GRAPHS OF EDGE-INTERSECTING NON-SPLITTING PATHS

by

Arman Boyacı

B.S., Industrial Engineering, Galatasaray University, 2007

M.S., Industrial Engineering, Bogazici University, 2009

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

Graduate Program in FBE Program for which the Thesis is Submitted
Boğaziçi University
2015

ACKNOWLEDGEMENTS

First and foremost I would like to thank my advisor Tınaz Ekim Aşıcı for her guidance during these past six years. Without her guidance and constant feedback this work would not have been achievable. She have been a great mentor by allowing and even encouraging me to choose my own direction. I acknowledge all her efforts for introducing me to people in her academic network and for encouraging me to present our work at various conferences. I am also thankful for the example she has provided as a successful scientist and professor.

I would thank Mordechai Shalom for all his support during this journey. I had the pleasure of working closely with him and I have enjoyed the opportunity to watch and learn from his knowledge and experience. I appreciate his positive attitude, his creativity to solve problems and dedication.

I would thank Shmuel Zaks for inviting me to Technion and I want in particular thank him for proposing to study this topic. I would thank Bernard Ries. The chapter five would not exist without fruitful and long discussions with him during my visit to Paris Dauphine University, LAMSADE. I am looking forward to working with him more in the future. I am fortunate to have had Taylan Cemgil and Caner Taşkın serve on my committee. I appreciate all the valuable comments and advises they provided.

I gratefully acknowledge the funding sources that made my PhD work possible. I was funded by the Tübitak BİDEB 2210 fellowship for five years. My work was also supported by the following grants Tübitak 111M303, 111M482, 213M620 and BAP 6461D.

I also thank my friends for providing support and friendship that I needed. I would especially like to thank Serhan Daniş for being such a good flatmate for three years during my PhD.

Last but not least, I want to thank my loving and caring family. Thanks to my parents and my brother Mark for all their love, encouragement and support in all my pursuits. Thanks to my loving, encouraging and patient wife Bilge for the wonderful life that we share together.

ABSTRACT

GRAPHS OF EDGE-INTERSECTING NON-SPLITTING PATHS

In this work, we introduce and study a new graph class: namely the graphs of Edge-Intersecting Non-Splitting Paths (ENP). First, we consider a special case where the host graph is a tree: the graphs of Edge-Intersecting Non-Splitting Paths in a Tree (ENPT). We study the characterization of the ENPT representations of chordless cycles (holes) which are one of the important basic graph structures. Under some assumption, we give an algorithm that returns the unique minimal representation if it exists. However, we show that the problem is NP-complete in general that is when this assumption does not necessarily hold. Then, we consider a more general case for which the host graph can be an arbitrary graph. As opposed to the Edge Intersection Graphs of Paths in an arbitrary graph which includes all graphs, we show that this is not true for ENP that is there exist some graphs which are not ENP. We also show that the class ENP coincides with the family of graphs of Edge-Intersecting and Non-Splitting Paths in a Grid (ENPG). Following similar studies for EPG graph class, we study the implications of restricting the number of bends of the individual paths in the grid. We show that restricting the number of bends also restricts the graph class. More concretely, by restricting the number of bends one gets an infinite sequence of classes such that every class is properly included in the next one. In particular, we show that one bend ENPG graphs are properly included in two bend ENPG graphs. In addition, we show that trees and cycles are one bend ENPG graphs, and characterize split graphs and co-bipartite graphs that are one bend ENPG. We prove that the recognition problem of one bend ENPG graphs is NP-complete even in a very restricted subclass of split graphs. Last, we provide a linear time recognition algorithm for one bend ENPG co-bipartite graphs.

ÖZET

KENAR KESİŞEN VE AYRILMAYAN YOLLARIN ÇİZGELERİ

Bu çalışmada yeni bir çizge sınıfı olan kenar-kesişen ve ayrılmayan yollar (ENP) çizge sınıfını sunuyor ve çalışıyoruz. İlk önce, özel ama nispeten doğal bir durum olan evsahibi çizgenin bir ağaç olduğu bir ağaçta kenar-kesişen ayrılmayan yollar (ENPT) durumunu ele aldık. Temel ve önemli yaplar olan kirişsiz halkaların ENPT gösterimini çalıştık. Özel bir varsayım altında tekil enküçük gösterimi dönen bir algoritma verdik. Ancak gösterdik ki bu problem genel haliyle NP-zordur. Daha sonra daha problemi evsahibi çizgenin herhangi bir çizge olabildiği durum ile genelleştirdik. Herhangi bir evsahibi çizgede yer alan kenar-kesişimli yollar tüm çizgeleri içerse de, gösterdik ki bu durum ENP için doğru değildir. EPG çizge sınıfı için yapılan çalışmalara paralel olarak yolların ızgaradaki bükülme sayısını kısıtlamanın etkilerini çalıştık. Somut olarak, bükülme sayısının kısıtlanması ile birbirini tam olarak içeren sonsuz çizge sınıfları dizesi vardır. Tek bükümlü ENPG çizge sınıfının çift bükümlü çizge sınıfının tam olarak içerildiğini gösterdik. Ayrıca gösterdik ki ağaçlar ve halkaların tek bükümlü ENPG'dir. Yarık (split) ve bütün-ikikümeli (co-bipartite) çizgelerin tek bükümlü ENPG gösterimlerini karakterize ettik. Tek bükümlü ENPG tanıma probleminin yarık çizge sınıfıyla sınırlandırıldığı durumda bile zor olduğunu kanıtladık. Son olarak tek bükümlü ENPG bütün-ikikümeli çizgeler için doğrusal zamanlı bir tanıma algoritması verdik.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

$\{x_1, x_2, \ldots, \}$	set of size k
$\|A\|$	cardinality of set A
$A \subset B$	A is a subset of B
$A \subsetneq B$	A is a subset of B and $A \neq B$
$\mathcal{C}_{nn}$	The set of all cobipartite graphs
$G(V, E)$	A graph with a vertex set V and an edge set E
v	A vertex
e	An edge
P	A path
C	A cycle
$\langle H, \mathcal{P} \rangle$	A set of paths P defined on a host graph H
(G,G')	A pair of graph where G and G' are an EPT graph and an ENPT graph respectively
$\mathcal{O}(G, C)$	An outerplanar graph of a pair (G, C)
$\mathcal{W}(G, C)$	The weak dual tree of an outerplanar graph $\mathcal{O}(G, C)$

LIST OF ACRONYMS/ABBREVIATIONS

EP	Graphs of Edge Intersection of Paths
EPT	Graphs of Edge Intersection of Paths in a Tree
EPG	Graphs of Edge Intersection of Paths in a Grid
VPT	Graphs of Vertex Intersection of Paths in a Tree
ENPT	Graphs of Edge Intersecting Non Splitting Paths in a Tree
ENPG	Graphs of Edge Intersecting Non Splitting Paths in a Grid
B_k-EPG	Graphs of Edge Intersection of k-bend Paths in a Grid
B_k-ENPG	Graphs of Edge Intersecting Non Splitting of k-bend Paths in a Grid

1. INTRODUCTION

Good planning in business is crucial since production costs, service quality etc. are all determined by the execution of the plans. Moreover, planning becomes more and more challenging as the supply chain systems grow. At some point, human intuition and computational power fail to find good plans. In fact, these challenging problems (e.g. production planning, fleet management, workforce management and supply chain management) can be modeled as a mathematical problem which enables us to solve these problems using computers thus with more computational power.

Modeling is an abstraction of the real-life and graphs are one of the abstraction tools in mathematics. Graphs are mathematical structures able to model pairwise relations between objects and graph theory is the study of graphs. Graphs can be used to model many types of relations and processes in physical, social and information systems. Many practical problems (production scheduling, job assignment etc) can be represented by graphs and modeled as a graph optimization problems. Some fundamental graph optimization problems are: maximum matching, maximum flow, maximum clique, maximum independent set, minimum coloring etc. Unfortunately efficient algorithms are not known for many of the graph optimization problems. Of course the efficiency of an algorithm has a precise definition in computer science, we can however simply say that an algorithm is efficient if it terminates correctly in a reasonable time, i.e. the running time of the algorithm is a polynomial of the input size. A problem for which most probably no efficient algorithm exists is technically called NP-hard.

Considering special cases of a problem is one of the approaches to cope with NP-hardness. A typical example is restricting a graph optimization problem to a special graph class where a graph class is usually defined by some structural property. Structural graph theory deals with the characterization of various properties of graphs in order to use them in the design of efficient algorithms. In the literature, hundreds of graph classes are introduced and studied. Some classical graph classes are: trees, bipartite graphs, interval graphs and planar graphs. There exist efficient optimization algorithms (for NP-complete problems) for these graph classes. There are many ways of defining a graph class. For example bipartite graphs are graphs whose vertex set can be partitioned into two independent sets. They are equivalently defined as (odd cycle)-free graphs. Trees are connected cycle-free graphs, or equivalently graphs in which any two vertices are connected by a unique path. These characterizations play a fundamental role for designing efficient algorithms. The main tools in structural graph theory are (i) detecting forbidden substructures and (ii) describing decomposition methods of graphs enabling to design divide-and-conquer algorithms.

Intersection graphs are graphs defined by intersection of a collection of objects, i.e. objects are represented by vertices and an edge between two vertices exists if the corresponding objects intersect. Any graph can be seen as an intersection graph by considering the set of edges incident to each vertex as objects. However, the intersection of geometric objects (e.g. intervals, disks, rectangles) defines graph classes that are used to model real-life applications. Similarly some interesting graph classes are defined by the intersection of graph theoretic objects such as trees and paths. For example, the well-known graph class of triangulated graphs coincides with the intersection graphs of subtrees of a tree. Motivated by applications in telecommunication networks, the Edge Intersection Graphs of Paths in a Tree (resp. in a Grid) EPT (resp. EPG) are also well studied graphs in the literature.

We say that two paths are *splitting* (and we call the corresponding edges *red*) if their union is not a path/cycle, otherwise two paths do not split from each other (and we call the corresponding edges *blue*). In some sense, we color the edges of EPT (or EPG) graphs such that the edges now contain an extra information which can be useful in some applications. In this work, we introduce and study a new graph class: Graphs of Edge-Intersecting Non-Splitting Paths (ENP) which are essentially the graphs consisting of only blue edges.

1.1. Motivating applications

We continue by presenting some applications which motivate the study of these graph classes.

EPT graphs have applications in communication networks. Consider a communication network of a tree topology T. The message routes to be delivered in this communication network are paths on T. Two paths *conflict* if they both require the use of the same link. This conflict model is equivalent to an EPT graph. Suppose we try to find a schedule for the messages such that no two messages sharing a link are scheduled in the same time interval. Then a vertex coloring of the EPT graph corresponds to a feasible schedule on this network.

EPT graphs also appear in all-optical telecommunication networks. The so-called Wavelength Division Multiplexing (WDM) technology can multiplex different signals onto a single optical fiber by using different wavelength ranges of the laser beam [1, 2]. WDM is a promising technology enabling us to deal with the massive growth of traffic in telecommunication networks, due to applications such as video-conferencing, cloud computing and distributed computing [3]. A stream of signals traveling from its source to its destination in optical form is called a *lightpath*. A lightpath is realized by signals traveling through a series of fibers, on a certain wavelength. Specifically, Wavelength Assignment problems (WLA) are a family of path coloring problems that aim to assign wavelengths (i.e. colors) to lightpaths, such that no two lightpaths with a common link receive the same wavelength and a certain objective function (depending on the problem) is minimized.

In optical networks, regenerators have to be placed on lightpaths in order to amplify (regenerate) the signal. *Traffic Grooming* is the term used for the combination of several low capacity requests (modeled by paths of a network) into one lightpath (modeled by a path or cycle of the network) using Time Division Multiplexing (TDM) technology [4]. Traffic grooming decreases the required number of regenerators since requests in the same lightpath share the same regenerators. In this context a set of paths can be combined into one lightpath, as long as they satisfy the following two conditions:

- The load condition: On any given fiber, at most g requests can use the same lightpath, where g is an integer called *the grooming factor*.
- The no-split condition: a lightpath (i.e. the union of the requests using the lightpath) constitutes a path or a cycle of the network.

Clearly, the second condition cannot be checked in the EPT model. For this reason, we introduce ENPT graphs that provide the required information.

Readers unfamiliar with optical networks may consider the following analogous problem in transportation. Consider a set of transportation requests modeled by paths, and trucks traveling along paths or cycles. Trucks are able to load and drop items during their journey as long as at any given time their load does not exceed their capacity. The no-split condition reflects the fact that a truck has to follow a path or a cycle.

By the no-split condition, a (feasible) traffic grooming corresponds (in graph theoretical terms) to a vertex coloring of the graph consisting of red edges (EPT $\setminus$ ENPT). Moreover, by the load condition, every color class induces a sub-graph of an EPT graph with its clique number at most g. Therefore, it makes sense to analyze the structure of these graph pairs (EPT and ENPT).

Under this setting one can consider various objective functions such as:

Minimize the number of wavelengths / trucks. When the number of wavelengths (resp. trucks) is scarce, one aims to minimize this number. We note that when the parameter g is sufficiently big (i.e. $g = \infty$ and resp. no capacity constraint for trucks) the problem boils down to the minimum vertex coloring problem of the graph consisting of red edges. Note that in the transportation case, we assume that disjoint itineraries

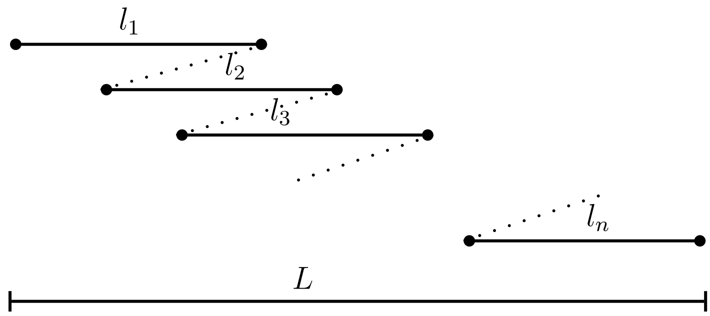

Figure 1.1. The gain obtained by grooming n requests of length $l_1, l_2, l_3, \ldots l_n$ into a lightpath of length L.

can be traveled by the same truck.

Minimize the number of regenerators / total distance traveled. The signal traveling on a lightpath has to be regenerated along its way, implying a regeneration cost roughly proportional to its length [5] (similarly, a truck incurs operational expenses proportional to the distance it travels). The problem of minimizing the number of regenerators is equivalent to maximizing the gain obtained by grooming requests. By definition, the gain obtained by grooming the requests is equal to the length of these requests minus the length of the lightpath, see Figure 1.1. If $g = \infty$ then we can assume that no path is included in another path since such paths do not increase the cost (the length of a lightpath); indeed we can omit such paths during the optimization process and add them afterwards. In such a setting, the requests of a lightpath have an order according to their (both left and right) endpoints.

Observation 1.1. *Let σ be the increasing order of the endpoints of requests of a lightpath. The gain obtained by grooming these requests is equal to the sum of the overlaps of each two consecutive requests in σ.*

In order to model this problem in graph theoretical terms, one has to assign weights to the ENPT edges indicating the length of the overlap of two non-splitting requests. Any feasible solution corresponds to a partition of the vertex set of ENPT (equivalently the vertices of EPT) into sets $V_1, \ldots V_k$, called *blue components*, such that the ENPT graph induced by V_i is connected and its clique number is less than g, and the EPT graph induced by V_i is empty. Note that these blue components correspond to some collection of paths whose union is still a path.

Observation 1.2. *The gain corresponding to a blue component is equal to the heaviest (in terms of weight) path in that component.*

Sketch of Proof. First note that every path of a blue component corresponds to an ordering of the requests (corresponding to the vertices of that blue component). Consider a blue component and its corresponding requests, and number them according to the increasing orders of their endpoints. From Observation 1.1, it follows that the gain is equal to the sum of the overlaps of each two consecutive requests. It is enough to show that this sum is the maximum (heaviest) among all possible orders of requests. Let σ is the ordering corresponding to the heaviest path of the blue component. Suppose σ is not equal to the increasing order. Consider the smallest i such that $\sigma(i+1) = i+k$, $k > 1$. Let σ' be the order obtained by swapping $i+1$ and $i+k$ in σ. One can easily checked that σ' also corresponds to a heavier path in the blue component, a contradiction. □

In [5], it is shown (under the assumption $g = \infty$) that a greedy algorithm, in which the paths are merged in decreasing order of their overlaps, is an optimal algorithm. Consider a set of requests in a tree yielding a pair of EPT and ENPT graphs. We can simulate the greedy algorithm in this $(\mathrm{ENPT}, \mathrm{EPT})$ pair in the following fashion: contract an edge having the maximum weight to a vertex. After this operation parallel edges will possibly appear, replace them with the following rule: if there is a red edge replace parallel edges with that red edge otherwise replace them with the edge having maximum weight. This greedy algorithm is optimal if we assume that the host graph is a tree and $g = \infty$. Our motivation to study the structural properties of related graph classes is to enrich the collection of tools to solve further applications.

1.2. Literature Survey

The related graph class EPT has been extensively studied in the literature. It is shown that recognizing EPT is NP-complete [6]. Similarly, the minimum vertex coloring problem remains NP-complete in EPT graphs [6]. In contrast, one can solve in polynomial time the maximum clique problem [6]. The basic idea is to show that there are two types of cliques and therefore the maximal cliques can be enumerated in polynomial time by a graph search on the host tree. The problem can be solved in polynomial time even if the representation is not available using a clique enumeration algorithm [7] since the number of maximal cliques is polynomial.

In [8] Tarjan proposes a decomposition algorithm, "decomposition by clique separators", which is applicable to EPT graphs; this approach is used to solve the maximum independent set problem in polynomial time. The main idea is to find at each iteration a clique whose removal separates at least two connected components. Tarjan calls an *atom* a subgraph which is not decomposable. This decomposition can be represented by a tree. He describes how to form recursively a solution at some parent node given the optimal solutions of its children. Therefore if we can solve a problem in polynomial time in the atoms then we can solve it in polynomial time in the whole graph. This approach works in EPT graphs mainly because of the following observation. Let T be the host tree and e an edge of T. The removal of e divides T into two trees T_1 and T_2. We can partition the set of paths into three: (i) the paths sharing the edge e which constitutes a clique (ii) the set of paths completely in T_1 (iii) and the set of paths completely in T_2. The later two sets are disjoint and therefore the atoms of EPT graphs have a very specific structure.

After these studies on EPT graphs in the early 80's, this topic neglected until very recently. The studies of Golumbic et al. [9,10] compare various intersection graphs of paths in a tree and their relation to chordal and weakly chordal graphs. Also, some tolerance model is studied via k-edge intersection graphs where two vertices are adjacent if their corresponding paths intersect on at least k edges [11].

Several recent papers consider the edge intersection graphs of paths on a grid (e.g [12]). Since all graphs are EPG (see [13]), studies focus mostly on the sub-classes of EPG where the paths have a limited number of bends. An EPG graph is B_k-EPG if it admits a representation in which every path has at most k bends. The *bend number* of a graph G is the minimum number k such that G has a B_k-EPG representation. Clearly, a graph is B_0-EPG if and only if it is an interval graph. B_1-EPG graphs are studied in [13] in which every tree is shown to be B_1-EPG, and a characterization of C_4 representations is given. In [14] it is shown that there exists an outer-planar graph that is not a B_1-EPG. The recognition problem of B_1-EPG graphs is shown to be NP-complete in [15]. The work [14] investigates the bend number of some special graph classes. In [16] the authors give a characterization of B_1-EPG graphs belonging to some subclasses of chordal graphs. It is shown in [17] that the minimum coloring and maximum independent problems remain NP-complete even in B_1-EPG and provide in the same paper a 4-approximation algorithm for both problems assuming that the representation is given.

1.3. Structure of the thesis

Chapter 2 contains a brief list of graph-theoretical definitions and notations essential to the rest of the thesis.

Chapter 3 explores different ENPT representations of cycles which are simple yet important structures in graph theory. We observe that there can be multiple representations for the same cycle therefore we also consider the underlying EPT graph and call them a pair. We propose a pair recognition problem. We define three properties and we characterize the pairs and the corresponding representations (under some minimality definition) satisfying these properties. Then, we relax the properties one by one and provide each time a recognition algorithm except for the last property. Finally, we present an NP-completeness proof for the general pair recognition problem (if we don't assume the last property holds).

Chapter 4 covers the generalization of the host graph from a tree to an arbitrary graph, namely ENP graphs. We start with two important theorems, showing that not all graphs are ENP and the graph classes ENP and ENPG are equivalent. Therefore, it is sufficient to consider without loss of generality ENPG graphs. As in the case of EPG graphs, we consider the graphs having representation on the grid where the paths have at most k bends, namely B_k-ENPG graphs. We analyze the inclusion of graph classes B_k-ENPG parametrized by k.

Chapter 5 deals with the case $k = 1$. We present some basic results showing that all trees and cycles are B_1-ENPG. We then consider two subclasses B_1-ENPG split and B_1-ENPG cobipartite graphs. We show that the recognizing B_1-ENPG split is NP-complete whereas decide whether a given cobipartite graph is B_1-ENPG is decidable in polynomial time.

Chapter 6 summarizes the main results and proposes some future research directions.

2. PRELIMINARIES

In this chapter we will consider some notions essential for the understanding of the thesis. We start with basic graph terminology, followed by basic definitions for EPT and ENPT graphs. For the omitted definitions and notations see [18].

2.1. General Definitions

A *graph* is an ordered pair (V, E) where $E \subseteq V \times V$. V is the vertex (or node) set and E is the edge set of G. We assume throughout this work that graphs are undirected and simple (no loops and multiple edges). We say that two vertices $u \in V, v \in V$ are adjacent in G if $uv \in E$. Given a graph $G = (V, E)$ and a vertex v of V, we denote by $\delta_G(v)$ the set of edges of G incident to v, by $N_G(v)$ the set consisting of v and its neighbors (the vertices adjacent to v) in G, and by $d_G(v) = |\delta_G(v)|$ the degree of v in G. Whenever there is no ambiguity we omit the subscript G and write $d(v)$, $\delta(v)$ and $N(v)$. We call a vertex v *isolated* if $d(v) = 0$. Given a graph $G = (V, E)$ and a subset $V' \subseteq V$, $H = (V', E')$ is the (induced) *subgraph* of G *induced* by V' where $E' \subseteq E$ is the set of edges whose both endpoints are in V'. Given a graph $G = (V, E)$, $\bar{V} \subseteq V$ and $\bar{E} \subseteq E$ we denote by $G[\bar{V}]$ and $G[\bar{E}]$ the subgraphs of G induced by $\bar{V}$ and by $\bar{E}$, respectively. For a graph $G = (V, E)$, $\bar{G} = (V, \bar{E} = V \times V \setminus E)$ is its *complement* graph. The *union* of two graphs G, G' is the graph $G \cup G' \stackrel{def}{=} (V(G) \cup V(G'), E(G) \cup E(G'))$. The *join* $G + G'$ of two disjoint graphs G, G' is the graph $G \cup G'$ together with all the edges joining $V(G)$ and $V(G')$, i.e. $G + G' \stackrel{def}{=} (V(G) \cup V(G'), E(G) \cup E(G') \cup (V(G) \times V(G')))$. A *path* is a sequence of edges which connect a sequence of distinct vertices. We say that the length of a path is k if it contains k edges. A *cycle* is a sequence of adjacent vertices starting and ending at the same vertex. A cycle C of a graph $G = (V, E)$ is *Hamiltonian* if $V(C) = V$. A graph is *connected* if there is a path between any pairs of vertices Two graphs G and H are *isomorphic* if there is a bijection f from $V(G)$ to $V(H)$ such that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$.

Given a graph, we call a subset of its vertices *independent* or *stable* (resp. *a* *clique*) if no vertex in this subset has a neighbor in this subset. A set is *maximal* with respect to some property if it is not a subset of another set satisfying the property, i.e. the set cannot be expended with vertices without losing this property.

The *maximum independent set* (resp. clique) problem is the problem of finding a maximum cardinality independent set (resp. clique) in a given graph. The *minimum vertex coloring* problem consists in partitioning the vertices of a graph into a minimum number of independent sets. A graph is *k-colorable* if its vertices can be partitioned into k independent sets.

A *family of graphs* (or a *graph class*) is a collection of graphs satisfying some specific property. Given a graph G and a graph class $\mathcal{F}$, we call $\mathcal{F}$ *recognition* the problem of deciding whether G belongs to $\mathcal{F}$ or not.

A *tree* is a connected graph that does not contain any cycles, we usually call the vertices of a tree as nodes. A *subtree* is an induced subgraph of a tree. We can make any tree *rooted* by choosing an arbitrary node as root node which introduces a parent-child relationship between the nodes. A node of a tree is called a *leaf* (resp. *intermediate node*, *junction*) if $d_G(v) = 1$ (resp. $= 2$, ≥ 3). For two nodes u, v of a tree T we denote by $p_T(u, v)$ the unique path between u and v in T. A *star* denoted by $K_{1,k}$ is a tree consisting of one intermediate node and k leaves. A star with 3 edges is called a *claw*.

A graph is *bipartite* (resp. *co-bipartite*, resp. *split*) if its vertex set can be partitioned into two independent sets (resp. two cliques, resp. a clique and an independent set). Note that these partitions are not necessarily unique. We denote bipartite, co-bipartite and split graphs as $X(V_1, V_2, E)$ where

(i) $X = B$ (resp. C, S) whenever G is bipartite (resp. cobipartite, split),

(ii) $V_1 \cap V_2 = \emptyset$,

(iii) for bipartite graphs V_1, V_2 are stable sets,

(iv) for co-bipartite graphs V_1 and V_2 are cliques,

(v) for split graphs V_1 is a clique and V_2 is a stable set, and

(vi) $E \subseteq V_1 \times V_2$ (in other words E does not contain the cliques' edges).

Unless otherwise stated, we assume that G is connected and none of V_1, V_2 is empty.

An $n \times m$ *grid graph* $G_{n,m} = (V, E)$ where $V = [n] \times [m]$ and $(i, j)(i', j') \in E \Leftrightarrow |i - i'| + |j - j'| = 1$. A *bend* of a path P in a grid H is an internal point of P whose edges have different directions, i.e. one vertical and one horizontal.

Given a graph G and a cycle C of it, a *chord* of C in G is an edge of $E(G) \setminus E(C)$ connecting two vertices of $V(C)$. The *length* of a chord connecting vertices i,j is the length of a shortest path between i and j on C. C is a *hole* (chordless cycle) of G if G does not contain any chord of C. This is equivalent to saying that the subgraph $G[V(C)]$ of G induced by the vertices of C is a cycle. For this reason a chordless cycle is also called an *induced* cycle.

A graph G is *chordal* (resp. weakly chordal) if every cycle of G (resp. G and $\bar{G}$) of length at least 4 (resp. at least 5) has a chord.

Let $\mathcal{P}$ be a set of paths in a graph H. The graphs $\textsc{Ep}(\mathcal{P})$ and $\textsc{Enp}(\mathcal{P})$ are such that $V(\textsc{Enp}(\mathcal{P})) = V(\textsc{Ep}(\mathcal{P})) = V$, and there is a one-to-one correspondence between $\mathcal{P}$ and V, i.e. $\mathcal{P} = \{P_v : v \in V\}$. Given two paths $P_u, P_v \in \mathcal{P}$, $\{u, v\}$ is an edge of $\textsc{Ep}(\mathcal{P})$ if and only if P_u and P_v have a common edge, whereas $\{u, v\}$ is an edge of $\textsc{Enp}(\mathcal{P})$ if and only if $P_u \sim P_v$. Clearly, $E(\textsc{Enp}(\mathcal{P})) \subseteq E(\textsc{Ep}(\mathcal{P}))$. A graph G is ENP if there is a graph H and a set of paths $\mathcal{P}$ of H such that $G = \textsc{Enp}(\mathcal{P})$. In this case $\langle H, \mathcal{P} \rangle$ is an ENP *representation* of G. When H is a tree (resp. grid) $\textsc{Ep}(\mathcal{P})$ is an EPT (resp. EPG) graph, and $\textsc{Enp}(\mathcal{P})$ is an ENPT (resp. ENPG) graph; these graphs are denoted also as $\textsc{Ept}(\mathcal{P})$, $\textsc{Epg}(\mathcal{P})$, $\textsc{Enpt}(\mathcal{P})$ and $\textsc{Enpg}(\mathcal{P})$. We say that two representations are *equivalent* if they are representations of the same graph.

Let $\langle H, \mathcal{P} \rangle$ be a representation of an ENP graph G. $\mathcal{P}_e \overset{def}{=} \{P \in \mathcal{P} | \ e \in P\}$ denotes the set of trails of $\mathcal{P}$ containing the edge e of H. For a subset $S \subseteq V(G)$

we define $\mathcal{P}_S \stackrel{def}{=} \{P_v \in \mathcal{P} : v \in S\}$. When H is a tree (resp. grid) $\textsc{Ep}(\mathcal{P})$ is an EPT (resp. EPG) graph, and $\textsc{Enp}(\mathcal{P})$ is an ENPT (resp. ENPG) graph; these graphs are denoted also as $\textsc{Ept}(\mathcal{P})$, $\textsc{Epg}(\mathcal{P})$, $\textsc{Enpt}(\mathcal{P})$ and $\textsc{Enpg}(\mathcal{P})$.

Given two paths $P = (e_1, e_2, \ldots, e_\ell)$ and $P' = (e'_1, e'_2, \ldots, e'_{\ell'})$, a *segment* of $P \cap P'$ is a maximal path that constitutes a sub-path of both P and P'. Clearly, $P \cap P'$ is the union of edge disjoint segments. We denote the set of these segments by $\mathcal{S}(P, P')$.

2.2. The EPT and ENPT graphs

For the following discussion we refer the reader to Figure 2.1. Given two paths P, P' in a graph, we write $P \parallel P'$ to denote that P and P' are non-intersecting, i.e. *edge-disjoint*. The *split vertices* of P and P' is the set of junctions in their union $P \cup P'$ and is denoted by $split(P, P')$. Whenever P and P' intersect and $split(P, P') = \emptyset$ we say that P and P' are *non-splitting* and denote this by $P \sim P'$. In this case $P \cup P'$ is a path or a cycle. When P and P' intersect and $split(P, P') \neq \emptyset$ we say that they are *splitting* and denote this by $P \nsim P'$. Clearly, for any two paths P and P' exactly one of the following holds: $P \parallel P'$, $P \sim P'$, $P \nsim P'$. When the graph G is a tree, the union $P \cup P'$ of two intersecting paths P, P' on G is a tree with at most two junctions, i.e. $|split(P, P')| \leq 2$ and $P \cup P'$ is a path whenever $P \sim P'$. A vertex w of a path P that is not an endpoint of P is called an *internal vertex* of P. We also say that P *crosses* w. For an edge $e = \{p, q\}$ we use $split(e)$ as a shorthand for $split(P_p, P_q)$. Throughout this work, in all figures, the edges of the tree T of a representation $\langle H, \mathcal{P} \rangle$ are drawn as solid lines whereas the paths on the tree are shown by dashed, dotted, etc. edges.

For the following discussion please refer to Figure 2.1. The pairs of paths (P_2, P_4) and (P_3, P_4) do not share a common edge, therefore $P_2 \parallel P_4$ and $P_3 \parallel P_4$. P_1 and P_4 have a common edge $11, 12$, and 12 is a common internal vertex constituting a split of P_1 and P_4, therefore $P_1 \nsim P_4$. Similarly P_1 and P_3 have three common edges and 10 is a split vertex of P_1 and P_3, therefore $P_1 \nsim P_3$. P_1 and P_2 have three common edges but no splits, then $P_1 \sim P_2$. The same holds for the pair (P_2, P_3). However, we note that in

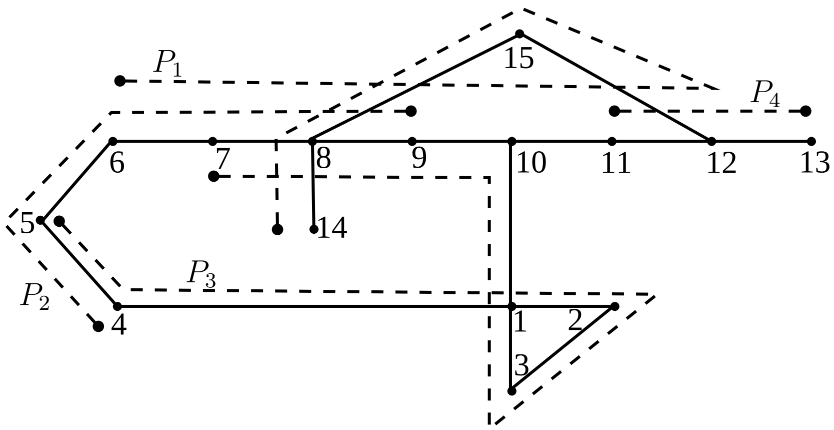

Figure 2.1. A set of paths in an arbitrary graph.

the latter case the common edges are separated into two segments. The vertex 8 is not a split point of P_1 and P_2 because the only internal points of P_1 involving this vertex are $(e_{7,8}, 8, e_{7,9}), (e_{14,8}, 8, e_{8,15})$, and the only internal point of P_2 involving it is $(e_{7,8}, 8, e_{7,9})$. Moreover, $|\{e_{7,8}, e_{7,9}\} \cap \{e_{7,8}, e_{7,9}\}| = 2 \neq 1$ and $|\{e_{7,8}, e_{7,9}\} \cap \{e_{14,8}, e_{7,15}\}| = 0 \neq 1$.

When the graph G is a tree, the union $P \cup P'$ of two intersecting paths P, P' on G is a tree with at most two junctions, i.e. $|split(P, P')| \leq 2$ and $P \cup P'$ is a path whenever $P \sim P'$.

3. GRAPHS OF EDGE-INTERSECTING NON-SPLITTING PATHS IN A TREE

3.1. Overview

In this chapter, we consider Graphs of Edge-Intersecting Non-Splitting Paths in a Tree (ENPT). The results presented in this chapter are organized in several papers [19–21]. The host graph being a tree implies the host graph itself does not contain any cycle. However the corresponding ENPT graph can contain arbitrarily large cycles. We study ENPT representations of trees and cycles. For the later we show that they are more complex compared to EPT representations of cycles, see Figure 3.1. In Section 3.6, we showed that all ENPT representations satisfying the condition ($P3$), defined in Section 3.3, of a cycle have the form as in Figure 3.1c.

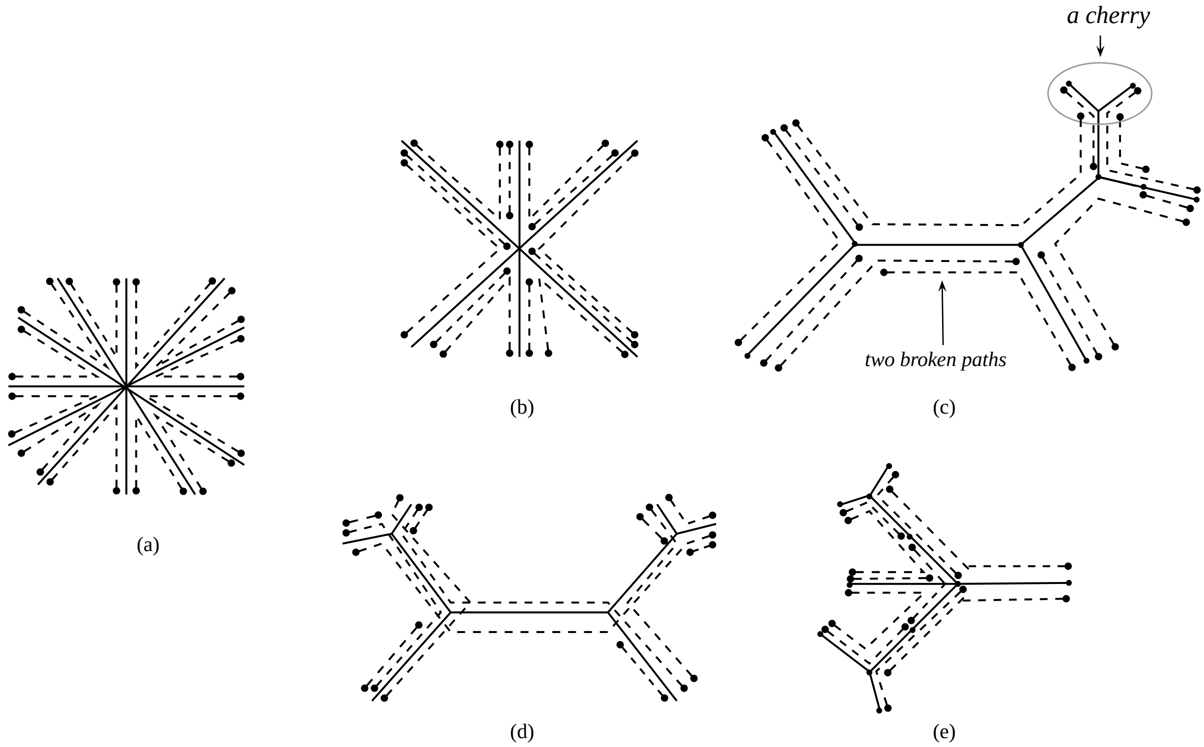

Figure 3.1. (a) The EPT representation of a C_{12}, (b) a simple ENPT representation of a C_{12} (c) a broken planar tour with cherries representation of a C_{12}, (d) a non-planar tour representation of a C_{12}, (e) a non-tour representation of a C_{10}.

Our work in this section mainly follows the lines of Golumbic and Jamison's research [6, 22] in which they defined the EPT graph class, and characterized the representations of chordless cycles (holes). It turns out that ENPT holes have a more complex structure than EPT holes. Even for a small cycle of length 4, we can observe that there are two different ENPT representations, however the corresponding EPT graphs are different, see Figure 3.6. For this reason, in our analysis, we assume that the EPT graph corresponding to a representation of an ENPT hole is given. We also introduce three assumptions $(P1)$, $(P2)$, $(P3)$ defined on EPT, ENPT pairs of graphs.

In Section 3.2 we start with definitions and notations. We obtain in Section 3.3 some basic results regarding ENPT graphs, and their relationship with EPT graphs. We consider pairs of graphs (G, C) where C is a Hamiltonian cycle of G, such that $\text{Ept}(\mathcal{P}) = G$ and $\text{Enpt}(\mathcal{P}) = C$. Given a pair (G, C) we call the problem of determining whether there is a minimal representation $\langle T, \mathcal{P} \rangle$ of (G, C) as HamiltonianPairRec.

We introduce the properties $(P1), (P2)$ and $(P3)$ under which, in Section 3.4, we characterize the representations of pairs in Theorem 3.19. It turns out that the unique minimal representation is a planar tour of the weak dual tree of G where a planar tour of a tree is a collection of two types of paths; short (from a leaf to its parent) and long (from a leaf to another leaf consecutive in the cyclic permutation), see Figure 3.10.

In Section 3.5, we relax assumption $(P1)$. We present basic results regarding the contraction operation and describe an algorithm for pairs satisfying assumptions $(P2)$ and $(P3)$. In Theorem 3.30, we showed that these representations are exactly broken planar tours where a *broken tour* is a representation obtained from a tour by subdividing edges and breaking apart some long paths. When a long path is broken apart into two paths, the non-leaf endpoint for each one of the two paths should be determined. We call this procedure AdjustEndpoint which is a subroutine in Algorithm FindMinimalRepresentation-P2-P3.

In Section 3.6, we aslo relax assumption $(P2)$. We first characterize the representations of (K_4, P_4) and call them cherries. We then provide in Theorem 3.33 a family of graphs which does not belong to ENPT. It should be noted that non-ENPT graphs do not directly follow from their definition. Indeed showing that a given graph do not admit an ENPT representation requires the knowledge of some structural properties on ENPT. We define aggressive contraction operations, see Figures 3.19 and 3.20. Using these results, we present Algorithm FindMinimalRepresentation-P3 that returns the minimal representation of a given pair (G, C) satisfying $(P3)$ in polynomial-time.

In Section 3.7, we show that in general, i.e. in the case where assumption $(P3)$ does not hold, there does not exist a polynomial-time algorithm that provides the minimal representation of a given pair (G, C), unless P = NP. This result extends the NP-completeness result for EPT-recognition problem. More specifically EPT-recognition problem remains NP-complete even the edges of graphs corresponding to the splitting paths are labeled. The main difficulty originates from deciding whether a given clique is represented by an edge clique or a claw clique. The complexity of the recognition problem when this information is provided by an oracle is open. On the other hand, ENPT recognition in general is open. We showed that pair recognition is NP-complete however this result does not imply that ENPT recognition is NP-complete. Given an ENPT graph, the flexibility of choosing EPT edges in various ways might render the recognition problem polynomial-time solvable.

