Second-order Birkhoff Polytope and the Problem of Graph Isomorphism Detection

A Thesis Submitted
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

by

Pawan Kumar Aurora

to the

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

June, 2015

CERTIFICATE

It is certified that the work contained in the thesis entitled "*Second-order Birkhoff Polytope and the Problem of Graph Isomorphism Detection*", by "*Pawan Kumar Aurora*", has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

(Dr. Shashank K. Mehta)
Professor,
Department of Computer Science and Engineering
Indian Institute of Technology Kanpur

June, 2015

Synopsis

Graph Isomorphism (GI) problem is a classic problem in the theory of computing. Given two graphs G, G' on n vertices each, GI problem is to decide if G, G' are isomorphic or not. The problem remains elusive for general graphs in the sense that neither a polynomial time algorithm is known for it nor is it established that this problem is NP-complete. Here it must be acknowledged that polynomial time algorithms are known for the GI problem on special classes of graphs. The best known algorithm for general graphs is moderately exponential.

In this research work we attempt to devise a polynomial time algorithm for GI for general graphs. Most of the attempts for designing a polynomial time algorithm for this problem have been based on combinatorial, group theoretic or graph theoretic techniques or a combination of these. A small number of attempts have also been based on linear programming. In this work we revisit linear programming approach to design an algorithm for the problem.

If we label the vertices of both the input graphs by $1, \ldots, n$, then each isomorphism is a permutation σ such that vertices i and j are adjacent in G if and only if the vertices $\sigma(i)$ and $\sigma(j)$ are adjacent in G'. The earlier LP based approaches use the commutation relation $AP = PA'$ satisfied by isomorphic pairs of graphs where A and A' denote the adjacency matrices of the two graphs and P denotes the permutation matrix corresponding to an isomorphism between the two graphs. Replacing P by a doubly stochastic variable matrix Y gives a linear program whose zero-one solutions are the isomorphisms between the two graphs. The feasible region of this linear program is known as the standard GI polytope or Tinhofer polytope (let's say $\mathcal{T}$). It has been shown that GI can be decided for certain classes of graphs in a constant number of rounds of the Sherali-Adams (SA) lift and project method starting with the polytope $\mathcal{T}$.

Consider the polytope $\mathcal{P}$ obtained by one SA lift step applied to the Birkhoff polytope, $\mathcal{B}$. It can be shown that there is a one to one correspondence between the integer points of $\mathcal{P}$ and the integer points of $\mathcal{B}$, with each integer point corresponding to a permutation. We refer to the convex hull of integer points in $\mathcal{P}$ as the *second-order Birkhoff polytope* and denote it by $\mathcal{B}^{[2]}$. In the literature this polytope is studied in the context of the Quadratic Assignment Problem and hence is also called the QAP-polytope. We show that $\mathcal{B}^{[2]}$ is full-dimensional in the affine plane of polytope $\mathcal{P}$, i.e., the two polytopes have the same dimension.

Adding graph based conditions, also referred to as the edge/non-edge conditions, to the linear description of polytope $\mathcal{P}$, we get our linear program, LP-GI. The feasible region of this linear program contains exactly those integer points (permutations) which correspond to the isomorphisms between the given pair of graphs. It can be shown that the two graphs are isomorphic if and only the feasible region intersects with $\mathcal{B}^{[2]}$. In the case of non-isomorphic pairs of graphs this fact implies that the feasible region is either empty or confined to $\mathcal{P} \setminus \mathcal{B}^{[2]}$. We present an exact algorithm to determine if the feasible region is contained in $\mathcal{P} \setminus \mathcal{B}^{[2]}$ (i.e., deciding non-isomorphism). The most significant result of this thesis is that under a reasonable assumption this algorithm decides non-isomorphism in polynomial time. The said assumption is related to the facial structure of $\mathcal{B}^{[2]}$ which is discussed next.

The non-negativity constraints in the linear program LP-GI define all the facets of polytope $\mathcal{P}$. Each of these constraints also define facets of polytope $\mathcal{B}^{[2]}$. We refer to these as the *trivial* facets of $\mathcal{B}^{[2]}$. There are other facets of the polytope which we will refer to as non-trivial facets. Exponentially many non-trivial facets of $\mathcal{B}^{[2]}$ are known in the QAP literature and in this thesis we identify exponentially many additional facets. We give three general inequalities which define three classes of (exponentially many) supporting planes of $\mathcal{B}^{[2]}$. These supporting planes define faces of the polytope including all its known facets. We also define a partial ordering on these supporting planes/inequalities and show that no minimal inequality is ever violated by any solution of the LP, irrespective of whether the graphs are isomorphic or not.

We analyze our exact algorithm for those cases when each point in the feasible region of LP-GI, for a pair of non-isomorphic graphs, violates one of the above mentioned inequalities. In such case clearly each point violates at least one minimal inequality (an inequality that it violates but does not violate any lower inequality in the ordering). If there exists a single inequality which is a minimal violated inequality for all points in the feasible region, then we show that the algorithm terminates in polynomial time. We perform several experiments with strongly regular graphs and CFI-graphs and report the results. In every non-isomorphic instance in these experiments we find that the feasible region is *zero-one reducible*, a property which ensures that the algorithm takes polynomial time to detect non-isomorphism.

We also consider the general case when no single minimal violated inequality exists for every point in the feasible region. We modify our exact algorithm to handle the general case efficiently. If k is the minimal number of regions into which the feasible region can be divided such that each region has a single minimal violated inequality then the modified algorithm runs in time exponential in k. We believe that the value of k should be small.

We also investigate if all the non-trivial facets of $\mathcal{B}^{[2]}$ are discovered or more are yet to be found out. We develop a single generic inequality such that all the known facets are its instances and prove that there must be at least one facet of the polytope which is not an instance of this generic equation, implying that all facets of the polytope are not yet known.

Finally, we restrict the feasible region of LP-GI to the cone of positive semidefinite matrices and observe that the resulting semidefinite program is the Lovász Theta function of a product of the input graphs. We also perform experiments using this formulation and find that the algorithm converges in fewer iterations than with the linear program, as should be expected.

Acknowledgements

I am forever indebted to my advisor Prof. Shashank K. Mehta for making this thesis possible. Most of the work in this thesis is the outcome of several hours of discussions that we have had over the past five years.

I would like to thank Prof. Arunava Banerjee at UFL for teaching with such passion and enthusiasm that sparked my interest in algorithms and theoretical computer science.

I would like to express my gratitude and thanks to my parents for being supportive of my decision to pursue doctoral studies and for sharing several of my responsibilites during the past several years so as to let me focus on my research.

Last but not the least I would like to thank my wife Sonal for bearing with me all these years. Your support at difficult times kept me going.

To Mummy, Papa, Sonal, Kartikeya and Kritika

Contents

List of Figures

Chapter 1

Introduction

1.1 The Graph Isomorphism Problem

The graph isomorphism problem (GI) is a well-studied computational problem; listed as an open problem in (Kar72) and (GJ79). Formally, given two graphs G_1 and G_2 on n vertices each, GI is a decision problem that asks if there exists a bijection $\sigma : V(G_1) \to V(G_2)$ such that $\{u, v\} \in E(G_1)$ iff $\{\sigma(u), \sigma(v)\} \in E(G_2)$. Each such bijection is called an isomorphism between G_1, G_2. Without loss of generality, we assume that the vertices in both the graphs are labeled by integers $1, \ldots, n$. Hence $V(G_1) = V(G_2) = [n]$ and each bijection is a permutation of $1, \ldots, n$. Apart from its practical applications in chemical identification (Jr.65), scene analysis (ABB+73) and construction and enumeration of combinatorial configurations (CM78), what makes this problem interesting is the fact that it is not known to be NP-complete (AT05) nor is there an algorithm known that can solve it in polynomial time. In fact, if GI were NP-complete then the "polynomial time hierarchy", a hierarchy of complexity classes between P and PSPACE, would collapse to its second level (BHZ87; Sch88), an unlikely scenario. Moreover, the problem of counting the number of isomorphisms between the input pair of graphs is known to be polynomial-time equivalent to GI itself (Mat79), another unlikely scenario since for all known NP-complete problems the counting version seems to be much harder. The fastest known graph isomorphism algorithm has running time $2^{O(\sqrt{n \log n})}$ (BL83; ZKT82; Bab81). However, polynomial time algorithms are known for special graph classes: trees (AHU74; CB81), planar graphs (HT72; HW74), bounded genus

(FM80; Mil80), bounded eigenvalue multiplicity (BGM82), bounded degree (Luk82), graphs with excluded minors (Pon88), bounded tree width (Bod90), interval graphs (LB79; Kle96; KKLV10), graphs with excluded topological subgraphs (GM12). It may be noted here that there are certain graph classes for which the problem is as hard as the general problem (GI-complete). These include bipartite graphs, chordal graphs, rectangle intersection graphs (Ueh08), graphs of bounded degeneracy and graphs of bounded expansion. At the same time, there are softwares, the leading one called *Nauty* (MP14), based on heuristics that can solve the problem fairly well in practice for almost all graphs.

A problem closely related to the graph isomorphism problem is the problem of finding a canonical or standard form of a class of isomorphic graphs (also known as the graph canonization problem). Intuitively, it is a labeling procedure which labels each graph uniquely. Then two graphs are isomorphic if and only if their canonical forms are identical.

All the known algorithms for GI employ one or more of three broad approaches: combinatorial, graph theoretic, and group theoretic. A polyhedral approach applies quite naturally to the graph isomorphism problem. Recently there has been a renewed interest in this approach. In the following sections we briefly review the major results and algorithms based on these four approaches.

1.1.1 Combinatorial Approach to Graph Isomorphism

Combinatorial algorithms for GI are generic algorithms that do not use the properties of specific graph classes. Most combinatorial algorithms for graph isomorphism attempt to find a canonical labeling. Two techniques are proposed to this end. First one is called refinement. It treats two vertices with the same labels as potentially isomorphic (belonging to the same orbit). One begins with uniform labeling for all vertices. In each step each class (set of vertices with same label) is split into multiple classes. For example if two vertices with same labels have different number of neighbors in various classes, then these vertices are themselves placed in different classes in the subsequent refinement step. The second technique is called individualization. Given a set of vertices with the same labels, we choose a subset and force unique labels on them. Then proceed with refinement. If it leads to a unique labeling for each vertex then canonization task is complete. The time complexity of solving GI using this technique is exponential in the size of the subset on which unique labels were forced. The popular software for finding automorphism groups of a graph, called *nauty*, uses the techniques of individualization and refinement.

The process of labeling can alternatively be viewed as that of dividing the vertices into equivalence classes that are closed under automorphisms.

The most basic algorithm in this class is known as the classic vertex classification (C-V-C) algorithm (Mal14). A single graph is obtained by taking the disjoint union of G_1, G_2. Then the following procedure is applied to the resulting graph. Initially all vertices are assigned the same label, i.e., all the vertices belong to a single class. In subsequent steps, also known as the refinement steps, two vertices belonging to a class, continue to remain in the same class in the following step, if they have the same number of neighbors in each class. In particular, after the first refinement step two vertices continue in the same class if and only if they have the same number of neighbors (or degree) in the graph, since initially there is only one class. The refinement process stops when the classes have stabilized i.e., a refinement step does not result in any new classification. It can be shown that a stable classification is reached in at most n refinement steps. If there exists a class in the stable classification that has different number of vertices from the two graphs, then it can be concluded that these graphs are not isomorphic to each other. However, the converse need not be true, i.e., all classes in a stable classification having the same number of vertices from both the graphs does not guarantee that the graphs are isomorphic except when both graphs have exactly one vertex in each class in which case the graphs have only the trivial automorphism (or are rigid graphs).

In (BK79), the authors show that the application of only two refinement steps of the above procedure results in a canonical labeling of random graphs with high probability. Immerman and Lander show in (IL90) that a stable classification can be used to solve GI in polynomial time for the case of all trees. However, the C-V-C algorithm fails to even start in the case of regular graphs.

In order to deal with more difficult graphs Weisfeiler and Lehman proposed k-WL algorithm, (WL68). This is a generalization of the vertex classification method described above. In this case all k-tuples of vertices are considered. So the C-V-C method can be considered as 1-WL or 1-dimensional WL. The procedure starts with an initial classification having two k-tuples $(u_1, \ldots, u_k), (v_1, \ldots, v_k)$ in the same class if $u_i \to v_i$ for $i = 1, \ldots, k$ is an isomorphism between the induced graphs on $\{u_1, \ldots, u_k\}$ and $\{v_1, \ldots, v_k\}$.

Just as a vertex in a graph has a single set of neighbors, a k-tuple of vertices has k sets of neighbors. The i-th set of neighbors of a k-tuple $(u_1, \ldots, u_k)$ is the set of all k-tuples obtained by replacing u_i with each vertex in the graph (other than u_i). So in the refinement step two tuples continue in the same partition if they have the same number of i^{th} neighbors in each partition for all i. Stable partition (or classification) is reached after at most n^k steps. Kucera shows in (Kuc87) that 2-WL can decide isomorphism almost surely on random regular graphs. This follows an earlier result of (Bol82) on distinguishing almost all pairs of regular graphs. There exists a fixed constant k for which the k-WL algorithm is able to distinguish any pair of non-isomorphic planar graphs (Gro98). Moreover, there is a k such that k-WL can distinguish all pairs of interval graphs (Lau10). For every class $\mathcal{C}$ of graphs with excluded minors there is a k such that k-WL can distinguish all pairs in $\mathcal{C}$ (Gro11). However, on the negative side, for every k there are non-isomorphic 3-regular graphs G_k, H_k of size $O(k)$ that are not distinguished by k-WL (CFI92).

Malkin gives another combinatorial algorithm in (Mal14) that he calls the k-dimensional C-V-C algorithm. It differs from the k-dimensional WL algorithm in the way it defines the neighbors of a tuple, but then the algorithm considers not only the number of neighbors in all the partitions but also the number of non-neighbors. He shows that the C-V-C algorithm and the WL algorithm are strongly related: the k-dim WL algorithm is stronger than the k-dim C-V-C algorithm, but the $(k+1)$-dim C-V-C algorithm is stronger than the k-dim WL algorithm.

1.1.2 Graph Theoretic Approach to Graph Isomorphism

In this section we briefly review the polynomial time isomorphism algorithms for planar graphs, interval graphs and graphs of bounded genus.

The Hopcroft-Tarjan algorithm for planar graphs (HT72) essentially reduces the problem of testing isomorphism of general planar graphs to that of testing isomorphism of 3-connected planar graphs. The given pair of graphs are first divided into connected components, then each connected component is further divided into biconnected components and finally each biconnected component is divided into 3-connected components. They show that the 3-connected components are unique and efficiently computable. Moreover the 3-connected components form a tree. Also 3-connected components have at most two embeddings in the plane. Using these facts they are able to assign unique labels to the graphs and hence determine their isomorphism.

Hopcroft and Wong (HW74) gave a linear time algorithm for planar graphs improving the $O(n \log n)$ bound obtained by the algorithm described above. Their algorithm basically improves the complexity of the 3-connected planar graph isomorphism problem, which dominated the complexity of the previous algorithm. In their approach they apply *reductions* simultaneously to the given pair of graphs such that the reduced graphs are isomorphic if and only if the original graphs are. These reductions replace certain subgraphs with subgraphs of another type. Moreover all possible reductions are assigned a priority and every time the highest priority reduction, among those applicable, is applied. It is shown that when no further reduction is applicable, the graph is either the five regular polyhedral graph or the trivial graph having a single vertex. These graphs can be tested for isomorphism by exhaustive matching in constant time.

Lueker and Booth (LB79) gave a linear time algorithm for deciding if a given pair of interval graphs are isomorphic. They reduce the problem to that of tree isomorphism for which they use a modified version of the algorithm due to Aho-Hopcroft-Ullman (AHU74). In (LB79), the authors construct a PQ-tree for each interval graph and then attach labels to the two trees in such a way as to guarantee that the labeled trees represent the graphs up to isomorphism. Their result exploits the fact that a graph is interval if and only if there exists a linear ordering of its cliques such that for each vertex v, the set of cliques that contain v appear consecutively within this ordering. PQ-tree is a data structure that is used to efficiently decide if a given graph is an interval graph.

Miller's (Mil80) algorithm for testing isomorphism of bounded genus graphs extends the algorithm of (FMR79) for embedding a graph on a surface of genus g. Both the algorithms take $n^{O(g)}$ steps. Filotti and Mayer (FM80) came up with a similar algorithm independently.

The algorithms described above are all polynomial time algorithms that exploit specific graph theoretic properties. There is also a moderately exponential time result for strongly regular graphs due to Babai (Bab80) that exploits the fact that for graphs belonging to this class there exists a vertex set of size $O(\sqrt{n}\log n)$ that when individualized leads to a canonical labeling. This was later improved by Spielman (Spi96), who showed that every strongly regular graph that has degree at most $d = o(n^{2/3})$ and second largest eigenvalue $o(d)$, contains a set of $O(n^{1/4}\sqrt{\log n})$ vertices whose individualization will result in a unique labeling after two refinement steps. Babai et al. (BCS$^+$13) further improve this result by combining the above method with ideas from group theory.

1.1.3 Group Theoretic Approach to Graph Isomorphism

The group theoretic approach dominated research on the graph isomorphism problem since the early 1980s. In fact this line of research has led to significant developments in computational group theory. One of the earliest results (Mat79) establishes that the problems of isomorphism recognition (GI), computation of an isomorphism (if one exists), computation of the number of isomorphisms, computation of a generating set of the automorphism group, computation of the order of the automorphism group and the problem of computing an automorphism partition (two vertices x, y belong to the same partition if and only if there exists at least one automorphism that maps x to y) of a graph are polynomially equivalent. Out of these computationally equivalent problems, most algorithms for graph isomorphism that make use of permutation group theory, compute the generating set of the automorphism group of the disjoint union of the given pair of graphs. It is well known that any group $\mathcal{G}$ has a generating set of size less than $\log_2(|\mathcal{G}|)$. So, even though the order of the automorphism group of a graph may be exponential, it has a generating set of polynomial size.

The first poly-time algorithm for a restricted class of graphs using group-theoretic methods is due to (Bab79; FHL80) for the class of colored graphs with bounded color-class-size. Let a vertex coloring induce a partition $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$ on the vertex set of a graph G. Then $Aut_{\mathcal{C}}(G)$, the automorphism group of G that respects this partition, can be computed in $O((k!)^6 \cdot poly(n))$ time where $k = \max_i |C_i|$ is a constant.

A significant algorithm based on group theory is the one due to Luks. He gave a polynomial time algorithm for graphs of bounded degree (Luk82). The algorithm with best known time complexity for general graphs combines this result with Zemlyachenko's degree reduction technique (ZKT82; Bab81; BL83), which uses combinatorial ideas of individualization and refinement.

Other algorithms that use ideas from group theory include those for graphs with bounded eigenvalue multiplicity (BGM82), k-contractible graphs (Mil83) and graphs with excluded minors (Pon88).

1.1.4 Polyhedral Approach to Graph Isomorphism

Before discussing the next approach, it would be appropriate to describe briefly the Sherali-Adams lift-and-project method as it plays an important role in the polyhedral approach.

The Sherali-Adams hierarchy of progressively stronger relaxations of integer polytopes, is defined as follows. We are given an explicit description of a starting polytope P_0 in terms of a system of linear inequalities $Ax - b \geq 0$. Also, P_0 is contained in the unit cube in $\mathbb{R}^n$. We have the integer polytope, $P = conv(P_0 \cap \{0,1\}^n)$. Starting from P_0, the Sherali-Adams method constructs a hierarchy of progressively stronger linear relaxations of P, given as $P_0 \supseteq P_1 \supseteq \cdots \supseteq P_n = P$. The procedure for obtaining P_k for some $k \geq 1$, is summarized below.

First we multiply each constraint $A_i x - b_i \geq 0$ by each product $\prod_{i \in I} x_i \prod_{j \in J}(1 - x_j)$ where I, J are disjoint subsets of $\{1, \ldots, n\}$ such that $|I \cup J| = k$, to obtain a set of polynomial inequalities. To this we add the inequalities $\prod_{i \in I} x_i \prod_{j \in J}(1 - x_j) \geq 0$ where I, J are disjoint subsets of $\{1, \ldots, n\}$ such that $|I \cup J| = min(k+1, n)$. The next step is to linearize the system of polynomial inequalities. This is done by replacing each x_i^2 by x_i and subsequently substituting each product monomial $\prod_{l \in L} x_l$ with a new variable y_L. So we have $y_{\{x_i\}} = x_i$. The resulting polytope $\hat{P}_k$ lies in $\mathbb{R}^d$ where $d = \sum_{j \in \{1,\ldots,k+1\}} \binom{n}{j}$. Finally, polytope P_k is obtained by projecting $\hat{P}_k$ back onto $\mathbb{R}^n$: $P_k = \{x \in \mathbb{R}^n : x_i = y_{\{i\}} \ \forall\, y \in \hat{P}_k\}$.

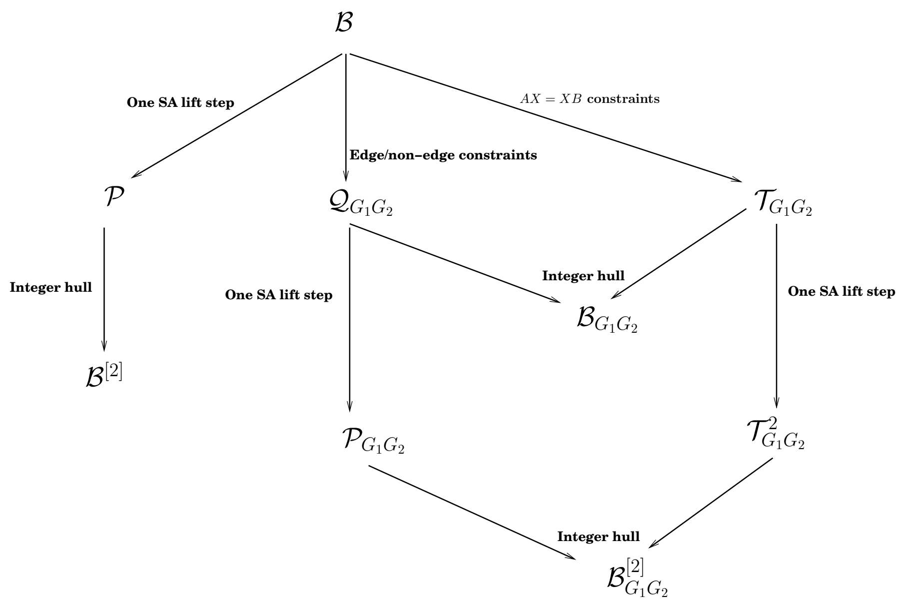

Figure 1.1: Relationships between the various polytopes

Next, we look at the definitions of various polytopes that we will encounter in the thesis. Some of these appear in the literature related to the polyhedral approach.

Figure 1.1 shows the relationships between the various polytopes and also gives an idea about their origins. We will now define these formally in the order in which they appear in the figure, starting from the top.

$\mathcal{B}$: is popularly known as the Birkhoff polytope. It is the convex hull of all the permutation matrices, where a permutation matrix P_σ is a $n \times n$ 0/1 matrix such that $(P_\sigma)_{ij} = 1$ if and only if $\sigma(i) = j$, for some permutation $\sigma \in S_n$, where S_n is the symmetric group of permutations of $[n]$. Clearly, $\mathcal{B}$ lies in $\mathbb{R}^{n\times n}$. From the Birkhoff-von Neumann theorem that every doubly-stochastic matrix can be expressed as the convex combination of permutation matrices, we have $\mathcal{B} = \{X \in [0,1]^{n\times n} | Xe = X^T e = e\}$, where e is a vector of all 1s.

$\mathcal{P}$: is the polytope that is obtained after one lift step of the Sherali-Adams (SA) (SA90) lift-and-project method, starting with the polytope $\mathcal{B}$. $\mathcal{P}$ lies in the space $\mathbb{R}^{n^2\times n^2}$.

$\mathcal{Q}_{G_1G_2}$: is actually not a polytope, but simply an algebraic set. It is defined as $\{X \in [0,1]^{n\times n} |\ X_{uv}X_{pq} = 0$ for all $\{u,p\} \in E_{G_1}$, $\{v,q\} \notin E_{G_2}$ or $\{u,p\} \notin$

$E_{G_1}, \{v,q\} \in E_{G_2}, Xe = X^T e = e\}$ in (Mal14). Here G_1, G_2 are simple undirected graphs on n vertices each. Clearly, $\mathcal{Q}_{G_1G_2} \subseteq \mathcal{B}$.

$\mathcal{T}_{G_1G_2}$: is the standard GI polytope, also known as the Tinhofer polytope. It is defined as $\{X \in [0,1]^{n\times n} | \ AX = XB, Xe = eX = e\}$ in (Tin91). Here A, B are the adjacency matrices of the graphs G_1, G_2, respectively. Again, $\mathcal{T}_{G_1G_2} \subseteq \mathcal{B}$.

$\mathcal{B}_{G_1G_2}$: is the convex hull of those permutation matrices P_σ that correspond to the isomorphisms (given by σ) between G_1, G_2. It can be shown that a permutation matrix P_σ belongs to $\mathcal{Q}_{G_1G_2}$ if and only if σ is an isomorphism between G_1, G_2. Same is true for $\mathcal{T}_{G_1G_2}$, i.e., a permutation matrix P_σ belongs to $\mathcal{T}_{G_1G_2}$ if and only if σ is an isomorphism between G_1, G_2. Hence, $\mathcal{B}_{G_1G_2} \subseteq \mathcal{Q}_{G_1G_2}$. Also, $\mathcal{B}_{G_1G_2} \subseteq \mathcal{T}_{G_1G_2}$. Moreover, $\mathcal{B}_{G_1G_2}$ is the convex hull of the integer points in $\mathcal{Q}_{G_1G_2}$. Similarly, $\mathcal{B}_{G_1G_2}$ is the convex hull of the integer points in $\mathcal{T}_{G_1G_2}$.

$\mathcal{B}^{[2]}$: is the convex hull of the integer points in $\mathcal{P}$. Each integer point in $\mathcal{P}$ is a $n^2 \times n^2$ symmetric 0/1 matrix, $P_\sigma^{[2]}$, that we call the second-order permutation matrix since there is a one-to-one correspondence between these matrices and the permutation matrices, given as $P_\sigma^{[2]}(ij,kl) = P_\sigma(i,j)P_\sigma(k,l)$. For the same reason, we call this polytope as the second-order Birkhoff polytope.

$\mathcal{P}_{G_1G_2}$: is the polytope in $\mathbb{R}^{n^2\times n^2}$ that is obtained after one lift step of SA, starting with $\mathcal{Q}_{G_1G_2}$. Polytope $\mathcal{P}$ is same as $\mathcal{P}_{G_1G_2}$ with G_1, G_2 as either empty or complete graphs. Hence, polytope $\mathcal{P}$ can be referred to as the superset of $\mathcal{P}_{G_1G_2}$ for all G_1, G_2.

$\mathcal{T}^2_{G_1G_2}$: is the polytope in $\mathbb{R}^{n^2\times n^2}$ that is obtained after one lift step of SA, starting with $\mathcal{T}_{G_1G_2}$. It can be shown that $\mathcal{T}^2_{G_1G_2} \subseteq \mathcal{P}_{G_1G_2}$. It follows from (Mal14, Lemma 3.2) for the case of $k = 2$.

$\mathcal{B}^{[2]}_{G_1G_2}$: is the convex hull of those vertices of $\mathcal{B}^{[2]}$ that correspond to isomorphisms between G_1, G_2. It can be shown that a second-order permutation matrix $P_\sigma^{[2]}$ belongs to $\mathcal{P}_{G_1G_2}$ if and only if σ is an isomorphism between G_1, G_2. Same is true for $\mathcal{T}^2_{G_1G_2}$, i.e., a second-order permutation matrix $P_\sigma^{[2]}$ belongs to $\mathcal{T}^2_{G_1G_2}$ if and only if σ is an isomorphism between G_1, G_2. Hence, $\mathcal{B}^{[2]}_{G_1G_2} \subseteq \mathcal{P}_{G_1G_2}$. Also, $\mathcal{B}^{[2]}_{G_1G_2} \subseteq \mathcal{T}^2_{G_1G_2}$. Moreover, $\mathcal{B}^{[2]}_{G_1G_2}$ is the convex hull of the integer points in $\mathcal{P}_{G_1G_2}$. Similarly, $\mathcal{B}^{[2]}_{G_1G_2}$ is the convex hull of the integer points in $\mathcal{T}^2_{G_1G_2}$.

Atserias and Maneva in (AM12) show that if k-WL distinguishes G_1, G_2 then the k^{th} level of the Sherali-Adams relaxation (SA90) (henceforth referred to as k-SA), starting with $\mathcal{T}_{G_1G_2}$, has no solution, or $\mathcal{T}^k_{G_1G_2} = \emptyset$. Also, in the same paper they show that if k-SA has no solution, then $(k+1)$-WL distinguishes G_1, G_2. Later Grohe and Otto in their paper (GO12) prove the existence of graph pairs G_1, G_2 such that k-WL does not distinguish G_1, G_2 (k-WL has a solution) and k-SA has no solution (able to distinguish). They also show the existence of graph pairs G_1, G_2 such that k-SA has a solution and $(k+1)$-WL distinguishes G_1, G_2. These results establish that the distinguishing power of k-SA is sandwiched between those of k-WL and $(k+1)$-WL.

Malkin in a recent paper (Mal14) basically confirms the results described above. More than that he shows that k-SA defined above, is the geometric analogue of the k-dimensional C-V-C algorithm. Further, he shows that k lift-and-project steps of SA starting with $\mathcal{Q}_{G_1G_2}$ is the geometric analogue of the k-WL algorithm. Finally, he proves the relationship $\mathcal{Q}^{k+1}_{G_1G_2} \subseteq \mathcal{T}^{k}_{G_1G_2} \subseteq \mathcal{Q}^{k}_{G_1G_2}$ which is the same as that established by Atserias and Maneva in (AM12).

