)) + 4$, and hence

$$\begin{aligned} 2x(\delta(T_0)) + 4x(F) &\geq 2b(T_0) + 4l(F) & (5.39) \\ &\geq b(T_0) + l(\delta(T_0)) + 4l(F) + 4. & (5.40) \end{aligned}$$

2. $b(T_0) = l(\delta(T_0))$, and hence

$$\begin{aligned} 2x(\delta(T_0)) + 4x(F) &= 2x(\delta(T_1)) + 2x(\delta(T_2)) & (5.41) \\ &\geq 2l(\delta(T_1)) + 2l(\delta(T_2)) + 4 & (5.42) \\ &= 2l(\delta(T_0)) + 4l(F) + 4 & (5.43) \\ &= b(T_0) + l(\delta(T_0)) + 4l(F) + 4. & (5.44) \end{aligned}$$

3. $b(T_0) \leq l(\delta(T_0)) - 4$, and hence

$$\begin{aligned} 2x(\delta(T_0)) + 4x(F) &\geq 2l(\delta(T_0)) + 4l(F) & (5.45) \\ &\geq b(T_0) + l(\delta(T_0)) + 4l(F) + 4. & (5.46) \end{aligned}$$

If $k = 3$, our assumptions imply that $b(T_0) \equiv l(\delta(T_0)) + 1 \pmod 4$. In particular T_0 is odd. With a bit more work we see that all possible configurations of T_0, T_1, T_2, T_3 can be obtained as follows: Let $(A_{0000}, A_{0011}, A_{0101}, A_{0110}, A_{1001}, A_{1010}, A_{1100}, A_{1111})$ be a partition of the vertex set with $T_0 = A_{0011} \cup A_{0101} \cup A_{1001} \cup A_{1111}$, $T_1 = A_{0011} \cup A_{0110} \cup A_{1010} \cup A_{1111}$, $T_2 = A_{0101} \cup A_{0110} \cup A_{1100} \cup A_{1111}$, and $T_3 = A_{1001} \cup A_{1010} \cup A_{1100} \cup A_{1111}$. Just as before, we can verify that

$$x(\delta(T_0)) + 2x(F) = x(\delta(T_1)) + x(\delta(T_2)) + x(\delta(T_3)) \tag{5.47}$$

and

$$l(\delta(T_0)) + 2l(F) = l(\delta(T_1)) + l(\delta(T_2)) + l(\delta(T_3)). \tag{5.48}$$

Hence, the inequality

$$x(\delta(T_0)) + \sum_{i=1}^{3} x(\delta(T_i)) + 2x(F) \geq b(T_0) + \sum_{i=1}^{3} l(\delta(T_i)) + 2l(F) + 5 \tag{5.49}$$

is equivalent to

$$2x(\delta(T_0)) + 4x(F) \geq b(T_0) + l(\delta(T_0)) + 4l(F) + 5. \tag{5.50}$$

To see that this inequality is redundant, note that one of the following is true:

1. $b(T_0) \geq l(\delta(T_0)) + 5$, and hence

$$2x(\delta(T_0)) + 4x(F) \geq 2b(T_0) + 4l(F) \tag{5.51}$$

$$\geq b(T_0) + l(\delta(T_0)) + 4l(F) + 5. \tag{5.52}$$

2. $b(T_0) = l(\delta(T_0)) + 1$, and hence

$$2x(\delta(T_0)) + 4x(F) = 2x(\delta(T_1)) + 2x(\delta(T_2)) + 2x(\delta(T_3)) \tag{5.53}$$

$$\geq 2l(\delta(T_1)) + 2l(\delta(T_2)) + 2l(\delta(T_3)) + 6 \tag{5.54}$$

$$= 2l(\delta(T_0)) + 4l(F) + 6 \tag{5.55}$$

$$= b(T_0) + l(\delta(T_0)) + 4l(F) + 5. \tag{5.56}$$

3. $b(T_0) \leq l(\delta(T_0)) - 3$, and hence

$$2x(\delta(T_0)) + 4x(F) \geq 2(l(\delta(T_0)) + 1) + 4l(F) \tag{5.57}$$

$$\geq b(T_0) + l(\delta(T_0)) + 4l(F) + 5. \tag{5.58}$$

On the other hand, some inequalities with $k = 4$ are not redundant. The fractional point shown in Figure 5.8 violates two facet inducing inequalities of $\mathcal{P}^1_{BEPT}(G, l, b)$ that can be obtained from (5.32) with the two families of subsets

$$T_0 = \{1,2,3\}, T_1 = \{5\}, T_2 = \{6\}, T_3 = \{2,5\}, T_4 = \{2,4,5\} \tag{5.59}$$

and

$$T_0 = \{1,2,3\}, T_1 = \{5\}, T_2 = \{6\}, T_3 = \{3,6\}, T_4 = \{3,4,6\}. \tag{5.60}$$

5.5.4 Generalization

Let $S_1, \ldots, S_j$ and $T_1, \ldots, T_k$ be subsets of V such that

1. $\sum_{i=1}^{j} b(S_i) \equiv \sum_{i=1}^{j} l(\delta(S_i)) + k + 2 \pmod 4$,
2. $T_1, \ldots, T_k$ are odd, and
3. each $e \in E$ appears an even number of times in $\delta(S_1), \ldots, \delta(S_j), \delta(T_1), \ldots, \delta(T_k)$.

For $e \in E$, let s_e be the number of times that e appears in $\delta(S_1), \ldots, \delta(S_j)$, and let t_e be the number of times that e appears in $\delta(T_1), \ldots, \delta(T_k)$. Note that t_e is odd if and only if s_e is odd. Define the subset $F \subseteq E$ by $e \in F$ if and only if either $s_e \equiv 1 \pmod 4$ and $t_e \equiv 3 \pmod 4$, or $s_e \equiv 0 \pmod 4$ and $t_e \equiv 2 \pmod 4$.

As before, we can obtain after two roundings the valid inequality

$$\sum_{i=1}^{j} x(\delta(S_i)) + \sum_{i=1}^{k} x(\delta(T_i)) + 2x(F) \geq \sum_{i=1}^{j} b(S_i) + \sum_{i=1}^{k} (l(\delta(T_i)) + 1) + 2l(F) + 2. \quad (5.61)$$

We observe that, if $T \subseteq V$ is odd, it is possible to obtain the odd-cut constraints (5.13) from (5.61) by setting $T_1 = T$ and $T_2 = V \setminus T$.

5.6 Approximation Algorithms

In this section we describe four approximation algorithms for MINIMUM EDGES POSTMAN TOUR. Our main contribution is an algorithm that has both a guarantee of $\frac{4}{3}$ for MINIMUM EDGES POSTMAN TOUR, and a guarantee of 2 for MINIMUM EDGES POSTMAN SET. To the best of our knowledge, this is the first positive result about approximating the cost of an optimal postman set for any $\mathcal{NP}$-hard postman problem.

In the analysis of our algorithms, we use the following easy consequence of Theorem 2.8.

Lemma 5.9 *Let $G = (V, E)$ be a 2-edge-connected undirected graph, let $T \subseteq V$ be an even set, and let $c \in \mathbb{Q}_+^E$. Then G has a T-join J with cost $c(J) \leq \frac{1}{2}c(E)$.*

The next lemma allows us to consider only 2-edge-connected undirected graphs.

Lemma 5.10 (Veerasamy [85]) *Let (G, b, c) be an instance of* MINIMUM EDGES POSTMAN TOUR *with $G = (V, E)$ connected. Assume that $e \in E$ is a cut-edge, and that $S \subseteq V$ is such that $\delta(S) = \{e\}$. Then any optimal solution $x^* \in \mathbb{Z}_+^E$ for (G, b, c) has $x_e^* = |b(S)|$ if $b(S) \neq 0$, and $x_e^* = 2$ if $b(S) = 0$.*

Proof. Without loss of generality, we can assume that $b(S) \geq 0$. If $b(S) > 0$, then any feasible solution for (G, b, c) must carry $b(S)$ units of flow from S to $\bar{S}$ along e, and the optimal way of doing this is setting $x_e^* = b(S)$. If $b(S) = 0$, then in any feasible solution for (G, b, c) no flow is sent through e. Since e must be used at least once, it follows that it must be used at least twice. Hence, the optimal way of doing this is setting $x_e^* = 2$, and carrying one unit of flow in each direction. ■

Therefore, before applying any of the algorithms that we describe later, we apply the following procedure due to Veerasamy [85] to a given instance (G, b): As long as G has a cut-edge $e = \{u, v\}$, let S be as in the above lemma, with $u \in S$, $v \in \bar{S}$, let $G' = G \setminus e$, define b' as $b'_u = b_u - b(S)$, $b'_v = b_v + b(S)$, and $b'_w = b_w$ for all $w \in V \setminus \{u, v\}$, and redefine (G, b) as (G', b'). After at most $|V|$ iterations, all connected components of G are 2-edge-connected, and we apply any of our four algorithms to each of them.

To show the workings of the four algorithms we describe next, we apply each of them to the instance (M, c) shown in Figure 5.9, with $\text{MEPT}(M, c) = 26$ and $\text{MEPSP}(M, c) = 6$.

5.6.1 A $\frac{5}{2}$-Approximation Algorithm

The first approximation algorithm we describe (called EDGES1) was given by Veerasamy in his doctoral thesis [85]. Given an instance (G, b, c) of MINIMUM EDGES POSTMAN TOUR where G is a 2-edge-connected undirected graph, find a minimum cost feasible flow x^F on the directed graph $\vec{G}$, with demands b and costs $c_{e^+} = c_{e^-} = c_e$ for all $e \in E$. Let $U = \{e \in E : x_{e^+}^F + x_{e^-}^F = 0\}$, which we call the set of *unused* edges by the flow x^F. Let $T = \{v \in V : d_U(v) \text{ is odd}\}$, and let J be a minimum cost T-join of (G, T, c). Note that U together with J is an even subgraph of G, and hence each of its connected components is Eulerian. The output of EDGES1 is x^1, the incidence vector of the edges postman tour of (G, b) obtained from adding U and J to the flow x^F.

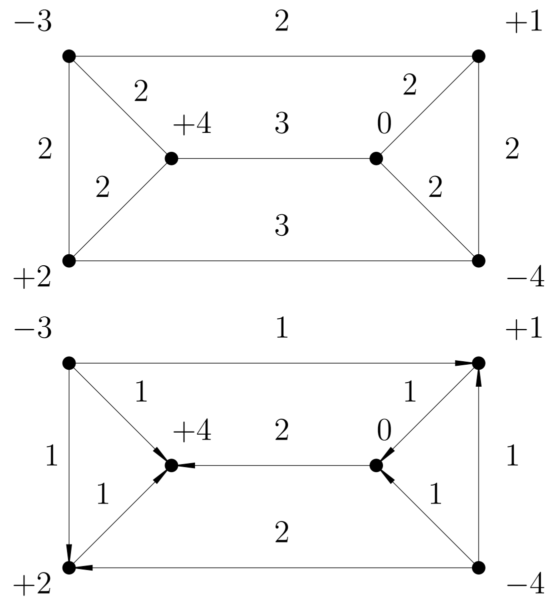

Figure 5.9: An example solved optimally.

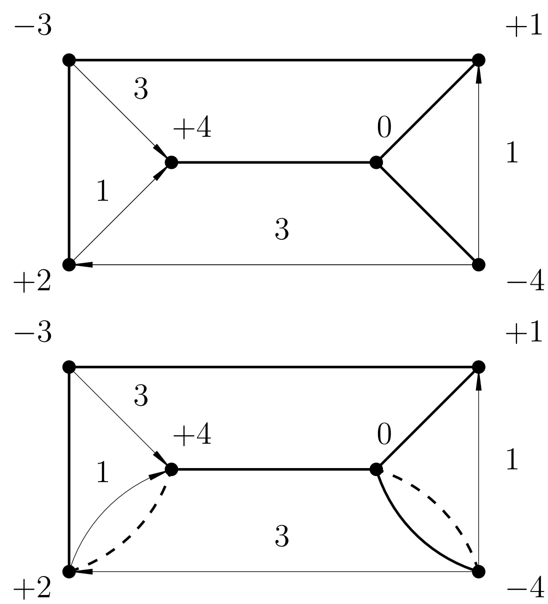

Figure 5.10: Veerasamy's algorithm applied to an example.

On the top of Figure 5.10 we indicate with arrows the flow x^F, with cost 19, and with bold edges the set U, with cost 11. On the bottom, we show with dashed edges the set J, with cost 4. The edges postman tour found by EDGES1 has cost 34.

Veerasamy claimed that EDGES1 has a guarantee of $\frac{3}{2}$ for MINIMUM EDGES POSTMAN TOUR. However, we prove that the guarantee of EDGES1 is somewhere between 2 and $\frac{5}{2}$.

Theorem 5.11 *Algorithm* EDGES1 *is a* $\frac{5}{2}$*-approximation algorithm for* MINIMUM EDGES POSTMAN TOUR. *The guarantee of* EDGES1 *is not better than* 2. *Algorithm* EDGES1 *has no guarantee for* MINIMUM EDGES POSTMAN SET.

Proof. Let $G = (V, E)$ be a 2-edge-connected undirected graph, and let (G, b, c) be an instance of MINIMUM EDGES POSTMAN TOUR with optimal value $C^* \geq c(E)$. Since any edges postman tour of (G, b) corresponds with a feasible flow of $(\vec{G}, b)$, it follows that C^* is at least the cost C^F of x^F. In the worst case, all edges are left unused by x^F, and hence $c(U) \leq c(E)$. By Lemma 5.9, $c(J) \leq \frac{1}{2}c(E)$. Hence, the cost C^1 of x^1 satisfies

$$C^1 = C^F + c(U) + c(J) \leq C^* + \tfrac{3}{2}c(E) \leq \tfrac{5}{2}C^*. \tag{5.62}$$

For each $\epsilon > 0$, consider the undirected graph consisting of two parallel edges e and f, with ends u and v, and with $c_e = 1$, $c_f = 1 + \epsilon$, $b_u = +2$, and $b_v = -2$. It is easy to see that an application of EDGES1 to this instance gives $x_e^F = 2$, $x_f^F = 0$, $U = \{f\}$, and $J = \{e\}$, with cost $C^1 = 2 + (1 + \epsilon) + 1 = 4 + \epsilon$. However, an optimal solution to this instance has $x_e^* = x_f^* = 1$, with cost $C^* = 2 + \epsilon$. Hence $\lim_{\epsilon \to 0} \frac{C^1}{C^*} = 2$. Observe that an optimal edges postman set has cost 0, while EDGES1 outputs an edges postman set of cost 2. ■

5.6.2 A 2-Approximation Algorithm

The second approximation algorithm we describe (called EDGES2) is an adaptation of the 2-approximation algorithm for MINIMUM WINDY POSTMAN TOUR due to Win. In fact, the first part of our proof is very similar to Win's proof for his algorithm [89]. Given an instance (G, b, c) of MINIMUM EDGES POSTMAN TOUR where G is a 2-edge-connected undirected graph, find an extreme point optimal solution x^L with cost C^L of the linear programming relaxation LMEPT1(G, b, c) of MINIMUM EDGES POSTMAN TOUR. Recall that x^L is a half-integral vector. Let F be the set of edges with fractional components. The output of EDGES2 is x^2, the integral vector obtained by adding $\frac{1}{2}$ to each fractional component of x^L. Note that x^2 is the incidence vector of an edges postman tour of (G, b).

On the top of Figure 5.11 we indicate with arrows the optimal solution x^L to the linear programming relaxation, with cost 25, and with bold edges the set F. On the bottom, we indicate with arrows the edges postman tour x^2 found by EDGES2, with cost 29.

Theorem 5.12 *Algorithm* EDGES2 *is a tight* 2*-approximation algorithm for* MINIMUM EDGES POSTMAN TOUR, *and it has no guarantee for* MINIMUM EDGES POSTMAN SET.

Proof. Let $G = (V, E)$ be a 2-edge-connected undirected graph, and let (G, b, c) be an instance of MINIMUM EDGES POSTMAN TOUR whose optimal solution x^* has value $C^* \geq c(E)$. Since x^* is feasible for LMEPT1(G, b, c), it follows that $C^L \leq C^*$. In the worst case, all edges are fractional, and hence $c(F) \leq c(E)$. Hence, the cost C^2 of x^2 satisfies

$$C^2 = C^L + c(F) \leq C^* + c(E) \leq 2C^*. \tag{5.63}$$

For each $\epsilon > 0$, consider the undirected graph consisting of two parallel edges e and f, with ends u and v, and with $c_e = \epsilon$, $c_f = 1$, $b_u = +1$, and $b_v = -1$. It is possible that an application of EDGES2 to this instance gives $x^L_{e^+} = 1$, $x^L_{e^-} = 0$, $x^L_{f^+} = x^L_{f^-} = \frac{1}{2}$, and $F = \{f\}$, with cost $C^2 = (1+\epsilon)+1 = 2+\epsilon$. However, an optimal solution to this instance has $x^*_{e^+} = x^*_{e^-} = 1$, $x^*_{f^+} = 1$, and $x^*_{f^-} = 0$, with cost $C^* = 1 + 2\epsilon$. Hence $\lim_{\epsilon \to 0} \frac{C^2}{C^*} = 2$. Observe that an optimal edges postman set has cost ϵ, while EDGES2 outputs an edges postman set of cost 1. Hence $\lim_{\epsilon \to 0} \frac{C^2 - c(E)}{C^* - c(E)} \to +\infty$. ■

5.6.3 A $\frac{3}{2}$-Approximation Algorithm

The third approximation algorithm we describe (called EDGES3) is a slight improvement of EDGES2. As before, find an extreme point optimal solution x^L with cost C^L of the linear programming relaxation LMEPT1(G, b, c) of MINIMUM EDGES POSTMAN TOUR, and let F be the set of edges with fractional components. Let $T = \{v \in V : d_F(v) \text{ is odd}\}$, and let J be a minimum cost T-join of (G, T, c). Note that F together with J is an even subgraph of G. The output of EDGES3 is x^3, the vector obtained by adding J to x^L.