In Section 3.7, we provide a recognition algorithm for a pair satisfying $(P3)$ and we characterize the representation of such pairs. However a stronger result is still missing: the characterization of pairs satisfying $(P3)$, we achieve a partial result in Section 3.5 by characterizing pairs satisfying $(P1), (P2), (P3)$.

We know that ENPT $\setminus$ EPT is not empty since the wheel graph on five vertices (a graph consisting of a cycle and a vertex adjacent to all vertices of this cycle) is an element of this set. However whether EPT $\not\subseteq$ ENPT is still open.

Optimization problems in EPT graphs is another interesting research direction. The clique enumeration algorithm described in [7] is output-sensitive. This means that the time complexity of the algorithm depend on the output, i.e. number of cliques.

As in the case of EPT graphs maximum clique problem is polynomial-time solvable in ENPT graphs with a clique enumeration algorithm since there are polynomial number of maximal cliques. Searching for a more efficient algorithm instead of this generic approach is another research topic. Independent set problem in EPT is polynomial-time solvable using Tarjan's decomposition by clique separators [8]. In this approach we recursively decompose the graph using a clique into two connected components. We call non-decomposable components as atoms. In the same paper Tarjan considers some fundamental problems, e.g. independent set, minimum vertex coloring and describe how to combine the solutions of these problems in the atoms to get a global solution. Therefore if one can solve a problem from this list efficiently in some graph class then this problem is polynomial-time solvable. Following this approach it is shown in the same paper that independent set problem in EPT graphs is polynomial-time solvable and there is a $\frac{3}{2}$-approximation algorithm for the minimum vertex coloring problem in the same graph class. Is there a special structure of the atoms of ENPT graphs? Probably not. Consider a representation $\langle T, \mathcal{P} \rangle$ of an EPT graph. Since T is a tree for any edge e, e divides T into two subtrees T_1 and T_2 and the set of paths into three sets: the set of paths containing e (the corresponding vertices form a clique) and the set of paths in $T_1 \setminus e$ and $T_2 \setminus e$ (the corresponding vertices form two connected components). However this is not true for ENPT graphs. Consider a set of paths containing an edge e, note that the corresponding vertices do not necessarily form a clique. For this reason if we remove a clique we do not have necessarily two connected components. One research direction is to investigate decomposition algorithms for ENPT. On the other hand since ENPT is a partial graph of EPT an independent set of EPT is also an independent set of ENPT. Based on this fact, another interesting question is whether the maximum independent set problem remains polynomial in ENPT graphs.

3.2. Preliminaries

In this Section, we give necessary definitions, present known results related to our work, and develop basic results. In Section 3.2.1 we present known results on EPT graphs that are closely related to our work. In Section 3.2.2 we show that cycles, trees and cliques are ENPT graphs.

3.2.1. EPT Graphs

We now present definitions and results from [6] that we use throughout this work.

Two graphs G and G' such that $V(G) = V(G')$ and $E(G') \subseteq E(G)$ are termed a *pair (of graphs)* denoted as (G, G'). If $\textsc{Ept}(\mathcal{P}) = G$ (resp. $\textsc{Enpt}(\mathcal{P}) = G$) we say that $\langle T, \mathcal{P} \rangle$ is an EPT (resp. ENPT) representation for G. If $\textsc{Ept}(\mathcal{P}) = G$ and $\textsc{Enpt}(\mathcal{P}) = G'$ we say that $\langle T, \mathcal{P} \rangle$ is a representation for the pair (G, G'). Given a pair (G, G') the sub-pair induced by $\bar{V} \subseteq V(G)$ is the pair $(G[\bar{V}], G'[\bar{V}])$. Clearly, any representation of a pair induces representations for its induced sub-pairs, i.e. the pairs have the hereditary property. We note that e is a red edge if and only if $split(e) \neq \emptyset$.

Given a (simple) graph G and $e \in E(G)$, we denote by $G_{/e}$ the (simple) graph obtained by contracting the edge e of G, i.e. by coinciding the two endpoints of $e = \{p, q\}$ to a single vertex $p.q$, and then removing loops and parallel edges. Let $\bar{E} = \{e_1, e_2, \ldots e_k\} \subseteq E(G)$. We denote by $G_{/e_1,\ldots,e_k}$ the graph obtained from $G_{/e_1,\ldots,e_{k-1}}$ by contracting the (image of the) edge e_k. The effect of such a sequence of contractions is equivalent to contracting every connected component of $G[\{e_1, \ldots, e_k\}]$ to a vertex. Therefore the order of contractions is not important, i.e. for any permutation π of $\{1, \ldots, k\}$ we have $G_{/e_1,\ldots,e_{k-1}} = G_{/e_{\pi(1)},\ldots,e_{\pi(k-1)}}$. Based on this fact, we denote by $G_{/\bar{E}}$ the graph obtained by contracting the edges of $\bar{E}$ (in any order).

An *outerplanar* graph is a planar graph that can be embedded in the plane such that all its vertices are on the unbounded face of the embedding. An outerplanar graph is Hamiltonian if and only if it is biconnected; in this case the unbounded face forms the unique Hamiltonian cycle. The *weak dual* graph of a planar graph G is the graph obtained from its dual graph, by removing the vertex corresponding to the unbounded face of G. The weak dual graph of an outerplanar graph is a forest, and in particular the weak dual graph of a Hamiltonian outerplanar graph is a tree [23]. When working with outerplanar graphs we use the term *face* to mean a bounded face.

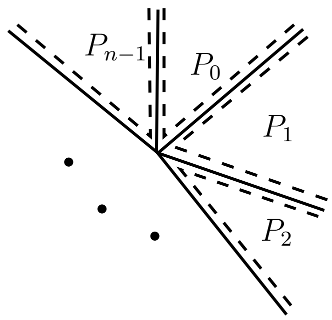

Figure 3.2. The representation of EPT cycle; a pie.

A *cherry of a tree* T is a connected subgraph of T consisting of two leaves of T adjacent to an internal vertex of T. Similarly a *cherry of a representation* $\langle T, \mathcal{P} \rangle$ is a cherry of T with leaves v, v' such that v (resp. v') is an endpoint of exactly one path P (resp. P') of $\mathcal{P}$, and $P \neq P'$.

A *pie* of a representation $\langle T, \mathcal{P} \rangle$ of an EPT graph is an induced star $K_{1,k}$ of T with k leaves $v_0, v_1, \ldots, v_{k-1} \in V(T)$, and k paths $P_0, P_1, \ldots P_{k-1} \in \mathcal{P}$, such that for every $0 \leq i \leq k-1$ both v_i and $v_{(i+1) \bmod k}$ are vertices of P_i. We term the central vertex of the star as the *center* of the pie (See Figure 3.2). It is easy to see that the EPT graph of a pie with k leaves is the hole C_k on k vertices. Moreover, this is the only possible EPT representation of C_k when $k \geq 4$.

Theorem 3.1. *[6] If an* EPT *graph contains a hole with $k \geq 4$ vertices, then every representation of it contains a pie with k paths.*

Let $\mathcal{P}_e \stackrel{def}{=} \{p \in \mathcal{P} | \ e \in p\}$ be the set of paths in $\mathcal{P}$ containing the edge e. A star $K_{1,3}$ is termed a *claw*. For a claw K of a tree T, $\mathcal{P}[K] \stackrel{def}{=} \{p \in \mathcal{P} | \ p$ uses two edges of $K\}$. It is easy to see that both $\textsc{Ept}(\mathcal{P}_e)$ and $\textsc{Ept}(\mathcal{P}[K])$ are cliques. These cliques are termed *edge-clique* and *claw-clique*, respectively. Moreover, these are the only possible representations of cliques.

Theorem 3.2. *[6] Any maximal clique of an* EPT *graph with representation $\langle T, \mathcal{P} \rangle$ corresponds to a subcollection $\mathcal{P}_e$ of paths for some edge e of T, or to a subcollection $\mathcal{P}[K]$ of paths for some claw K of T.*

Note that a claw-clique is a pie with 3 leaves.

3.2.2. Some ENPT graphs

In this section we show that trees, cycles and cliques are contained in the family of ENPT graphs, and give a complete characterization of the ENPT representations of cliques:

Lemma 3.3. *Every clique K of* $\textsc{Enpt}(\mathcal{P})$ *corresponds to an edge-clique, such that the union of the paths representing K is a path.*

Proof. $\textsc{Enpt}(\mathcal{P})$ is a subgraph of $\textsc{Ept}(\mathcal{P})$. Therefore a clique K of $\textsc{Enpt}(\mathcal{P})$ is a clique of $\textsc{Ept}(\mathcal{P})$. Assume, by way of contradiction that K does not correspond to an edge-clique. By Theorem 3.2, K corresponds to either an edge-clique or a claw-clique. A claw-clique that is not an edge-clique, contains two paths P_p, P_q each of which uses a different pair of the three edges of the claw. Therefore $P_p \nsim P_q$, i.e. $\{p,q\} \notin E(K)$, a contradiction. Therefore K corresponds to an edge-clique. To show the second part of the claim, assume that the union of the paths corresponding to the vertices of K is not a path. Then it contains at least one split vertex, i.e. it contains two paths P_p, P_q such that $P_p \nsim P_q$, i.e. $\{p,q\} \notin E(K)$, a contradiction. $\square$

A direct consequence of Lemma 3.3 is that the maximum clique problem in ENPT graphs can be solved in polynomial time. Let G be an ENPT graph and $\langle T, \mathcal{P} \rangle$ be an ENPT representation for G. Consider an edge e of T, the union of paths in $\mathcal{P}_e$ induces a subtree T_e of T. Let $l_1, l_2, \ldots, l_k \in V(T)$ be the leaves of T_e. Let $\mathcal{P}_e^{l_i,l_j} \stackrel{def}{=} \{P \in \mathcal{P}_e | P \subseteq p_T(l_i, l_j)\}$. The maximal cliques of G correspond to the sets $\mathcal{P}_e^{l_i,l_j}$. Therefore, there are at most $O(V(T)^3)$ maximal cliques in G. We conclude that (even if a representation $\langle T, \mathcal{P} \rangle$ is not given) a maximum clique can be found using a clique enumeration algorithm, e.g. [7], since there are only a polynomial number of maximal cliques.

Lemma 3.4. *Every tree is an* ENPT *graph.*

Proof. Given a tree T', the following procedure provides an ENPT representation $\langle T, \mathcal{P} \rangle$ of T': 1) $T \leftarrow T'$, 2) choose an arbitrary vertex r as the root of T and hang T from r, 3) add two vertices $\bar{r}, \bar{\bar{r}}$ and two edges $\{\bar{\bar{r}}, \bar{r}\}$ $\{\bar{r}, r\}$ to T, 4) $\mathcal{P} = \{P_v |\ v \in T'\}$ where P_v is a path of length 2 between v and its ancestor at distance 2. It remains to show that $\{u, v\} \in T'$ if and only if $P_u \sim P_v$. Indeed, let $\{u, v\} \in T'$, and assume without loss of generality that u is the parent of v in T. Then P_u intersects P_v because they both use the edge connecting u to its parent. Moreover they do not split, because their union is the path from v to its ancestor at distance 3. Therefore $P_u \sim P_v$. Conversely, assume that $P_u \sim P_v$. Then P_u and P_v intersect. As every vertex is a starting vertex of at most one path and the paths are of length 2, the second edge of one of the paths, say P_v is the first edge of P_u, therefore u is the parent of v in T, i.e. $\{u, v\} \in T'$. $\square$

Let T be a tree with k leaves and $\pi = (\pi_0, \ldots, \pi_{k-1})$ a cyclic permutation of the leaves. The *tour* (T, π) is the following set of $2k$ paths: (T, π) contains k *long* paths, each of which connecting two consecutive leaves $\pi_i, \pi_{i+1 \mod k}$. (T, π) contains k *short* paths, each of which connecting a leaf π_i and its unique neighbor in T (see Figure 3.3c). Note that ENPT$((T, \pi))$ is a cycle.

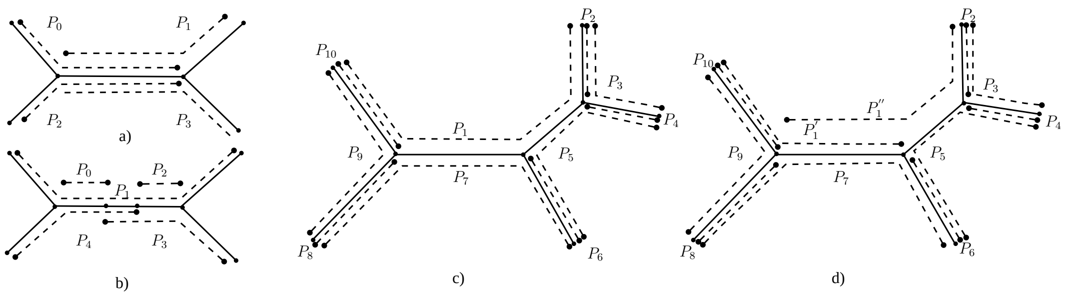

Figure 3.3. (a) A minimal representation of C_4, (b) a minimal representation of C_5 (c) a tour representation of the even hole C_{10}, (d) a representation of the odd hole C_{11}.

A *planar embedding* of a tour is a planar embedding of the underlying tree such that any two paths of the tour do not cross each other. A tour is *planar* if there exists a planar embedding of it. The tour in Figure 3.3c is a planar embedding of a tour. Note that a tour (T, π) is planar if and only if π corresponds to the order in which the leaves are encountered by some DFS traversal of T.

The opposite of a sequence of union operations that create one path is termed *breaking apart*. Namely, breaking apart a path P is to replace it with paths $P_1, \ldots, P_k$ such that $\cup_{i=1}^{k} P_i = P$, $\forall 1 \leq i < k, P_i \cap P_{i+1} \neq \emptyset$, and $P_i \subseteq P_j$ if and only if $i = j$. A *broken tour* is a representation obtained from a tour by subdividing edges and breaking apart long paths of a tour. Clearly, if the tour is planar the broken tour is also planar, i.e. has a planar embedding.

Lemma 3.5. *Every cycle C_k is an* ENPT *graph.*

Proof. $C_3 = K_3$ is an ENPT graph by Lemma 3.3. As for C_4 and C_5, possible ENPT representations are shown in Figures 3.3a and 3.3b, respectively. Any even hole C_{2k}, $(k \geq 3)$ is an ENPT graph. Indeed, for any tree T with k leaves, and a cyclic permutation π of its leaves, the tour (T, π) constitutes an ENPT representation of C_{2k}. Any odd hole C_{2k+1}, $(k \geq 3)$ is an ENPT graph. Let T be a tree with k leaves. Split any long path of some tour (T, π) into two intersecting sub-paths such that no chord is created (if necessary subdivide an edge of the tree into two edges) (see Figure 3.3d). The set of $2k+1$ paths obtained in this way constitutes an ENPT representation for C_{2k+1}. □

3.3. Basic Properties of $\mathrm{EPT}, \mathrm{ENPT}$ Pairs

In this section we develop the basic tools that we use in subsequent sections towards our goal of characterizing representations of $\mathrm{ENPT}, \mathrm{EPT}$ pairs. We define an equivalence relation on representations, namely two representations will be equivalent in this relation if they are representations of the same pair. We also define a partial order on representations. In this work, we focus on finding representations that are minimal with respect to this partial order. We define the contraction operation on pairs, and the union operation on representations. The contraction operation is a restricted variant of graph contraction operation that operates on both graphs of a pair. The union operation is the operation of replacing two paths by their union whenever possible.

Equivalent and minimal representations. We say that the representations $\langle T_1, \mathcal{P}_1\rangle$ and $\langle T_2, \mathcal{P}_2\rangle$ are *equivalent*, and denote it by $\langle T_1, \mathcal{P}_1\rangle \approxeq \langle T_2, \mathcal{P}_2\rangle$, if their corresponding EPT and ENPT graphs are isomorphic under the same isomorphism (in other words, if they constitute representations of the same pair of graphs (G, G')).

We write $\langle T_1, \mathcal{P}_1\rangle \rightsquigarrow \langle T_2, \mathcal{P}_2\rangle$ if $\langle T_2, \mathcal{P}_2\rangle$ can be obtained from $\langle T_1, \mathcal{P}_1\rangle$ by one of the following two operations that we term *minifying operations*:

- Contraction of an edge e of T_1 (and of all the paths in $\mathcal{P}_1$ using e)
- Removal of an initial edge (*tail*) of a path in $\mathcal{P}_1$.

The partial order $\gtrsim$ is the reflexive-transitive closure of the relation $\rightsquigarrow$, and $\langle T_1, \mathcal{P}_1\rangle \lesssim \langle T_2, \mathcal{P}_2\rangle$ is equivalent to $\langle T_2, \mathcal{P}_2\rangle \gtrsim \langle T_1, \mathcal{P}_1\rangle$. $\langle T, \mathcal{P}\rangle$ is a *minimal* representation if it is minimal in the partial order $\lesssim$ restricted to its equivalence class $[\langle T, \mathcal{P}\rangle]_{\approxeq}$ i.e., over all the representations representing the same pair as $\langle T, \mathcal{P}\rangle$. Throughout the work we aim at characterizing minimal representations.

Lemma 3.6. *Let $\langle T_1, \mathcal{P}_1\rangle \gtrsim \langle T_2, \mathcal{P}_2\rangle$, and s be a minimal sequence of minifying operations transforming $\langle T_1, \mathcal{P}_1\rangle$ to $\langle T_2, \mathcal{P}_2\rangle$. Then every permutation of s also transforms $\langle T_1, \mathcal{P}_1\rangle$ to $\langle T_2, \mathcal{P}_2\rangle$.*

Proof. If $contract(e)$ is an operation of s then there is no other operation in s involving e. This is because such an operation is impossible after $contract(e)$, and if it appears before $contract(e)$ it contradicts the minimality of s. To conclude the result, we observe that any two successive operations in s are interchangeable. Indeed, for two distinct edges e, e' the operations $contract(e), contract(e')$ (resp. $contract(e), tr(P, e')$) are interchangeable, and for two not necessarily distinct edges e, e' the operations $tr(P, e), tr(P', e')$ are interchangeable. $\square$

Lemma 3.7. *If $\langle T_1, \mathcal{P}_1\rangle \gtrsim \cdots \gtrsim \langle T_n, \mathcal{P}_n\rangle$ and $\langle T_1, \mathcal{P}_1\rangle \approxeq \langle T_n, \mathcal{P}_n\rangle$, then $\langle T_1, \mathcal{P}_1\rangle \approxeq \cdots \approxeq \langle T_n, \mathcal{P}_n\rangle$.*

Proof. Let $G_i = \textsc{Ept}(\mathcal{P}_i)$ and $G'_i = \textsc{Enpt}(\mathcal{P}_i)$. We observe that both minifying operations are monotonic in the sense that they neither introduce neither new intersections, nor new splits. Namely, for $1 \leq i < n$, $E(G_{i+1}) \subseteq E(G_i)$ and $E(G_{i+1}) \setminus E(G'_{i+1}) \subseteq E(G_i) \setminus E(G'_i)$. As $\langle T_1, \mathcal{P}_1 \rangle \approxeq \langle T_n, \mathcal{P}_n \rangle$ we have $(G_1, G'_1) = (G_n, G'_n)$, i.e. $E(G_n) = E(G_1)$ and $E(G_n) \setminus E(G'_n) = E(G_1) \setminus E(G'_1)$. Therefore $E(G_1) = \cdots = E(G_n)$ and $E(G_1) \setminus E(G'_1) = \cdots = E(G_n) \setminus E(G'_n)$, concluding $(G_1, G'_1) = \cdots = (G_n, G'_n)$. $\square$

EPT *Holes.*

Lemma 3.8. *A hole of size at least* 4 *of an* EPT *graph does not contain blue (i.e.* ENPT*) edges.*

Proof. Consider the pie representing the hole under consideration. For any two paths P_p, P_q of this pie, we have either $P_p \nsim P_q$ or $P_p \parallel P_q$, therefore $\{p, q\}$ is not an ENPT edge. $\square$

Combining with Theorem 3.1, we obtain the following characterization of pairs (C_k, G'):

- $k > 3$. In this case C_k is represented by a pie. Therefore G' is an independent set. In other words C_k consists of red edges. We term such a hole, a red hole.
- $k = 3$ and C_k consists of red edges (G' is an independent set). We term such a hole a red triangle.
- $k = 3$ and C_k contains exactly one ENPT (blue) edge ($G' = P_1 \cup P_2$). We term such a hole a BRR triangle, and its representation is an edge-clique.
- $k = 3$ and C_k contains two ENPT (blue) edges ($G' = P_3$). We term such a hole a BBR triangle, and its representation is an edge-clique.
- $k = 3$ and C_k consists of blue edges ($G' = C_3$). We term such a hole a blue triangle.

EPT *contraction.* Let $\langle T, \mathcal{P}\rangle$ be a representation and $P_p, P_q \in \mathcal{P}$ such that $P_p \sim P_q$. We denote by $\langle T, \mathcal{P}\rangle_{/P_p,P_q}$ the representation that is obtained from $\langle T, \mathcal{P}\rangle$ by replacing the two paths P_p, P_q by the path $P_p \cup P_q$, i.e. $\langle T, \mathcal{P}\rangle_{/P_p,P_q} \stackrel{def}{=} \langle T, \mathcal{P} \setminus \{P_p, P_q\} \cup \{P_p \cup P_q\}\rangle$. We term this operation a *union*, and note the following important property of split vertices with respect to the union operation:

Observation 3.9. *For every $P_p, P_q, P_r \in \mathcal{P}$ such that $P_p \sim P_q$, split$(P_p \cup P_q, P_r) =$ split$(P_p, P_r) \cup$ split(P_q, P_r).*

Lemma 3.10. *Let $\langle T, \mathcal{P}\rangle$ be a representation for the pair (G, G'), and let $e = \{p, q\} \in E(G')$. Then $G_{/e}$ is an* EPT *graph. Moreover $G_{/e} = \textsc{Ept}(\langle T, \mathcal{P}\rangle_{/P_p,P_q})$.*

Proof. Let s be the vertex of $G_{/e}$ created by the contraction of e. We claim that s corresponds to the path $P_s = P_p \cup P_q$. Consider a path $P_r \in \mathcal{P} \setminus \{P_p, P_q\}$. We observe that $\{r, s\} \in E(G_{/e}) \iff \{r, p\} \in E(G)$ or $\{r, q\} \in E(G)$ (by definition of the contraction operation) $\iff P_r$ intersects with at least one of P_p and P_q in T (because $G = \textsc{Ept}(\mathcal{P})$) $\iff P_r$ intersects $P_p \cup P_q = P_s$ in $T \iff \{r, s\} \in E(\textsc{Ept}(\langle T, \mathcal{P}\rangle_{/P_p,P_q}))$. $\square$

We now extend the definition of the contraction operation to pairs. Based on Observation 3.9, the contraction of an ENPT edge does not necessarily preserve ENPT edges. More concretely, let P_p,P_q and $P_{q'}$ such that $P_p \sim P_q$, $P_p \sim P_{q'}$ and $P_q \nsim P_{q'}$. Then $G'_{/p,q}$ is not isomorphic to $\textsc{Enpt}(\langle T, \mathcal{P}\rangle_{/P_p,P_q})$ as $\{q', p.q\} \notin E(\textsc{Enpt}(\langle T, \mathcal{P}\rangle_{/P_p,P_q}))$. Let (G, G') be a pair and $e \in E(G')$. If for every edge $e' \in E(G')$ incident to e, the edge $e'' = e \triangle e'$ (forming a triangle together with e and e') is not an edge of G then $(G, G')_{/e} \stackrel{def}{=} (G_{/e}, G'_{/e})$, otherwise $(G, G')_{/e}$ is undefined. Whenever $(G, G')_{/e}$ is defined we say that (G, G') is *contractible* on e, and when there is no ambiguity about the pair under consideration we say that e is *contractible.* A pair (G, G') is *contractible* if it contains at least one contractible edge. Clearly, (G, G') is non-contractible if and only if every edge of G' is contained in at least one BBR triangle.

HamiltonianPairRec
Input: A pair (G, C_n) where C_n is a Hamiltonian cycle of G
Output: A minimal representation $\langle T, \mathcal{P} \rangle$ of (G, C_n) if such a representation exists, "NO" otherwise.

Figure 3.4. The general pair recognition problem.

P3-HamiltonianPairRec
Input: A pair (G, C_n) where C_n is a Hamiltonian cycle of G and $n \geq 4$.
Output: A minimal representation $\langle T, \mathcal{P} \rangle$ of (G, C_n) that satisfies $(P3)$ if such a representation exists, "NO" otherwise.

Figure 3.5. The pair recognition problem under assumption $(P3)$.

Problem Definition. Our goal in this work is to characterize the representations of ENPT holes. More precisely we characterize representations of pairs (G, C_n) where C_n is a Hamiltonian cycle of G. For this purpose we define the following problem.

The ENPT representations of C_3 are characterized by Lemma 3.3. Therefore we assume $n > 3$, which implies that (G, C_n) does not contain blue triangles. In the sequel we confine ourselves to pairs (G, C_n) and representations $\langle T, \mathcal{P} \rangle$ satisfying the following three assumptions:

- $(P1)$: (G, C_n) is not contractible.
- $(P2)$: (G, C_n) is (K_4, P_4)-free, i.e., it does not contain an induced sub-pair isomorphic to a (K_4, P_4).
- $(P3)$: Every red triangle of (G, C_n) is a claw-clique, i.e. corresponds to a pie of $\langle T, \mathcal{P} \rangle$.

Note that $(P1)$ and $(P2)$ are assumptions about the pair (G, C) and $(P3)$ is an assumption about the representation $\langle T, \mathcal{P} \rangle$. We say that $(P3)$ holds for a pair (G, C) if it has a representation $\langle T, \mathcal{P} \rangle$ satisfying $(P3)$. It will be convenient to define the following problem.

Without loss of generality we let $V(G) = V(C_n) = \{0, 1, \ldots, n-1\}$ where the numbering of the vertices is consistent with their order in C. All arithmetic operations on vertex numbers are done modulo n. We denote the corresponding set of paths in the representation as $\mathcal{P} = \{P_0, \ldots, P_{n-1}\}$.

3.4. Pairs (G, C) Satisfying $(P1), (P2)$ and $(P3)$

In this section we characterize the minimal representations of (G, C) pairs satisfying $(P1)$, $(P2)$ and $(P3)$. To achieve this goal we present an algorithm solving the P3-HamiltonianPairRec problem for instances satisfying $(P1)$ and $(P2)$. In Section 3.4.1 we handle the case $n = 4$. In Section 3.4.2 we analyze properties of weak dual trees based on which, in Section 3.4.3 we present an algorithm for the case $n > 4$. C_4 is exceptional because all its representations satisfy assumptions $(P1-3)$, but some of our results that we prove for $n > 4$ fail to hold in this case.

3.4.1. The pair (G, C_4)

Lemma 3.11. *(i) All the representations of* (G, C_4) *satisfy assumptions* $(P1-3)$*, (ii)* G *is one of the two graphs in Figure 3.6, and (iii) each of these two graphs has a unique minimal representation (also depicted in Figure 3.6).*

Proof. (i) (G, C_4) is clearly (K_4, P_4)-free. Moreover it satisfies $(P3)$ vacuously, because it does not contain any red triangle. $G \neq C_4$, because otherwise C_4 would constitute a blue hole of length 4, contradicting Lemma 3.8. Without loss of generality let $\{1, 3\}$ be a red edge of G. We observe that $\{1, 3\}$ is incident to all the edges of C_4, therefore (G, C_4) is not contractible, so it satisfies $(P1)$.

(ii) Depending on whether or not $\{0, 2\} \in E(G)$, G is one of the two graphs in Figure 3.6.

(iii) Consider a representation $\langle T, \mathcal{P} \rangle$ of (G, C_4), and consider the path $P = P_1 \cap P_3$. Let e_0 (resp. e_2) be an edge defining the edge-clique $\{1, 3, 0\}$ (resp. $\{1, 3, 2\}$).

Both of e_0 and e_2 are in P. Let $u \in split(P_1, P_3)$. u is an endpoint of P. As P_0 intersects P (at e_0), it can not cross u, because in this case it has to split from at least one of P_1, P_3 at u. The same holds for P_2. Therefore neither one of P_0, P_2 crosses a vertex of $split(P_1, P_3)$. We consider two cases: (Case 1) G is isomorphic to K_4. Then there is one edge defining the clique, i.e. without loss of generality $e_0 = e_2$. If $|split(P_1, P_3)| = 2$ then, none of these two vertices can be crossed by P_0 or P_2. Therefore $P_0 \subseteq P$ and $P_2 \subseteq P$, we conclude that they can not split, a contradiction. Therefore $split(P_1, P_3)$ consists of one vertex that is not crossed by P_0 and P_2. We conclude that P_0 and P_2 cross the other endpoint of P and split. The representation in Figure 3.6a is the only minimal representation satisfying these conditions. (Case 2) G is not isomorphic to K_4. Therefore $e_0 \neq e_2$, and without loss of generality $e_0 \in P_0 \setminus P_2, e_2 \in P_2 \setminus P_0$. P has at least one endpoint u in $split(P_1, P_3)$. Without loss of generality e_0 is closer to u than e_2. Therefore P_0 lies between u and e_2, and P_2 starts after P_1 and crosses e_2. The representation in Figure 3.6b is the only minimal representation satisfying these conditions. □

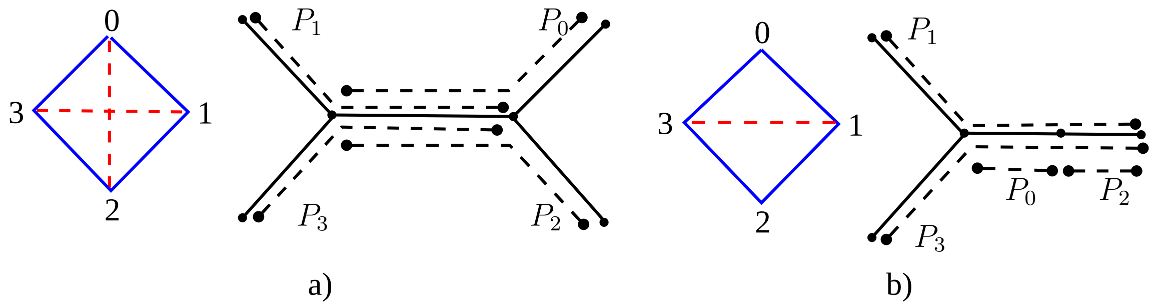

Figure 3.6. The two minimal ENPT representations of C_4.

3.4.2. Weak Dual Trees

We extend the definition of the weak dual tree of Hamiltonian outerplanar graphs to any Hamiltonian graph as follows. Given a pair (G, C) where C is a Hamiltonian cycle of G, a weak dual tree of (G, C) is the weak dual tree $\mathcal{W}(G, C)$ of an arbitrary Hamiltonian maximal outerplanar subgraph $\mathcal{O}(G, C)$ of G. $\mathcal{O}(G, C)$ can be built by starting from C and adding to it arbitrarily chosen chords from G as long as such chords exist and the resulting graph is planar. We note that under the assumptions of

$(P1-3)$, G will be shown to be outerplanar, and therefore there is actually one weak dual tree.

By definition of a dual graph, vertices of $\mathcal{W}(G,C)$ correspond to faces of $\mathcal{O}(G,C)$. By maximality, the faces of $\mathcal{O}(G,C)$ correspond to holes of G. The degree of a vertex of $\mathcal{W}(G,C)$ is the number of red edges in the corresponding face of $\mathcal{O}(G,C)$. To emphasize the difference, for an outerplanar graph G we will refer to *the* weak dual tree of G, whereas for a (not necessarily outerplanar) graph G we will refer to *a* weak dual tree of G.

We proceed with observations on $\mathcal{W}(G,C)$:

- Edges of $\mathcal{W}(G,C)$ correspond to red edges of $\mathcal{O}(G,C)$ (by definition of a weak dual graph, and observing that the edges of the unbounded face are exactly the blue edges).
- The degree of a vertex of $\mathcal{W}(G,C)$ is the number of red edges in the corresponding face of $\mathcal{O}(G,C)$, therefore the leaves (resp. intermediate vertices, junctions) of $\mathcal{W}(G,C)$ correspond to BBR triangles (resp. BRR triangles, red holes) of (G,C) (recalling Lemma 3.8).
- $|V(G)| = |V(C)| = |E(C)| = 2\ell + i$ where ℓ is the number of leaves of $\mathcal{W}(G,C)$ and i is the number of its intermediate vertices.

Lemma 3.12. *Let $n > 4$ and (G, C_n) be a pair satisfying $(P1-3)$. Then every edge of C_n is in exactly one BBR triangle.*

Proof. Let $\langle T, \mathcal{P}\rangle$ be a representation of (G, C_n) satisfying $(P3)$. As (G, C_n) is not contractible, every edge of C_n is in at least one BBR triangle. Assume, by contradiction and without loss of generality, that the blue edge $\{1,2\}$ is part of the two possible BBR triangles $\{0,1,2\}$ and $\{1,2,3\}$. $\{0,3\}$ is not an edge of C_n, because $n > 4$. Moreover, it is not an edge of G, because otherwise the sub-pair induced by $\{0,1,2,3\}$ is isomorphic to a (K_4, P_4). Let e_0 (resp. e_3) be an edge of T defining the edge-clique $\{0,1,2\}$ (resp. $\{1,2,3\}$) such that e_3 is closest to e_0. Clearly, $e_0 \neq e_3$, because otherwise we get a

(K_4, P_4). The vertices $\{4, \ldots, n-1\}$ constitute a connected component of G, therefore the union of the corresponding paths is a subtree T' of T. T' intersects both P_0 and P_3, therefore there is at least one path $P_j \notin \{P_0, P_1, P_2, P_3\}$ that contains e_3. We conclude that $\{1, 2, 3, j\}$ is an edge-clique. If $j = 4$ then it induces a pair isomorphic to (K_4, P_4), otherwise $\{1, 3, j\}$ is a red edge-clique. Both cases contradict our assumptions. $\square$

Lemma 3.13. *Let (G, C) be a pair satisfying $(P2), (P3)$ and let $\mathcal{W}(G, C)$ be a weak dual tree of (G, C). Then (i) there is a bijection between the contractible edges of (G, C) and the intermediate vertices of $\mathcal{W}(G, C)$, (ii) the tree obtained from $\mathcal{W}(G, C)$ by smoothing out the intermediate vertex corresponding to a contractible edge e is a weak dual tree of $(G, C)_{/e}$.*

Proof. (i) We define the bijection f as follows: Let $\mathcal{W}(G, C)$ be the weak dual tree corresponding to some $\mathcal{O}(G, C)$, and let e be a contractible edge of (G, C). Then e is not part of any BBR triangle. As every blue edge must be in some triangle, e is in a non-empty set of BRR triangles. Exactly one of these triangles is in $\mathcal{O}(G, C)$, and this triangle corresponds to an intermediate vertex of $\mathcal{W}(G, C)$ that we designate as $f(e)$. f is one-to-one because every intermediate vertex corresponds to one BRR triangle of $\mathcal{O}(G, C)$, and every BRR has one blue edge. We now show that f is onto. Assume by contradiction that f is not onto. Then, without loss of generality there is a BRR triangle $\{1, 2, j\}$ $(j \notin \{0, 1, 2, 3\})$ of $\mathcal{O}(G, C)$ where $e = \{1, 2\}$ is not contractible. Then either $\{0, 2\}$ or $\{1, 3\}$ is an edge of $E(G) \setminus E(C)$. Let, without loss of generality $\{0, 2\}$ be an edge of $E(G) \setminus E(C)$. Then $\{0, 1, 2\}$ is an edge-clique. Let E' be the set of edges of (the path of) T defining this edge-clique. We claim that $\forall k \notin \{0, 1, 2\}, P_k \cap E' = \emptyset$. Indeed, if $k = n-1$ and P_k contains an edge of E', then $\{n-1, 0, 1, 2\}$ induces a (K_4, P_4), and if $k \neq n-1$ then $\{k, 0, 2\}$ induces a red edge-clique. In both cases we reach a contradiction. Consider the subtrees of T separated by E'. As $P_j \cap E' = \emptyset$ it is completely contained in one of these subtrees, say T_j. P_1 and P_2 intersect P_j, therefore they intersect T_j. However P_0 does not intersect T_j as this would either contradict the definition of E' or P_0 would split from P_1. On the other hand, the vertices $\{j+1, j+2 \ldots, 0\}$ constitute a connected component of G, therefore the union of the paths $\{P_{j+1}, P_{j+2} \ldots, P_0\}$ is a subtree T' of T. T' intersects both P_0 and P_j, therefore T' intersects E'. In other words there is at least one path $P_l \in \{P_{j+1}, P_{j+2} \ldots, P_0\}$ that intersects E', a contradiction.

(ii) Let $e = \{i, i+1\}$ and $\{i, i+1, j\}$ the BRR triangle of $\mathcal{O}(G, C)$ (that corresponds to $f(e)$). After the contraction of e, this triangle reduces to a red edge. The same holds for $\mathcal{O}(G, C)_{/e}$ that contains all the faces of $\mathcal{O}(G, C)$ except the BRR triangle that disappeared. The corresponding weak dual tree is $\mathcal{W}(G, C)$ with $f(e)$ smoothed out. □

We note that if $n = 4$ Lemma 3.12 does not hold. However the following corollary of lemmata 3.12 and 3.13 holds for every n.

Corollary 3.14. *If (G, C) is a pair satisfying $(P1-3)$ with C isomorphic to C_n, then: (i) $\mathcal{W}(G, C)$ does not have intermediate vertices, (ii) n is even and $\mathcal{W}(G, C)$ has $n/2$ leaves, and (iii) $\mathcal{W}(G, C)$ is a path if and only if $n = 4$.*

3.4.3. The Minimal Representation

In this section we present an algorithm solving P3-HamiltonianPairRec for $n \geq 5$, provided that assumptions $(P1) - (P2)$ hold. The representation returned by the algorithm is a planar tour. We show that it is the unique minimal representation of (G, C) satisfying $(P3)$.

Lemma 3.15. *If (G, C) is a hamiltonian pair with $n = |V(G)| > 4$ for which properties $(P1 - 3)$ hold then G is outerplanar and the unique minimal representation of (G, C) satisfying $(P3)$ is a planar tour of the weak dual tree of G.*

Proof. The proof is by induction on the smallest number h of junctions of a weak dual tree of (G, C). Let $\mathcal{W}(G, C)$ be a weak dual tree of (G, C) with h junctions, and $\mathcal{O}(G, C)$ the corresponding maximal outerplanar graph. Index arithmetic is modulo n through the proof, and $\langle T, \mathcal{P} \rangle$ is a minimal representation of (G, C) satisfying $(P3)$. We first recall that since (G, C) satisfies $(P1)$, by Corollary 3.14, $\mathcal{W}(G, C)$ contains only junctions and leaves. In the sequel we show that T is isomorphic to $\mathcal{W}(G,C)$ and $\mathcal{P}$ is a planar tour of T. We do this by combining planar tours of subtrees into a planar tour a tree. Two basic tools that we use in the construction are the following two claims that state, roughly speaking, (i) that two adjacent holes of $\mathcal{O}(G,C)$ are represented by two pies with distinct centers, and (ii) that the representations associated with disjoint subtrees of $\mathcal{W}(G,C)$, reside in disjoint subtrees of T.

For a junction x of $\mathcal{W}(G,C)$, let H_x be the set of vertices of the hole corresponding to x in $\mathcal{O}(G,C)$. By Property $(P3)$, H_x is represented by a pie. We denote by $f(x)$ be the center of the pie in $\langle T,\mathcal{P}\rangle$ representing H_x.