From the above we can conclude that there is a strong relationship between the existing polyhedral and combinatorial approaches to GI. Also, as noted earlier, there exist graphs (CFI graphs) for every k that cannot be distinguished by the k-WL algorithm. Hence, we see the limitations of the polyhedral approach.

Recently, ODonnell, Wright, Wu, and Zhou (OWWZ14) and Codenotti, Schoenbeck, and Snook (SSC14) studied the Lasserre hierarchy (Las01) of semi-definite relaxations of the integer linear program for GI. They proved that the same CFI-graphs as mentioned above cannot even be distinguished by $o(n)$ levels of the Lasserre hierarchy.

1.2 Our Contributions

In the previous section we observed that the lift-and-project methods are not good enough to solve the Graph Isomorphism problem for general graphs, at least when we start with the polytope that corresponds to a natural formulation of the problem. All the attempts so far have been in showing that a constant number of rounds will not suffice to obtain the integer hull of the starting polytope. However, there has been no attempt to study an intermediate polytope that is obtained after a constant number of rounds. It is possible that an intermediate polytope, though not equal to the integer hull, has some interesting properties that can allow the problem to be solved efficiently, at least in some special situations. This applies to not just the Graph Isomorphism problem but also to other problems where only hardness results are known. In this thesis we look closely at the polytopes, $\mathcal{P}, \mathcal{B}^{[2]}$, and observe some interesting aspects of their geometry that can be exploited to obtain a procedure for solving GI. However, several aspects of the geometry still remain unknown and in future it would be nice to have a better understanding of them.

In this thesis we show that a given pair of graphs G_1, G_2 are isomorphic if and only if $\mathcal{P}_{G_1G_2}$ contains at least one point from $\mathcal{B}^{[2]}$. Hence, for non-isomorphic graphs, $\mathcal{P}_{G_1G_2}$, if non-empty, must be confined to $\mathcal{P} \setminus \mathcal{B}^{[2]}$. Moreover, we show that the polytopes $\mathcal{P}$ and $\mathcal{B}^{[2]}$ have the same dimension and each facet plane of $\mathcal{P}$ defines a facet of $\mathcal{B}^{[2]}$. $\mathcal{B}^{[2]}$ has other facets as well. Clearly, the region $\mathcal{P} \setminus \mathcal{B}^{[2]}$ is defined by these other facets of $\mathcal{B}^{[2]}$.

We study the facial structure of $\mathcal{B}^{[2]}$. This polytope is also studied in the literature of Quadratic Assignment Problem (QAP) (Kai97), where exponentially many facets of this polytope are identified. In this work we identify exponentially many additional facets of $\mathcal{B}^{[2]}$. Further, we define a partial ordering on each of the known families of facets.

We observe that the polytope $\mathcal{P}_{G_1G_2}$ is the disjoint union of $\mathcal{B}_{G_1G_2}$ and the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$. So the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ must violate the inequalities associated with one or more facets of $\mathcal{B}^{[2]}$. We call an inequality X a *minimal violated inequality* for a point p in the feasible region, if X is violated by p but any inequality less than X in the partial ordering, is not violated by p. We show that if there exists a single X for all p in the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$, then a simple search algorithm can solve the graph isomorphism problem in polynomial time. We also study the general case when more than one minimal violated inequalities are required to separate $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ from $\mathcal{B}^{[2]}$. We present an algorithm for GI that runs in $O(n^k)$ time where k is a minimal number of minimal violated inequalities that separate $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ from $\mathcal{B}^{[2]}$.

We believe that one of the contributions of this thesis is to provide a geometric characterization of the hard instances of the graph isomorphism problem. Clearly, these are the cases when a large number of facet defining inequalities of $\mathcal{B}^{[2]}$ are required to separate $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ from $\mathcal{B}^{[2]}$.

1.3 Organization

The rest of the thesis is organized as follows. In Chapter 2 we present our integer programming formulation of the graph isomorphism problem and its linear programming relaxation. We study the feasible regions and introduce the two polytopes, $\mathcal{P}$ and $\mathcal{B}^{[2]}$, and the special relationship that they share. We also introduce the concept of zero-one reducibility and show that a simple search algorithm must terminate in polynomial time if the feasible region is zero-one reducible. In Chapter 3 we study the facial structure of polytope $\mathcal{B}^{[2]}$ and present an exponential set of its facets. In Chapter 4 we show that there exists a partial ordering on the facet planes described in chapter 3 and that this ordering extends to some planes that support lower dimensional faces of $\mathcal{B}^{[2]}$. Further, we show that if there exists a single *minimal violated inequality* for all points in the feasible region of our LP, then the simple search algorithm presented in Chapter 2 terminates in polynomial time for non-isomorphic graphs. In the same chapter we discuss the general case when such an inequality does not exist. In Chapter 5 we show that there are facets of polytope $\mathcal{B}^{[2]}$ that still need to be discovered. In Chapter 6 we present some results related to the semidefinite program obtained by restricting the feasible region of the LP presented in Chapter 2, to the cone of positive semidefinite matrices. We show that this program is nothing but the Lovász Theta function of a product graph. Finally in Chapter 7 we present the results of some experiments first using the LP formulation and later the SDP formulation. These experiments are done with graphs taken from two classes considered difficult for GI. We conclude in Chapter 8 summarizing our contributions and present some open problems that provide avenues for further research.

Chapter 2

A Linear Programming Approach to Graph Isomorphism

2.1 Introduction

In this chapter we derive an integer linear program for graph isomorphism. Each (integer) solution of this program corresponds to a permutation that gives an isomorphism between the two graphs. The convex hull of these points is denoted by $\mathcal{B}^{[2]}$ when both graphs are either empty $((V, \emptyset))$ or complete (K_n). The polytope of the corresponding linear program (LP), after relaxing the integrality constraint, is denoted by $\mathcal{P}$. We show that the graphs are isomorphic if and only if the feasible region of the LP intersects $\mathcal{B}^{[2]}$. Further, we describe a simple *search* algorithm to determine if the feasible region of the linear program is confined to $\mathcal{P} \setminus \mathcal{B}^{[2]}$, thereby establishing non-isomorphism.

2.2 Integer Linear Program for GI

Define a second-order permutation matrix $P_\sigma^{[2]}$ corresponding to a permutation σ as $(P_\sigma^{[2]})_{ij,kl} = (P_\sigma)_{ij}(P_\sigma)_{kl}$, where P_σ is the permutation matrix corresponding to σ. We call the convex hull of the second-order permutation matrices, the second-order Birkhoff polytope $\mathcal{B}^{[2]}$ since the first-order Birkhoff polytope $\mathcal{B}^{[1]}$ or simply the well known Birkhoff polytope is defined as the convex hull of all the permutation matrices. In (PR09) a completely positive formulation of *Quadratic Assignment Problem* (QAP) is given. The feasible region of this program is precisely $\mathcal{B}^{[2]}$, see theorem 3 in (PR09).

Let $\mathcal{B}^{[2]}_{G_1G_2}$ denote the convex hull of the $P^{[2]}_\sigma$ where σ are the isomorphisms between G_1 and G_2. If the graphs are non-isomorphic, then $\mathcal{B}^{[2]}_{G_1G_2} = \emptyset$. Clearly $\mathcal{B}^{[2]} = \mathcal{B}^{[2]}_{G_1G_2}$ when $G_1 = G_2 = ([n], \emptyset)$ or $G_1 = G_2 = K_n$. As $\mathcal{B}^{[2]}_{G_1G_2}$ is a polytope we have an obvious observation.

Observation 2.2.1. *Given a pair of graphs, there exists a linear program (probably with exponentially many constraints) such that the feasible region of the program ($\mathcal{B}^{[2]}_{G_1G_2}$) is non-empty if and only if the graphs are isomorphic.*

Next we will develop an integer linear program such that the convex hull of its feasible points is $\mathcal{B}^{[2]}_{G_1G_2}$. It is easy to verify that for every permutation σ, $Y = P^{[2]}_\sigma$ satisfies equations 2.1a-2.1e.

$$Y_{ij,kl} - Y_{kl,ij} = 0 \qquad , \forall\ i,j,k,l \tag{2.1a}$$
$$Y_{ij,il} = Y_{ji,li} = 0 \qquad , \forall\ i, \forall\ j \neq l \tag{2.1b}$$
$$\sum_k Y_{ij,kl} = Y_{ij,ij} \qquad , \forall\ i,j,l \tag{2.1c}$$
$$\sum_k Y_{ij,lk} = Y_{ij,ij} \qquad , \forall\ i,j,l \tag{2.1d}$$
$$\sum_j Y_{ij,ij} = \sum_j Y_{ji,ji} = 1 \ , \forall\ i \tag{2.1e}$$

The solution plane of these equations will be denoted by P.

2.2.1 Affine Plane of $\mathcal{B}^{[2]}$

Lemma 2.2.2. *The solution plane of equations 2.1a-2.1e, P, is the affine plane spanned by $P^{[2]}_\sigma$s, i.e., $P = \{\sum_\sigma \alpha_\sigma P^{[2]}_\sigma | \sum_\sigma \alpha_\sigma = 1\}$.*

Proof. We will first show that the dimension of the solution plane is no more than $n!/(2(n-4)!) + (n-1)^2 + 1$.

In the following discussion we will split matrix Y into n^2 non-overlapping sub-matrices of size $n \times n$ which will be called *blocks*. The n blocks that contain the diagonal entries of Y will be called diagonal blocks. Note that $Y_{ij,kl}$ is the jl-th entry of the ik-th block. From the equation 2.1b, the off-diagonal entries of the diagonal blocks are zero and the diagonal entries of the off-diagonal blocks are also zero. Assume that the first $n-1$ diagonal entries of the first $n-1$ diagonal blocks are given. Then all the remaining diagonal entries can be determined using equations 2.1e. Therefore there are at most $(n-1)^2$ independent entries in the diagonal blocks.

Consider any off-diagonal block with first entry given by $Y_{r1,s1}$ where $r \leq n-3, r < s < n$. The sum of the entries in any row of this block is same as the main diagonal entry of that row in Y, see equation 2.1d. Same holds for the columns from symmetry condition 2.1a. Hence by fixing all but one off-diagonal entries of the first principal sub-matrix of the block, of size $(n-1) \times (n-1)$, we can fill in all the remaining entries. From equation 2.1c all the entries in the right most blocks $1n, \ldots, (n-3)n$ can be determined. Lower triangular entries of Y are determined by symmetry. At this stage we have only three blocks viz., $(n-2)(n-1)$, $(n-2)n$ and $(n-1)n$ above the main diagonal and their symmetric blocks below the main diagonal, whose entries remain to be determined. Consider the first principal $(n-1) \times (n-1)$ sub-matrix of the $(n-2)(n-1)$-th block. Let's say we are given the entries in the upper triangular portion of this sub-matrix. We will show that this is sufficient to determine the remaining entries in matrix Y. Clearly the entry $Y_{(n-2)1,(n-1)n}$ can be determined from equation 2.1d. Now using equation 2.1c we can determine all the entries in the first row of the $(n-2)n$-th block. So we know the entries in the first column of the $n(n-2)$-th block from symmetry. Again using condition 2.1c we can determine the entries in the first column of the $n(n-1)$-th block. That leads to the determination of entries in the first column of the $(n-2)(n-1)$-th block using conditions 2.1a and 2.1c. Now using a similar argument for the second row onwards of the $(n-2)(n-1)$-th block, we can fill in the remaining entries.

Hence we see that the number of independent variables, including those in diagonal blocks, is no more than $(n-1)^2 + ((n-1)(n-2)-1)(2+\cdots+(n-2)) + (n-1)(n-2)/2 = n!/(2(n-4)!) + (n-1)^2 + 1$.

In Appendix A we show that the dimension of $\mathcal{B}^{[2]}$ polytope is $\frac{n!}{2(n-4)!} + (n-1)^2 + 1$ (the same is also shown in (Kai97)). This claim along with the result of the previous paragraph leads to the conclusion that equations 2.1a-2.1e define the affine plane spanned by the $P_\sigma^{[2]}$s. $\square$

Corollary 2.2.3. *$\mathcal{B}^{[2]}$ is full dimensional in P.*

Lemma 2.2.4. *The only $0/1$ solutions of equations 2.1a-2.1e are $P_\sigma^{[2]}$s.*

Proof. Let Y be a 0/1 solution of the system of linear equations given by 2.1a-2.1e. Note that equations 2.1e and the non-negativity of the entries ensure that the diagonal of the solution is a vectorized doubly stochastic matrix. As the solution is a 0/1 matrix, the diagonal must be a vectorized permutation matrix, say P_σ. Then $Y_{ij,ij} = (P_\sigma)_{ij}\ \forall i,j$.

Equations 2.1a and 2.1d imply that $Y_{ij,kl} = 1$ if and only if $Y_{ij,ij} = 1$ and $Y_{kl,kl} = 1$. Hence $Y_{ij,kl} = Y_{ij,ij} \cdot Y_{kl,kl} = (P_\sigma)_{ij} \cdot (P_\sigma)_{kl} = (P_\sigma^{[2]})_{ij,kl}$. □

Let $G_1 = ([n], E_1)$ and $G_2 = ([n], E_2)$ be simple graphs on n vertices each. Define a graph $G = (V, E)$, where $V = [n] \times [n]$ and $\{ij, kl\} \in E$ if either $\{i,k\} \in E_1$ and $\{j,l\} \in E_2$ or $\{i,k\} \notin E_1$ and $\{j,l\} \notin E_2$. G is called the symmetric tensor product of G_1 and G_2.

Corollary 2.2.5. *The only 0/1 solutions of equations 2.1a-2.1e and $Y_{ij,kl} = 0\ \forall \{ij,kl\} \notin E$, are the $P_\sigma^{[2]}$ where σ are the isomorphisms between G_1 and G_2.*

Corollary 2.2.5 gives the following integer program for GI.

$$
\begin{aligned}
\text{IP-GI:}\quad \text{Find a point}\quad & Y \\
\text{subject to}\quad & 2.1a\text{-}2.1e && (2.2a)\\
& Y_{ij,kl} = 0 \qquad ,\ \forall\ \{ij,kl\} \notin E && (2.2b)\\
& Y_{ij,kl} \in \{0,1\}\ ,\ \forall\ i,j,k,l
\end{aligned}
$$

Note IP-GI is a feasibility formulation of GI. To formulate an optimization program for GI, replace the conditions 2.1e by $\sum_i Y_{ij,ij} \le 1$ and $\sum_j Y_{ij,ij} \le 1$, and set the objective to maximize $\sum_{i,j} Y_{ij,ij}$. The solutions of IP-GI coincide with those solutions of the optimization version where the objective function evaluates to n.

2.3 Linear Programming Relaxation

Program LP-GI, given below, is the linear programming relaxation of IP-GI. Here we only require that $Y_{ij,kl} \ge 0$ for all i,j,k,l. The condition $Y_{ij,kl} \le 1$ is implicit for all i, j, k, l. Let $\mathcal{P}_{G_1G_2}$ denote the feasible region of LP-GI. Note that $\mathcal{P}_{G_1G_2}$ is same as $\hat{\mathcal{Q}}^2_{G_1G_2}$ as defined in (Mal14). In (Mal14), the author defines a semi-algebraic set $\mathcal{Q}_{GH}$ as the set of $n \times n$ doubly stochastic matrices X satisfying $X_{ij}X_{kl} = 0\ \forall\ \{i,k\} \in E(G), \{j,l\} \notin E(H)$ or $\{i,k\} \notin E(G), \{j,l\} \in E(H)$. The author refers to $\mathcal{Q}^k_{GH}$ as the k-th Sherali-Adams relaxation (SA90) of $\mathcal{Q}_{GH}$ and $\hat{\mathcal{Q}}^k_{GH}$ as the lifted polytope in $\mathbb{R}^{n^k \times n^k}$ given via an extended formulation, whose projection in $\mathbb{R}^{n \times n}$ is the polytope $\mathcal{Q}^k_{GH}$. Clearly $\mathcal{B}^{[2]}_{G_1G_2} \subseteq \mathcal{P}_{G_1G_2}$. Define $\mathcal{P} = \mathcal{P}_{G_1G_2}$ where $G_1 = G_2 = ([n], \emptyset)$ or $G_1 = G_2 = K_n$. $\mathcal{P}$ is contained in the unit-cube $\{0,1\}^{n^2 \times n^2}$, so it is a polytope. It is also contained in the plane P, hence it too is a full-dimensional polytope in that plane.

$$
\begin{aligned}
\text{LP-GI:}\quad \text{Find a point}\quad & Y & & \\
\text{subject to}\quad & Y_{ij,kl} - Y_{kl,ij} = 0 & , \forall\ i,j,k,l & \qquad (2.3a)\\
& Y_{ij,il} = Y_{ji,li} = 0 & , \forall\ i, \forall j \neq l & \qquad (2.3b)\\
& \sum_k Y_{ij,kl} = \sum_k Y_{ij,lk} = Y_{ij,ij} & , \forall\ i,j,l & \qquad (2.3c)\\
& \sum_j Y_{ij,ij} = \sum_j Y_{ji,ji} = 1 & , \forall\ i & \qquad (2.3d)\\
& Y_{ij,kl} = 0 & , \forall\ \{ij,kl\} \notin E & \qquad (2.3e)\\
& Y_{ij,kl} \geq 0 & , \forall\ i,j,k,l &
\end{aligned}
$$

Lemma 2.3.1. *$\mathcal{P} = \mathcal{B}^{[2]}$ for $n \leq 3$.*

Proof. $\mathcal{P} = \mathcal{B}^{[2]}$ holds trivially for $n = 1$ and $n = 2$. We will show that for $n = 3$ any point $Y \in \mathcal{P}$ can be expressed as a convex combination of $P^{[2]}_\sigma$s. Fix an arbitrary index pair ij such that $Y_{ij,ij} > 0$. Define an $n \times n$ matrix M where $M_{kl} = Y_{ij,kl}$. Conditions 2.1c-2.1d ensure that M is a doubly stochastic matrix (one that is scaled by $Y_{ij,ij}$). Every doubly stochastic matrix can be expressed as a convex combination of permutation matrices so it follows that there exists a permutation σ such that $Y_{ij,k\sigma(k)} > 0$ for all $k \in [n]$. Due to symmetry of Y (condition 2.1a) and condition 2.1d, we have $Y_{k\sigma(k),k\sigma(k)} > 0$ and $Y_{k\sigma(k),ij} > 0$ for all k. W.l.o.g. let $Y_{11,11} > 0$. Also let σ be the identity permutation. So we have $Y_{11,22} = Y_{22,11} > 0$, $Y_{11,33} = Y_{33,11} > 0$, $Y_{22,22} > 0$ and $Y_{33,33} > 0$. Now if $Y_{22,33} = Y_{33,22} > 0$, we can subtract $\min\{Y_{11,22}, Y_{22,33}, Y_{33,11}\}$ times $P^{[2]}_\sigma$ from Y without introducing any negative entries in the residual matrix. At least one positive entry will reduce to zero. A suitably scaled residual matrix would still belong to $\mathcal{P}$ and the same process can be repeated. To complete the argument we will show that $Y_{22,33} = Y_{33,22}$ cannot be zero. Assume that $Y_{22,33} = Y_{33,22} = 0$. Then from the conditions $\sum_i Y_{22,3i} = Y_{22,22} = \sum_i Y_{22,i1}$ and $Y_{22,32} = Y_{22,21} = 0$ we conclude that $Y_{22,11} = 0$, contradicting the fact that $Y_{22,11}$ is positive. $\square$

The following observations are in order.

Observation 2.3.2. $\mathcal{P}_{G_1G_2} \cap \mathcal{B}^{[2]} = \mathcal{B}^{[2]}_{G_1G_2}$ *is the convex hull of* $P^{[2]}_\sigma$*s where* σ *are the isomorphisms between* G_1 *and* G_2*.*

Proof. From Corollary 2.2.5, we know that $P^{[2]}_\sigma$s are the only 0/1 points in $\mathcal{P}_{G_1G_2}$ where σ are the isomorphisms between G_1 and G_2. Clearly, $\mathcal{B}^{[2]}_{G_1G_2} \subseteq \mathcal{P}_{G_1G_2} \cap \mathcal{B}^{[2]}$. Let $Y \in (\mathcal{P}_{G_1G_2} \cap \mathcal{B}^{[2]}) \setminus \mathcal{B}^{[2]}_{G_1G_2}$. Since $Y \in \mathcal{B}^{[2]}$ but $Y \notin \mathcal{B}^{[2]}_{G_1G_2}$, we can express Y as a convex combination of $P^{[2]}_\sigma$s such that at least one of these does not correspond to an isomorphism between G_1, G_2. Let such a $P^{[2]}_\sigma$ correspond to some permutation σ_1. So we have $Y_{ij,kl} = 0$ for some ij, kl such that $\sigma_1(i) = j$ and $\sigma_1(k) = l$ i.e., $P^{[2]}_{\sigma_1}(ij, kl) = 1$. This is impossible. Hence $(\mathcal{P}_{G_1G_2} \cap \mathcal{B}^{[2]}) \setminus \mathcal{B}^{[2]}_{G_1G_2} = \emptyset$. $\square$

Consider the feasible region $\mathcal{P}_{G_1G_2}$ after a sequence of 0/1 assignments to some of the free variables. We have the following corollary to the above observation.

Corollary 2.3.3. *Let* $x_i = \alpha_i$ *for* $i = 1, \ldots, k$ *be a sequence of* k *0/1 assignments to the free variables* x_i *(each* x_i *corresponds to some* $Y_{ij,kl}$ *in LP-GI). Then* $(\mathcal{P}_{G_1G_2}|_{x_1=\alpha_1,\ldots,x_k=\alpha_k}) \cap \mathcal{B}^{[2]} = conv(\mathcal{P}^{[2]}_\sigma|\ \sigma$ *is an isomorphism between* G_1, G_2 *that respects* $x_i = \alpha_i$ *for all* $i \in \{1, \ldots, k\}$*).*

Proof. Similar to that of Observation 2.3.2. Note that $\sum_i \beta_i y_i = \gamma$ for $\gamma \in \{0, 1\}$, $\sum_i \beta_i = 1$, $\beta \geq 0$ implies that $y_i = \gamma$ for all i such that $\beta_i > 0$. $\square$

Observation 2.3.4. *Graphs* G_1, G_2 *are isomorphic if and only if the feasible region of LP-GI shares at least one point with* $\mathcal{B}^{[2]}$*.*

Proof. Let Y be a point in the feasible region of LP-GI such that $Y \in \mathcal{B}^{[2]}$. From Observation 2.3.2 Y can be expressed as $\sum_\sigma \alpha_\sigma P^{[2]}_\sigma$ where the sum is over the isomorphisms between G_1 and G_2. Since Y is non-empty, $\alpha_{\sigma'} > 0$ for some isomorphism σ'. Also, since Y respects the constraints 2.3e, $P^{[2]}_{\sigma'}$ must also respect these constraints.

Hence, $P^{[2]}_{\sigma'}$ belongs to the feasible region of LP-GI and σ' gives an isomorphism between G_1, G_2.

For the other direction, let σ'' be an isomorphism between G_1, G_2. So $P^{[2]}_{\sigma''}$ must respect the constraints 2.3e. All the other constraints of LP-GI are satisfied by every $P^{[2]}_{\sigma}$. Hence $P^{[2]}_{\sigma''}$ belongs to the feasible region of LP-GI. Also, $P^{[2]}_{\sigma''}$ is a vertex of $\mathcal{B}^{[2]}$. □

Observation 2.3.5. *Vertices of $\mathcal{B}^{[2]}$ (i.e., $P^{[2]}_{\sigma}$) are a subset of the vertices of $\mathcal{P}$.*

Proof. Since $\mathcal{P}$ is contained in the unit cube in $\mathbb{R}^{n^2 \times n^2}$, any vertices of the unit cube in $\mathbb{R}^{n^2 \times n^2}$ that are contained in $\mathcal{P}$ must form the vertices of $\mathcal{P}$. Clearly, $P^{[2]}_{\sigma}$ are such vertices. Hence, $\mathcal{B}^{[2]} \subseteq \mathcal{P}$. □

Observation 2.3.6. *The complete set of facet planes of $\mathcal{P}$ is $Y_{ij,kl} = 0\, \forall i \neq k, \forall j \neq l$.*

Proof. Observe that $Y_{ij,kl} = 0\, \forall i \neq k, \forall j \neq l$ are the only bounding planes of polytope $\mathcal{P}$. Hence, these are the only planes that can form the facets of polytope $\mathcal{P}$. However, there is no way to differentiate one of these from the other. So, if one of these defines a facet, then so must the other. Hence, $Y_{ij,kl} = 0\, \forall i \neq k, \forall j \neq l$ is a complete set of facet planes of $\mathcal{P}$. □

From Observations 2.3.6, 2.3.5 and Theorem 3.2.10 we have the following result.

Theorem 2.3.7. *All facet defining planes of $\mathcal{P}$ also define facets of $\mathcal{B}^{[2]}$ and all vertices of $\mathcal{B}^{[2]}$ are also vertices of $\mathcal{P}$. Besides, the dimensions of the two polytopes are same (both are full dimensional polytopes in plane P).*

2.4 Using the LP to Solve GI

We first define the notion of *zero-one reducibility* of a region.

2.4.1 Zero-One Reducibility

Definition 2.4.1. *Let R be a region in $\mathbb{R}^N$ and let $x_1, \ldots, x_N$ denote the coordinate variables. The region R is said to be zero-one reducible if either $R = \emptyset$ or R is a single point with all $0/1$ coordinates or there exists some index i and $\alpha \in \{0, 1\}$ such that $R|_{x_i = 1-\alpha} = \emptyset$ and $R|_{x_i = \alpha}$ is zero-one reducible.*

Suppose a region $R \subset \mathbb{R}^N$ is zero-one reducible. Then $x_{j_1}, x_{j_2}, \ldots, x_{j_r}$ will be called a reduction sequence if there exist $\alpha_{j_1}, \ldots, \alpha_{j_r} \in \{0,1\}$ such that $R|_{x_{j_1}=\alpha_{j_1},\ldots, x_{j_{i-1}}=\alpha_{j_{i-1}}, x_{j_i}=1-\alpha_{j_i}} = \emptyset \ \forall i$ and $R|_{x_{j_1}=\alpha_{j_1},\ldots,x_{j_r}=\alpha_{j_r}} = \emptyset$. Suppose $x_{j_1}, x_{j_2}, \ldots, x_{j_r}$ is a reduction sequence for R. Also suppose that there exists x_j and $\beta_j \in \{0,1\}$ such that $R|_{x_j=1-\beta_j} = \emptyset$. If $x_j \neq x_{j_i}$ for any i, then $x_{j_1}, x_{j_2}, \ldots, x_{j_r}$ is also a reduction sequence for $R|_{x_j=\beta_j}$. On the other hand, if $j = j_i$, then $x_{j_1}, \ldots, x_{j_{i-1}}, x_{j_{i+1}}, \ldots, x_{j_r}$ is a reduction sequence for $R|_{x_j=\beta_j}$. Hence $R|_{x_j=\beta_j}$ is also zero-one reducible.

From the above observation we can design a polynomial time recursive procedure to detect zero-one reducibility of a given region if we can detect in polynomial time whether the given region is empty or not, which in our case is equivalent to solving LP-GI with additional constraints of the form $x_i = \alpha_i$. The resulting linear program clearly has constraints that are polynomially many in the number of variables, and hence can be solved efficiently using say the ellipsoid method. Given the region R, a sub-routine $SearchVar()$ will consider each dimension j and each value $\alpha \in \{0,1\}$ and check if $R|_{x_j=1-\alpha} = \emptyset$. If such j and α are found, then repeat the procedure to detect the zero-one reducibility of $R|_{x_j=\alpha}$.

The procedure will try to detect if the feasible region of LP-GI for the given pair of graphs is zero-one reducible. In the case when the region does not reduce along any dimension, i.e., it is not zero-one reducible, we select any variable x_j and invoke the procedure for both the regions: $R|_{x_j=0}$ and $R|_{x_j=1}$. Therefore it will detect if there is any 0/1 point in the feasible region.

2.4.2 The Search Algorithm

The objective of the algorithm is to check if the feasible region intersects $\mathcal{B}^{[2]}$ or not. We know from Observation 2.3.2 that if the feasible region intersects $\mathcal{B}^{[2]}$, then there must be at least one $P_\sigma^{[2]}$ in the region because the entire intersection is the convex hull of $P_\sigma^{[2]}$s that correspond to the isomorphisms between the two graphs. Since $P_\sigma^{[2]}$ are the only 0/1 points in the feasible region (in the entire $\mathcal{P}$), all we need to detect is whether there is any 0/1 point in the feasible region $\mathcal{P}_{G_1G_2}$.

Algorithm 1 gives the algorithm which returns *true* if the graphs are isomorphic otherwise it returns $false$. It is based on the procedure described above. Parameter Q denotes the equations of the form $x = 0$ and $x = 1$ which are set in the process.