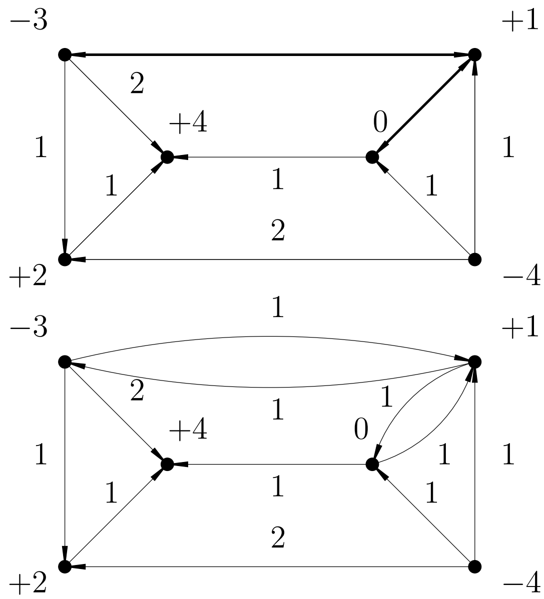

Figure 5.11: Win's algorithm and algorithm EDGES3 applied to an example.

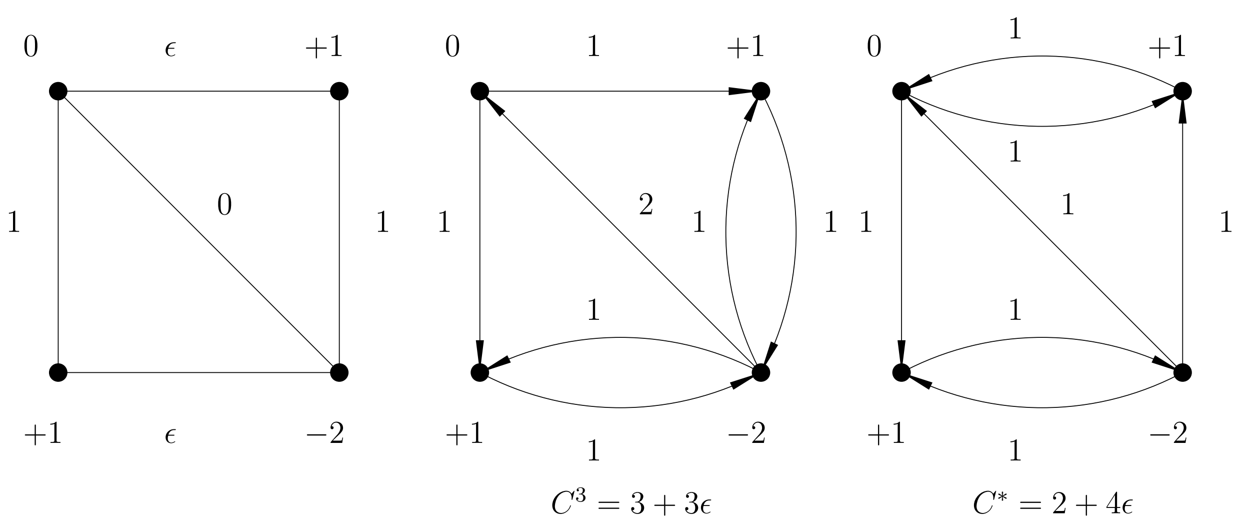

Figure 5.12: Worst case example for EDGES3.

Theorem 5.13 *Algorithm* EDGES3 *is a tight $\frac{3}{2}$-approximation algorithm for* MINIMUM EDGES POSTMAN TOUR, *and it has no guarantee for* MINIMUM EDGES POSTMAN SET.

Proof. Let $G = (V, E)$ be a 2-edge-connected undirected graph, and let (G, b, c) be an instance of MINIMUM EDGES POSTMAN TOUR whose optimal solution x^* has value $C^* \geq c(E)$. Since x^* is feasible for LMEPT1(G, b, c), it follows that $C^L \leq C^*$. By Lemma 5.9, $c(J) \leq \frac{1}{2}c(E)$. Hence, the cost C^3 of x^3 satisfies

$$C^3 = C^L + c(J) \leq C^* + \tfrac{1}{2}c(E) \leq \tfrac{3}{2}C^*. \tag{5.64}$$

For each $\epsilon > 0$, consider the undirected graph shown in Figure 5.12 consisting of a circuit of length four $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_1)$ and a diagonal $e_5 = v_1v_3$, with $c_1 = \epsilon$, $c_2 = 1$, $c_3 = \epsilon$, $c_4 = 1$, $c_5 = 0$, $b_1 = 0$, $b_2 = +1$, $b_3 = -2$, $b_4 = +1$. An application of EDGES3 to this instance gives $x^L_{1+} = 1$, $x^L_{1-} = 0$, $x^L_{2+} = x^L_{2-} = \frac{1}{2}$, $x^L_{3+} = x^L_{3-} = \frac{1}{2}$, $x^L_{4+} = 0$, $x^L_{4-} = 1$, $x^L_{5+} = 0$, $x^L_{5-} = 2$, $F = \{e_2, e_3\}$, $T = \{v_2, v_4\}$, $J = F$, with cost $C^3 = (2 + 2\epsilon) + (1 + \epsilon) = 3 + 3\epsilon$. However, an optimal solution to this instance has $x^*_{1+} = x^*_{1-} = 1$, $x^*_{2+} = 0$, $x^*_{2-} = 1$, $x^*_{3+} = x^*_{3-} = 1$, $x^*_{4+} = 0$, $x^*_{4-} = 1$, $x^*_{5+} = 0$, $x^L_{5-} = 1$, with cost $C^* = 2 + 4\epsilon$. Hence $\lim_{\epsilon \to 0} \frac{C^3}{C^*} = \frac{3}{2}$. Observe that an optimal edges postman set has cost 2ϵ, while EDGES3 outputs an edges postman set of cost $1 + \epsilon$. Hence $\lim_{\epsilon \to 0} \frac{C^3 - c(E)}{C^* - c(E)} \to +\infty$. ∎

5.6.4 A $\frac{4}{3}$-Approximation Algorithm

The previous three approximation algorithms have the common feature that first they satisfy the demands at the vertices, and then they correct for those edges that were left unused or used fractionally by adding a certain T-join. The fourth approximation algorithm that we describe (called EDGES4) performs these two steps in the reverse order.

Let $T = \{v \in V : b_v + d(v) \text{ is odd}\}$. The key observation is that any edges postman set of (G, b) must contain a T-join. Let J be a minimum cost T-join of (G, T, c). The output of EDGES4 is x^4, an extreme point optimal solution with cost C^4 of the linear programming relaxation LMEPT1(G, l, b, c) of MINIMUM EDGES POSTMAN TOUR, with lower bounds $l_e = 2$ if $e \in J$, and $l_e = 1$ otherwise. Since $b_v + l(\delta(v))$ is even for all $v \in V$, it follows that x^4 is an integral vector, and hence an edges postman tour of (G, b).

On the top of Figure 5.13 we indicate with bold edges the set J. On the bottom, we indicate with arrows the edges postman tour x^4 found by Edges4, with cost 28.

Theorem 5.14 *Algorithm* Edges4 *is a tight* $\frac{4}{3}$*-approximation algorithm for* Minimum Edges Postman Tour*, and also a tight* 2*-approximation algorithm for* Minimum Edges Postman Set.

Proof. Let $G = (V, E)$ be a 2-edge-connected undirected graph, and let (G, b, c) be an instance of Minimum Edges Postman Tour whose optimal solution x^* has value C^*. In order to satisfy the parity constraints at the vertices, any edges postman set of (G, b) must contain a T-join of (G, T), and hence $C^* \geq c(E) + c(J)$. By Lemma 5.9, $c(J) \leq \frac{1}{2}c(E)$, and hence $C^* \geq 3c(J)$. We can obtain a feasible solution to LMEPT1(G, l, b, c) by adding $\frac{1}{2}$ to each component of x^* corresponding with an edge in J. Hence

$$C^4 \leq C^* + c(J) \leq \tfrac{4}{3}C^* \tag{5.65}$$

and

$$C^4 - c(E) \leq C^* - c(E) + c(J) \leq 2(C^* - c(E)). \tag{5.66}$$

For each $\epsilon > 0$, consider the instance in Figure 5.14, consisting of an undirected circuit of length four $(v_1, e_1, v_2, e_2, v_3, e_3, v_4, e_4, v_1)$, with $c_1 = 4 - 2\epsilon$, $c_2 = 1$, $c_3 = 2 - \epsilon$, $c_4 = 1$, $b_1 = +1$, $b_2 = -1$, $b_3 = +4$, $b_4 = -4$. An application of Edges4 to this instance gives $T = \{v_1, v_2\}$, $J = \{e_1\}$, $x^4_{1+} = x^4_{1-} = 1$, $x^4_{2+} = 1$, $x^4_{2-} = 0$, $x^4_{3+} = 0$, $x^4_{3-} = 3$, $x^4_{4+} = 1$, $x^4_{4-} = 0$, with cost $C^4 = 16 - 7\epsilon$. However, an optimal solution to this instance has $x^*_{1+} = 1$, $x^*_{1-} = 0$, $x^*_{2+} = 2$, $x^*_{2-} = 0$, $x^*_{3+} = 0$, $x^*_{3-} = 2$, $x^*_{4+} = 2$, $x^*_{4-} = 0$, with cost $C^* = 12 - 4\epsilon$. Hence $\lim_{\epsilon \to 0} \frac{C^4}{C^*} = \frac{4}{3}$, and $\lim_{\epsilon \to 0} \frac{C^4 - c(E)}{C^* - c(E)} = \lim_{\epsilon \to 0} \frac{8 - 4\epsilon}{4 - \epsilon} = 2$. ■

We observe that none of the four algorithms we described outputs a solution with cost less than $16 - 7\epsilon$ for the instance in Figure 5.14. Hence, it is not possible to obtain an approximation algorithm for Minimum Edges Postman Tour with guarantee less than $\frac{4}{3}$ by just running the four algorithms with the same input and choosing the best output.

A problem that remains open is whether there exists an approximation algorithm (or even a polynomial-time algorithm to decide feasibility) for Minimum Bounded Edges Postman Tour, other than for the case $l < u$ (Theorems 3.14 and 3.15).

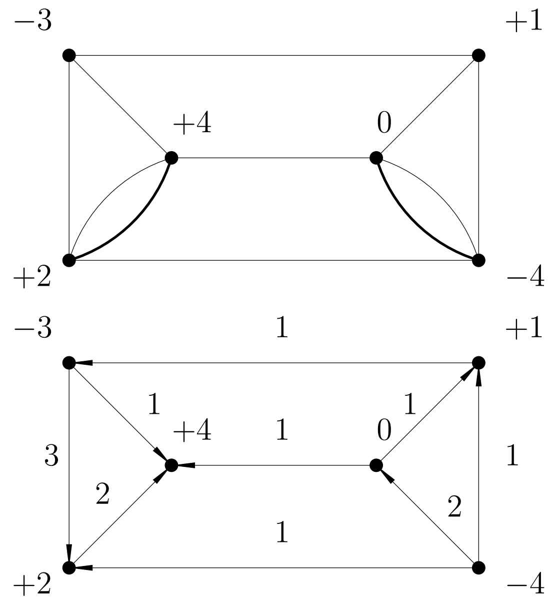

Figure 5.13: Algorithm EDGES4 applied to an example.

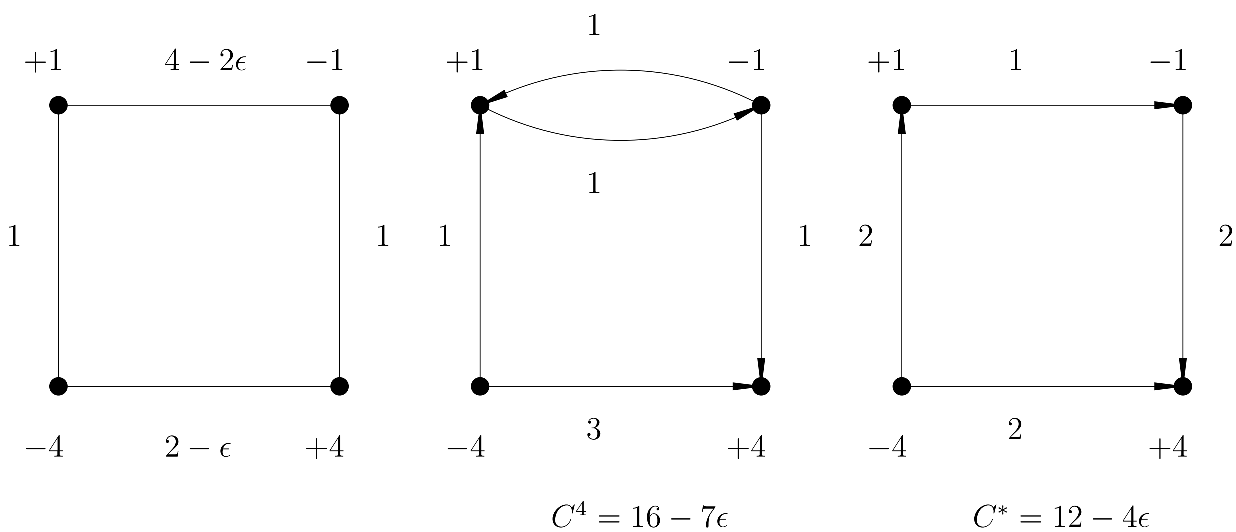

Figure 5.14: Worst case example for EDGES4.

Chapter 6

The Arcs Postman Problem

Language is a process of free creation; its laws and principles are fixed, but the manner in which the principles of generation are used is free and infinitely varied. Even the interpretation and use of words involves a process of free creation.

Language and Freedom, Noam Chomsky

We study another $\mathcal{NP}$-hard special case of MINIMUM RESTRICTED MIXED POSTMAN TOUR, called MINIMUM ARCS POSTMAN TOUR, as well as its feasibility version, called ARCS POSTMAN TOUR. The complexity of the latter was open, but we prove it is $\mathcal{NP}$-complete. In the rest of the chapter we study a large class of necessary conditions for feasibility, and we investigate some special cases of MINIMUM ARCS POSTMAN TOUR that can be solved in polynomial time.

6.1 Introduction

The second special case of MINIMUM RESTRICTED MIXED POSTMAN TOUR that we study is when the restricted set of the input mixed graph $M = (V, E, A)$ coincides with its edge set. We say that a postman tour of M is an *arcs postman tour* if it uses each edge of M exactly once. In Figure 6.1 we show a mixed graph and one of its arcs postman tours:

$$(u, e, v, b, w, c, x, d, z, g, v, b, c, x, d, z, i, a, u, f, w, c, x, h, y, a, u). \tag{6.1}$$

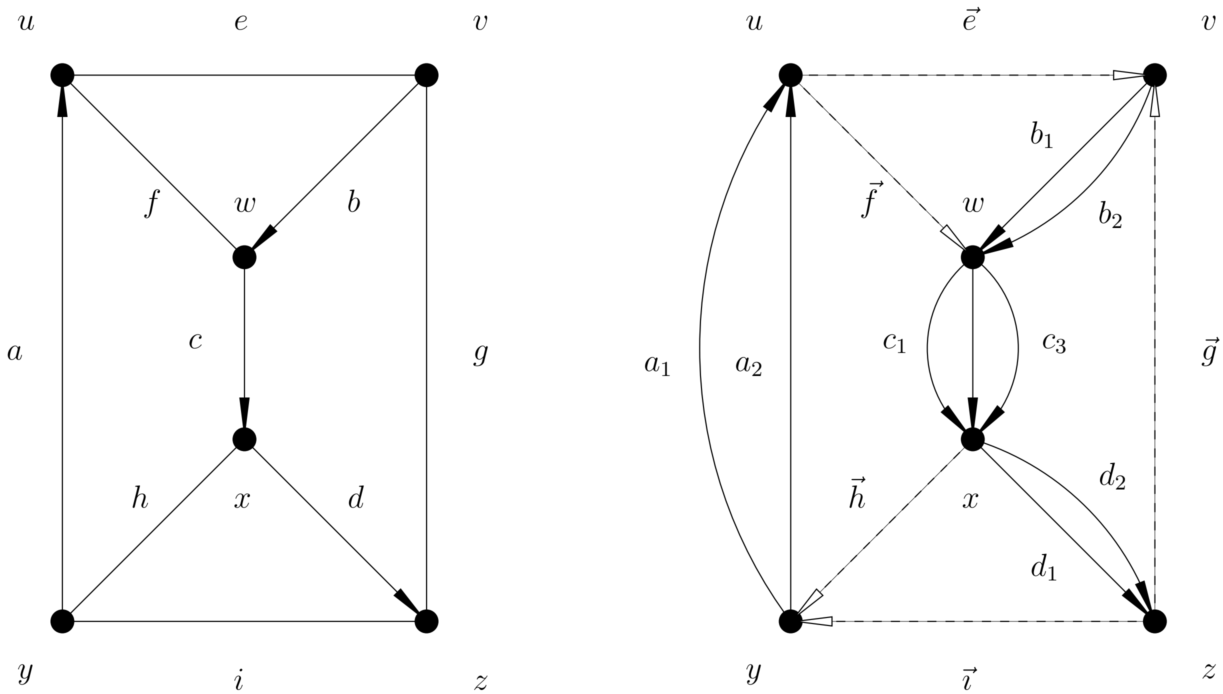

Figure 6.1: A mixed graph and an arcs postman tour.

Note that in this tour, edge e has been *oriented* from u to v to obtain the arc $\vec{e}$. In order to avoid confusion with the original arcs, or the original edges, we draw these oriented edges as dashed and hollow arrows. Now we define the two problems that we study in this chapter. Note that since edges cannot be replicated, we do not assign them a cost in the minimization version of the problem.

Problem: ARCS POSTMAN TOUR.
Input: A mixed graph $M = (V, E, A)$.
Output: Does M have an arcs postman tour?