Claim 3.16. *If u,v are two adjacent junctions of $\mathcal{W}(G,C)$ then $f(u)\neq f(v)$.*

Proof. Let $\{i,j\}$ be the edge common to the holes H_u and H_v. In other words $\{i,j\}$ is the dual of the edge $\{u,v\}$ of $\mathcal{W}(G,C)$. Let $k\neq i$ and $k'\neq i$ be the vertices adjacent to j in these two holes. Assume for a contradiction that $f(u)=f(v)$ (see Figure 3.7 for an illustration). P_i,P_j,P_k are consecutive in one pie and $P_i,P_j,P_{k'}$ are consecutive in the other. Then P_k and $P_{k'}$ intersect P_j on the same edge (incident to $f(u)$), thus forming an edge-clique of G. We will show that this is a red edge-clique of G, contradicting $(P3)$. Indeed, $\{j,k\}$ and $\{j,k'\}$ are red edges of $\mathcal{O}(G,C)$. If $\{k,k'\}$ is a blue edge then $\{j,k,k'\}$ constitutes a BRR triangle of $\mathcal{O}(G,C)$ that corresponds to an intermediate vertex of $\mathcal{W}(G,C)$, contradicting Corollary 3.14. $\square$

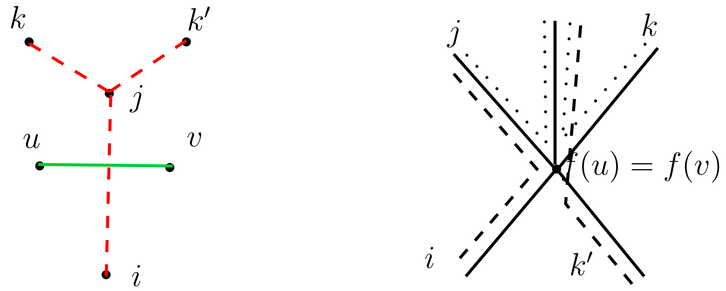

Figure 3.7. Adjacent holes of $\mathcal{O}(G,C)$ are mapped to different centers.

Claim 3.17. *Let u be a junction of degree d of a weak dual tree of (G,C). Let $S_1,S_2,\ldots,S_d$ be the connected components of $C\setminus H_u$, and let $\mathcal{P}_i$ be the set of paths* *representing the vertices of* S_i *in a minimal representation. Then* H_u *is represented by a pie with edges* $e_1, e_2, \ldots, e_d$ *whose removal together with* $f(u)$ *divides* T *into subtrees* $T_1, T_2, \ldots, T_d$*, such that:*

(i) $\cup \mathcal{P}_i \subseteq T_i + e_i$ *for every* $i \in [d]$*,*

(ii) $\cup \mathcal{P}_i \subseteq T_i$ *whenever* S_i *is not a singleton, and*

(iii) $E(T_i) = \emptyset$ *whenever* S_i *is a singleton.*

Proof. (i,ii) The removal of $f(u)$ from T (together with its incident edges) defines at least d subtrees $T_1, T_2, \ldots T_d$ of T where e_i has one endpoint in T_i for $i \in [d]$. We consider two vertices i, j consecutive on the hole H_u. Without loss of generality P_i contains e_0 and e_1, P_j contains e_1 and e_2. Consider the *segment* (i.e. connected component) $S = \{i+1, i+2, \ldots, j-1\}$ of $G \setminus H_u$. We will conclude the proof of (i,ii) by showing that $\cup \mathcal{P}_S \subseteq T_1$ where $\mathcal{P}_S$ is the set of paths in $\mathcal{P}$ that represent the vertices of S.

If S contains at least two vertices, then the hole adjacent to H_u is a red hole H_v. By Claim 3.16, $f(u) \neq f(v)$. Since $f(u), f(v) \in \mathit{split}(P_i, P_j)$ and $|\mathit{split}(P_i, P_j)| \leq 2$ this implies that $\mathit{split}(P_i, P_j) = \{f(u), f(v)\}$.

Let $P = p_T(f(u), f(v))$ and let $T_u, T_v, T'_1, T'_2, \ldots$ be the trees of the forest obtained by the removal of the edges of P from T, where $V(T_u) \cap V(P) = f(u)$, $V(T_v) \cap V(P) = f(v)$ and $V(T'_\ell) \cap V(P)$ is an intermediate vertex of P. We observe that if a path P_k intersects some subtree T'_ℓ in at least one edge and also intersects P, then $\{i, j, k\}$ is a red edge-clique, contradicting property $(P3)$. Therefore, every path intersecting T'_ℓ is contained in T'_ℓ implying that set of vertices represented by such paths are disconnected from the rest of G, contradicting the connectedness of G. We conclude that the tree T'_ℓ contains no paths and since $\langle T, \mathcal{P} \rangle$ is minimal, T'_ℓ consists of a single vertex, namely an intermediate vertex of P. Summarizing, we have $T = T_u \cup T_v \cup P$. Finally, we note that $T_1 = P \cup T_v$. In the sequel we show that $\mathcal{P}_S \subseteq T_v$ and P consists of the edge e_1.

Let $S' = S \backslash \{i+1, j-1\}$. H_v contains at least one vertex $k \in S'$, and P_k is part of the pie centered at $f(v)$. Therefore, $P_k \subseteq T_v$, implying that $\mathcal{P}_{S'} \cap T_v \neq \emptyset$. If $P_{k'}$ crosses v for some $k' \in S'$ then $\{i, j, k'\}$ constitutes a red edge clique, contradicting property $(P3)$. Therefore, $\cup \mathcal{P}_{S'} \subseteq T_v$. We now show that $P_{i+1} \cup P_{j-1} \subseteq T_v$. Since $i+1$ (resp. $j-1$) is adjacent to $i+2 \in S'$ (resp. $j-2 \in S'$), both of P_{i+1} and P_{j-2} intersect $\cup \mathcal{P}_{S'}$ implying that the both intersect T_v. We now consider the BBR triangle $j, j+1, j+2$. We have $P_j \nsim P_{j+2}$, therefore $\emptyset \neq \mathit{split}(P_j, P_{j+2}) \subseteq V(P_{j+2}) \subseteq V(T_v)$. Let x be vertex of $\mathit{split}(P_j, P_{j+2})$ closest to v (possibly $x = v$). Assume, by way of contradiction that P_{j+1} crosses v. If P_{j+1} does not cross x then $P_{j+1} \parallel P_{j+2}$, otherwise $P_{j+1} \nsim P_{j+2}$ or $P_{j+1} \nsim P_j$. Both cases contradict the fact that $j, j+1, j+2$ are consecutive in C. Therefore, P_{j+1} does not cross v, i.e. $P_{j+1} \subseteq T_v$. Similarly, $P_{i-1} \subseteq T_v$. We conclude that $\cup \mathcal{P}_S \subseteq T_v$. Since the only paths intersecting P are P_i and P_j, and by the minimality of the representation, P consists of only one edge, namely e_1, concluding the proof of (i) and (ii) for this case. Otherwise, S is a singleton. Then $S = \{i+1\}$ and $i, i+1, j$ are consecutive in C. Therefore, $P_{i+1} \sim P_i$ and $P_{i+1} \sim P_j$. Then, P_{i+1} is contained in $T_1 + e_1$

(iii) If S is a singleton, the only paths intersecting $T_1 + e_1$ are P_i, P_{i+1}, P_j, since all the other paths are in their respective subtrees, each disjoint from T_1. Together with the minimality of the representation, this implies that P_{i+1} consists of the single edge e_1 and $E(T_1) = \emptyset$. $\square$

We now proceed with the proof the lemma. If $h = 0$, $\mathcal{W}(G, C)$ contains at most two vertices implying that $n \leq 4$. Therefore, $h \geq 1$.

Consult Figures 3.8a and 3.8b for the following discussion. To keep the figure simple, most of the paths are omitted and the segments having more than one vertex, i.e. S_1, S_3, S_5, are depicted by arcs. Let u be a junction of $\mathcal{W}(G, C)$, $H_u = \{h_0, h_1, \ldots, h_{d-1}\}$, S_i be the segment of $C \setminus H_u$ between h_i and h_{i+1}, and let $e_1, \ldots, e_d$, $T_1, \ldots, T_d$ as in Claim 3.17.

If $h = 1$ all the segments $S_i, i \in [0, d-1]$ are singletons. Then $T_i = \emptyset$ and the only vertex $h_i + 1$ of S_i is represented by a path consisting of e_i, for every $i \in [0, d-1]$. Therefore, T is a star isomorphic to $\mathcal{W}(G, C)$ and $\mathcal{P}$ is a planar tour of it.

If $h > 1$ for a singleton segment S_i we have $T_i = \emptyset$ and the only vertex $h_i + 1$ of S_i is represented by a path consisting of e_i. For a segment S_i consisting of at least two vertices we proceed as follows: the edge h_i, h_{i+1} separates H_u from another red hole H_v where $f(u) \neq f(v)$ by Claim 3.16. This implies that $\mathit{split}(P_{h_i}, P_{h_{i+1}}) = \{f(u), f(v)\}$, and without loss of generality $f(v) \in V(T_i)$.

For the following discussion see Figures 3.8c and 3.8d. Let $\bar{S}_i = S_i \cup \{h_i, h_{i+1}\}$ and let $(\bar{G}_i, \bar{C}_i)$ be the pair obtained from the pair $(G[\bar{S}_i], C[\bar{S}_i])$ by adding to it a new vertex v_i and two edges $\{v_i, h_i\}, \{v_i, h_{i+1}\}$. Let also $\bar{T}_i = T_i + e_i$ and $\bar{\mathcal{P}}_i = \mathcal{P}_{S_i} \cup \left\{P_{h_i} \cap \bar{T}_i, P_{h_{i+1}} \cap \bar{T}_i, P_{v_i}\right\}$ where P_x is the path consisting of the edge e_i. Then $\left\langle \bar{T}_i, \bar{\mathcal{P}}_i \right\rangle$ is a representation of $(\bar{G}_i, \bar{C}_i)$, since the paths P_{h_i} and $P_{h_{i+1}}$ split in a vertex $f(v)$ of T_i and the other parts are completely contained in T_i, by Claim 3.17.

We note that a minifying operation applied to an edge of T_i to get a representation equivalent to $\left\langle \bar{T}_i, \bar{\mathcal{P}}_i \right\rangle$ can be applied to $\langle T, \mathcal{P} \rangle$ to get an equivalent representation to $\langle T, \mathcal{P} \rangle$ contradicting the minimality of $\langle T, \mathcal{P} \rangle$. Moreover, any minifying operation on e_i will cause P_{v_i} to have an empty intersection with P_{h_i} or with $P_{h_{i+1}}$. Therefore, $\left\langle \bar{T}_i, \bar{\mathcal{P}}_i \right\rangle$ is minimal. By the inductive hypothesis (i) $\bar{G}_i$ is outerplanar, (ii) $\bar{\mathcal{P}}_i$ is a planar tour of $\bar{T}_i$, and, (iii) $\bar{T}_i$ is isomorphic to the weak dual tree of $\bar{G}_i$. Consider $\mathcal{W}(G, C)$ as rooted at u, and let $\mathcal{W}_i$ be the subtree of u containing v. We note that the weak dual tree of $\bar{G}_i$ is isomorphic to $\mathcal{W}_i$.

It remains to observe that T is the union of all trees $\bar{T}_i$, i.e. isomorphic to $\mathcal{W}(G, C)$, and that $\mathcal{P}$ is a planar tour of it. G is outerplanar, since each G_i is outerplanar, and if G is not outerplanar, then there is an edge between a vertex of S_i and a vertex of S_j for $i, j \in [d]$ and $i \neq j$. However, this is a contradiction to the fact that, by Claim 3.17, $\cup S_i \subseteq \bar{T}_i$, $\cup S_j \subseteq \bar{T}_j$ and $\bar{T}_i, \bar{T}_j$ are edge-disjoint.

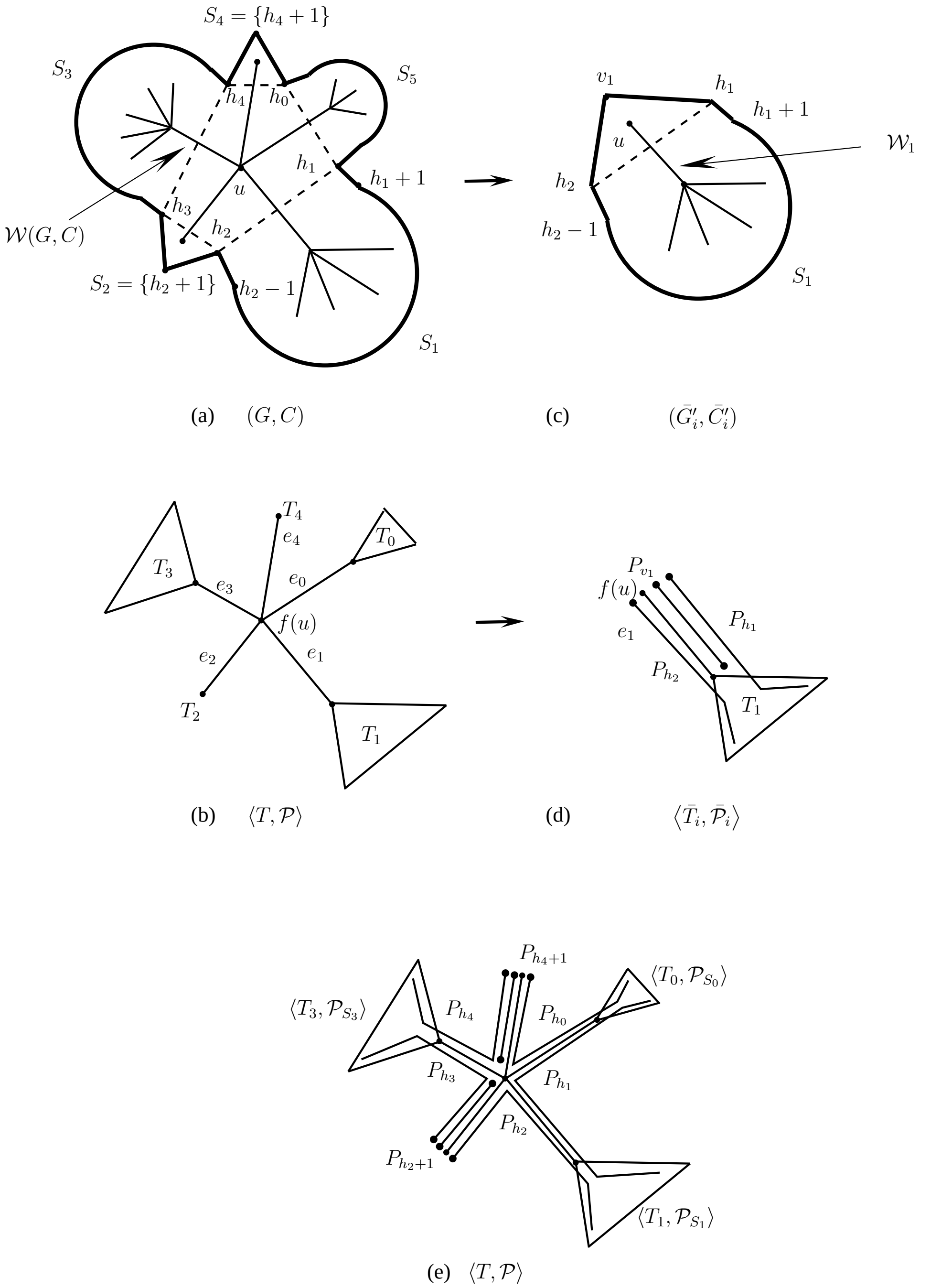

Figure 3.8. (a), (b), (c), (d) An induction step of the proof of Lemma 3.15 illustrated for $d = 5$, (e) The unique minimal representation of (G, C) satisfying $(P3)$ obtained by combining the subtrees and the paths of the segments with a representation of a hole H_u.

□

Lemma 3.18. *If (G, C) is a hamiltonian pair on $n > 4$ vertices and G is outerplanar such that every edge of the outer face of G is contained in a BBR triangle, then properties $(P1-3)$ hold for (G, C).*

Proof. (G, C_n) satisfies $(P1)$ since every edge of C_n is in a BBR triangle. Since G is outerplanar it does not contain a K_4. Therefore, (G, C_n) satisfies $(P2)$. To prove that $(P3)$ holds, we show by induction on the number h of junctions of the weak dual tree $\mathcal{W}(G, C)$ of G that a planar tour of $\mathcal{W}(G, C)$ is a representation of (G, C_n) satisfying $(P3)$:

If $h = 1$ then $\mathcal{W}(G, C)$ is a star, and $G = \mathcal{O}(G, C)$ is a hole surrounded by BBR triangles. Then, a planar tour of $\mathcal{W}(G, C)$ is a representation of (G, C_n) satisfying $(P3)$.

If $h > 1$ we pick an chord $e = \{i, j\}$ of C separating two red holes of G and construct two pairs $(G', C'), (G'', C'')$ in a way similar to the proof of Lemma 3.15. $V' = \{i, i+1, \ldots, j\}$ and $V'' = \{j, j+1, \ldots, i\}$. (G', V') (resp. (G'', V'')) consists of $(G[V'], C[V'])$ (resp. $(G[V''], C[V'']))$ and an additional vertex with two adjacent edges closing the cycle. By the inductive assumption, a planar tour of $\mathcal{W}(G', C')$ (resp. $\mathcal{W}(G'', C''))$ is a representation of (G', C') (resp. $(G'', C''))$. Removing from these planar tours the short paths corresponding to the vertices not in $V(G)$ and gluing together the rest by identifying the common endpoints of the paths P_i, P_j we get a planar tour of $\mathcal{W}(G, C)$ that represents (G, C). □

We get the following theorem as a corollary of lemmata 3.12, 3.15 and 3.18.

Theorem 3.19. *The following statements are equivalent whenever $n > 4$:*

(i) *(G, C_n) satisfies assumptions $(P1-3)$.*

(ii) *(G, C_n) has a unique minimal representation satisfying $(P3)$ which is a planar tour of a weak dual tree of G.*

(iii) G is Hamiltonian outerplanar and every face adjacent to the unbounded face F is a triangle having two edges in common with F, (i.e. a BBR triangle).

Algorithm BuildPlanarTour calculates a planar tour of a weak dual tree $\mathcal{W}(G, C)$. Therefore,

Theorem 3.20. *Instances of* P3-HamiltonianPairRec *satisfying properties* $(P1), (P2)$ *can be solved in polynomial time.*

Require: $|V(G)| \geq 5$
Require: (G, C) satisfies assumptions $(P1), (P2)$
Ensure: $\langle \bar{T}, \bar{\mathcal{P}} \rangle$ is the unique minimal representation of (G, C) satisfying $(P3)$
  if G is not outerplanar **then**
    return "NO"
  $\bar{T} \leftarrow \mathcal{W}(G, C)$. ▷ Corresponding to $\mathcal{O}(G, C)$
  Build the planar tour:
  Let $\{v_0, v_1, \ldots, v_{k-1}\}$ be the leaves of $\bar{T}$ ordered as they are encountered in a DFS traversal of $\bar{T}$ corresponding to the planar embedding suggested by $\mathcal{O}(G, C)$.
  Let $L_i = p_{\bar{T}}(v_i, v_{(i+1) \bmod k})$
  Let S_i be the path of length 1 starting at v_i.
  $\bar{\mathcal{P}}_L \leftarrow \{L_i | \ 0 \leq i \leq n-1\}$.
  $\bar{\mathcal{P}}_S \leftarrow \{S_i | \ 0 \leq i \leq n-1\}$.
  Let $\bar{P}_i = \begin{cases} L_{i/2} & \text{if } i \text{ is even} \\ S_{\lfloor i/2 \rfloor} & \text{otherwise} \end{cases}$
  $\bar{\mathcal{P}} \leftarrow \left\{\bar{P}_i | \ 0 \leq i \leq 2n-1\right\}$ ▷ $= \bar{\mathcal{P}}_L \cup \bar{\mathcal{P}}_S$
  return $\langle \bar{T}, \bar{\mathcal{P}} \rangle$

Figure 3.9. BuildPlanarTour(G, C) algorithm.

Figure 3.10 depicts a YES instance of P3-HamiltonianPairRec.

3.5. Pairs (G, C) Satisfying $(P2)$ and $(P3)$

In this section we characterize the minimal representations of (G, C) pairs satisfying $(P2)$ and $(P3)$. Similar to the previous section, our first goal is to present an algorithm solving the P3-HamiltonianPairRec problem for instances satisfying $(P2)$. In other words, our goal is to extend Theorem 3.20 to pairs that do not satisfy $(P1)$, i.e. which are contractible. In Section 3.5.1 we investigate the properties of the

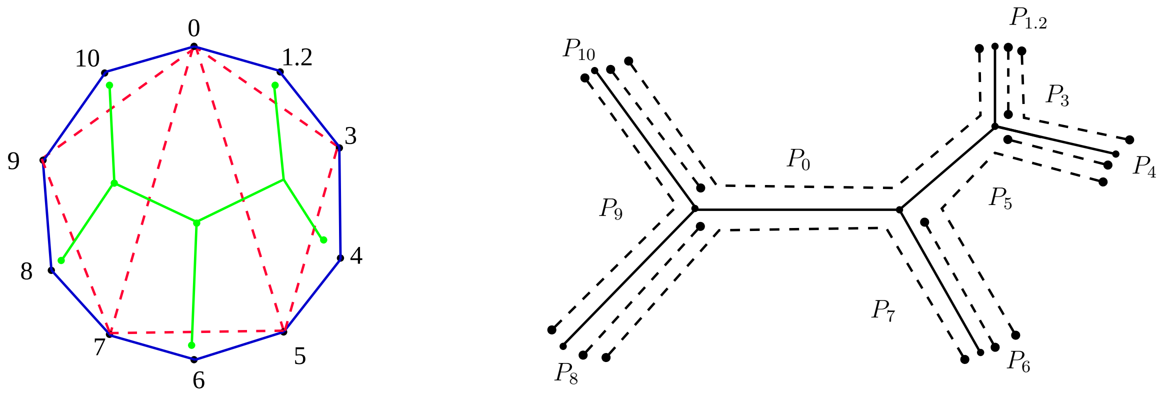

Figure 3.10. A pair (G, C), its weak dual tree $\mathcal{W}(G, C)$ and the representation of (G, C) returned by BuildPlanarTour.

contraction operation, in Section 3.5.2 we analyze the special case of $n \leq 6$ and finally in Section 3.5.3 we present the algorithm and characterization for $n > 6$.

3.5.1. Contraction of Pairs

In this section we show that (i) the contraction operation preserves ENPT edges, (ii) the order of contractions is irrelevant, and (iii) the contraction operation preserves $(P2)$, $(P3)$.

Lemma 3.21. *Let $\langle T, \mathcal{P} \rangle$ be a representation for the pair (G, G'), and let $e = \{p, q\} \in E(G')$. If $(G, G')_{/e}$ is defined then $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$ is a representation for the pair $(G, G')_{/e}$.*

Proof. By Lemma 3.10 $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$ is an EPT representation for $G_{/e}$. It remains to show that it is an ENPT representation for $G'_{/e}$, i.e. that for any two paths $P_{p'}, P_{q'} \in \langle T, \mathcal{P} \rangle_{/P_p, P_q}$, the edge $e' = \{p', q'\}$ is in $E(G'_{/e}) \iff P_{p'} \sim P_{q'}$. Let $P_s = P_p \cup P_q$ and s be the vertex obtained by the contraction. We assume first that $P_s \notin \{P_{p'}, P_{q'}\}$. Then $e' \in E(G'_{/e}) \iff e' \in E(G') \iff P_{p'} \sim P_{q'}$ as required. Now we assume without loss of generality that $P_{p'} = P_s$ and we recall that $e = \{p, q\} \in E(G')$ is the contracted edge. We have to show that $e' = \{s, q'\} \in E(G'_{/e}) \iff P_s \sim P_{q'}$.

We observe that

$$\begin{aligned}\{s,q'\} \in E({G'}_{/e}) &\iff \{p,q'\} \in E(G') \vee \{q,q'\} \in E(G') \\ &\iff P_p \sim P_{q'} \vee P_q \sim P_{q'}\end{aligned} \tag{3.1}$$

and

$$\begin{aligned}P_s \sim P_{q'} &\iff (P_p \cup P_q) \cap P_{q'} \neq \emptyset \wedge \mathit{split}(P_p \cup P_q, P_{q'}) = \emptyset \\ &\iff (P_p \cap P_{q'} \neq \emptyset \vee P_q \cap P_{q'} \neq \emptyset) \wedge \mathit{split}(P_p, P_{q'}) = \emptyset \wedge \mathit{split}(P_q, P_{q'})\end{aligned} \tag{3.2}$$

Clearly, (3.2) implies (3.1). To conclude the proof, assume that (3.1) holds. Then $P_p \cap P_{q'} \neq \emptyset \vee P_q \cap P_{q'} \neq \emptyset$. Now assume, by way of contradiction, that (3.2) does not hold. Then $\mathit{split}(P_p, P_{q'}) \neq \emptyset \vee \mathit{split}(P_q, P_{q'}) \neq \emptyset$ implying $P_p \nsim P_{q'} \vee P_q \nsim P_{q'}$. Combining with (3.1) this implies that exactly one of $P_p \sim P_{q'}$ and $P_q \sim P_{q'}$ holds. Therefore without loss of generality $P_p \sim P_{q'}, P_q \nsim P_{q'}$. Then $e' = \{p,q'\} \in E(G')$ and $e \triangle e' \in E(G)$, therefore $(G,G')_{/e}$ is undefined, thus constituting a contradiction to the assumption of the lemma. $\square$

Let $\bar{E} = \{e_1, e_2, \ldots, e_k\} \subseteq E(G')$. For every $k > 1$ we define $(G,G')_{/e_1,\ldots,e_k} \stackrel{def}{=} (G,G')_{/e_1,\ldots,e_{k-1}/e_k}$ provided that both contractions on the right hand side are defined, otherwise it is undefined. The following Lemma follows from Lemma 3.21 and states that the order of contraction of the edges is irrelevant.

Lemma 3.22. *Let (G,G') be a pair, $\bar{E} = \{e_1, e_2, \ldots, e_k\} \subseteq E(G)$, and π a permutation of the integers $\{1,\ldots,k\}$. Then $(G,G')_{/e_1,\ldots,e_k}$ is defined if and only if $(G,G')_{/e_{\pi(1)},\ldots,e_{\pi(k)}}$ is defined. Moreover when they are defined $(G,G')_{/e_1,\ldots,e_k} = (G,G')_{/e_{\pi(1)},\ldots,e_{\pi(k)}}$.*

Proof. Assume that $(G,G')_{/e_1,\ldots,e_k}$ is defined. By $k-1$ successive applications of Lemma 3.21 we conclude that a representation of $(G,G')_{/e_1,\ldots,e_k}$ can be obtained from a representation of (G,G') by applying a sequence of $k-1$ union operations. The result of $k-1$ union operations yield a set of paths. As union is commutative and associative, this result will remain the same, (i.e. a set of paths) if we change the order of the operations. On the other hand as union preserves split vertices, and the result does not contain split vertices, there are no split vertices at any given step of new sequence of union operations. We conclude that $(G,G')_{/e_{\pi(1)},\dots,e_{\pi(k)}}$ is defined. The other direction holds by symmetry. Whenever both $(G,G')_{/e_1,\dots,e_k}$ and $(G,G')_{/e_{\pi(1)},\dots,e_{\pi(k)}}$ are defined we have $(G,G')_{/e_1,\dots,e_k}$=$(G_{/e_1,\dots,e_k}, G'_{/e_1,\dots,e_k})$= $(G_{/e_{\pi(1)},\dots,e_{\pi(k)}}, G'_{/e_{\pi(1)},\dots,e_{\pi(k)}}) = (G,G')_{/e_{\pi(1)},\dots,e_{\pi(k)}}$. □

Based on this result, we denote the contracted pair as $(G,G')_{/\bar{E}}$ and say that $\bar{E}$ is contractible.

If $\langle T_2, \mathcal{P}_2\rangle = \langle T_1, \mathcal{P}_1\rangle_{/P_p,P_q}$ for two paths $P_p, P_q \in \mathcal{P}_1$ we denote this by $\langle T_1, \mathcal{P}_1\rangle \rightsquigarrow_U \langle T_2, \mathcal{P}_2\rangle$. The relation $\gtrsim_U$ is the reflexive-transitive closure of $\rightsquigarrow_U$, and $\langle T_1, \mathcal{P}_1\rangle \lesssim_U \langle T_2, \mathcal{P}_2\rangle$ is equivalent to $\langle T_2, \mathcal{P}_2\rangle \lesssim_U \langle T_1, \mathcal{P}_1\rangle$.

$(G_1, G_1') \rightsquigarrow_C (G_2, G_2')$ if $(G_2, G_2') = (G_1, G_1')_{/e}$ for some $e \in E(G_1')$. The relation $\gtrsim_C$ is the reflexive-transitive closure of $\rightsquigarrow_C$, and $\langle T_1, \mathcal{P}_1\rangle \lesssim_C \langle T_2, \mathcal{P}_2\rangle$ is equivalent to $\langle T_2, \mathcal{P}_2\rangle \lesssim_C \langle T_1, \mathcal{P}_1\rangle$.

By Lemma 3.21, $\lesssim_U$ is homomorphic to $\lesssim_C$.

Following the above definitions, a non-contractible pair of graphs is said to be *contraction-minimal*, because it is minimal in the partial order $\lesssim_C$.

We proceed by showing that the contraction operation preserves assumptions $(P2), (P3)$.

Lemma 3.23. *Let $\{p,q,r\}$ be a BBR triangle of $(G,G')_{/e}$ with $\{p,r\}$ being the red edge. Then $q \in V(G)$, i.e. q is not the vertex obtained by the contraction.*

Proof. Assume, by contradiction that $e = \{q', q''\}$ and q is the vertex obtained by the contraction of e. Assume without loss of generality that $\{p, q'\}$ and $\{q'', r\}$ are edges of G'. Then both $\{p, q''\}$ and $\{r, q'\}$ are non-edges of G, because otherwise e is not contractible. Then $\{p, q', q'', r\}$ is a hole of size 4 with blues edges, a contradiction. □

Lemma 3.24. *(i) If $(P2)$ holds for (G, G') then $(P2)$ holds for $(G, G')_{/e}$.*
(ii) If $(P3)$ holds for (G, G') then $(P3)$ holds for $(G, G')_{/e}$.

Proof. (i) Assume, by contradiction, that (G, G') does not have an induced sub-pair isomorphic to (K_4, P_4) and without loss of generality $(G, G')_{/e}$ has a sub-pair isomorphic to (K_4, P_4) induced by the vertices $U = \{p, q, r, s\}$ where p and s are the endpoints of the subgraph isomorphic to P_4. Let v be the vertex created by the contraction of e. If $v \notin U$ then the sub-pair induced by U is also a sub-pair of (G, G'), contradicting our assumption. Therefore $v \in U$. By Lemma 3.23 we have that $v \notin \{q, r\}$. Therefore, let without loss of generality $v = p$, $e = \{p', p''\}$ and p'' is adjacent to q in G'. $\{p', q\}$ is a non-edge of G, because e is contractible. As $\{p, s\}$ and $\{p, r\}$ are edges of $G_{/e}$, $\{p', s\}$ or $\{p'', s\}$ is an edge of G, and $\{p', r\}$ or $\{p'', r\}$ is an edge of G. If $\{p'', s\}$ is an edge of G then $\{p'', r\}$ is a non-edge of G since otherwise $\{p'', q, r, s\}$ induce a sub-pair isomorphic to (K_4, P_4) in (G, G'). Therefore $\{p', r\}$ is an edge of G. Then $\{p', p'', q, r\}$ induces a hole of size 4 with blue edges, a contradiction. Thus $\{p', s\}$ is an edge of G, since $\{p', q\}$ and $\{p'', s\}$ are non-edges of G, $\{p', p'', q, s\}$ induce a hole of length 4 with blue edges, a contradiction.

(ii) Assume, by contradiction, that (G, G') has a representation $\langle T, \mathcal{P} \rangle$ satisfying $(P3)$ and that no representation of $(G, G')_{/e}$ satisfies $(P3)$. Let v be the vertex created by the contraction of $e = \{v', v''\}$. Then by Lemma 3.10 $\langle T, \mathcal{P} \rangle_{/P_{v'}, P_{v''}}$ is a representation of $(G, G')_{/e}$ and it contains a red edge-clique $\{p, q, r\}$. If $v \notin \{p, q, r\}$ then $\{p, q, r\}$ is an edge-clique of $\langle T, \mathcal{P} \rangle$, contradicting our assumption. Assume without loss of generality that $v = p$. Let e_0 be an edge of T defining the edge-clique $\{v, q, r\}$. Since $e_0 \in P_{v'} \cup P_{v''}$, without loss of generality $e_0 \in P_{v'}$. Then $\{v', q, r\}$ induces an edge-clique on e_0. This is clearly a red edge-clique since e is contractible, a contradiction. □

We now describe how minimal representations of $(G, G')_{/e}$ can be obtained from minimal representations of (G, G').

Lemma 3.25. *Let $\langle T, \mathcal{P} \rangle$ be a minimal representation, $\langle T', \mathcal{P}' \rangle$ a representation such that $\langle T', \mathcal{P}' \rangle \lesssim \langle T, \mathcal{P} \rangle_{/P_p, P_q}$ and $\langle T', \mathcal{P}' \rangle \approxeq \langle T, \mathcal{P} \rangle_{/P_p, P_q}$. Let e be an edge of T involved in a minimal sequence of minifying operations s that obtains $\langle T', \mathcal{P}' \rangle$ from $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$. There is an operation of s and a path P such that the operation removes e from P ($tr(P, e)$, or $contract(e)$ and $e \in P$) where at least one of the following holds:*
(i) e is a tail of $P_p \cap P_q$, $P \cap P_p \cap P_q = \{e\}$ and $P \cap (P_p \cup P_q) \supsetneq \{e\}$.
(ii) e is incident to an internal vertex u of $P_p \cup P_q$, e is a tail of P, P is not in a pie with center u.

Proof. Let $G = \text{EPT}(\mathcal{P})$ and $G' = \text{ENPT}(\mathcal{P})$ and consider an operation *op* of s. Without loss of generality we can assume that *op* is the first operation of s, by Lemma 3.6. Furthermore, by Lemma 3.7, the representation obtained by applying *op* is also equivalent to $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$. Therefore without loss of generality *op* is the only operation of s. Note that *op* is defined on $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$ except when *op* is $tr(P_p \cup P_q, e)$. In this case e is a tail of P_p or P_q (or both). In the following discussion, whenever we apply *op* to $\langle T, \mathcal{P} \rangle$, we mean that we apply $tr(P_p, e)$ or $tr(P_q, e)$ one of which is well defined on $\langle T, \mathcal{P} \rangle$.

By the minimality of $\langle T, \mathcal{P} \rangle$, *op* cannot be applied to $\langle T, \mathcal{P} \rangle$. More precisely, if *op* is applied, either an edge of G becomes a non-edge, or a red edge of (G, G') becomes a blue edge. We term such an edge of (G, G') *an affected edge* and the corresponding paths of $\langle T, \mathcal{P} \rangle$ *affected pair of paths.* Let $\{r, s\}$ be an affected edge of (G, G'). If $\{P_r, P_s\} \cap \{P_p, P_q\} = \emptyset$ then P_r, P_s is a pair of affected paths in $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$, contradicting the fact that *op* can be applied to $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$. We conclude that $\{P_r, P_s\} \cap \{P_p, P_q\} \neq \emptyset$. Assume without loss of generality that $P_s = P_p$, i.e. P_r, P_p is an affected pair of paths. We consider two disjoint cases:

Case 1) $\{r, p\}$ becomes a non-edge after applying *op*. Then $P_r \cap P_p = \{e\}$ for some edge e of T, and after the removal of e the intersection becomes empty. On the other hand $P_r \cap (P_p \cup P_q) \supsetneq \{e\}$, because otherwise P_r and $P_p \cup P_q$ constitute an affected pair of paths in $\langle T, \mathcal{P} \rangle_{/P_p,P_q}$. Then e is a tail of $P_p \cap P_q$ (see Figure 3.11a) and P_r is the claimed path P (note that possibly $r = q$ as opposed to the figure). In this case i) holds.

Case 2) $\{r, p\}$ is a red edge, and it becomes a blue edge after applying *op*. Then $P_r \not\sim P_p$ (therefore $r \neq q$) and P_r, P_p do not split after applying *op*. Therefore *split*$(P_r, P_p) = \{u\}$ for an endpoint u of e, e is a tail of exactly one of P_r, P_p, and u is an internal vertex of P_p thus of $P_p \cup P_q$. Let $\{P, P'\} = \{P_r, P_p\}$ such that e is a tail of P, and $e \notin P'$. If P is not in a pie with center u then ii) holds. Otherwise P has two neighbors $\{P', P''\}$ in this pie. $e \in P''$ because $e \notin P$ and e is an edge incident to u, the center of the pie. Recalling that e is a tail of P we conclude that after the removal of e from P, its intersection with P'' becomes empty. Therefore i) holds. $\square$

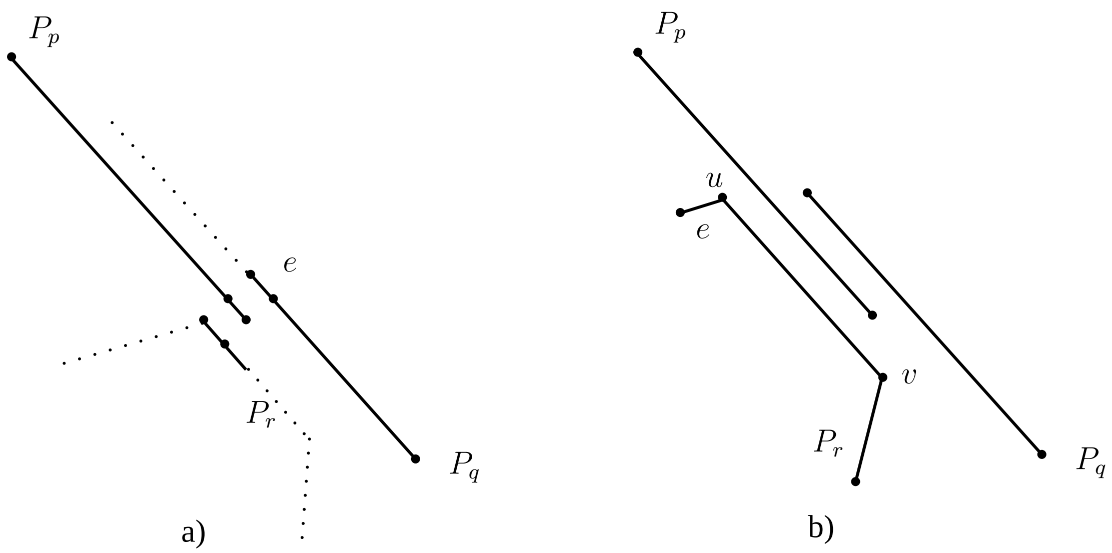

Figure 3.11. Possible minifying operations on $\langle T, \mathcal{P} \rangle_{/P_p,P_q}$.

Lemma 3.26. *Every split vertex of a path P of a broken planar tour is a center of a pie containing P.*

Proof. By construction, every split vertex of a path P of a tour is a center of a pie containing P. We will show that the same holds for a broken planar tour. Let P'_1, P'_2 be two paths of a broken planar tour such that $v \in$ *split*(P'_1, P'_2). These paths are sub-paths of two paths P_1, P_2 of a tour and $v \in$ *split*(P_1, P_2). Then v is a center of a pie containing P_1, P_2 and also other paths. Each one of the other paths has at least one sub-path in the broken planar tour that crosses v. These paths, together with P'_1, P'_2 constitute a pie with center v of the broken planar tour. We conclude that every split vertex of P is a center of a pie, and therefore case (ii) of Lemma 3.25 is impossible. $\square$

We notice that by Lemma 3.26 it follows that the case (ii) of Lemma 3.25 is impossible in the context of this section.

We now return to the study of the representations of pairs (G', C') satisfying $(P2), (P3)$. Without loss of generality we let $V(G') = V(C') = \{0, 1, \ldots, n-1\}$, $n \geq 5$ and note that all arithmetic operations on vertex numbers are done modulo n.

3.5.2. Small Cycles: the pairs (G, C_5) and (G, C_6)

In this Section we analyze the special cases of $n \in \{5, 6\}$. This cases are special because our technique for the general case is based on contraction of cycles to smaller ones and assumes that the representation of a non-contractible pair is a planar tour (Theorem 3.20). However this theorem does not hold when $n = 4$. The following lemma analyzes the case $n = 5$. We note that in this case $(P3)$ holds vacuously.

Lemma 3.27. *If (G', C_5) satisfies $(P2)$ then (i) G' is the graph depicted in Figure 3.12, and (ii) (G', C_5) has a unique minimal representation also depicted in Figure 3.12.*

Proof. (i) G' contains at least two non-crossing red edges, because otherwise there is a hole of size 4 with blue edges. Without loss of generality, let these edges be $\{1, 3\}$ and $\{1, 4\}$. If one of $\{2, 4\}$ or $\{0, 3\}$ is a red edge, then we have a (K_4, P_4), contradicting our assumption. If $\{0, 2\}$ is a red edge, then we have a hole of size 4 containing blue edges, contradicting Lemma 3.8. Therefore $\{1, 3\}$ and $\{1, 4\}$ are the only red edges in this pair.