$LP(Q)$ represents LP-GI with additional equations Q. The second parameter U denotes the set of variables that are free (not yet set to either zero or one). Initially $Q = Q_0$ is an empty set and $U = U_0$ is the set of all the variables not set in 2.3b and 2.3e. Sub-routine $SearchVar(Q, U)$ returns a tuple (x, α) when the feasible region of LP-GI with additional conditions Q and $x = 1 - \alpha$ is empty. If no such variable/value pair is found, then it returns $(null, -1)$.

```
Function: GISolver(Q, U)
if LP(Q) is infeasible then
    return false /* Graphs are non-isomorphic */
else
    if LP(Q) is feasible and U = ∅ then
        return true /* Graphs are isomorphic */
    else
        (x, α) := SearchVar(Q, U);
        if α = 1 then
            return GISolver(Q ∪ {x = 1}, U \ {x});
        else
            if α = 0 then
                return GISolver(Q ∪ {x = 0}, U \ {x});
            else
                Select an arbitrary variable x from U;
                return GISolver(Q ∪ {x = 0}, U \ {x}) ∨
                GISolver(Q ∪ {x = 1}, U \ {x});
            end
        end
    end
end
```

Algorithm 1: Algorithm for GI

If we view the space searched by GISolver as a tree with (Q_0, U_0) as the root, then those nodes, (Q, U), have two children where $SearchVar(Q, U)$ returns $(null, -1)$. Call them split nodes. All other internal nodes have one child each. Let there be at most τ split nodes along any path from root to the leaves. Then the time complexity of this algorithm is $O(p(n)2^\tau)$ where $p(n)$ denotes a polynomial in n. Observe that if the feasible region of $LP(Q_0)$ is zero-one reducible, then the tree will not have any split nodes and the procedure will require polynomial time.

2.5 Conclusion

In this chapter we gave an integer program for GI where there is a one to one correspondence between the isomorphisms and the solution points. The convex hull of the solution points is denoted by $\mathcal{B}^{[2]}_{G_1G_2}$. The polytope of the LP relaxation of the integer program is denoted by $\mathcal{P}_{G_1G_2}$. We studied the relationship between the two polytopes when $G_1 = G_2 = (V, \emptyset)$ or $G_1 = G_2 = K_n$. We defined the concept of zero-one reducibility and presented an exact algorithm for GI that takes polynomial time if $\mathcal{P}_{G_1G_2}$ is zero-one reducible.

Chapter 3

Facial Structure of $\mathcal{B}^{[2]}$

3.1 Introduction

Among various definitions of the Quadratic Assignment problem (QAP), see (Kai97), one is $min\{\sum_{ij,kl}(A_{ik}B_{jl} + D_{ij,kl})Y_{ij,kl}|Y \in \mathcal{B}^{[2]}\}$ (PR09) where A, B, D are input matrices. This may also be stated as $min\{\langle(A \otimes B + D), Y\rangle| \ Y \in \mathcal{B}^{[2]}\}$. Thus QAP is an optimization problem over $\mathcal{B}^{[2]}$. In the literature (Kai97) $\mathcal{B}^{[2]}$ is referred to as QAP-polytope.

$\mathcal{B}^{[2]}$ is a zero-one polytope as is the Birkhoff polytope. But unlike the latter which has only n^2 facets, $\mathcal{B}^{[2]}$ has exponentially many known facets (JK97; Kai97) and exponentially many additional facets are identified in this chapter. We also present a generic inequality such that all the previously known facets and the new facets discovered in this chapter are special instances of this inequality. In chapter 5 we will show that there are some facets which are yet to be discovered.

3.2 Some Facets of $\mathcal{B}^{[2]}$

In the previous chapter we have shown that equations 2.1a-2.1e describe the affine plane, P, spanned by the vertices of $\mathcal{B}^{[2]}$. The linear description of the polytope now requires the description of the planes which define its facets. In this section we will identify exponentially many new facets of $\mathcal{B}^{[2]}$, in addition to exponentially many already known facets given in (JK97; Kai97). We will represent a facet by an inequality $f(x) \geq 0$ which defines the half space that contains the polytope and the plane $f(x) = 0$ contains the facet. Observe that the facet plane is given by the intersection of P with the plane given by $f(x) = 0$.

All the known facets of $\mathcal{B}^{[2]}$, including the ones that we are going to present in this chapter, are special instances of the following general inequality.

$$\sum_{ijkl} n_{ij} n_{kl} Y_{ij,kl} + (\beta - 1/2)^2 \geq (2\beta - 1) \sum_{ij} n_{ij} Y_{ij,ij} + 1/4 \tag{3.1}$$

where $\beta \in \mathbb{Z}$ and $n_{ij} \in \mathbb{Z}$ for all (ij).

Lemma 3.2.1. *$\mathcal{B}^{[2]}$ respects the inequality (3.1).*

Proof. We will show that $P_\sigma^{[2]}$ for every $\sigma \in S_n$ satisfies (3.1). The same must then hold for their convex combination since (3.1) defines a half-space, which is convex. (3.1) reduces to $(\sum_i n_{i\sigma(i)} - (\beta - 1/2))^2 \geq 1/4$ for $Y = P_\sigma^{[2]}$. Since $n_{ij}, \beta \in \mathbb{Z}$, the left hand side expression is the square of a positive number whose fractional part is 1/2. Clearly, this square is at least 1/4. □

The first set of facets are the instances of (3.1) where $n_{i_0 j_0} = n_{k_0 l_0} = 1$ for some $(i_0 j_0) \neq (k_0 l_0)$, all other $n_{ij} = 0$, and $\beta = 1$.

Theorem 3.2.2. *$Y_{i_0 j_0, k_0 l_0} \geq 0$ defines a facet of $\mathcal{B}^{[2]}$ for every i_0, j_0, k_0, l_0 such that $i_0 \neq k_0$ and $j_0 \neq l_0$.*

The above theorem is proved in (Kai97). We give an alternative proof in section 3.2.2.

The next set of facets are due to $\beta = n_{p_1 q_1} = n_{p_2 q_2} = n_{p_1 q_2} = 1$, $n_{kl} = -1$, and the rest of the n_{ij} are zero. Here p_1, p_2, k are any distinct indices. Similarly q_1, q_2, l are also any distinct indices.

Theorem 3.2.3. *Inequality $Y_{p_1 q_1, kl} + Y_{p_2 q_2, kl} + Y_{p_1 q_2, kl} \leq Y_{kl,kl} + Y_{p_1 q_1, p_2 q_2}$ defines a facet of $\mathcal{B}^{[2]}$, where p_1, p_2, k are distinct and q_1, q_2, l are also distinct and $n \geq 6$.*

The third set of facets is due to $\beta = n_{i_1 j_1} = \cdots = n_{i_m j_m} = 1$, $n_{kl} = -1$ and the remaining $n_{ij} = 0$.

Theorem 3.2.4. *Inequality $Y_{i_1 j_1, kl} + Y_{i_2 j_2, kl} + \ldots + Y_{i_m j_m, kl} \leq Y_{kl,kl} + \sum_{r<s} Y_{i_r j_r, i_s j_s}$, defines a facet of $\mathcal{B}^{[2]}$, where $i_1, \ldots, i_m, k$ are all distinct and $j_1, \ldots, j_m, l$ are also distinct. In addition, $n \geq 6, m \geq 3$.*

Proofs of theorems 3.2.3, 3.2.4 appear in sections 3.2.4, 3.2.5, respectively.

The next two sets of facets are established in (Kai97). Let P_1 and P_2 be disjoint subsets of $[n]$. Similarly let Q_1 and Q_2 also be disjoint subsets of $[n]$. In these facets $n_{ij} = 1$ if $(ij) \in (P_1 \times Q_2) \cup (P_2 \times Q_1)$ and $n_{ij} = -1$ if $(ij) \in (P_1 \times Q_1) \cup (P_2 \times Q_2)$. All other n_{ij} are zero. In the following case $P_2 = Q_1 = \emptyset$.

Theorem 3.2.5. *(Kai97, Definition 8.5) Following inequality defines a facet of* $\mathcal{B}^{[2]}$
$(\beta - 1) \sum_{(ij)\in P_1 \times Q_2} Y_{ij,ij} \leq \sum_{(ij),(kl)\in P_1\times Q_2, i<k} Y_{ij,kl} + (1/2)(\beta^2 - \beta)$
when (i) $\beta + 1 \leq |P_1|, |Q_2| \leq n - 3$*, (ii)* $|P_1| + |Q_2| \leq n - 3 + \beta$*, (iii)* $\beta \geq 2$.

The next set of facets, with $Q_1 = \emptyset$, is given by the following theorem.

Theorem 3.2.6. *(Kai97, Definition 8.6) Following inequality defines a facet of* $\mathcal{B}^{[2]}$
$-(\beta - 1) \sum_{(ij)\in P_1\times Q_2} Y_{ij,ij} + \beta \sum_{(ij)\in P_2\times Q_2} Y_{ij,ij} + \sum_{(ij),(kl)\in P_1\times Q_2, i<k} Y_{ij,kl} +$
$\sum_{(ij),(kl)\in P_2\times Q_2, i<k} Y_{ij,kl} - \sum_{(ij)\in P_1\times Q_2,(kl)\in P_2\times Q_2} Y_{ij,kl} + (1/2)(\beta^2 - \beta) \geq 0$
where the conditions on the parameters are as given in (Kai97, Definition 8.6).

3.2.1 A Useful Identity

The following lemma gives a method to establish a facet.

Let X be a set of vectors. Then $LS(X)$ denotes the subspace spanned by the vectors in X.

Lemma 3.2.7. *Let V be the set of vertices of a polytope such that the affine plane of V does not contain the origin and $f(x) \geq 0$ be a linear inequality satisfied by all the vertices. Let $S = \{v \in V | f(v) = 0\}$ such that $V \setminus S \neq \emptyset$. Also, let a vertex $v_0 \in V \setminus S$ be such that $V \subset LS(\{v_0\} \cup S)$. Then the affine plane of S defines a facet, i.e., $f(x) \geq 0$ defines a facet (intersection of $f(x) = 0$ plane with the affine plane of V is the facet plane).*

Proof. Let d denote the dimension of $LS(V)$. So the dimension of the affine plane of V is $d-1$. Also $V \subset LS(\{v_0\} \cup S)$ so the dimension of $LS(S)$ is at least $d-1$. As the affine plane of S does not contain the origin, the dimension of the affine plane of S is at least $d-2$. Observe that V is not contained in $LS(S)$ since $f(x)$ is non-zero for $x \in V \setminus S$. We conclude that the dimension of the affine plane of S is exactly one less than that of the affine plane of V. □

Corollary 3.2.8. *Let $G = (V \setminus S, E)$ be a graph with the property that $\{u, v\} \in E$ iff $u - v \in LS(S)$. If G is connected, then S is a facet.*

Proof. Since G is connected, there exists a simple path in G between every pair of vertices. Let us fix an arbitrary vertex as v_0. Now consider any vertex u in $V(G) \setminus \{v_0\}$. There must exist a simple path between u and v_0 via some vertices $u_1, u_2, \ldots, u_k$. Since $\{u, u_1\}, \{u_1, u_2\}, \ldots, \{u_k, v_0\}$ are edges on this path and $u - v \in LS(S)$ for $\{u, v\} \in E$, we have $u_k - v_0 \in LS(S)$ or $u_k \in LS(S \cup \{v_0\})$. Now, since $u_{k-1} - u_k \in LS(S)$, we have $u_{k-1} \in LS(S \cup \{u_k\})$ or $u_{k-1} \in LS(S \cup \{v_0\})$. This could now be extended to lead to $u \in LS(S \cup \{v_0\})$. The rest follows from the lemma. □

We say that a permutation σ' is a transposition of another permutation σ (or that σ and σ' are transpositions of each other) if the images of the two permutations differ at two indices, i.e., if there are two distinct indices x, y such that $\sigma(x) = \sigma'(y), \sigma(y) = \sigma'(x)$ and $\sigma(z) = \sigma'(z)$ for all $z \in [n] \setminus \{x, y\}$.

Let k_1, k_2, k_3 be any three integers belonging to $[n]$. Let $\sigma_1, \ldots, \sigma_6$ be a set of permutations of S_n which have same image for each element of $[n] \setminus \{k_1, k_2, k_3\}$, i.e., $\sigma_i(z) = \sigma_j(z)$ for all $z \in [n] \setminus \{k_1, k_2, k_3\}$ for every $i, j \in \{1, \ldots, 6\}$. Let images of k_1, k_2, k_3 under $\sigma_1, \ldots, \sigma_6$ be (a, b, c), (a, c, b), (b, a, c), (b, c, a), (c, a, b), (c, b, a) respectively. Further, suppose x, y be any two elements of $[n] \setminus \{k_1, k_2, k_3\}$. Let σ'_i be transposition of σ_i on indices x and y, for each $i = 1, \ldots, 6$. Following is a useful identity.

Lemma 3.2.9. *Let $\Sigma = \{\sigma_1, \ldots, \sigma_6, \sigma'_1, \ldots, \sigma'_6\}$ be a set of permutations as defined above. Then $\sum_{\sigma \in \Sigma} sign(\sigma) P^{[2]}_{\sigma}(ij, kl) = 0 \; \forall \; i, j, k, l$.*

Proof. All we need to show is that $\sum_{\sigma \in \Sigma, \sigma(i)=j, \sigma(k)=l} sign(\sigma) = 0 \; \forall \; i, j, k, l$. Note that two permutations have opposite signs if they are transpositions of each other. So from the above we have $sign(\sigma_1) = sign(\sigma_4) = sign(\sigma_5) = -sign(\sigma_2) = -sign(\sigma_3)$ $= -sign(\sigma_6)$. Also, $sign(\sigma_i) = -sign(\sigma'_i)$ leading to $\sum_{\sigma \in \{\sigma_1, \ldots, \sigma_6\}} sign(\sigma) = 0$ as well as $\sum_{\sigma \in \{\sigma'_1, \ldots, \sigma'_6\}} sign(\sigma) = 0$. Let us assume that $\sigma_i(x) = d$ and $\sigma_i(y) = e$. So, $\sigma'_i(x) = e$ and $\sigma'_i(y) = d$, for all $i \in \{1, \ldots, 6\}$. We consider the following six cases.
Case (i). $i \in \{k_1, k_2, k_3\}$, $k \in \{x, y\}$: here the only interesting scenario is when $j \in \{a, b, c\}$ and $l \in \{d, e\}$ since otherwise none of the permutations in Σ would contribute anything to the sum. Now, depending on l we either get contributions from

two permutations in $\{\sigma_1, \ldots, \sigma_6\}$ or we get contributions from two permutations in $\{\sigma'_1, \ldots, \sigma'_6\}$. Moreover, these permutations must be transpositions of each other. W.l.o.g. let $i = k_1, j = a, k = x, l = d$. So the contributing permutations are σ_1, σ_2. Note that these are transpositions of each other.
Case (ii). $\{i, k\} \subset \{k_1, k_2, k_3\}$: here we get contributions from two permutations, one from σ_i, say σ_1 w.l.o.g. and the other σ'_1. Clearly these are transpositions of each other.
Case (iii). $\{i, k\} = \{x, y\}$: depending on j, l we have either contributions from $\{\sigma_1, \ldots, \sigma_6\}$ or contributions from $\{\sigma'_1, \ldots, \sigma'_6\}$.
Case (iv). $i \in \{k_1, k_2, k_3\}$, $k \in [n] \setminus \{k_1, k_2, k_3, x, y\}$: we get contributions from two permutations from σ_i and the corresponding two permutations from σ'_i. Clearly their signs cancel each other.
Case (v). $i \in \{x, y\}$, $k \in [n] \setminus \{k_1, k_2, k_3, x, y\}$: depending on j we have either $\{\sigma_1, \ldots, \sigma_6\}$ contributing to the sum or $\{\sigma'_1, \ldots, \sigma'_6\}$ contributing to the sum.
Case (vi). $\{i, k\} \cap \{k_1, k_2, k_3, x, y\} = \emptyset$: here we have all the twelve permutations contributing to the sum and hence their signs add up to zero. $\square$

3.2.2 Facets Due to the Non-negativity Constraint

In this section V will denote the set $\{P^{[2]}_\sigma | \sigma \in S_n\}$ and S will denote $\{P^{[2]}_\sigma | f(P^{[2]}_\sigma) = 0\}$.

Theorem 3.2.10. *The non-negativity constraint $Y_{ij,kl} \geq 0$, defines a facet of $\mathcal{B}^{[2]}$ for every i, j, k, l such that $i \neq k$ and $j \neq l$.*

Proof. Observe that the non-negativity condition is satisfied by every $P^{[2]}_\sigma$. Every vertex in the set $V \setminus S$ corresponds to a permutation σ where $\sigma(i) = j$ and $\sigma(k) = l$. Consider a graph $G = (V \setminus S, E)$ where $E = \{P^{[2]}_\sigma, P^{[2]}_{\sigma'}\}$ such that σ, σ' are transpositions of each other. Since the set of permutations corresponding to the vertices in $V \setminus S$ is isomorphic to the group S_{n-2}, G must be a connected graph.

Let $P^{[2]}_{\sigma_1}$ and $P^{[2]}_{\sigma'_1}$ be a pair of matrices in $V \setminus S$ where σ and σ' are transpositions of each other. Let $k_1 = i, k_2 = k$ and k_3 be any element other than x, y, i, k. Consider all the permutations $\sigma_2, \ldots, \sigma_6, \sigma'_2, \ldots, \sigma'_6$ as defined in the context of lemma 3.2.9. Observe that all the $P^{[2]}_\sigma$s corresponding to these ten permutations belong to S. Hence we can express $P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma'_1}$ in terms of vertices in S using the identity in the lemma. From Corollary 3.2.8 the inequality defines a facet. $\square$

We will use the following lemma to show that the graph under consideration is connected, hence the name. This is required to prove that certain inequality defines a facet of $\mathcal{B}^{[2]}$, as shown in Corollary 3.2.8.

3.2.3 The Connection Lemma

Lemma 3.2.11. *(1) Let X be a set of vertices $P_\sigma^{[2]}$ such that $\sigma(1) = 1, \ldots, \sigma(a) = a$ and $\sigma(a+1) \notin I_1, \ldots, \sigma(a+b) \notin I_b$ where I_j are subsets of $[n] \setminus \{1, 2, \ldots, a\}$ such that $|\cup_i I_i| \leq n - a - b$. Let $G = (X, E)$ be a graph in which $\{P_\sigma^{[2]}, P_{\sigma'}^{[2]}\} \in E$ iff σ and σ' are transpositions of each other. Then G is connected.*

(2) Let X be a set of vertices $P_\sigma^{[2]}$ such that $\sigma(1) = 1, \ldots, \sigma(a) = a, \sigma(a+1) \neq x_1, \ldots, \sigma(a+b) \neq x_b$, where all x_i are distinct and greater than a and $a + b < n$. Let $G = (X, E)$ be a graph in which $\{P_\sigma^{[2]}, P_{\sigma'}^{[2]}\} \in E$ iff σ and σ' are transpositions of each other. Then G is connected.

Proof. (1) Let $I = \cup_i I_i$. Without loss of generality assume that $I \subseteq \{a+b+1, a+b+2, \ldots, n\}$. Hence the $P_\sigma^{[2]}$ corresponding to the identity permutation belongs to X.

Given any vertex $P_{\sigma_0}^{[2]} \in X$, we will show that there is a path from $P_{\sigma_0}^{[2]}$ to $P_{\sigma:\sigma(i)=i}^{[2]}$ in G. Starting from $P_{\sigma_0}^{[2]}$, suppose the path has been built up to $P_\sigma^{[2]}$ for some σ such that for some $i \in \{a+1, \ldots, a+b\}$, $\sigma(i) \in \{a+b+1, \ldots, n\}$. Hence there must exist a $j \in \{a+b+1, a+b+2, \ldots, n\}$ such that $\sigma(j) \in \{a+1, \ldots, a+b\}$. Consider the permutation σ' which is the transposition of σ with respect to the indices i, j. $P_{\sigma'}^{[2]}$ is also in X and $\{P_\sigma^{[2]}, P_{\sigma'}^{[2]}\}$ is an edge. Extend the path to $P_{\sigma'}^{[2]}$. Finally we will reach a permutation in which all indices in the range $a+1, \ldots, a+b$ map to $a+1, \ldots, a+b$ and hence all the indices in $a+b+1, \ldots, n$ map to $a+b+1, \ldots, n$.

Next perform transpositions within indices of $a+1, \ldots, a+b$ so that finally $\sigma(i)$ maps to i for all i in this range. Note that the vertices corresponding to the permutations generated in the process, all belong to X. In the end we do the same for indices in the range $a+b+1, \ldots, n$.

(2) The claim is vacuously true if X is empty. So we assume that it is non-empty. By relabeling we can make sure that $x_i \neq a+i$ for all $1 \leq i \leq b$. So without loss of generality we can assume that $P_{\sigma:\sigma(i)=i}^{[2]}$ belongs to X. To prove the claim we will show that starting from any arbitrary vertex $P_{\sigma_0}^{[2]} \in X$ there is a path from $P_{\sigma_0}^{[2]}$ to

$P^{[2]}_{\sigma:\sigma(i)=i}$. While tracing this path, the current permutation σ has $\sigma(a+i) \neq a+i$ while $\sigma(a+j) = a+j$ for all $j < i$. Let $\sigma^{-1}(a+i) = a+k$.

If either $a+k > a+b$ or $a+k \leq a+b$ and $\sigma(a+i) \neq x_k$, then perform transposition on indices $a+i$ and $a+k$ resulting into the new permutation σ' that is "closer" to the identity and $P^{[2]}_{\sigma'} \in X$.

Now consider the case where $\sigma(a+i) = x_k$. Observe that there must be at least three indices beyond $a+i-1$. Let $a+j$ be any index greater than $a+b$. Perform transposition on indices $a+j$ and $a+k$ giving σ' and then perform transposition on $a+i$ and $a+j$. Let the new permutation be σ''. Observe that both, $P^{[2]}_{\sigma'}$ and $P^{[2]}_{\sigma''}$, belong to X. So the path extends by edges $\{P^{[2]}_{\sigma}, P^{[2]}_{\sigma'}\}$ and $\{P^{[2]}_{\sigma'}, P^{[2]}_{\sigma''}\}$. Further, σ'' is closer to the identity.

Thus the path eventually reaches the identity. $\square$

3.2.4 A Polynomial Sized Family of Facets

Theorem 3.2.12. *Inequality $Y_{p_1q_1,kl} + Y_{p_2q_2,kl} + Y_{p_1q_2,kl} \leq Y_{kl,kl} + Y_{p_1q_1,p_2q_2}$, defines a facet of $\mathcal{B}^{[2]}$, where p_1, p_2, k are distinct and q_1, q_2, l are also distinct and $n \geq 6$.*

Proof. The set of vertices which satisfy the inequality strictly is the union of $X_1 = \{P^{[2]}_{\sigma} | \sigma(p_1) = q_1, \sigma(p_2) = q_2, \sigma(k) \neq l\}$ and $X_2 = \{P^{[2]}_{\sigma} | \sigma(p_1) \neq q_1,\ \sigma(p_1) \neq q_2,\ \sigma(p_2) \neq q_2,\ \sigma(k) = l\}$. So $V \setminus S = X_1 \cup X_2$.

Define a graph $G = (X_1 \cup X_2, E)$ where E is the set of edges $\{P^{[2]}_{\sigma}, P^{[2]}_{\sigma'}\}$ where σ is a transposition of σ'. From lemma 3.2.11 the subgraphs on X_1 and X_2 are each connected. We also notice that there is no edge connecting these components. So we add a special edge $\{P^{[2]}_{\alpha_1}, P^{[2]}_{\alpha_2}\}$ to G making the graph connected, where $P^{[2]}_{\alpha_1}$ is an arbitrary member of X_1 and α_2 is defined below. Let $i_2 = \alpha_1^{-1}(l)$ and r be any index other than p_1, p_2, k, i_2. So α_1 maps $p_1 \to q_1, p_2 \to q_2, k \to b, i_2 \to l, r \to a$ for some a and b. Define α_2 to be the permutation which maps $p_1 \to a, p_2 \to q_1, k \to l, i_2 \to b, r \to q_2$. At all other indices the images of α_1 and α_2 coincide. Observe that $P^{[2]}_{\alpha_2} \in X_2$.

Now we will show that for each edge $\{P^{[2]}_{x'}, P^{[2]}_{y'}\}$ of the graph, $P^{[2]}_{x'} - P^{[2]}_{y'}$ belongs to $LS(S)$. We begin with the edge $\{P^{[2]}_{\alpha_1}, P^{[2]}_{\alpha_2}\}$. Let $\sigma_1 = \alpha_1$. Define $\sigma_2, \ldots, \sigma_6$ using $k_1 = p_1, k_2 = p_2, k_3 = r$ as described before lemma 3.2.9. Taking $x = k$ and $y = i_2$, define $\sigma'_1, \ldots, \sigma'_6$. See that $\alpha_2 = \sigma'_5$. The rest of the permutations are in S. Hence from lemma 3.2.9 $P^{[2]}_{\alpha_1} - P^{[2]}_{\alpha_2}$ can be expressed as a linear combination of vertices in S.

Next we will show that each edge in the graph on X_1 has the same property. Let $\{P^{[2]}_{\sigma_1}, P^{[2]}_{\sigma'_1}\}$ be an edge in the graph on X_1. In both permutations p_1 and p_2 map to q_1 and q_2 respectively. Define $k_1 = p_1$ and $k_2 = p_2$. Also, define k_3 as the index different from p_1, p_2, k, such that $\sigma_1(k_3) = \sigma'_1(k_3)$. Note that such an index must exist since $n \geq 6$. Consider 5 new permutations formed from σ_1 by permuting the images of k_1, k_2 and k_3. Call them $\sigma_2, \ldots, \sigma_6$. Similarly define $\sigma'_2, \ldots, \sigma'_6$ from σ'_1. Observe that in each σ_i for $i \geq 2$, k does not map to l. In addition either p_1 does not map to q_1 or p_2 does not map to q_2. Hence $P^{[2]}_{\sigma_2}, \ldots, P^{[2]}_{\sigma_6}$ belong to S. Similarly $P^{[2]}_{\sigma'_2}, \ldots, P^{[2]}_{\sigma'_6}$ also belong to S. From lemma 3.2.9, $P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma'_1} \in LS(S)$.

Now we consider the edges of X_2. Let $\{P^{[2]}_{\sigma_1}, P^{[2]}_{\sigma'_1}\}$ be one such edge. Let x, y be the indices at which σ_1 and σ'_1 differ. Consider two cases of σ_1: (1) $\sigma_1(p_1) = a, \sigma_1(p_2) = b, \sigma_1(k) = l, \sigma_1(r) = q_1, \sigma_1(s) = q_2$, (2) $\sigma_1(p_1) = a, \sigma_1(p_2) = q_1, \sigma_1(k) = l, \sigma_1(r) = q_2$.

Case (1) Subcase $|\{p_1, p_2, r, s\} \cap \{x, y\}| \leq 1$: If $p_1 \notin \{x, y\}$, then define $k_1 = k$, $k_2 = p_1$, and k_3 be any index in $\{r, s\} \setminus \{x, y\}$. Otherwise $k_1 = k, k_2 = p_2, k_3 = s$. All the permutations $\sigma_2, \ldots, \sigma_6$ and $\sigma'_2, \ldots, \sigma'_6$ as defined before lemma 3.2.9 are in S. So $P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma'_1}$ can be expressed as a linear combination of points in S using the identity.

Subcase $\{x, y\} \subset \{p_1, p_2, r, s\}$: Only three cases are possible here: $x = p_2, y = r$; $x = \sigma_1^{-1}(q_1), y = \sigma_1^{-1}(q_2)$; and $x = p_1, y = p_2$, apart from exchanging the roles of x and y. In the first case let $k_1 = p_1, k_2 = s, k_3 = k$ and use lemma 3.2.9. The remaining two cases are proved differently.

In these two cases we will not show that $P^{[2]}_{\sigma} - P^{[2]}_{\sigma'}$ can be expressed as a linear combination of vertices in S. Instead, we will delete such edges from E and show that the reduced graph is still connected. Consider an edge $\{P^{[2]}_{\sigma}, P^{[2]}_{\sigma'}\}$ of the second type where σ maps: $p_1 \to a, p_2 \to b, k \to l, r \to q_1, s \to q_2, u \to v$ and σ' maps: $p_1 \to a, p_2 \to b, k \to l, r \to q_2, s \to q_1, u \to v$. Note that since $n \geq 6$, the pair u, v always exists. Rest of the indices have the same images in the two permutations. To show that after dropping the edges of this class the graph remains connected, define two new permutations: α_1: $p_1 \to a, p_2 \to b, k \to l, r \to v, s \to q_2, u \to q_1$ and α_2: $p_1 \to a, p_2 \to b, k \to l, r \to q_2, s \to v, u \to q_1$. Other mappings are same as in σ. Observe that $\{P^{[2]}_{\sigma}, P^{[2]}_{\alpha_1}\}, \{P^{[2]}_{\alpha_1}, P^{[2]}_{\alpha_2}\}$ and $\{P^{[2]}_{\alpha_2}, P^{[2]}_{\sigma'}\}$ are edges in the reduced graph, hence there is a path from $P^{[2]}_{\sigma}$ to $P^{[2]}_{\sigma'}$ in it.

Let $\{P^{[2]}_{\sigma}, P^{[2]}_{\sigma'}\}$ be an edge of the third type. So σ maps $p_1 \to a, p_2 \to b, k \to l, r \to q_1, s \to q_2$ and σ' maps $p_1 \to b, p_2 \to a, k \to l, r \to q_1, s \to q_2$. Again to show a path from $P^{[2]}_{\sigma}$ to $P^{[2]}_{\sigma'}$ in the graph after deleting both types of edges, define α_1:$p_1 \to a, p_2 \to q_1, k \to l, r \to b, s \to q_2$ and α_2: $p_1 \to b, p_2 \to q_1, k \to l, r \to a, s \to q_2$. Other mappings are same as in σ. Note that $\{P^{[2]}_{\sigma}, P^{[2]}_{\alpha_1}\}$ is an edge of the first type. The remaining edges $\{P^{[2]}_{\alpha_1}, P^{[2]}_{\alpha_2}\}$ and $\{P^{[2]}_{\alpha_2}, P^{[2]}_{\sigma'}\}$ are covered in Case (2).