Problem: MINIMUM ARCS POSTMAN TOUR.
Input: A strongly connected mixed graph $M = (V, E, A)$, and a vector $c \in \mathbb{Q}_+^A$.
Output: The minimum cost MAPT(M, c) of an arcs postman tour of M.

6.2 Computational Complexity

As Veerasamy pointed out [85], Papadimitriou's original proof [70] of $\mathcal{NP}$-completeness of the decision version of MINIMUM MIXED POSTMAN TOUR does not need to duplicate edges, that is, MINIMUM ARCS POSTMAN TOUR is $\mathcal{NP}$-hard even with unit costs. We prove some stronger negative results.

Theorem 6.1 ARCS POSTMAN TOUR *is* $\mathcal{NP}$*-complete.*

Proof. We are going to reduce NOT ALL EQUAL SATISFIABILITY (which is $\mathcal{NP}$-complete by Theorem 1.5) to ARCS POSTMAN TOUR. Let I be an instance of NOT ALL EQUAL SATISFIABILITY with $n \in \mathbb{N}$ variables $x_1, \ldots x_n$ and $m \in \mathbb{N}$ clauses $C_1, \ldots, C_m$, where each clause C_i contains exactly three literals z_i^1, z_i^2, z_i^3, each of which is either a positive variable (say x_j) or a negative variable (say $\neg x_j$). For each variable x_j, let p_j be the number of times it appears as a literal in its positive form, let n_j be the number of times it appears as a literal in its negative form, and let $q_j = \max\{p_j, n_j\}$.

For each variable x_j, we construct a subgraph G_j consisting of a *directed cycle* with $2q_j$ vertices $v_1^j, \ldots, v_{2q_j}^j$, and $2q_j$ edges joining new vertices $u_1^j, \ldots, u_{2q_j}^j$ to $v_1^j, \ldots, v_{2q_j}^j$. See Figure 6.2 for an example with $q_j = 4$. Assume the *dotted* arcs can be used either once or twice. Observe that any arcs postman tour must traverse G_j in one of the following ways:

1. The edges are oriented from u_i^j to v_i^j if i is odd, and from v_i^j to u_i^j if i is even. The arcs (v_i^j, v_{i+1}^j) are used twice if i is odd, and once if i is even.

2. The edges are oriented from u_i^j to v_i^j if i is even, and from v_i^j to u_i^j if i is odd. The arcs (v_i^j, v_{i+1}^j) are used twice if i is even, and once if i is odd.

To complete the construction, replace each dotted arc in the directed cycle with a copy of the subgraph on the right of Figure 6.2, which has the property that it must be traversed either once or twice from w to z, according to whether arc c is duplicated or not. We associate an edge that goes from u_i^j to v_i^j with a true assignment, and an edge that goes from v_i^j to u_i^j with a false assignment. We also associate the edges joining u_i^j to v_i^j with positive literals if i is odd, and with negative literals if i is even.

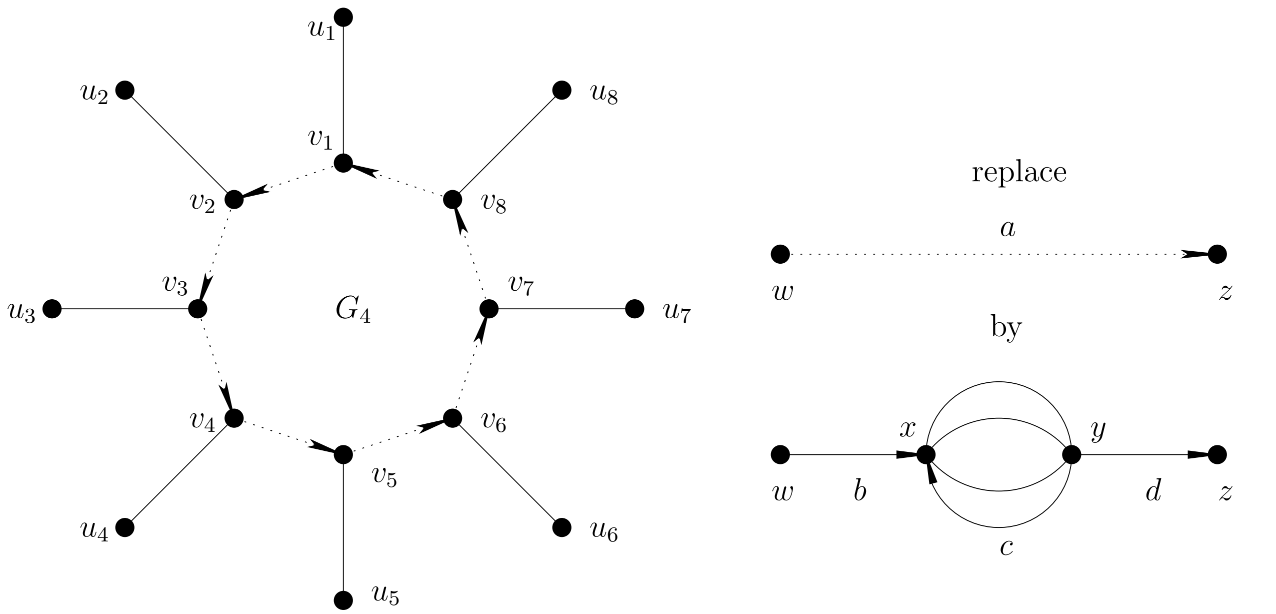

Figure 6.2: The subgraph G_j for $q_j = 4$.

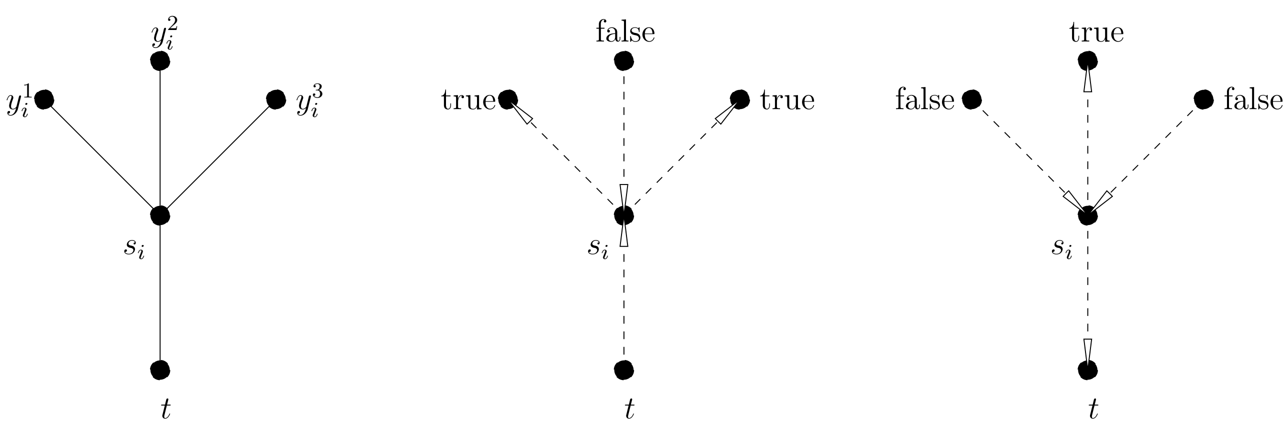

Figure 6.3: The not-all-equal clause subgraph H_i.

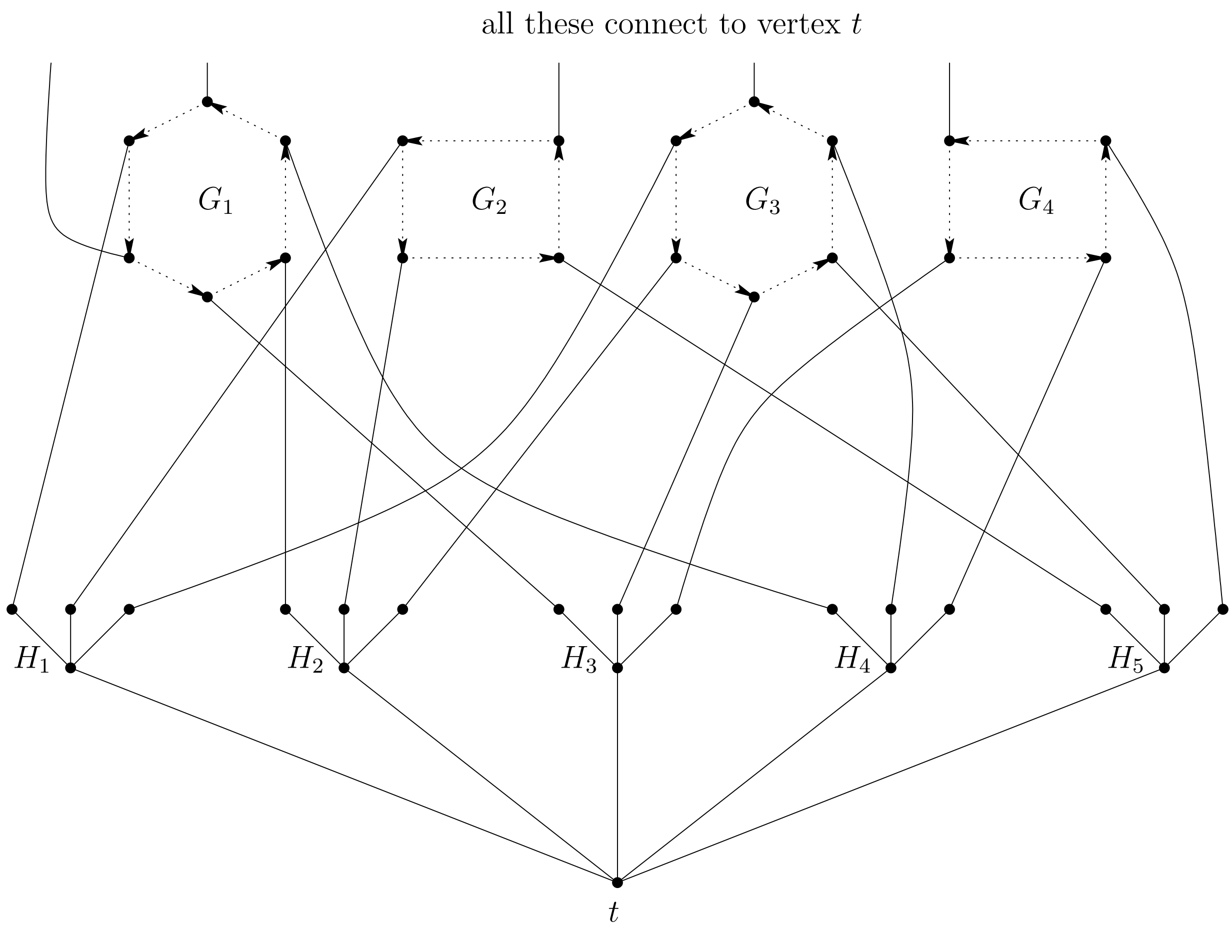

Figure 6.4: An instance of ARCS POSTMAN TOUR.

For each clause C_i, we construct a subgraph H_i consisting of a vertex s_i and four edges joining it to vertices y_i^1, y_i^2, y_i^3, and t, as in Figure 6.3. Note that in any arcs postman tour, two of these edges must enter s_i and the other two must leave s_i. In particular, in any arcs postman tour, either one or two of the edges joining s_i to y_i^1, y_i^2 and y_i^3 must leave s_i.

Let M be the mixed graph consisting of the subgraphs $G_1, \ldots, G_n$ and the subgraphs $H_1, \ldots, H_m$ where, for each $1 \leq i \leq m$ and each $1 \leq k \leq 3$, if z_i^k is a literal of variable x_j, vertex y_i^k has been identified with one of the vertices u_l^j with l odd if z_i^k is a positive literal, and l even otherwise. For each $1 \leq i \leq m$ and each $1 \leq k \leq 3$, if y_i^k has not been identified yet, we identify it with the common vertex t. See Figure 6.4.

We claim that M has an arcs postman tour if and only if I is a yes instance of NOT ALL EQUAL SATISFIABILITY. By the properties of the G_j and the H_i, if M has an arcs postman tour, then there is an assignment of the variables for which every clause contains either one or two true literals and, conversely, if there is such an assignment, then we can orient all edges in M and duplicate some of its arcs to obtain an arcs postman tour. Finally, we observe that M has size polynomial in the size of I. ■

A consequence of our result is the promised strengthening of Theorem 3.5 about the inapproximability of MINIMUM MIXED POSTMAN SET.

Theorem 6.2 *For any function $\alpha : \mathbb{N} \to \mathbb{R}$ (on the size of the input), there is no α-approximation algorithm for* MINIMUM MIXED POSTMAN SET*, unless $\mathcal{P} = \mathcal{NP}$.*

Proof. Assume that, for some function $\alpha : \mathbb{N} \to \mathbb{R}$, there exists an α-approximation algorithm $\mathcal{A}$ for MINIMUM MIXED POSTMAN SET. Let $M = (V, E, A)$ be a strongly connected mixed graph, and define a vector $c \in \mathbb{Q}_+^{E \cup A}$ as $c_e = 1$ for all $e \in E$, and $c_a = 0$ for all $a \in A$. Let $z = \mathcal{A}(M, c)$, and let $z^* = \text{MMPSP}(M, c)$. If M has an arcs postman tour, then M has a postman set with cost 0, and hence $0 \leq z \leq \alpha(M, c) z^* = 0$, that is, $z = 0$. If M does not have an arcs postman tour, then $z \geq z^* \geq 1$. Therefore, M has an arcs postman tour if and only if $z = 0$, and we can decide this in polynomial time. This is a contradiction, unless $\mathcal{P} = \mathcal{NP}$. ■

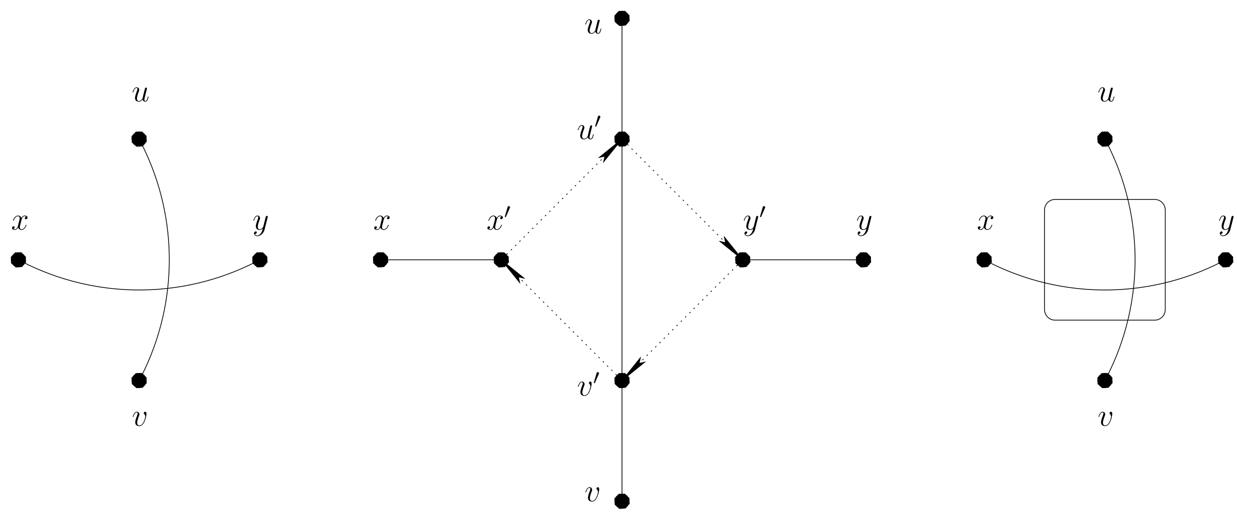

Figure 6.5: The uncrossing subgraph.

6.2.1 Planar Case

In the problems studied in previous chapters, it was possible to modify the proof of $\mathcal{NP}$-completeness to show that the respective problem remained $\mathcal{NP}$-complete even if restricted to planar mixed graphs, using that PLANAR 3-SATISFIABILITY and PLANAR 1-IN-3 SATISFIABILITY are $\mathcal{NP}$-complete (Theorems 1.3 and 1.6, respectively). However, PLANAR NOT ALL EQUAL SATISFIABILITY is solvable in polynomial time, and hence this strategy does not work for ARCS POSTMAN TOUR. Nevertheless, we can prove that ARCS POSTMAN TOUR remains $\mathcal{NP}$-complete even if restricted to planar mixed graphs, by removing the non-planarity from the construction in the proof of Theorem 6.1.