(ii) We contract $\{3, 4\}$ of (G', C_5) and obtain the pair (G, C_4) with one red edge $\{1, 3.4\}$. This pair has a unique minimal representation $\langle T', \mathcal{P}' \rangle$ characterized in [20].

Any representation of (G', C_5) is obtained by splitting the path $P'_{3,4}$ of $\langle T', \mathcal{P}' \rangle$ into two overlapping paths and making sure that both of them split from P_1. This leads to the minimal representation depicted in Figure 3.12. $\square$

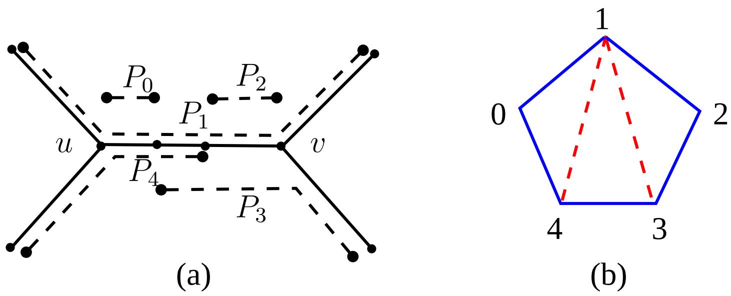

Figure 3.12. (a) The unique ENPT representation of C_5 satisfying $(P2)$ and (b) corresponding pair (G, C_5).

Lemma 3.28. *If (G', C_6) satisfies $(P2)$ and $(P3)$ then it is not contractible, i.e. it satisfies $(P1)$.*

Proof. Assume, by way of contradiction, that (G', C_6) satisfies $(P2)$ $(P3)$ and the edge $e = \{0,1\}$ is contractible. Therefore, $\{0,2\}$ and $\{5,1\}$ are non-edges of G'. $\{2,5\}$ is also a non-edge, because otherwise $\{0,1,2,5\}$ is a hole of size 4 with blue edges. Then $\{0,1\}$ must be in a BRR triangle. From the two possible options remaining, assume without loss of generality that this triangle is $\{0,1,4\}$. At least one of $\{2,4\}$ and $\{1,3\}$ is an edge of G' because otherwise $\{1,2,3,4\}$ is a hole of size 4 with blue edges. On the other hand, if both of them are edges then $\{1,2,3,4\}$ is a (K_4, P_4), a contradiction. Therefore exactly one of them is an edge of G'. We analyze these cases separately.

- $\{2,4\}$ is an edge of G', $\{1,3\}$ is not an edge of G': In this case $\{0,3\}$ is not an edge, because otherwise $\{0,1,2,3\}$ is a hole of size 4 with blue edges. Then $\{3,5\}$ is not an edge, because otherwise $\{0,1,2,3,5\}$ is a hole of size 5 with blue edges. Then $\{5,0,1,2,3\}$ induces a path on 4 vertices in G'. Since none of the paths P_0, P_1, P_2, P_3, P_5 split from another, their union is a graph with maximum degree two, i.e. every representation of them is an interval representation where no three paths intersect at one edge. Now $P_4 \sim P_5$ and $P_4 \sim P_3$. Therefore, P_4 intersects all of P_0, P_1 and P_2 and does not split from them. Then $\{4,0\}, \{4,1\}, \{4,2\}$ are blue edges, a contradiction.

- $\{1,3\}$ is an edge of G', $\{2,4\}$ is not an edge of G': Assume by way of contradiction $\{0,1\}$ is contracted, the contracted pair is the same as the pair in Figure 3.12b where contracted edge $\{0,1\}$ corresponds to vertex 1 of (G, C_5). We will show that 1 can not be a vertex obtained by a contraction. Let $\{1', 1''\}$ be the contracted edge. For the following discussion consult Figure 3.12a. One endpoint of each one of $P_{1'}$, $P_{1''}$ is the same as the endpoints of P_1 since $P_1 = P_{1'} \cup P_{1''}$. $P_{1'}$ (resp. $P_{1''}$) can not cross v since otherwise $\{1', 2\}$ (resp. $\{0, 1''\}$) is a blue chord. $P_{1'} \sim P_{1''}$, therefore there exist some edge e such that $P_{1'} \cap P_{1''} \ni e$ and $e \in p_T(u, v)$. But $e \in p_T(u, v) \subseteq P_3 \cup P_4$ then either $\{1', 3\}$ or $\{1'', 4\}$ (or both) is a blue chord.

□

3.5.3. The General Case

Algorithm shown in Figure 3.13 is a recursive description of FindMinRep-P2-P3. It follows the paradigm of obtaining a non-contractible pair by successive contractions, and then reversing the corresponding union operations in the representation. The reversal of the union operation, i.e. the breaking apart of a path is done by duplicating the path and then moving one endpoint of each path to an appropriate internal vertex of the original path, and possibly subdividing an edge. The key to the correctness of the algorithm is the following lemma that, among others, enables us to consider at this stage, only one minifying operation.

Lemma 3.29. *Let $\langle T, \mathcal{P} \rangle$ be a minimal representation of (G, C), $\langle T', \mathcal{P}' \rangle$ a broken planar tour representation such that $\langle T', \mathcal{P}' \rangle \lesssim \langle T, \mathcal{P} \rangle_{/P_p, P_q}$ and $\langle T', \mathcal{P}' \rangle \approxeq \langle T, \mathcal{P} \rangle_{/P_p, P_q}$. Every operation in a minimal sequence of operations that obtains $\langle T', \mathcal{P}' \rangle$ from $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$ is a contract(e) operation, where e is a tail of $P_p \cap P_q$.*

Proof. Consider an operation in a minimal sequence of minifying operations as in the statement of the lemma. Let e be the edge involved in the operation, and let P_r be a path whose existence is guaranteed by Lemma 3.25. By Lemma 3.26, case (ii) of Lemma 3.25 is impossible. Then case (i) of the lemma holds, i.e. there is a path P_r

such that (i) the minifying operation removes e from P_r, (ii) e is a tail of $P_p \cap P_q$, c) $P_r \cap P_p \cap P_q = \{e\}$, and d) $P_r \cap (P_p \cup P_q) \supsetneq \{e\}$.

The minifying operation is either *contract*(e) or $tr(P_r, e)$. We will show that if $tr(P_r, e)$ can be applied, i.e. no affected pair after applying $tr(P_r, e)$, then *contract*(e) can also be applied. For the following discussion consult Figure 3.11a where *split*$(P_r, P_p) = \emptyset$, i.e. the dotted part of P_r adjacent to e in the figure, is empty.

Without loss of generality we assume that e is a tail of P_p. Since e is not a tail of $P_p \cup P_q$, we have $r \neq p.q.$ e divides T into two subtrees T_1, T_2. As e is a tail of P_p, P_p can not intersect both subtrees. We assume without loss of generality that $T_2 \cap P_p = \emptyset$. Let $\bar{\mathcal{P}}$ denote the set of paths of $\langle T, \mathcal{P} \rangle_{/P_p, P_q}$, i.e. $\bar{\mathcal{P}} = \mathcal{P} \setminus \{P_p, P_q\} \cup \{P_p \cup P_q\}$ and e' be the edge adjacent to e in $P_r \cap (P_p \cup P_q)$. Every path of $P \in \bar{\mathcal{P}}$ that contains e contains also e', because otherwise $P \cap P_r = \{e\}$ and (P, P_r) would constitute an affected pair of $tr(P_r, e)$. For $k \in \{1, 2\}$, let $\mathcal{P}_k = \left\{P \in \bar{\mathcal{P}} | \ P \cap T_k \neq \emptyset \wedge e \text{ is a tail of } P\right\}$. Note that by definition, $\mathcal{P}_1 \cap \mathcal{P}_2 = \emptyset$. As $e' \in T_2 \cap P_r$, we have $P_r \in \mathcal{P}_2$. We note that for every path $P_s \in \mathcal{P}_2$, $P_p \sim P_s$, i.e. $\{p, s\}$ is an edge of C. As the degree of p is 2 in C and both of q and r neighbors of p in C, we conclude that $\mathcal{P}_2 = \{P_r\}$. On the other hand, $\mathcal{P}_1 = \emptyset$ because for every path $P_s \in \mathcal{P}_1$, $\{s, r\}$ is an affected pair of $tr(P_r, e)$ (as $P_s \cap P_r = \{e\}$). Therefore $\mathcal{P}_1 \cup \mathcal{P}_2 = \{P_r\}$, i.e. the only path with tail e is P_r.

Assume by way of contradiction that there exists an affected pair $\{s, t\}$ of *contract*(e). As $e' \in P_s \cap P_t$, they intersect after the contraction. Therefore $\{s, t\}$ is a red-edge that becomes blue after the contraction. This can happen only if e is a tail of exactly one of P_s, P_t. Therefore, $r \in \{s, t\}$ from the above discussion. But then $\{s, t\}$ constitute an affected pair of $tr(P_r, e)$, contradicting to our initial assumption. We conclude that *contract*(e) has no affected pairs. $\square$

Theorem 3.30. *Instances of* P3-HamiltonianPairRec *satisfying property* $(P2)$ *can be solved in polynomial time. "YES" instances have a unique solution, and whenever* $n \geq 6$ *this solution is a broken planar tour.*

Require: $C' = \{0, 1, \ldots, |V(G')| - 1\}$ is an Hamiltonian cycle of G', $|V(G')| > 5$
Ensure: A minimal representation $\langle \bar{T}', \bar{\mathcal{P}}' \rangle$ of (G', C') satisfying $(P3)$ if any
  if (G', C') is contraction-minimal **then**
    if G' is outerplanar **then**
      return BuildPlanarTour(G', C')
    else
      return "NO"
  Contract:
  Pick an arbitrary contractible edge $e = \{i, i+1\}$ of C', $(G, C) \leftarrow (G', C')_{/e}$
  Let j be the vertex of (G, C) created by the contraction of the edge e
  Recurse:
  $\langle \bar{T}, \bar{\mathcal{P}} \rangle \leftarrow$ FindMinRep-P2-P3(G, C).
  Uncontract:
  $\langle \bar{T}', \bar{\mathcal{P}}' \rangle \leftarrow \langle \bar{T}, \bar{\mathcal{P}} \rangle$
  Let u and v be the endpoints of P_j such that u (resp. v) is contained in P_{i-1} (resp. P_{i+2})
  Replace $P_j \in \bar{\mathcal{P}}'$ by two copies P_i and P_{i+1} of itself
  AdjustEndpoint$(\langle \bar{T}', \bar{\mathcal{P}}' \rangle, G, i, u)$, AdjustEndpoint$(\langle \bar{T}', \bar{\mathcal{P}}' \rangle, G, i+1, v)$
  Validate:
  if Ept$(\bar{\mathcal{P}}') = G'$ and $\langle \bar{T}', \bar{\mathcal{P}}' \rangle$ satisfies $(P3)$ **then**
    return $\langle \bar{T}', \bar{\mathcal{P}}' \rangle$
  else
    return "NO"
  function AdjustEndpoint$(\langle \bar{T}, \bar{\mathcal{P}} \rangle, G, p, w)$ ▷ w will be adjusted
    e_w denotes the tail of P_p incident to w
    $\mathcal{X}_w$ denotes $\{P_x : e_w \in P_x \text{ and } \{p, x\} \notin E(G)\}$
    $\mathcal{Y}_w$ denotes $\{P_y : P_p \cap P_y = \{e_w\} \text{ and } \{p, y\} \in E(G)\}$
    while $\mathcal{Y}_w = \emptyset$ and $\mathcal{X}_w \neq \emptyset$ **do**
      $tr(P_p, e_w)$
    if $\mathcal{X}_w \neq \emptyset$ **then** ▷ Also $\mathcal{Y}_w \neq \emptyset$ as the while loop terminated
      Subdivide e_w into two edges e_w, e'_w ▷ Revert the minifying operation
      for $P_x \in \mathcal{X}_w$ **do**
        $tr(P_w, e_{w'})$
      $tr(P_p, e_w)$

Figure 3.13. FindMinRep-P2-P3(G', C') algorithm.

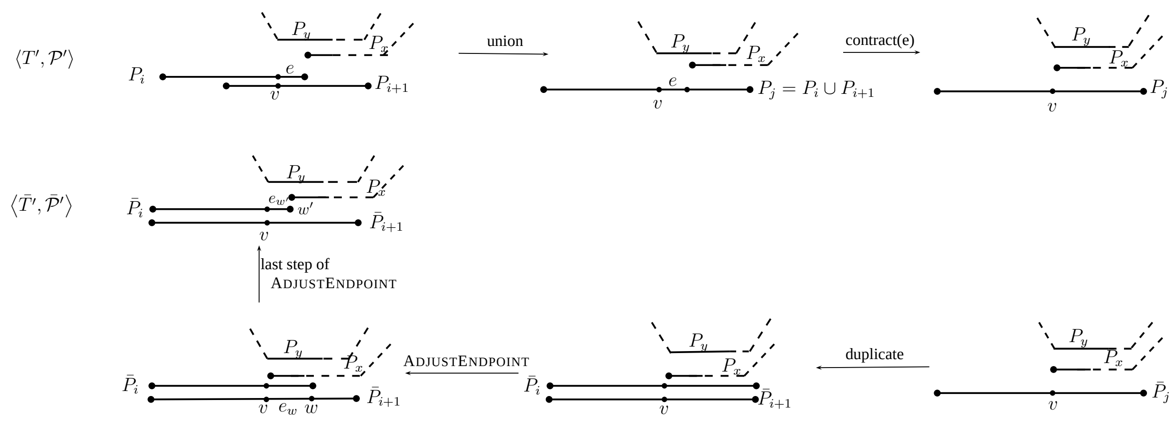

Figure 3.14. The effect of union and minifying operations, and the reversal of this effect by Procedure AdjustEndpoint (invoked with $p = i$).

Proof. If $n = 5$ the result follows from Lemma 3.27. If (G', C') is a "NO" instance, then FindMinRep-P2-P3 returns "NO" in the validation phase. Therefore we assume that $n \geq 6$, and that (G', C') is a "YES" instance, i.e. it has at least one representation satisfying $(P3)$. We will show that for any pair (G', C') satisfying $(P2)$, and a minimal representation $\langle T', \mathcal{P}' \rangle$ of (G', C') that satisfies $(P3)$, the representation $\left\langle \bar{T}', \bar{\mathcal{P}}' \right\rangle$ returned by FindMinRep-P2-P3 is a broken planar tour and that

$$\left\langle \bar{T}', \bar{\mathcal{P}}' \right\rangle \cong \langle T', \mathcal{P}' \rangle \text{ and } \left\langle \bar{T}', \bar{\mathcal{P}}' \right\rangle \lesssim \langle T', \mathcal{P}' \rangle .$$

We will prove by induction on the number k of contractible edges of (G', C'). If $k = 0$ then (G', C') is not contractible, therefore satisfies $(P1)$. In this case the algorithm invokes BuildPlanarTour and the claim follows from Theorem 3.20.

Otherwise $k > 0$. We assume that the claim holds for $k - 1$ and prove that it holds for k. As (G', C') contains at least one contractible edge, one such edge $\{i, i+1\}$ is chosen arbitrarily by the algorithm and contracted. The resulting pair $(G, C) = (G', C')_{/\{i,i+1\}}$ has the following properties:

- Satisfies $(P2), (P3)$. (By Lemma 3.24)
- The number of contractible edges is $k - 1$.

- $|V(G)| \geq 6$. This is because $|V(G)| = |V(G')| - 1$ and $|V(G')| > 6$. Indeed, if $|V(G')| = 6$, we have $k = 0$ by Lemma 3.28.

Therefore, (G, C) satisfies the assumptions of the inductive hypothesis. Let $\langle T', \mathcal{P}' \rangle$ be a minimal representation of (G', C') satisfying $(P3)$. Then $\langle T', \mathcal{P}' \rangle_{/P_i, P_{i+1}}$ is a representation of $(G, C) = (G', C')_{/\{i, i+1\}}$. By the inductive hypothesis, $\langle \bar{T}, \bar{\mathcal{P}} \rangle$ is a broken planar tour that satisfies

$$\langle \bar{T}, \bar{\mathcal{P}} \rangle \cong \langle T', \mathcal{P}' \rangle_{/P_i, P_{i+1}} \text{ and } \langle \bar{T}, \bar{\mathcal{P}} \rangle \lesssim \langle T', \mathcal{P}' \rangle_{/P_i, P_{i+1}}.$$

In other words $\langle \bar{T}, \bar{\mathcal{P}} \rangle$ is obtained from $\langle T', \mathcal{P}' \rangle$ by replacing the two paths P_i, P_{i+1} with the path $P_i \cup P_{i+1}$, then applying a (possibly empty) sequence of minifying operations. By Lemma 3.29, these minifying operations are *contract*(e) for a tail e of $P_i \cap P_{i+1}$. In the Uncontract phase, FindMinRep-P2-P3 performs a reversal of these transformations. See Figure 3.14 for the following discussion. One endpoint of each one of P_i and P_{i+1} is an endpoint of $P_i \cup P_{i+1}$. Therefore one needs to determine only one endpoint of each one of P_i and P_{i+1}. First $P_i \cap P_{i+1}$ is duplicated and the so obtained paths are called P_i, P_{i+1}.

For $p \in \{i, i+1\}$, let w be the endpoint of P_p to be adjusted. e_w denotes the tail of P_p incident to w. We denote by $\mathcal{X}_w$ the set of paths containing e_w such that vertices of G' corresponding to these paths are not adjacent to p. We denote by $\mathcal{Y}_w$ the set of paths intersecting P_p only on e_w and whose corresponding vertices in G' are adjacent to p. If $\mathcal{Y}_w$ is empty (that is, every path that intersects P_p also intersects $P_p \setminus \{e_w\}$, e_w can be safely removed from P_p without losing intersections. If $\mathcal{X}_w$ is non-empty this removal is a necessary operation. The algorithm performs these tail removals as long as they are necessary and safe. If at the end of this loop, $\mathcal{X}_w$ is empty then we are done. Otherwise $\mathcal{X}_w$ and $\mathcal{Y}_w$ are non-empty, then e_w can not be safely removed from P_p. In this case AdjustEndpoint subdivides e_w (thus reversing the minifying operation *contract*(e)) and removes one tail from P_p and one tail from every path $X \in \mathcal{X}_w$, so that P_p does not intersect X but still intersects every path $Y \in \mathcal{Y}_w$. $\square$

3.6. Pairs (G, C) Satisfying $(P3)$

In the previous section we relaxed assumption $(P1)$. In this section we relax assumption $(P2)$, i.e. we allow sub-pairs isomorphic to (K_4, P_4). In Section 3.6.1 we investigate the basic properties of the representations of such sub-pairs, and characterize the representations of pairs (G, C) with at most 6 vertices. In Section 3.6.2 we show that in bigger cycles such pairs can intersect only in a particular way, and we define the *aggressive contraction* operation that transforms a pair (G'', C'') with a (K_4, P_4) to a pair (G', C') with one less vertex and at least one (K_4, P_4) less. Using these results, in Section 3.6.3 we present an algorithm that finds the unique minimal representation of a given pair (G, C) satisfying $(P3)$ and having more than 6 vertices.

3.6.1. Representations of (K_4, P_4) and Small Cycles

We denote a set of 4 vertices inducing a sub-pair isomorphic to (K_4, P_4) as an ordered quadruple where the first vertex is one of the endpoints of the the induced P_4, the second vertex is its neighbor and so on. (p, q, r, s) is a (K_4, P_4) of (G, G') whenever $\{p, q, r, s\}$ induces a sub-pair (K_4, P_4) of (G, G') and p, s are the endpoints of the induced sub-path isomorphic to P_4. Clearly, $(p, q, r, s) = (s, r, q, p)$.

We start with a Lemma that characterize representations of (K_4, P_4) pairs in general. This lemma will be useful in developing our results. Then we present the unique minimal representation of (G, C_5) pairs containing a (K_4, P_4). Together with Lemma 3.27 this completes the characterization of all the (G, C_5) pairs because a (G, C_5) satisfies $(P3)$ vacuously. We continue by proving more properties of minimal representations of induce (K_4, P_4) sub-pairs of pairs (G, C) with at least 6 vertices. Using these properties we show that a (G, C_6) satisfying $(P3)$ does not contain sub-pairs isomorphic to (K_4, P_4).

Lemma 3.31. *Let* $K = (i, i+1, i+2, i+3)$ *be a* (K_4, P_4), $\langle T, \mathcal{P} \rangle$ *be a representation of* K, *and* $\bigcap \mathcal{P}_K \overset{def}{=} P_i \cap P_{i+1} \cap P_{i+2} \cap P_{i+3}$. *There is a path* $core(K)$ *of* T *with endpoints* u, v *such that:*

(i) split$(P_i, P_{i+2}) = \{u\}$, *split*$(P_{i+1}, P_{i+3}) = \{v\}$, *$P_{i+1}$ (resp. P_{i+2}) does not cross u (resp. v).*

(ii) $\emptyset \neq \bigcap \mathcal{P}_K \subseteq (P_{i+1} \cap P_{i+2}) \subseteq core(K)$*. In particular* $u \neq v$*.*

(iii) At least one of P_i, P_{i+3} crosses both endpoints of $core(K)$ and $\emptyset \neq$ *split*$(P_i, P_{i+3}) \subseteq \{u, v\}$*.*

(iv) $P_{i+1} \cup P_{i+2}$ crosses both endpoints of $core(K)$.

(v) The removal of the edges of $P_{i+1} \cup P_{i+2}$ from T disconnects P_i from P_{i+3}.

Proof. (i) Assume, by way of contradiction, that $|split(P_i, P_{i+2})| = 2$. Let these two vertices be w, w'. As $P_{i+1} \sim P_i$ and $P_{i+1} \sim P_{i+2}$ we conclude that $P_{i+1} \subseteq p_T(w, w')$. $P_{i+3} \nsim P_{i+1}$ therefore P_{i+3} splits from P_{i+1} in at least one vertex w'' that is an intermediate vertex of $p_T(w, w')$. Then P_{i+3} splits from P_{i+2} at w'' contradicting the fact that $\{i+2, i+3\}$ is an ENPT edge. Therefore $|split(P_i, P_{i+2})| = 1$ and by symmetry, $|split(P_{i+1}, P_{i+3})| = 1$. Let $split(P_i, P_{i+2}) = \{u\}$ and $split(P_{i+1}, P_{i+3}) = \{v\}$. We define $core(K) = p_T(u, v)$. For the rest of the claim, assume by contradiction that P_{i+1} crosses u. Then either $P_{i+1} \nsim P_i$ or $P_{i+1} \nsim P_{i+2}$, contradicting the the fact that $\{i, i+1\}$ and $\{i+1, i+2\}$ are ENPT edges.

At this point we can uniquely define the following edges that will be used in the rest of the proof: e_i (resp. e_{i+2}) is the edge of $P_i \setminus P_{i+2}$ (resp. $P_{i+2} \setminus P_i$) incident to $split(P_i, P_{i+2})$, and e_{i+1} and e_{i+3} are defined similarly. Note that $e_i \neq e_{i+2}$ and $e_{i+1} \neq e_{i+3}$, but the definition does not exclude the possibility that, for instance $e_i = e_{i+1}$.

(ii) A claw-clique of size 4 contains exactly one ENPT edge, however a path isomorphic to P_4 contains three edges. Therefore the representation of K_4 is an edge-clique. Let e be an edge defining this edge-clique, i.e. $e \in \bigcap \mathcal{P}_K$. The removal of e from T disconnects it into two subtrees. In order to prove that $\bigcap \mathcal{P}_K \subseteq core(K)$ it suffices to show that u and v are in different subtrees. Assume, by way of contradiction that u, v are in the same subtree T_r with root r where r is an endpoint of e. Let r' be the least common ancestor of u, v in T_r (possibly $u = v$ in which case $r' = u = v$). All the 4 paths contain e and cross r' (so that each one crosses at least one of u, v), i.e. they "enter"

r' from the same edge e' (where possibly $r' = r$ and $e' = e$). If $r' \notin \{u, v\}$ then as P_{i+1} crosses v and P_{i+2} crosses u, $r' \in \mathit{split}(P_{i+1}, P_{i+2})$, contradicting $P_{i+1} \sim P_{i+2}$. Therefore we can assume without loss of generality that $r' = u$. Then the edges e_i and e_{i+2} are incident to r'. Then P_{i+1} (resp. P_{i+3}) contains e_i (resp. e_{i+2}) because $P_{i+1} \sim P_i$ (resp. $P_{i+3} \sim P_{i+2}$). Therefore $r' \in \mathit{split}(P_{i+1}, P_{i+2})$, contradicting $P_{i+1} \sim P_{i+2}$. Therefore u and v are in different subtrees, i.e. $e \in p_T(u, v) = core(K)$. As e can be any edge defining the edge-clique this implies that $\bigcap \mathcal{P}_K \subseteq core(K)$. It remains to prove that $P_{i+1} \cap P_{i+2} \subseteq core(K)$. For this purpose, it is sufficient to show that both of P_{i+1} and P_{i+2} have one endpoint in $core(K)$. Indeed, assume without loss of generality that P_{i+1} does not have an endpoint in $core(K)$. Then P_{i+1} crosses u and does not include at least one of the edges e_i, e_{i+2}. Therefore $P_{i+1} \nsim P_i$ or $P_{i+1} \nsim P_{i+2}$, a contradiction.

Consult Figure 3.15 for the rest of the proof.

(iii) By the above discussion u (resp. v) is an intermediate vertex of P_i and P_{i+2} (resp.P_{i+1} and P_{i+3}), and they all intersect in at least one edge $e \in core(K)$. In order to see the first part of the claim assume, by way of contradiction, that both of P_i and P_{i+3} have an endpoint in $core(K)$, in this case $\bigcap \mathcal{P}_K$ is between these two endpoints. Therefore $P_i \sim P_{i+3}$, a contradiction.

We now proceed to show the rest of the claim: Let $w \in \mathit{split}(P_i, P_{i+3})$. $e_i \notin P_{i+3}$ because otherwise $P_{i+3} \nsim P_{i+2}$, and by symmetry $e_{i+3} \notin P_i$. Therefore, w is on $core(K)$. On the other hand w is not an intermediate vertex of $core(K)$. Indeed, consider the two sub-paths obtained by removing e from $core(K)$. If w is an intermediate vertex of $core(K)$, then at least one of $P_{i+3} \nsim P_{i+2}$, $P_i \nsim P_{i+1}$ holds, depending on the sub-path w belongs. We conclude $w \in \{u, v\}$. Together with $P_i \nsim P_{i+3}$, this implies the claim. Note that $\mathit{split}(P_i, P_{i+3}) = \{u, v\}$ if and only if both of P_i and P_{i+3} cross both endpoints u, v of $core(K)$.

(iv) As $\{i+1, i+2\}$ is an ENPT edge, $Q \stackrel{def}{=} P_{i+1} \cup P_{i+2}$ is a path. Moreover, $e_{i+1} \in P_{i+1}$ and $e_{i+2} \in P_{i+2}$, therefore $\{e_{i+1}, e_{i+2}\} \subseteq Q$, implying the claim.

(v) It suffices to show that $core(K)$ separates P_i and P_{i+3}. Suppose that after the removal of $core(K)$ the two paths still intersect. This is possible only if $e_{i+3} \in P_i$ or $e_i \in P_{i+3}$. Assume without loss of generality that $e_{i+3} \in P_i$. Then $P_i \nsim P_{i+1}$, a contradiction. $\square$

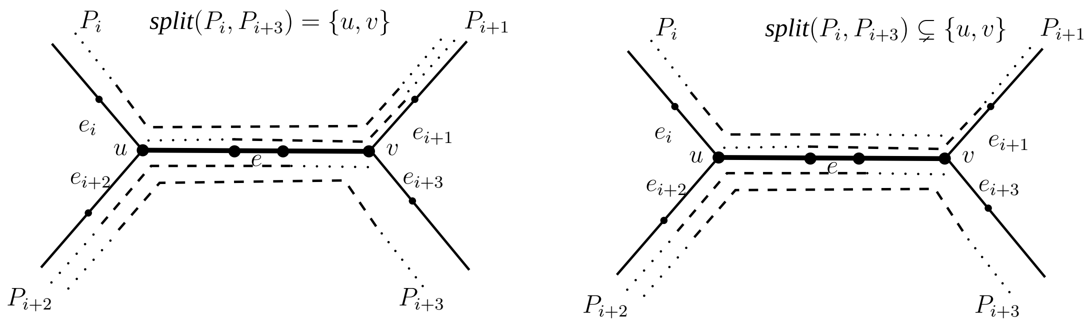

Figure 3.15. Representations of (K_4, P_4) pairs where $split(P_i, P_i + 3) = \{u, v\}$ and $split(P_i, P_i + 3) \subsetneq \{u, v\}$, respectively.

Pairs (G, C_5) with induced (K_4, P_4) pairs are different than bigger cycles in a few respects. Therefore we analyse this case separately. We recall that a pair (G, C_5) satisfies $(P3)$ vacuously, and that in Section 3.5.2 we found the unique minimal representation of a pair (G, C_5) that satisfies $(P2)$. We now investigate the representation of a pair (G, C_5) that does not satisfy $(P2)$.

Theorem 3.32. *If (G, C_5) does not satisfy $(P2)$ then (i) G is isomorphic to the graph depicted in Figure 3.16, and (ii) (G, C_5) has a unique minimal representation also depicted in Figure 3.16.*

Proof. Assume without loss of generality $K = (0, 1, 2, 3)$ is a (K_4, P_4) of (G, C_5), and let $core(K) = P_T u, v$. If $split(P_0, P_3) = \{u, v\}$ then $P_4 \subseteq core(K)$, implying that $P_4 \sim P_1$ or $P_4 \sim P_2$, i.e. at least one of $\{1, 4\}$ or $\{2, 4\}$ is an ENPT edge, a contradiction.

Assume without loss of generality $split(P_0, P_3) = \{u\}$, and that P_3 crosses both u and v. Then P_0 has one endpoint u' in $core(K)$, and $P_0 \cap P_3 = p_T(u, u')$.

As $P_4 \sim P_0$ and $P_4 \sim P_3$, we have $P_4 \sim p_T(u, u')$ and P_4 does cross u. Therefore P_4 intersects $core(K)$. By Lemma 3.31 (iv) $core(K) \subseteq P_1 \cup P_2$. We conclude that

$P_4 \cap (P_1 \cup P_2) \neq \emptyset$, i.e. $P_4 \cap P_1 \neq \emptyset$ or $P_4 \cap P_2 \neq \emptyset$. As $\{4,1\}$ and $\{4,2\}$ are not ENPT edges, we have that $P_4 \nsim P_1$ or $P_4 \nsim P_2$. On the other hand P_4 does not cross u and by Lemma 3.31 (i), P_2 does not cross v, thus $split(P_2, P_4) = \emptyset$. Therefore $P_4 \nsim P_1$ and $P_4 \parallel P_2$. Moreover, $split(P_4, P_1) = \{v\}$, i.e. P_4 crosses v. Therefore one endpoint u'' of P_4 is in $p_T(u, u')$, and must be between u' and the endpoint of P_1 in $core(K)$.

It is easy to see that the path $\bigcap \mathcal{P}_K$ can be contracted to one edge e without affecting the relationships between the paths. Similarly, any edge between u and e, and any edge between e and u'' can be contracted. The path $p_T(u', u'')$ can be contracted to one edge, and the path $p_T(u', v)$ can be contracted to a single vertex v. This leads to the representation in Figure 3.16. □

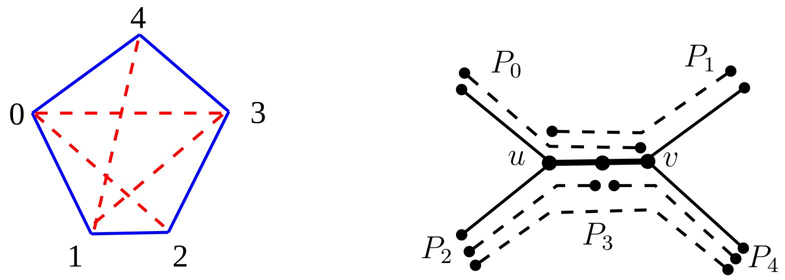

Figure 3.16. The unique (G, C_5) pair that does not satisfy $(P2)$ and its unique minimal representation.

We now observe a property of the representations of (G, C_5) in order to demonstrate the first family of non-ENPT graphs.

Theorem 3.33. *$G + C_5$ is not an* ENPT *graph whenever G is not a complete graph.*

Proof. A pair (G', C_5) satisfies $(P3)$ vacuously. If (G', C_5) satisfies $(P2)$ then by Lemma 3.27, its unique minimal representation is the one depicted in Figure 3.12. Otherwise, by Theorem 3.32, its unique minimal representation is the one depicted in Figure 3.16. Let $i \in V(G)$. i is adjacent to every vertex of C_5. We observe that in both cases above (i) P_i is a sub-path of $p_T(u, v)$, and (ii) there is a specific edge e of $p_T(u, v)$ that is also in P_i. Therefore, for any two vertices $i, j \in V(G)$ P_i and P_j are intersecting sub-paths of $p_T(u, v)$, thus $P_i \sim P_j$. We conclude that G is a complete graph. □

We now extend Lemma 3.31. As opposed to Lemma 3.31 that investigates the structure of a (K_4, P_4) regardless of any specific context, the next lemma provides us with further properties of minimal representations satisfying $(P3)$ of pairs (G, C).

Lemma 3.34. *Let $K = (i, i+1, i+2, i+3)$ be a (K_4, P_4) of a pair (G, C) satisfying $(P3)$ with at least 6 vertices. Let $\langle T, \mathcal{P} \rangle$ be a minimal representation of (G, C) and let $\mathcal{P}_K = \{P_i : i \in K\}$.*
(i) $\bigcap \mathcal{P}_K = \{e\}$ for some edge e which is used exclusively by the paths of $\mathcal{P}_K$, i.e. $e \in P_j \Rightarrow j \in K$.
(ii) e divides T into two subtrees T_1, T_2 such that T_1 is a cherry of $< T, \mathcal{P}_K >$ with center w_1. We denote this subtree as cherry(K).
(iii) split$(P_i, P_{i+3}) = \{w_2\} \subseteq V(T_2)$.
(iv) $N_G(j) = K$ if and only if split$(P_j, P_i) \cup$ split$(P_j, P_{i+3}) = \{w_1\}$. The unique vertex j satisfying this condition is one of $i+1, i+2$.

Proof. Consult Figure 3.17 for this proof.

(i) Let without loss of generality $i = 0$. By Lemma 3.31, $\bigcap \mathcal{P}_K$ is not empty. By contradiction assume that a path $P_j \notin \mathcal{P}_K$ intersects $\bigcap \mathcal{P}_K$. Then $K \cup \{j\}$ is an edge-clique of G. We claim that this K_5 contains at least one red triangle, contradicting $(P3)$. Indeed, as C has at least 6 vertices, j is adjacent in C to at most one vertex $k \in \{0, 3\}$. $K \setminus \{k\}$ contains one red edge. The endpoints of this edge together with j constitute a red edge-clique. Therefore, no path of $\mathcal{P} \setminus \mathcal{P}_K$ intersects $\bigcap \mathcal{P}_K$. Then no intermediate vertex of $\bigcap \mathcal{P}_K$ is a split vertex. By the minimality of $\langle T, \mathcal{P} \rangle$, $\bigcap \mathcal{P}_K$ consists of one edge, say e.

(ii) Let T_1, T_2 be the subtrees obtained by the removal of e from T. As $V(G) \setminus K$ is a connected component of G, the union of the paths $\mathcal{P} \setminus \mathcal{P}_K$ is a subtree T' of T. T' is a subtree of T_1 or a subtree of T_2, because otherwise there is at least one path of $\mathcal{P} \setminus \mathcal{P}_K$ using e, contradicting (i). Without loss of generality let T_2 be the subtree containing T', and T_1 be the subtree that intersects only paths of $\mathcal{P}_K$. By Lemma 3.31 (ii), T_1 contains exactly one endpoint of $core(K)$. For $i \in \{1, 2\}$, let w_i be the endpoint of $core(K)$ that is in T_i. w_1 is the only split vertex in T_1 because it contains only paths of $\mathcal{P}_K$. As the representation is minimal, there are no edges between e and w_1, as otherwise they could be contracted. Any subtree of T_1 starting with an edge incident to w_1 can be contracted to one path because the subtree does not contain split vertices. Moreover this path can be contracted to one edge, because all the paths entering the subtree intersect in its first edge. There are only two such subtrees, therefore T_1 is isomorphic to P_3 and w_1 is its center.

(iii) Assume that $|split(P_0, P_3)| = 2$. Then by Lemma 3.31, $split(P_0, P_3) = \{w_1, w_2\}$, i.e. w_1 is an internal vertex of both P_0 and P_3. In this case, one can remove from P_0 its unique edge in T_1 without affecting the relationships between the paths. This contradicts the minimality of $\langle T, \mathcal{P}\rangle$. Indeed, (i) any change in T_1 affects relationships between paths of $\mathcal{P}_K$ only, (ii) $\bigcap \mathcal{P}_K$ is not affected, therefore all the paths of $\mathcal{P}_K$ still intersect, (iii) $\{w_1\} = split(P_0, P_3) = split(P_0, P_2)$ and $\{w_2\} = split(P_1, P_3)$ hold after the tail removal.

Now assume that $split(P_0, P_3) = \{w_1\}$. P_0 crosses w_2 because $split(P_0, P_2) = \{w_2\}$. Then P_3 does not cross w_2. As $P_4 \sim P_3$, P_4, P_2, P_0 intersect in the last edge of P_3, and thus constitute a red edge-clique, contradicting $(P3)$. We conclude that $split(P_0, P_3) = \{w_2\}$.

(iv) First assume $j \notin \{i+1, i+2\}$. Clearly, $N_G(j) \neq K$. Moreover, we have $split(P_j, P_i) \cup split(P_j, P_{i+3}) \neq \{w_1\}$. Indeed, if $j \notin K$ then w_1 is not a vertex of P_j and if $j \in \{i, i+3\}$ the condition holds because (iii).

We now assume $j \in \{i+1, i+2\}$. By Lemma 3.31 (v), the removal of $P_1 \cup P_2$ disconnects P_0 from P_3. Then the tree T' intersects $P_1 \cup P_2$. Therefore, at least one of P_1, P_2 intersects T'. By Lemma 3.31 i) one of P_1, P_2 does not cross w_2, i.e. does not intersect T_2 which in turn includes T', a contradiction. We conclude that exactly one of P_1, P_2 intersects T'. In other words exactly one of $1, 2$ is adjacent to $V(G) \setminus K$. Assume $N_G(i+1) = K$. Then $P_{i+1} \cap T_1 \neq \emptyset$, therefore $split(P_{i+1}, P_i) = \emptyset$, i.e. $split(P_{i+1}, P_{i+3}) = \{w_1\}$, concluding the claim. The case $N_G(i+2) = K$ is symmetric. $\square$

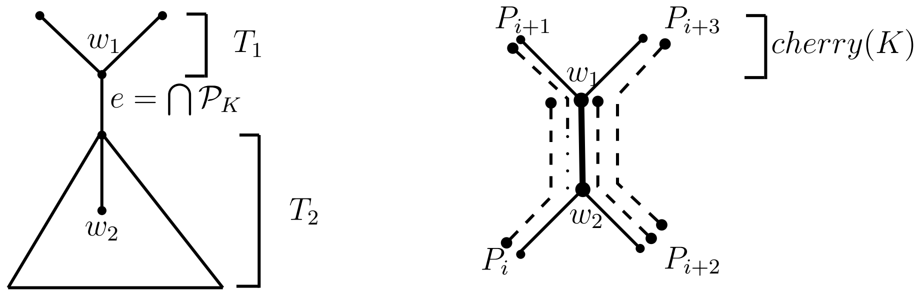

Figure 3.17. A minimal representation of a pair (G, C) with an induced (K_4, P_4) with $N_G(i+1) = K$.

We term, as *isolated*, the vertex $j \in \{i+1, i+2\}$ of $K = (i, i+1, i+2, i+3)$ satisfying $N_G(j) = K$ whose existence and uniqueness are guaranteed by Lemma 3.34 (iv). We recall that $(i, i+1, i+2, i+3) = (i+3, i+2, i+1, i)$, and in view of this result, we introduce an alternative notation: We denote K as $[i, i+1, i+2, i+3]$ if $i+1$ is its isolated vertex, and as $[i+3, i+2, i+1, i]$ otherwise.