Case (2) Subcase $\{p_1, p_2, r = \sigma^{-1}(q_2)\} \cap \{x, y\} = \emptyset$: In this case define $k_1 = p_1, k_2 = p_2, k_3 = r$. See that $\sigma_2, \ldots, \sigma_6$ and $\sigma'_2, \ldots, \sigma'_6$ belong to S.

Subcase $|\{p_1, p_2, r = \sigma^{-1}(q_2)\} \cap \{x, y\}| = 1$: If $x = p_1$ or $y = p_1$, then $k_1 = p_2, k_2 = r, k_3 = k$. If $x = p_2$ or $y = p_2$, then $k_1 = p_1, k_2 = r, k_3 = k$. Finally if $x = r$ or $y = r$, then $k_1 = p_1, k_2 = p_2, k_3 = k$. In each case lemma 3.2.9 gives a desired linear expression in terms of points in S for $P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma'_1}$.

Subcase $\{x, y\} \subset \{p_1, p_2, r = \sigma^{-1}(q_2)\}$ does not arise because in this case every transposition leads to a permutation in S.

From Corollary 3.2.8 we conclude that S is a facet. $\square$

Total number of facets defined by this theorem is $n^2(n-1)^2(n-2)^2$.

3.2.5 An Exponential Sized Family of Facets

Consider the following inequality

$$Y_{i_1j_1,kl} + Y_{i_2j_2,kl} + \ldots + Y_{i_mj_m,kl} \leq Y_{kl,kl} + \sum_{r<s} Y_{i_rj_r,i_sj_s} \tag{3.2}$$

where $n \geq 6, 3 \leq m \leq n-3$, indices $i_1, \ldots, i_m, k$ are distinct and $j_1, \ldots, j_m, l$ are also distinct. In the rest of this section we will show that inequality (3.2) also defines a facet of $\mathcal{B}^{[2]}$.

We will continue to use S to denote the set of vertices for which the given inequality is tight. Let T denote the set of remaining vertices. Set T can be subdivided into the following classes:

1. $T_1 : k \to l, i_1 \not\to j_1, i_2 \not\to j_2, \ldots, i_m \not\to j_m$.

2. $T_2 : k \to l$ and three or more $i_r \to j_r$.

3. $T_3 : k \not\to l$ and two or more $i_r \to j_r$.

In classes T_2 and T_3 we do further subdivision. If a permutation in T_2 maps i_r to j_r for x out of m indices, then such a permutation belongs to subclass denoted by $T_{2,x}$. Similarly $T_{3,x}$ is defined. Observe that $T_2 = \cup_{x\geq 3} T_{2,x}$ and $T_3 = \cup_{x\geq 2} T_{3,x}$.

Lemma 3.2.13. *Let $m \geq 3$. The graph G_1 on T_1, with edge set $\{P^{[2]}_{\sigma'}, P^{[2]}_{\sigma''}\}$ where σ' is a transposition of σ'', is connected. Further the difference vector corresponding to each edge belongs to $LS(S)$.*

Proof. The first part of the lemma is established from lemma 3.2.11(2).

For the second part let $\{P^{[2]}_{\sigma_1}, P^{[2]}_{\sigma'_1}\}$ be an edge in G_1 where $\sigma_1(x) = \sigma'_1(y)$ and $\sigma_1(y) = \sigma'_1(x)$. As m is at least 3, there exists $r \leq m$ such that $i_r \notin \{x, y\}$ and $j_r \notin \{\sigma_1(x), \sigma_1(y)\}$. Without loss of generality assume that $r = 1$. So we have description of σ_1 and σ'_1 as follows: $\sigma_1 : k \to l, x \to \alpha, y \to \beta, i_1 \to \gamma, \delta \to j_1, \ldots$ and $\sigma'_1 : k \to l, x \to \beta, y \to \alpha, i_1 \to \gamma, \delta \to j_1, \ldots$.

Taking $k_1 = k, k_2 = i_1, k_3 = \delta$, x as x and y as y, generate permutations $\sigma_2, \ldots, \sigma_6, \sigma'_2, \ldots, \sigma'_6$ as defined before lemma 3.2.9. Vertices corresponding to each of these permutations belong to S. Hence from lemma 3.2.9, $P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma'_1} \in LS(S)$. $\square$

Corollary 3.2.14. *Given any $P^{[2]}_{\sigma*}$ in T_1, each $P^{[2]}_{\sigma}$ in T_1 belongs to $LS(\{P^{[2]}_{\sigma*}\} \cup S)$.*

Lemma 3.2.15. *Let $n \geq 5$. Then $T_{3,2} \subset LS(T_1 \cup S)$.*

Proof. Consider any arbitrary permutation, σ, with the corresponding vertex belonging to $T_{3,2}$. Let $\beta = \sigma^{-1}(l)$ and γ be any arbitrary element from $[n]\setminus\{k, i_1, i_2, \beta\}$. The description of σ is: $k \to \alpha, i_1 \to j_1, i_2 \to j_2, \beta \to l, \gamma \to \delta$ and all other maps are different from (i_p, j_p) for any p. Our goal is to show that $P^{[2]}_{\sigma} \in LS(T_1 \cup S)$. Consider two cases.

Case: $(\beta, \alpha) \neq (i_p, j_p)$ for any p. Take $\sigma_1 = \sigma, k_1 = i_1, k_2 = i_2, k_3 = \gamma, x = k, y = \beta$. All the vertices corresponding to permutations $\sigma_2, \ldots, \sigma_6, \sigma'_1, \ldots, \sigma'_6$ generated with these parameters belong to $S \cup T_1$. From lemma 3.2.9 $P^{[2]}_{\sigma_1} \in LS(T_1 \cup S)$.

Case: $(\beta, \alpha) = (i_3, j_3)$. In this case $\sigma : k \to j_3, i_1 \to j_1, i_2 \to j_2, i_3 \to l, \gamma \to \delta$. Take $\sigma_1 = \sigma, k_1 = k, k_2 = i_1, k_3 = i_2, x = i_3, y = \gamma$. Then we see that $P^{[2]}_{\sigma_1}$ and $P^{[2]}_{\sigma'_1}$ both belong to $T_{3,2}$ and the vertices corresponding to the remaining ten permutations belong to S. So $P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma'_1} \in LS(S)$. Now from the first case $P^{[2]}_{\sigma'_1}$ belongs to $LS(T_1 \cup S)$. Therefore $P^{[2]}_{\sigma_1}$ also belongs to $LS(T_1 \cup S)$. $\square$

Lemma 3.2.16. *Let $n \geq 5$. Then $T_{2,3} \subset LS(T_1 \cup S)$.*

Proof. Let $P_\sigma^{[2]}$ be an arbitrary element of $T_{2,3}$. We will express $P_\sigma^{[2]}$ as a linear combination of some members of $T_{3,2} \cup S$. The rest will follow from lemma 3.2.15.

Without loss of generality assume that the given permutation σ in $T_{2,3}$ maps $k \to l, i_1 \to j_1, i_2 \to j_2, i_3 \to j_3$. Also let σ map $\alpha \to \beta$ for some $\alpha \notin \{k, i_1, i_2, i_3\}$. Now generate the permutations $\sigma_2, \ldots, \sigma_6, \sigma_1', \ldots, \sigma_6'$ with parameters $\sigma_1 = \sigma, k_1 = i_2, k_2 = i_3, k_3 = \alpha, x = k, y = i_1$. See that $P_{\sigma_1'}^{[2]} \in T_{3,2}$ and the remaining ten permutations belongs to S. So $P_{\sigma_1}^{[2]} - P_{\sigma_1'}^{[2]} \in LS(S)$. From lemma 3.2.15, $P_\sigma^{[2]} \in LS(S \cup T_1)$. □

Lemma 3.2.17. *Let $n \geq 6$. Given any $P_\sigma^{[2]}$ in $T_{3,r}$ with $r > 2$, it can be expressed as a linear combination of elements in $T_1 \cup S$.*

Proof. Let $P_{\sigma_1}^{[2]} \in T_{3,r}$ with $r \geq 3$. Assume that σ_1 maps $\alpha \to l, k \to \gamma, i_1 \to j_1, i_2 \to j_2, i_3 \to j_3, \ldots, i_r \to j_r$. If $r = 3$, then consider the parameters $x = i_3, y = \alpha, k_1 = i_1, k_2 = i_2, k_3 = \beta \notin \{i_1, i_2, i_3, k, \alpha\}$. Otherwise let $x = i_4, y = \alpha, k_1 = i_1, k_2 = i_2, k_3 = i_3$. Generate $\sigma_2, \ldots, \sigma_6, \sigma_1', \ldots, \sigma_6'$. Corresponding vertices either belong to S or to $\cup_{2 \leq x < r} T_{3,x}$. So using induction on r and the result of lemma 3.2.15 as the base case, lemma 3.2.9 gives that $P_{\sigma_1}^{[2]} \in LS(T_1 \cup S)$. □

Similarly following lemma can also be proved.

Lemma 3.2.18. *Let $n \geq 6$. Given any $P_\sigma^{[2]}$ in $T_{2,r}$ with $r > 3$, it can be expressed as a linear combination of elements in $T_1 \cup S$.*

Lemmas 3.2.15-3.2.18 lead to the following corollary.

Corollary 3.2.19. *If $n \geq 6$, then $T_2 \cup T_3 \subset LS(T_1 \cup S)$.*

Theorem 3.2.20. *If $n \geq 6$, then inequality (3.2) defines a facet of $\mathcal{B}^{[2]}$.*

Proof. From corollaries 3.2.14 and 3.2.19 every vertex in T can be expressed as a linear combination of a fixed vertex in T and the vertices in S. Now the result follows from lemma 3.2.7. □

The number of facets defined by this theorem is $\sum_{m=3}^{n-3} \frac{n^2(n-1)^2 \ldots (n-m)^2}{m!}$.

3.3 Conclusion

In this chapter we presented several facets of the $\mathcal{B}^{[2]}$ polytope. These include two new families of facets apart from those which are known in the QAP literature. We also defined a general inequality that captures all these facets. In chapter 6 we will reformulate this inequality using the fact that every matrix in $\mathcal{B}^{[2]}$ is positive semidefinite.

The conditions given in Theorem 3.2.3 can be included in LP-GI because their count is polynomial in the size of the graphs. Therefore $\mathcal{P}_{G_1G_2}$ for non-isomorphic G_1, G_2, if non-empty, must violate one or more of the inequalities associated with the remaining facets of $\mathcal{B}^{[2]}$. These include the three exponential families defined in this chapter as well as the facets which are still unknown. In the next chapter we will analyze Algorithm 1 when every point in the feasible region violates one or more of the inequalities given by 3.2.4, 3.2.5 and 3.2.6.

Chapter 4

Non-Isomorphism Detection

4.1 Introduction

In this chapter we will first show that under a reasonable assumption, Algorithm 1 determines non-isomorphism in polynomial time. Later we will present a modified version of Algorithm 1 to handle the general case efficiently. We will show that the new algorithm runs in time that is exponential in a certain geometric parameter.

In section 4.2 we will define a partial ordering on certain hyperplanes that support the polytope $\mathcal{B}^{[2]}$. These include those that define the facets described in chapter 3. We exploit this ordering to show the polynomiality of Algorithm 1 in section 4.3.

We begin with a review of the results of the previous chapters. We have seen that the feasible region of LP-GI is $\mathcal{P}_{G_1G_2}$ which contains the polytope $\mathcal{B}^{[2]}_{G_1G_2}$. Polytope $\mathcal{P}$ is the intersection of the half spaces $Y_{ij,kl} \geq 0$ and the affine plane P. Therefore its facets are due to the planes $Y_{ij,kl} = 0$ (called zero planes). Each $\mathcal{P}_{G_1G_2}$ is a face of $\mathcal{P}$ that is restricted to the planes $Y_{ij,kl} = 0$ where $\{i,k\} \in E_1, \{j,l\} \notin E_2$ or $\{i,k\} \notin E_1, \{j,l\} \in E_2$. Two graphs are isomorphic if and only if $\mathcal{P}_{G_1G_2} \cap \mathcal{B}^{[2]} = \mathcal{B}^{[2]}_{G_1G_2} \neq \emptyset$. So for non-isomorphic graphs, if $\mathcal{P}_{G_1G_2}$ is non-empty, then it must belong to $\mathcal{P} \setminus \mathcal{B}^{[2]}$.

In chapter 3 we have seen that the zero planes $Y_{ij,kl} = 0$ also define the facets of $\mathcal{B}^{[2]}$. We refer to these as the *trivial* facets of $\mathcal{B}^{[2]}$. Several non-trivial facets were also presented in chapter 3. Clearly, for the case of non-isomorphic graphs, the feasible region must violate the inequalities associated with one or more non-trivial facets of $\mathcal{B}^{[2]}$.

Since $P^{[2]}_\sigma$ are the only integral (0/1) points in P and they form the vertices of $\mathcal{B}^{[2]}$, every point in $\mathcal{P}_{G_1G_2}$ must be non-integral if the input graphs are non-isomorphic. Algorithm 1 exhaustively searches for a 0/1 point in the feasible region of LP-GI. We investigate the time complexity of the algorithm in this chapter.

4.2 Partial Ordering on Supporting Planes of $\mathcal{B}^{[2]}$

In chapter 3 we described several families of facet defining inequalities. In this section we revisit the three exponential families and relax some of the conditions so that each family now also includes inequalities that define some of the lower dimensional faces of $\mathcal{B}^{[2]}$. Then we define a partial ordering on these supporting planes/inequalities. Let X be a supporting plane such that the corresponding inequality is violated by some point p in the feasible region. Also, let the point p does not violate any inequality that lies at a level lower than that of X in the ordering. Then we call X a *minimal violated inequality* for point p. We show in the next section that Algorithm 1 determines non-isomorphism in polynomial time if there exists a common minimal violated inequality for all points in the feasible region of LP-GI. In this section we also show that all the minimal inequalities, with respect to the partial ordering, are satisfied by all points in $\mathcal{P}$. Hence if a point in $\mathcal{P}_{G_1G_2}$ violates any one of the inequalities belonging to one of the three exponentially large families described in Chapter 3, then there must exist a minimal violated inequality for that point.

The first family of inequalities, described in Theorem 3.2.4, is given below. Let $i_1, \ldots, i_m, k$ be $m+1$ distinct indices. Similarly let $j_1, \ldots, j_m, l$ be distinct indices. Let $A = \{(i_1, j_1), \ldots, (i_m, j_m)\}$. Then the inequality $Q_1(k, l, A)$ is given by

$$\sum_{(i,j)\in A} Y_{ij,kl} \leq Y_{kl,kl} + \sum_{(i,j)\neq(i',j')\in A} Y_{ij,i'j'}. \tag{4.1}$$

Let $A' \subseteq A$. Then we define $Q_1(k, l, A') \prec Q_1(k, l, A)$. Note that the inequalities in this family corresponding to $|A| = 1$, i.e., $Y_{ij,kl} \leq Y_{kl,kl}$ for all $i, j \in [n]$, cannot be violated by any point in $\mathcal{P}$. The same is true for inequalities corresponding to $A = \emptyset$ i.e., $0 \leq Y_{kl,kl}$. An inequality corresponding to $m \geq 2$ can however by violated by a point in $\mathcal{P}\setminus \mathcal{B}^{[2]}$. Therefore if an inequality of this class is violated, then there will be a minimal inequality which will be violated and all the lower inequalities will be satisfied. Note that the facets in theorem 3.2.4 require $m \geq 3$. So here we have relaxed that condition to also include the inequalities corresponding to the case of

$m = 2$, which define lower dimensional faces of $\mathcal{B}^{[2]}$.

The next are the one-box inequalities, described in Theorem 3.2.5. Let P and Q be sets of indices and $\beta \geq 0$ be an integer, then the inequality $Q_2(P, Q, \beta)$ is

$$(\beta - 1) \sum_{(ij) \in P \times Q} Y_{ij,ij} \leq \sum_{(ij),(kl) \in P \times Q, i<k} Y_{ij,kl} + (\beta^2 - \beta)/2. \tag{4.2}$$

It may be noted that these correspond to facets when $\beta + 1 \leq \min\{|P|, |Q|\}, |P| + |Q| \leq n - 3 + \beta, \beta \geq 2$. Here we consider the inequality without these restrictions.

If $P' \subseteq P$ and $Q' \subseteq Q$, then ordering is defined as $Q_2(P', Q', \beta) \prec Q_2(P, Q, \beta)$ and if $0 \leq \beta' \leq \beta$, then $Q_2(P, Q, \beta') \prec Q_2(P, Q, \beta)$. In this case the lowest level inequalities correspond to $\beta = 0, |P| = |Q| = 2$, i.e., $0 \leq \sum_{(ij) \in P \times Q} Y_{ij,ij} + \sum_{(ij),(kl) \in P \times Q, i<k} Y_{ij,kl}$ which cannot be violated.

The last family of inequalities discussed in Chapter 3 is the two-box inequality. Let Q, P_1, and P_2 be index sets such that $P_1 \cap P_2 = \emptyset$ and β be any integer.

Then the inequality $Q_3(P_1, P_2, Q, \beta)$ is

$$\begin{aligned} &-(\beta - 1) \sum_{(ij) \in P_1 \times Q} Y_{ij,ij} + \beta \sum_{(ij) \in P_2 \times Q} Y_{ij,ij} + \sum_{(ij),(kl) \in P_1 \times Q, i<k} Y_{ij,kl} \\ &+ \sum_{(ij),(kl) \in P_2 \times Q, i<k} Y_{ij,kl} - \sum_{(ij) \in P_1 \times Q, (kl) \in P_2 \times Q} Y_{ij,kl} + \frac{\beta^2 - \beta}{2} \geq 0. \end{aligned} \tag{4.3}$$

Once again these inequalities define facets of $\mathcal{B}^{[2]}$ when parameters P_1, P_2, Q and β satisfy certain conditions. However, we consider $Q_3(P_1, P_2, Q, \beta)$ without any of these conditions. Note that the inequalities still define planes that support lower dimensional faces of $\mathcal{B}^{[2]}$.

Observe that if $\beta \geq 0$, then $Q_3(P_1, \emptyset, Q, \beta) = Q_2(P_1, Q, \beta)$ and if $\beta < 0$, then $Q_3(\emptyset, P_2, Q, \beta) = Q_2(P_2, Q, -\beta + 1)$. Let $i_1 \in P_1$ and $i_2 \in P_2$ be arbitrary indices. Let $P'_1 = P_1 \backslash \{i_1\}$ and $P'_2 = P_2 \backslash \{i_2\}$. Then we define $Q_3(P'_1, P'_2, Q, \beta) \prec Q_3(P_1, P_2, Q, \beta)$. The partial ordering will be the transitive closure of this relation. In the case of the 2-box family of inequalities, the inequalities at the lowest level in the partial ordering correspond to one of the following: (a) a 1-box inequality, (b) $Q_3(\emptyset, P_2, Q, \beta)$ where $\beta \geq 0$, or (c) $Q_3(P_1, \emptyset, Q, \beta)$ where $\beta < 0$. The cases (b) and (c) cannot be violated by any point in $\mathcal{P}$. Thus all the inequalities at the lowest level in the partial ordering will be non-violating if all the 1-box inequalities are satisfied by the solution face $\mathcal{P}_{G_1 G_2}$.

4.3 Polynomiality of Algorithm 1

An important question, which influences the performance of Algorithm 1 is whether a single minimal violated inequality exists for all points in the feasible region of LP-GI. In this section we will assume that such an inequality exists and then we consider three cases where the minimal violated inequality belongs to one of the three families described in section 4.2. Subsequently in section 4.4 we will drop this assumption and consider the general case.

4.3.1 A Minimal Violated Inequality of Type (4.1)

Lemma 4.3.1. *If the feasible region of LP-GI violates $Q_1(k, l, A)$ and satisfies $Q_1(k, l, A')$ where $A' = A \setminus \{(i, j)\}$ for each $(i, j) \in A$, then Algorithm 1 detects non-isomorphism in polynomial time.*

Proof. Suppose the solution face for a non-isomorphic pair violates an inequality defined by $\sum_{r\in[m]} Y_{i_rj_r,kl} \leq Y_{kl,kl} + \sum_{r<s\in[m]} Y_{i_rj_r,i_s,j_s}$, then each solution point will satisfy $\sum_{r=1}^{m} Y_{i_rj_r,kl} > Y_{kl,kl} + \sum_{r<s} Y_{i_rj_r,i_s,j_s}$. Let a be an arbitrary element of $[m]$ and define $S = [m] \setminus \{a\}$. Then we have the inequality $\sum_{r\in S} Y_{i_rj_r,kl} \leq Y_{kl,kl} + \sum_{r<s\in S} Y_{i_rj_r,i_s,j_s}$ which must be satisfied by every point in the feasible region of LP-GI. Subtracting the second from the first we have $Y_{i_aj_a,kl} > \sum_{r\in S} Y_{i_rj_r,i_aj_a} \geq 0$. The last inequality is due to the non-negativity condition in the linear program. This implies that during a call of $SearchVar()$ when $Y_{i_aj_a,kl}$ will be set to zero in the algorithm, the linear program will declare it infeasible. Hence $Y_{i_aj_a,kl}$ will be set to 1. Since a is any arbitrary index, eventually $Y_{i_aj_a,kl}$ will be set to 1 for each $a \in [m]$. These will force $Y_{kl,kl}$ and $Y_{i_rj_r,i_sj_s} \forall\, r, s \in [m]$ to 1. Then the first inequality will be violated since the left hand side will be m but the right hand side will be $1 + \binom{m}{2}$ where $m \geq 2$. □

4.3.2 A Minimal Violated inequality of Type (4.2)

4.3.2.1 Restriction to Facets

Similar to a minimal violated inequality we define the notion of a minimal violated *facet* inequality. Here only the facet defining inequalities are considered in the partial ordering. We consider two separate cases. In the first case the minimal violated facet inequality has $max\{|P|, |Q|\} > \beta+1$, whereas in the second case the minimal violated facet inequality has $|P| = |Q| = \beta+1$ and $\beta > 2$. The minimal inequalities in the partial ordering correspond to the case when $\beta = 2$ and $|P| = |Q| = 3$. Since we want them to be satisfied by every point in the feasible region, we add all these inequalities to the linear program. Note that the number of these inequalities is polynomial in the size of the graphs, hence adding them to LP-GI will not affect its poly-time solvability.

Lemma 4.3.2. *If the solution face violates $Q_2(P, Q, \beta)$ and satisfies $Q_2(P', Q', \beta')$ such that either (i) $|P| > \beta+1$ and $P' = P\setminus\{i\}$ for arbitrary $i \in P$, $Q' = Q, \beta' = \beta$, or (ii) $|Q| > \beta + 1$ and $Q' = Q \setminus \{j\}$ for arbitrary $j \in Q$, $P' = P, \beta' = \beta$, then Algorithm 1 detects non-isomorphism in polynomial time.*

Proof. Suppose the inequality $(\beta - 1)\sum_{(ij)\in P\times Q} Y_{ij,ij} \leq \sum_{(ij),(kl)\in P\times Q,i<k} Y_{ij,kl} + (1/2)(\beta^2 - \beta)$ is violated. Note that the roles of P and Q can be interchanged without affecting the inequality. Hence it is sufficient to consider only one case, namely, $|P| > \beta + 1$.

$$(\beta - 1) \sum_{(ij)\in P\times Q} Y_{ij,ij} > \sum_{(ij),(kl)\in P\times Q,i<k} Y_{ij,kl} + (1/2)(\beta^2 - \beta) \tag{4.4}$$

Let $i_0 \in P$ and $j_0 \notin Q$. Define $P' = P \setminus \{i_0\}$. Suppose during a call of $SearchVar()$ the algorithm forces $Y_{i_0j_0,i_0j_0}$ to 1. Since P' and Q both have at least $\beta + 1$ elements, the solution must satisfy the following inequality

$$(\beta - 1) \sum_{(ij)\in P'\times Q} Y_{ij,ij} \leq \sum_{(ij),(kl)\in P'\times Q,i<k} Y_{ij,kl} + (1/2)(\beta^2 - \beta). \tag{4.5}$$

(4.4) minus (4.5) gives $(\beta - 1)\sum_{j\in Q} Y_{i_0j,i_0j} > \sum_{j\in Q}\sum_{(kl)\in P'\times Q} Y_{i_0j,kl}$.

Since $Y_{i_0j_0,i_0j_0} = 1$ where $j_0 \notin Q$, $\sum_{j\in Q} Y_{i_0j,i_0j} = 0$. The non-negativity condition implies that the right-hand-side is non-negative so we conclude that $0 > 0$. As $Y_{i_0j_0,i_0j_0} = 1$ renders the problem infeasible, the algorithm will set $Y_{i_0j,i_0j} = 0$ for all $j \notin Q$. As i_0 was an arbitrary element of P, eventually the algorithm will set $Y_{ij,ij} = 0$ for all $i \in P$ and all $j \notin Q$.

Next consider an arbitrary $(i_0j_0) \in P \times Q$. Suppose algorithm sets $Y_{i_0j_0,i_0j_0} = 1$. Let $P' = P \setminus \{i_0\}$. Then the violated inequality (4.4) reduces to $(\beta - 1)(1 + \sum_{(ij)\in P'\times Q} Y_{ij,ij}) > \sum_{(ij),(kl)\in P'\times Q,i<k} Y_{ij,kl} + \sum_{j\in Q}\sum_{(kl)\in P'\times Q} Y_{i_0j,kl} + \frac{\beta^2-\beta}{2}$.

Subtracting (4.5) from the above inequality gives $(\beta - 1) > \sum_{j\in Q}\sum_{(kl)\in P'\times Q} Y_{i_0j,kl}$. Since $Y_{i_0j,kl} = 0$ for all $j \neq j_0$, $\sum_{j\notin Q}\sum_{(kl)\in P'\times Q} Y_{i_0j,kl} = 0$. Adding this term to the right hand side of the inequality we get $(\beta - 1) > \sum_{j\in[n]}\sum_{(kl)\in P'\times Q} Y_{i_0j,kl} = \sum_{(kl)\in P'\times Q} Y_{kl,kl}$. From the first part of the proof, $Y_{kl,kl} = 0$ for any $k \in P$ and $l \notin Q$. So we have $\sum_{(kl)\in P'\times Q} Y_{kl,kl} = \sum_{(kl)\in P'\times[n]} Y_{kl,kl} = |P'| > \beta + 1 - 1 = \beta$. It reduces to infeasible $\beta - 1 > \beta$, which leads the algorithm to set $Y_{i_0j_0,i_0j_0} = 0$. Hence eventually $Y_{ij,ij}$ is set to zero for all $(ij) \in P \times Q$. Combining with the fact that $Y_{ij,ij} = 0$ for all $i \in P, j \notin Q$, we have $1 = \sum_{j\in[n]} Y_{ij,ij} = 0$ for any $i \in P$. Hence the algorithm will report an empty feasible region and conclude that the graphs are non-isomorphic. At no stage is the algorithm required to invoke two calls, one with $x = 0$ and the other with $x = 1$ for any variable x. So we see that the feasible region is zero-one reducible and the algorithm requires polynomial time. □

Lemma 4.3.3. *If the solution face violates $Q_2(P, Q, \beta)$ where $|P| = |Q| = \beta+1, \beta > 2$ and satisfies $Q_2(P', Q', \beta')$ such that either (i) $P' = P \setminus \{i\}$ for arbitrary $i \in P$, $Q' = Q, \beta' = \beta - 1$, or (ii) $Q' = Q \setminus \{j\}$ for arbitrary $j \in Q$, $P' = P, \beta' = \beta - 1$, then Algorithm 1 detects non-isomorphism in polynomial time.*

Proof. The violation of $(\beta-1)\sum_{(ij)\in P\times Q} Y_{ij,ij} \leq \sum_{(ij),(kl)\in P\times Q, i<k} Y_{ij,kl} + (1/2)(\beta^2 - \beta)$ gives inequality (4.4), given in the last proof.

Let $i_0 \in P$ and $P' = P\setminus\{i_0\}$. Then the solution must satisfy the inequality with parameters $P', Q, \beta - 1$. So we have

$$(\beta - 2)\sum_{(ij)\in P'\times Q} Y_{ij,ij} \leq \sum_{(ij),(kl)\in P'\times Q, i<k} Y_{ij,kl} + (1/2)((\beta-1)^2 - (\beta - 1)) \tag{4.6}$$

(4.4) minus (4.6) gives

$$\sum_{(ij)\in P'\times Q} Y_{ij,ij} + (\beta - 1)\sum_{j\in Q} Y_{i_0j,i_0j} > \sum_{j\in Q}\sum_{(kl)\in P'\times Q} Y_{i_0j,kl} + (\beta - 1). \tag{4.7}$$

Since $(\beta-1)\sum_{j\in Q} Y_{i_0j,i_0j} = (\beta-1) - (\beta-1)\sum_{j\notin Q} Y_{i_0j,i_0j}$, the inequality transforms to $\sum_{(ij)\in P'\times Q} Y_{ij,ij} > (\beta - 1)\sum_{j\notin Q} Y_{i_0j,i_0j} + \sum_{j\in Q}\sum_{(kl)\in P'\times Q} Y_{i_0j,kl} = (|P'| - 1)\sum_{j\notin Q} Y_{i_0j,i_0j} + \sum_{j\in Q}\sum_{k\in P'}\sum_{l\in Q} Y_{i_0j,kl}$, because $\beta + 1 = |P| = |P'| + 1$.