Theorem 6.3 ARCS POSTMAN TOUR *is $\mathcal{NP}$-complete, even if the input mixed graph M is restricted to be planar.*

Proof. Consider the reduction given in the proof of Theorem 6.1. It is enough to show that it is possible to obtain in polynomial time a plane mixed graph M' with the property that M has an arcs postman tour if and only if M' does. Draw the mixed graph M on the plane in such a way that the only crossings involve pairs of edges leaving variable all these connect to vertex t

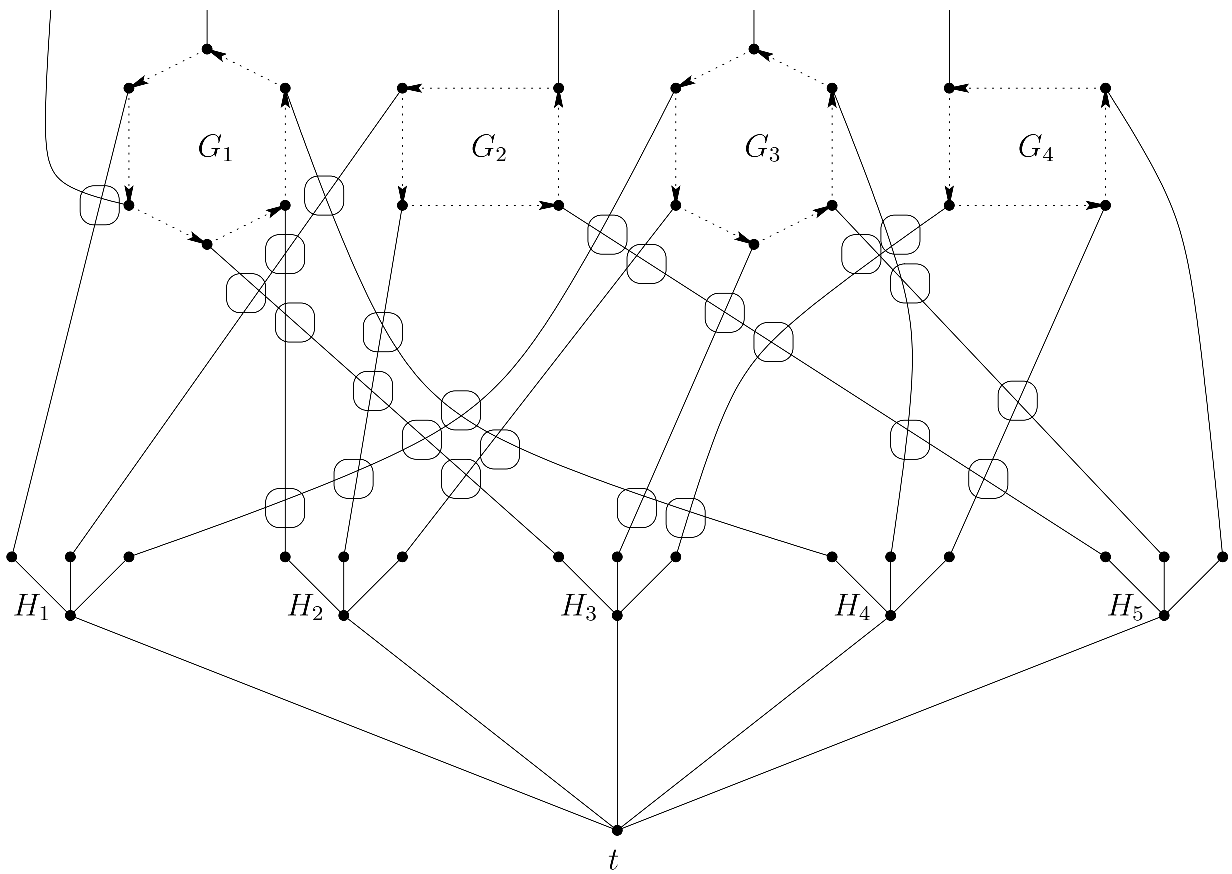

Figure 6.6: A planar instance of ARCS POSTMAN TOUR.

subgraphs, and any such pair crosses at most once. The resulting drawing may contain many crosses, but not more than $\frac{1}{2}(\sum_{i=1}^{n} 2q_j)^2 \leq 18m^2$. Replace each crossing by the *uncrossing* subgraph in Figure 6.5 to obtain a plane mixed graph M' as in Figure 6.6. Recall that the dotted arcs can be used either once or twice, and that the subgraph they represent is in Figure 6.2. This procedure can be done in polynomial time.

We claim that in any arcs postman tour of M', the edges uu' and vv' are traversed in the same direction (both up or both down), and the edges xx' and yy' are traversed in the same direction (both left or both right). Hence they replicate the behaviour of the two crossing edges uv and xy. Since the degree of u' is even, both arcs (x', u') and (u', y') must be used the same number of times, and similarly for arcs (y', v') and (v', x'). Also, since the degree of x' is odd, the arcs (v', x') and (x', u') must be used a different number of times, and similarly for arcs (u', y') and (y', v'). From this, it is easy to see that if xx' is oriented to the right, then the two arcs (x', u') and (u', y') must be used twice, the two arcs (y', v') and (v', x') must be used once, edge yy' is oriented to the right, and the three edges uu', $u'v'$, and vv' are traversed in the same direction (all up or all down). Conversely, if xx' is oriented to the left, then the two arcs (x', u') and (u', y') must be used once, the two arcs (y', v') and (v', x') must be used twice, edge yy' is oriented to the left, and the three edges uu', $u'v'$, and vv' are traversed in the same direction (all up or all down).

It follows that M' has the desired properties. ■

We observe that, if we assign a unit cost to each arc of M', then all arcs postman tours of M' have the same cost. Hence all of them are optimal.

Corollary 6.4 *The decision version of* MINIMUM ARCS POSTMAN TOUR *with unit costs is $\mathcal{NP}$-complete, even if the input mixed graph M is restricted to be planar.*

6.3 Necessary Conditions

In this section we study some necessary conditions that a mixed graph must satisfy in order to have an arcs postman tour. Note that since ARCS POSTMAN TOUR is $\mathcal{NP}$-complete, we cannot expect to obtain a complete (and short) list of necessary and sufficient conditions. Let $M = (V, E, A)$ be a mixed graph, and let $I \subseteq E$. If we assign some direction to the edges in I we obtain a *partial orientation* N of M, which we denote as $N = (V, I, E \setminus I, A)$. Since the edges in I have been assigned a direction, we call them *oriented edges*. Sometimes, this direction is forced by some condition, and hence we also call them *implied edges*. An arcs postman tour of N is an arcs postman tour of M where the edges in I have been traversed in the direction indicated by I.

6.3.1 Miscellaneous Conditions

We describe some necessary conditions for feasibility due to Veerasamy [85]. First, if M has an arcs postman tour then M must be strongly connected. We can obtain other necessary conditions as follows: Let $S \subseteq V$. We say that an arc or edge e *crosses* the cut induced by S if e has one end in S and the other end in $\bar{S}$, that is, if $e \in \delta_M(S)$. We say that an arc a *leaves* S if $a \in \delta_A(S)$, and that it enters S if $a \in \delta_A(\bar{S})$. We say that S is *outgoing* if no arc enters S, and that it is *undirected* if no arc crosses the cut induced by S.

Theorem 6.5 (Veerasamy) *If a mixed graph $M = (V, E, A)$ has an arcs postman tour then it must be connected and satisfy the following two conditions:*

$$d_E(S) \geq d_A(S) \text{ for all outgoing } S \subseteq V, \text{ and} \tag{6.2}$$

$$d_E(S) \text{ is even for all undirected } S \subseteq V. \tag{6.3}$$

Proof. If M is not connected, it does not have a tour. Now assume that M has an arcs postman tour T. Let $S \subseteq V$ be an outgoing set. Note that T must leave S at least $d_A(S)$ times, but it can only enter S at most $d_E(S)$ times. Hence $d_E(S) \geq d_A(S)$. Let $S \subseteq V$ be an undirected set. Since T must leave S the same number of times as it enters S, and since these two numbers add up to $d_E(S)$, it follows that $d_E(S)$ must be even. We note that condition (6.2) and M being connected imply that M is strongly connected. ■

Lemma 6.6 *Conditions (6.2) and (6.3) can be tested in polynomial time.*

Proof. Let $M = (V, E, A)$ be a mixed graph, and let $M^* = (V, E \cup A)$ be any orientation of M. To test condition (6.2), define vectors $l, u \in \mathbb{Z}^{E \cup A}$ as $l_e = -1$ and $u_e = 1$ for all

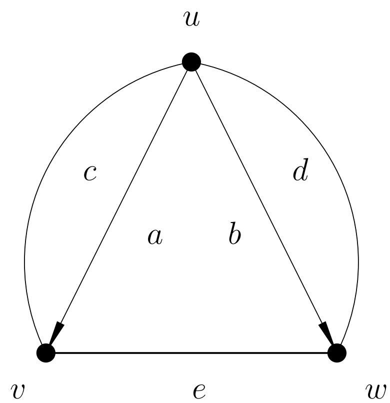

Figure 6.7: A mixed graph with no arcs postman tour.

$e \in E$, and $l_a = 1$ and $u_a = +\infty$ for all $a \in A$. Note that the incidence vector of any arcs postman tour of M is an integral circulation of (M^*, l, u). By Theorems 2.11 and 2.12, (M^*, l, u) has a circulation $x \in \mathbb{Z}^{E \cup A}$ if and only if $u(\delta^*(\bar{S})) \geq l(\delta^*(S))$ for all $S \subseteq V$, and this can be decided in polynomial time. If S is not outgoing, then $u(\delta^*(\bar{S})) = +\infty$, and the condition holds. If S is outgoing, then $u(\delta^*(\bar{S})) = d^*_E(\bar{S})$ and $l(\delta^*(S)) = d^*_A(S) - d^*_E(S)$, and the condition is $d^*_E(S) + d^*_E(\bar{S}) \geq d^*_A(S)$, that is, $d_E(S) \geq d_A(S)$. To test condition (6.3), observe that arcs do not belong to any undirected cut, and hence, the mixed graph M' obtained from contracting all arcs in M has the same set of undirected cuts. Note that all cuts of M' have even cardinality if and only if all its vertices have even degree, and that this can be tested in linear time. ■

We note that conditions (6.2) and (6.3) are independent: K_2 satisfies (6.2) but not (6.3), and an orientation of K_2 satisfies (6.3) but not (6.2). Furthermore, these conditions are not sufficient: In Figure 6.7, we show a connected mixed graph that satisfies both conditions (6.2) and (6.3), but does not have an arcs postman tour.

We call (6.2) the *outgoing-set condition*, and (6.3) the *even-cut condition*. We can generalize these conditions to partial orientations. Let $N = (V, I, E, A)$ be a partial orientation of a mixed graph. For $S \subseteq V$, let $\delta_I(S) \subseteq I$ be the set of edges oriented from a vertex in S to a vertex in $\bar{S}$, and denote by $d_I(S)$ its cardinality. We say that $e \in I$ *crosses* the cut induced by S if $e \in \delta_I(S) \cup \delta_I(\bar{S})$, that it *leaves* S if $e \in \delta_I(S)$, and that it *enters* S if $e \in \delta_I(\bar{S})$. Let $\delta_N(S) = \delta_A(S) \cup \delta_A(\bar{S}) \cup \delta_E(S) \cup \delta_I(S) \cup \delta_I(\bar{S})$ and let $d_N(S)$ be its cardinality. Observe that orienting an edge does not change the set of outgoing or even subsets of V. The proof of the following is very similar to that of Theorem 6.5 and Lemma 6.6, and hence it is omitted.

Corollary 6.7 *If a partial orientation $N = (V, I, E, A)$ of a mixed graph has an arcs postman tour then it must be connected and satisfy the following two conditions:*

$$d_E(S) + d_I(\bar{S}) \geq d_A(S) + d_I(S) \text{ for all outgoing } S \subseteq V, \text{ and} \tag{6.4}$$

$$d_E(S) + d_I(S) + d_I(\bar{S}) \text{ is even for all undirected } S \subseteq V. \tag{6.5}$$

Furthermore, (6.4) and (6.5) can be tested in polynomial time.

We call (6.4) the *(generalized) outgoing-set condition*, and (6.5) the *(generalized) even-cut condition*. Again, these conditions are independent, and not sufficient for feasibility.

Let $M = (V, E, A)$ be a mixed graph. We say that $S \subseteq V$ has the *parity* of $d_M(S)$. For every outgoing $S \subseteq V$ define $\mathrm{sur}_M(S) = d_E(S) - d_A(S)$ to be the *surplus* of S. Note that condition (6.2) can be rewritten as $\mathrm{sur}_M(S) \geq 0$ for all outgoing $S \subseteq V$. Also note that an outgoing set has the same parity as its surplus. Similarly, let $N = (V, I, E, A)$ be a partial orientation of a mixed graph. We say that $S \subseteq V$ has the *parity* of $d_N(S)$. For every outgoing $S \subseteq V$ define $\mathrm{sur}_N(S) = d_E(S) + d_I(\bar{S}) - d_A(S) - d_I(S)$ to be the *surplus* of S. Now, condition (6.4) can be rewritten as $\mathrm{sur}_N(S) \geq 0$ for all outgoing $S \subseteq V$. If $S, T \subseteq V$ are outgoing, a straightforward calculation shows that

$$\mathrm{sur}_N(S) + \mathrm{sur}_N(T) = \mathrm{sur}_N(S \cup T) + \mathrm{sur}_N(S \cap T) + 2|E(S \setminus T, T \setminus S)|. \tag{6.6}$$

6.3.2 Two Outgoing Sets

Let $N = (V, I, \emptyset, A)$ be a partial orientation of the mixed graph $M = (V, E, A)$ in which all edges have been oriented in such a way that N has an arcs postman tour. Let $S \subseteq V$ be outgoing. Using the generalized outgoing-set condition (6.4) we obtain

$$d_I(\bar{S}) \geq d_A(S) + d_I(S). \tag{6.7}$$

Adding $d_I(S)$ to both sides, we obtain

$$d_E(S) = d_I(S) + d_I(\bar{S}) \geq d_A(S) + 2d_I(S). \tag{6.8}$$

Solving for $d_I(S)$ we conclude that

$$d_I(S) \leq \lfloor \tfrac{1}{2}(d_E(S) - d_A(S)) \rfloor = \lfloor \tfrac{1}{2}\mathrm{sur}_M(S) \rfloor. \tag{6.9}$$

Similarly, if $T \subseteq V$ is also outgoing, we obtain

$$d_I(T) \leq \lfloor \tfrac{1}{2}\mathrm{sur}_M(T) \rfloor. \tag{6.10}$$

Adding these two inequalities we get

$$d_I(S) + d_I(T) \leq \lfloor \tfrac{1}{2}\mathrm{sur}_M(S) \rfloor + \lfloor \tfrac{1}{2}\mathrm{sur}_M(T) \rfloor. \tag{6.11}$$

Now note that the left-hand side is the number of oriented edges that leave S and T (an oriented edge is counted twice if it leaves both sets). We can obtain a lower bound for this number by observing that oriented edges with one end in $S \setminus T$ and the other end in $T \setminus S$ must leave exactly one of S and T. Hence, we have proved the following:

Theorem 6.8 *Let $M = (V, E, A)$ be a mixed graph with an arcs postman tour, and let $S, T \subseteq V$ be two outgoing sets. Then*

$$|E(S \setminus T, T \setminus S)| \leq \lfloor \tfrac{1}{2} sur_M(S) \rfloor + \lfloor \tfrac{1}{2} sur_M(T) \rfloor. \tag{6.12}$$

We call (6.12) the 2-*outgoing-sets condition*. We show that this condition is a common generalization of the outgoing-set condition and the even-cut condition.

Theorem 6.9 *Let $M = (V, E, A)$ be a mixed graph that satisfies the 2-outgoing-sets condition. Then M satisfies the outgoing-set and the even-cut conditions.*

Proof. Let $S \subseteq V$ be an outgoing set. Then, by (6.12):

$$0 = |E(S \setminus S, S \setminus S)| \leq \lfloor \tfrac{1}{2}\mathrm{sur}_M(S) \rfloor + \lfloor \tfrac{1}{2}\mathrm{sur}_M(S) \rfloor = 2\lfloor \tfrac{1}{2}\mathrm{sur}_M(S) \rfloor, \tag{6.13}$$

that is, $\mathrm{sur}_M(S) \geq 0$, the outgoing-set condition. Now, let $S \subseteq V$ be an undirected set. Then both S and $\bar{S}$ are outgoing, and by (6.12):

$$d_E(S) = |E(S \setminus \bar{S}, \bar{S} \setminus S)| \leq \lfloor \tfrac{1}{2}\mathrm{sur}_M(S) \rfloor + \lfloor \tfrac{1}{2}\mathrm{sur}_M(\bar{S}) \rfloor = 2\lfloor \tfrac{1}{2} d_E(S) \rfloor, \tag{6.14}$$

which can only happen if $d_E(S)$ is even, the even-cut condition. ■

In fact, condition (6.12) is stronger than conditions (6.2) and (6.3): Take $S = \{u, v\}$ and $T = \{u, w\}$ in the mixed graph of Figure 6.7 to obtain the contradiction $1 \leq 0$.

Just as before, we can generalize these results to partial orientations of mixed graphs.

Corollary 6.10 *Let $N = (V, I, E, A)$ be a partial orientation of a mixed graph with an arcs postman tour, and let $S, T \subseteq V$ be two outgoing sets. Then*

$$|E(S \setminus T, T \setminus S)| \leq \lfloor \tfrac{1}{2} sur_N(S) \rfloor + \lfloor \tfrac{1}{2} sur_N(T) \rfloor. \tag{6.15}$$

We call (6.15) the *(generalized) 2-outgoing-sets condition*.

Corollary 6.11 *Let $N = (V, I, E, A)$ be a partial orientation of a mixed graph that satisfies the generalized 2-outgoing-sets condition. Then N satisfies the generalized outgoing-set condition and the generalized even-cut condition.*

6.3.3 Testing the 2-Outgoing-Sets Condition

Now we describe how to verify in polynomial time whether the generalized 2-outgoing-sets condition holds. We say that a pair (S, T) of outgoing sets $S, T \subseteq V$ is *tight* for the mixed graph M if (6.12) holds with equality. We say that an outgoing set $S \subseteq V$ is *tight* for M if (S, S) is tight for M. Equivalently, S is tight for M if $\lfloor \frac{1}{2}\mathrm{sur}_M(S) \rfloor = 0$. Note that if S is a tight *even* set then $\mathrm{sur}_M(S) = 0$, and that if S is a tight *odd* set then $\mathrm{sur}_M(S) = 1$. If $v \in V$, we say that v is *tight* for M if $\{v\}$ is tight for M, and we say that $\bar{v}$ is *tight* for M if $V \setminus v$ is tight for M. Similarly, we say that a pair (S, T) of outgoing sets $S, T \subseteq V$ is *tight* for the partial orientation N if (6.15) holds with equality, and we say that an outgoing set $S \subseteq V$ is *tight* for N if $\lfloor \frac{1}{2}\mathrm{sur}_N(S) \rfloor = 0$. If $v \in V$, we say that v is *tight* for N if $\{v\}$ is tight for N, and we say that $\bar{v}$ is *tight* for N if $V \setminus v$ is tight for N.

The following two lemmas, which we state without proof, describe operations involving tight even sets that preserve feasibility.