Lemma 3.35. *Let $K = [i, i+1, i+2, i+3]$ a (K_4, P_4) of a pair (G, C) with at least 6 vertices, $\langle T, \mathcal{P} \rangle$ a minimal representation of (G, C) satisfying $(P3)$. If there is a path $P_j \notin \mathcal{P}_K$ intersecting $core(K)$, then $j = i-1$ and $|core(K)| = 2$, otherwise $|core(K)| = 1$.*

Proof. Let $\bigcap \mathcal{P}_K = \{e\}$, and assume that $j \notin K$ and $P_j \cap core(K) \neq \emptyset$. Recall that $e \notin P_j$. If P_j splits from $core(K)$ then it splits from each one of P_i, P_{i+2}, P_{i+3}. In particular $\{j, i, i+2\}$ constitutes a red edge-clique, thus violating $(P3)$. If $P_j \subseteq core(K)$ then $P_j \sim P_{i+2}$ implying $j \in \{i+1, i+3\} \subset K$, contradicting our assumption. Therefore P_j crosses the endpoint w_2 of $core(K)$. Then P_j intersects with each one of P_i, P_{i+2}, P_{i+3} in the last edge of $core(K)$. Therefore (i) $P_j \nsim P_{i+2}$ because $j \notin \{i+1, i+3\}$, and (ii) $P_i \nsim P_{i+2}$. If $P_j \nsim P_i$ then $\{j, i, i+2\}$ constitutes a red edge-clique, violating $(P3)$. Therefore $P_j \sim P_i$, implying $j = i-1$. Note that $P_{i+1} \cap P_{i-1} = \emptyset$ because $i+1$ is isolated. $P_{i-1} \cap core(K)$ consists of a single edge $e' (\neq e)$, because otherwise they can be contracted to a single edge without affecting the relationships between the paths $P_{i-1}, P_i, P_{i+2}, P_{i+3}$ that are the only paths that intersect the contracted edges. Then $core(K)$ consists of the two edges e, e'. If P_{i-1} does not intersect $core(K)$ then $\mathcal{P}_K$ are the only paths that intersect $core(K)$. Therefore, all the edges of $core(K)$ can be contracted to one edge. □

Lemma 3.36. *A pair (G, C) with 6 vertices satisfying $(P3)$ does not contain an induced (K_4, P_4).*

Proof. Assume without loss of generality that $[0, 1, 2, 3]$ is a (K_4, P_4) of (G, C). Let $\langle T, \mathcal{P}\rangle$ be a representation of (G, C) satisfying $(P3)$. For $i \in \{0, 3\}$ let T_i be the unique connected component of $T \setminus core(K)$ intersecting P_i. By Lemma 3.35, P_4 does not cross w_2. Therefore P_4 is completely in T_3. As $P_4 \cap P_5 \neq \emptyset$, P_5 intersects T_3. If P_5 is completely in T_3 then $P_5 \parallel P_0$, otherwise $P_5 \nsim P_0$. Both cases contradict the fact that $\{5, 0\}$ is an edge of C. □

3.6.2. Intersection of (K_4, P_4) pairs and Aggressive Contraction

We now focus on pairs with at least 7 vertices. We start by analyzing the intersection of their (K_4, P_4) sub-pairs.

Lemma 3.37. *Let (G, C) be a pair with at least 7 vertices satisfying $(P3)$, and $K = [i, i+1, i+2, i+3]$ a (K_4, P_4) of (G, C). Then*
(i) there is at most one (K_4, P_4), $K' \neq K$ such that $E(C[K]) \cap E(C[K']) \neq \emptyset$ and if such a (K_4, P_4) exists then $K' = [i+5, i+4, i+3, i+2]$ (and therefore $\{i+2, i+4\}$ is an edge of G);
(ii) if $\{i+2, i+4\}$ is an edge of G then $K' = [i+5, i+4, i+3, i+2]$ induces a (K_4, P_4) of (G, C).

Proof. Let without loss of generality $i = 0$.

(i) Since 1 is isolated, $1 \notin K'$. Therefore if $E(C[K]) \cap E(C[K']) \neq \emptyset$ for some (K_4, P_4) K' then $E(C[K]) \cap E(C[K']) = \{\{2, i\}\}$, i.e. $K' = (2, 3, 4, 5)$. As 3 is adjacent to 1, 3 is not isolated in K'. Therefore, $K' = [5, 4, 3, 2]$.

(ii) Assume $\{2, 4\}$ is an edge of G and that, by way of contradiction, $K' =$

$\{2,3,4,5\}$ is not a (K_4, P_4). Consult Figure 3.18 for the following discussion. For $j \in \{0,3\}$ let T_j be the connected component of $T \setminus core(K)$ intersecting P_j. As $P_4 \sim P_3$, Lemma 3.35 implies that P_4 is completely in T_3. $P_4 \nsim P_2$, by our assumption. Let w_3 be the endpoint of P_3 in T_3 and w_4 be the split vertex of P_2 and P_4. Then $w_3 \in p_T(w_2, w_4)$ (possibly $w_3 = w_4$). P_5 does not intersect at least one of P_2 and P_3, because otherwise K' is a (K_4, P_4). Then it does not intersect P_3. The union of the paths $P_6, \ldots P_{n-1}$ constitutes a subtree T' of T that intersects both P_0 and P_5. Therefore there is at least one path $P_j \in \{P_6, \ldots P_{n-1}\}$ crossing the last edge of P_3 (incident to w_3). Then $\{2,4,j\}$ is an edge-clique defined by this edge. Moreover, (i) $P_2 \nsim P_4$, (ii) $P_j \nsim P_2$ because $j \notin \{1,3\}$, $P_j \nsim P_4$ because $j \notin \{3,5\}$. Therefore $\{2,4,j\}$ is a red edge-clique, contradicting the assumption that $(P3)$ is satisfied. □

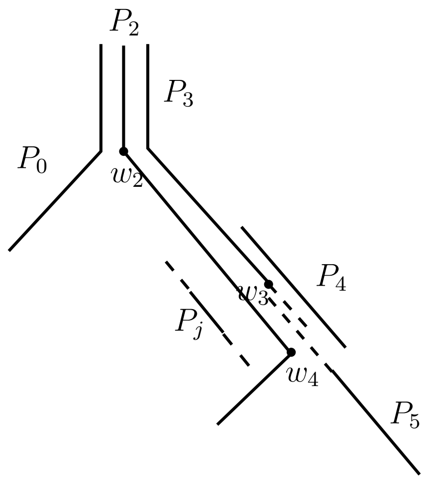

Figure 3.18. Proof of Lemma 3.37.

By the above lemma (K_4, P_4) sub-pairs may intersect only in pairs. We term two intersecting (K_4, P_4) pairs as *twins*, and a (K_4, P_4) not intersecting with another as a *single* (K_4, P_4).

Given a (K_4, P_4) $K = [i, i+1, i+2, i+3]$ of a pair (G'', C'') satisfying $(P3)$, the *aggressive contraction* operation is the replacement of the vertices $i+2, i+3$ by a single vertex $(i+2).(i+3)$. We denote the resulting pair $(G''_{/e}, C''_{/e})$ (where $e = \{i+2, i+3\}$) as $(G'', C'')_{/K}$. The following lemma characterizes the aggressive contraction operation in the representation domain.

Lemma 3.38. *Let (G'', C'') be a pair with at least 7 vertices, $\langle T'', \mathcal{P}'' \rangle$ be a representation of it satisfying $(P3)$, and $K = [i, i+1, i+2, i+3]$ be a (K_4, P_4) of (G'', C''). Then: $(G'', C'')_{/K}$ is a pair satisfying $(P3)$ and a representation $\langle T', \mathcal{P}' \rangle$ of $(G', C') = (G'', C'')_{/K}$ satisfying $(P3)$ is obtained from $\langle T'', \mathcal{P}'' \rangle$ by first removing $cherry(K)$ and also $cherry(K')$ if K and K' are twins, and then applying the union operation to P_{i+2} and P_{i+3}.*

Proof. Let without loss of generality $i = 0$. Recall that by Lemma 3.37, $\{2, 4\}$ is an edge of G'', if and only if K is a twin. Figure 3.19 illustrates the following two steps in the case that K is a single.

(Step 1) We remove $cherry(K)$ (and also $cherry(K')$ when K and K' are twins) from T''. By Lemma 3.34 we know that by removing cherries we don't lose any edge intersection, and we loose exactly one split vertex per cherry, namely the center of the cherry. This vertex (or vertices) is $split(P_1, P_3)$ (and also $split(P_2, P_4)$ when K is a twin). Thus the edge $\{1, 3\}$ (and also $\{2, 4\}$ when K is a twin) becomes blue. As no new red edges are introduced, the resulting representation does not contain red edge-cliques, i.e. satisfies $(P3)$.

(Step 2) We contract the resulting graph on the edge $\{2, 3\}$. We claim that this contraction is defined. Indeed assume by contradiction that $\{2, 3\}$ participates in a BBR triangle. This BBR triangle is one of $\{1, 2, 3\}$ and $\{2, 3, 4\}$. Then one of $\{1, 3\}$ and $\{2, 4\}$ is a red edge, contradicting the fact that these edges (if exist) becomes blue after step 1. This contraction corresponds to the union operation on the paths P_2, P_3, and by Lemma 3.24 the resulting graph satisfies $(P3)$. □

3.6.3. Algorithm

Lemma 3.38 implies an algorithm for finding the unique minimal representation of pairs satisfying $(P3)$. Algorithm FindMinimalRep-P3 is a recursive algorithm that processes a single (K_4, P_4) or a twin of (K_4, P_4)s at every invocation. The processing is done by applying aggressive contraction to convert the involved (K_4, P_4)(s) to (K_3, P_3)

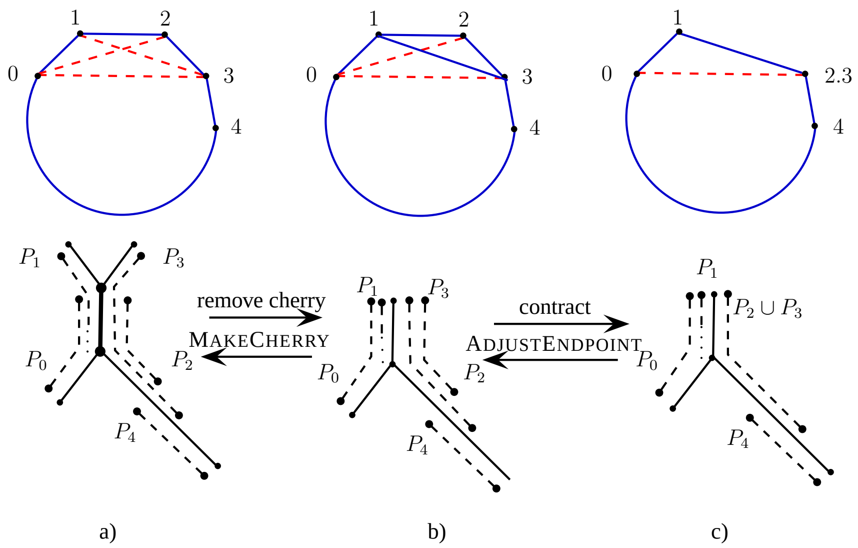

Figure 3.19. Aggressive contraction of a single (K_4, P_4).

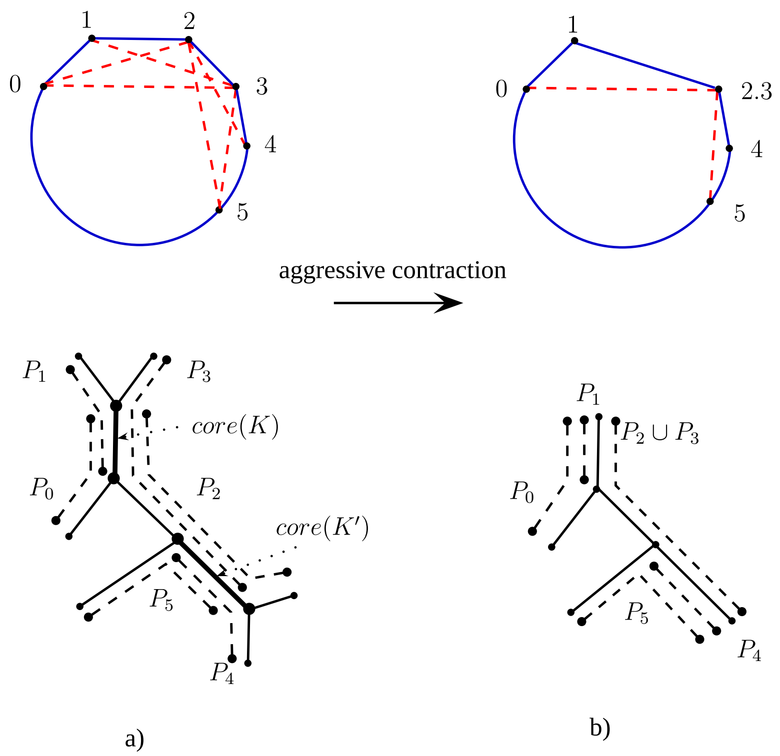

Figure 3.20. Aggressive contraction of twins.

(s), solving the problem recursively, and finally transforming the representation of the (K_3, P_3) to a representation of a (K_4, P_4). In the Build Representation phase, Algorithm FindMinimalRep-P3 performs the reversal of steps 1 and 2 described in Lemma 3.38, (see Figures 3.19, 3.20).

A *broken tour with cherries* is a representation obtained by adding cherries to a broken tour.

Theorem 3.39. P3-HamiltonianPairRec *can be solved in polynomial time. "YES" instances have a unique solution, and whenever $n \geq 6$ this solution is a broken planar tour with cherries.*

Proof. As the case $|V(G'')| < 6$ is already solved, we will show that for any given pair (G'', C'') with $|V(G'')| \geq 6$, FindMinimalRep-P3 solves P3-HamiltonianPairRec. If (G'', C'') is a "NO" instance, then the instance has no representation satisfying $(P3)$. In this case then the algorithm returns "NO" at the validation phase. Therefore we assume that (G'', C'') is a "YES" instance, and prove the claim by induction on the number k of induced (K_4, P_4) pairs of (G'', C'').

If $k = 0$ then (G'', C'') does not contain any (K_4, P_4) pairs, therefore satisfies $(P2)$. In this case the algorithm invokes FindMinRep-P2-P3 and the claim follows from Theorem 3.30.

Otherwise $k > 0$. We assume that the claim holds for any $k' < k$ and prove that it holds for k. In this case, as the pair contains at least one (K_4, P_4), one such pair K is chosen arbitrarily by the algorithm and aggressively contracted. The resulting pair $(G', C') = (G'', C'')_{/K}$ has the following properties:

- Satisfies $(P3)$. (By Lemma 3.38)
- The number of (K_4, P_4) pairs is less then k.
- $|V(G')| \geq 6$. This is because $|V(G')| = |V(G'')| - 1$ and $|V(G'')| > 6$. Indeed, if $|V(G'')| = 6$, we have $k = 0$ by Lemma 3.36.

Require: $C'' = \{0, 1, \ldots, |V(G'')| - 1\}$ is an Hamiltonian cycle of G'' and $|V(G'')| \geq 6$
Ensure: A minimal representation $\langle \bar{T}, \bar{\mathcal{P}} \rangle$ of (G'', C'') satisfying $(P3)$ if any
  if (G'', C'') is (K_4, P_4)-free **then**
    return FindMinRep-P2-P3$(G'', C'', \mathcal{W}(G'', C''))$

  Aggressive Contraction:
  Pick a (K_4, P_4), $K = [i, i+1, i+2, i+3]$ of (G'', C''). ▷ Renumber vertices if necessary.
  $(G', C') \leftarrow (G'', C'')_{/K}$.

  Recurse:
  $\langle \bar{T}', \bar{\mathcal{P}}' \rangle \leftarrow$ FindMinimalRep-P3(G', C').

  Build Representation:
  $\langle \bar{T}, \bar{\mathcal{P}} \rangle \leftarrow \langle \bar{T}', \bar{\mathcal{P}}' \rangle$.
  Replace $P_{(i+2).(i+3)}$ by two copies P_{i+2} and P_{i+3} of itself.
  if $i+2$ is adjacent to $i+4$ in G'' **then**
    ▷ $K' = [i+5, i+4, i+3, i+2]$ is the twin of K in (G'', C'')
    MakeCherry$(\langle \bar{T}, \bar{\mathcal{P}} \rangle, i+4, i+2)$.
  else ▷ K is a single
    $w \leftarrow$ the endpoint of P_{i+2} which is not in $core(K)$.
    AdjustEndpoint$(\langle \bar{T}, \bar{\mathcal{P}} \rangle, G'', P_{i+2}, w)$.
  MakeCherry$(\langle \bar{T}, \bar{\mathcal{P}} \rangle, i+1, i+3)$.

  Validate:
  if Ept$(\bar{\mathcal{P}}) = G''$ and $\bar{\mathcal{P}}$ satisfies $(P3)$ **then**
    return $\langle \bar{T}, \bar{\mathcal{P}} \rangle$
  else
    return "NO"
  function MakeCherry$(\langle \bar{T}, \bar{\mathcal{P}} \rangle, p, q)$
    Let $v \in V(\bar{T})$ be the common endpoint of P_p, P_q.
    Add two new vertices v', v'' and two edges $\{v, v'\}, \{v, v''\}$ to $\bar{T}$.
    Extend P_p so that the endpoint v is moved to v'.
    Extend P_q so that the endpoint v is moved to v''.

Figure 3.21. FindMinimalRep-P3(G'', C'') algorithm.

Therefore, (G', C') satisfies the assumptions of the inductive hypothesis. Then, $\langle \bar{T}', \bar{\mathcal{P}}' \rangle$ is the unique minimal representation of $(G'', C'')_{/K}$ satisfying $(P3)$. It remains to show that the representation $\langle \bar{T}', \bar{\mathcal{P}}' \rangle$ is obtained from the representation $\langle \bar{T}, \bar{\mathcal{P}} \rangle$ returned by the algorithm, by applying the steps described in Lemma 3.38.

Let without loss of generality $K = [i, i+1, i+2, i+3]$. By Lemma 3.37, K has a twin $K' = [i+5, i+4, i+3, i+3]$ if and only if $\{i+2, i+4\}$ is an edge of G''. The algorithm checks the existence of this edge and takes two different actions, accordingly.

If K is not a twin then step 2, i.e. the union operation is reversed by breaking apart the path $P_{(i+2).(i+3)}$ into two paths P_{i+2} and P_{i+3}. Then step 1 is reversed by invoking procedure MakeCherry (see Figure 3.19).

If K is a twin, then *cherry*(K) and *cherry*(K') are uniquely determined by Lemma 3.34 (ii) and procedure MakeCherry acts accordingly. This determines all the endpoints of $P_i, P_{i+1}, P_{i+2}, P_{i+3}, P_{i+4}, P_{i+5}$ that are different from the representation $\langle \bar{T}', \bar{\mathcal{P}}' \rangle$ (see Figure 3.20). $\square$

3.7. General Pairs (G, C)

In this section we show that it is impossible to generalize the algorithms presented in the previous sections to the case where $(P3)$ does not hold, unless P = NP.

We start with a definition and a related lemma that are central to this section. Given a pair (G, G') and a subset S of $V(G)$, the *component graph* *comp*(G, G', S) is a graph whose vertices correspond to the connected components $G_1, G_2, \ldots$ of $G \setminus S$ and two vertices corresponding to components G_i, G_j are connected by an edge if and only if there is a vertex $v \in S$ adjacent to both of G_i and G_j in G' (see Figure 3.23 for an example). Whenever G' is a cycle we term a connected component of $G' \setminus S$ an *arc* of G' separated by S. Clearly, whenever $|S| \geq 2$ every arc is adjacent to exactly 2 vertices of S.

Lemma 3.40. *Let (G, C) be a pair where C is a Hamiltonian cycle of G, and K be a maximal clique of $G \setminus C$. If there is a representation $\langle T, \mathcal{P} \rangle$ of G where $\Delta(T) \leq 3$, then comp(G, C, K) is 3-colorable.*

Proof. If $|K| \leq 3$, $G \setminus K$ has at most 3 connected components, thus *comp*(G, C, K) is 3-colorable. Therefore we assume $|K| > 3$. If K is an edge-clique defined by an edge e then the paths $\mathcal{P}_K = \{P_v : v \in K\}$ are exactly the paths in $\mathcal{P}$ that contain e. The edge e divides T into two subtrees T_1, T_2 rooted at the endpoints r_1, r_2 of e. Similarly, if K is a claw-clique defined by a claw $\{e_1, e_2, e_3\}$, as T has maximum degree 3, the claw divides the tree into three subtrees T_1, T_2, T_3, rooted at the center $r_1 = r_2 = r_3 = r$ of the claw. In both cases the following two statements hold: (i) every path of $\mathcal{P} \setminus \mathcal{P}_K$ is contained in one of these subtrees, (ii) every path of $\mathcal{P}_K$ that intersects a subtree T_i crosses its root r_i.

All the vertices of a connected component G_i are represented by paths that are in the same subtree T_j ($j \in \{1, 2, 3\}$). This is because otherwise there are at least two adjacent vertices in G_i that are in two different subtrees, a contradiction. We color every vertex G_i of *comp*(G, C, K) with color $j \in \{1, 2, 3\}$ depending on the subtree on which the paths representing its vertices reside. It remains to show that if two connected components are adjacent in *comp*(G, C, K) they are colored with different colors.

Assume by contradiction that two components G_1, G_2 of $G \setminus K$ which are adjacent in *comp*(G, C, K) are colored with the same color i. Then, there is a vertex $v \in K$ and two vertices $v_1 \in G_1, v_2 \in G_2$ adjacent to v in C. Moreover v_1 and v_2 are not adjacent in G, because they are in different connected components. Therefore, (i) $P_v \sim P_{v_1}, P_v \sim P_{v_2}$, (ii) $P_{v_1} \parallel P_{v_2}$, (iii) P_{v_1} and P_{v_2} are in T_i, (iv) P_v intersects T_i and crosses its root r_i. Furthermore, we assume without loss of generality that P_{v_1} is closer to r_i than P_{v_2} (see Figure 3.22). Consider the subtree $T' = \cup_{u \in G_2} P_u$ of T_i. $P_{v_1} \cap T' = \emptyset$, because otherwise there is a path P_u representing a vertex $u \in G_2$ that intersects P_{v_1}, in other words $u \in G_2$ is adjacent to $v_1 \in G_1$, a contradiction. Let

$\{v, v'\}$ be the vertices of K adjacent to the arc v_2 belongs to. $P_{v'}$ intersects T_i and crosses its root r_i. Moreover, $P_{v'}$ intersects T', as it is adjacent to at least one vertex of G_2. We conclude that $P_{v'}$ contains P_{v_1}. Then $v_1 \sim v'$, i.e. v' and v_1 are adjacent in C. Therefore $K = \{v, v'\}$, contradicting $|K| > 3$. $\square$

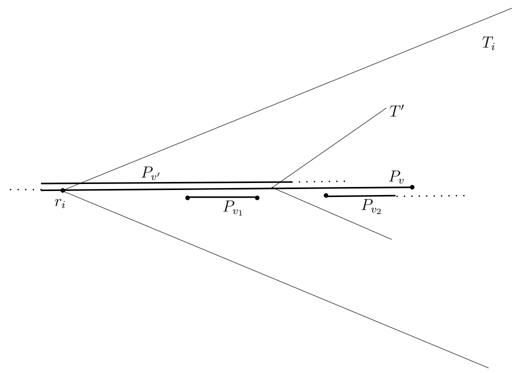

Figure 3.22. Proof of Lemma 3.40.

Lemma 3.41. *It is* NP*-hard to determine whether a given pair* (G, C) *where* C *is a Hamiltonian cycle of* G *has representation* $\langle T, \mathcal{P} \rangle$ *with* $\Delta(T) \leq 3$.

Proof. The proof is by reduction from the 3-colorability problem. Given a graph H, we transform it to a pair (G, C) such that (G, C) has a representation on a tree with maximum degree 3 if and only if H is 3-colorable.

Consult Figure 3.23 for the following construction. Let $V(H) = \{v_0, \ldots, v_{n-1}\}$, $E(H) = \{e_0, \ldots, e_{m-1}\}$, and let $d_i = d_H(v_i)$. The pair (G, C) consists of $6m$ vertices. For every edge $e_k = \{v_i, v_j\}$ we build a path $S_k = (u_{i,k} - u'_{i,k} - u_{j,k} - u'_{j,k} - u_k - u'_k)$ with 6 vertices. The graph C is a cycle obtained by concatenating these m paths, in the order $S_0, S_1, \ldots, S_{m-1}, S_0$, i.e. u'_k is connected to $u_{i',k+1}$ where $e_{k+1} = \{v_{i'}, v_{j'}\}$. K is a clique of all the vertices in the even positions of the paths, i.e. $K = \left\{u'_{i,k}, u'_k : 0 < k < m, i \in e_k\right\}$ (most of the edges induced by K are not shown in the figure). For every $i < n$, Q_i is a path $(u_{i,k_1} - \cdots - u_{i,k_{d_i}})$ where $e_{k_1}, \ldots, e_{k_{d_i}}$ are the edges incident to v_i in H. The set E_i^{KQ} of edges connects vertices of Q_i with vertices of K. Specifically, $E_i^{KQ} = \left\{\left\{u_{i,k_j}, u'_{i,k_{j'}}\right\} \mid 1 \leq j' < j \leq d_i\right\}$. Finally, $G = C \cup K \cup (\cup_{i<n} Q_i) \cup \left(\cup_{i<n} E_i^{KQ}\right)$.

We claim that the vertices of the graph $H' = comp(G, C, K)$ can be partitioned into two sets A, B such that (i) $H'[A]$ is isomorphic to H, (ii) $H'[B]$ is an independent set, (iii) $d_{H'}(v) \leq 2$ for every vertex $v \in B$. Indeed, $G \setminus K$ contains the vertices $\{u_{i,k}, u_k : k < m, i \in e_k\}$ where each u_k is an isolated vertex and the rest is the disjoint union of the paths Q_i. Therefore the component graph H' consists of the vertices $A = \{Q_i : i < n\}$ and $B = \{u_k : k < m\}$. For two vertices v_i, v_j of H, Q_i and Q_j are connected by the vertex $u'_{i,k} \in V(K)$ if and only if $e_k = \{v_i, v_j\}$ is an edge of H. Therefore $H'[A]$ is isomorphic to H. Moreover, a vertex u_k of G is connected to at most two paths Q_i via its two neighbors in C. Therefore $H'[B]$ is an independent set and every vertex of B has degree at most 2 in H'. We conclude that H is 3-colorable if and only if H' is 3-colorable. If (G, C) has a representation $\langle T, \mathcal{P} \rangle$ with $\Delta(T) \leq 3$ then, by Lemma 3.40, H' is 3-colorable. It remains to show that if H' is 3-colorable then (G, C) has such a representation. Given a 3-coloring of H', in the sequel we present such a representation $\langle T, \mathcal{P} \rangle$ (see Figure 3.24).

We start with the construction of the tree T. T has a vertex r of degree at most 3 that divides it into at most 3 subtrees T_1, T_2, T_3, each of which with maximum degree 3. Each T_i corresponds to one color of the given 3-coloring of H'. We describe in detail the subtree T_1, assuming without loss of generality that the vertices of H' colored with color 1 are $Q_1, Q_2, \ldots, Q_{n'}$ and $u_1, u_2, \ldots, u_{m'}$. T_1 contains a path $(r - e_1 - \cdots - e_{m'} - v_1 - \cdots - v_{n'})$. Each vertex e_k starts a path $(e_k - \ell_k)$ of length 1. Each vertex v_i starts a path $(v_i - w_i - w_{i,k_1} - \cdots - w_{i,k_{d_i}})$ where $e_{k_1}, \ldots, e_{k_{d_i}}$ are the edges incident to v_i in G. Each vertex $w_{i,k}$ starts a path $(w_{i,k} - \ell_{i,k})$ of length 1.

We proceed with the construction of the paths $\mathcal{P}$. Every vertex u_k of G is represented by a path P_k of length 1 starting at vertex ℓ_k. Each vertex $u_{i,k}$ of G is represented by a path $P_{i,k}$ of length 3 starting at $\ell_{i,k}$ and towards r. It remains to describe the representation of the vertices of K. Every vertex u' of K is adjacent to two vertices of $V(C) \setminus K$ in C. We represent u' by a path between two leaves of T (not all of them shown in the figure). These leaves are exactly the leaves that constitute endpoints of the paths corresponding to the two neighbors of u'. Specifically:

- A vertex $u'_{i,k}$ of S_k that is between two vertices $u_{i,k}$ and $u_{j,k}$ of S_k is represented by a path $P'_{i,k}$ between the two leaves $\ell_{i,k}$ and $\ell_{j,k}$.
- A vertex $u'_{j,k}$ of S_k that is between two vertices $u_{j,k}$ and u_k of S_k is represented by a path $P'_{j,k}$ between the two leaves $\ell_{j,k}$ and ℓ_k.
- A vertex u'_k of S_k that is between two vertices u_k of S_k and $u_{i,k+1}$ of S_{k+1} is represented by a path P'_k between the two leaves ℓ_k and $\ell_{i,k+1}$.

The vertices $u_{i,k}$ and $u_{j,k}$ are in the connected components Q_i and Q_j respectively, which in turn are adjacent in H' (by the existence of $u'_{i,k} \in K$ between them). They are therefore assigned different colors, i.e. the leaves $\ell_{i,k}$ and $\ell_{j,k}$ are in different subtrees of T. Therefore $P'_{i,k}$ crosses r. It can be verified that this holds for the other two cases too. We conclude that the vertices of K are represented by paths that cross r. If H' is 2-colorable then they constitute an edge-clique, otherwise they constitute a claw-clique. We leave to the reader to verify that $\langle T, \mathcal{P} \rangle$ is a representation of (G, C). $\square$

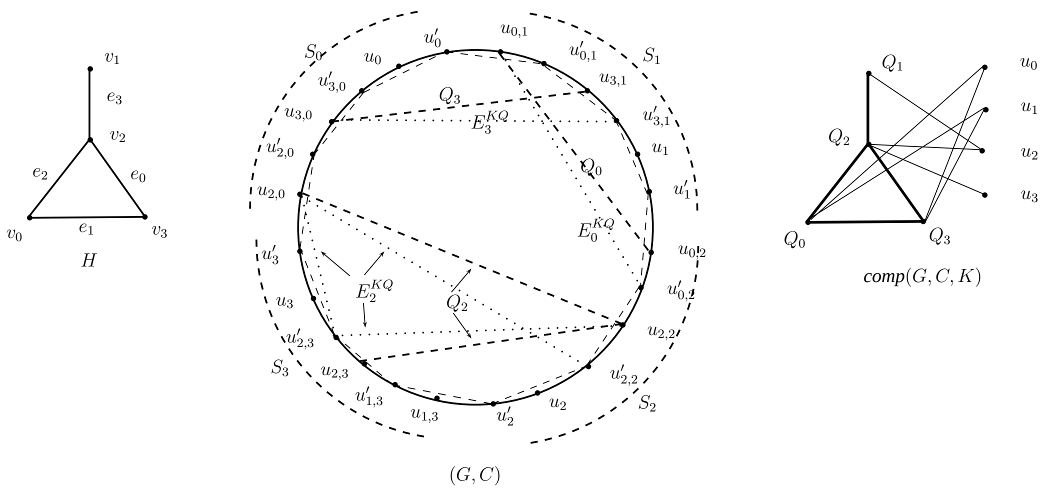

Figure 3.23. A graph H, the corresponding pair (G, C) and the component graph *comp*(G, C, K) where $K = \left\{u'_{i,k}, u'_k : 0 < k < m, i \in e_k\right\}$.

Theorem 3.42. HAMILTONIANPAIRREC *is* NP*-hard.*

Proof. We claim that the decision version of the problem is NP-hard even when G is restricted to the family of VPT graphs. If the instance is a "YES" instance, then G is both a VPT and an EPT graph. In this case, by Theorem 2 of [6], (G, C) has a representation on a tree with maximum degree 3. If the instance is a "NO" instance

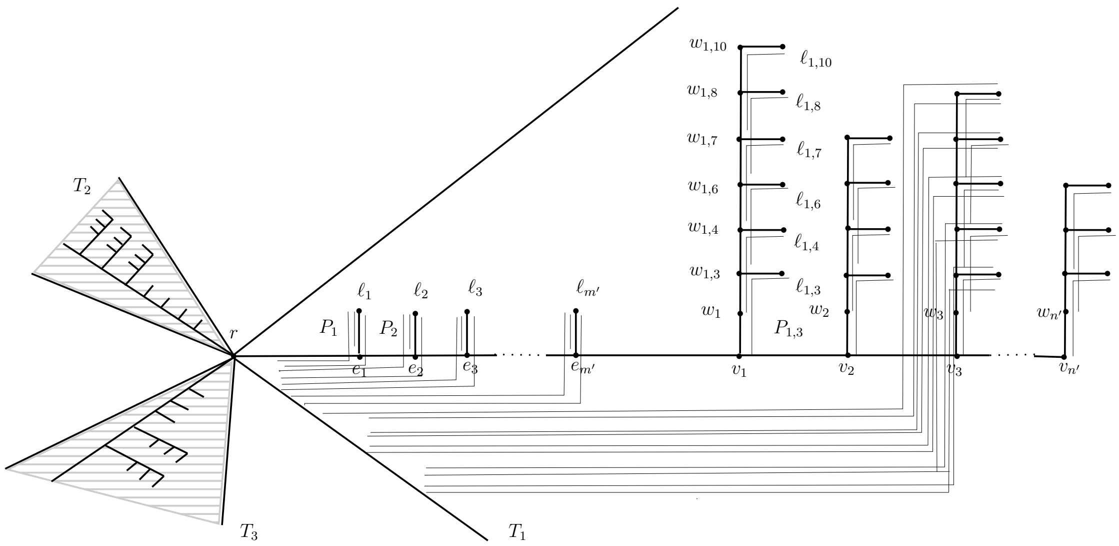

Figure 3.24. A representation $\langle T, \mathcal{P} \rangle$ of a pair (G, C) corresponding to some 3-colorable graph H.

then, clearly, (G, C) does not have a representation on a tree with maximum degree 3. By Lemma 3.41 it is NP-hard to decide whether (G, C) has a representation on a tree with maximum degree 3. $\square$

4. GRAPHS OF EDGE-INTERSECTING NON-SPLITTING PATHS IN A GRID

4.1. Overview

The family of paths on graphs is a commonly studied family of sets. To distinguish the graph on which the paths are defined, from the resulting intersection graph, this graph is called the *host* graph. Often the host graphs are restricted to certain families such as paths, cycles, trees, grids, etc. When H is restricted to paths and cycles we get the well known families of interval graphs [24] and circular arc graphs [25], respectively. When H is restricted to trees, we obtain the family of Edge Intersection Graph of Paths in a tree (EPT) [22], and when H is a grid, the corresponding graph is called an EPG graph [13].

In the previous chapter we assumed that the host graph was a tree, now we generalize it to any graph. This chapter is based on our recent publication [26]. In Section 4.2 we start with necessary definitions and notations. In Section 4.3 we show in Theorem 4.5 that cobipartite graphs are not included in ENP. This result implies that although the Edge Intersection Graphs of Paths in an arbitrary graph includes all graphs, this is not the case for ENP. By a counting argument, we show that not all cobipartite graphs are ENP. The main observation is that the ENP representations of cliques are the collections of paths whose union is a path or a cycle. Therefore in the representation of a cobipartite graph we consider the intersections, called segments, of two paths (or cycles). The number of possible graphs is a function of the number of segments and the number of endpoints in these segments. An analysis shows that this number is less than the number of possible cobipartite graphs.

In the same section, Theorem 4.6 shows that the class ENP coincides with the family of graphs of Edge-Intersecting and Non-Splitting Paths in a Grid (ENPG). Given an arbitrary representation, we first transform the host graph into a planar graph. We then replace every vertex of the host graph having degree more than 4 and the paths crossing this vertex with a special gadget, see Figure 4.2. Finally using a known result we embed this planar graph in a grid.

In a grid, a *bend* of a path is a pair of consecutive edges of the path one of which is vertical and the other is horizontal. Following similar studies for EPG graph class, we study in Section 4.4 the implications of restricting the number of bends of the individual paths in the grid. It is shown in [27] that for every odd integer k, B_k-EPG $\subsetneq$ B_{k+1}-EPG, i.e. the bend numbers imply an infinite hierarchy within the family of EPG graphs. We showed in Theorem 4.13 that there is an infinite sequence of integers $\{k_i : i = 1, 2, \ldots\}$ such that B_0-ENPG $\subsetneq$ B_1-ENPG $\subsetneq$ B_{k_1}-ENPG $\subsetneq$ B_{k_2}-ENPG $\subsetneq \cdots$. Later in Chapter 5 we show that B_1-ENPG $\subsetneq$ B_2-ENPG however the question whether B_2-ENPG $\subsetneq$ B_3-ENPG $\subsetneq \cdots$ is still open.

4.2. Definitions and Notations

A *walk* in a graph $G = (V(G), E(G))$ is a sequence $P = (e_1, e_2, \ldots, e_\ell)$ of edges of $E(G)$ such that there are vertices $v_0, v_1, \ldots, v_\ell$ satisfying $e_i = \{v_{i-1}, v_i\}$ for every $i \in [\ell]$. Clearly, the reverse sequence $(e_\ell, \ldots, e_1)$ is also a walk. The *length* of P is the number ℓ of (not necessarily distinct) edges in the sequence. In this work we do not consider *trivial* (zero length) walks, as such walks do not intersect others. P is *closed* whenever $v_0 = v_\ell$, and *open* otherwise. A *trail* is a walk consisting of distinct edges. A *(simple) path* is a walk consisting of distinct vertices except possibly $v_0 = v_\ell$. A contiguous sub-sequence of a walk (resp. trail, path) is termed a sub-walk (resp. sub-trail, sub-path).

Let $P = (e_1, e_2, \ldots, e_\ell)$ be a trail with vertices $v_0, v_1, \ldots, v_\ell$ as above. For every $i \in [\ell - 1]$, the triple (e_i, v_i, e_{i+1}) is an *internal point* of P. Whenever P is closed, the triple $(e_\ell, v_\ell = v_0, e_1)$ too, is an internal point of P. We denote the set of internal points of P by $INT(P)$. We say that a vertex v is an internal vertex of P, or equivalently that P crosses v if v is in (i.e. is the second entry of) a triple in $INT(P)$. If P is open $END(P) \stackrel{def}{=} \{v_0, v_\ell\}$ and $TAIL(P) \stackrel{def}{=} \{(e_1, v_0), (e_\ell, v_\ell)\}$ are the sets of *endpoints* of P and tails of P, respectively. Given a set $\mathcal{P}$ of trails, we define $\mathit{TAIL}(\mathcal{P}) \stackrel{def}{=} \cup_{P \in \mathcal{P}} \mathit{TAIL}(P)$, $\mathit{END}(\mathcal{P}) \stackrel{def}{=} \cup_{P \in \mathcal{P}} \mathit{END}(P)$ and $\mathit{INT}(\mathcal{P}) \stackrel{def}{=} \cup_{P \in \mathcal{P}} \mathit{INT}(P)$. For brevity, in the text we often refer to internal points as vertices and to tails as edges. Moreover, when we apply the intersection and union operations on two trails we consider them as sets of internal points and endpoints.

Given two trails $P = (e_1, e_2, \ldots, e_\ell)$ and $P' = (e'_1, e'_2, \ldots, e'_{\ell'})$, a *segment* of $P \cap P'$ is a maximal trail that constitutes a sub-trail of both P and P'. Since P and P' are trails, $P \cap P'$ is the union of edge disjoint segments. We denote this set by $\mathcal{S}(P, P')$. A tail (resp. endpoint) of a segment is *terminating* if it is in $\mathit{TAIL}(P, P')$ (resp. $\mathit{END}(P, P')$). A *split* of P and P' is a pair of internal points $(e_i, v_i, e_{i+1}), (e'_j, v'_j, e'_{j+1}) \in \mathit{INT}(P) \times \mathit{INT}(P')$ such that $v_i = v'_j$ and $\left|\{e_i, e_{i+1}\} \cap \{e'_j, e'_{j+1}\}\right| = 1$. Note that the common edge and the common vertex constitute a non-terminating tail of a segment of $P \cap P'$ and conversely every non-terminating tail of a segment corresponds to a split. We denote by $\mathit{split}(P, P')$ the set of all splits of P and P', which corresponds to the set of all non-terminating tails of the segments $\mathcal{S}(P, P')$.