For Y is a solution of the LP, $Y_{i_0j,i_0j} = \sum_{l\in[n]} Y_{i_0j,kl}$ for any k. So $|P'|\sum_{j\notin Q} Y_{i_0j,i_0j} = \sum_{k\in P'}\sum_{j\notin Q}\sum_{l\in[n]} Y_{i_0j,kl}$. Plugging this equation in the previous inequality we get $\sum_{(ij)\in P'\times Q} Y_{ij,ij} > -\sum_{j\notin Q} Y_{i_0j,i_0j} + \sum_{k\in P'}\sum_{l\in[n]}\sum_{j\notin Q} Y_{i_0j,kl} + \sum_{k\in P'}\sum_{l\in Q}\sum_{j\in Q} Y_{i_0j,kl}$. Combining the last two terms, ignoring $l \notin Q$ terms due to non-negativity, we get $\sum_{(ij)\in P'\times Q} Y_{ij,ij} > -\sum_{j\notin Q} Y_{i_0j,i_0j} + \sum_{(kl)\in P'\times Q}\sum_{j\in[n]} Y_{i_0j,kl} = -\sum_{j\notin Q} Y_{i_0j,i_0j} + \sum_{k\in P'}\sum_{l\in Q} Y_{kl,kl}$. It simplifies to $\sum_{j\notin Q} Y_{i_0j,i_0j} > 0$.

If the algorithm sets $Y_{i_0j,i_0j} = 1$ for some $j \in Q$, then the above inequality will reduce to $0 > 0$ making it infeasible. So eventually algorithm will set $Y_{ij,ij} = 0$ for all $(ij) \in P \times Q$. This will make (4.4) infeasible. Again we find that the feasible region of LP-GI is zero-one reducible. $\square$

Lemmas 4.3.2 and 4.3.3 lead to the following corollary.

Corollary 4.3.4. *If the solution of LP-GI for a non-isomorphic pair of graphs violates $Q_2(P,Q,\beta)$ and satisfies $Q_2(P',Q',\beta')$ such that $Q_2(P',Q',\beta') \prec Q_2(P,Q,\beta)$ and each $Q_2(P,Q,\beta)$ defines a facet of $\mathcal{B}^{[2]}$, then Algorithm 1 will detect non-isomorphism in polynomial time.*

4.3.2.2 General 1-box Inequality

Now we withdraw the restriction to facets and consider all the 1-box inequalities subject to the partial ordering defined in section 4.2.

Lemma 4.3.5. *If the solution face violates $Q_2(P,Q,\beta)$ and satisfies $Q_2(P',Q',\beta')$ for all $Q_2(P',Q',\beta') \prec Q_2(P,Q,\beta)$, then Algorithm 1 detects non-isomorphism in polynomial time.*

The proof of this lemma is same as that of Lemma 4.3.2 while ignoring the restriction $min\{|P|,|Q|\} \geq \beta + 1$.

4.3.3 A Minimal Violated Inequality of Type (4.3)

In this case we directly consider the unrestricted inequality because the base case of restricted inequality (associated with facets) may not always hold true for all points in $\mathcal{P}$ and their number is not polynomial so that we cannot incorporate them into LP-GI, forcing them to hold true. In the case of unrestricted inequality Q_3 the base cases always hold true provided the feasible region satisfies all the 1-box inequalities. If it fails 1-box inequality, then the minimal violated inequality will be a Q_2 and we can use the previous section's argument to prove polynomiality of the algorithm.

Lemma 4.3.6. *If the solution face violates $Q_3(P_1, P_2, Q, \beta)$ and satisfies $Q_3(P_1', P_2', Q, \beta)$ where $P_1' = P_1 \setminus \{i\}$ for arbitrary $i \in P_1$ and $P_2' = P_2 \setminus \{j\}$ for arbitrary $j \in P_2$ and also satisfies all $Q_2(P, Q, \beta)$, then Algorithm 1 detects non-isomorphism in polynomial time.*

Proof. Given that a 2-box inequality (P_1, P_2, Q, β) is violated by the solution face, every solution point satisfies

$$\begin{aligned} &-(\beta-1)\sum_{(ij)\in P_1\times Q} Y_{ij,ij} + \beta \sum_{(ij)\in P_2\times Q} Y_{ij,ij} + \sum_{(ij),(kl)\in P_1\times Q, i<k} Y_{ij,kl} \\ &+ \sum_{(ij),(kl)\in P_2\times Q, i<k} Y_{ij,kl} - \sum_{(ij)\in P_1\times Q,(kl)\in P_2\times Q} Y_{ij,kl} + \frac{\beta^2-\beta}{2} < 0. \end{aligned} \tag{4.8}$$

Let $i_0 \in P_1$ and $i_0' \in P_2$ be two arbitrary indices. Let $P_1' = P_1 \setminus \{i_0\}$ and $P_2' = P_2 \setminus \{i_0'\}$. Then every solution point must also satisfy the inequality corresponding to (P_1', P_2', Q, β). We have

$$\begin{aligned} &-(\beta-1)\sum_{(ij)\in P_1'\times Q} Y_{ij,ij} + \beta \sum_{(ij)\in P_2'\times Q} Y_{ij,ij} + \sum_{(ij),(kl)\in P_1'\times Q, i<k} Y_{ij,kl} \\ &+ \sum_{(ij),(kl)\in P_2'\times Q, i<k} Y_{ij,kl} - \sum_{(ij)\in P_1'\times Q,(kl)\in P_2'\times Q} Y_{ij,kl} + \frac{\beta^2-\beta}{2} \geq 0. \end{aligned} \tag{4.9}$$

Case 1: In the algorithm when $Y_{i_0j_0,i_0'j_0'}$ is set to 1, where $j_0, j_0' \in Q, j_0 \neq j_0'$, (4.8) minus (4.9) gives $0 < 0$ which is absurd. Hence algorithm will set $Y_{ij,i'j'} = 0$ for all $i \in P_1, i' \in P_2, j, j' \in Q$.

Case 2: When $SearchVar()$ sets $Y_{i_0j_0,i_0'j_0'} = 1$, where $j_0 \notin Q, j_0' \in Q$. Then (4.8) minus (4.9) gives $\beta + \sum_{(i,j)\in P_2'\times Q} Y_{ij,ij} < 0$, where we used the result of the previous case, i.e., $Y_{ij,kl} = 0$ for all $ij \in P_1 \times Q$ and $kl \in P_2 \times Q$. Note that it is impossible if $\beta \geq 0$.

Case 3: When $SearchVar()$ sets $Y_{i_0j_0,i_0'j_0'} = 1$, where $j_0 \in Q, j_0' \notin Q$. Then (4.8) minus (4.9) gives $-(\beta - 1) + \sum_{(i,j)\in P_1'\times Q} Y_{ij,ij} < 0$, which is impossible if $\beta \leq 1$.

If $\beta \geq 0$, then combining the results of cases 1 and 2 we see that the algorithm sets $Y_{ij,kl} = 0$ for all $i \in P_1, k \in P_2, j \in [n], l \in Q$ which is same as setting $Y_{ij,ij} = 0$ for all $ij \in P_2 \times Q$. Similarly we can see that if $\beta < 0$, then the algorithm will set $Y_{ij,ij} = 0$ for all $(ij) \in P_1 \times Q$.

Plugging these values in inequality (4.8) we have following simplified cases

$$0 \le \beta \le 1 : \text{Impossible. Recall that } \beta \in \mathbb{Z}. \tag{4.10}$$

$$\beta \le -1 : \beta \sum_{(ij) \in P_2 \times Q} Y_{ij,ij} + \sum_{(ij),(kl) \in P_2 \times Q, i<k} Y_{ij,kl} + \frac{\beta^2 - \beta}{2} < 0. \tag{4.11}$$

$$\beta \ge 2 : -(\beta - 1) \sum_{(ij) \in P_1 \times Q} Y_{ij,ij} + \sum_{(ij),(kl) \in P_1 \times Q, i<k} Y_{ij,kl} + \frac{\beta^2 - \beta}{2} < 0. \tag{4.12}$$

The inequality 4.12 clearly violates $Q_2(P_1, Q, \beta)$ whereas the inequality 4.11 violates $Q_2(P_2, Q, 1-\beta)$. But we have assumed that all the Q_2 inequalities are satisfied by the solution face. This leads to a contradiction and hence at this stage the algorithm will find no solution and conclude that the graphs are non-isomorphic.

For the completeness sake we will prove that these inequalities cannot hold true even if we assume that only those Q_2 inequalities that are associated with facets hold true. For this we need the following additional restrictions: (i) $|P_1|, |P_2| \ge 3$, (ii) if $\beta \ge 0$ and $\min\{|Q|, |P_1|\} \ge \beta + 1$ then $|Q| + |P_1| + 3 \le n + \beta$, (iii) if $\beta < 0$ and $\min\{|Q|, |P_2|\} \ge 2 - \beta$ then $|Q| + |P_2| + 3 \le n + 1 - \beta$.

We will first consider inequality (4.12). If $|P_1|, |Q| \ge \beta + 1$, then (4.12) implies that the 1-box inequality corresponding to (P_1, Q, β) is violated. But that is not possible due to the assumption that all Q_2 inequalities associated with facets hold true for the feasible region. So the only case that needs to be considered is $\min\{|P_1|, |Q|\} \le \beta$.

First assume that $|P_1| \le |Q|$. Consider the identity $\sum_{(ij),(kl) \in P_1 \times Q, i<k} Y_{ij,kl} = |P_1|(|P_1|-1)/2 + \sum_{(ij),(kl) \in P_1 \times \overline{Q}, i<k} Y_{ij,kl} - (|P_1|-1) \sum_{(ij) \in P_1 \times \overline{Q}} Y_{ij,ij}$. Plugging into the inequality (4.12) gives $-(\beta-1) \sum_{(ij) \in P_1 \times Q} Y_{ij,ij} + |P_1|(|P_1|-1)/2 + \sum_{(ij),(kl) \in P_1 \times \overline{Q}, i<k} Y_{ij,kl} - (|P_1| - 1) \sum_{(ij) \in P_1 \times \overline{Q}} Y_{ij,ij} + \beta(\beta - 1)/2 < 0$. But the left hand side of the inequality is greater than or equal to $-(\beta-1) \sum_{i \in P_1, j \in [n]} Y_{ij,ij} + (\beta(\beta-1) + |P_1|(|P_1| - 1))/2 = ((\beta - |P_1|)^2 - (\beta - |P_1|))/2 \ge 0$ since $\sum_{i \in P_1, j \in [n]} Y_{ij,ij} = |P_1|$ and $\beta, |P_1|$ are both integral. Hence we find that inequality (4.12) is impossible. The case of $|Q| \le |P_1|$, is handled similarly since P_1 and Q have similar role. So we conclude that inequality (4.12) is impossible.

In case of inequality (4.11) we rewrite it by replacing β by $-(\gamma - 1)$. We get $-(\gamma - 1) \sum_{(ij) \in P_2 \times Q} Y_{ij,ij} + \sum_{(ij),(kl) \in P_2 \times Q, i<k} Y_{ij,kl} + (1/2)(\gamma^2 - \gamma) < 0$, where $\gamma \ge 2$. We can now use the same argument as above to establish that (4.11) is also impossible. □

Theorem 4.3.7. *Algorithm 1 solves the Graph Isomorphism problem in polynomial time if there exists a common minimal violated inequality of type 4.1, 4.2 or 4.3 for all points in the feasible region of LP-GI outside* $\mathcal{B}^{[2]}$*, namely* $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$*.*

4.4 The General Case

In the general case we consider the situation when more than one non-trivial minimal facets are required to separate $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ from $\mathcal{B}^{[2]}$. The assumption here is that $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ is separable from $\mathcal{B}^{[2]}$ exclusively by the presently known facets (described in Chapter 3). Let these minimal facets be $F_1, \ldots, F_{k_0}$ and the regions separated by them be $R_1, R_2, \ldots$ respectively. Clearly these regions need not be mutually exclusive and $\cup_i R_i = \mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$.

So we have the region of $\mathcal{P}_{G_1G_2}$ outside $\mathcal{B}^{[2]}$, divided into subregions such that for each subregion there exists a common minimal violated inequality of type 4.1, 4.2 or 4.3. In this section we describe a procedure that solves the Graph Isomorphism problem in $O(k_0(2n)^{k_0})$ time.

From Section 4.3 we know that each R_i as defined above, is zero-one reducible (Section 2.4.1). Let $x_{i1}, x_{i2}, \ldots$ be a reduction sequence for region R_i, for each i. Let $\alpha_{i1}, \alpha_{i2}, \ldots \in \{0, 1\}$ be the respective values. So we have $R_i|_{x_{i1}=\alpha_{i1}, x_{i2}=\alpha_{i2}, \ldots} = \emptyset$. Also, $R_i|_{x_{i1}=\overline{\alpha_{i1}}} = \emptyset$ for all $i \in [k]$. Clearly, $(\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]})|_{x_{11}=\overline{\alpha_{11}}, x_{21}=\overline{\alpha_{21}}, \ldots, x_{k1}=\overline{\alpha_{k1}}} = \emptyset$. Note that for isomorphic graphs, $\mathcal{P}_{G_1G_2}$ reduces to the convex hull of $\mathcal{P}_\sigma^{[2]}$s for those σ that correspond to the isomorphisms between G_1 and G_2 and are consistent with $x_{11} = \overline{\alpha_{11}}, x_{21} = \overline{\alpha_{21}}, \ldots, x_{k1} = \overline{\alpha_{k1}}$. So in this case $\mathcal{P}_{G_1G_2}|_{x_{11}=\overline{\alpha_{11}}, x_{21}=\overline{\alpha_{21}}, \ldots, x_{k1}=\overline{\alpha_{k1}}} \subseteq \mathcal{B}_{G_1G_2}$.

We will now use the above information to devise a procedure for the Graph Isomorphism problem.

4.4.1 A Generalized Algorithm for GI

In this section we will describe an extended procedure for the general case. This procedure subsumes Algorithm 1.

4.4.1.1 k-SearchVar()

Recall that the procedure $SearchVar()$ in Algorithm 1 returns a variable x and a 0/1 value α such that $x = 1-\alpha$ makes the linear program infeasible. We modify this procedure to now consider all subsets of k variables and return a subset $x_1, x_2, \ldots, x_k$ with the respective values $\alpha_1, \alpha_2, \ldots \alpha_k$ such that the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ becomes empty on setting $x_i = 1 - \alpha_i$ for all $i \in [k]$. For non-isomorphic graphs the linear program becomes infeasible when such an assignment is found. However, for isomorphic graphs the feasible region of LP-GI need not become empty even if all the subregions do become empty. The feasible region in this case is reduced to the convex hull of a subset of the isomorphisms between the input pair that survive the assignments to the k variables (Corollary 2.3.3). So an optimization step is included that would optimize a random direction over this convex hull, thus detecting one of its corners, which will be a $P_\sigma^{[2]}$ where σ is an isomorphism. The invoking procedure can then declare isomorphism and terminate.

The new procedure must also ensure that the assignments $x_i = 1 - \alpha_i$ are consistent with the conditions 2.1c-2.1e for all $i \in [k]$, i.e., the assignment must not lead to a solution outside the plane P. We will refer to this modified procedure as k-$SearchVar()$. Clearly, $SearchVar()$ is same as 1-$SearchVar()$.

4.4.1.2 The Procedure

The number k_0 of minimal violating inequalities that separate the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$, is unknown. So we first determine the value of k_0 by invoking k-$SearchVar()$ for $k = 1, 2, \ldots, k_0$ until a combination of k_0 variables and respective k_0 values either renders the linear program infeasible or concludes that G_1 and G_2 are isomorphic. Subsequent procedure is given in Algorithm 2.

At each step (one full execution of the *While* loop) we reduce the given problem on N variables with parameter k to N problems with strictly smaller parameter (less than or equal to $k-1$) because each reduced problem has at least one of the regions R_i missing from the feasible region of LP-GI. Each iteration of the *While* loop makes assignments to k variables. Figure 4.1 shows the result of the complete run of the *While* loop. When the *While* loop terminates, each of the regions R_i would be empty.

The following recurrence sums up the performance of Algorithm 2. Here $T(1)$

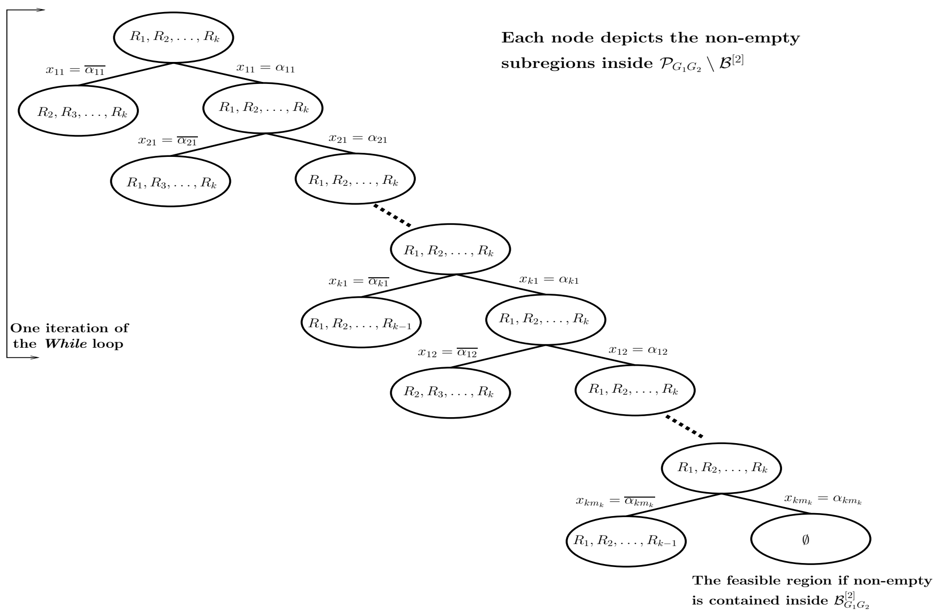

Figure 4.1: Execution of Algorithm 2

is $O(N)$ since each R_i is zero-one reducible. The value of $T(k)$ gives the number of times LP-GI is solved. So the final time complexity will be $poly(N) \cdot T(k)$.

$$T(k) \leq N \cdot T(k-1) + \binom{N}{k} 2^k + \binom{N-k}{k} 2^k + \ldots + \binom{k}{k} 2^k \tag{4.13}$$

On solving the above recurrence, we get $T(k) = O(k \cdot (2N)^k)$. Note that the cost of finding the value of k can be absorbed in this. So $T(k)$ gives the total cost of the procedure.

The following lemma justifies that Algorithm 2 solves the GI problem in $O(k \cdot (2N)^k)$ steps.

Lemma 4.4.1. *Algorithm 2 decides in $O(k \cdot (2N)^k)$ steps if the input pair of graphs is isomorphic or not, where k is the number of subregions into which $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ is divided such that each subregion has a common minimal violated inequality of type 4.1, 4.2 or 4.3.*

Proof. First we will show that the algorithm does not wrongly declare isomorphic graphs as non-isomorphic. Let $\mathcal{P}^{[2]}_\sigma$ be a point in the feasible region for a given pair of isomorphic graphs. Note that the algorithm only assigns values to those variables that appear in the reduction sequences of the subregions of $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$. Let $\gamma_{ij} \in \{0,1\}$ be the values that these variables take in $\mathcal{P}^{[2]}_{\sigma}$. Since the algorithm pursues both the paths corresponding to $x_{ij} = \alpha_{ij}$ and $x_{ij} = \overline{\alpha_{ij}}$ for every x_{ij}, there must exist an assignment that corresponds to γ_{ij}, and for this assignment the linear program returns a non-empty feasible solution. Hence in this case the algorithm cannot output *non-isomorphic*.

What remains to be shown now is that the algorithm outputs *non-isomorphic* when the graphs are not isomorphic. In this case $\mathcal{P}_{G_1G_2} = \cup_i R_i$. Since the algorithm assigns the variables in the zero-one reduction sequences of each of these regions, at the end all the final restricted regions (due to various variable assignments) will be empty. Hence the algorithm will correctly characterize this case as non-isomorphic.

The recurrence relation 4.13 now gives the worst case number of assignments to $O(k \cdot (2n)^k)$. □

Theorem 4.4.2. *Algorithm 2 solves the graph isomorphism problem in $O(k \cdot 2^k \cdot N^{k+c})$ time where $N = O(n^4)$ is the number of variables in LP-GI and k is the number of subregions into which $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ is divided such that each subregion has a common minimal violated inequality of type 4.1, 4.2 or 4.3. Here $O(N^c)$ denotes the cost of running the LP solver.*

Remark: There is a situation that appears to lead to a conflict in building a k-dimensional zero-one reduction sequence for a general case. In case x_{11} and x_{21} are same but $\alpha_{11} = \overline{\alpha_{21}}$, then $\mathcal{P}_{G_1G_2}|_{x_{11}=\overline{\alpha_{11}},x_{21}=\overline{\alpha_{21}},\ldots,x_{k1}=\overline{\alpha_{k1}}}$ is not well defined. However this does not cause any difficulty. To understand consider the $k = 2$ case. Suppose the respective sequences are $x = \alpha, y = \beta, \ldots$ and $x = \overline{\alpha}, z = \gamma, \ldots$. Then $\mathcal{P}_{G_1G_2}|_{x=\alpha,y=\overline{\beta}} = \emptyset$ and $\mathcal{P}_{G_1G_2}|_{x=\overline{\alpha},z=\overline{\gamma}} = \emptyset$. So we will have two $k = 1$ order problems namely $\mathcal{P}_{G_1G_2}|_{x=\overline{\alpha}}$ and $\mathcal{P}_{G_1G_2}|_{x=\alpha}$. Therefore there is no conflict.

Finally we present a bound for k for some special cases.

4.4.1.3 A Bound for k

Suppose the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ violates an inequality of type 4.1. Note that when the left hand side of this inequality takes its maximum value of m, the right hand side has a value of $\binom{m}{2} + 1$. Clearly, less than $\sqrt{m}$ variables $Y_{ij,kl}$ can be assigned a value of 1 before the violated inequality becomes infeasible. So *k-SearchVar*() cannot return a value of k larger than $\sqrt{m}$. Next consider the case when the portion of the feasible region outside $\mathcal{B}^{[2]}$ violates an inequality of type 4.2. Using a similar argument we get a bound of $\sqrt{m\beta}$ on the value of k. Here $m = min\{|P|, |Q|\}$. Finally, for a violated inequality of type 4.3, we have $k = O(\sqrt{|Q||\beta|})$ where $|\beta|$ is the absolute value of β.

4.5 Conclusion

We have formulated GI as a geometric problem. We have defined a partial ordering on the facets of $\mathcal{B}^{[2]}$. In fact the ordering is extended to some smaller dimensional faces as well such that the minimal supporting planes are never violated by any solution point. Then we exploited this fact to show that Algorithm 1 solves the Graph Isomorphism problem in polynomial time as long as the feasible region of LP-GI outside $\mathcal{B}^{[2]}$ violates an inequality X belonging to any of the known classes such that no other inequality $Y \prec X$ is violated by any point in that region. For the general case where such an X does not exist, we extended Algorithm 1 with time complexity exponential in a parameter k that depends on the supporting planes of $\mathcal{B}^{[2]}$ intersecting the feasible region of LP-GI. We believe that the value of k should be small. We also gave an upper bound for k when there exists a single inequality belonging to any of the three exponential families, violated by the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$.

Data: Q: set of constraints; U: set of unassigned variables
Result: *true* if the given pair of graphs are isomorphic; $false$ if the given pair of graphs are non-isomorphic
Function: k-GISolver(Q, U)

```
if LP(Q) is infeasible then
    return false /* Graphs are non-isomorphic */
else
    if LP(Q) is feasible and U = ∅ then
        return true /* Graphs are isomorphic */
    else
        while LP(Q) is feasible do
            (x, α, k) := k-SearchVar(Q, U);
            if k = −1 then
                return true /* Graphs are isomorphic */
            end
            for i := 1 to k do
                val := k-GISolver(Q ∪ {x_i = ᾱ_i}, U \ {x_i});
                if val = true then
                    return true /* Graphs are isomorphic */
                end
                Q := Q ∪ {x_i = α_i};
                U := U \ {x_i};
            end
            if OPT(LP(Q)) = P_σ^[2] then
                return true /* Graphs are isomorphic */
            end
        end
        return false /* Graphs are non-isomorphic */
    end
end
```

Algorithm 2: Algorithm for GI

Chapter 5

There are more Facets

5.1 Introduction

All the known facets of $\mathcal{B}^{[2]}$ are instances of a general inequality

$$\sum_{i,j,k,l} n_{ij} n_{kl} Y_{ij,kl} + (\beta - 1/2)^2 \geq (2\beta - 1) \sum_{ij} n_{ij} Y_{ij,ij} + 1/4 \tag{5.1}$$

where $n_{ab}, \beta \in \mathbb{Z} \; \forall \; a, b$.

It is possible that there are more instances of this inequality which are also facets. Could there be facets of this polytope which are not instances of this inequality? The answer to this question is in the affirmative. In this chapter we will show that there must exist facets which are not instances of the above inequality. Further we derive an inequality that captures all the supporting planes of $\mathcal{B}^{[2]}$ with the expectation that such an inequality may be helpful in identifying those unknown facets.

5.2 Insufficiency of inequality (5.1)

If we substitute $P_\sigma^{[2]}$ for Y in the inequality (5.1) then it simplifies to $(\sum_i n_{i\sigma(i)} - (\beta - 1/2))^2 \geq 1/4$. Since all n_{ab} and β are integers, every $P_\sigma^{[2]}$ satisfies this inequality. Therefore every point in $\mathcal{B}^{[2]}$ also satisfies it. Let R denote that part of the polytope $\mathcal{P}$ which satisfies (5.1) for all integral values of n_{ab} and β. Clearly $\mathcal{B}^{[2]} \subseteq R$. However, $R \subseteq \mathcal{B}^{[2]}$ would imply that $R = \mathcal{B}^{[2]}$ which will hold if and only if every facet of $\mathcal{B}^{[2]}$ is an instance of (5.1). Following lemma gives an alternative characterization of this equality.

Lemma 5.2.1. *Let P_σ denote the row-major vectorization of the corresponding permutation matrix. Following statements are equivalent.*

1. Region of $\mathcal{P}$, satisfying conditions $\sum_{ijkl} x_{ij}x_{kl}\ Y_{ij,kl}\ -(2z-1)\sum_{ij} x_{ij}Y_{ij,ij}$ $+z^2-z \geq 0$ for all $x_{ij}, z \in \mathbb{Z}$ is exactly equal to $\mathcal{B}^{[2]}$.

2. Given any set of permutations I such that $\{P_\sigma^{[2]}|\sigma \in I\}$ is linearly independent, then $\sum_{\sigma\in I}\alpha_\sigma((P_\sigma^T\cdot x)^2-(2z-1)(P_\sigma^T\cdot x))+z^2-z\geq 0$ for all $x\in\mathbb{Z}^{n^2}, z\in\mathbb{Z}$ if and only if $\alpha_\sigma \geq 0\ \forall\ \sigma\in I$ and $\sum_{\sigma\in I}\alpha_\sigma = 1$.

Proof. Assume (2).

Let $Y \in P$. From Lemma 2.2.2 there exists a set I of permutations σ such that $P_\sigma^{[2]}$ for $\sigma \in I$ form a linearly independent set and $Y = \sum_{\sigma\in I}\alpha_\sigma P_\sigma^{[2]}$ and $\sum_{\sigma\in I}\alpha_\sigma = 1$. Then $Y \in \mathcal{B}^{[2]}$ if and only if $\alpha_\sigma \geq 0\,\forall\ \sigma \in I$ if and only if $\sum_{\sigma\in I}\alpha_\sigma((P_\sigma^T\cdot x)^2-(2z-1)(P_\sigma^T\cdot x))+z^2-z\geq 0$ for all $x\in\mathbb{Z}^{n^2}, z\in\mathbb{Z}$ if and only if $\sum_{\sigma\in I}\alpha_\sigma(\sum_{ijkl}P_\sigma^{[2]}(ij,kl)x_{ij}x_{kl}-(2z-1)\sum_{ij}P_\sigma^{[2]}(ij,ij)x_{ij}+z^2-z\geq 0$ for all $x\in\mathbb{Z}^{n^2}, z\in\mathbb{Z}$ if and only if $\sum_{ijkl}Y_{ij,kl}x_{ij}x_{kl}-(2z-1)\sum_{ij}Y_{ij,ij}x_{ij}+z^2-z\geq 0$ for all $x\in\mathbb{Z}^{n^2}, z\in\mathbb{Z}$.

Assume (1).

$\alpha_\sigma \geq 0$ for all $\sigma\in I$ and $\sum_{\sigma\in I}\alpha_\sigma = 1$ if and only if $Y = \sum_{\sigma\in I}\alpha_\sigma P_\sigma^{[2]}, \in \mathcal{B}^{[2]}$ if and only if $\sum_{ijkl} x_{ij}x_{kl}\ Y_{ij,kl}\ -(2z-1)\sum_{ij} x_{ij}Y_{ij,ij}\ +z^2-z\geq 0$ for all $x_{ij}, z\in\mathbb{Z}$ if and only if $\sum_{\sigma\in I}\alpha_\sigma((P_\sigma^T\cdot x)^2-(2z-1)(P_\sigma^T\cdot x))+z^2-z\geq 0$ for all $x\in\mathbb{Z}^{n^2}, z\in\mathbb{Z}$. □

We first prove a useful lemma. In the following let $z-P_\sigma^T\cdot x = z-\sum_{i=1}^n x_{i,\sigma(i)}$ be denoted by y_σ.

Lemma 5.2.2. $\sum_\sigma \alpha_\sigma(y_\sigma^2-y_\sigma) = 0$ *for all* $x\in\mathbb{Z}^{n^2}$, $z\in\mathbb{Z}$ *if and only if* $\sum_\sigma\alpha_\sigma y_\sigma^2 = 0$ *for all* $x\in\mathbb{Z}^{n^2}$, $z\in\mathbb{Z}$.