Lemma 6.12 *Assume that $S \subseteq V$ is a tight even set of N such that both $N[S]$ and $N[\bar{S}]$ are connected. Let N_1 and N_2 be the partial orientations obtained from N by contracting all edges and arcs in $N[\bar{S}]$ and $N[S]$, respectively. Then N has an arcs postman tour if and only if both N_1 and N_2 have arcs postman tours.*

Lemma 6.13 *Assume that $S \subseteq V$ is a tight even set of N. Let $a \in \delta_A(S) \cup \delta_I(S)$, let $e \in \delta_E(S) \cup \delta_I(\bar{S})$, and let $N' = (V, I \cup i, E \setminus e, A \setminus a)$ be the partial orientation obtained from N by deleting a and e, and adding an edge i oriented from the tail of a to the end of e in S. Then N has an arcs postman tour if and only if each component of N' does.*

We say that N is *simplified* if, for all sets S satisfying the conditions of Lemma 6.12, either $|S| \leq 1$ or $|\bar{S}| \leq 1$. Note that, in this case, at least one of N_1 and N_2 is equal to N.

Theorem 6.14 *There exists a polynomial-time algorithm that, with input a partial orientation $N = (V, I, E, A)$ of a connected mixed graph, outputs correctly either that N has no arcs postman tour, or a list of simplified partial orientations $N_1, \ldots, N_k$, each of them satisfying the generalized 2-outgoing-sets condition (6.15), such that N has an arcs postman tour if and only if all of $N_1, \ldots, N_k$ do, and $\sum_{i=1}^{k} size(N_i) \leq size(N)$.*

Proof. By Corollary 6.7, we can decide in polynomial time whether N satisfies condition (6.4) or not, and hence we can assume that it does. For outgoing sets $S, T \subseteq V$, condition (6.15) is equivalent to $\mathrm{sur}_N(S) + \mathrm{sur}_N(T) \geq k + 2|E(S \setminus T, T \setminus S)|$, where $k = 0$ if both S and T are even, $k = 1$ if S and T have different parity, and $k = 2$ if both S and T are odd, which by (6.6) is equivalent to:

$$\mathrm{sur}_N(S \cup T) + \mathrm{sur}_N(S \cap T) \geq k. \tag{6.16}$$

Since N satisfies condition (6.4), this inequality holds if both S and T are even (since $\mathrm{sur}_N(S \cup T) \geq 0$ and $\mathrm{sur}_N(S \cap T) \geq 0$) and if S and T have different parity (since $S \cup T$ and $S \cap T$ have different parities, one of $\mathrm{sur}_N(S \cup T)$ and $\mathrm{sur}_N(S \cap T)$ is at least 0, and the other is at least 1). Furthermore, if both S and T are odd, then $S \cup T$ and $S \cap T$ have the same parity, and if this parity is odd then $\mathrm{sur}_N(S \cup T) \geq 1$ and $\mathrm{sur}_N(S \cap T) \geq 1$.

Hence, we only need to be able to test in polynomial time whether (6.16) holds for odd S and T such that $S \cup T$ and $S \cap T$ are even. Equivalently, we need to be able to test in polynomial time whether there exist odd outgoing sets S and T such that both $S \cup T$ and $S \cap T$ are tight even sets, violating (6.16). We call such a pair (S, T) a *bad* pair. Observe that if (S, T) is a bad pair then both $S \setminus T$ and $T \setminus S$ are odd.

Assume first that N is simplified. Let $G = (V, A)$ be the underlying undirected graph of $D = (V, A)$, let R be the set of vertices of odd degree of N, and define a vector $x \in \mathbb{Z}_+^A$ by $x_a = 0$ if a is incident with a vertex v such that v or $\bar{v}$ is tight, and $x_a = 1$ otherwise. We claim that there exists a bad pair if and only if there exists $U \subseteq V$ with $|U \cap R|$ odd and $x(\delta_G(U)) = 0$. Observe that we can decide the latter in polynomial time using the algorithm in Theorem 2.9. Also note that $|U \cap R|$ is odd if and only if U is odd. Suppose first that (S, T) is a bad pair. Since $S \cap T$ and $S \cup T$ are tight, then each $v \in S \cap T$ is tight, and each $v \in \overline{S \cup T}$ has $\bar{v}$ tight. Let $U = S \setminus T$. If $e \in \delta_G(U)$, then e must be incident with at least one vertex in $(S \cap T) \cup (\overline{S \cup T})$, which implies $x_e = 0$. Therefore $x(\delta_G(U)) = 0$, and U satisfies the required properties. Now suppose that $U \subseteq V$ is odd, with $x(\delta_G(U)) = 0$. Define the sets $P = \{v \in V : v \text{ is tight}\}$, $Q = \{v \in V : \bar{v} \text{ is tight}\}$, $S = (U \cup P) \setminus Q$, and $T = (\bar{U} \cup P) \setminus Q$. Since S and T have the same parity as U, they are both odd. Since $S \cap T = P$ and $S \cup T = \bar{Q}$, they are both tight even. Hence (S, T) is a bad pair.

If N is not simplified, we can *decompose* it recursively into simplified pieces as follows: First we test whether there exists any proper $S \subset V$ violating condition (6.4). If there is any, stop and output that N does not have an arcs postman tour. Second we test whether there exists any proper $S \subset V$ such that $\delta_N(S) \subseteq I$. Note that any such set must be tight. We can do this in polynomial time by finding a minimum weight cut in the underlying undirected graph $\bar{N}$ of N with weights $w_e = 1$ if $e \in E \cup A$, and $w_e = 0$ otherwise. If there is a set S such that $w(\bar{\delta}_N(S)) = 0$ then we pair arbitrarily the elements of $\delta_I(S)$ with the elements of $\delta_I(\bar{S})$, and apply to each pair the reduction described in Lemma 6.13. We continue recursively with the connected components obtained. This procedure stops in at most $|I|$ steps since, at each step, we reduce the number of implied edges by at least one.

We continue to test recursively each connected component $N' = (V', I', E', A')$ obtained. As before, first we test whether there exists any proper $S \subset V$ violating condition (6.4). If there is any, stop and output that N does not have an arcs postman tour. Second we test whether there exists any tight set S. Note that all such sets satisfy

$$|\delta'_N(S) \cap (A' \cup E')| > 0. \tag{6.17}$$

We can do this in polynomial time modifying the feasible flow algorithm used in the proof of Corollary 6.6. Let $N^* = (V, I' \cup E' \cup A')$ be any orientation of N such that each arc in I' is oriented exactly as in N. Let $\epsilon = \frac{1}{|A'|+|E'|}$, and define two vectors $l, u \in \mathbb{Z}^{A' \cup E' \cup I'}$ as $l_e = u_e = 1$ if $e \in I'$, $l_e = \epsilon - 1$ and $u_e = 1 - \epsilon$ if $e \in E'$, and $l_e = 1 + \epsilon$ and $u_e = +\infty$ if $e \in A$. We claim that there exists an integral circulation $l \leq x \leq u$ of (N^*, l, u) if and only if there is no tight $S \subset V$. First, if $S \subset V$ is tight then (6.17) implies that:

$$\begin{array}{rcll} 0 & < & d'_E(S) + d'_A(S) & (6.18) \\ 0 & < & d^*_E(S) + d^*_E(\bar{S}) + d^*_A(S) & (6.19) \\ -\epsilon d^*_E(\bar{S}) & < & \epsilon d^*_E(S) + \epsilon d^*_A(S) & (6.20) \\ (1-\epsilon) d^*_E(\bar{S}) + d^*_I(\bar{S}) & < & (\epsilon - 1) d^*_E(S) + d^*_I(S) + (1+\epsilon) d^*_A(S) & (6.21) \\ u(\delta^*_N(\bar{S})) & < & l(\delta^*_N(S)). & (6.22) \end{array}$$

Conversely, if $S \subset V$ satisfies condition (6.4) strictly then:

$$\begin{array}{rcll} d'_E(S) + d'_I(\bar{S}) & \geq & d'_A(S) + d'_I(S) + 1 & (6.23) \\ d^*_E(S) + d^*_E(\bar{S}) + d^*_I(\bar{S}) & \geq & d^*_A(S) + d^*_I(S) + 1 & (6.24) \\ d^*_E(S) + d^*_E(\bar{S}) + d^*_I(\bar{S}) & \geq & d^*_A(S) + d^*_I(S) + \epsilon(|A^*| + |E^*|) & (6.25) \\ (1-\epsilon) d^*_E(\bar{S}) + d^*_I(\bar{S}) & \geq & (\epsilon - 1) d^*_E(S) + d^*_I(S) + (1+\epsilon) d^*_A(S) & (6.26) \\ u(\delta^*_N(\bar{S})) & \geq & l(\delta^*_N(S)). & (6.27) \end{array}$$

If we find a tight set S, we pair arbitrarily the elements of $\delta'_A(S) \cup \delta'_I(S)$ with the elements of $\delta'_E(S) \cup \delta'_I(\bar{S})$, and apply to each pair the reduction described in Lemma 6.13. We continue recursively with the connected components obtained. As before, this procedure must stop in at most $|A| + |E| + |I|$ steps. ∎

Corollary 6.15 *There exists a polynomial-time algorithm that, with input a connected mixed graph M, outputs correctly either that M has no arcs postman tour, or a list of simplified partial orientations $N_1, \ldots, N_k$, each of them satisfying the generalized 2-outgoing-sets condition (6.15), such that M has an arcs postman tour if and only if all of $N_1, \ldots, N_k$ do, and $\sum_{i=1}^{k} size(N_i) \leq size(M)$.*

6.3.4 Testing the 2-Outgoing-Sets Condition on Planar Graphs

We give a simpler polynomial algorithm to decide whether a partial orientation of a planar mixed graph satisfies the generalized 2-outgoing-sets condition. We show first how to perform the reduction described in Lemma 6.13 preserving planarity. Let S be a tight set with $N[S]$ and $N[\bar{S}]$ connected, and C be a curve enclosing S in a given planar embedding of N, as in Figure 6.8. Label each $a \in \delta_A(S) \cup \delta_I(S)$ with a $+1$, and each $e \in \delta_E(S) \cup \delta_I(\bar{S})$ with a -1. Since S is tight, all these labels add up to zero. Furthermore, there must be two elements in $\delta_N(S)$ with distinct labels and consecutive along C. In Figure 6.8, we show in bold edges some possible pairs. Observe that if we apply the reduction to these two elements, we obtain a planar embedding for N'.

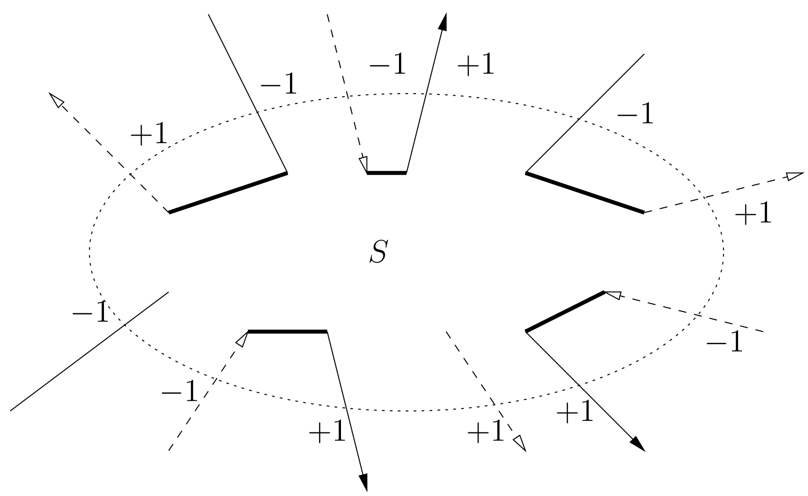

Figure 6.8: A region on the plane containing a tight set and a pairing.

Our algorithm for planar mixed graphs differs from our general algorithm in the way it verifies the outgoing-set condition, and in the way it finds tight sets. One of these procedures is based in the following result about cycles in a directed graph [60]:

Theorem 6.16 (Karp) *Let $D = (V, A)$ be a directed graph, and let $l \in \mathbb{Z}^A$ be a vector of* lengths*. For each simple directed cycle $C = (W, B)$ of D with at least one arc, let $m(C) = \frac{1}{|B|} l(B)$ be its* mean length*. There exists a polynomial-time algorithm to find a simple directed cycle C^* of D with at least one arc, and minimum mean length.*

Let $N = (V, I, E, A)$ be a partial orientation of a connected *plane* mixed graph. The *dual* of N is a partial orientation N^* of a connected plane mixed graph constructed as follows: Let $G = (V, A \cup E \cup I)$ be the underlying undirected plane graph of N, let F be its set of faces, and let $G^* = (F, A \cup E \cup I)$ be its dual. To obtain N^* from G^*, replace each edge $a \in A$ in G^* by an arc a in N^* oriented from the face *to the right* of a in N to the face *to the left* of a in N. Similarly, replace each edge $i \in I$ in G^* by an implied edge i in N^* oriented from the face *to the right* of i in N to the face *to the left* of i in N. See Figure 6.9 for an example where the vertices of N are dots ($\bullet$) and the vertices of N^* are squares ($\blacksquare$). An interesting property of duals of partial orientations constructed in this way is that if S is an outgoing set in N, then the cut $\delta_N(S)$ is the disjoint union of some cycles in N^*, where implied edges may be traversed in either direction.

Theorem 6.17 *There exists a polynomial-time algorithm that, with input a partial orientation $N = (V, I, E, A)$ of a connected planar mixed graph, outputs correctly either that N has no arcs postman tour, or a list of simplified planar partial orientations $N_1, \ldots, N_k$, each of them satisfying the generalized 2-outgoing-sets condition (6.15), such that N has an arcs postman tour if and only if all of $N_1, \ldots, N_k$ do, and $\sum_{i=1}^{k} size(N_i) \leq size(N)$.*

Proof. Let $N^* = (F, I, E, A)$ be the dual of N, and let $D^* = (F, A \cup \vec{E} \cup \vec{I})$ be the associated directed graph of N^*, where $\vec{E} = \{e^+, e^- : e \in E\}$, $\vec{I} = \{i^+, i^- : i \in I\}$, and i^+ is oriented as i. Define a vector w of weights in the arcs of D^* as $w_a = -1$ if $a \in A$, $w_{e^+} = w_{e^-} = 1$ if $e \in E$, and $w_{i^+} = -1$, $w_{i^-} = 1$ if $i \in I$. See Figure 6.10 for an example. Observe that D^* has a cycle of negative weight if and only if N fails condition (6.4). Hence, we can replace the use of the feasible flow algorithm at the beginning of the proof of Theorem 6.14 by the minimum mean length algorithm of Theorem 6.16.

Assume that N satisfies condition (6.4). In order to *simplify* N, we need to be able to detect cycles of length zero in N^*. A straightforward application of Theorem 6.16 fails due to the cycles corresponding with implied edges. Since Karp's algorithm uses dynamic programming, it is possible to modify it in order to consider only cycles with at least three arcs. Alternatively, since N has no cycles of negative length, we can use a shortest path algorithm (such as the ones in [20, 37, 88]) as follows: For each $e \in A \cup E \cup I$, we delete its corresponding arcs from N^*, compute shortest paths between its ends (in both directions),

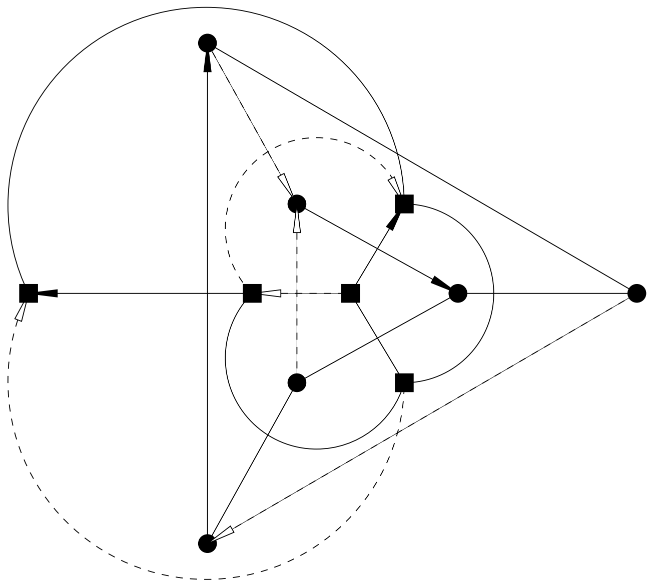

Figure 6.9: A partial orientation and its dual.

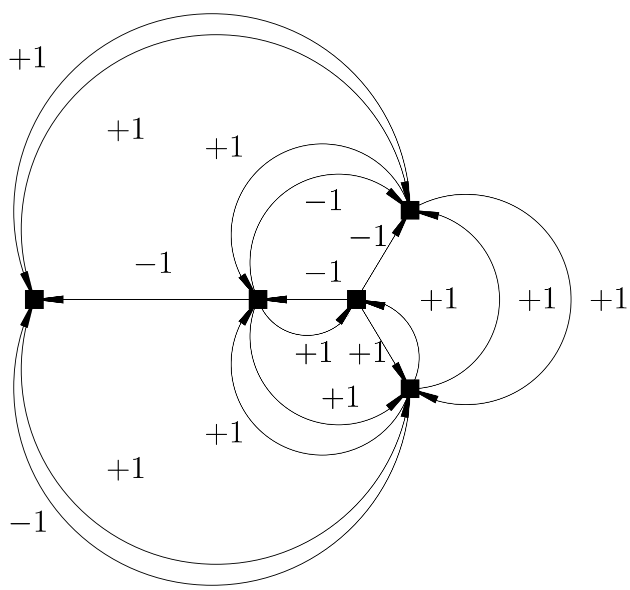

Figure 6.10: The associated directed graph of the dual.

and verify whether any of the deleted arcs forms a cycle of length zero together with one of the computed shortest paths. Any such cycle of length zero separates the vertex set of N into parts S and $\bar{S}$, at least one of them being outgoing and tight. Pair the elements of $\delta_N(S)$, as described above in order to preserve planarity, and apply to each pair the reduction described in Lemma 6.13. Continue recursively with the connected components obtained. This procedure must stop in at most $|V|$ steps because, at every step, we increase the number of connected components without increasing the number of vertices. Once we have simplified N, we apply the procedure described in the proof of Theorem 6.14. ■

Corollary 6.18 *There exists a polynomial-time algorithm that, with input a connected planar mixed graph M, outputs correctly either that M has no arcs postman tour, or a list of simplified planar partial orientations $N_1, \ldots, N_k$, each of them satisfying the generalized 2-outgoing-sets condition (6.15), such that M has an arcs postman tour if and only if all of $N_1, \ldots, N_k$ do, and $\sum_{i=1}^{k} size(N_i) \leq size(M)$.*

6.3.5 Many Outgoing Sets

Since ARCS POSTMAN TOUR is $\mathcal{NP}$-complete, and the 2-outgoing-sets condition can be tested in polynomial time, we cannot expect that these conditions are also sufficient for a mixed graph to have an arcs postman tour. In fact, we show in Figure 6.11 an example of a planar mixed graph that satisfies the 2-outgoing-sets condition, but does not have an arcs postman tour. With this in mind, we have two different options: We can try to find a class of mixed graphs for which the 2-outgoing-sets condition is sufficient, or we can try to find other necessary conditions for having arcs postman tours. Here we address the second option, while later in this chapter we address the first option.