Lemma 4.1. *Let K be a clique of an* ENP *graph. Then one of the following holds:*

- *(i) $\cup \mathcal{P}_K$ is an open trail and $\cap P_K \neq \emptyset$.*
- *(ii) $\cup \mathcal{P}_K$ is a closed trail, and for every edge e of $\cup \mathcal{P}_K$ there exists an edge e' of $\cup \mathcal{P}_K$ such that $P \cap \{e, e'\} \neq \emptyset$ for every path $P \in \mathcal{P}_K$.*

Proof. Assume that $\cup \mathcal{P}_K$ contains two internal points (e_1, v, e_2) and (e'_1, v, e'_2) such that $|\{e_1, e_2\} \cap \{e'_1, e'_2\}| = 1$, then there are two paths $P, P' \in \mathcal{P}_K$ such that $(e_1, v, e_2) \in \mathit{INT}(P)$ and $(e'_1, v, e'_2) \in \mathit{INT}(P')$. Therefore, $\mathit{split}(P, P') \neq \emptyset$ and $P \nsim P'$ contradicting the fact that K is a clique. Therefore, $\cup \mathcal{P}_K$ is a disjoint union of trails. However, if $\cup \mathcal{P}_K$ contains two disjoint trails, then $P \parallel P'$ for any two paths P, P' from two distinct trails of $\cup \mathcal{P}_K$, contradicting the fact that K is a clique. Therefore $\cup \mathcal{P}_K$ is one trail.

- (i) If $\cup \mathcal{P}_K$ is an open trail, then we can embed it on the real line, so that the individual paths of $\mathcal{P}_K$ are intervals on the real line. Then, the result follows from the Helly property of intervals.

(ii) If $\cup\mathcal{P}_K$ is an closed trail, let e be any edge of this trail. Let $\mathcal{P}_e$ be the set of trails in $\mathcal{P}_K$ containing the edge e. Then $\cup(\mathcal{P}_K \setminus \mathcal{P}_e)$ is an open trail. By the previous result there is an edge e' of this trail that is contained of all these paths. Therefore, all the paths of $\mathcal{P}_K$ contain either e or e'.

□

Based on this lemma we say that K is an *open* (resp. *closed*) clique if $\cup\mathcal{P}_K$ is an open (resp. closed) trail. It will be convenient to use the following corollary of Lemma 4.1 in order to unify the two cases into one.

Corollary 4.2. *Let K be a clique of an* ENP *graph, with a representation $\langle H, \mathcal{P}\rangle$. Then $\cup\mathcal{P}_K$ is a sub-trail of a closed trail in which for every edge e there exists an edge e' such that $P \cap \{e, e'\} \neq \emptyset$ for every $P \in \mathcal{P}_K$.*

We denote a closed trail whose existence is guaranteed by Corollary 4.2 as $\mathcal{P}^{(K)}$. Note that $\mathcal{P}^{(K)}$ consists of at most one edge more than $\cup\mathcal{P}_K$.

4.3. ENP

In this section we show that (i) the family of ENP graphs does not include all co-bipartite graphs (Theorem 4.5), and (ii) the family of ENP graphs coincides with the family of ENPG graphs (Theorem 4.6).

We proceed with definitions regarding the relationship between the representations of two cliques. Given two vertex disjoint cliques K, K' of an ENP graph G with a representation $\langle H, \mathcal{P}\rangle$, we denote $\mathcal{S}(K, K') \overset{def}{=} \mathcal{S}(\mathcal{P}^{(K)}, \mathcal{P}^{(K')})$. A segment $S \in \mathcal{S}(K, K')$ is *quiet in* K if it does not contain tails of paths of $\mathcal{P}_K$, and *busy in* K, otherwise. The importance of segments stems from the following observation:

Observation 4.3. *Consider a pair of trails $(P, P') \in \mathcal{P}_K \times \mathcal{P}_{K'}$. Then,* *(i)* $P \cap P' \subseteq \cup\mathcal{S}(K, K')$, *and*

(ii) *split*(P, P') *corresponds to the set of all non-terminating segment endpoints crossed by both* P *and* P'.

$\mathcal{C}_{n,n}$ is the set of all co-bipartite graphs $G(K, K', E)$ where $K = [n]$ and $K' = \{i' : i \in [n]\}$. We first prove the following lemma that bounds the number of graphs of this form as a function of the number of segments.

Lemma 4.4. *For any* $s \geq 0$, *the number of graphs* $G = (K, K', E) \in \mathcal{C}_{n,n}$ *with a representation* $\langle H, \mathcal{P}\rangle$ *such that* $|\mathcal{S}(K, K')| \leq s$ *is at most* $(4n)!((2n + 2s)!)^2$.

Proof. Let $G \in \mathcal{C}_{n,n} \cap$ ENP, with a representation $\langle H, \mathcal{P}\rangle$. As K and K' are cliques, their representations satisfy Corollary 4.2, i.e. $\cup\mathcal{P}_K$ and $\cup\mathcal{P}_{K'}$ are sub-trails of two closed trails $\mathcal{P}^{(K)}, \mathcal{P}^{(K')}$. We now consider all the possible orders of *END*$(\mathcal{P}_K) \cup$ *END*$(\mathcal{P}_{K'}) \cup$ *END*$(\mathcal{S}(K, K'))$ on $\mathcal{P}^{(K)}$ and $\mathcal{P}^{(K')}$. This is only an upper bound on the number of possible representations, thus to the number of graphs. This is because some of the orders do not induce a representation of the cliques K and K', and some others may imply two intersecting segments.

Let $s = |\mathcal{S}(K, K')|$, and consider the set Π_K of all the cyclic orders on the closed trail $\mathcal{P}^{(K)}$ of the at most $2n$ endpoints *END*$(\mathcal{P}_K)$ and the at most $2s$ endpoints *END*$(\mathcal{S}(K, K'))$. $|\Pi_K| \leq 2(2n + 2s - 1)!/(2s)!$, because the $2s$ endpoints are identical except for a circular shift by one position (that cause segments to become non-segments and vice versa). For any order $\pi \in \Pi_K$ we consider the set $\Pi_{K'}(\pi)$ of all the orders, on the closed trail $\mathcal{P}^{(K')}$, of the at most $2n$ endpoints *END*$(\mathcal{P}_{K'})$ and the at most $2s$ endpoints *END*$(\mathcal{S}(K, K'))$. This time the segments are umbered according to the order π and are therefore considered as distinct. Clearly, $|\Pi_{K'}(\pi)| \leq (2n + 2s - 1)!$. Summarizing, there are at most $2((2n+2s-1)!)^2/(2s)!$ possible orders, not considering the different orders of vertices of *END*$(\mathcal{P}_K)$ and *END*$(\mathcal{P}_{K'})$ within the same segment. We fix an order $\pi' \in \Pi_{K'}(\pi)$. Let $k(S)$ (resp. $k'(S)$) be the number of endpoints of *END*$(\mathcal{P}_K)$ (resp. *END*$(\mathcal{P}_{K'})$) within segment S, i.e. $k(S) = |$*END*$(\mathcal{P}_K) \cap V(S)|$ and $k'(S) = |$*END*$(\mathcal{P}_{K'}) \cap V(S)|$. The $k(S) + k'(S)$ endpoints can be ordered within

S in $\begin{pmatrix} k(S)+k'(S) \\ k(S) \end{pmatrix}$ different ways because the order of the vertices within each set is fixed. We have $\prod_{S\in\mathcal{S}(K,K')}\begin{pmatrix} k(S)+k'(S) \\ k(S) \end{pmatrix} < \prod_{S\in\mathcal{S}(K,K')}(k(S)+k'(S))! < \left(\sum_{S\in\mathcal{S}(K,K')}(k(S)+k'(S))\right)! = (4n)!$ orders. Therefore, the total number of possible orders of the $4n+2s$ endpoints is at most $2(4n)!((2n+2s-1)!)^2/(2s)! < (4n)!((2n+2s)!)^2$. □

Theorem 4.5. Co-Bipartite $\not\subseteq$ ENP.

Proof. $|\mathcal{C}_{n,n}| = 2^{n^2}$ because there are n^2 pairs of vertices $(v,v') \in K \times K'$, and for every such pair, either $(v,v') \in E$ or $(v,v') \notin E$. In the rest of the proof we show that every $G \in \mathcal{C}_{n,n}$ has a representation $\langle H', \mathcal{P}' \rangle$ for which $s = |\mathcal{S}(K,K')| \leq 12n$. By Lemma 4.4, the number of such representations and therefore $|\mathcal{C}_{n,n} \cap \text{ENP}|$ is at most $(4n)!((2n+2s)!)^2 \leq (4n)!((2n+24n)!)^2 = (4n)!(26n)!(26n)!$. Therefore, $\log|\mathcal{C}_{n,n} \cap \text{ENP}| = O(n\log n)$, whereas $\log|\mathcal{C}_{n,n}| = n^2$ concluding the proof. It remains to show that G has a representation with $s \leq 12n$ segments.

The number of busy segments of $\mathcal{S}(K,K')$ is at most $4n$, because $|END(\mathcal{P}_K)| = 2n$ and an endpoint can be in at most 2 segments. We now bound the number of quiet segments of $\mathcal{S}(K,K')$. Consider two endpoints from $END(\mathcal{P}_K)$ that are consecutive on $\mathcal{P}^{(K)}$ and let P be the sub-trail of $\mathcal{P}^{(K)}$ between these two endpoints. By this choice, every trail of $\mathcal{P}_K$ intersecting P includes P. Let $\bar{\mathcal{S}}$ be the set of segments S that are sub-trails of P (thus $V(S) \subseteq INT(P)$). Suppose that $\left|\bar{\mathcal{S}}\right| > 4$. Consider the two edges $e_{a'}$ and $e_{b'}$ of $\mathcal{P}^{(K')}$ whose existence are guaranteed by Corollary 4.2. These two edges divide $\mathcal{P}^{(K')}$ into at most two open trails. One of these open trails contains (at least) 3 segments $S_1, S_2, S_3 \in \bar{\mathcal{S}}$ where the indices are in the order they appear on this open trail from $e_{a'}$ to $e_{b'}$ (see Figure 4.1). Let also v_{i1}, v_{i2} be the endpoints of S_i in the same order. We claim that the representation obtained by adding to H a new vertex x and two edges $\{v_{21}, x\}, \{x, v_{22}\}$ and finally modifying all the trails intersecting P (that therefore include S_2) so that the segment S_2 is replaced by the trail $(\{v_{21}, x\}, \{x, v_{22}\})$ is an equivalent representation. Clearly, any trail that does not intersect S_2 is not affected

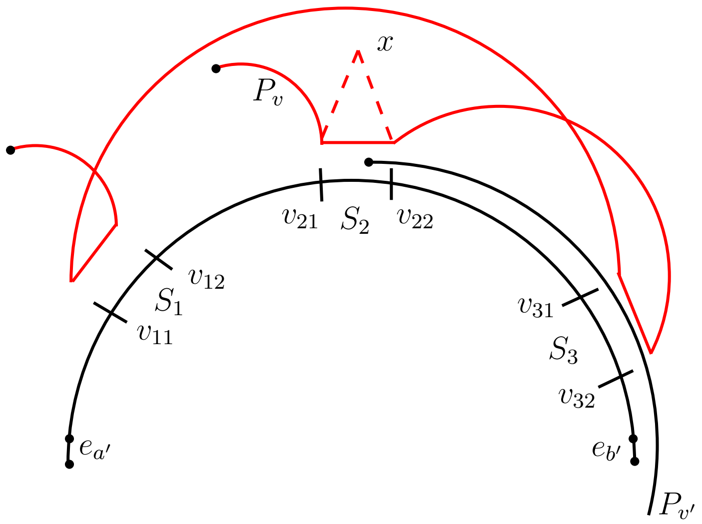

Figure 4.1. Getting a representation with at most $8n$ quiet segments in the proof of Theorem 4.5. Whenever there are 3 segments on one side of the closed trail, the middle one can be bypassed.

by this modification. Consider two trails P_v and $P_{v'}$ such that $(v, v') \in K \times K'$ and both intersect S_2. P_v includes P and therefore includes all the vertices of S_2, in particular crosses v_{21} and v_{22}. On the other hand, by Corollary 4.2, $P_{v'}$ contains at least one of $e_{a'}$ and $e_{b'}$. Without loss of generality let $e_{b'} \in P_{v'}$. Then, v_{22} is an internal vertex of $P_{v'}$. We conclude that $v_{22} \in \mathit{split}(P_v, P_{v'})$, i.e. $(v, v') \notin E(G)$. After the modification, we have $v_{31} \in \mathit{split}(P_v, P_{v'})$, thus (v, v') is not an edge of the resulting graph. Therefore, the new representation is equivalent to $\langle H, \mathcal{P} \rangle$. After this modification, S is not a segment of $\mathcal{S}(K, K')$ and the new representation has one segment less. We can apply this transformation until we get an equivalent representation $\langle H', \mathcal{P}' \rangle$ having at most 4 quiet segments between every two consecutive vertices of $\mathit{END}(\mathcal{P}_K)$. In other words, $\langle H', \mathcal{P}' \rangle$ has at most $8n$ quiet segments of $\mathcal{S}(K, K')$. Adding the at most $4n$ busy segments, we conclude that $s \leq 12n$. $\square$

Theorem 4.6. ENP=ENPG

Proof. Clearly, ENPG $\subseteq$ ENP. To prove the other direction, consider an ENP graph G with a representation $\langle H, \mathcal{P} \rangle$. We transform this representation into an equivalent ENPG representation, in three steps. In the first step, we obtain an equivalent representation $\langle H', \mathcal{P}' \rangle$ where H' is planar. In the second step, we transform $\langle H', \mathcal{P}' \rangle$ to an equivalent representation $\langle H'', \mathcal{P}'' \rangle$ where H'' is planar and $\Delta(H'') \leq 4$. Finally, we transform $\langle H'', \mathcal{P}'' \rangle$ to an ENPG representation.

The host graph H can be embedded in a plane such that the vertices are mapped to a set of points in general position on the plane and the edges are drawn as straight line segments. Specifically, no three points are co-linear and no three segments intersect at one point. Note that the mapping of the edges might intersect, however as the points are in general position, we can assume that no three edges intersect at the same point. For every intersection point of two edges e, e', we can add a vertex v to H and subdivide the edges e and e' (and consequently the paths in $\mathcal{P}$ containing e and e') such that the resulting 4 edges are incident to v. Every pair of paths P, P' that include e and e' respectively now intersect at v. However as we are not concerned with vertex intersections, the resulting representation is a representation of G. We continue in this way until all intersection points are replaced by a vertex. The graph H' of the resulting representation $\langle H', \mathcal{P}' \rangle$ is clearly planar.

We now transform the representation $\langle H', \mathcal{P}' \rangle$ to a representation $\langle H'', \mathcal{P}'' \rangle$ where H'' is planar with maximum degree at most 4. We start with $\langle H'', \mathcal{P}'' \rangle = \langle H', \mathcal{P}' \rangle$, and as long as there is a vertex v with $d_{H''}(v) > 4$, we eliminate such a vertex without introducing new vertices of degree more than 4 using the following procedure described in Figure 4.2: we number the edges incident to v as $e_1, e_2, \ldots, e_{d_v}$ in counterclockwise order according to the planar embedding of H'. Then e_1 and e_{d_v} are in the same face F of H''. We replace the vertex v with a path of d_v vertices $v_1, v_2, \ldots, v_{d_v}$ such that each edges e_i is incident to v_i. Clearly, the constructed path is part of F. We now construct the gadget in Figure 4.2 within the face F, where every path crossing v from an edge e_i to another edge e_j with $i < j$ is modified as described in the figure. Clearly, we do not lose intersections in this process. On the other hand, every pair of paths that intersect within the gadget have at least one edge incident to v in common before the transformation. Moreover, two paths have a split vertex within the gadget if and only if they split at v before the transformation.

The last step is implied by the following theorem.

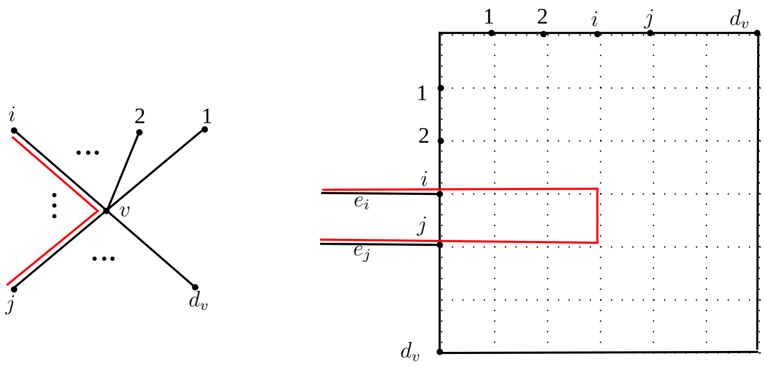

Figure 4.2. The gadget used in the second transformation in the proof of Theorem 4.6.

Theorem 2.3 [28]: A planar graph H'' with maximum degree at most 4 can be embedded in a grid graph H''' of polynomial size: the vertices u'' of H'' are mapped to vertices u''' of H'''; each edge $e'' = \{u'', v''\}$ of H'' is mapped to a path e''' between u''' and v''' in H'''; the intermediate vertices of e''' belong to exactly one such path. Given an embedding of H'' guaranteed by the theorem, we embed every trail $P'' \in \mathcal{P}''$ to a trail P''' of H''' by embedding every edge e'' of it to the corresponding path e''' of H'''. P''' is clearly a walk. P''' a trail, because otherwise there is an edge of H''' that is contained in the embedding of two distinct edges of H'', contradicting the last guarantee of the theorem. Clearly two trails P''_1, P''_2 of $\mathcal{P}''$ intersect if and only if the corresponding paths P'''_1, P'''_2 in $\mathcal{P}'''$ intersect. Moreover a split $(e''_{11}, v'', e''_{12}), (e''_{21}, v'', e''_{22})$ of two paths P''_1, P''_2 is mapped to a split $(e'''_{11}, v''', e'''_{12}), (e''_{21}, v''', e''_{22})$ of the corresponding paths P'''_1, P'''_2 and this mapping is one to one. $\square$

4.4. B$_k$-ENPG

An ENPG graph is B$_k$-ENPG if it has an ENPG representation $\langle H, \mathcal{P} \rangle$, in which every path $P \in \mathcal{P}$ has at most k bends. By definition, B$_k$-ENPG $\subseteq$ B$_{k'}$-ENPG whenever $k \leq k'$. However, the question whether B$_k$-ENPG $\subsetneq$ B$_{k'}$-ENPG holds is not trivial and is the subject of this section. We show in Theorem 4.13 that for some infinite and increasing sequence of numbers $k_1, k_2, \ldots$ there is a graph in B$_{k_{i+1}}$-ENPG that is not B$_{k_i}$-ENPG, thus proving the existence of an infinite hierarchy within the family of ENPG graphs.

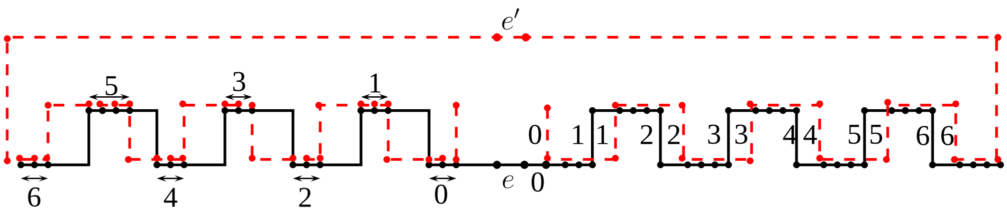

Figure 4.3. A $\mathrm{B}_{(6x+1)}$-ENPG representation of the graph $\mathrm{PM}_{(6x^2+5x-3)}$ for $x = 3$. The solid and dotted lines represent the union of the paths corresponding to two cliques. The individual paths are intentionally omitted but described in detail in Figure 4.4.

The graph $\mathrm{PM}_n \in \mathcal{C}_{n,n}$ is the co-bipartite graph (K, K', E) with $|K| = |K'| = n$ and E constitutes a perfect matching. We denote by $\hat{n}_k$ the biggest number n such $\mathrm{PM}_n \in \mathrm{B}_k$-ENPG. In Corollary 4.8 and Lemma 4.12 we present lower and upper bounds for $\hat{n}_k$, respectively. Using these bounds we show in Theorem 4.13 that $\hat{n}_{k_1} < \hat{n}_{k_2} < \ldots$ for some infinite increasing sequence of integers $k_1, k_2, \ldots$. We start with the lower bound.

Lemma 4.7. $\mathrm{PM}_{(6x^2+5x-3)} \in \mathrm{B}_{(6x+1)}$-ENPG *for every integer* $x > 1$.

Proof. Given an integer $x > 1$ we construct a $\mathrm{B}_{(6x+1)}$-ENPG representation of $\mathrm{PM}_{(6x^2+5x-3)}$. Figure 4.3 depicts the structure of the open clique representations $\cup\mathcal{P}_K$ and $\cup\mathcal{P}_{K'}$. The segments $\mathcal{S}(K, K')$ are numbered from 0 to $2x$ in increasing distance from the edge e. The non-segments of $\cup\mathcal{P}_K$ (maximal paths of $\cup\mathcal{P}_K \setminus \cup\mathcal{P}_{K'}$) are numbered in the same manner, and the non-segments of $\cup\mathcal{P}_{K'}$ (maximal paths of $\cup\mathcal{P}_{K'} \setminus \cup\mathcal{P}_K$) are numbered in decreasing order of their distance from e'.

Let $\alpha_i = \min(3x + 1 - i, 2x)$ and $\alpha'_i = \max(0, x + 2 - i)$ for $i \in [0, 2x]$. α_i (resp. α'_i) is chosen as the most distant non-segment reachable by a path of $\mathcal{P}_K$ (resp. $\mathcal{P}_{K'}$) starting at segment i and having at most $6x+1$ bends. We observe that $\alpha_i, \alpha'_i \in [0, 2x]$ for every i. Indeed, (i) $3x + 1 - i \geq x + 1 > 1$ implying that α_i is positive, (ii) α'_i is non-negative by definition, (iii) α_i is at most $2x$ by definition, and (iv) if $\alpha'_i > 0$ we have $\alpha'_i = x + 2 - i \leq x + 2 \leq 2x$. Furthermore, we observe that $\alpha'_i \leq \alpha_i$ for every $i \in [0, 2x]$. Indeed $\alpha'_i > \alpha_i$ would imply $x + 2 - i > 3x + 1 - i$ which is equivalent to $2x < 1$, a contradiction. We conclude that $0 \leq \alpha'_i \leq \alpha_i \leq 2x$.

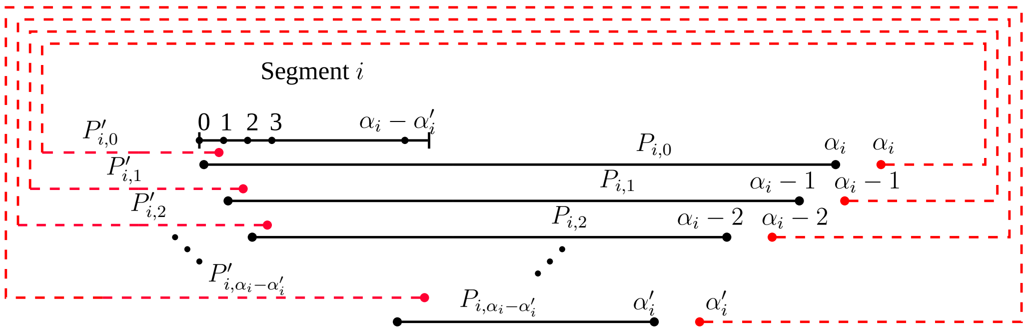

Figure 4.4. The paths terminating at segment i.

Given the above facts, we proceed with our construction. A segment numbered i contains $\alpha_i - \alpha'_i + 1$ internal vertices numbered from 0 in decreasing order of their distance from e. Every non-segment consists of one horizontal and one vertical edge as shown in Figure 4.3. For every $i \in [0, 2x]$ and every $j \in [0, \alpha_i - \alpha'_i]$ our construction contains four paths, two of which start at segment i on the left (as shown in Figure 4.4) and two of which start at segment i on the right.

- The path $P_{i,j} \in \mathcal{P}_K$ (resp. $\bar{P}_{i,j} \in \mathcal{P}_K$) starts at vertex j of segment i on the left (resp. right) side of e and ends at the unique intermediate vertex (the bend) of the non-segment $\alpha_i - j$ of $\cup\mathcal{P}_K$ on the right (resp. left) side of e.
- The path $P'_{i,j} \in \mathcal{P}_{K'}$ (resp. $\bar{P}'_{i,j} \in \mathcal{P}_{K'}$) starts at vertex $j+1$ of segment i on the left (resp. right) side of e' and ends at the unique intermediate vertex of the non-segment $\alpha_i - j$ of $\cup\mathcal{P}_{K'}$ on the right (resp. left) side of e'.

We note that all the paths $P_{i,j}$ and $\bar{P}_{i,j}$ cross the edge e and therefore correctly represent the clique K. Similarly, the paths $P'_{i,j}$ and $\bar{P}'_{i,j}$ represent the clique K'.

We now show that every path contains at most $6x+1$ bends. For this purpose we first observe that the number of bends between

- segment i and the edge e is $2i$,
- segment i and the edge e' is $4x - 2(i-1)$,
- non-segment $\bar{i}$ and the edge e is $\max(2\bar{i} - 1, 0)$,

- non-segment $\bar{i}$ and the edge e' is $4x - 2\bar{i} + 3$.

Since the path $P_{i,j}$ starts from segment i and ends at non-segment $\alpha_i - j$, its number of bends is

$$\begin{aligned} & 2i + \max(2(\alpha_i - j) - 1, 0) \leq 2i + \max(2\alpha_i - 1, 0) \\ \leq \;& 2i + \max(6x + 1 - 2i, 0) = \max(6x + 1, 2i) \\ \leq \;& \max(6x + 1, 4x) = 6x + 1. \end{aligned}$$

Similarly, the number of bends of $P'_{i,j}$ is

$$\begin{aligned} & 4x - 2(i-1) + (4x - 2(\alpha_i - j) + 3) \\ = \;& 8x + 5 - 2i - 2(\alpha_i - j) \leq 8x + 5 - 2i - 2\alpha'_i \\ = \;& 8x + 5 - 2i - 2\max(0, x + 2 - i) \leq 8x + 5 - 2i - 2(x + 2 - i) \\ = \;& 6x + 1. \end{aligned}$$

Since the paths $\bar{P}_{i,j}$ and $\bar{P}'_{i,j}$ are symmetric to $P_{i,j}$ and $P'_{i,j}$ respectively we conclude that our construction is a $\mathrm{B}_{(6x+1)}$-ENPG representation of the cobipartite graph (K, K', E). It remains to show that this graph is $\mathrm{P}\textsc{m}_{(6x^2+5x-3)}$.

The number of vertices of K is the number of paths $P_{i,j}$ and $\bar{P}_{i,j}$ which is equal to twice the number of paths $P_{i,j}$. Therefore

$$|K| = 2\sum_{i=0}^{2x}(\alpha_i - \alpha'_i + 1) = 2\sum_{i=0}^{2x}\alpha_i - 2\sum_{i=0}^{2x}\alpha'_i + 4x.$$

This is also the number of vertices of K'. Moreover we have

$$\sum_{i=0}^{2x} \alpha_i = \sum_{i=0}^{x} 2x + \sum_{i=x+1}^{2x} (3x + 1 - i) = \frac{7}{2}x^2 + \frac{5}{2}x$$

and

$$\sum_{i=0}^{2x} \alpha'_i = \sum_{i=0}^{x+2} (x + 2 - i) = \frac{(x+2)(x+3)}{2}.$$

By combining the above equations we conclude

$$|K| = 6x^2 + 5x - 3.$$

We now conclude the proof by showing that the edges E constitute a perfect matching of K and K'. We will show that given two paths $P \in \mathcal{P}_K$ and $P' \in \mathcal{P}_K$ we have $P \sim P'$ if and only if $P = P_{i,j}$ and $P' = P_{i,j}$ for some i, j or $P = \bar{P}_{i,j}$ and $P' = \bar{P}'_{i,j}$ for some i, j. Since the case for paths $\bar{P}_{i,j}$ is symmetric, we will consider only paths $P_{i,j}$ and $P'_{i,j}$. We observe that every path crosses at least one segment boundary. Therefore, $P \sim P'$ if and only if (i) P and P' intersect in a segment that contains an endpoint from both P and P', and (ii) P and P' do not cross a common segment endpoint. Then for two paths $P_{i,j}$ and $P'_{i',j'}$ we have $P_{i,j} \sim P'_{i',j'}$ only if $i = i'$ and they intersect at segment i. By our construction, this can happen only if $j' \geq j$ in which case $P_{i,j} \cap P'_{i,j'}$ is the path between vertices j and $j' + 1$ of segment i. We now recall that $P_{i,j}$ and $P'_{i,j'}$ end at non-segments $\alpha_i - j$ of $\cup \mathcal{P}_K$ and $\alpha_i - j'$ of $\cup \mathcal{P}_{K'}$, respectively. Then, whenever $j' > j$ $P_{i,j}$ and $P'_{i,j'}$ cross a common segment endpoint. Therefore, $P_{i,j} \sim P'_{i',j'}$ only if $i = i'$ and $j = j'$. By the same observations, $P_{i,j} \sim P'_{i,j}$ for every $i \in [0, 2x]$ and $j \in [0, \alpha_i - \alpha'_i]$. $\square$

Corollary 4.8. $\hat{n}_{6x+1} \geq 6x^2 + 5x - 3$ *for every positive integer* x.

We now provide an upper bound on $\hat{n}_k$. We first show an upper bound on the number of bends in a B_k-ENPG representation of a clique (Lemma 4.9). Then we show that this bound implies an upper bound on the number of segments in the representation (Lemma 4.11), and finally using this result we bound $\hat{n}_k$ from above (Lemma 4.12).

Lemma 4.9. *Let K be a complete graph with B_k-ENPG representation $\langle H, \mathcal{P} \rangle$ where every path contains at least m bends. Then $\cup \mathcal{P}_K$ contains at most $2 \left\lfloor \frac{2k+m \cdot \delta_K}{2} \right\rfloor$ bends, where δ_K is 1 whenever K is a closed clique and 0 otherwise.*

Proof. If K is an open clique ($\delta_K = 0$), then there exists an edge e contained in every $P \in \mathcal{P}_K$. e divides $\cup \mathcal{P}_K$ into two trails each of which contains at most k bends. Therefore, $\cup \mathcal{P}_K$ contains at most $2k = 2 \left\lfloor \frac{2k+m \cdot \delta_K}{2} \right\rfloor$ bends. If K is a closed clique ($\delta_K = 1$), consider a trail $P = (e_1, \ldots, e_\ell)$ of $\mathcal{P}_K$ with m bends. Let P_1 (resp. P_ℓ) be a trail containing e_1 (resp. e_ℓ) with the maximum number of edges from $\cup \mathcal{P}_K \setminus P$. We have $\cup \mathcal{P}_K = P \cup P_1 \cup P_\ell$, because otherwise there is an edge $e \in \cup \mathcal{P}_K \setminus (P \cup P_1 \cup P_\ell)$ that is included in some trail P' that contains neither e_1 nor e_ℓ and P' is not contained in P. Then P' does not intersect P, contradicting the fact that K is a clique. Since P has m bends and P_1, P_ℓ have at most k bends each, the number of bends of $\cup \mathcal{P}_K$ is at most $2k + m$. Moreover, this number is even because $\cup \mathcal{P}_K$ is a closed trail. Therefore, the number of bends of $\cup \mathcal{P}_K$ is at most $2 \lfloor \frac{2k+m}{2} \rfloor = 2 \left\lfloor \frac{2k+m \cdot \delta_K}{2} \right\rfloor$. $\square$

Corollary 4.10. *A clique K of a B_1-ENPG graph is an open clique and $\cup \mathcal{P}_K$ has at most 2 bends.*

Lemma 4.11. *Let $G = (K, K', E) \in \mathcal{C}_{n,n}$ with a B_k-ENPG representation $\langle H, \mathcal{P} \rangle$ where the minimum number of bends of a path $P \in \mathcal{P}_K$ (resp. $P' \in \mathcal{P}_{K'}$) is m (resp. m'). Then*

$$|\mathcal{S}(K, K')| \leq 2k + \frac{\max(m, 2) + \max(m', 2)}{2}.$$

Proof. Let $\delta \stackrel{def}{=} 2 - \delta_K - \delta_{K'} \in [0, 2]$ be the number of open cliques among K, K', and $s \stackrel{def}{=} |\mathcal{S}(K, K')|$. If $\cup \mathcal{P}_K = \cup \mathcal{P}_{K'}$ then $s = 1$, satisfying the claim. Otherwise, there are exactly $2s$ segment endpoints at most 2δ of which can be terminating. At every non-terminating segment endpoint there is at least one bend of one of $\cup\mathcal{P}_K, \cup\mathcal{P}_{K'}$. Therefore, the total number of bends of $\cup\mathcal{P}_K$ and $\cup\mathcal{P}_{K'}$ is at least $2s - 2\delta$. By Lemma 4.9, $\cup\mathcal{P}_K$ (resp. $\cup\mathcal{P}_{K'}$) contains at most $2\left\lfloor \frac{2k+m\cdot\delta_K}{2} \right\rfloor$ (resp. $2\left\lfloor \frac{2k+m'\cdot\delta_{K'}}{2} \right\rfloor$) bends. Therefore,

$$
\begin{aligned}
2s - 2\delta &\leq 2\left\lfloor \frac{2k+m\cdot\delta_K}{2} \right\rfloor + 2\left\lfloor \frac{2k+m'\cdot\delta_{K'}}{2} \right\rfloor \\
s - \delta &\leq \left\lfloor \frac{2k+m\cdot\delta_K}{2} \right\rfloor + \left\lfloor \frac{2k+m'\cdot\delta_{K'}}{2} \right\rfloor \leq 2k + \frac{m\cdot\delta_K + m'\cdot\delta_{K'}}{2} \\
s &\leq 2k + \frac{m\cdot\delta_K + m'\cdot\delta_{K'}}{2} + 2 - \delta_K - \delta_{K'} \\
&\leq 2k + \frac{\max(m,2) + \max(m',2)}{2}
\end{aligned}
$$

where the last step can be easily verified by substituting the three possible values of the pair $\delta_K, \delta_{K'}$. □

Lemma 4.12.

$$\hat{n}_k \leq 8k^2 + 8k + 4.$$

Proof. Let $\mathrm{P{\small M}}_n = (K, K', E)$ with a B_k-ENPG representation $\langle H, \mathcal{P} \rangle$. Let $\mathcal{S} = \mathcal{S}(K, K')$, and m (resp. m') the smallest number of bends of a path of $\mathcal{P}_K$ (resp. $\mathcal{P}_{K'}$).

For an edge $e = \{v, v'\} \in E$ we say that e is *realized* in segment $S \in \mathcal{S}$ if $P_v \cap P_{v'} \cap S \neq \emptyset$. Every edge $\{v, v'\}$ is realized in at least one segment, because otherwise $P_v \cap P_{v'} = \emptyset$, contradicting the fact that $\{v, v'\} \in E$. For a segment S let E_S be the set of edges realized in segment S. Then $E = \cup_{S \in \mathcal{S}} E_S$. In the following, we first provide an upper bound on $|E_S|$, and using Lemma 4.11 which bounds the number of segments we derive a bound on $|E|$.

Let without loss of generality $E_S = \{\{v_1, v'_1\}, \{v_2, v'_2\}, \ldots\}$. Let $\mathcal{P}_S = \{P_{v_1}, P_{v_2}, \ldots\}$

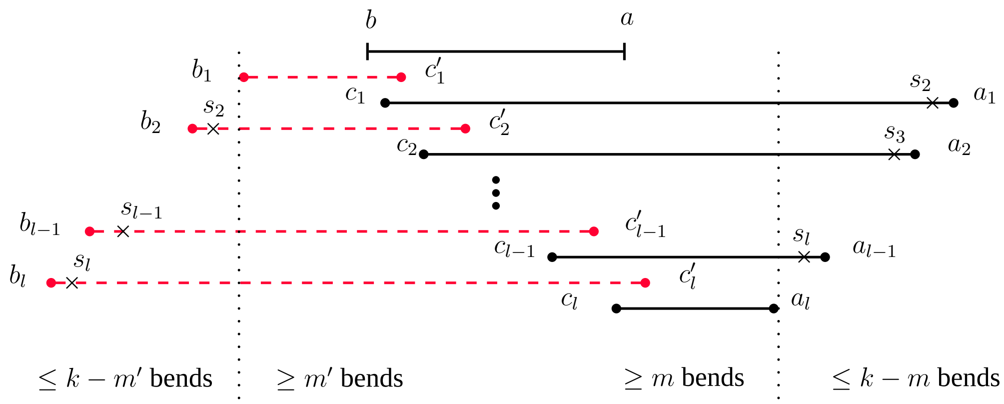

Figure 4.5. The structure of the path $\mathcal{P}_S$ and $\mathcal{P}'_S$ in the proof of Lemma 4.12.

and $\mathcal{P}'_S = \{P_{v'_1}, P_{v'_2}, \ldots\}$. We first assume that every path $\mathcal{P}_S \cup \mathcal{P}'_S$ crosses at least one endpoint of S, an assumption that will be relaxed at the end of the proof. Then, every such path crosses exactly one endpoint of S, since if a path, say P_{v_i}, crosses both endpoints of S, P_{v_i} splits from every other path of $\mathcal{P}_S \cup \mathcal{P}'_S$ contradicting $P_{v_i} \sim P_{v'_i}$. Let without loss of generality $P_{v_1}, \ldots, P_{v_\ell}$ be the paths of $\mathcal{P}_S$ that cross a given endpoint, say a, of S. Then, $P_{v'_1}, \ldots, P_{v'_\ell}$ are paths that cross the other endpoint, say b, of S, since P_{v_i} and P'_{v_i} cannot cross the same endpoint.

Consult Figure 4.5 for the rest of the discussion. Let c_i (resp. c'_i) be the endpoint of P_{v_i} (resp. $P_{v'_i}$) in S. Let also a_i (resp. b_i) be the endpoint of P_{v_i} (resp. $P_{v'_i}$) that is not in S. Assume without loss of generality that the vertices a_i are ordered in decreasing distance from a (on P_i). Since P_{v_1} and Pv'_1 have an intersection in S, c'_1 is between c_1 and a. We claim that c_2 is between c'_1 and a. Indeed, otherwise $P_{v_2} \cap P_{v'_1} \neq \emptyset$, and since $\{v_2, v'_1\} \notin E_S$, it must be the case that $P_{v_2} \nsim P_{v'_1}$, i.e. P_{v_2} and $P_{v'_1}$ cross a common segment endpoint. Then P_{v_1} crosses this endpoint too, implying that $P_{v_1} \nsim P_{v'_1}$, a contradiction.

For the same reason as above, c'_2 is between c_2 and a. Then $P_{v'_2} \cap P_{v_1} \neq \emptyset$. Therefore, there is a segment endpoint s_2 common to $P_{v'_2}$ and P_{v_1}. Clearly, s_2 is not in $INT(P_{v_2})$, since in such a case $P_{v_2} \nsim P_{v'_2}$, implying that $\{v_2, v'_2\} \notin E$. We conclude that $s_2 \in INT(P_{v_1}) \cap INT(P_{v'_2}) \setminus INT(P_{v_2})$. Continuing in this way, we get segment endpoints $s_3 \in INT(P_{v_2}) \cap INT(P_{v'_3}) \setminus INT(P_{v_3}), \ldots s_\ell \in INT(P_{v_{\ell-1}}) \cap INT(P_{v'_\ell}) \setminus INT(P_{v_\ell})$. From this relations it follows that all these endpoints are distinct elements of $INT(P_{v_1}) \setminus INT(P_{v_\ell})$. By symmetry, we conclude that $INT(P_{v'_\ell}) \setminus INT(P_{v'_1})$ contains at least $\ell - 1$ segment endpoints.