Proof. (If) Let $S(x,z) = \sum_\sigma \alpha_\sigma y_\sigma^2$. We have $S(x,z)=0$ for all $x\in\mathbb{Z}^{n^2}$ and $z\in\mathbb{Z}$. Select arbitrary $a\in\mathbb{Z}^{n^2}$, $b\in\mathbb{Z}$ and indices i,j. Define a' as $a'_{i'j'} = a_{i'j'}$ if $i'\neq i$ or $j'\neq j$ and $a'_{ij} = a_{ij}+1$. So $S(a',b) = S(a,b)-\sum_{\sigma:\sigma(i)=j}\alpha_\sigma(2y_\sigma(a,b)-1)$. Define a'' in the similar way as a' is defined, except here $a''_{ij} = a_{ij}-1$. Then we get $S(a'',b) = S(a,b)+\sum_{\sigma:\sigma(i)=j}\alpha_\sigma(2y_\sigma(a,b)+1)$. So $(S(a'',b)-S(a',b))/4 = \sum_{\sigma:\sigma(i)=j}\alpha_\sigma y_\sigma(a,b)$. Setting $S(a',b)=S(a'',b)=0$ we have $\sum_{\sigma:\sigma(i)=j}\alpha_\sigma y_\sigma(a,b)=0$. So $\sum_\sigma \alpha_\sigma y_\sigma(a,b) = \sum_j\sum_{\sigma:\sigma(i)=j}\alpha_\sigma y_\sigma(a,b) = 0$. As a,b,i,j is arbitrarily chosen we have $\sum_\sigma\alpha_\sigma y_\sigma = 0$ for all $x\in\mathbb{Z}^{n^2}$ and all $z\in\mathbb{Z}$.

(Only if) This part is trivial because $S(x,z) = 0.5(\ T(x,z) + T(-x,-z))$ where $T(x,z) = \sum_\sigma \alpha_\sigma\ (y_\sigma^2 - y_\sigma)$. $\square$

Let $\tilde{P}_\sigma$ be the (n^2+1)-dimensional vector in which the first n^2 entries are the vectorized P_σ and the last entry is 1. Define $\tilde{P}_\sigma^{[2]} = \tilde{P}_\sigma \cdot \tilde{P}_\sigma^T$.

Lemma 5.2.3. *$\{P_\sigma^{[2]} | \sigma \in I\}$ is linearly independent if and only if $\{\tilde{P}_\sigma^{[2]} | \sigma \in I\}$ is linearly independent.*

Proof. (If) Suppose $\{P_\sigma^{[2]} | \sigma \in I\}$ is linearly dependent. So there exist coefficients $\alpha_\sigma \in \mathbb{R}$, not all zero, such that $\sum_{\sigma\in I} \alpha_\sigma P_\sigma^{[2]}(ij,kl) = 0$ for all i,j,k,l. Since $\tilde{P}_\sigma^{[2]}(ij, n^2+1) = \tilde{P}_\sigma^{[2]}(n^2+1, ij) = P_\sigma^{[2]}(ij,ij)$, we have from the above equation $\sum_{\sigma\in I} \alpha_\sigma \tilde{P}_\sigma^{[2]}(ij,n^2+1) = \sum_{\sigma\in I} \alpha_\sigma \tilde{P}_\sigma^{[2]}(n^2+1,ij) = 0$ for all i,j. Finally, $\sum_{\sigma\in I} \alpha_\sigma \tilde{P}_\sigma^{[2]}(n^2+1,n^2+1) = \sum_{\sigma\in I} \alpha_\sigma(1) = \sum_{\sigma\in I} \alpha_\sigma(\sum_j P_\sigma^{[2]}(ij,ij)) = 0$ where i in the last expression is arbitrary.

(Only if) Follows trivially from the fact that the $n^2 \times n^2$ matrix $P_\sigma^{[2]}$ is a submatrix of $\tilde{P}_\sigma^{[2]}$. $\square$

Lemma 5.2.4. *$\sum_{\sigma\in I} \alpha_\sigma \tilde{P}_\sigma^{[2]} = 0$ if and only if $\sum_{\sigma\in I} \alpha_\sigma q^T \tilde{P}_\sigma^{[2]} q = 0\ \forall\ q \in \mathbb{Q}^{n^2+1}$.*

Proof. (If) Let $f(q) = \sum_{\sigma\in I} \alpha_\sigma q^T \tilde{P}_\sigma^{[2]} q$. So we are given that $f(q) = 0\ \forall\ q \in \mathbb{Q}^{n^2+1}$. We can rewrite $f(q)$ as $f(q) = \sum_{\sigma\in I} \alpha_\sigma(vec(\tilde{P}_\sigma^{[2]})^T vec(qq^T))$ which is same as $f(q) = \sum_{\sigma\in I} \alpha_\sigma(\sum_{\sigma(i)=j,\sigma(k)=l} q_{ij}q_{kl} + q_{n^2+1}^2)$. Consider a vector q' with $q'_{n^2+1} = 1$ and remaining entries zero. Since $f(q') = 0$, we have $\sum_{\sigma\in I} \alpha_\sigma = 0$. Next consider a vector q'' with $q''_{ij} = 1/q''_{kl}$ for some i,j,k,l such that $\sigma(i) = j$ and $\sigma(k) = l$ for $\sigma \in I$. The remaining entries of q'' are all zero. So $f(q'') = 0$ leads to $\sum_{\sigma\in I,\sigma(i)=j,\sigma(k)=l} \alpha_\sigma = 0$. The above argument can be repeated for all i,j,k,l such that $\sigma(i) = j$ and $\sigma(k) = l$ for $\sigma \in I$. Thus, $\sum_{\sigma\in I} \alpha_\sigma \tilde{P}_\sigma^{[2]} = 0$.

(Only if) Consider an arbitrary vector $q \in \mathbb{Q}^{n^2+1}$. We have $f(q) = \sum_{\sigma\in I} \alpha_\sigma$ $(\sum_{\sigma(i)=j,\sigma(k)=l} q_{ij}q_{kl} + q_{n^2+1}^2)$ which reduces to $f(q) = \sum_{\sigma\in I} \alpha_\sigma(\sum_{\sigma(i)=j,\sigma(k)=l} q_{ij}q_{kl})$ since we are given that $\sum_{\sigma\in I} \alpha_\sigma = 0$. The reduced expression can be rewritten as $f(q) = \sum_{i,j,k,l} \sum_{\sigma\in I,\sigma(i)=j,\sigma(k)=l} \alpha_\sigma q_{ij}q_{kl}$, which is same as $f(q) = \sum_{i,j,k,l} q_{ij}q_{kl}$ $\sum_{\sigma\in I,\sigma(i)=j,\sigma(k)=l} \alpha_\sigma$. But from $\sum_{\sigma\in I} \alpha_\sigma \tilde{P}_\sigma^{[2]} = 0$ we know that $\sum_{\sigma\in I,\sigma(i)=j,\sigma(k)=l} \alpha_\sigma = 0$ for all i,j,k,l. Hence $f(q) = 0$. $\square$

Lemma 5.2.5. *$\{\tilde{P}_\sigma^{[2]} | \sigma \in I\}$ is linearly independent if and only if $\{y_\sigma^2 - y_\sigma | \sigma \in I\}$ is linearly independent.*

Proof. Consider $f(q)$ where first n^2 components of q is $-x$ and the last component is z. It can be rewritten as $\sum_{\sigma\in I}\alpha_\sigma(z-P_\sigma^T\cdot x)^2=0\ \forall x\in\mathbb{Q}^{n^2}$, $\forall z\in\mathbb{Q}$. Writing in terms of y_σ, the above statement is equivalent to $\sum_{\sigma\in I}\alpha_\sigma y_\sigma^2=0\ \forall x\in\mathbb{Q}^{n^2}\ \forall z\in\mathbb{Q}$. From lemma 5.2.2, this is equivalent to $\sum_{\sigma\in I}\alpha_\sigma(y_\sigma^2-y_\sigma)=0\ \forall x\in\mathbb{Q}^{n^2}$, $\forall z\in\mathbb{Q}$. $\square$

Corollary 5.2.6. *$\{P_\sigma^{[2]}|\sigma\in I\}$ is linearly independent if and only if $\{y_\sigma^2-y_\sigma|\sigma\in I\}$ is linearly independent.*

Consider the polynomial ring $A=\mathbb{Q}[\{x_{ij}|1\le i,j\le n\}\cup\{z\}]$. The subspace of A generated by $\{x_{ij}|1\le i,j\le n\}\cup\{z\}\cup\{x_{ij}x_{kl}|1\le i,j,k,l\le n\}\cup\{zx_{ij}|1\le i,j\le n\}$ is the direct sum of components of degree 1 and 2. Its dimension is $d=1+(n^2+1)+(n^4+n^2)/2+n^2$. For $n>6$, $d\le n!$. So the set $\{y_\sigma^2-y_\sigma|\sigma\in S_n\}$ is linearly dependent for all $n>6$.

Let J be a minimal set of permutations such that $\{y_\sigma^2-y_\sigma|\sigma\in J\}$ is linearly dependent. So there exist α_σ such that $\sum_{\sigma\in J}\alpha_\sigma(y_\sigma^2-y_\sigma)=0$. Since no set of two $P_\sigma^{[2]}$ is linearly dependent, the same holds for any pair of $y_\sigma^2-y_\sigma$. Hence at least three coefficients are non-zero. Assume that $\alpha_{\sigma_1},\alpha_{\sigma_2},\alpha_{\sigma_3}$ are non-zero. Let the sign of the first two be same. We may assume that α_{σ_1} and α_{σ_2} are negative. If not, then invert the sign of every coefficient. Note that $(-\alpha_{\sigma_1})(y_{\sigma_1}^2-y_{\sigma_1})$ is non-negative for all $x\in\mathbb{Z}^{n^2},z\in\mathbb{Z}$. So $\sum_{\sigma\in J}\alpha_\sigma(y_\sigma^2-y_\sigma)+(-\alpha_{\sigma_1})(y_{\sigma_1}^2-y_{\sigma_1})$ is non-negative for all $x\in\mathbb{Z}^{n^2},z\in\mathbb{Z}$. This simplifies to $\sum_{\sigma\in J\setminus\{\sigma_1\}}\alpha_\sigma(y_\sigma^2-y_\sigma)$ which is non-negative for all $x\in\mathbb{Z}^{n^2},z\in\mathbb{Z}$ and $\{y_\sigma^2-y_\sigma|\sigma\in J\setminus\{\sigma_1\}\}$ is linearly independent. But α_{σ_2} is negative. Hence we have established that the second statement of lemma 5.2.1 does not hold.

Theorem 5.2.7. *Region of $\mathcal{P}$ satisfying conditions (5.1), properly contains $\mathcal{B}^{[2]}$.*

Corollary 5.2.8. *There exists at least one facet of $\mathcal{B}^{[2]}$ which is not an instance of (5.1).*

5.3 Towards a general inequality for all facets of $\mathcal{B}^{[2]}$

Lemma 5.3.1. *For any pair of distinct vertices of $\mathcal{B}^{[2]}$, there exists a hyperplane that contains only these vertices and the entire polytope is on one side of this hyperplane.*

Proof. Let $\mathcal{P}^{[2]}_{\sigma_1}, \mathcal{P}^{[2]}_{\sigma_2}$ be a distinct pair of vertices of $\mathcal{B}^{[2]}$. Next consider the hyperplane $\sum_{\sigma_1(i)=j,\sigma_1(k)=l \text{ or } \sigma_2(i)=j,\sigma_2(k)=l} Y(ij,kl) = n^2$. Clearly both $\mathcal{P}^{[2]}_{\sigma_1}, \mathcal{P}^{[2]}_{\sigma_2}$ satisfy this equation. Note that for any point X in $\mathcal{B}^{[2]}$, $\sum_{\sigma_1(i)=j,\sigma_1(k)=l \text{ or } \sigma_2(i)=j,\sigma_2(k)=l} X(ij,kl) \leq n^2$. So the entire polytope $\mathcal{B}^{[2]}$ lies in the corresponding halfspace. All we need to show now is that no other vertex of $\mathcal{B}^{[2]}$ can lie on this hyperplane. Let us assume that such a vertex exists. Let us call it $\mathcal{P}^{[2]}_{\sigma_3}$. So we have at least one of $\sigma_1(i) = j, \sigma_1(k) = l$ and $\sigma_2(i) = j, \sigma_2(k) = l$ true whenever $\sigma_3(i) = j, \sigma_3(k) = l$. However, since σ_1 and σ_2 are distinct, there must exist some i_0, j_0, k_0, l_0 such that $\sigma_3(i_0) = j_0, \sigma_3(k_0) = l_0$ and exactly one of $\sigma_1(i_0) = j_0, \sigma_1(k_0) = l_0$ and $\sigma_2(i_0) = j_0, \sigma_2(k_0) = l_0$ is true. W.l.o.g. let $\mathcal{P}^{[2]}_{\sigma_1}(i_0j_0, k_0l_0) = 0$ and $\mathcal{P}^{[2]}_{\sigma_2}(i_0j_0, k_0l_0) = 1$.

Recall that a $\mathcal{P}^{[2]}_{\sigma}$ matrix is defined as $\mathcal{P}^{[2]}_{\sigma}(ij, kl) = P_\sigma(i, j) \cdot P_\sigma(k, l)$, where P_σ is the corresponding permutation matrix. So if $\mathcal{P}^{[2]}_{\sigma}(ij, kl) = 0$, it implies that at least one of $P_\sigma(i, j) = 0$ and $P_\sigma(k, l) = 0$ is true. If $P_\sigma(i, j) = 0$ it means the (ij)-th row of $\mathcal{P}^{[2]}_{\sigma}$ must be zero. Similarly, a value of $\mathcal{P}^{[2]}_{\sigma}(ij, kl) = 1$ implies that $P_\sigma(i, j) = 1$ and $P_\sigma(k, l) = 1$. Which implies that $\mathcal{P}^{[2]}_{\sigma}(ij, rs) = P_\sigma(r, s)$ for all r, s. Similarly, $\mathcal{P}^{[2]}_{\sigma}(kl, rs) = P_\sigma(r, s)$ for all r, s.

From the above we have the (i_0j_0)-th row of $\mathcal{P}^{[2]}_{\sigma_1}$ zero and $\mathcal{P}^{[2]}_{\sigma_2}(i_0j_0, rs) = P_{\sigma_2}(r, s)$ for all r, s. So we can conclude from the above that $\mathcal{P}^{[2]}_{\sigma_3}(i_0j_0, rs) = P_{\sigma_2}(r, s)$ for all r, s. But this implies that σ_2 and σ_3 must be identical.

Hence, we cannot have a vertex different from $\mathcal{P}^{[2]}_{\sigma_1}, \mathcal{P}^{[2]}_{\sigma_2}$ on the hyperplane defined above. □

Corollary 5.3.2. *For any pair of distinct permutations σ_1, σ_2, the line segment $(P^{[2]}_{\sigma_1}, P^{[2]}_{\sigma_2})$ is an edge (1-D face) of $\mathcal{B}^{[2]}$.*

Notation: For any face, F, of $\mathcal{B}^{[2]}$, Y_F will denote the projection of the origin on the affine plane of F. In particular Y_0 will denote the projection of the origin on plane P, the affine plane of $\mathcal{B}^{[2]}$.

Lemma 5.3.3. *The projection of the origin on any edge (1-D face) $F = \{P^{[2]}_{\sigma_1}, P^{[2]}_{\sigma_2}\}$ is $Y_F = (P^{[2]}_{\sigma_1} + P^{[2]}_{\sigma_2})/2$.*

Proof. $((P^{[2]}_{\sigma_1} + P^{[2]}_{\sigma_2})/2 - 0) \cdot (P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma_2}) = (P^{[2]}_{\sigma_1} \cdot P^{[2]}_{\sigma_1} - P^{[2]}_{\sigma_2} \cdot P^{[2]}_{\sigma_2})/2 = (n^2 - n^2)/2 = 0$. Here the product operator is the Frobenius inner product. □

Lemma 5.3.4. $Y_0 = \sum_\sigma P^{[2]}_\sigma / n!$.

Proof. By the definition Y_0 is the projection of the origin on the affine space of $\mathcal{B}^{[2]}$, namely, P. Clearly $\sum_\sigma P_\sigma^{[2]}/n!$ is a convex combination of $P_\sigma^{[2]}$s hence it is in the plane P. Now we only need to show that for any arbitrary σ_1, σ_2, $P_{\sigma_1}^{[2]} - P_{\sigma_2}^{[2]}$ is normal to Y_0. Since $P_{\sigma'}^{[2]} \cdot \sum_\sigma P_\sigma^{[2]}/n!$ is independent of σ', $(P_{\sigma_1}^{[2]} - P_{\sigma_2}^{[2]}) \cdot \sum_\sigma P_\sigma^{[2]}/n! = 0$. □

Observation 5.3.5. *Let F_1 and F_2 be faces of $\mathcal{B}^{[2]}$ where F_2 is contained in F_1. Then Y_{F_2} is the projection of Y_{F_1} on the affine plane of F_2.*

Proof. If F_2 is zero dimensional, then the claim is trivially true. So assume that F_2 is at least one dimensional. Since F_2 is a face of F_1, F_1 is also at least one dimensional. Let z be any arbitrary point in F_2. So $(Y_{F_2} - z).(Y_{F_2} - 0) = 0$. Besides, Y_{F_2} and z belong to F_1 so $(Y_{F_2} - z).(Y_{F_1} - 0) = 0$. Hence $(Y_{F_2} - z).(Y_{F_2} - Y_{F_1}) = 0$. □

Corollary 5.3.6. *For any face F of $\mathcal{B}^{[2]}$, Y_F is the projection of Y_0 on the affine plane of F.*

Lemma 5.3.7. *For any facet F of $\mathcal{B}^{[2]}$, there exists a half space H given by $\sum_p \sum_{ijkl} n_{ij}^p n_{kl}^p Y_{ij,kl} \leq c$ where n_{ij} and c are non-negative reals, such that the half space $H \cap P$ in P defines F, i.e., the bounding plane of $H \cap P$ is the plane of F.*

Proof. $Y_F - Y_0$ is the normal to F in the affine plane P. Hence the inequality for the half space H_F, in P, associated with F is $(Y - Y_F) \cdot (Y_F - Y_0) \leq 0$. Let Y'_F be a point in $\mathcal{B}^{[2]}$ such that $(Y_F - Y_0) = c(Y'_F - Y_0)$ for some constant c. So we have $H_F : (Y - Y_F) \cdot c(Y'_F - Y_0) \leq 0$. Note that inside the plane P, $(Y - Y_F) \cdot Y_0 = 0$. Consider the halfspace $H : c(Y - Y_F) \cdot Y'_F \leq 0$ or $Y \cdot Y'_F \leq Y_F \cdot Y'_F$ where Y is no longer constrained to lie in plane P. Clearly, $H_F = H \cap P$. We will show in chapter 6 that every point in $\mathcal{B}^{[2]}$ is a completely positive matrix. So $(Y'_F)_{ij,kl} = \sum_p n_{ij}^p n_{kl}^p$ where each n_{rs}^p is a non-negative real. So we can rewrite the equation of H as $\sum_p \sum_{ijkl} n_{ij}^p n_{kl}^p Y_{ij,kl} \leq c_F$, where $c_F = Y_F \cdot Y'_F$ is a constant dependent on F. □

Let us denote the inequality $\sum_p \sum_{ijkl} n_{ij}^p n_{kl}^p Y_{ij,kl} \leq c_F$ by $f(Y) \leq c_F$. To ensure that the corresponding equation is a supporting plane for $\mathcal{B}^{[2]}$, at least one $P_\sigma^{[2]}$ must satisfy it with equality. $f(P_\sigma^{[2]}) \leq c_F$ simplifies to $\sum_p \sum_{i,k} n_{i\sigma(i)}^p n_{k,\sigma(k)}^p \leq c_F$. So we have $c_F = \max_\sigma \sum_p \sum_{i,k} n_{i\sigma(i)}^p n_{k,\sigma(k)}^p$. In section 6.4 we will revisit this topic.

5.4 Conclusion

In the first part of this chapter we showed that the facets described in chapter 3 are only a subset of the facets of $\mathcal{B}^{[2]}$. So finding the remaining facets remains an open problem. In the second part we gave a general inequality that defines every supporting plane for $\mathcal{B}^{[2]}$.

Chapter 6

A Semidefinite Formulation

6.1 Introduction

An $m \times m$ symmetric real matrix M is said to be *positive semidefinite* if it can be expressed as QQ^T for some $m \times k$ real matrix Q. If the row vectors of Q are $v_1^T, \ldots, v_m^T$, then we will call this set a vector-realization of M in k-dimensional space. It is easy to see that there is always a vector realization in $k = rank(M)$ dimensional space.

If matrix M has a vector realization in which each $v_i \in \mathbb{R}^k_{\geq 0}$, then it is called a *completely positive*(CP) matrix. *cp-rank* of a completely positive matrix M is the smallest integer k such that M has a non-negative vector realization in k dimensional space. Following result gives a useful bound for cp-rank(M).

Theorem 6.1.1. *(HL83) For any completely positive matrix M of rank r, cp-rank $(M) \leq r(r+1)/2$.*

A CP formulation is a program with linear constraints and a linear objective function, where the variable matrix is confined to the cone of CP matrices. In this chapter we give a CP formulation of GI and study its feasible region. We also study the Semidefinite (SDP) relaxation of the formulation and prove certain results about its feasible region.

6.2 CP Formulation of GI

Let us recall the definition of the product of two graphs dicussed in the definition of integer linear program for GI. Let $G_1 = ([n], E_1)$ and $G_2 = ([n], E_2)$ be simple graphs on n vertices each. Define a graph $G = (V, E)$, where $V = [n] \times [n]$ and $\{ij, kl\} \in E$ if either $\{i, k\} \in E_1$ and $\{j, l\} \in E_2$ or $\{i, k\} \notin E_1$ and $\{j, l\} \notin E_2$, provided $i \neq k$ and $j \neq l$. It follows from a result in (Koz78) that G_1 and G_2 are isomorphic if and only if G contains a clique of size n. G is an n-partite graph so the largest clique in G cannot be of size more than n. Therefore G_1 and G_2 are isomorphic if and only if G has clique number n. Moreover there is a 1-1 correspondence between n-cliques of G and the isomorphisms between G_1 and G_2.

It is shown in (dKP02) that by replacing the positive semidefinite condition in a SDP formulation of the Lovász Theta number of a graph by *completely positive* condition, the optimal value of the resulting program is the stability number (independence number) of that graph. So the following completely-positive program (CP-LT)

$$
\begin{aligned}
\text{CP-LT:}\quad \text{maximize}\quad & \sum_{i,j\in[n]} \tilde{Y}_{ij,ij} && \\
\text{subject to}\quad & \tilde{Y} \in \mathcal{C}^* && (6.1a)\\
& \tilde{Y}_{ij,ik} = 0 \quad , 1 \le i, j, k \le n,\ j \neq k && (6.1b)\\
& \tilde{Y}_{ji,ki} = 0 \quad , 1 \le i, j, k \le n,\ j \neq k && (6.1c)\\
& \tilde{Y}_{\omega,\omega} = 1 && (6.1d)\\
& \tilde{Y}_{ij,\omega} = \tilde{Y}_{ij,ij} \quad ,1 \le i, j \le n && (6.1e)\\
& \tilde{Y}_{ij,kl} = 0 \quad , \{ij, kl\} \notin E && (6.1f)
\end{aligned}
$$

returns value n if and only if the stability number of $\overline{G}$ or equivalently, the clique number of G is n. Here $\tilde{Y}$ is a $(n^2 + 1) \times (n^2 + 1)$ matrix of variables with index set $(([n] \times [n]) \cup \{\omega\}) \times (([n] \times [n]) \cup \{\omega\})$ and $\mathcal{C}^*$ denotes the cone of completely positive matrices. The optimum value of CP-LT will be denoted by $cp\vartheta$.

Combining the two observations we deduce that this program returns n if and only if G_1 and G_2 are isomorphic. And it returns $n - 1$ or less if the graphs are non-isomorphic.

Let X be an $N \times N$ completely positive matrix or, more generally, a positive semidefinite matrix. Then there exists a set of vectors $\{u_i | 1 \le i \le N\}$ such that

$X_{ij} = u_i \cdot u_j$ for all i, j. We will refer to $\{u_i | 1 \leq i \leq N\}$ as a *vector realization* of X.

Given a vector realization of a solution of CP-LT, $\{u_{ij} | 1 \leq i, j \leq n\}$, a subset $\{u_{ij_i} | i \in I\}$ of non-zero vectors will be called a *consistent set* if $u_{pj_p} \cdot u_{qj_q} > 0$ for all $p, q \in I$. If $I = \{1, \ldots, n\}$, then it will be called a *complete consistent set*. Let $\{u_{ij_i} | 1 \leq i \leq n\}$ be a complete consistent set. Define a function f as $f(i) = j_i$ for $1 \leq i \leq n$. From 6.1c we see that f is a permutation. From the graph conditions 6.1f we see that f is an isomorphism between G_1 and G_2.

Observation 6.2.1. *If $\{u_{ij} | 1 \leq i, j \leq n\}$ is the vector realization of a solution Y of CP-LT which contains a complete consistent set $\{u_{1\sigma(1)}, \ldots, u_{n\sigma(n)}\}$, then σ is an isomorphism between G_1 and G_2. This is true also when Y is a positive semidefinite matrix.*

6.2.1 United Vectors

We take a short diversion.

Let w be any fixed unit vector. Then for every unit vector v, we call $u = (w+v)/2$ a *united vector with respect to w*.

Observation 6.2.2. *With respect to a fixed unit vector w,*
(i) a vector u is united if and only if $u \cdot w = u^2$.
(ii) if u_1 and u_2 are mutually orthogonal united vectors, then $u_1 + u_2$ is also a united vector.
(iii) let $u_1, \ldots, u_k$ be a set of pairwise orthogonal united vectors. This set is maximal (i.e., no new united vector can be added to it while preserving pairwise orthogonality) if and only if w belongs to the subspace spanned by these vectors.
(iv) let $u_1, \ldots, u_k$ be a set of pairwise orthogonal united vectors. w belongs to the subspace spanned by these vectors if and only if $\sum_i u_i = w$.
(v) let $u_1, \ldots, u_k$ be a set of pairwise orthogonal united vectors. $\sum_i u_i = w$ if and only if $\sum_i u_i^2 = 1$.

Let $\tilde{Y}$ be a solution of CP-LT. Since it is a completely positive matrix, there exist vectors u_{ij} for $1 \leq i, j \leq n$ and a unit vector w such that $\tilde{Y}_{ij,kl} = u_{ij} \cdot u_{kl}$ and $\tilde{Y}_{ij,w} = \tilde{Y}_{w,ij} = u_{ij} \cdot w$. Note that the same would be true if $\tilde{Y}$ was any positive semidefinite matrix. From conditions 6.1d and 6.1e we see that u_{ij} are united vectors with respect to w. From conditions 6.1b and 6.1c we see that $\{u_{i1}, \ldots, u_{in}\}$ is a set of mutually orthogonal united vectors, for each i. Same is true for $\{u_{1i}, \ldots, u_{ni}\}$. The objective function $\sum_{ij} \tilde{Y}_{ij,ij}$ can achieve its maximum value n if and only if each of the sets, $\{u_{i1}, \ldots, u_{in}\}$, (equivalently, $\{u_{1i}, \ldots, u_{ni}\}$) is a maximal pairwise orthogonal set of united vectors.

Observe that $\tilde{Y}$ is a $(n^2+1) \times (n^2+1)$ matrix having its last row and the last column equal to its diagonal, see condition 6.1e. Hence the $n^2 \times n^2$ principal submatrix contains complete information of $\tilde{Y}$. We will refer to it as Y. So $Y_{ij,kl} = u_{ij} \cdot u_{kl}$ for all i, j, k, l. Clearly $\tilde{Y}$ can be obtained from Y by setting $\tilde{Y}_{ij,n^2+1} = \tilde{Y}_{n^2+1,ij} = Y_{ij,ij}$ and $\tilde{Y}_{n^2+1,n^2+1} = 1$. The remaining entries of $\tilde{Y}$ are same as those of Y.

6.2.2 Difference in cpϑ values for Isomorphic and Non-Isomorphic Graphs

From the result of (dKP02) discussed at the start of the chapter we know that the value of *cp*ϑ is n if the graphs are isomorphic. Otherwise it is $n-1$ or less. Hence it has a gap of at least 1 between isomorphic and non-isomorphic cases. Here we present a geometric argument to establish a much smaller gap between the two cases. Although this is not significant in the light of the above, nonetheless the approach is different and basically shows that if we are too close to the optimum value of *cp*ϑ then the graphs must be isomorphic.

Theorem 6.2.3. *Let Y be a solution of CP-LT. If $\sum_j Y_{ij,ij} \geq 1 - 1/(4n^4)$ for each i, then G_1 and G_2 are isomorphic.*

Proof. Let N denote the cp-rank of Y. So we have a non-negative vector realization of Y, $\{u_{ij} | i, j \in [n]\} \cup \{w\}$ in an N-dimensional space. So we have $u_{ij} \cdot u_{kl} = Y_{ij,kl}$ for all i, j, k, l and there is an orthonormal basis of this space, $B = \{e_p | p \in [N]\}$, such that every u_{ij} belongs to the closed positive orthant of this basis.

From Theorem 6.1.1 $N < n^4$ because the rank of Y is at most n^2. Hence from the statement of this theorem $w \cdot \sum_j u_{ij} = \sum_j u_{ij}^2 = \sum_j Y_{ij,ij} \geq 1 - 1/(4N)$ for each i. Let $S_i = \{u_{i1}, \ldots, u_{in}\}$ for each i. Since it is a set of orthogonal united vectors with respect to w, $w. \sum_j u_{ij} = \sum_j u_{ij}^2 \leq 1$.