We introduce a generalization of the 2-outgoing-sets condition. For each $k \in \mathbb{N}$, define the *Sperner graph of order* k to be the undirected graph $\mathcal{SP}_k = (\mathcal{V}_k, \mathcal{E}_k)$, with vertex set $\mathcal{V}_k = 2^{[k]}$, and edge set $\mathcal{E}_k = \{\{X, Y\} | X, Y \in \mathcal{V}_k, X \not\subseteq Y, Y \not\subseteq X\}$. The name *Sperner* reflects that every clique of $\mathcal{SP}_k$ defines a *Sperner system* on $[k]$ and, conversely, every Sperner system on $[k]$ defines a clique of $\mathcal{SP}_k$. We show the Sperner graphs of orders 2 and 3 in Figure 6.12. Let $M = (V, E, A)$ be a mixed graph, and let $N = (V, I, \emptyset, A)$ be a

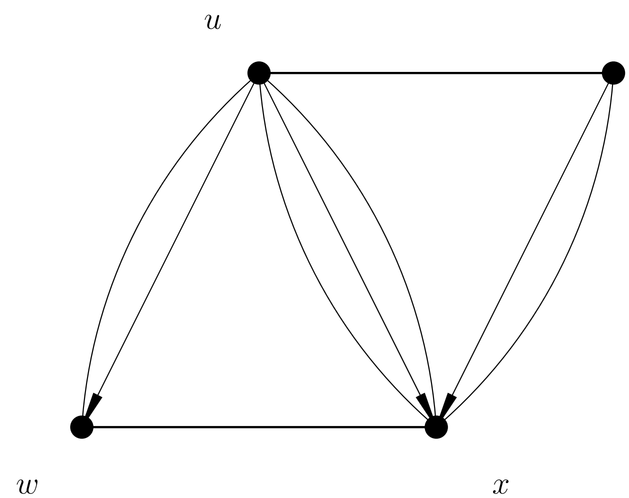

Figure 6.11: A planar mixed graph with no arcs postman tour.

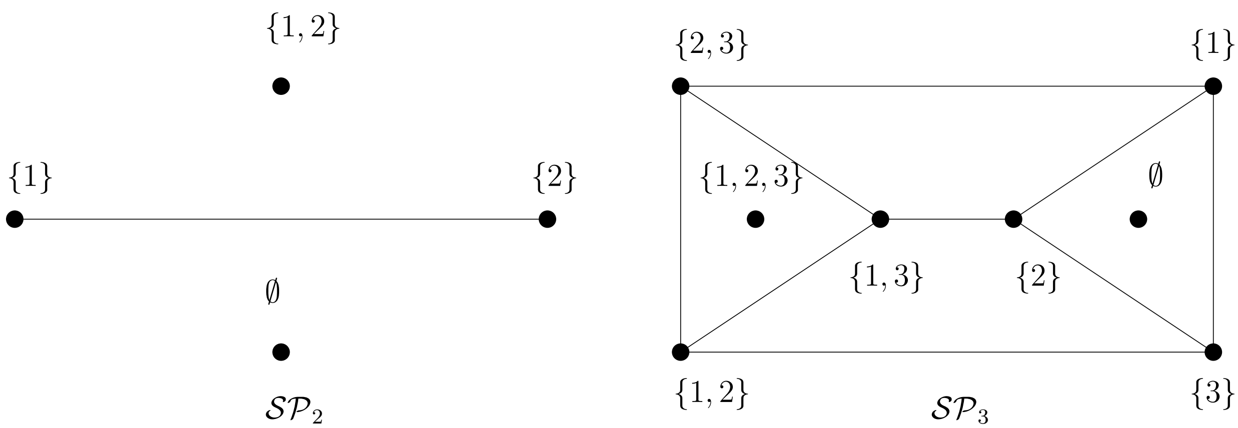

Figure 6.12: The Sperner graphs of orders 2 and 3.

partial orientation of M in which all edges have been oriented in such a way that N has an arcs postman tour. Let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a family of k outgoing subsets of V. For each $X \subseteq [k]$, we define the vertex sets

$$S_X = \bigcap_{i \in X} S_i \bigcap_{i \notin X} \bar{S}_i, \tag{6.28}$$

and we define the edge set

$$E_k^{\mathcal{S}} = \bigcup_{\{X,Y\} \in \mathcal{E}_k} E(S_X, S_Y). \tag{6.29}$$

As in the proof of Theorem 6.8, for each $1 \le i \le k$ we have that

$$d_I(S_i) \le \lfloor \tfrac{1}{2} \mathrm{sur}_M(S_i) \rfloor, \tag{6.30}$$

and adding all these inequalities, we obtain

$$\sum_{i=1}^{k} d_I(S_i) \le \sum_{i=1}^{k} \lfloor \tfrac{1}{2} \mathrm{sur}_M(S_i) \rfloor, \tag{6.31}$$

an upper bound for the number of edges that leave some S_i. To obtain a lower bound for this quantity, observe that if $e \in E(S_X, S_Y)$ (with $e = \{u, v\}$) for some $\{X, Y\} \in \mathcal{E}_k$, then there exist $i \in X \setminus Y$ and $j \in Y \setminus X$, such that $u \in S_i \cap \bar{S}_j$ and $v \in S_j \cap \bar{S}_i$ and, since e has to leave either S_i or S_j, e will be counted at least once by the left-hand side of (6.31). Hence we have proved the following:

Theorem 6.19 *Let $M = (V, E, A)$ be a mixed graph with an arcs postman tour, and let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a family of k outgoing subsets. Then*

$$|E_k^{\mathcal{S}}| \le \sum_{i=1}^{k} \lfloor \tfrac{1}{2} sur_M(S_i) \rfloor. \tag{6.32}$$

We call (6.32) the *k-outgoing-sets condition.* Note that setting $k = 1$ or $k = 2$ in (6.32), we obtain the outgoing-set condition or the 2-outgoing-sets condition, respectively. Furthermore, (6.32) is strictly stronger than they are: Our example of Figure 6.11 does not satisfy the 3-outgoing-sets condition, as shown by the family $\mathcal{S}$ consisting of the three sets $S_1 = \{v\}$, $S_2 = \{u, w\}$, and $S_3 = \{u, v, x\}$, for which $\lfloor \frac{1}{2} \mathrm{sur}_M(S_1) \rfloor = 0$, $\lfloor \frac{1}{2} \mathrm{sur}_M(S_2) \rfloor = 1$, $\lfloor \frac{1}{2} \mathrm{sur}_M(S_3) \rfloor = 0$, $E_3^{\mathcal{S}} = \{uv, wx\}$, and hence $|E_3^{\mathcal{S}}| = 2 > 1 = \sum_{i=1}^{3} \lfloor \frac{1}{2} \mathrm{sur}_M(S_i) \rfloor$.

As is usual by now, we can easily obtain the following consequence of Theorem 6.19:

Corollary 6.20 *Let $N = (V, I, E, A)$ be a partial orientation of a mixed graph with an arcs postman tour, and let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a family of k outgoing subsets. Then*

$$|E_k^{\mathcal{S}}| \leq \sum_{i=1}^{k} \lfloor \tfrac{1}{2} sur_N(S_i) \rfloor. \tag{6.33}$$

We can obtain a strengthening of Theorem 6.19 as follows: When we obtained a lower bound for the left-hand side of (6.31), we used only the fact that e must leave at least one outgoing set. However, if both $X \setminus Y$ and $Y \setminus X$ contain more than one element (which can only happen if $k \geq 4$) this bound can be improved, since e must leave at least $|X \setminus Y|$ sets in $\mathcal{S}$ if oriented from u to v, and at least $|Y \setminus X|$ sets in $\mathcal{S}$ if oriented from v to u. Using the same notation as before, for every $e \in E$ we define the coefficients:

$$m_{\mathcal{S}}(e) = \min\{|X \setminus Y|, |Y \setminus X|\} \text{ and } m^+_{\mathcal{S}}(e) = \max\{|X \setminus Y|, |Y \setminus X|\}. \tag{6.34}$$

Theorem 6.21 *Let $M = (V, E, A)$ be a mixed graph with an arcs postman tour, and let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a family of k outgoing subsets. Then*

$$\sum_{e \in E} m_{\mathcal{S}}(e) \leq \sum_{i=1}^{k} \lfloor \tfrac{1}{2} sur_M(S_i) \rfloor. \tag{6.35}$$

Corollary 6.22 *Let $N = (V, I, E, A)$ be a partial orientation of a mixed graph with an arcs postman tour, and let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a family of k outgoing subsets. Then*

$$\sum_{e \in E} m_{\mathcal{S}}(e) \leq \sum_{i=1}^{k} \lfloor \tfrac{1}{2} sur_N(S_i) \rfloor. \tag{6.36}$$

We call (6.35) the *strong k-outgoing-sets condition.* We believe that (6.35), and possibly (6.32) as well, together with connectivity, form a set of sufficient conditions for having an arcs postman tour. Furthermore, it is likely that we need to consider only a finite number of outgoing sets depending on the size of M. To be more precise:

Conjecture 6.23 *There exists a function $f : \mathbb{N} \to \mathbb{N}$ for which the following statement is true: A connected mixed graph $M = (V, E, A)$ has an arcs postman tour if and only if it satisfies the k-outgoing-sets condition for all $1 \leq k \leq f(|V|)$.*

Conjecture 6.24 *A connected mixed graph has an arcs postman tour if and only if it satisfies the k-outgoing-sets condition for all $k \in \mathbb{N}$.*

Conjecture 6.25 *A connected mixed graph has an arcs postman tour if and only if it satisfies the strong k-outgoing-sets condition for all $k \in \mathbb{N}$.*

Observe that Conjecture 6.23 implies Conjecture 6.24, which in turn implies Conjecture 6.25. Since we can test in polynomial time the k-outgoing-sets condition for $k \leq 2$, we may ask whether there exists, for each constant parameter $k \in \mathbb{N}$, a polynomial-time algorithm that tests the k'-outgoing-sets condition for all $k' \leq k$. We may also ask whether, for each constant parameter $k \in \mathbb{N}$, there exists an interesting class $\mathcal{M}_k$ of mixed graphs such that $M \in \mathcal{M}_k$ has an arcs postman tour if and only if M satisfies the k'-outgoing-sets condition for all $k' \leq k$. We exhibit a family of examples demonstrating simultaneously that the function f in the statement of Conjecture 6.23 is at least linear in $|V|$, and that there is no $k \in \mathbb{N}$ for which the class of series-parallel mixed graphs is a subset of $\mathcal{M}_k$.

Let $n \in \mathbb{N}$, and let $M_n = (V_n, E_n \cup F_n, A_n)$ be the series-parallel mixed graph with vertex set $V_n = \{v_i : 0 \leq i \leq 2n\}$, arc set $A_n = \{a_{2i+1} = (v_{2i+1}, v_{2i}), a_{2i+2} = (v_{2i+1}, v_{2i+2}) : 0 \leq i \leq n-1\}$, and edge set consisting of $E_n = \{e_i = \{v_i, v_{i+2}\} : 0 \leq i \leq n-2\}$, and $F_n = \{f_i : 1 \leq i \leq 2n-2\}$, where $f_1 = \{v_0, v_1\}$, $f_{2n-2} = \{v_{2n-1}, v_{2n}\}$, and for each $2 \leq j \leq n-1$, f_{2j-2}, f_{2j-1} are two edges parallel to arc a_j. See Figure 6.13 for an example with $n = 5$. Note also that Figure 6.7 corresponds with $n = 1$.

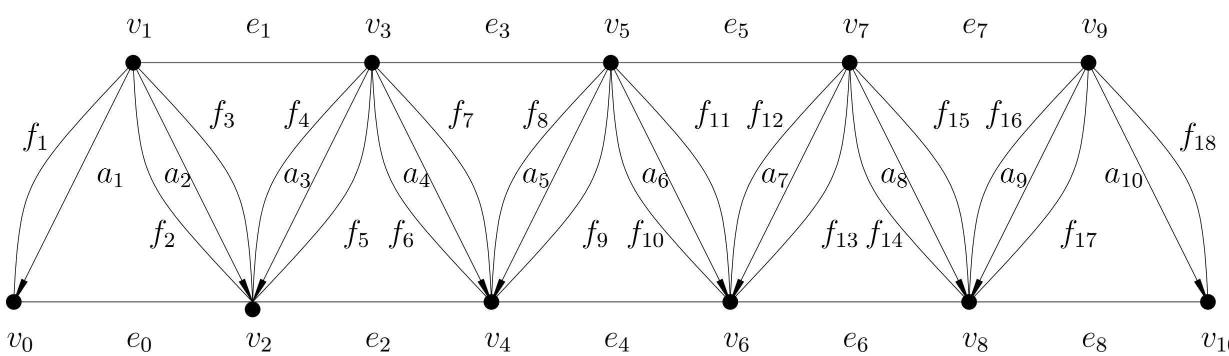

Figure 6.13: The mixed graph M_5.

Theorem 6.26 *Let $n \in \mathbb{N}$. The mixed graph M_n satisfies the k-outgoing-sets condition for each $1 \leq k \leq 2n-1$, but does not satisfy the $2n$-outgoing-sets condition.*

Proof. Let $S \subseteq V$ be an outgoing subset of M_n. We say that $v_{2j-1}, v_{2j+1}, \ldots, v_{2l+1}$ form an *odd interval* of S if $v_{2j-1}, v_{2j+1}, \ldots, v_{2l+1} \in S$, and that $v_{2j}, v_{2j+2}, \ldots, v_{2l}$ form an *even interval* of S if $v_{2j}, v_{2j+2}, \ldots, v_{2l} \in S$. Note that $v_0 \in S$ implies $v_1 \in S$, $v_{2n} \in S$ implies $v_{2n-1} \in S$ and, for all $1 \leq i \leq n-1$, $v_{2i} \in S$ implies that $v_{2i-1}, v_{2i+1} \in S$. Hence, the connected components of $M_n[S]$ are determined by the maximal odd intervals of S. Let $S^1, \ldots, S^h$ be the vertex sets of the connected components of $M_n[S]$, then:

$$\lfloor \tfrac{1}{2}\mathrm{sur}_{M_n}(S) \rfloor \geq \sum_{i=1}^{h} \lfloor \tfrac{1}{2}\mathrm{sur}_{M_n}(S^i) \rfloor, \tag{6.37}$$

and hence, if we try to construct a family $\mathcal{S}_n$ that violates (6.32), we can assume that each of its members induces a connected component of M_n. Also note that the only edges that can be counted on the left-hand side of (6.32) are in E_n, since the edges in F_n are parallel to at least one arc. Furthermore, for an edge $e \in E_n$ to be counted in this way, the family $\mathcal{S}_n$ must contain at least two outgoing sets crossed by e. Therefore, the maximal contribution of S to the left-hand side of (6.32) is $\mathrm{lhs}(S) = \frac{1}{2}d_{E_n}(S)$.

Let $v_{2j-1}, v_{2j+1}, \ldots, v_{2l+1}$ be the unique maximal odd interval of S, and let m be the number of even intervals of S. Also, let $s_i = 0$ if $v_i \in S$, and $s_i = 1$ otherwise. We split the analysis into four cases, depending on the values of j and l:

1. If $j = 1$ and $l < n-1$ then $d_{A_n}(S) = 1 + s_0 + 2\sum_{i=1}^{l} s_{2i}$, $d_{E_n}(S) = s_0 + 2m$, and $d_{F_n}(S) = 2 + s_0 + 4\sum_{i=1}^{l} s_{2i}$. Hence $\mathrm{lhs}(S) = m + \frac{1}{2}s_0$ and $\lfloor \frac{1}{2}\mathrm{sur}_{M_n}(S) \rfloor = s_0 + m + \sum_{i=1}^{l} s_{2i}$, and therefore $\mathrm{lhs}(S) < \lfloor \frac{1}{2}\mathrm{sur}_{M_n}(S) \rfloor$ unless $s_0 = s_2 = \cdots = s_{2l} = 0$, that is, unless $S = \{s_i : 0 \leq i \leq 2l+1\}$. In this case $\mathrm{lhs}(S) = \lfloor \frac{1}{2}\mathrm{sur}_{M_n}(S) \rfloor$.

2. If $j > 1$ and $l = n-1$ the situation is symmetric, and a similar analysis gives that $\mathrm{lhs}(S) < \lfloor \frac{1}{2}\mathrm{sur}_{M_n}(S) \rfloor$ unless $S = \{s_i : 2j-1 \leq i \leq 2n\}$.