We are now ready to upper bound $|E_S|$ depending the lower and upper bounds m, m' and k on the number of bends of a path. $|E_S|$ will be shown to be decreasing in m and m'. However, the number of segments are increasing with m and m'. In the sequel we analyze this tradeoff. P_{v_1} has at most k bends and P_{v_ℓ} contains at least m bends. Therefore, $INT(P_{v_1}) \setminus INT(P_{v_\ell})$ contains at most $k - m$ bends. Similarly, $INT(P_{v'_\ell}) \setminus INT(P_{v'_1})$ contains at most $k - m'$ bends. Every segment endpoint is a bend of at least one of the involved paths. Therefore,

$$\ell - 1 \leq k - m + k - m'.$$

Considering also the ℓ' paths of $\mathcal{P}_S$ that cross b and the paths of $\mathcal{P}'_S$ that cross a, we conclude that

$$|E_S| = \ell + \ell' \leq 4k + 4 - 2(m + m').$$

By Lemma 4.11, $|\mathcal{S}| \leq \min(2\left\lfloor \frac{2k+m\cdot\delta_K}{2} \right\rfloor, 2\left\lfloor \frac{2k+m'\cdot\delta_K}{2} \right\rfloor) \leq 2k + m + m'$. Therefore,

$$|E| \leq \sum_{S \in \mathcal{S}} |E_S| \leq (2k + M)(4k + 4 - 2M)$$

where M is $m + m'$.

Finally we relax our assumption that every path crosses at least one segment boundary. There is at most one path $P \in \mathcal{P}_K$ that does not cross segment boundaries, for two such paths do not intersect, thus cannot be in the representation of a clique.

We conclude that

$$|E| \leq (2k + M)(4k + 4 - 2M) + 2 \leq (2k + 1)(4k + 2) + 2 = 8k^2 + 8k + 4$$

where the second inequality holds because the maximum of the left hand side is attained at $M = 1$. $\square$

We are now ready to prove the main result of this section.

Theorem 4.13. *There is an infinite increasing sequence of integers* $\{k_i : i = 1, 2, \ldots\}$ *such that*

$$\mathrm{B}_0\text{-ENPG} \subsetneq \mathrm{B}_1\text{-ENPG} \subsetneq \mathrm{B}_{k_1}\text{-ENPG} \subsetneq \mathrm{B}_{k_2}\text{-ENPG} \subsetneq \cdots$$

where $\lim_{i\to\infty} \frac{k_{i+1}}{k_i} = \sqrt{48}$.

Proof. We first note that C_4 is not in the family of interval graphs which coincides with the family of B_0-ENPG graphs. On the other hand a B_1-ENPG representation of C_4 ie easily obtained by surrounding a 2×2 square with four L-shaped paths.

We now provide an infinite sequence $k_0 = 1, k_1, k_2, \ldots$ such that $\hat{n}_{k_i} < \hat{n}_{k_{i+1}}$ for every $i > 0$, implying B_{k_i}-ENPG $\subsetneq$ $\mathrm{B}_{k_{i+1}}$-ENPG. By Lemma 4.12 we have $\hat{n}_{k_i} < 8k_i^2 + 8k_i + 5$ for any $k_i \geq 1$. Let x_i be the smallest integer such that $8k_i^2 + 8k_i + 5 \leq 6x_i^2 + 5x_i - 3$. Note that the left hand side is at least 21, and therefore $x_i > 1$. Let $k_{i+1} = 6x_i + 1$. By Corollary 4.8, $\hat{n}_{k_{i+1}} \geq 6x_i^2 + 5x_i - 3$. Therefore, $\hat{n}_{k_{i+1}} > \hat{n}_{k_i}$.

We now show that k_{i+1}/k_i converges to $\sqrt{48}$:

$$6(x_i - 1)^2 + 5(x_i - 1) - 3 < 8k_i^2 + O(k_i)$$

by the way x_i is chosen. Therefore, $x_i = \frac{2}{\sqrt{3}}k_i + o(k_i)$, and finally $k_{i+1} = 6x_i + 1 = \sqrt{48}k_i + o(k_i)$. $\square$

We conclude this section by possible improvements of the above result. In the construction of Lemma 4.7 we use open cliques. One can use closed cliques that, by Lemma 4.9, lead to more segments than open cliques, consequently increasing the lower bound. Another observation is that the non-segments of the construction contain bends that are clearly not segment endpoints. Recalling the proof of Lemma 4.11, we conclude that this example is not tight. One can modify the construction such that almost every bend is an endpoint of a segment, implying a further improvement of the lower bound. On the other hand, the upper bound can be improved by considering the minimum number, say m_S, of bends of a path in the set $\mathcal{P}_S$, instead of the global minimum m that we consider in the proof of Lemmata 4.11 and 4.12. These improvements will certainly decrease the ratio of $\sqrt{48}$ at the expense of overly complicating the analysis, with the asymptotic behaviour of the sequence k_i remaining exponential.

5. GRAPHS OF EDGE-INTERSECTING NON-SPLITTING ONE BEND PATHS IN A GRID

5.1. Overview

In this chapter, we consider graphs of Edge-Intersecting and Non-Splitting One Bend Paths in a Grid (B_1-ENPG). In the previous chapter we showed that ENP = ENPG. Whenever the host graph is a grid, it is common to use the following notion: a *bend* of a path on a grid is an internal point in which the path changes direction. An ENPG graph is B_k-ENPG if it has a representation in which every path has at most k bends. In the same chapter, it was shown that ENPG contains an infinite hierarchy of subclasses that are obtained by restricting the number of bends in the paths. Motivated by this result, in this chapter we focus on one bend ENPG graphs.

In Section 5.2 we start with some basic results. We show that cycles and trees are B_1-ENPG by providing a representation construction for an arbitrary input.

In Section 5.3 we consider a special case of B_1-ENPG graphs: B_1-ENPG$\cap$ split graphs. We first give a characterization of these graphs in Theorem 5.2: a split graph $G = S(K, S, E)$ is B_1-ENPG if and only if S can be partitioned into two sets S_L, S_R such that the K-S_L and K-S_R incidence matrices have the consecutive ones property for its columns. By using this result, it is possible to design efficient algorithms for problems known to be NP-complete in split graphs. For example maximum cut and domination problems are NP-hard in split graphs. The complexity of these problems in B_1-ENPG split graphs is open. We then show in Theorem 5.8 that the B_1-ENPG recognition problem is NP-complete even for a very restricted subfamily of split graphs. The hardness comes mainly from the difficulty of deciding the position of each path (left or right of the common edge of paths representing the clique). This result however do not necessarily imply the NP-completness of B_k-ENPG recognition problem. The complexity of this problem is open. Another research direction is to investigate (G, S)

pair recognition where G is an arbitrary graph and $S \subseteq G$ is a split graph. Introducing red edges can possibly make the problem polynomial time solvable.

In Section 5.4, we consider another special case of B_1-ENPG graphs: B_1-ENPG$\cap$ cobipartite graphs. We show that there are two types of representations and provide a characterization for each type in Lemmata 5.17 and 5.18. Theorem 5.19 combines these two results. This theorem implies a naive polynomial time $O(n^4)$ recognition algorithm. In the sequel we provide a linear time recognition algorithm. The forbidden subgraph characterization of this graph class is open. It would be also interesting to consider B_k-ENPG cobipartite graphs.

The maximum cut problem is the problem of partitioning the vertices of a graph such that the number of edges incident to both sets are maximum. This problem remains NP-complete even in co-bipartite graphs and in split graphs. By Theorem 5.19 we know that if a cobipartite graph is B_1-ENPG then either there are two connected co-bipartite chain graphs or there are at most 4 vertices whose removal leave a co-bipartite chain graph. In [29] we show that maximum cut problem in co-bipartite chain graphs can be solved in polynomial time by using a dynamic programming algorithm. With some adjustments, the same algorithm can be used to solve maximum cut problem in B_1-ENPG co-bipartite graphs. On the other hand, Theorem 5.3 characterizes similarly B_1-ENPG split graphs, a natural next step is to consider maximum cut problem in this graph class.

5.2. Prelimineries

We first observe that some well-known graph classes are included in B_1-ENPG.

Proposition 5.1. *(i) Every cycle is* B_1-ENPG.
(ii) Every tree is B_1-ENPG.

Proof. (i) For $k = 3$ three identical paths consisting of one edge constitutes a B_1-ENPG representation of C_3. For $k = 4$ Figure 5.1a depicts a B_1-ENPG

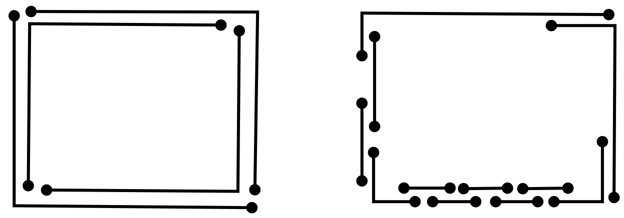

Figure 5.1. (a) A B_1-EPG representation of C_4, (b) a B_1-EPG representation of C_{11}.

representation of C_4. Finally for any $k > 4$, we can construct a C_k as shown in Figure 5.1b for the case $k = 11$.

(ii) Given a representation $\langle H, \mathcal{P} \rangle$ of a B_1-ENPG graph G, we denote by R_U the bounding rectangle of $\mathcal{P}_U$ for $U \subseteq V(G)$. Let T be a tree with a root r. We prove the following claim by induction on the structure of T (see Figure 5.2). T has a B_1-ENPG representation $\langle H, \mathcal{P} \rangle$ in which the corners of R_T can be renamed as a_T, b_T, c_T, d_T in counterclockwise order such that i) every path of $\mathcal{P}$ has exactly one bend, ii) b_T is a bend of P_r, iii) a_T is an endpoint of P_r, iv) a_T is used exclusively by P_r.

If T is an isolated vertex, any path with one bend is a representation of T. Moreover, it is easy to verify that it satisfies conditions i) through iv).

Otherwise let $T_1, \ldots, T_k$ be the subtrees of T obtained by the removal of r, with roots $r_1, \ldots, r_k$ respectively. By the inductive hypothesis every such subtree T_i has a representation with bounding box $a_{T_i}, b_{T_i}, c_{T_i}, d_{T_i}$ satisfying conditions i) through iv). We now build a representation of T satisfying the same conditions. We shift and rotate the representations of $T_1, \ldots, T_k$ so that the bounding rectangles do not intersect and the vertices $a_{T_1}, b_{T_1}, a_{T_2}, b_{T_2}, \ldots, a_{T_k}, b_{T_k}$ are on the same horizontal line and in this order. We extend the paths $P_{r_2}, \ldots, P_{r_k}$ representing the roots of the trees $T_2, \ldots, T_k$ such that the endpoint a_{T_i} of P_{r_i} is moved to a_{T_1}. Since a_{T_i} is used exclusively by P_{r_i} this modification does not cause P_{r_i} to split from a path of $\mathcal{P}_{V(T_i)}$. Therefore, the individual trees $T_1, \ldots, T_K$ are properly represented. Clearly, if two paths from different subtrees T_i, T_j $(i < j)$ intersect, then one of the intersecting paths must be P_{r_j}. P_{r_j} intersects the bounding rectangle of T_i only at the path between a_i and b_i. As every path of $\mathcal{P}_{V(T_i)}$, in particular one intersecting P_{r_j} has one bend, such a path splits from P_{r_j}. Therefore, for any pair of vertices $(v_i, v_j) \in T_i \times T_j$ we have that v_i and v_j are

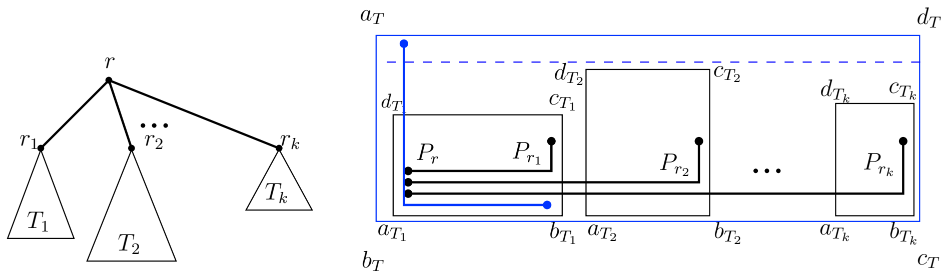

Figure 5.2. A construction for B_1-ENPG representation of trees.

non-adjacent in ENPG($\mathcal{P}$), as required.

We rename the corners of the bounding rectangle R_T such that $b_T = a_{T_1}$. We now add the path P_r from b_{T_1} to a_T with a bend at b_T. The conditions i), ii), iii) are satisfied. We extend P_r by one edge at a_T to make sure that a_T is exclusively used by P_r, thus satisfying condition iv). P_r intersects only R_{T_1}. This intersection is the path between b_{T_1} and d_{T_1} bending at a_{T_1}. Every path that intersects P_r and does not split from it must bend at a_{T_1}. As a_{T_1} is used exclusively by P_{r_1}, P_{r_1} is the only path that possibly satisfies $P_{r_1} \sim P_r$. We now observe that $P_{r_i} \sim P_r$ for every $i \in [k]$. Therefore r is adjacent to the root of T_j in ENPG($\mathcal{P}$), as required.

$\square$

5.3. Split Graphs

In this section we present a characterization theorem (Theorem 5.3) for B_1-ENPG split graphs in Section 5.3.1. Then, Section 5.3.2 proceeds with some properties of these graphs implied by this theorem. An interesting implication of one of these properties is that the family of B_1-ENPG is properly included in the family of B_2-ENPG graphs. Finally, using Theorem 5.3, we prove in Section 5.3.3 that the recognition problem of B_1-ENPG graphs is NP-complete even in a very restricted subfamily of split graphs.

5.3.1. Characterization of B_1-ENPG Split Graphs

We recall that a binary matrix has the *consecutive ones property (for columns)* if there is a permutation of its rows such that in every column all the one entries are consecutive.

The following lemma shows that if G is B_1-ENPG then G has a representation $\langle H, \mathcal{P} \rangle$ with H being a tree.

Lemma 5.2. $B_1\text{-ENPG} \cap \text{SPLIT} \subseteq \text{ENPT} \cap \text{SPLIT}$.

Proof. Let $G = S(K, S, E)$ be a B_1-ENPG split graph with a representation $\langle H, \mathcal{P} \rangle$. We want to show that there is a representation $\langle H', \mathcal{P}' \rangle$ of G such that $\cup\mathcal{P}'$ is a tree, i.e. $\cup\mathcal{P}'$ does not contain any cycle.

By Corollary 4.10, we know that $\cup\mathcal{P}_K$ is a path with at most two bends in every representation $\langle H, \mathcal{P} \rangle$ of G. Suppose that there exists a vertex $s \in S$ such that $|\mathcal{S}(P_s, \cup\mathcal{P}_K)| > 1$. Then $P_s \cup \cup\mathcal{P}_K$ contains a cycle, therefore at least 4 bends. But P_s has at most one bend and $\cup\mathcal{P}_K$ has at most two bends, a contradiction. Therefore, $\mathcal{S}(P_s, \cup\mathcal{P}_K)$ consists of one segment for every vertex $s \in S$.

If $\cup\mathcal{P}_K$ has two bends, (without loss of generality the subpath between the bends is vertical) then we subdivide the top and bottom edges of this vertical subpath, so that the vertical distance between any two horizontal edges in different subpaths of $\cup\mathcal{P}_K$ is at least three. Consider the path P_s for some $s \in S$. By the discussion in the previous paragraph, P_s intersects $\mathcal{P}_K$ in one segment. Consider the (at most two) subpaths (that we term tails in this discussion) of $P_s \setminus \mathcal{P}_K$. Every such tail can be shortened to one edge without affecting the relationship of P_s with the paths $\mathcal{P}_K$ as P_s intersects with $\mathcal{P}_S$ in one segment. Moreover, for every $s' \in S$, (i) s is not adjacent to s', and (ii) after the shortening of the tails of P_s and $P_{s'}$, the two paths are non intersecting. Let $\langle H', \mathcal{P}' \rangle$ be the resulting representation. Then H' consists of a path $\mathcal{P}'_K$ with at most 2 bends where the horizontal edges are at distance at least 3 to each other. Moreover,

$\cup\mathcal{P}' \setminus \cup\mathcal{P}'_K$ consists of edges each of which intersects $\mathcal{P}'_K$ in one vertex. We conclude that $\cup\mathcal{P}'$ is a tree. Therefore, $S(K, S, E)$ is B_1-ENPT. $\square$

In the rest of this section we assume without loss of generality that K is maximal, i.e. that no vertex of S is adjacent to all vertices of K. We also assume that G does not contain isolated vertices and twins.

Theorem 5.3. *A split graph $G = S(K, S, E)$ is* B_1-ENPG *if and only if S can be partitioned into two sets S_L, S_R such that the K-S_L and K-S_R incidence matrices have the consecutive ones property. Moreover, if G is* B_1-ENPG *it has a representation $\langle H, \mathcal{P}\rangle$ such that*

(i) P_u has no bends whenever $u \in K$, and

(ii) whenever $v \in S$ (i) P_v has one bend, (ii) $e_K \notin P_v$, and (iii) $P_v \cap \cup\mathcal{P}_K \neq \emptyset$.

Proof. ($\Rightarrow$) Assume that G is B_1-ENPG. By Lemma 5.2, G has a representation $\langle H, \mathcal{P}\rangle$ with H being a tree. We assume without loss of generality that $\cup\mathcal{P}_K$ is a straight line between two vertices $q_L \in T_L$, $q_R \in T_R$. Because otherwise we can transform $\cup\mathcal{P}_K$ into a straight line, by first replacing e_K by a sufficiently long path and then rotating the entire subtree hanging from a bend point by 90 degrees. The edge e_K divides the tree into two subtrees T_L and T_R and the path $\cup\mathcal{P}_K$ into two paths P_L and P_R. We subdivide the edge e_K into three edges e_L, e_K, e_R such that $e_L \in T_L$ (resp. $e_R \in T_R$). Consequently, every path $P \in \mathcal{P}$ that contains one of these three edges contains all of them. Suppose that a path P_v representing a vertex $v \in S$ contains e_K. If P_v does not have a bend then v is adjacent to all the vertices of K, contradicting the fact that K is maximal. Therefore, P_v has one bend. Assume without loss of generality that the bend of P_v is in T_R. Then we can remove all the edges $P_v \cap (T_L \cup \{e_K\})$ from P_v to get an equivalent representation in which P_v does not contain e_K. Therefore, there is a representation of G in which every path of $\mathcal{P}_S$ is contained in one of T_L, T_R.

For $X \in \{L, R\}$, let $S_X \overset{def}{=} \{v \in S : P_v \subseteq T_X\}$. By the preceding discussion $\{S_L, S_R\}$ is a partition of S. Consider a vertex $v \in S_X$, i.e. $P_v \subseteq T_X$. If P_v does not

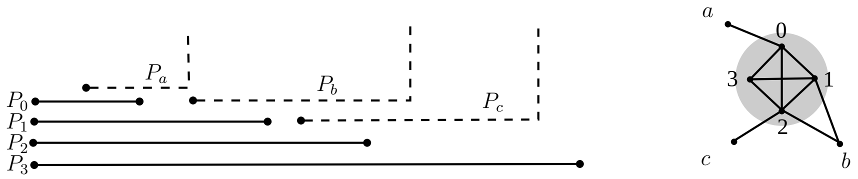

Figure 5.3. The representation of a B_1-ENPG split graph.

have a bend then it does not split from any path of $\mathcal{P}_K$. Therefore, we can get an equivalent representation in which P_v has one bend by first moving the endpoint of P_v that is farther from e_K to q_X, and then adding an edge to P_v at q_X so that q_X becomes a bend of P_v. Let $P'_v \stackrel{def}{=} P_v \cap \cup\mathcal{P}_K$ for every $v \in S$. If the bend of P_v is the endpoint of P'_v closer to e_K then P_v splits from every path of $\mathcal{P}_K$ that it intersects. In this case v is isolated, contradicting our assumption. We conclude that the bend of P_v is the endpoint of P'_v that is farther from e_K. Figure 5.3 depicts the subtree T_R of such a representation.

As every path $P_u \in \mathcal{P}_K$ contains e_K, it has one endpoint in P_L and one endpoint in P_R. For $X \in \{L, R\}$ the order of the endpoints of $\mathcal{P}_K$ on P_X induces a permutation σ_X on K. Consider the K-S_X incidence matrix, so that the rows representing vertices $u \in K$ are ordered in accordance to the permutation σ_X. Consider a vertex $v \in S_X$ and its corresponding path $P_v \subseteq T_X$. Let $u \in K$ be a neighbor of v in G. We observe that the endpoint of P_u in T_X is in P'_v. Then the endpoints of all the paths representing neighbors of v are in P'_v, i.e. they are consecutive in the permutation σ_X. In other words all the ones in column v of the K-S_X incidence matrix are consecutive.

($\Leftarrow$) Assume that S is partitioned into two sets S_L and S_R such that for $X \in \{L, R\}$ the K-S_X incidence matrix has the consecutive ones property, and let σ_X be a permutation of K that makes the ones of every column of the corresponding matrix consecutive. We now construct a B_1-ENPG representation of G. For a vertex $u \in K$, P_u is the path between the vertices $(-2\sigma_L(u), 0)$ and $(2\sigma_R(u), 0)$. For $v \in S_X$, let $u_1(v), u_2(v)$ be the indices of the first and last ones of column v of the K-S_X incidence matrix. If $v \in S_R$ then P_v is a one bend path from $(2u_1(v) - 1, 0)$ to $(2u_2(v) + 1, 1)$ with a bend at $(2u_2(v) + 1, 0)$, otherwise P_v is a one bend path from $(-2u_1(v) + 1, 0)$

to $(-2u_2(v)-1,1)$ with a bend at $(-2u_2(v)-1,0)$. We first note that K is a clique because $\mathcal{P}_K$ is a horizontal path and every path of $\mathcal{P}_K$ contains the edge $(0,0),(1,0)$. Second, we note that S is an independent set because all the paths of $\mathcal{P}_S$ are L shaped with the same orientation. Moreover, their bend points are distinct. Therefore any two intersecting such pats split at one of these bend points. We now observe that for any $v \in S_X$ and $u \in K$, $P_u \sim P_v$ if and only if $\sigma_X(u) \in [u_1(v), u_2(v)]$. By the way u an v are chosen, the last statement holds if and only if the corresponding entry in the $K-S_L$ incidence matrix is one, i.e. u and v are adjacent in G. Therefore, the constructed paths constitute a representation of G. □

5.3.2. Two Consequences of The Characterization of B_1-ENPG Split Graphs

The next two results (Lemma 5.4 and Theorem 5.5) are implied by the above characterization of Theorem 5.3.

Lemma 5.4. *(i) If $S(K,S,E)$ is a twin-free* B_1-ENPG *split graph then*

$$\sqrt{|K|} \leq |S| < |K|^2 .$$

(ii) All split graphs $S(K,S,E)$ with $|K| \leq 4$ are B_1-ENPG.

(iii) There is a split graph $S(K,S,E)$ with $|K|=5$ that is not B_1-ENPG.

Proof. (i) Let $\{S_L, S_R\}$ be a partition of S and σ_L, σ_R be the permutations of K satisfying the conditions of Theorem 5.3. We order the rows of the K-S_L and K-S_R incidence matrices by these permutations so that the one entries of every column are consecutive. For $X \in \{L,R\}$, every column of K-S_X has one row containing its first 1 and at most one row containing its first zero after its last 1 entry. Consider the (at most) $2|S_X|$ rows defined in this way. We observe that any other row of the K-S_X incidence matrix is identical to one of these rows. To see this observation, let i be a row from the $2|S_X|$ rows and $j > i$ the first row different from i. If there is a column that contains a 1 in the i-th row and a 0 in the j-th row, then j contains the first 1 of this column. Similarly, if a column contains a 0 in the i-th row and a 1 in the j-th row, then row j contains the first 0 after the 1-s of this column. Now suppose that $|K| > 4\,|S_L| \cdot |S_R|$. Then there are at least two vertices of K whose corresponding rows in both of K-S_L and K-S_R matrices are identical, contradicting our assumption that G is twin-free. Therefore $|K| \leq 4\,|S_L| \cdot |S_R| \leq |S|^2$.

Let S_d be the set vertices of S having degree d, and let $v \in S_d \cap S_L$. Then, when the rows of the K-S_L incidence matrix are ordered by the permutation σ_L, the column v contains exactly d consecutive ones. There are $|K|+1-d$ possible such columns. As the graph is twin-free, we have $|S_d \cap S_L| \leq |K| + 1 - d$ implying

$$|S_d| \leq 2(|K| + 1 - d). \tag{5.1}$$

We conclude

$$\begin{aligned} |S| &= |S_1| + \sum_{d=2}^{|K|-1} |S_d| + \left|S_{|K|}\right| \leq |K| + \sum_{d=2}^{|K|-1} 2(|K| + 1 - d) + 1 \\ &= |K| + (|K| - 2)(|K| + 1) + 1 = |K|^2 - 1. \end{aligned}$$

(ii) It is sufficient to show for $K = [4]$ and $S = 2^K$ where every vertex of S is adjacent to a different subset of vertices of K. Every two permutations σ_L, σ_S satisfy the consecutiveness condition for subsets of size $0, 1$ and 4. Let σ_L be the identity permutation and $\sigma_R = (3142)$. It is easy to verify that they satisfy the consecutiveness conditions of all the sets.

(iii) Consider a split graph $G = (K, S, E)$ with $K = [5]$ and $|S| = 9 < \left(\begin{array}{c} 5 \\ 2 \end{array} \right)$ where every vertex of S is adjacent to a distinct pair of K. We have $|S_2| = |S| = 9$. Therefore, G is not B_1-ENPG as otherwise it would constitute a contradiction to (5.1).

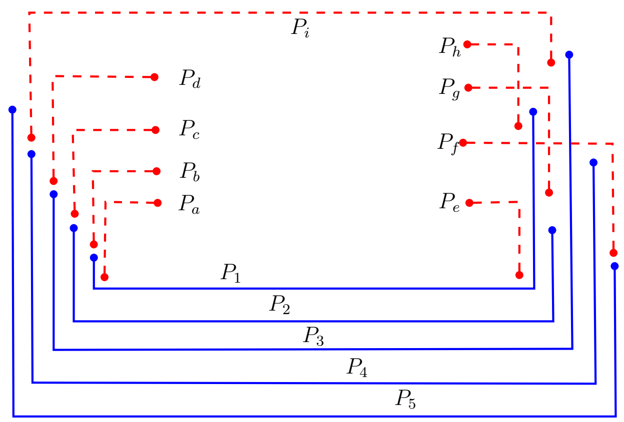

Figure 5.4. The B_2-ENPG representation of a non-B_1-ENPG split graph described in the proof of Theorem 5.5.

□

Theorem 5.5. B_1-ENPG $\subsetneq$ B_2-ENPG.

Proof. Consider the split graph $G = (K, S, E)$ where $K = [5]$, $S = \{a, b, c, d, e, f, g, h, i\}$ and $N(a) = \{1, 2\}$, $N(b) = \{2, 3\}$, $N(c) = \{3, 4\}$, $N(d) = \{4, 5\}$, $N(e) = \{2, 5\}$, $N(f) = \{2, 4\}$, $N(g) = \{1, 4\}$, $N(h) = \{1, 3\}$, $N(i) = \{3, 5\}$. We have shown in the proof of Lemma 5.4 that $G \notin B_1$-ENPG. Figure 5.4 depicts a B_2-ENPG representation of G. □

5.3.3. NP-completeness of B_1-ENPG split graph recognition

We now proceed with the NP-completeness of B_1-ENPG recognition. We first present a preliminary result that can be useful per se. Clearly, if the edge set of a graph G can be partitioned into two Hamiltonian cycles, then G is 4-regular. However, in the opposite direction we have the following:

Theorem 5.6. *The problem of determining whether the edge set of a 4-regular graph can be partitioned into two Hamiltonian cycles is* NP*-complete.*

Proof. The Hamiltonian cycle problem is NP-complete even for 3-regular graphs [30]. The theorem now follows from the fact that a 3-regular graph is Hamiltonian if and only if the edge set of its (4-regular) line graph can be partitioned into two Hamiltonian cycles [31]. □

A graph is *almost* d-regular if it can be obtained by removing a vertex from a d-regular graph. Clearly, a graph is almost d-regular if and only if all its vertices have degree d, except for d vertices with degree d-1. If the edge set of a graph can be partitioned into two hamiltonian paths, then it is almost 4-regular. On the other hand the edge set of an almost 4-regular graph can be partitioned into two hamiltonian paths if and only if the edge set of the corresponding 4-regular graph can be partitioned into two hamiltonian cycles. Therefore,

Corollary 5.7. *The problem of determining whether the edge set of an almost* 4*-regular graph can be partitioned into two Hamiltonian paths is* NP*-complete.*

Before stating the main result of this section we remark that a column of a binary matrix containing at most one 1 entry has consecutive ones under every permutation of the rows of the matrix. Therefore, a split graph is B_1-ENPG if and only if the graph obtained from it by the removal of all isolated vertices and degree 1 vertices is B_1-ENPG. However,

Theorem 5.8. *The* B_1-ENPG *recognition problem is* NP*-complete even for split graphs* (K, S, E) *where* $d(v) = 2$ *for every* $v \in S$.

Proof. The proof is by reduction from the problem of decomposing an almost 4-regular graph into two Hamiltonian paths. Given an almost 4-regular graph G, we construct the split graph (K, S, E) where $K = V(G)$, $S = E(G)$ and the edges of the split graph $E = \{\{e, u\}, \{e, v\} : \forall e = \{u, v\} \in E(G)\}$. It remains to show that (K, S, E) is B_1-ENPG if and only if $E(G)$ can be partitioned into two Hamiltonian paths.

Assume that $E(G)$ can be partitioned into two Hamiltonian paths H_L and H_R. This induces a partition of S into $S_L = E(H_L)$ and $S_R = E(H_R)$. Moreover, for $X \in \{L, R\}$ the order of the vertices of G in H_X induces a permutation σ_X of the vertices of $K = V(G)$. Let $X \in \{L, R\}$ and $e = \{u, v\} \in H_X$. Then u and v are consecutive in the permutation σ_X. However, u and v are the only indices that contain a one in the column of e. Therefore, the K-S_L incidence matrix with rows ordered according to σ_X has consecutive ones in every column. Therefore, by Theorem 5.3, (K, S, E) is B_1-ENPG.

Now assume that (K, S, E) is B_1-ENPG. Then, by Theorem 5.3, S can be partitioned into two sets S_L and S_R and there are two permutations σ_L, σ_R of K such that for $X \in \{L, R\}$ the K-S_X incidence matrix has consecutive ones in every column when its rows are ordered according to σ_X. The partition $\{S_L, S_R\}$ induces a partition $\{E_L, E_R\}$ of $E(G)$. The permutations σ_L, σ_R correspond to Hamiltonian paths H_L, H_R of K (a priori, not necessarily a Hamiltonian path of G). Let $e = \{u, v\} \in S_X = E_X$. Then u and v are consecutive in σ_X, thus adjacent in the Hamiltonian path H_X. Therefore, $e \in E(H_X)$. We conclude

$$\begin{aligned} E_L &\subseteq E(H_L) \\ E_R &\subseteq E(H_R) \\ E(G) &= E_L \cup E_R \subseteq E(H_L) \cup E(H_R) \\ |E(G)| &\leq |E(H_L)| + |E(H_R)| - |E(H_L) \cap E(H_R)| \end{aligned}$$

Let $n = |V(G)|$. As G is almost 4-regular, $|E(G)| = (4(n-4) + 3 \cdot 4)/2 = 2n - 2$. Moreover, $|E(H_R)| = |E(H_L)| = n - 1$ as H_L and H_R are Hamiltonian paths of K. Substituting in the above inequality, we get

$$2n - 2 \leq 2(n-1) - |E(H_L) \cap E(H_R)|$$

implying that (i) $E(H_L) \cap E(H_R) = \emptyset$ and that (ii) all inclusions above can be replaced by equalities. By (i) H_L and H_R are disjoint Hamiltonian paths of K, and by (ii) all their edges are edges of G, i.e. they are Hamiltonian paths of G. $\square$

A *double interval* graph is the intersection graph of a set of pairs of intervals in the real line. It is known that every 2-split graph is a double interval graph [32].

Corollary 5.9. *The* B_1-ENPG *recognition problem is* NP-*complete even when restricted to double interval graphs.*

5.4. Cobipartite Graphs

In Section 5.4.1, we characterize B_1-ENPG co-bipartite graphs. We show that there are two types of representations for B_1-ENPG co-bipartite graphs. For each type of representation, we characterize their corresponding graphs. These characterizations lead to a polynomial-time recognition algorithm. However we show in Section 5.4.2 that there is also a linear-time recognition algorithm.

By the following two observations, in the sequel we focus on connected twin-free graphs.

Observation 5.10. *Let G be a graph and G' obtained from G by removing a twin vertex until no twins remain. Then, G is* B_k-ENPG *if and only if G' is* B_k-ENPG.

Observation 5.11. *A graph G is* B_k-ENPG *if and only if every connected component of G is* B_k-ENPG.

5.4.1. Characterization of B_1-ENPG Co-bipartite Graphs

We proceed with definitions and two related lemmas (Lemma 5.14, Lemma 5.15) that will be used in each of the above mentioned characterizations.

Let S be a path of a graph H with endpoints u, v. Two path sets $\mathcal{P}_u$, $\mathcal{P}_v$ *meet at* S if (i) every path of $\mathcal{P}_x$ contains x where $x \in \{u, v\}$, (ii) has an endpoint among the internal vertices of S, and (iii) a pair of paths $P_u \in \mathcal{P}_u$, $P_v \in \mathcal{P}_v$ may intersect only in S (see Figure 5.5).

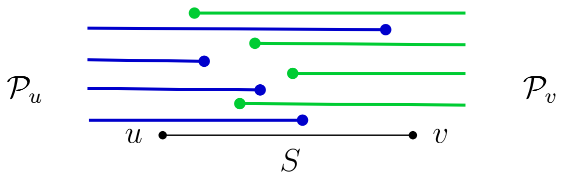

Figure 5.5. Two path sets $\mathcal{P}_u$, $\mathcal{P}_v$ meet at a path S with endpoints u and v.

A graph $G = (V, E)$ is a *difference graph* (equivalently *bipartite chain graph*) if every $v_i \in V$ can be assigned a real number a_i and there exists a positive real number T such that (i) $|a_i| < T$ for all i and (ii) $\{v_i, v_j\} \in E$ if and only if $|a_i - a_j| \geq T$. Every difference graph is bipartite where the bipartition is according to the sign of a_i.

Theorem 5.12. *[33] If $G = (V, E)$ be a bipartite with bipartition $V = X \cup Y$. Then the following statements are equivalent:*

(i) G is a difference graph with bipartition $V = (X \cup Y)$.

(ii) Let $\delta_1 < \delta_2 < \ldots \delta_s$ be distinct nonzero degrees in X, and set $\delta_0 = 0$. Let $\sigma_1 < \sigma_2 < \ldots \sigma_t$ be distinct nonzero degrees in Y, and set $\sigma_0 = 0$. Let $X = X_0 \cup X_1 \cup \ldots X_s$, $Y = Y_0 \cup Y_1 \cup \ldots \cup Y_t$, where $X_i = \{x \in X | d(x) = \delta_i\}$, $Y_j = \{y \in Y | d(y) = \delta_j\}$. Then $s = t$ and for $x \in X_i$, $y \in Y_j$, $\{x, y\} \in E$ if and only if $i + j > t$.

Theorem 5.13. *[33] A graph is a difference graph if and only if it is bipartite and $2K_2$-free.*

Lemma 5.14. *Given a difference graph $G = (K, K', E)$ and a path S of length at least $t + 2$ there is a* B_1-ENPG *representation in which $\mathcal{P}_K$ and $\mathcal{P}_{K'}$ meet at S where t is the number of distinct nonzero degrees of K in G.*

Proof. Let $\delta_1 < \delta_2 < \ldots \delta_s$ (resp. $\sigma_1 < \sigma_2 < \ldots \sigma_t$) be the distinct nonzero degrees in K (resp in K'). By Theorem 5.12 we have $s = t$. Assume that the given path S has a length $t+2$. We construct the paths of $\mathcal{P}_K$ (resp. $\mathcal{P}_{K'}$) between the vertex $(0, -1)$ and $(0, i)$ (resp. between the vertex $(0, t - j)$ and $(0, t + 1)$) for $x \in K$ (resp. $x' \in K'$) such that $d(x) = \delta_i$ (resp. $(d(x') = \sigma_j)$). With this construction $\mathcal{P}_K, \mathcal{P}_{K'}$ meet at S between $(0, -1)$ and $(0, t + 1)$. By Theorem 5.12 two paths $P_x \in \mathcal{P}_K$, $P_{x'} \in \mathcal{P}_{K'}$ intersect if and only if $i+j>t$. Now assume desired length of S is bigger than $t+2$ then we subdivide the edges of S without changing the relations of paths in $\mathcal{P}$. $\square$

Lemma 5.15. *If two sets $\mathcal{P}_K, \mathcal{P}_{K'}$ of one-bend paths meet at a path S then G_B is a difference graph.*

Proof. Let u, v be the endpoints of S. Let $T = |E(S)| + 1$ and r_i (resp. l_j) be the endpoint of the path $P_i \in \mathcal{P}_K$ (resp. $P_j \in \mathcal{P}_{K'}$) among the internal vertices of S. Let $a_i = |E(p_S(u, r_i))|$ (resp. $a_j = -|E(p_S(l_j, v))|$) where $p_T(x, y)$ is the unique path between vertices x and y of a tree T. By definition, $|a_i| \leq |E(S)| < T$ for every $i \in K \cup K'$. Two paths $P_i \in \mathcal{P}_K$, $P_j \in \mathcal{P}_{K'}$ have an edge in common if and only if $|a_i - a_j| \geq |E(S)| + 1 = T$. Therefore, G_B is a difference graph. $\square$

Two representations $\langle H, \mathcal{P} \rangle$ and $\langle H', \mathcal{P}' \rangle$ are *bend-equivalent* if they are representations of the same graph G and there is a one to one correspondence between $\mathcal{P}$ and $\mathcal{P}'$ such that the corresponding paths have the same number of bends. We proceed with the following lemma that classifies all the B_1-ENPG representations of a co-bipartite graph into two types.

Lemma 5.16. *Let $G = (K, K', E)$ be a connected* B_1-ENPG *co-bipartite graph with a representation $\langle H, \mathcal{P} \rangle$. Then*

- *(i)* $|\mathcal{S}(K, K')| \in \{1, 2\}$, *and*
- *(ii) whenever* $|\mathcal{S}(K, K')| = 1$ *there is a bend-equivalent representation* $\langle H', \mathcal{P}' \rangle$ *such that* $\cup\mathcal{P}'$ *is a tree* T' *with* $\Delta(T') \leq 3$ *with at most two vertices of degree* 3.
- *(iii) whenever* $|\mathcal{S}(K, K')| = 2$ *the paths* $\cup\mathcal{P}_K$ *and* $\cup\mathcal{P}_{K'}$ *intersect as depicted in Figure 5.6b.*

Proof. By Corollary 4.10, $\cup\mathcal{P}_K$ and $\cup\mathcal{P}_{K'}$ are two paths with at most 2 bends each. Let e_K (resp. $e_{K'}$) be an arbitrary edge of $\cap\mathcal{P}_K$ (resp. $\cap\mathcal{P}_{K'}$). $\cup\mathcal{P}_K$ and $\cup\mathcal{P}_{K'}$ intersect in at least one edge, because otherwise G is not connected. Therefore, $|\mathcal{S}(K, K')| \geq 1$. We consider two disjoint cases:

- $|\mathcal{S}(K,K')| = 1$. In this case it is sufficient to prove ii). Let $T = \cup\mathcal{P}$ and S be the unique segment of $\mathcal{S}(K,K')$. Any vertex of degree at least 3 in T is an endpoint of S, therefore there are at most 2 such vertices. On the other hand an endpoint of S has degree at most 3. Therefore $\Delta(T) \leq 3$ and there are at most 2 vertices of degree 3 in T.

  If T does not contain a cycle then T is a tree and the claim holds. Assume $\cup\mathcal{P}$ contains a cycle C. We will modify the paths and end up with a representation where C does not exist and we will make sure that we keep the bends of the paths. If $\cup\mathcal{P}_K \subseteq \cup\mathcal{P}_{K'}$ then $C \subseteq \cup\mathcal{P}_{K'}$ implying that $\cup\mathcal{P}_{K'}$ contains 4 bends, a contradiction. Therefore there exist two edges $e_1 \in \mathcal{P}_K \setminus \mathcal{P}_{K'}$ and $e_2 \in \mathcal{P}_{K'} \setminus \mathcal{P}_K$. We can also assume that e_1 and e_2 are consecutive, since otherwise either $\mathcal{S}(K,K') > 1$ or $C \subseteq S$ but S may contain at most 2 bends.