Without loss of generality assume that $w \cdot e_1 \geq w \cdot e_j$ for all j. So $w \cdot e_1 \geq 1/\sqrt{N}$ because w is a unit vector. If every vector in S_i is orthogonal to e_1, then $w \cdot \sum_j u_{ij}$

can be at most $|w - (w \cdot e_1)e_1|$, which is at most $(1 - (w \cdot e_1)^2)^{1/2} \le (1 - 1/N)^{1/2} < 1 - 1/(2N)$, contradicting the earlier inequality for $w \cdot \sum_j u_{ij}$. So there exists a vector $u_{ij_i} \in S_i$ such that $u_{ij_i} \cdot e_1 > 0$. Let there be a $k \ne j_i$ such that $u_{ik} \cdot e_1 > 0$. As all vectors are in the closed positive orthant, $u_{ij_i} \cdot u_{ik} \ge (u_{ij_i} \cdot e_1)(u_{ik} \cdot e_1) > 0$. This contradicts the fact that the vectors of S_i are pairwise orthogonal. Hence we conclude that for each i there exists a unique vector $u_{ij_i} \in S_i$ such that $u_{ij_i} \cdot e_1 > 0$. Thus $\sum_{j=1}^{n} u_{ij} \cdot e_1 = u_{ij_i} \cdot e_1$.

Next we will show that $u_{ij_i} \cdot u_{kj_k} \ge 1/(16N^2)$ for all $i, k \in [n]$. For any i, from the given facts $1 - 1/(4N) \le w \cdot \sum_j u_{ij} = (w \cdot e_1)(\sum_j u_{ij} \cdot e_1) + (w - (w \cdot e_1)e_1) \cdot (\sum_j u_{ij} - (\sum_j u_{ij} \cdot e_1)e_1)$. Since $(\sum_j u_{ij})^2 \le 1$, $(w - (w \cdot e_1)e_1) \cdot (\sum_j u_{ij} - (\sum_j u_{ij} \cdot e_1)e_1) \le |w - (w \cdot e_1)e_1| \le 1 - 1/(2N)$. The last inequality has been established in the previous paragraph. So $(w \cdot e_1)(\sum_j u_{ij} \cdot e_1) \ge 1/(4N)$ for all i. Hence $u_{ij_i} \cdot e_1 = \sum_j u_{ij} \cdot e_1 \ge 1/(4N)$.

All vectors of each S_i are in the closed positive orthant hence $u_{ij_i} \cdot u_{kj_k} \ge (u_{ij_i} \cdot e_1)(u_{kj_k} \cdot e_1) \ge 1/(16N^2)$ for all $i, k \in [n]$. Thus the set $\{u_{1j_1}, \ldots, u_{nj_n}\}$ is pairwise non-orthogonal, and hence a complete consistent set. From Observation 6.2.1 we know that the permutation, $\sigma(i) = j_i$ for all i, is an isomorphism between G_1 and G_2. □

Corollary 6.2.4. *If G_1 and G_2 are non-isomorphic, then the value of $cp\vartheta$ must be less than $n - 1/(4n^4)$. Hence the gap between isomorphic and non-isomorphic cases is at least $1/4n^4$.*

6.2.3 The Second-order Birkhoff Polytope

In (PR09) a completely positive formulation of *Quadratic Assignment Problem* (QAP) is given. It is established there that the optimum feasible region (where $cp\vartheta = n$) of CP-LT without (6.1f) is $\mathcal{B}^{[2]}$, see theorem 3 in (PR09). Here we will refine that result by showing that the optimum feasible region of CP-LT is $\mathcal{B}^{[2]}_{G_1G_2}$.

Lemma 6.2.5. *The optimal feasible region of CP-LT (i.e., where $\sum_{i,j\in[n]} Y_{ij,ij} = n$) is $\mathcal{B}^{[2]}_{G_1G_2}$.*

Proof. $P_\sigma^{[2]}$ is a completely positive rank-1 matrix because it is the outer product of the vectorized P_σ with itself. If σ is an isomorphism between G_1 and G_2 then $P_\sigma^{[2]}$ is feasible for CP-LT since it satisfies all the linear conditions. Conversely, if $P_\sigma^{[2]}$, for some σ, is feasible for CP-LT, then it implies that the product graph G has a clique of size n (i.e., the program returns the optimal value n) and hence G_1 and G_2 are isomorphic with σ as the isomorphism.

From the above, $\mathcal{B}^{[2]}_{G_1G_2}$ is contained in the feasible region of CP-LT with objective function equal to n. Consider a non-negative vector realization $\{u_{ij}|i,j \in [n]\} \cup \{w\}$ for a CP-LT solution Y with the objective function $cp\vartheta = n$. Let W denote an $n \times n$ matrix with (i,j)-th entry being u_{ij}. Conditions 6.1b and 6.1c ensure that vectors in any row or any column of W are pairwise orthogonal. Since objective function attains value n, from Observation 6.2.2 vectors of each row/column form a maximal set of pairwise orthogonal united vectors. Also from the same observation each row and each column adds up to w. Assume that the vector realization is in an N-dimensional space. Consider the r-th component of the matrix, i.e., the matrix formed by the r-th component of each vector. Let us denote it by D_r. Each element of D_r is non-negative and each row and each column adds up to w_r, the r-th component of w. Hence D_r is w_r times a doubly-stochastic matrix. But the vectors of the same row (resp. column) are pairwise orthogonal so exactly one entry is non-zero in each row (resp. column) of D_r if $w_r > 0$. So $D_r = w_r P_{\sigma_r}$ for some permutation σ_r. We can express W by $\sum_r w_r P_{\sigma_r} e_r$ where e_r denotes the unit vector along the r-th axis. $Y_{ij,kl}$ is the inner product of the vectors u_{ij} and u_{kl} which is $(\sum_r w_r (P_{\sigma_r})_{ij} e_r) \cdot (\sum_s w_s (P_{\sigma_s})_{kl} e_s) = \sum_r w_r^2 (P_{\sigma_r})_{ij} . (P_{\sigma_r})_{kl} = \sum_r w_r^2 (P^{[2]}_{\sigma_r})_{ij,kl}$. Thus $Y = \sum_r w_r^2 P^{[2]}_{\sigma_r}$. Since $\sum_r w_r^2 = w^2 = 1$, Y is a convex combination of some of the $P^{[2]}_{\sigma}$'s. Each σ_r, where $w_r > 0$, is an isomorphism between G_1 and G_2. Hence $Y \in \mathcal{B}^{[2]}_{G_1G_2}$. So the feasible region is contained in $\mathcal{B}^{[2]}_{G_1G_2}$. □

In these sections we have seen that the Graph Isomorphism problem can be solved via a Completely Positive formulation. However, it is NP-Hard to optimize over the cone of completely positive matrices. But the same is not true of a Semidefinite formulation. So a natural question arises. Can a semidefinite relaxation of CP-LT be used to solve GI? In the next section we attempt to answer this question.

6.3 SDP Relaxation - Lovász Theta Function

Consider the semidefinite relaxation SDP-LT, given below. The optimum value of SDP-LT is $\vartheta(\overline{G})$, where G is the symmetric tensor product of G_1 and G_2 defined earlier.

Let $\{u_{ij}|1 \leq i,j \leq n\}$ be a vector realization of a solution Y of this SDP. Equations 6.1d and 6.1e imply that u_{ij} are united vectors, as in CP-LT. Equations 6.1b to 6.1e imply that $\{u_{i1},\ldots,u_{in}\}$ are orthogonal sets and so are $\{u_{1i},\ldots,u_{ni}\}$. From Observation 6.2.2 we know that each of these sets add up to a vector of length at most 1. The objective function is $\sum_{ij} Y_{ij,ij} = \sum_{ij} u_{ij}^2 = \sum_i w \cdot \sum_j u_{ij} \leq \sum_i 1 = n$. The objective function reaches its maximum value n if and only if $\sum_i u_{ij} = w = \sum_j u_{ij}\ \forall i,j$.

$$\begin{aligned} \text{SDP-LT:} \quad \text{maximize} \quad & \sum_{i,j\in[n]} Y_{ij,ij} \\ \text{subject to} \quad & Y \succeq 0 && (6.2a) \\ & 6.1b\text{-}6.1f \text{ from CP-LT} \\ & Y_{ij,kl} \geq 0 && (6.2b) \end{aligned}$$

The program can be stated as a feasibility program by adding the *optimality condition* $\sum_{ij} Y_{ij,ij} = n$ to its set of conditions. The feasible region of the feasibility version of SDP-LT (let's say SDP-LT-OPT) will be denoted by $\mathcal{F}_{G_1G_2}$. It will be denoted by $\mathcal{F}$ if $G_1 = G_2 = (V,\emptyset)$ or $G_1 = G_2 = K_n$.

Lemma 6.3.1. *$P_\sigma^{[2]}$ is in the feasible region of the above SDP if and only if σ is an isomorphism between G_1 and G_2.*

Proof. Let Y be a solution of SDP-LT.

$Y = P_\sigma^{[2]}$ iff ($Y_{ij,kl} = 1$ if and only if $j = \sigma(i)$ and $l = \sigma(k)$ else $Y_{ij,kl} = 0$). Hence from condition (6.1f) $Y = P_\sigma^{[2]}$ implies $((i,k) \in E_1$ iff $(\sigma(i),\sigma(k)) \in E_2)$, i.e., σ is an isomorphism between G_1 and G_2.

Conversely, let σ be an isomorphism, then $(i,k) \in E_1$ iff $(\sigma(i),\sigma(k)) \in E_2$. It can be verified that $P_\sigma^{[2]}$ satisfies (6.1f). All other conditions of SDP-LT are satisfied by $P_{\sigma'}^{[2]}$ for every σ'. Hence $P_\sigma^{[2]}$ is a solution of SDP-LT. □

Lemma 6.3.2. *$P_\sigma^{[2]}$ are the only rank-1 points in $\mathcal{F}$. Also, these constitute some of the extreme points of $\mathcal{F}$.*

Proof. If Y is a rank-1 point in $\mathcal{F}$, then there exists a vector $v = \{v_{ij}|1 \leq i,j \leq n\} \in \mathbb{R}^{n^2}$ such that $Y_{ij,kl} = v_{ij}.v_{kl}$ (scalar product). Since $Y_{ij,il} = 0$ for $j \neq l$, for any given i, v_{ij} must be zero for at least $n-1$ values of j. Similarly for a given j, v_{ij} is zero for at least $n-1$ values of i.

If $v_{ij} = 0$ for all j, then for any arbitrary k, l, $Y_{kl,kl} = \sum_j Y_{ij,kl} = 0$. Hence $\sum_{kl} Y_{kl,kl} = 0$. This is absurd because $\sum_{kl} Y_{kl,kl}$ must be n as Y is a point in $\mathcal{F}$. So we see that for each i there exists a unique j_i such that $v_{ij_i} \neq 0$ and $v_{ij} = 0$ for all $j \neq j_i$. Y belongs to $\mathcal{F}$ so $\sum_j Y_{ij,ij} = 1$ for each i. So $1 = \sum_j Y_{ij,ij} = \sum_j v_{ij}^2 = v_{ij_i}^2$. Hence v_{ij_i} is either 1 or -1. But $v_{ij_i}.v_{kj_k} \geq 0$ for all i, k. So either all v_{ij_i} are 1 or all are -1. Let V denote the $n \times n$ matrix with $V_{ij} = v_{ij}$ for all i, j. We see that each row of V has one 1 (or -1) and the rest of the entries are 0. Similarly we can show that each column has one 1 (respectively, -1). So V is a permutation matrix, say P_σ or its negation, and $Y = P_\sigma^{[2]}$. Since rank-1 points lie on extreme rays of the PSD cone they form some of the extreme points of $\mathcal{F}$. $\square$

It follows from lemma 6.3.2 that following SDP would solve GI,

$$\begin{array}{lll} \text{SDP-GI:} & \text{minimize} & \textbf{Rank}\, Y \\ & \text{subject to} & Y \in \mathcal{F}_{G_1G_2} \end{array}$$

by observing that the optimum solution of SDP-GI is $P_\sigma^{[2]}$ if and only if the graphs have an isomorphism σ. However, this test too is NP-Hard.

Lemma 6.3.3. *Consider a point Y in $\mathcal{F}$ of rank $r \leq n$. If it has a vector realization $\{u_{ij} | 1 \leq i, j \leq n\}$ such that there exist $i, j_1, \ldots, j_r$ with $u_{ij_1}^2 > 0$, $u_{ij_2}^2 > 0$, $\ldots$, $u_{ij_r}^2 > 0$, then it belongs to the CP-feasible region, $\mathcal{B}_{G_1G_2}^{[2]}$.*

Proof. Vectors u_{ij_1}, u_{ij_2}, $\ldots$, u_{ij_r} are mutually orthogonal vectors so they can be taken as a basis of the r-dimensional space in which all the vectors lie. Now since the remaining vectors make non-negative dot products with these r vectors, all the vectors lie in the positive orthant of this basis. Thus the given matrix Y is completely positive and hence belongs to the CP-feasible region. $\square$

Lemma 6.3.4. *All rank-2 points of $\mathcal{F}$ belong to the CP-feasible region, $\mathcal{B}_{G_1G_2}^{[2]}$.*

Proof. Let Y be a rank-2 point in $\mathcal{F}$ with vector realization $\{u_{ij} | 1 \leq i, j \leq n\}$. The vectors must be in 2-dimensional space so for each i there are at most two values of j such that u_{ij} is non-zero. If for some i there is only one such index, $j = j_1$, such that u_{ij} is non-zero, then $\sum_{j=1}^n u_{ij} = w$ (this must be true since $\sum_{ij} Y_{ij,ij} = n$)

implies that $u_{ij_1} = w$. If u_{ij} is non-zero for only one value of j for all i, then Y will be a rank-1 matrix, i.e., $P_\sigma^{[2]}$ for some σ. But Y is a rank-2 matrix so there exists an i such that u_{ij_1} and u_{ij_2} are non-zero for some $j_1 \neq j_2$. From lemma 6.3.3 Y is a completely positive matrix. □

Lemma 6.3.5. *Let $\{u_{j1}, u_{j2}\}$ be a pair of orthogonal united vectors with $u_{j1} + u_{j2} = w$, for $j = 1, 2, \ldots, r$. Then there exists a consistent set of vectors $\{v_j \in \{u_{j1}, u_{j2}\} | j = 1, \ldots, r\}$.*

Proof. Suppose $\{u_{j1}, u_{j2}\} = \{u_{k1}, u_{k2}\}$ for some j, k. In that case we take $v_j = v_k$.

Since $u_{j1} \cdot u_{j2} = 0$ and $u_{j1} + u_{j2} = w$, $u_{j1}^2 + u_{j2}^2 = 1$ for each j. Without loss of generality assume that for each j, $u_{j1}^2 \geq u_{j2}^2$. Let $v_j = u_{j1}$. So $v_j^2 \geq 1/2\ \forall j$. For arbitrary p, q, we will show that $u_{p1} \cdot u_{q1} > 0$ where $(u_{p1}, u_{p2}) \neq (u_{q1}, u_{q2})$. Contrary to the claim, assume that $u_{p1} \cdot u_{q1} = 0$ where $\{u_{p1}, u_{p2}\} \neq \{u_{q1}, u_{q2}\}$.

$u_{p1}^2 = u_{p1} \cdot w = u_{p1} \cdot u_{q1} + u_{p1} \cdot u_{q2} = u_{p1} \cdot u_{q2}$. Similarly $u_{q1}^2 = u_{p2} \cdot u_{q1}$. So using these equations we have $1 = w^2 = (u_{p1} + u_{p2}) \cdot (u_{q1} + u_{q2}) = u_{p1}^2 + u_{q1}^2 + u_{p2} \cdot u_{q2}$.

First consider the case that $u_{p1}^2 > 1/2$ or $u_{q1}^2 > 1/2$. In this case $1 = u_{p1}^2 + u_{q1}^2 + u_{p2} \cdot u_{q2} > 1$, which is absurd.

Now we consider the remaining case that $u_{p1}^2 = u_{q1}^2 = 1/2$. From the above, $u_{p1} \cdot u_{q2} = 1/2$ and $u_{p2} \cdot u_{q1} = 1/2$. From the last para $1 = 1 + u_{p2} \cdot u_{q2}$. So $u_{p2} \cdot u_{q2} = 0$. So $(u_{p1} - u_{q2})^2 = 1/2 + 1/2 - 2u_{p1} \cdot u_{q2} = 0$, giving $u_{p1} = u_{q2}$. Similarly we can show that $u_{p2} = u_{q1}$. This gives $\{u_{p1}, u_{p2}\} = \{u_{q1}, u_{q2}\}$, which is absurd. □

Lemma 6.3.6. *The vector realization $\{u_{ij} | 1 \leq i, j \leq n\}$ of any rank-3 point in $\mathcal{F}$ contains at least one complete consistent set.*

Proof. If there exist i, j_1, j_2, j_3 such that $u_{ij_1}^2 > 0$, $u_{ij_2}^2 > 0$, $u_{ij_3}^2 > 0$, then the claim holds from lemma 6.3.3.

Next suppose the only non-zero vectors are $\{u_{ij_i}, u_{ik_i} | 1 \leq i \leq r\} \cup \{u_{ij_i} | i > r\}$. So $u_{ij_i} = w$ for $i > r$. From lemma 6.3.5 there exist $v_i \in \{u_{ij_i}, u_{ik_i}\}$ for $1 \leq i \leq r$ such $v_{i'} \cdot v_{i''} > 0$ for all $1 \leq i', i'' \leq r$. The desired complete consistent set is $\{x_i | 1 \leq i \leq n\}$ where $x_i = v_i$ for $i \leq r$ and $x_i = w$ for $i > r$. Observe that $v_i \cdot w = v_i^2 > 0$. □

Corollary 6.3.7. *If SDP-LT with additional $\sum_{ij} Y_{ij.ij} = n$ condition has a solution of rank at most 3, then G_1 and G_2 are isomorphic.*

Lemma 6.3.8. *For any point $Y \in \mathcal{F}$, rank$(Y) \leq n^2 - 2n + 2$.*

Proof. Consider the vector realization $\{u_{ij}|1 \le i,j \le n\}$ of Y. The rank of Y is equal to the largest number of linearly independent vectors among the u_{ij}s. We know that $\sum_i u_{ij} = w$ for all j and $\sum_j u_{ij} = w$ for all i. These equations allow us to express u_{ni} and u_{in} for all $1 \le i \le n$ in terms of $\{u_{ij}|1 \le i,j \le n-1\} \cup \{w\}$. Therefore the number of linearly independent vectors are at most $(n-1)^2 + 1 = n^2 - 2n + 2$. So the rank of Y is at most $n^2 - 2n + 2$. $\square$

Lemma 6.3.9. *There exists a point within $\mathcal{B}^{[2]}$ having rank $n^2 - 2n + 2$.*

Proof. Let $A = \frac{1}{n!}\sum_\sigma P_\sigma^{[2]}$ which is in $\mathcal{B}^{[2]}$. The diagonal entries of A are $1/n$ and the non-zero off-diagonal entries $1/(n(n-1))$. Observe that vector $\mathbf{1}$ (all 1s) is an eigenvector of A with eigenvalue 1.

Now we will show that A also has eigenvalue $1/(n-1)$ and its multiplicity is $(n-1)^2$. Let $B = A - \frac{1}{n-1}I$. Let R_{ij} denote the (ij)-th row of B. Then $R_{ij} = R_{i1} + R_{1j} - R_{11}\ \forall i,j$. So $R_{22}, \dots, R_{2n}, R_{32}, \dots, R_{3n}, \dots, R_{n2}, \dots, R_{nn}$ can be expressed as linear combinations of the remaining rows. So $rank(B) \le n^2 - (n-1)^2$. Then the null space of B has dimension at least $(n-1)^2$. Since $B = A - \frac{1}{n-1}I$, A has eigenvalue $1/(n-1)$ with multiplicity at least $(n-1)^2$.

These two facts imply that $rank(A) \ge 1 + (n-1)^2 = n^2 - 2n + 2$. $\square$

Corollary 6.3.10. *The ranks of points in $\mathcal{B}^{[2]}$ range from 1 to $n^2 - 2n + 2$.*

6.3.1 Null space lemma

While working on this problem, we got the following interesting result, although it finds no use in this thesis.

Let matrices A, B belong to the cone of positive semidefinite matrices. The following lemma gives a necessary and sufficient condition for $A - \epsilon B$, for some $\epsilon > 0$, to continue to belong to the same cone.

Lemma 6.3.11. *Given positive semidefinite matrices A, B, there exists some $\epsilon > 0$ such that $A - \epsilon B \succeq 0$ if and only if $N(A) \subseteq N(B)$.*

Proof. (Only if) We are given that $A - \epsilon B$ is positive semidefinite for some $\epsilon > 0$. Assume that $N(A) \setminus N(B)$ is non-empty. Let $x \in N(A) \setminus N(B)$. So $x^T(A - \epsilon B)x = -\epsilon x^T B x < 0$.

(If) Let $x_1, x_2, \ldots, x_k$ be eigenvectors of B which are mutually orthogonal and span all the eigenspaces of B. Let their respective eigenvalues be $\lambda_1, \ldots, \lambda_k$. Let $span(x_1, \ldots, x_k)$ denote the space spanned by the eigenvectors, i.e., the subspace orthogonal to the nullspace of B. Similarly let $y_1, \ldots, y_m$ be an orthogal set of eigenvectors of A spanning its eigenspaces. Let their respective eigenvalues be $\alpha_1, \ldots, \alpha_m$. Hence $N(A) \subseteq N(B)$ implies that $span(x_1, \ldots, x_k) \subseteq span(y_1, \ldots, y_m)$. Without loss of generality we assume that $\lambda_1 \geq \lambda_i$ for all i and $\alpha_1 \leq \alpha_j$ for all j.

Let u be a vector orthogonal to the $N(A)$, i.e., in $span(y_1, \ldots, y_m)$. Then $u = \sum_i c_i y_i$ and $u^T A u = \sum_i c_i y_i^T A \sum_l c_l y_l = \sum_i \alpha_i c_i^2 y_i^2 \geq \alpha_1 u^2$. Similarly, let v be a vector orthogonal to the $N(B)$, i.e., in $span(x_1, \ldots, x_k)$. Then $v = \sum_j d_j x_j$ and $v^T B v = \sum_j d_j x_j^T B \sum_r d_r x_r = \sum_j \lambda_j d_j^2 x_j^2 \leq \lambda_1 v^2$.

Let z be any arbitrary vector. Let the projection of z on the null space of A be z_1 and that on the null space of B be z_2. The fact $N(A) \subseteq N(B)$ implies that $z_1^2 \leq z_2^2$ or $(z - z_1)^2 \geq (z - z_2)^2$. So $z^T(A - \epsilon B)z = (z - z_1)^T A (z - z_1) - \epsilon (z - z_2)^T B (z - z_2)$. Observe that $z - z_1 \in span(y_1, \ldots, y_m)$ and $z - z_2 \in span(x_1, \ldots, x_k)$.

Since $u^T A u \geq \alpha_1 u^2$ for $u \in span(y_1, \ldots, y_m)$ we have $(z - z_1)^T A (z - z_1) \geq \alpha_1 (z - z_1)^2$ and since $v^T B v \leq \lambda_1 v^2$ for $v \in span(x_1, \ldots, x_k)$ we have $(z - z_2)^T B (z - z_2) \leq \lambda_1 (z - z_2)^2$. Then $z^T (A - \epsilon B) z \geq \alpha_1 (z - z_1)^2 - \epsilon \lambda_1 (z - z_2)^2 \geq (z - z_2)^2 (\alpha_1 - \epsilon \lambda_1)$ where the last inequality is due to the fact that $(z - z_1)^2 \geq (z - z_2)^2$. By taking $\epsilon_0 = \alpha_1 / \lambda_1$ we see that $z^T (A - \epsilon_0 B) z \geq 0$. □

6.4 A unified equation for the known Facets of $\mathcal{B}^{[2]}$

Let $\{w\} \cup \{u_{ij} | 1 \leq i, j \leq n\}$ represent a (united) vector realization of any point $Y \in \mathcal{B}^{[2]}$. Define a vector $A = \sum_{ij} n_{ij} u_{ij}$ for some choice of $n_{ij} \in \mathbb{Z}$ and let $\beta \in \mathbb{Z}$. Consider the following inequality.

$$(A - (\beta - 0.5) w)^2 \geq 0.25. \tag{6.4}$$

The above inequality defines the half space

$$\sum_{i,j,k,l} n_{ij} n_{kl} Y_{ij,kl} + \beta^2 - \beta \geq (2\beta - 1) \sum_{ij} n_{ij} Y_{ij,ij}$$

which is the same as inequality 5.1.

The united vector realization of $P_\sigma^{[2]}$ is $u_{ij} = w$ if $\sigma(i) = j$, else $u_{ij} = 0$. It is easy to see that every $P_\sigma^{[2]}$, hence every point of $\mathcal{B}^{[2]}$, satisfies the inequality (6.4).

If there exists a permutation σ such that $\sum_{(ij):\sigma(i)=j} n_{ij}$ is either equal to β or $\beta-1$, then $P_\sigma^{[2]}$ satisfies (6.4) with equality. In this case the plane $(A-(\beta-0.5)w)^2 = 0.25$ is a supporting plane of $\mathcal{B}^{[2]}$ and hence defines a face.

It may be pointed out that another inequality, which can define faces, is $(A - \beta w)^2 \geq 0$. But no known facet corresponds to this inequality.

All the sets of facets discussed in Chapter 3 can be restated in terms of united vectors. So the facets given by non-negativity conditions 3.2.2 are equivalent to $(u_{ij} + u_{kl} - 0.5w)^2 \geq 0.25$, for every i, j, k, l such that $i \neq k$ and $j \neq l$, the set of facets given by 3.2.3 are equivalent to $(u_{p_1q_1} + u_{p_2q_2} + u_{p_1q_2} - u_{kl} - 0.5w)^2 \geq 0.25$, where p_1, p_2, k are distinct and q_1, q_2, l are also distinct and $n \geq 6$, and those given by 3.2.4 are equivalent to $(u_{i_1j_1} + \cdots + u_{i_mj_m} - u_{kl} - 0.5w)^2 \geq 0.25$ where $i_1, \ldots, i_m, k$ are all distinct, $j_1, \ldots, j_m, l$ are also distinct and $n \geq 6, m \geq 3$.

Let P_1, P_2 be disjoint subsets of $[n]$. Similarly Q_1, Q_2 are also disjoint subsets of $[n]$. Then the 4-box inequality discussed in (JK97; Kai97) is $(-\sum_{i\in P_1, j\in Q_1} u_{ij} - \sum_{i\in P_2, j\in Q_2} u_{ij} + \sum_{i\in P_1, j\in Q_2} u_{ij} + \sum_{i\in P_2, j\in Q_1} u_{ij} - (\beta - 0.5)\omega)^2 \geq 0.25$. The 1-box inequality is equivalent to $(\sum_{i\in P_1, j\in Q_2} u_{ij} - (\beta - 0.5)\omega)^2 \geq 0.25$ and is obtained by setting $P_2 = Q_1 = \emptyset$ in the 4-box inequality, whereas the 2-box inequality corresponds to $(-\sum_{i\in P_2, j\in Q_2} u_{ij} + \sum_{i\in P_1, j\in Q_2} u_{ij} - (\beta - 0.5)\omega)^2 \geq 0.25$ and is obtained by setting $Q_1 = \emptyset$ in the 4-box inequality. All the facets listed in (JK97; Kai97) are special instances of either the 1-box or the 2-box inequality.

6.4.1 Geometry of the Feasible region

The linear conditions of SDP-LT-OPT (the feasible region of SDP-LT where the objective function takes value n) are same as those of LP-GI. This follows from the following argument.

Consider a solution Y in SDP-LT-OPT having a vector realization $\{u_{ij} | 1 \leq i, j \leq n\}$. Since the objective function achieves its maximum value, namely, n for Y, each set $\{u_{i1}, \ldots, u_{in}\}$ is a maximal orthogonal set. Similarly each set $\{u_{1i}, \ldots, u_{ni}\}$ is also a maximal orthogonal set. In that case from united vector property $\sum_i u_{ij} = \sum_j u_{ij} = w$. We then have $1 = w^T \cdot w = \sum_i u_{ij}^T \cdot w = \sum_i Y_{ij,\omega} = \sum_i Y_{ij,ij}$. Similarly $\sum_j Y_{ij,ij} = 1$. We also have $\sum_k Y_{ij,kl} = u_{ij}^T \cdot (\sum_k u_{kl}) = u_{ij}^T \cdot w = Y_{ij,ij}$. Similarly

$\sum_l Y_{ij,kl} = Y_{ij,ij}$. Note that these conditions are same as those in 2.1c-2.1e.

Since SDP-LT solutions are also positive semidefinite, we can deduce that $\mathcal{B}^{[2]}_{G_1G_2} \subseteq \mathcal{F}_{G_1G_2} \subseteq \mathcal{P}_{G_1G_2}$. Thus the performance of Algorithm 1 can only improve when the linear program is replaced by semidefinite program.

6.5 Conclusion

In this chapter we gave a CP formulation of GI and later studied its SDP relaxation. We showed that the optimal feasible region of CP-LT is equal to the convex hull of all isomorphisms between the given pair of graphs, i.e., $\mathcal{B}^{[2]}_{G_1G_2}$. We also showed that for a small set of special cases we can solve GI using SDP-LT. Finally we arrived at the general inequality 5.1 using a different approach.

Chapter 7

Experiments

7.1 Introduction

We present the results of the experiments we conducted to test non-isomorphism using Algorithm 1 on two families of graphs. We run two variants of Algorithm 1: one using LP-GI and the other using LP-GI-2 which is described in section 7.2. The two graph families are (i) strongly regular graphs and (ii) CFI-graphs. A d-regular n vertex graph is said to be (n, d, λ, μ)-strongly regular if all adjacent pairs of vertices have λ common neighbors and all non-adjecent pairs of vertices have μ common neighbors. CFI is short for Cai-Fürer-Immerman and these graphs are named so since they use a construction given by Cai, Fürer and Immerman in (CFI92).