3. If $j = 1$ and $l = n-1$ then $d_{A_n}(S) = s_0 + s_{2n} + 2\sum_{i=1}^{n-1} s_{2i}$, $d_{E_n}(S) = s_0 + s_{2n} + 2m - 2$, and $d_{F_n}(S) = s_0 + s_{2n} + 4\sum_{i=1}^{n-1} s_{2i}$. Hence $\mathrm{lhs}(S) = m - 1 + \frac{1}{2}(s_0 + s_{2n})$ and $\lfloor \frac{1}{2}\mathrm{sur}_{M_n}(S) \rfloor = m - 1 + \lfloor \frac{1}{2}(s_0 + s_{2n}) \rfloor + \sum_{i=1}^{n-1} s_{2i}$, and therefore $\mathrm{lhs}(S) < \lfloor \frac{1}{2}\mathrm{sur}_{M_n}(S) \rfloor$

unless $s_2 = \cdots = s_{2n-2} = 0$, that is, unless $\{s_i : 1 \le i \le 2n-1\} \subseteq S$. In this case, if $s_0 \neq s_{2n}$ then $\frac{1}{2} = \text{lhs}(S) > \lfloor \frac{1}{2}\text{sur}_{M_n}(S) \rfloor = 0$, otherwise $\text{lhs}(S) = \lfloor \frac{1}{2}\text{sur}_{M_n}(S) \rfloor$.

4. If $j > 1$ and $l < n-1$ then $d_{A_n}(S) = 2 + 2\sum_{i=j}^{l} s_{2i}$, $d_{E_n}(S) = 2 + 2m$, and $d_{F_n}(S) = 4 + 4\sum_{i=j}^{l} s_{2i}$. Hence $\text{lhs}(S) = 1 + m$ and $\lfloor \frac{1}{2}\text{sur}_{M_n}(S) \rfloor = 2 + m + \sum_{i=j}^{l} s_{2i}$, and therefore $\text{lhs}(S) < \lfloor \frac{1}{2}\text{sur}_{M_n}(S) \rfloor$ always.

For $0 \le j \le 2n$, let $L_j = \{v_i : 0 \le i \le j\}$ and $R_j = \{v_i : j \le i \le 2n\}$. Our analysis shows that the only two outgoing sets S for which $\text{lhs}(S) > \lfloor \frac{1}{2}\text{sur}_{M_n}(S) \rfloor$ are L_{2n-1} and R_1, and hence both must be included in any family $\mathcal{S}_n$ violating (6.32). For L_{2n-1} to contribute its $\frac{1}{2}$ to the left-hand side of (6.32), $\mathcal{S}_n$ must contain an outgoing set S with $v_{2n} \in S$, $v_{2n-2} \notin S$, and $\text{lhs}(S) \ge \lfloor \frac{1}{2}\text{sur}_{M_n}(S) \rfloor$, and the only such set is R_{2n-1}. Similarly, for R_1 to contribute its $\frac{1}{2}$ to the left-hand side of (6.32), $\mathcal{S}_n$ must contain an outgoing set S with $v_0 \in S$, $v_2 \notin S$, and $\text{lhs}(S) \ge \lfloor \frac{1}{2}\text{sur}_{M_n}(S) \rfloor$, and the only such set is L_1. Proceeding by induction, $\mathcal{S}_n$ must contain all of $L_1, L_3, \ldots, L_{2n-1}$ and $R_1, R_3, \ldots, R_{2n-1}$. Furthermore, the family $\mathcal{S}_n$ consisting precisely of these $2n$ outgoing sets satisfies

$$|E_{2n}^{\mathcal{S}}| = |E_n| = 2n - 1 > 2(n-1) = \sum_{i=1}^{n} \lfloor \tfrac{1}{2}\text{sur}_{M_n}(L_i) \rfloor + \sum_{i=1}^{n} \lfloor \tfrac{1}{2}\text{sur}_{M_n}(R_i) \rfloor, \tag{6.38}$$

and hence violates (6.32). ■

To close this section, we offer a characterization of mixed graphs that have an arcs postman tour for each possible orientation of their edges.

Theorem 6.27 *Let $M = (V, E, A)$ be a connected mixed graph. The following statements are equivalent:*

1. *M has an arcs postman tour for every orientation of its edges.*

2. *The only outgoing sets of M are $\emptyset$ and V.*

3. *$D = (V, A)$ is strongly connected.*

4. *For all $k \in \mathbb{N}$, and for all families $\mathcal{S} = \{S_1, \ldots, S_k\}$ of outgoing sets, M satisfies:*

$$\sum_{e \in E} m_{\mathcal{S}}^{+}(e) \le \sum_{i=1}^{k} \lfloor \tfrac{1}{2} sur_M(S_i) \rfloor. \tag{6.39}$$

Proof. ($1 \Rightarrow 2$) Let $S \subset V$, and orient all edges in $\delta_E(S)$ from S to $\bar{S}$. Since this partial orientation has an arcs postman tour, there must be arcs in $\delta_A(\bar{S})$. Hence S is not outgoing. ($2 \Rightarrow 3$) If D is not strongly connected, then there exists $S \subset V$ such that $\delta_A(\bar{S})$ is empty, that is, S is outgoing, a contradiction. ($3 \Rightarrow 1$) Let I be an arbitrary orientation of E. Construct an arcs postman tour T of $N = (V, I, \emptyset, A)$ as follows: Start with an empty arcs postman tour T. For each $i \in I$ from u to v, add to T a cycle consisting of i and a directed path in D from v to u. For each unused $a \in A$ from u to v, add to T a cycle consisting of a and a directed path in D from v to u. ($2 \Rightarrow 4$) Since $\mathcal{S}$ consists of copies of $\emptyset$ and V, we have that $m^+_{\mathcal{S}}(e) = 0$ for all $e \in E$, and $\mathrm{sur}_M(S_i) = 0$ for all $1 \leq i \leq k$. Hence (6.39) holds. ($4 \Rightarrow 1$) Let $N = (V, I, E \setminus I, A)$ be any partial orientation of M. We claim that for all $k \in \mathbb{N}$, and for all families $\mathcal{S} = \{S_1, \ldots, S_k\}$ of outgoing sets, N satisfies:

$$\sum_{e \in E} m^+_{\mathcal{S}}(e) \leq \sum_{i=1}^{k} \lfloor \tfrac{1}{2}\mathrm{sur}_N(S_i) \rfloor. \tag{6.40}$$

We proceed by induction on $n = |I|$. Note that this is true for $n = 0$, and assume it is true for some $0 \leq n < |E|$. Choose any $e' \in E \setminus I$, and let $N' = (V, I', E', A)$ be the partial orientation obtained from N by orienting e' arbitrarily, say, from u to v. Let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a family of k outgoing sets. Let $X, Y \subseteq [k]$ such that $u \in S_X$ and $v \in S_Y$. Using that $|X \setminus Y| \leq m^+_{\mathcal{S}}(e')$ we obtain

$$\begin{aligned}
\sum_{e \in E'} m^+_{\mathcal{S}}(e) &= \sum_{e \in E} m^+_{\mathcal{S}}(e) - m^+_{\mathcal{S}}(e') && (6.41)\\
&\leq \sum_{e \in E} m^+_{\mathcal{S}}(e) - |X \setminus Y| && (6.42)\\
&\leq \sum_{i=1}^{k} \lfloor \tfrac{1}{2}\mathrm{sur}_N(S_i) \rfloor - |X \setminus Y| && (6.43)\\
&= \sum_{i=1}^{k} \lfloor \tfrac{1}{2}\mathrm{sur}_{N'}(S_i) \rfloor, && (6.44)
\end{aligned}$$

which is what we wanted. Hence our claim is true. Now let $N = (V, I, \emptyset, A)$ be obtained from M after orienting arbitrarily all its edges. By our claim, N satisfies the outgoing-set condition and, as in the proof of Corollary 6.7, it has an arcs postman tour. ■

6.4 Integer Programming Formulations

Let $M = (V, E, A)$ be a strongly connected mixed graph, and let $c \in \mathbb{Q}_+^A$. We give two integer programming formulations for Minimum Arcs Postman Tour based on those we gave for Minimum Mixed Postman Tour in Section 3.3, as well as their linear programming relaxations. Later, we consider some valid inequalities for the solutions of the given integer programs. For the first integer programming formulation, let $\vec{M} = (V, A \cup E^+ \cup E^-)$ be the associated directed graph of M, and let $B = E^+ \cup E^-$. As before, we obtain the following integer program for Minimum Arcs Postman Tour:

$$\begin{aligned} \mathrm{MAPT1}(M, c) &= \min c^\top x_A && (6.45)\\ \text{subject to} & && \\ x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) &= 0 \text{ for all } v \in V && (6.46)\\ x_a &\geq 1 \text{ for all } a \in A && (6.47)\\ x_{e^+} + x_{e^-} &= 1 \text{ for all } e \in E && (6.48)\\ x_e &\geq 0 \text{ and integer for all } e \in A \cup E^+ \cup E^-, && (6.49) \end{aligned}$$

and its linear programming relaxation LMAPT1(M, c).

The second integer programming formulation is:

$$\begin{aligned} \mathrm{MAPT2}(M, c) &= \min c^\top x_A && (6.50)\\ \text{subject to} & && \\ x(\delta_A(v)) + x(\delta_A(\bar{v})) &\equiv d_E(v) \pmod 2 \text{ for all } v \in V && (6.51)\\ x(\delta_A(S)) - x(\delta_A(\bar{S})) &\geq d_E(S) \text{ for all } S \subseteq V && (6.52)\\ x_a &\geq 1 \text{ and integer for all } a \in A, && (6.53) \end{aligned}$$

and its linear programming relaxation is:

$$\begin{aligned} \mathrm{LMAPT2}(M, c) &= \min c^\top x_A && (6.54)\\ \text{subject to} & && \\ x(\delta_A(S)) - x(\delta_A(\bar{S})) &\geq d_E(S) \text{ for all } S \subseteq V && (6.55)\\ x_a &\geq 1 \text{ for all } a \in A. && (6.56) \end{aligned}$$

6.4.1 Valid Inequalities

As usual, if $S \subseteq V$ satisfies that $d_M(S)$ is odd, then at least one arc in $\delta_M(S)$ must be duplicated in any arcs postman tour of M. Hence, the odd-cut constraints

$$x(\delta_A(S)) + x(\delta_A(\bar{S})) \geq d_A(S) + d_A(\bar{S}) + 1 \tag{6.57}$$

are valid for the sets of solutions of the two integer programs above.

We can obtain another set of valid inequalities from the proof of Theorem 6.14. Recall that if M has an arcs postman tour then it does not have a bad pair, that is, there are no odd outgoing sets $S, T \subset V$, such that $S \cup T$ and $S \cap T$ are tight even sets, violating (6.16). Hence, for all odd outgoing sets $S, T \subset V$ with even $S \cup T$ and $S \cap T$, we have:

$$\mathrm{sur}_N(S \cup T) + \mathrm{sur}_N(S \cap T) \geq 2. \tag{6.58}$$

In terms of the variables of MAPT1(M, c), this constraint is equivalent to:

$$x(\delta_A(S \cup T)) + 2x(\delta_B(S \cup T)) + x(\delta_A(S \cap T)) + 2x(\delta_B(S \cap T)) \leq d_E(S \cup T) + d_E(S \cap T) - 2. \tag{6.59}$$

6.5 Solvable Cases

We enumerate in this section some special cases of MINIMUM ARCS POSTMAN TOUR and ARCS POSTMAN TOUR that can be solved in polynomial time.

6.5.1 Windy Postman Perfect Graphs

Let M be a connected mixed graph, and let $\bar{M}$ be its underlying undirected graph. If $\bar{M}$ is windy postman perfect, then the set of solutions of LMAPT1(M, c) that satisfy (6.57) is an integral polyhedron. Hence:

Theorem 6.28 *There exists a polynomial-time algorithm to solve* MINIMUM ARCS POSTMAN TOUR*, with input restricted to mixed graphs M whose underlying undirected graph $\bar{M}$ is windy postman perfect.*

6.5.2 Series-Parallel Mixed Graphs

Since series-parallel undirected graphs are windy postman perfect, we have the following:

Corollary 6.29 *There exists a polynomial-time algorithm to solve* MINIMUM ARCS POSTMAN TOUR*, with input restricted to series-parallel mixed graphs M.*

Corollary 6.30 *There exists a polynomial-time algorithm to solve* ARCS POSTMAN TOUR*, with input restricted to series-parallel mixed graphs M.*

There is another algorithm to decide whether a connected series-parallel mixed graph $M = (V, E, A)$ has an arcs postman tour or not. Since the class of series-parallel mixed graphs is closed under contractions of edges and arcs, the next lemma says that we can assume M does not contain directed cycles.

Lemma 6.31 *Let $M = (V, E, A)$ be a mixed graph. Assume that the directed cycle $C = (W, B)$ is a subgraph of $D = (V, A)$, and let $M' = (V', E', A')$ be the mixed graph obtained from M by contracting all edges and arcs in $M[W]$ into a vertex w. Then M has an arcs postman tour if and only if M' has an arcs postman tour.*

Proof. If T is an arcs postman tour of M, then the tour T' obtained by contracting all edges and arcs in $M[W]$ is an arcs postman tour of M'. Conversely, if T' is an arcs postman tour of M', we can construct an arcs postman tour T of M as follows: Start with $T = T'$. Each time that T' enters w through e_1 and leaves through e_2, insert in T a directed path between e_1 and e_2 using arcs in B. After doing this, for every $e \in (A[W] \cup E[W]) \setminus B$, add to T a cycle consisting of e and a directed path between its ends using arcs in B. Finally, if there are still some arcs in B that have not been used by T, add a copy of C to T. ■

Let $\vec{M}$ be any orientation of M. Let $m = |A|+|E|$, and define a function $S : A \cup E \to 2^{\mathbb{Z}}$ as $S(e) = \{-1, 1\}$ for all $e \in E$, and $S(a) = [1, m]$ for all $a \in A$. The motivation for this definition is that, if M is acyclic, each of its arcs leaves some outgoing set, and hence every arcs postman tour of M uses each of its arcs at most m times. For $X, Y \subseteq \mathbb{Z}$, let $-X = \{-x : x \in X\}$, $X+Y = \{x+y : x \in X, y \in Y\}$, and $X-Y = \{x-y : x \in X, y \in Y\}$. Since $\vec{M}$ is also series-parallel, if it has more than one arc, then it has two arcs in series or two arcs in parallel. As long as $\vec{M}$ contains more than one arc, apply iteratively the following reductions:

Series: If $a = (u, v)$ and $b = (v, w)$ are two arcs in series, replace them with an arc $c = (u, w)$ with $S(c) = S(a) \cap S(b)$. If $a = (u, v)$ and $b = (w, v)$ are two arcs in series, replace them with an arc $c = (u, w)$ with $S(c) = S(a) \cap (-S(b))$.

Parallel: If $a = (u, v)$ and $b = (u, v)$ are two arcs in parallel, replace them with an arc $c = (u, v)$ with $S(c) = S(a) + S(b)$. If $a = (u, v)$ and $b = (v, u)$ are two arcs in parallel, replace them with an arc $c = (u, v)$ with $S(c) = S(a) - S(b)$.

These reductions ensure that for every arc c, and at every stage of the algorithm, $S(c)$ contains the possible values of flow from the tail to the head of c. At termination, when $\vec{M}$ consists of only one arc a, M has an arcs postman tour if and only if $0 \in S(a)$, or a is a loop and $S(a) \neq \emptyset$. Since at every stage of the algorithm, and for every arc c, we have that $S(c) \subseteq [-m^2, m^2]$, we can perform all set operations in polynomial time.

However, we cannot expect to solve in polynomial time Arcs Postman Tour, even if the input mixed graph is the union of two series-parallel graphs. The first result below follows directly from the proof of Theorem 6.3, and the second follows from a slight modification of the subgraph on the right of Figure 6.2: Replace each edge by a left-to-right path consisting of an edge, an arc, and an edge.

Corollary 6.32 Arcs Postman Tour *is $\mathcal{NP}$-complete even if the input $M = (V, E, A)$ is restricted to be planar, and both $G = (V, E)$ and $D = (V, A)$ are series-parallel.*

Corollary 6.33 Arcs Postman Tour *is $\mathcal{NP}$-complete even if the input $M = (V, E, A)$ is restricted to be planar, and to satisfy that $G = (V, E)$ is a forest, $D = (V, A)$ is series-parallel, $d_A(v) + d_A(\bar{v}) \leq 2$ for all $v \in V$, and the longest directed path has length 2.*

6.5.3 Directed Forests

If the mixed graph $M = (V, E, A)$ satisfies that $D = (V, A)$ consists of one directed walk and some isolated vertices, then we can solve Arcs Postman Tour in polynomial time. Furthermore, we have the following characterization:

Theorem 6.34 *Let $M = (V, E, A)$ be a connected mixed graph where $D = (V, A)$ consists of one directed walk and isolated vertices. Then M has an arcs postman tour if and only if M satisfies the outgoing-set condition (6.2) and the even-cut condition (6.3).*

Proof. Necessity follows from Theorem 6.5. We give an algorithmic proof of sufficiency. Let $P = (u_0, a_1, u_1, \ldots, a_k, u_k)$ be the directed walk in D. Since M satisfies the outgoing-set condition, there exists an undirected path Q in $G = (V, E)$ from u_k to u_0. Let $H = (V, F)$ be the undirected graph obtained from G by deleting the edges of Q. We claim that all odd degree vertices in H are vertices of P: If $v \in V \setminus V(P)$ has odd degree in H then it also has odd degree in G and, since v and $\bar{v}$ are outgoing, it violates the even-cut condition, a contradiction. If u_i and u_j (with $i < j$) are two vertices of odd degree in the same component of H, there exists a path Q' from u_j to u_i which forms a cycle with $P' = (u_i, a_i, \ldots, u_j)$. Delete Q' from H, and continue until all connected components of H are even. Since each of these components can be oriented to form an Eulerian tour, we have given the cycle decomposition of an arcs postman tour of M, consisting of these Eulerian tours, together with the cycle formed by the paths P and Q, and all the cycles formed by the paths P' and Q'. ■

Corollary 6.35 *Let $M = (V, E, A)$ be a connected mixed graph where $D = (V, A)$ consists of one directed walk and isolated vertices. Then M has an arcs postman tour if and only if M satisfies the* 2*-outgoing-sets condition (6.12).*

Since all steps in the above algorithm can be carried out in polynomial time, we have:

Corollary 6.36 *There exists a polynomial-time algorithm to solve* ARCS POSTMAN TOUR, *with input restricted to mixed graphs $M = (V, E, A)$ such that $D = (V, A)$ consists of one directed walk and isolated vertices.*

Observe that the proof of Theorem 6.1 forbids us from extending Corollary 6.36 to the case when $D = (V, A)$ consists of an arbitrary number of vertex disjoint directed walks. However, if we disallow cycles in the underlying undirected graph of D, that is, if we restrict D to be a forest, we can solve in polynomial time not only ARCS POSTMAN TOUR, but also MINIMUM ARCS POSTMAN TOUR. We use the next lemma in our algorithm, which guarantees that certain arcs must be duplicated in any arcs postman tour.