  We subdivide e_1 and e_2 into e_1', e_1'' and e_2', e_2'' respectively. Assume e_1' (resp e_2') is closer to S in C than e_1'' (resp. e_2''). We remove all the edges of P_K (resp. $P_{K'}$) starting from e_1'' (resp. e_2'') to the tail of P_K (resp. $P_{K'}$) which is closer to e_1 (resp. e_2) to S. After this operation we do not lose any edge-intersection between any pair of paths since they do not belong to S. We also do not lose any splitting since any pair of splitting at e_1 or e_2 are now splitting at e_1' or e_2'. Let v be the common vertex of the consecutive edges e_1, e_2, v is not a bend since otherwise $\cup\mathcal{P}$ would have more than 4 bends. Therefore this new representation is bend equivalent to $\langle H, \mathcal{P}\rangle$.
- $|\mathcal{S}(K,K')| \geq 2$. We claim that $\cup\mathcal{S}(K,K')$ $(= \cup\mathcal{P}_K \cap \cup\mathcal{P}_{K'})$ contains only horizontal edges, or only vertical edges. Indeed, assume that there is a vertical edge e_V and a horizontal edge e_H in $\cup\mathcal{S}(K,K')$. We observe that there is a unique one bend path connecting e_V and e_H, and that any other connecting these edges contains at least three bends. Therefore, both $\cup\mathcal{P}_K$ and $\cup\mathcal{P}_{K'}$ contain this path. We conclude that e_V and e_H are in the same segment. As any other edge is either horizontal or vertical, we can proceed similarly for all the edges of $\cup\mathcal{S}(K,K')$ and prove that they all belong to the same segment, contradicting the fact that we have at least 2 segments. Assume without loss of generality that all the edges of $\cup\mathcal{S}(K,K')$ are vertical. Then every segment is a vertical path. No two segments

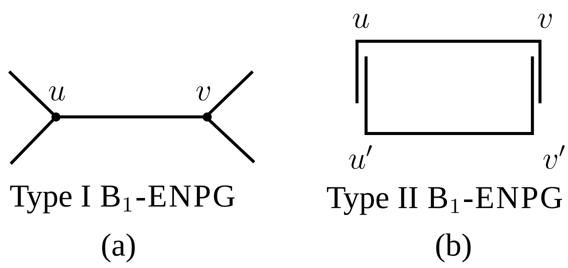

Figure 5.6. Two types of B_1-ENPG representation of connected co-bipartite graphs: (a) Type I: $|\mathcal{S}(K, K')| = 1$, $\cup\mathcal{P}$ is isomorphic to a tree T with $\Delta(T) \leq 3$ and at most two vertices u, v having degree 3, (b) Type II: $|\mathcal{S}(K, K')| = 2$, $\mathcal{P}_K$ (resp. $\mathcal{P}_{K'}$) has exactly two bend points u, v (resp. u', v').

can be on the same vertical line, because this will require at least one of $\cup\mathcal{P}_K$, $\cup\mathcal{P}_{K'}$ to contain four bends. Moreover, three vertical segments in distinct vertical lines imply that $\mathcal{P}_K$ and $\mathcal{P}_{K'}$ contain at least four bends each. Therefore, there are exactly 2 vertical segments and $\mathcal{P}_K$ (also $\mathcal{P}_{K'}$) has exactly two bends.
Let u, v (resp. u', v') be the bends of $\cup\mathcal{P}_K$ (resp. $\cup\mathcal{P}_{K'}$). Then $\mathcal{S}(K, K') = \{S_u, S_v\}$ where S_u (resp. S_v) is on the same vertical line as u and u' (resp. v and v'). Moreover e_K (resp. $e_{K'}$) is between u and v (resp. u' and v') since otherwise we would have paths crossing both u and v (resp. u' and v') and thus 2 bends. Now consider the situation where u and u' are on the same side of S_u on their common vertical line. Every path intersecting with S_u cross the same endpoint of S_u, implying that if a pair of paths from distinct cliques intersect at S_u, they split at this endpoint. As the same holds for the paths intersecting in S_v, we conclude that G is not connected, contradiction to our assumption. Therefore, u and u' (resp. v and v') are on different sides of S_u (resp. S_v), as depicted in Figure 5.6b.

□

Based on Lemma 5.16, a B_1-ENPG representation of a connected co-bipartite graph $G = (K, K', E)$ is *Type I* (resp. *Type II*) if $\mathcal{S}(K, K') = 1$ (resp. $\mathcal{S}(K, K') = 2$).

We proceed with the characterization of B_1-ENPG graphs having a Type II representation that turns out to be simpler than the characterization of the others.

Lemma 5.17. *A connected twin-free co-bipartite graph $G = C(K, K', E)$ has a Type II* B_1-ENPG *representation if and only if the bipartite graph $G_B = B(K, K', E)$ contains at most two non-trivial connected components each of which is a difference graph.*

Proof. ($\Rightarrow$) Let $\langle H, \mathcal{P} \rangle$ be a Type II B_1-ENPG representation of G and u, v (resp. u', v') be the bends of $\cup\mathcal{P}$ (resp. $\cup\mathcal{P}'$) as depicted in Figure 5.6b. For $x \in \{u, v\}$, let S_x be the segment contained in the path between x and x'. The paths of $\mathcal{P}$ not intersecting with any of S_u, S_v correspond to trivial connected components of G_B. There is at most one such path in $\mathcal{P}_K$ (resp. $\mathcal{P}_{K'}$) as G is twin-free.

Each one of the remaining paths intersects exactly one of S_u, S_v, as otherwise such a path would contain two bends. For $X \in \{K, K'\}$ and $x \in \{u, v\}$ let $\mathcal{P}_{X_x}$ be the paths of $\mathcal{P}_X$ intersecting S_x. Then $\mathcal{P}_{K_x}$ and $\mathcal{P}_{K'_x}$ meet at S_x. By Lemma 5.15, $G_B[K_x \cup K'_x]$ is a difference graph.

($\Leftarrow$) We construct a Type II representation for the maximal case, i.e. G_B contains exactly two trivial connected components and two non-trivial connected components. Let $w \in K$ and $w' \in K'$ be the two trivial connected components of G_B and $B(K_u, K'_u, E_u)$, $B(K_v, K'_v, E_v)$ be the two non-trivial connected components of G_B. We construct a rectangle as depicted in Figure 5.6b having vertical lines with $\max(\min(|K_u|, |K'_u|), \min(|K_v|, |K'_v|)) + 2$ edges, and horizontal lines with one edge $e_K = \{u, v\}$ and $e_{K'} = \{u', v'\}$. For $X \in \{K, K'\}$, and $x \in \{u, v\}$ the paths $\mathcal{P}_{X_x}$ start with e_X and enter segment S_x. The other endpoints of the paths will be in the segment S_x. Then, for $x \in \{u, v\}$, $\mathcal{P}_{K_x}$ and $\mathcal{P}_{K'_x}$ meet at S_x. Since $B(K_x, K'_x, E_x)$ is a difference graph, by Lemma 5.14, the endpoints can be determined such that $\mathcal{P}_{K_x} \cup \mathcal{P}_{K'_x}$ is a representation of $B(K_x, K'_x, E_x)$. P_w (resp. $P_{w'}$) consists of the edge e_K (resp. $e_{K'}$). It is easy to verify that this is a representation of G. $\square$

We proceed with the characterization of the B_1-ENPG graphs with a Type I characterization. For this purpose we resort to the following definitions from [34].

Let $G = B(V, V', E)$ be a bipartite graph and $M \subseteq V \cup V'$. A vertex $v \in V \setminus M$ (resp. $v \in V' \setminus M$) *distinguishes* M if it has a neighbour in $M \cap V'$ (resp. $M \cap V$) and a non-neighbour in $M \cap V'$ (resp. $M \cap V$). A nonempty subset M of $V \cup V'$ is a *bimodule* of G if no vertex distinguishes M. It follows from the definition that $V \cup V'$ is a bimodule of G, and so are all the singletons and all the pairs with exactly one vertex from V. These bimodules are the *trivial* bimodules of G.

A *zed* is a graph isomorphic to a P_4 or any induced subgraph of it. We note that a trivial bimodule is a zed.

Lemma 5.18. *A connected twin-free co-bipartite graph $G = C(K, K', E)$ has a Type I* B$_1$-ENPG *representation if and only if G contains a zed Z such that*

(i) Z is a bimodule of $G_B = B(K, K', E)$, and

(ii) $G_B \setminus Z$ is a difference graph.

Moreover, if Z is a minimal set of vertices that satisfies i),ii) and $G[Z]$ is a set of two isolated vertices, then for the unique segment S of $\mathcal{S}(K, K')$ the following hold

(i) S is contained in at least one of the paths of $\mathcal{P}_Z$,

(ii) the endpoints of S have degree 3 in $\cup\mathcal{P}$ and these endpoints constitutes split$(\cup\mathcal{P}_K, \cup\mathcal{P}_{K'})$.

Proof. ($\Rightarrow$) Let $\langle H, \mathcal{P}\rangle$ be a Type I B$_1$-ENPG representation of G. By Lemma 5.16, $|\mathcal{S}(K, K')| = 1$ and $\cup\mathcal{P}$ is a tree. Let u, v be the endpoints of the unique segment S of $\mathcal{S}(K, K')$. We consider the following disjoint cases

- $\{e_K, e_{K'}\} \nsubseteq E(S)$: Let without loss of generality $e_K \notin E(S)$ and u closer to e_K than v. Consider two paths $P_{x'}, P_{y'} \in \mathcal{P}_{K'}$ that cross u. We observe that these paths are indistinguishable by the paths of $\mathcal{P}_K$. Namely, every path of $\mathcal{P}_K$ either does not intersect any one of $P_{x'}, P_{y'}$, or intersects both and splits from both at u. Therefore the corresponding vertices x', y' are twins. As G is twin-free we conclude that there is at most one path of $\mathcal{P}_{K'}$ that crosses u. Using the same

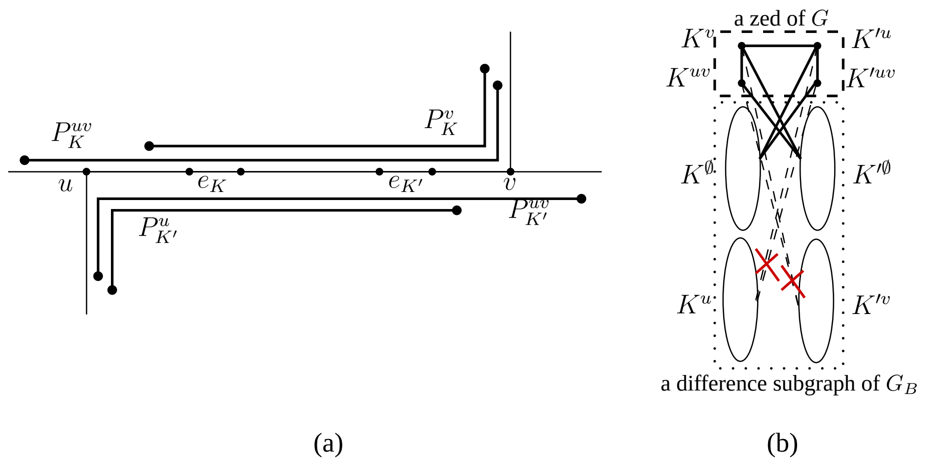

Figure 5.7. (a) Four special paths which are corresponding to a zed (b) The type of vertices and edge relations of a B_1-ENPG cobipartite graph having a Type I representation.

argument it can be shown that there is at most one path of $\mathcal{P}_K$ that crosses v. Let $\mathcal{P}_{Z'}$ be a set of these at most two paths. Namely, $\mathcal{P}_{Z'}$ consists of all the paths of $\mathcal{P}_{K'}$ crossing u and all the paths of $\mathcal{P}_K$ that cross v. We now observe that $\cup(\mathcal{P} \setminus \mathcal{P}_{Z'})$ is a path. Let S' be the sub-path of this path between the edges e_K and $e_{K'}$. The paths $\mathcal{P} \setminus \mathcal{P}_{Z'}$ meet at S'. Therefore, $G_B \setminus Z'$ is a difference graph. We note that the path $P_x \in \mathcal{P}_{K'}$ that crosses u is an isolated vertex of G_B, therefore for $Z = Z' \setminus \{x\}$ we have that $G_B \setminus Z$ is a difference graph too. Since $|Z| \leq 1$ we have: (i) Z is a trivial bimodule of G_B, (ii) Z is a zed, (iii) the second part of the claim holds vacuously.

- $\{e_K, e_{K'}\} \subseteq E(S)$: Assume without loss of generality that e_K is closer to u than $e_{K'}$, see Figure 5.7. Consider two paths $P_{x'}, P_{y'} \in \mathcal{P}_{K'}$ that cross u but not v. We observe that these paths are indistinguishable by the paths of $\mathcal{P}_K$. Therefore, the corresponding vertices are twins. As G is twin-free we conclude that there is at most one path $P_{K'}^u$ of $\mathcal{P}_{K'}$ that crosses u and does not cross v. Similarly there is at most one path $P_{K'}^{u,v}$ of $\mathcal{P}_{K'}$ that crosses both u and v, at most one path P_K^v of $\mathcal{P}_K$ that crosses v but does not cross u and at most one path $\mathcal{P}_K^{u,v}$ of $\mathcal{P}_K$ that crosses both u and v. Let $\mathcal{P}_Z$ be the set of these at most four paths. As in the previous case, after the removal of these paths we remain with a path and the $G_B \setminus Z$ is a difference graph. Assuming that all the four paths exist, it is easy to verify that their corresponding vertices constitute a zed where the only edge is between the vertices corresponding to P_K^v and $P_{K'}^u$. Therefore, Z is a zed. Finally, we observe that P_K^v and $P_K^{u,v}$ are distinguishable only by $P_{K'}^u \in \mathcal{P}_Z$. In other words they are indistinguishable by paths from $\mathcal{P}_{K'} \setminus \mathcal{P}_Z$. By symmetry, we conclude that Z is a bimodule of G_B. Finally we note that if $G_B[Z]$ consists of two vertices and none of the corresponding paths contains the segment S then these paths are $P_{K'}^u$ and P_K^v. But $P_{K'}^u \sim P_K^v$. Therefore, if $G_B[Z]$ consists of two isolated vertices then at least one of the corresponding paths contains S. If both paths contain S, then these paths are P_K^{uv} and $P_{K'}^{uv}$ and we have $\mathit{split}(\cup\mathcal{P}_K, \cup\mathcal{P}_{K'}) \supseteq \mathit{split}(P_K^{uv}, P_{K'}^{uv})$ as claimed. Otherwise, one of the paths does not contain S. Let, without loss of generality this path be $P_{K'}^u$. Then no path of $\mathcal{P}_{K'}$ crosses v. We conclude that $\cup(\mathcal{P} \setminus \{P_{K'}^u\})$ is a path. Contradicting the assumption that Z is a minimal set satisfying the claimed properties.

($\Leftarrow$) Given a zed Z of G satisfying the conditions of the lemma, we construct a Type I representation $\langle H, \mathcal{P} \rangle$ as follows. Without loss of generality we assume that Z is a P_4. Let $\ell = \min(|K|, |K'|) + 2$. The subgraph $G_B \setminus Z$ is a difference graph. Let $V(Z) = \{y, x, x', y'\}$ where $x, y \in K$, $x', y' \in K'$ and $\{x, x'\} \in E$. P_x (resp. P_y) is a path between $(0,0)$ (resp. $(-1,0)$) and $(\ell, 1)$ with a bend at $(\ell, 0)$. $P_{x'}$ (resp. $P_{y'}$) is a path between $(\ell, 0)$ (resp. $(\ell+1, 0)$) and $(0, -1)$ with a bend at $(0,0)$. It is easy to verify that this correctly represents Z. The representation of the difference graph $G_B \setminus Z$ is two sets of paths that meet at the line segment between $(0,0)$ and $(\ell, 0)$. By Lemma 5.14, the endpoints of the paths within this segment can be determined according to the difference graph $G_B \setminus Z$. The other endpoints of these paths are determined as follows. As Z is a bimodule we have $N_G(x) \setminus \{x'\} = N_G(y)$ and $N_G(x') \setminus \{x\} = N_G(y')$. The other endpoint of every path of $\mathcal{P}_{K' \cap N_G(y)}$ (resp. $\mathcal{P}_{K' \setminus N_G(y)}$) is $(\ell, 0)$ (resp. $(\ell+1, 0)$). The other endpoint of every path of $\mathcal{P}_{K \cap N_G(y')}$ (resp. $\mathcal{P}_{K \setminus N_G(y')}$) is $(0,0)$ (resp. $(-1, 0)$). $\square$

By Lemmata 5.17 and 5.18 we have

Theorem 5.19. *Let $G = C(K, K', E)$ be a connected, twin-free co-bipartite graph, and $G_B = C(K, K', E)$. G is* B_1-ENPG *if and only if at least one of the following holds:*

(i) G_B *contains at most two non-trivial connected components each of which is a difference graph.*

(ii) G *contains a zed* Z *that is a bimodule of* G_B *such that* $G_B \setminus Z$ *is a difference graph.*

Since all the properties mentioned in Theorem 5.19 can be tested in polynomial time we have

Corollary 5.20. B_1-ENPG *co-bipartite graphs can be recognized in polynomial time.*

5.4.2. Efficient Recognition Algorithm

In the sequel we describe an efficient implementation of the above idea.

Theorem 5.21. *Given a co-bipartite graph* $G = (K, K', E)$*, Algorithm 5.8 decides in time* $O(|K| + |K'| + |E|)$ *whether* G *is* B_1-ENPG.

Proof. The correctness of the algorithm follows from Observations 5.10, 5.11, Lemma 5.16 and from the correctness of isTypeI and isTypeII that we prove in the sequel.

Let $n = |K| + |K'|$, $m = |E|$. Let also $T_{diff}(n, m)$ be the running time of isDifference on a graph with n vertices and m edges. Similarly, let $T_{bm}(n, m)$ be the running time of findBimoduleZed that finds the minimum zed of G that is a bimodule of G_B and contains a given zed Z. All the twins of a graph can be removed in time $O(n+m)$ by constructing its modular decomposition tree [35] and then searching (near the leaves of the tree) modules consisting of two adjacent edges.

The correctness of isTypeI is based on Lemma 5.18. A zed Z of G that is a bimodule of G_B such that $G_B \setminus Z$ is a difference graph is termed as an *evidence* through this proof.

We show that given a twin-free co-bipartite graph G and $Z \subseteq V(G)$, isTypeI returns "YES" if and only if there exists an evidence $Z' \supseteq Z$. Moreover, we show that its running time is at most $5^{5-|Z|}(T_{diff}(n,m)+T_{bm}(n,m))$ when $|Z| \leq 4$ and constant otherwise. Since isTypeI is invoked initially with $Z=\emptyset$, by Lemma 5.18 this will imply that the algorithm is correct and its running time is $O(T_{diff}(n,m)+T_{bm}(n,m))$. We first observe that if Z is not a zed, then no superset of Z is a zed, and the algorithm returns correctly "NO" in constant time at line 7. Therefore, our claim is correct whenever Z is not a zed. We proceed by induction on $5-|Z|$. If $5-|Z|=0$, then Z is not a zed and the algorithm returns "NO" at constant time. In the sequel we assume that Z is a zed. In this case, isTypeI verifies at constant time that Z is a zed and proceeds to line 8 to find (in time $T_{bm}(n,m)$) the minimal bimodule Z' of G_B that contains Z and is a zed of G. We consider three cases according to the branching of isTypeI. We denote $\alpha(n,m) \stackrel{def}{=} T_{diff}(n,m)+T_{bm}(n,m)$.

- $\mathbf{Z'=Z}$ (i.e. Z is a bimodule of G_B), **and $\mathbf{G_B \setminus Z}$ is a difference graph:** isTypeI verifies at line 10 that $G_B \setminus Z$ is a difference graph. Finally it returns "YES" which is correct by Lemma 5.18 since Z is an evidence. The running time is $\alpha(n,m)$, and the result follows since $1 \leq 5^{5-|Z|}$.
- $\mathbf{Z'=Z}$ (i.e. Z is a bimodule of G_B), **but $\mathbf{G_B \setminus Z}$ is not a difference graph:** As $G_B \setminus Z$ is not a difference graph, there is a set $U \subseteq K \cup K' \setminus Z$ such that $G_B[U]$ is a $2K_2$. Every evidence $Z' \supseteq Z$ must contain at least one vertex of U because otherwise $G_B \setminus Z'$ contains $G_B[U]$ which is a $2K_2$. Therefore, the algorithm proceeds recursively by guessing each time a vertex $u \in U$. The algorithm returns "YES" if and only if one of the guesses succeeds. Then, the total running time is at most $\alpha(n,m) + 4 \cdot 5^{5-(|Z|+1)}\alpha(n,m) = \left(1+\frac{4}{5}\cdot 5^{5-|Z|}\right)\alpha(n,m)$. Since $1 \leq 5^{4-|Z|} = \frac{1}{5}5^{5-|Z|}$ we conclude that the running time is at most $5^{5-|Z|}\alpha(n,m)$.
- $\mathbf{Z' \neq Z}$ (i.e. Z is not a bimodule of G_B): If Z' exists, by definition, any evidence that contains Z has to contain Z'. Therefore, isTypeI(G,Z') is invoked and its result is returned. Otherwise, no evidence contains Z and "NO" is returned. The running time of isTypeI is $T_{bm}(n,m) + 5^{5-|Z'|}\alpha(n,m) < 5^{5-|Z|}\alpha(n,m)$.

We conclude that the running time of isTypeI is $O(T_{diff}(n,m)+T_{bm}(n,m))$.

The correctness of isTypeII follows directly from Lemma 5.17. The connected components of G_B can be calculated in time $O(n+m)$ using breadth first search. Therefore, the running time of isTypeII is $O(T_{diff}(n,m))$. Summarizing, the running time of Algorithm 5.8 is $O(T_{diff}(n,m) + T_{bm}(n,m))$.

$T_{diff}(m,n)$ is $O(m+n)$ [36]. We now show the correctness of findBimoduleZed and calculate its running time $T_{bm}(m,n)$.

- $Z = \emptyset$ or Z is a singleton or Z is a pair of vertices of $K \times K'$. By definition, Z is both a zed of G and a bimodule of G_B. Therefore, Z is the minimal bimodule of G_B that is a zed of G, and contains Z. The algorithm returns Z in constant time.
- Without loss of generality $Z \cap K$ contains at least two vertices u_1, u_2. We note that $Z \cap K = \{u_1, u_2\}$, because otherwise Z contains a K_3 contradicting the fact that it is a zed. Let Z' be the superset of Z obtained by adding to it all the vertices that distinguish u_1 and u_2. In other words, $Z' \overset{def}{=} (N_{G_B}(u_1) \triangle N_{G_B}(u_2)) \cup Z$. We note that Z' can be calculated in time $O(|K'|)$. If Z' is not a zed we can return at constant time that no superset of Z is both a zed of G and a bimodule of G_B. Assume Z' is a zed, let $U' = Z' \cap K'$. If $|U'| \leq 1$ then Z' is the minimal subset that contains Z and is both a zed of G and a bimodule of G_B. If $|U'| \geq 2$ then Z' is not a zed. Assume $|U'| = 2$ and let $U' = \{u'_1, u'_2\}$. We now add to Z', in time $O(K)$, the set of vertices of K that distinguish U' to get Z''. If $Z'' = Z'$ then Z' is the minimal superset of Z that is both a zed of G and a bimodule of G_B. Otherwise every bimodule that contains Z' has to contain also Z''. However $|Z'' \cap K| > |Z \cap K| = 2$, implying that Z'' contains a K_3, is thus not a zed. In this case, we conclude that there is no superset of Z as required.

We note that all the steps can be executed at time at most $O(|K| + |K'|) = O(n)$, i.e. $T_{bm}(m,n) = O(m,n)$. Therefore, the running time of the algorithm is $O(T_{diff}(n,m) + T_{bm}(n,m)) = O(n+m)$. □

We conclude with an interesting remark, pointing to a fundamental difference

Require: A co-bipartite graph $G = (K, K', E)$
  if G is not connected **then return** "YES"
  Make G twin-free using modular decomposition.
  if isTypeI$(G, \emptyset)$ or isTypeII(G) **then return** "YES".
  return "NO".
  function isTypeI$(G = C(K, K', E), Z)$
Require: G is connected, twin-free, $Z \subseteq V(G)$
Ensure: returns whether there is an evidence $Z' \supseteq Z$ for G being Type I
    $G_B \leftarrow B(K, K', E)$.
    if $G[Z]$ is not a zed **then return** "NO".
    $Z' \leftarrow$ findBimoduleZed(G, Z).
    if $Z' = Z$ **then** $\triangleright$ Z is a zed of G and also a bimodule of G_B
      if isDifference$(G_B \setminus Z)$ **then return** "YES".
      Let $U \subseteq (K \cup K') \setminus Z$ such that $G_B[U]$ is a $2K_2$.
      for $u \in U$ **do**
        if isTypeI$(G, Z \cup \{u\})$ **then return** "YES".
      return "NO".
    else
      if $Z' \neq NULL$ **then return** isTypeI(G, Z').
      else return "NO".
  function isTypeII$(G = C(K, K', E))$
Require: G is connected, twin-free
    $G_B \leftarrow B(K, K', E)$. Remove all isolated vertices from G_B.
    Calculate the connected components $G_1, \ldots, G_k$ of G_B.
    if $k > 2$ **then return** "NO".
    if isDifference(G_1) and isDifference(G_2) **then return** "YES".
    return "NO".
  function findBimoduleZed$(G = C(K, K', E), Z)$
Require: G is twin-free, Z is a zed of G
Ensure: Returns the minimum superset of Z (a zed of G) and a bimodule of G_B
    if $|Z \cap K| \leq 1$ and $|Z \cap K'| \leq 1$ **then return** Z.
    Let without loss of generality $Z \cap K = \{u_1, u_2\}$.
    $Z' \leftarrow (N_{G_B}(u_1) \triangle N_{G_B}(u_2)) \cup Z$.
    if Z' is not a zed **then return** NULL.
    $U' \leftarrow Z' \cap K'$.
    if $|U'| \leq 1$ **then return** Z'.
    Let without loss of generality $U' = \{u'_1, u'_2\}$.
    $Z'' \leftarrow (N_{G_B}(u'_1) \triangle N_{G_B}(u'_2)) \cup Z'$.
    if $Z'' = Z'$ **then return** Z'
    else return NULL.
  function isDifference(G) [36]
Require: G is bipartite
Ensure: Returns "YES" if G is a difference graph and a $2K_2$ of G otherwise.

Figure 5.8. B_1-ENPG $\cap$ Co-bipartite Recognition Algorithm.

between EPG and ENPG graphs. A graph is B_k-EPG if it has a EPG representation $\langle H, \mathcal{P} \rangle$ such that every path has at most k bends. It is known that given a B_k-EPG representation it is always possible to modify the paths such that every path has exactly k bends. The following corollary shows that this does not hold for B_k-ENPG graphs.

Proposition 5.22. *Every* B_1-ENPG *representation of a graph* $G = C(K, K', E)$ *such that* $G_B = B(K, K', E)$ *is isomorphic to* $3K_2$ *contains at least one path with zero bend.*

Proof. Let $\langle H, \mathcal{P} \rangle$ be a representation of G. Since G_B has three non-trivial connected components, by Lemma 5.17, $\langle H, \mathcal{P} \rangle$ is a Type I representation. By Theorem 5.13, G_B is not a difference graph since it contains $2K_2$. Let Z be a zed of G and a bimodule of G_B such that $G_B \setminus Z$ is a difference graph. If Z consists of two vertices of K (or K'), then these two vertices are twins, however G is twin-free. Therefore Z contains two vertices $x \in K$, $y' \in K'$. Moreover, without loss of generality, $\{x, y'\} \notin E$ as otherwise $G_B \setminus Z$ contains a $2K_2$. We now observe that $Z = \{x, y'\}$ is a zed of G, a bimodule of G_B and $G_B \setminus Z$ does not contain a $2K_2$. Let y and x' be the unique neighbors in G_B of x and y' respectively. Since $G[Z]$ is a set of two isolated vertices, by Lemma 5.18, there are two split points of $\cup\mathcal{P}_K$ and $\cup\mathcal{P}_{K'}$, say u and v. By the same lemma, P_x or $P_{y'}$ crosses u, v. Let P_x be such a path. Then $P_{x'}$ does not cross neither u nor v. Moreover in any representation of G there is no bend point between two split points otherwise P_x would have more than one bend. Therefore $P_{x'}$ is a path with zero bend. $\square$

6. CONCLUSIONS

In this thesis, we introduced and studied a new family of graphs, called the graphs of Edge- Intersecting Non-Splitting Paths (ENP). These graphs enable us to model optimization problems arising in telecommunication networks. We do not focus on solving this specific application but instead on establishing results characterizing various properties of ENP graphs in order to use them in the design of efficient algorithms. This work contains non-trivial results about this new graph class that opens a wide field of research. Many interesting questions are still open and the "split" condition introduced in this work is not limited to optical networks but enables new applications to be modeled as a graph problem.

In Chapter 3, we start with a fairly natural case where the host graph is a tree, namely ENPT graphs. We follow mainly the work of Golumbic and Jamison's [22] for a very related graph class EPT. Cycles are simple yet one of the fundamental graph structures. EPT representation of cycles have a unique and simple characterization [22]. It turns out that this is not the case for ENPT graphs. There are many non-equivalent ENPT representations for a given cycle. To make the problem tractable, we assume that the EPT graph is also given as an input. We propose a new problem; given a pair of graphs (G, C) where G is an arbitrary graph and C is an Hamiltonian cycle of G, is there any representation $< T, \mathcal{P} >$ such that $\textsc{Ept}(\mathcal{P})$ is isomorphic to G and $\textsc{Enpt}(\mathcal{P})$ is isomorphic to C. We show that this problem is NP-complete in general, however for a special case (which is a restriction on the representations) we propose an algorithm which decides (and constructs a representation) in polynomial-time. As a by-product, a family of non-ENPT graphs is presented.

In Chapter 4, we consider the general case where the host graph can be an arbitrary graph. Although the Edge Intersection Graphs of Paths in an arbitrary graph includes all graphs, we show that this is not true for ENP. We also show that the class ENP coincides with the family of graphs of Edge-Intersecting and Non-Splitting Paths in a Grid (ENPG). Following similar studies for EPG graph class, we study the implications of restricting the number of bends in the grid, of the individual paths. We show that restricting the number of bends also restricts the graph class. More concretely, by restricting the number of bends one gets an infinite sequence of classes such that every class is properly included in the next one. In addition, we show that one bend ENPG graphs are properly included in two bend ENPG graphs.

In Chapter 5, we show that trees and cycles are one bend ENPG graphs, and characterize the split graphs and co-bipartite graphs that are one bend ENPG. We prove that the recognition problem of one bend ENPG graphs is NP-complete even in a very restricted subfamily of split graphs. Last, we provide a linear time recognition algorithm for one bend ENPG co-bipartite graphs.

We now summarize open questions and research directions presented in the previous chapters.

It is known that EPT recognition is NP-complete. In this work, we show that it remains NP-complete even if we label edges "splitting" in case of the corresponding paths split in some representation. The main difficulty originates from deciding a given clique whether it is represented by an edge clique or a claw clique. The complexity of the recognition problem when this information is provided by an oracle is open. ENPT recognition in general is also open. We showed that pair recognition is NP-complete however this result does not imply that ENPT recognition is NP-complete. Given an ENPT graph we have the flexibility to choose EPT edge in which case there could be a polynomial time algorithm. Since the only known forbidden subgraphs of ENPT are also forbidden for EPT, to answer this question we need more forbidden structures of ENPT.

Recall that B_1-ENPG split recognition is NP-complete. By following a similar research direction as cycles, another interesting research direction is to investigate the complexity of (G, S) recognition where (i) G and S are defined on the same vertex set (ii) S is a split graph (3) $\textsc{Ept}(\mathcal{P}) = G$ and $\textsc{Enpt}(\mathcal{P}) = S$.

The maximum clique problem in ENPT graphs can be solved in polynomial time using a clique enumeration algorithm even if the representation is not available. Investigating a more efficient maximum clique algorithm can be an interesting research direction. The time complexity of other important graph problems such as maximum independent set and minimum vertex coloring are still open.

Even though Theorem 5.19 gives a characterization of B_1-ENPG co-bipartite graphs, a forbidden subgraph characterization is unkown. We have a list of forbidden structures, however a complete characterization is work in progress. Theorems 5.3 and 5.19 give characterizations for B_1-ENPG split and cobipartite graphs respectively. Maximum cut problem is NP-complete in split and co-bipartite graphs however using these structural properties one might be able to devise polynomial time algorithms for respectively B_1-ENPG split and co-bipartite graphs. We have some partial results for B_1-ENPG co-bipartite graphs [29].

REFERENCES

1. Chlamtac, I., A. Ganz and G. Karmi, “Lightpath Communications: An Approach to High Bandwidth Optical WAN’s”, *IEEE Transactions on Communications*, Vol. 40, No. 7, pp. 1171–1182, 1992.

2. Ramaswami, R., “Multi-wavelength lightwave networks for computer communication”, *IEEE Communications Magazine*, Vol. 31, pp. 78–88, 1993.

3. Du, D. H. C. and R. J. Vetter, “Distributed computing with high-speed optical networks”, *Proceeding of IEEE Computer*, Vol. 26, pp. 8–18, 1993.

4. Gerstel, O., R. Ramaswami and G. Sasaki, “Cost effective traffic grooming in WDM rings”, *INFOCOM’98, Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies*, 1998.

5. Flammini, M., G. Monaco, L. Moscardelli, M. Shalom and S. Zaks, “Optimizing Regenerator Cost in Traffic Grooming”, *Theoretical Computer Science*, Vol. 412, No. 52, pp. 7109–7121, 2011.

6. Golumbic, M. C. and R. E. Jamison, “Edge and vertex intersection of paths in a tree”, *Discrete Mathematics*, Vol. 55, No. 2, pp. 151–159, 1985.

7. Tsukiyama, S., M. Ide, H. Ariyoshi and I. Shirakawa, “A New Algorithm for Generating All the Maximal Independent Sets”, *SIAM Journal on Computing*, Vol. 6, No. 3, pp. 505–517, 1977.

8. Tarjan, R. E., “Decomposition by clique separators”, *Discrete Mathematics*, Vol. 55, No. 2, pp. 221 – 232, 1985.

9. Golumbic, M. C., M. Lipshteyn and M. Stern, “Equivalences and the complete hierarchy of intersection graphs of paths in a tree”, *Discrete Appl. Math.*, Vol. 156,

pp. 3203–3215, 2008.

10. Golumbic, M. C., M. Lipshteyn and M. Stern, "Representing edge intersection graphs of paths on degree 4 trees", *Discrete Mathematics*, Vol. 308, No. 8, pp. 1381–1387, 2008.

11. Golumbic, M. C., M. Lipshteyn and M. Stern, "The k-edge intersection graphs of paths in a tree", *Discrete Appl. Math.*, Vol. 156, pp. 451–461, 2008.

12. Biedl, T. C. and M. Stern, "On edge-intersection graphs of k-bend paths in grids", *Discrete Mathematics & Theoretical Computer Science*, Vol. 12, No. 1, pp. 1–12, 2010.

13. Golumbic, M. C., M. Lipshteyn and M. Stern, "Edge intersection graphs of single bend paths on a grid", *Networks*, Vol. 54, No. 3, pp. 130–138, 2009.

14. Biedl, T. C. and M. Stern, "On edge-intersection graphs of k-bend paths in grids", *Discrete Mathematics & Theoretical Computer Science*, Vol. 12, No. 1, pp. 1–12, 2010.

15. Heldt, D., K. Knauer and T. Ueckerdt, "Edge-intersection graphs of grid paths: The bend-number", *Discrete Applied Mathematics*, Vol. 167, No. 0, pp. 144–162, 2014.

16. Asinowski, A. and B. Ries, "Some properties of edge intersection graphs of single-bend paths on a grid", *Discrete Mathematics*, Vol. 312, No. 2, pp. 427–440, 2012.

17. Epstein, D., M. Golumbic and G. Morgenstern, "Approximation algorithms for B1-EPG graphs", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 8037 LNCS, pp. 328–340, 2013.

18. Golumbic, M. C., *Algorithmic Graph Theory and Perfect Graphs (Annals of Dis-*

crete Mathematics, Vol 57), North-Holland Publishing Co., Amsterdam, The Netherlands, 2004.

19. Boyacı, A., T. Ekim, M. Shalom and S. Zaks, "Graphs of Edge-Intersecting Non-Splitting Paths in a Tree: Towards Hole Representations", *The Proceedings of the 39th International Workshop on Graph-Theoretic Concepts in Computer Science (WG), Luebeck, Germany*, pp. 115–126, 2013.

20. Boyaci, A., T. Ekim, M. Shalom and S. Zaks, "Graphs of Edge-Intersecting Non-Splitting Paths in a Tree: Towards Hole Representations-Part I", *CoRR*, Vol. abs/1309.2898, 2013.

21. Boyaci, A., T. Ekim, M. Shalom and S. Zaks, "Graphs of Edge-Intersecting Non-Splitting Paths in a Tree: Representations of Holes-Part II", *CoRR*, Vol. abs/1309.6471, 2013.

22. Golumbic, M. C. and R. E. Jamison, "The edge intersection graphs of paths in a tree", *Journal of Combinatorial Theory, Series B*, Vol. 38, No. 1, pp. 8–22, 1985.

23. Chartrand, G. and F. Harary, "Planar permutation graphs", *Annales de l'institut Henri Poincaré (B) Probabilités et Statistiques*, Vol. 3, No. 4, pp. 433–438, 1967.

24. Hajnal, A. and J. Surányi, "Über die Auflösung von Graphen in vollständige Teilgraphen", *Ann. Univ. Sci. Budapest Eötrös Sect. Math.*, Vol. 1, pp. 113–121, 1958.

25. Tucker, A., "Characterizing circular-arc graphs", *Bulletin of the American Mathematical Society*, Vol. 76, No. 6, pp. 1257–1260, 1970.

26. Boyacı, A., T. Ekim, M. Shalom and S. Zaks, "Graphs of Edge-Intersecting and Non-Splitting Paths", *The proceedings of the 15th Italian Conference on Theoretical Computer Science (ICTCS), Perugia, Italy*, pp. 45–58, 2014.

27. Asinowski, A. and A. H. Suk., "Edge intersection graphs of a system of paths in a

grid", *Discrete Applied Mathematics*, Vol. 157, pp. 3174–3180, 2009.

28. Yanpei, L., A. Morgana and B. Simeone, "General theoretical results on rectilinear embedability of graphs", *Acta Mathematicae Applicatae Sinica*, Vol. 7, p. 187192, 1991.

29. Boyaci, A., T. Ekim and M. Shalom, "The Maximum Cut Problem in Co-bipartite Chain Graphs", *CoRR*, Vol. abs/1504.03666, 2015.

30. Garey, M., D. Johnson and R. Tarjan, "The Planar Hamiltonian Circuit Problem is NP-Complete", *SIAM Journal on Computing*, Vol. 5, No. 4, pp. 704–714, 1976.

31. Kotzig, A., "Aus der Theorie der endlichen regulären Graphen dritten und vierten Grades", *Časopis Pěst. Mat.*, Vol. 3, No. 82, pp. 76–92, 1957.

32. Bodlaender, H. L. and K. Jansen, "On The Complexity Of The Maximum Cut Problem", *Nordic Journal of Computing*, 2000.

33. Hammer, P. L., U. N. Peled and X. Sun, "Difference Graphs", *Discrete Appl. Math.*, Vol. 28, No. 1, pp. 35–44, 1990.

34. Fouquet, J.-L., M. Habib, F. de Montgolfier and J.-M. Vanherpe, "Bimodular Decomposition of Bipartite Graphs", *30th International Workshop on Graph Theoretic Concepts in Computer Science, WG*, pp. 117–128, 2004.

35. Tedder, M., D. Corneil, M. Habib and C. Paul, "Simpler Linear-Time Modular Decomposition Via Recursive Factorizing Permutations", *Proc. 35th International Colloquium on Automata, Languages and Programming (ICALP 2008)*, pp. 634–645, Springer-Verlag, 2008.

36. Heggernes, P. and D. Kratsch, "Linear-time Certifying Recognition Algorithms and Forbidden Induced Subgraphs", *Nordic J. of Computing*, Vol. 14, No. 1, pp. 87–108, 2007.