Lemma 6.37 *Assume that $S \subseteq V$ is an odd outgoing set of M such that $\delta_A(S)$ contains exactly one arc a. Let M' be the mixed graph obtained from M by adding an arc a' parallel to a. Then M has an arcs postman tour if and only if M' does.*

Proof. Since S is odd, any arcs postman tour of M must use at least twice some element of $\delta_M(S)$. Since a is the only arc in $\delta_M(S)$, any arcs postman tour of M must use a at least twice. ■

Theorem 6.38 *There exists a polynomial-time algorithm for* MINIMUM ARCS POSTMAN TOUR, *with input $M = (V, E, A)$ restricted so that $D = (V, A)$ is a forest.*

Proof. Let $T \subseteq V$ be the set of vertices v with $d_M(v)$ odd. If there exists $v \in T$ such that v is not incident to any arc, then M fails the even-cut condition at v, so we can assume that each vertex in T is incident to some arc. Let $F = (V_F, A_F)$ be a maximal tree in D. If $T_F = V_F \cap T$ is odd, then M fails the even-cut condition at V_F, so we can assume that it is even. For each $a \in A_F$ there exists a unique outgoing subset $V_a \subseteq V_F$ such that a is the unique element of $\delta_A(V_a)$. By Lemma 6.37, if V_a is odd then a must be used at least twice by any arcs postman tour of M. Let A_2 be the set of all such arcs, and let $A_1 = A \setminus A_2$. Note that A_2 is the unique minimal T-join of $(\bar{D}, T)$, and hence can be found in polynomial time. Let $l_e = u_e = 1$ for all $e \in E$, $l_a = 1$ and $u_a = +\infty$ for all $a \in A_1$, and $l_a = 2$ and $u_a = +\infty$ for all $a \in A_2$. By Theorem 3.13, since $l(\delta_M(v))$ is even for all $v \in V$, the polyhedron $\mathcal{Q}^1_{BMPT}(M, l, u)$ is integral, and hence the linear program

$$
\begin{aligned}
\text{LMAPTF}(M, c) \quad &= \quad \min c^\top x_A && (6.60)\\
\text{subject to} \quad & && \\
x(\vec{\delta}(\bar{v})) - x(\vec{\delta}(v)) \quad &= \quad 0 \text{ for all } v \in V && (6.61)\\
x_a \quad &\geq \quad 1 \text{ for all } a \in A_1 && (6.62)\\
x_a \quad &\geq \quad 2 \text{ for all } a \in A_2 && (6.63)\\
x_{e^+} + x_{e^-} \quad &= \quad 1 \text{ for all } e \in E && (6.64)\\
x_e \quad &\geq \quad 0 \text{ for all } e \in A \cup E^+ \cup E^-, && (6.65)
\end{aligned}
$$

solves the given instance of MINIMUM ARCS POSTMAN TOUR. ■

Bibliography

[1] D. Ahr. A survey of arc routing problems. `www.informatik.uni-heidelberg.de/groups/comopt/mitarbeiter/ahr/research/overviewArcRouting/`.

[2] A. A. Assad and B. L. Golden. Arc routing methods and applications. In *Network Routing*, volume 8 of *Handbooks Oper. Res. Management Sci.*, pages 375–483. North-Holland, Amsterdam, 1995.

[3] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi. *Complexity and Approximation. Combinatorial Optimization Problems and their Approximability Properties.* Springer-Verlag, Berlin, 1999.

[4] F. Barahona. On the computational complexity of Ising spin glass models. *J. Phys. A*, 15(10):3241–3253, 1982.

[5] F. Barahona. The max-cut problem on graphs not contractible to K_5. *Oper. Res. Lett.*, 2(3):107–111, 1983.

[6] F. Barahona. On some applications of the Chinese postman problem. In *Paths, Flows, and VLSI-Layout (Bonn, 1988)*, volume 9 of *Algorithms Combin.*, pages 1–16. Springer, Berlin, 1990.

[7] F. Barahona. On via minimization. *IEEE Trans. Circuits and Systems*, 37(4):527–530, 1990.

[8] F. Barahona. On cuts and matchings in planar graphs. *Math. Programming*, 60(1, Ser. A):53–68, 1993.

[9] N. L. Biggs, E. K. Lloyd, and R. J. Wilson. *Graph Theory. 1736–1936*. The Clarendon Press Oxford University Press, New York, second edition, 1986.

[10] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. American Elsevier Publishing Co., Inc., New York, 1976.

[11] P. Brucker. The Chinese postman problem for mixed graphs. In *Graph-Theoretic Concepts in Computer Science (Proc. Sixth Internat. Workshop, Bad Honnef, 1980)*, volume 100 of *Lecture Notes in Comput. Sci.*, pages 354–366. Springer, Berlin, 1981.

[12] T. Christof and A. Löbel. `PORTA` – POlyhedron Representation Transformation Algorithm. `www.zib.de/Optimization/Software/Porta/`, December 2002. © 1997–2002 by Konrad-Zuse-Zentrum für Informationstechnik Berlin.

[13] N. Christofides, E. Benavent, V. Campos, Á. Corberán, and E. Mota. An optimal method for the mixed postman problem. In *System Modelling and Optimization (Copenhagen, 1983)*, volume 59 of *Lecture Notes in Control and Inform. Sci.*, pages 641–649. Springer, Berlin, 1984.

[14] V. Chvátal. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Math.*, 4:305–337, 1973.

[15] S. A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.

[16] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver. *Combinatorial Optimization*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons Inc., New York, 1998.

[17] Á. Corberán, R. Martí, and J. M. Sanchis. A GRASP heuristic for the mixed Chinese postman problem. *European J. Oper. Res.*, 142(1):70–80, 2002.

[18] G. B. Dantzig. Maximization of a linear function of variables subject to linear inequalities. In *Activity Analysis of Production and Allocation*, Cowles Commission Monograph No. 13, pages 339–347. John Wiley & Sons Inc., New York, N. Y., 1951.

[19] M. Davis and H. Putnam. A computing procedure for quantification theory. *J. ACM*, 7:201–215, 1960.

[20] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numer. Math.*, 1:269–271, 1959.

[21] M. Dror, editor. *Arc Routing. Theory, Solutions and Applications.* Kluwer Academic Publishers, Boston, MA, 2000.

[22] L. Du and X. Sun. The complexity of two problems on windy graphs. *J. Qingdao Univ. Nat. Sci. Ed.*, 10(1):12–19, 1997.

[23] R. J. Duffin. Topology of series-parallel networks. *J. Math. Anal. Appl.*, 10:303–318, 1965.

[24] M. E. Dyer and A. M. Frieze. Planar 3DM is NP-complete. *J. Algorithms*, 7(2):174–184, 1986.

[25] J. Edmonds. The Chinese's postman problem. *Oper. Res. Bull.*, 13:B–73, 1965. S. 1.

[26] J. Edmonds. Maximum matching and a polyhedron with $0, 1$-vertices. *J. Res. Nat. Bur. Standards Sect. B*, 69B:125–130, 1965.

[27] J. Edmonds. Paths, trees, and flowers. *Canad. J. Math.*, 17:449–467, 1965.

[28] J. Edmonds and E. L. Johnson. Matching: A well-solved class of integer linear programs. In *Combinatorial Structures and their Applications (Proc. Calgary Internat. Conf., 1969)*, pages 89–92. Gordon and Breach, New York, 1970.

[29] J. Edmonds and E. L. Johnson. Matching, Euler tours and the Chinese postman. *Math. Programming*, 5:88–124, 1973.

[30] J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *J. ACM*, 19(2):248–264, 1972.

[31] H. A. Eiselt, M. Gendreau, and G. Laporte. Arc routing problems. I. The Chinese postman problem. *Oper. Res.*, 43(2):231–242, 1995.

[32] H. A. Eiselt, M. Gendreau, and G. Laporte. Arc routing problems. II. The rural postman problem. *Oper. Res.*, 43(3):399–414, 1995.

[33] L. Euler. Solutio problematis ad geometrian situs pertinentis. *Commentarii academiae scientarum Petropolitanae*, 8:128–140, 1736.

[34] C. G. Fernandes, O. Lee, and Y. Wakabayashi. The minimum cycle cover and the Chinese postman problems on mixed graphs with bounded tree width. Submitted, available at `www.ime.usp.br/~yw/`.

[35] H. Fleischner. *Eulerian Graphs and Related Topics. Part 1. Vol. 1*, volume 45 of *Annals of Discrete Mathematics*. North-Holland Publishing Co., Amsterdam, 1990.

[36] H. Fleischner. *Eulerian Graphs and Related Topics. Part 1. Vol. 2*, volume 50 of *Annals of Discrete Mathematics*. North-Holland Publishing Co., Amsterdam, 1991.

[37] R. W. Floyd. Algorithm 97: Shortest path. *Comm. ACM*, 5(6):345, 1962.

[38] L. R. Ford, Jr. and D. R. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton, N.J., 1962.

[39] A. Frank. A survey on T-joins, T-cuts, and conservative weightings. In *Combinatorics, Paul Erdős is Eighty, Vol. 2 (Keszthely, 1993)*, volume 2 of *Bolyai Soc. Math. Stud.*, pages 213–252. János Bolyai Math. Soc., Budapest, 1996.

[40] G. N. Frederickson. Approximation algorithms for some postman problems. *J. ACM*, 26(3):538–554, 1979.

[41] D. Gale. A theorem on flows in networks. *Pacific J. Math.*, 7:1073–1082, 1957.

[42] M. R. Garey and D. S. Johnson. *Computers and Intractability. A Guide to the Theory of NP-Completeness*. W. H. Freeman and Co., San Francisco, Calif., 1979. A Series of Books in the Mathematical Sciences.

[43] A. M. H. Gerards. *Graphs and Polyhedra, Binary Spaces and Cutting Planes*. PhD thesis, Katholieke Universiteit Brabant, Netherlands, March 1988.

[44] R. E. Gomory. Solving linear programming problems in integers. In *Proc. Sympos. Appl. Math., Vol. 10*, pages 211–215. AMS, Providence, R.I., 1960.

[45] P. Greistorfer. Solving mixed and capacitated problems of the Chinese postman. *Central European J. Oper. Res. Econom.*, 3(4):285–309, 1995.

[46] M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–197, 1981.

[47] M. Grötschel and Z. Win. A cutting plane algorithm for the windy postman problem. *Math. Programming*, 55(3, Ser. A):339–358, 1992.

[48] M. G. Guan. Graphic programming using odd or even points. *Chinese Math.*, 1:273–277, 1960.

[49] M. G. Guan. On the windy postman problem. *Discrete Appl. Math.*, 9(1):41–46, 1984.

[50] M. G. Guan. A survey on the Chinese postman problem. *J. Math. Res. Exposition*, 4(1):113–119, 1984.

[51] F. Hadlock. Finding a maximum cut of a planar graph in polynomial time. *SIAM J. Comput.*, 4(3):221–225, 1975.

[52] C. Hierholzer. Über die Möglichkeit, einen Linienzug ohne Wiederholung und ohne Unterbrechung zu umfahren. *Mathematische Annalen*, VI:30–32, 1873.

[53] D. S. Hochbaum, editor. *Approximation Algorithms for NP-Hard Problems*. Computer Science. PWS Publishing Company, Boston, 1997.

[54] A. J. Hoffman. Some recent applications of the theory of linear inequalities to extremal combinatorial analysis. In *Proc. Sympos. Appl. Math., Vol. 10*, pages 113–127. American Mathematical Society, Providence, R.I., 1960.

[55] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley Publishing Co., Reading, Mass., 1979. Addison-Wesley Series in Computer Science.

[56] T. Jeworrek and G. Schulz. Zur Lösung des Briefträgerproblems auf gemischten Graphen. *Wiss. Z. Tech. Univ. Magdeburg*, 31(5):58–62, 1987.

[57] C. H. Kappauf and G. J. Koehler. The mixed postman problem. *Discrete Appl. Math.*, 1(1-2):89–103, 1979.

[58] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[59] R. M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations (Proc. Sympos., IBM Thomas J. Watson Res. Center, Yorktown Heights, N.Y., 1972)*, pages 85–103. Plenum, New York, 1972.

[60] R. M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Math.*, 23(3):309–311, 1978.

[61] L. G. Khachiyan. A polynomial algorithm in linear programming. *Dokl. Akad. Nauk SSSR*, 244(5):1093–1096, 1979.

[62] D. Kőnig. *Theorie der endlichen und unendlichen Graphen. Kombinatorische Topologie der Streckenkomplexe.* Akademische Verlagsgesellschaft, Leipzig, 1936. Reprinted: Chelsea Publishing Co., New York, 1950.

[63] J. Kratochvíl and Z. Tuza. On the complexity of bicoloring clique hypergraphs of graphs. *J. Algorithms*, 45(1):40–54, 2002.

[64] K. Kuratowski. Sur le problème des courbes gauches en topologie. *Fund. Math.*, 15:271–283, 1930.

[65] D. Lichtenstein. Planar formulae and their uses. *SIAM J. Comput.*, 11(2):329–343, 1982.

[66] K. Menger. Zur allgemeinen Kurventheorie. *Fund. Math.*, 10:95–115, 1927.

[67] E. Minieka. The Chinese postman problem for mixed networks. *Management Sci.*, 25(7):643–648, 1979/80.

[68] Y. Nobert and J.-C. Picard. An optimal algorithm for the mixed Chinese postman problem. *Networks*, 27(2):95–108, 1996.

[69] M. W. Padberg and M. R. Rao. Odd minimum cut-sets and b-matchings. *Math. Oper. Res.*, 7(1):67–80, 1982.

[70] C. H. Papadimitriou. On the complexity of edge traversing. *J. ACM*, 23(3):544–554, 1976.

[71] C. H. Papadimitriou. *Computational Complexity.* Addison-Wesley Publishing Company, Reading, MA, 1994.

[72] C. H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *J. Comput. System Sci.*, 43(3):425–440, 1991.

[73] W. L. Pearn and J. B. Chou. Improved solutions for the Chinese postman problem on mixed networks. *Comput. Oper. Res.*, 26(8):819–827, 1999.

[74] W. L. Pearn and M. L. Li. Algorithms for the windy postman problem. *Comput. Oper. Res.*, 21(6):641–651, 1994.

[75] Y. Peng. Approximation algorithms for some postman problems over mixed graphs. *Chinese J. Oper. Res.*, 8(1):76–80, 1989.

[76] B. Raghavachari and J. Veerasamy. A 3/2-approximation algorithm for the mixed postman problem. *SIAM J. Discrete Math.*, 12(4):425–433, 1999.

[77] B. Raghavachari and J. Veerasamy. Approximation algorithms for the asymmetric postman problem. In *Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms (Baltimore, MD, 1999)*, pages 734–741, New York, 1999. ACM.

[78] T. K. Ralphs. On the mixed Chinese postman problem. *Oper. Res. Lett.*, 14(3):123–127, 1993.

[79] T. J. Schaefer. The complexity of satisfiability problems. In *Conference Record of the Tenth Annual ACM Symposium on Theory of Computing (San Diego, Calif., 1978)*, pages 216–226. ACM, New York, 1978.

[80] A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*, volume 24 of *Algorithms and Combinatorics*. Springer-Verlag, Heidelberg, Germany, 2003.

[81] P. D. Seymour. Decomposition of regular matroids. *J. Combin. Theory Ser. B*, 28(3):305–359, 1980.

[82] H. Tohyama and A. Adachi. Complexity of a restricted Chinese postman problem. *Trans. Inform. Process. Soc. Japan*, 37(11):1886–1896, 1996.

[83] V. V. Vazirani. *Approximation Algorithms*. Springer-Verlag, Berlin, 2001.

[84] O. Veblen. An application of modular equations in analysis situs. *Ann. of Math.*, 2(14):86–94, 1912/1913.

[85] J. Veerasamy. *Approximation Algorithms for Postman Problems*. PhD thesis, The University of Texas at Dallas, May 1999.

[86] K. Wagner. Über eine Eigenschaft der ebenen Komplexe. *Math. Ann.*, 114:570–590, 1937.

[87] H.-F. Wang. Insights into a mixed Chinese postman problem. *J. Chinese Inst. Engrs.*, 18(5):683–689, 1995.

[88] S. Warshall. A theorem on boolean matrices. *J. ACM*, 9:11–12, 1962.

[89] Z. Win. *Contributions to Routing Problems*. PhD thesis, Universität Augsburg, Germany, December 1987.

[90] Z. Win. On the windy postman problem on Eulerian graphs. *Math. Programming*, 44(1, (Ser. A)):97–112, 1989.

[91] H. Yan and G. L. Thompson. Finding postal carrier walk paths in mixed graphs. *Comput. Optim. Appl.*, 9(3):229–247, 1998.

[92] K. Yaoyuenyong, P. Charnsethikul, and C. Vira. A heuristic algorithm for the mixed Chinese postman problem. *Optim. Eng.*, 3(2):157–187, 2002. Special issue on optimization with high technology applications (Hong Kong, 2000).

Index