We also experiment with the SDP-formulation of GI described in chapter 6. For these experiments we replace LP-GI in Algorithm 1 with SDP-LT.

The purpose of these experiments is to show the polynomiality of our algorithm on instances taken from the above mentioned classes of graphs, rather than compare its running time with softwares like *nauty*. As it stands, our running time is prohibitive and is not comparable to any practical software.

7.2 LP-GI-2: An Alternate Linear Program

Two graphs have an isomorphism σ if and only if $AP_\sigma = P_\sigma B$ where P_σ is the permutation matrix corresponding to σ and A, B are the adjacency matrices of the two graphs. LP-GI-2 is obtained by replacing the edge/non-edge condition 2.2b by

7.1b-7.1c. Each row of $P_\sigma^{[2]}$ is either vectorized P_σ matrix or a zero vector. Also the diagonal of $P_\sigma^{[2]}$ is the vectorized P_σ matrix. Therefore applying the commutation relation to these gives $\sum_p A_{kp} \cdot P_\sigma^{[2]}(pl,pl) = \sum_p P_\sigma^{[2]}(kp,kp) \cdot B_{pl}$ and $\sum_p A_{kp} \cdot P_\sigma^{[2]}(ij,pl) = \sum_p P_\sigma^{[2]}(ij,kp) \cdot B_{pl}$ for all i,j,k,l. Replacing $P_\sigma^{[2]}$ with variable Y we get a new set of graph conditions 7.1b-7.1c. The feasible region of LP-GI-2 is the polytope obtained by applying one Shirali-Adams lift step to the Tinhofer polytope, see(Mal14).

$$
\begin{aligned}
\text{LP-GI-2:}\quad \text{Find a point}\quad & Y \\
\text{subject to}\quad & 2.1a\text{-}2.1e && (7.1a)\\
& \sum_p A_{kp} \cdot Y_{pl,pl} = \sum_p Y_{kp,kp} \cdot B_{pl}\ ,\ \forall\ k,l && (7.1b)\\
& \sum_p A_{kp} \cdot Y_{ij,pl} = \sum_p Y_{ij,kp} \cdot B_{pl}\ \ ,\ \forall\ i,j,k,l && (7.1c)\\
& Y_{ij,kl} \geq 0 \qquad\qquad\qquad\qquad ,\ \forall\ i,j,k,l && (7.1d)
\end{aligned}
$$

In the following lemma we show that the feasible region of LP-GI-2 is contained in that of LP-GI. We basically re-prove (Mal14, Lemma 3.2) for the case of $k = 2$.

Lemma 7.2.1. *Feasible region of LP-GI-2 is contained in the feasible region of LP-GI for a given pair of graphs.*

Proof. We will show that the edge/non-edge conditions given by 2.2b in LP-GI are implied by the conditions 7.1c in LP-GI-2. This will prove the statement of the lemma since the remaining conditions in LP-GI are also present in LP-GI-2.

Consider 7.1c with $l = j$ and $A_{ik} = 0$. The left hand side of the equation reduces to 0 since $Y_{ij,pj} = 0$ for all $p \neq i$. That leads to $\sum_p Y_{ij,kp} \cdot B_{pj} = 0$. This combined with 7.1d results in $Y_{ij,kp} = 0$ for any k such that $A_{ik} = 0$ and for all p such that $B_{jp} = 1$.

Next consider a value of l such that $B_{jl} = 0$. Also let $k = i$. So the right hand side of 7.1c reduces to zero. That gives $\sum_p A_{ip} \cdot Y_{ij,pl} = 0$ which implies from 7.1d that $Y_{ij,pl} = 0$ for all p such that $A_{ip} = 1$ and any l such that $B_{jl} = 0$. □

Here is an interesting evidence of this fact. Point Y_c defined below for d-regular n-vertex graphs $G_1 = (V, E_1), G_2 = (V, E_2)$ belongs to the feasible region of LP-GI but it belongs to the feasible region of LP-GI-2 only when the graphs are strongly regular.

$$Y_c(ij, kl) = \begin{cases} \frac{1}{n} & \text{if } i = k \text{ and } j = l \\ \frac{1}{nd} & \text{if } \{i,k\} \in E_1 \text{ and } \{j,l\} \in E_2 \\ 0 & \text{if } \{i,k\} \in E_1 \text{ and } \{j,l\} \notin E_2 \\ 0 & \text{if } \{i,k\} \notin E_1 \text{ and } \{j,l\} \in E_2 \\ \frac{1}{n(n-1-d)} & \text{if } \{i,k\} \notin E_1 \text{ and } \{j,l\} \notin E_2 \end{cases}$$

Lemma 7.2.2. *Y_c belongs to LP-GI-2 if and only if G_1, G_2 are strongly regular.*

Proof. Consider the case when $A_{ik} = B_{jl} = 1$. Here the left hand side of 7.1c reduces to $\lambda^A_{ik}/(nd)$ where λ^A_{ik} is the number of common neighbors of vertices $i, k \in G_1$. Similarly the right hand side reduces to $\lambda^B_{jl}/(nd)$. Since 7.1c is true for all i, j, k, l, we can conclude that $\lambda^A_{ik} = \lambda^B_{jl}$ for all i, j, k, l or there is a common λ value for all pairs of adjacent vertices in the two graphs.

Next we consider the case when $A_{ik} = B_{jl} = 0$. Here the left hand side of 7.1c reduces to $(d - \mu^A_{ik})/n(n-1-d)$ and the right hand side to $(d - \mu^B_{jl})/n(n-1-d)$. μ^A_{ik} is the number of common neighbors of vertices $i, k \in G_1$ whereas μ^B_{jl} is the number of common neighbors of vertices $j, l \in G_2$. So we have $\mu^A_{ik} = \mu^B_{jl}$ for all i, j, k, l or there is a common μ value for all pairs of non-adjacent vertices in the two graphs.

From the above and the definition of a strongly regular graph, we can conclude that for 7.1c to hold for Y_c, G_1, G_2 must be strongly regular with the same set of parameters. □

Corollary 7.2.3. *Given a pair of d-regular n-vertex graphs G_1, G_2 that are not strongly regular, the point Y_c is feasible for LP-GI but infeasible for LP-GI-2.*

7.3 Strongly Regular Graphs

In any solution of LP-GI (hence also LP-GI-2) a vertex of degree d in G_1 is never mapped to a vertex of degree d' in G_2, for $d \neq d'$, as shown by the following lemma.

Lemma 7.3.1. *$Y_{ij,ij} = 0$ if $deg(i) \neq deg(j)$ for $i \in V(G_1), j \in V(G_2)$.*

Proof. Let $d_1 = deg_{G_1}(i)$ and $d_2 = deg_{G_2}(j)$. So $d_1 \neq d_2$. Let $N_1(i)$ denote the open neighborhood of a vertex i in G_1 and $N_2(j)$ denote the open neighborhood of vertex j in G_2. So $\sum_{k \in N_1(i)} \sum_{l \in N_2(j)} Y_{ij,kl} = \sum_{k \in N_1(i)} Y_{ij,ij} = d_1 Y_{ij,ij}$. But rewriting the left hand side expression as $\sum_{l \in N_2(j)} \sum_{k \in N_1(i)} Y_{ij,kl}$ leads to $\sum_{l \in N_2(j)} Y_{ij,ij} = d_2 Y_{ij,ij}$. Since $d_1 \neq d_2$, $Y_{ij,ij} = 0$. $\square$

Lemma 7.3.1 suggests that the real difficulty lies in distinguishing between vertices that have the same degree. Moreover, from lemma 7.2.2 we know that both LP-GI and LP-GI-2 fail to establish non-isomorphism if the given non-isomorphic pair is strongly regular with the same parameters. The following quote from (Bab14), sums up the relevance of strongly regular graphs in the graph isomorphism problem.

> "Strongly regular graphs, while not believed to be Graph Isomorphism (GI) complete, have long been recognized as hard cases for GI, and, in this author's view, present some of the core difficulties of the general GI problem."

The best known graph isomorphism algorithm for strongly regular graphs runs in time that is slightly better than that for general graphs. In (Spi96), Spielman had given an algorithm with time exponential in $\tilde{O}(n^{1/3})$. This was recently improved by Babai et. al. to $exp(\tilde{O}(n^{1/5}))$ (BCS+13). Recall that the best algorithm for general graphs runs in $exp(\tilde{O}(n^{1/2}))$ time.

7.4 The Cai-Fürer-Immerman construction

CFI graphs are formed by replacing each vertex of degree k in some base graph with the CFI gadget F_k. We will refer to the instance of F_k used for a vertex u by u-gadget. We experiment with CFI graphs having 3-regular graphs as base graphs. Figure 7.1(a) shows the gadget F_3 where $x_1, y_1, x_2, y_2, x_3, y_3$ are the interface vertices and the remaining vertices are internal vertices. Figure 7.1(b) shows the same gadget by suppressing the internal vertices. An edge, $\{u, v\}$ in the base graph is replaced by a bond. Figure 7.2(a) shows a regular-bond between the u-gadget and the v-gadget as a pair of edges $\{x'_i, x''_j\}$ and $\{y'_i, y''_j\}$ for some i, j. Here x'_i, y'_i are i-th interface vertices of the u-gadget and x''_j, y''_j are j-th interface vertices of the v-gadget. Figure 7.2(b) shows a twisted bond. Here the edges are $\{x'_i, y''_j\}$ and $\{x''_j, y'_i\}$.

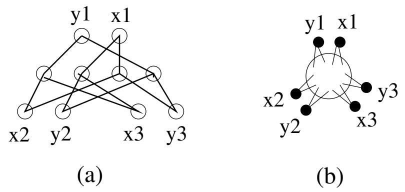

Figure 7.1: (a) CFI Gadget F_3, (b) Symbolic F_3

Figure 7.2: (a) A Regular Bond, (b) A Twisted Bond

Figure 7.3(a) shows the CFI graph with K_4 as base graph. Figure 7.3(b) shows the same graph with one of the bonds replaced by a twisted bond. Such pairs are non-isomorphic. We use such pairs for experiments.

The linear program LP-GI (also LP-GI-2) have $\theta(n^4)$ variables. Hence we are required to keep n low in our experiments. We consider two variants of CFI graphs having fewer vertices than the standard construction but with the inherent difficulty preserved. Whenever the standard CFI graph turns out to be too large for our experiments, we use these variants. These variants are formed by contracting edges in the bonds as shown in figures 7.4(b) and 7.4(c). Observe that a p-vertex 3-regular base graph results in a $10p$-vertex standard CFI graph whereas these variants give $7p$ and $4p$ vertex graphs respectively.

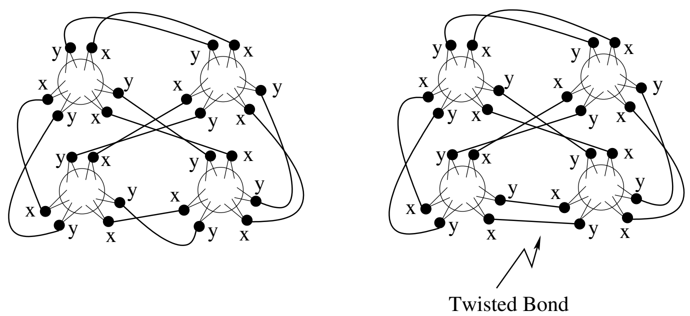

Figure 7.3: (a) CFI Graph based on K_4, (b) Same graph but with one Twisted Bond

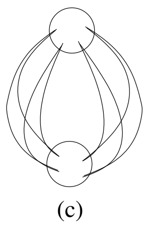

Figure 7.4: Bonds: (a) Standard, (b) After one level of contraction, (c) After two levels of contractions

7.5 Experiments with the Linear Program

In this section we give details of the software used and the settings under which the experiments were carried out. Finally we present our results.

7.5.1 Experimental setup

We use the GNU Linear Programming Kit (GLPK) version 4.52 to solve LP-GI/LP-GI-2 in our algorithm on graphs taken from the two families described above. The linear program is specified using the GNU MathProg Modeling Language (GMPL). The model file takes its data from a separate file. So for every pair of non-isomorphic graphs we have a separate data file whereas the model file is common. The algorithm is implemented in C language making use of the appropriate GLPK APIs. The experiments are run on a desktop computer running ubuntu 14.04 and having 16 GiB RAM with Intel Core i7-4770 CPU @ 3.40GHz $\times$ 8 processors. However, no parallelization is done. For the linear program, the default primal simplex algorithm is used and maximum 100000 iterations are allowed. If a program fails to converge in these many iterations, then it is taken to have a feasible solution.

The strongly regular graphs are taken from the collection available at (Spe). We take 10 distinct pairs of non-isomorphic graphs for each (n, d, λ, μ). Wherever that many are not available, all possible distinct pairs are taken. The largest graph used has 50 vertices.

For CFI graphs used in the experiments, the base graph is always a three regular graph. Apart from one graph, which is the 3-dimensional cube, all other base graphs are $2k$-cycles with chords. The vertices are labeled from $1 \ldots 2k$ and an edge is present between two vertices having labels u, v iff $u - v \equiv \pm 1 \pmod{2k}$ or

$u - v \equiv k \pmod{2k}$. We select those values of k for which the number of vertices in at least one of the three variants stays within 56.

7.5.2 Results

Results of experiments with non-isomorphic pairs of strongly regular graphs appear in table 7.1 whereas the results of experiments with non-isomorphic pairs of CFI graphs are presented in table 7.2. In all the cases of strongly regular graphs we observe that the feasible region is always zero-one reducible and in most cases Algorithm 1 is faster with LP-GI-2 than with LP-GI in terms of the number of times the respective linear program is solved. In the case of CFI graphs, only 3 out of 10 cases were zero-one reducible ($\tau = 0$)for LP-GI, but all the cases were zero-one reducible for LP-GI-2. Note that the algorithm remains poly-time as long as $\tau = O(logn)$.

Table 7.1

Class	Pair	No. of iterations of LP-GI	No. of iterations of LP-GI-2	Class	Pair	No. of iterations of LP-GI	No. of iterations of LP-GI-2
(16,6,2,2)	1-2	96	96	(36,14,4,6)	1-4	504	504
(25,12,5,6)	1-2	408	300	(36,14,4,6)	1-5	504	504
(25,12,5,6)	1-3	408	300	(36,14,4,6)	2-3	504	504
(25,12,5,6)	1-4	408	300	(36,14,4,6)	2-4	504	504
(25,12,5,6)	1-5	516	300	(36,14,4,6)	2-5	504	504
(25,12,5,6)	2-3	408	300	(36,14,4,6)	3-4	504	504
(25,12,5,6)	2-4	408	300	(36,14,4,6)	3-5	504	504
(25,12,5,6)	2-5	516	300	(36,14,4,6)	4-5	504	504
(25,12,5,6)	3-4	408	300	(37,18,8,9)	1-2	666	666
(25,12,5,6)	3-5	516	300	(37,18,8,9)	1-3	666	666
(25,12,5,6)	4-5	516	300	(37,18,8,9)	1-4	666	666
(26,10,3,4)	1-2	260	260	(37,18,8,9)	1-5	666	666
(26,10,3,4)	1-3	260	260	(37,18,8,9)	2-3	666	666
(26,10,3,4)	1-4	260	260	(37,18,8,9)	2-4	666	666
(26,10,3,4)	1-5	260	260	(37,18,8,9)	2-5	666	666
(26,10,3,4)	2-3	260	260	(37,18,8,9)	3-4	666	666
(26,10,3,4)	2-4	260	260	(37,18,8,9)	3-5	666	666
(26,10,3,4)	2-5	260	260	(37,18,8,9)	4-5	666	666
(26,10,3,4)	3-4	260	260	(40,12,2,4)	1-2	480	480
(26,10,3,4)	3-5	260	260	(40,12,2,4)	1-3	480	480
(26,10,3,4)	4-5	270	260	(40,12,2,4)	1-4	480	480
(28,12,6,4)	1-2	336	336	(40,12,2,4)	1-5	480	480
(28,12,6,4)	1-3	336	336	(40,12,2,4)	2-3	516	480
(28,12,6,4)	1-4	336	336	(40,12,2,4)	2-4	552	480
(28,12,6,4)	2-3	336	336	(40,12,2,4)	2-5	552	480
(28,12,6,4)	2-4	336	336	(40,12,2,4)	3-4	552	480

Continued on next page

Table 7.1 – continued from previous page

Class	Pair	No. of iterations of LP-GI	No. of iterations of LP-GI-2	Class	Pair	No. of iterations of LP-GI	No. of iterations of LP-GI-2
(28,12,6,4)	3-4	336	336	(40,12,2,4)	3-5	552	480
(29,14,6,7)	1-2	406	406	(40,12,2,4)	4-5	552	480
(29,14,6,7)	1-3	406	406	(45,12,3,3)	1-2	$*$	540
(29,14,6,7)	1-4	406	406	(45,12,3,3)	1-3	$*$	540
(29,14,6,7)	1-5	406	406	(45,12,3,3)	1-4	$*$	540
(29,14,6,7)	2-3	406	406	(45,12,3,3)	1-5	$*$	540
(29,14,6,7)	2-4	406	406	(45,12,3,3)	2-3	$*$	540
(29,14,6,7)	2-5	406	406	(45,12,3,3)	2-4	$*$	540
(29,14,6,7)	3-4	406	406	(45,12,3,3)	2-5	$*$	540
(29,14,6,7)	3-5	406	406	(45,12,3,3)	3-4	$*$	540
(29,14,6,7)	4-5	406	406	(45,12,3,3)	3-5	$*$	540
(35,18,9,9)	1-2	630	630	(45,12,3,3)	4-5	$*$	540
(35,18,9,9)	1-3	630	630	(50,21,8,9)	1-2	2331	1050
(35,18,9,9)	1-4	630	630	(50,21,8,9)	1-3	2331	1050
(35,18,9,9)	1-5	630	630	(50,21,8,9)	1-4	2331	1050
(35,18,9,9)	2-3	630	630	(50,21,8,9)	1-5	2331	1050
(35,18,9,9)	2-4	630	630	(50,21,8,9)	2-3	2331	1050
(35,18,9,9)	2-5	630	630	(50,21,8,9)	2-4	2331	1050
(35,18,9,9)	3-4	630	630	(50,21,8,9)	2-5	2331	1050
(35,18,9,9)	3-5	630	630	(50,21,8,9)	3-4	2331	1050
(35,18,9,9)	4-5	630	630	(50,21,8,9)	3-5	2331	1050
(36,14,4,6)	1-2	504	504	(50,21,8,9)	4-5	2331	1050
(36,14,4,6)	1-3	504	504				

Table 7.1: Results of experiments with non-isomorphic strongly regular graphs from (Spe). In LP-GI experiments 86 cases had zero-one reduction and 10 cases did not converge. In LP-GI-2 all cases converged and all had zero-one reduction.

7.6 Experiments with the Lovász Theta function

7.6.1 Experimental setup

We use a public domain software (MPRW11; MPRW09) based on Matlab to solve the semidefinite program for the Lovász Theta function. The experiments are run on a shared Intel(R) Xeon(R) CPU E7450 @2.40$GHz \times 24$ machine that runs Matlab R2013a (8.1.0.604) on GNU/Linux OS.

We replace LP-GI in Algorithm 1 with SDP-LT but without the non-negativity conditions 6.2b. This is so because the SDP solver does not allow the linear con-

Base Graph	CFI Construction	n	No. of iterations of LP-GI	No. of iterations of LP-GI-2	τ (LP-GI)	τ (LP-GI-2)
2×2 cycle	factor-4	16	528	144	0	0
2×3 cycle	factor-4	24	2736	408	0	0
2×4 cycle	factor-4	32	NA	2592	> 0	0
3-dim cube	factor-4	32	NA	2592	> 0	0
2×5 cycle	factor-4	40	NA	8400	> 0	0
2×2 cycle	factor-10	40	9360	1440	0	0
2×3 cycle	factor-7	42	NA	1650	> 0	0
2×6 cycle	factor-4	48	NA	20256	> 0	0
2×4 cycle	factor-7	56	NA	13752	> 0	0
3-dim cube	factor-7	56	NA	19184	> 0	0

Table 7.2: Results of experiments with non-isomorphic graph pairs that use the CFI construction. In this case only three cases were zero-one reducible for LP-GI but in LP-GI-2 all cases were zero-one reducible.

ditions to be specified as inequalities. However, since the diagonal vector of a PSD matrix is always non-negative, we can create a larger matrix with the original matrix as a principal sub-matrix, and with condition that all the diagonal entries outside the principal sub-matrix are equal to the off-diagonal entries of the original matrix. This would result in a variable matrix of dimension $(N + \binom{N}{2}) \times (N + \binom{N}{2})$ if the original matrix was of dimension $N \times N$. This is not practical since our original matrix is $n^2 \times n^2$ where n is the number of vertices in the input graphs. So for even ten vertex graphs, the variable matrix of the SDP would blow up to 5050×5050. Hence, we run the solver on the original variable matrix without the non-negativity conditions and it turns out that we can still differentiate non-isomorphic strongly regular graphs in polynomial time as summarized in table 7.3. However, the same is not true for CFI graphs. Here, we do get an optimal solution but with the solution matrix having negative entries. Thus our experiments with the Lovász theta function on CFI graphs are not interesting.

In these experiments we only set the diagonal entries of the first block to 1 one at a time and run the SDP solver. In all the cases we found that ϑ was less than n in all the n runs leading to the conclusion that the graphs are non-isomorphic.

7.6.2 Results

Since SDP-LT is an optimization program, we treat a solution having value less than n (its optimum value) as an infeasible solution. We observe than n iterations of SDP-LT suffice in all the cases. The variables examined by algorithm 1 are Y_{1j1j} $\forall j \in [n]$. Moreover, each of these variables when set to 1 independently, also leads to an infeasible solution. So the value that we report in column four of table 7.3 is $max_j\{\vartheta|Y_{1j1j} = 1\}$. First column of Table 7.3 identifies the strongly regular graph family by giving its parameters. Graphs of each family are indexed as $1, 2, 3, \ldots$. The second column identifies the two graphs on which the experiment was run.

7.7 Conclusion

In this chapter we presented our experience with a limited set of experimentation. Due to the large sizes of the linear and semidefinite programs, we could only experiment with graphs of modest sizes. However, the results are encouraging if not conclusive. We found that for most of the graphs the feasible region of LP-GI for non-isomorphic graphs is zero-one reducible while the feasible region of LP-GI-2 is zero-one reducible for all the graphs that we experimented with. In cases where the feasible region is zero-one reducible for both LP-GI and LP-GI-2, it turns out that the number of iterations is less for LP-GI-2. This calls for a closer look at the feasible region of LP-GI-2, which is contained inside that of LP-GI. In the case of CFI graphs obtained by a factor-4 construction, observe that these graphs are regular but not strongly regular. Hence from lemma 7.2.2 the feasible region of LP-GI-2 for such graphs is strictly contained inside that of LP-GI. This may account for the better performance of LP-GI-2 for these cases as shown in table 7.2.

It must be noted here that the results of chapter 4 are independent of which of the two linear programs we use. Moreover, for cases where the feasible region of LP-GI is not zero-one reducible, either some of the supporting planes corresponding to the minimal violated inequalities intersect the feasible region or none of the inequalities described in section 4.2 are violated and the unknown facet planes of $\mathcal{B}^{[2]}$ are not as "nice". But since the feasible region of LP-GI-2 for the same pair of graphs turns out to be zero-one reducible, an alternative approach is desired that in addition to looking at the violated inequalities also looks at the geometry of the feasible region.

Class	Pair	n	ϑ	Class	Pair	n	ϑ	Class	Pair	n	ϑ
(16,6,2,2)	1-2	16	12.0	(29,14,6,7)	2-5	29	21.1	(37,18,8,9)	3-5	37	26.5
(25,12,5,6)	1-2	25	21.1	(29,14,6,7)	3-4	29	22.3	(37,18,8,9)	4-5	37	26.3
(25,12,5,6)	1-3	25	20.9	(29,14,6,7)	3-5	29	20.9	(40,12,2,4)	1-2	40	33.3
(25,12,5,6)	1-4	25	21.0	(29,14,6,7)	4-5	29	20.9	(40,12,2,4)	1-3	40	35.8
(25,12,5,6)	1-5	25	19.0	(35,18,9,9)	1-2	35	26.0	(40,12,2,4)	1-4	40	33.1
(25,12,5,6)	2-3	25	20.9	(35,18,9,9)	1-3	35	26.0	(40,12,2,4)	1-5	40	36.2
(25,12,5,6)	2-4	25	21.0	(35,18,9,9)	1-4	35	25.8	(40,12,2,4)	2-3	40	35.9
(25,12,5,6)	2-5	25	19.0	(35,18,9,9)	1-5	35	25.8	(40,12,2,4)	2-4	40	33.5
(25,12,5,6)	3-4	25	21.0	(35,18,9,9)	2-3	35	26.3	(40,12,2,4)	2-5	40	36.0
(25,12,5,6)	3-5	25	18.7	(35,18,9,9)	2-4	35	28.5	(40,12,2,4)	3-4	40	34.1
(25,12,5,6)	4-5	25	21.0	(35,18,9,9)	2-5	35	27.2	(40,12,2,4)	3-5	40	34.8
(26,10,3,4)	1-2	26	21.9	(35,18,9,9)	3-4	35	28.5	(40,12,2,4)	4-5	40	33.1
(26,10,3,4)	1-3	26	19.5	(35,18,9,9)	3-5	35	26.9	(45,12,3,3)	1-2	45	41.0
(26,10,3,4)	1-4	26	19.2	(35,18,9,9)	4-5	35	26.1	(45,12,3,3)	1-3	45	40.1
(26,10,3,4)	1-5	26	19.2	(36,14,4,6)	1-2	36	31.0	(45,12,3,3)	1-4	45	40.2
(26,10,3,4)	2-3	26	19.4	(36,14,4,6)	1-3	36	31.0	(45,12,3,3)	1-5	45	40.4
(26,10,3,4)	2-4	26	19.2	(36,14,4,6)	1-4	36	29.3	(45,12,3,3)	2-3	45	40.2
(26,10,3,4)	2-5	26	19.2	(36,14,4,6)	1-5	36	31.8	(45,12,3,3)	2-4	45	40.6
(26,10,3,4)	3-4	26	19.8	(36,14,4,6)	2-3	36	30.1	(45,12,3,3)	2-5	45	40.4
(26,10,3,4)	3-5	26	19.9	(36,14,4,6)	2-4	36	30.9	(45,12,3,3)	3-4	45	36.2
(26,10,3,4)	4-5	26	21.9	(36,14,4,6)	2-5	36	29.0	(45,12,3,3)	3-5	45	40.4
(28,12,6,4)	1-2	28	20.2	(36,14,4,6)	3-4	36	30.0	(45,12,3,3)	4-5	45	40.4
(28,12,6,4)	1-3	28	22.6	(36,14,4,6)	3-5	36	29.7	(50,21,8,9)	1-2	50	36.5
(28,12,6,4)	1-4	28	24.0	(36,14,4,6)	4-5	36	30.9	(50,21,8,9)	1-3	50	36.4
(28,12,6,4)	2-3	28	24.0	(37,18,8,9)	1-2	37	26.6	(50,21,8,9)	1-4	50	36.4
(28,12,6,4)	2-4	28	20.7	(37,18,8,9)	1-3	37	26.6	(50,21,8,9)	1-5	50	36.4
(28,12,6,4)	3-4	28	24.0	(37,18,8,9)	1-4	37	26.7	(50,21,8,9)	2-3	50	36.3
(29,14,6,7)	1-2	29	21.3	(37,18,8,9)	1-5	37	27.0	(50,21,8,9)	2-4	50	36.3
(29,14,6,7)	1-3	29	21.0	(37,18,8,9)	2-3	37	26.4	(50,21,8,9)	2-5	50	36.3
(29,14,6,7)	1-4	29	21.0	(37,18,8,9)	2-4	37	26.4	(50,21,8,9)	3-4	50	36.4
(29,14,6,7)	1-5	29	21.1	(37,18,8,9)	2-5	37	26.4	(50,21,8,9)	3-5	50	36.4
(29,14,6,7)	2-3	29	21.2	(37,18,8,9)	3-4	37	26.5	(50,21,8,9)	4-5	50	36.4
(29,14,6,7)	2-4	29	21.1								

Table 7.3: Results of experiments with non-isomorphic strongly regular graphs from (Spe)

Chapter 8

Conclusions

If an optimization problem can be modeled as an integer linear program (IP), then our objective is to compute an optimal integer point solution. In general solution of an IP requires exponentially large computation time. If the progam is relaxed to the corresponding linear program (i.e., variables are allowed to be any real number in the range $[0, 1]$, then the feasible region, call it polyhedron P_0, contains all the integer solutions but all its extreme points may not be integer points. If the variables in the IP are restricted to $\{0, 1\}$, then there is a way to tackle this problem. Let Q denote the convex hull of all the integer points in P_0. Sherali-Adams and some other lift-and-project techniques allow us to iteratively enhance the constraints of the linear program such that the polytopes of the successive iterations progressively shrink to finally match Q. Actually, these successive linear programs are in larger and larger dimensional spaces. Let $P_1, P_2, \ldots$ be the projections of the polytopes of the feasible regions of the successive programs, into the original space. Then $P_0 \supseteq P_1 \supseteq P_2 \cdots \supseteq P_n = Q$. Here n denotes the number of variables in the original program. In this series, the complexity of optimizing a linear function over the k-th program is $O(n^k)$.

This iterative refinement technique always requires more than a constant number of iterations for NP-hard problems. But since GI is not known to be NP-hard, researchers investigated this technique for GI hoping that it may take only a constant number of iterations, say c, to get $P_c = Q$. That would have solved the GI problem in n^c time. Unfortunately it turns out that there are graph pairs for which the series requires $\Omega(n)$ iterations.

In present work we studied the polytope, in the higher space, after one iteration of Sherali-Adams series to find out if there are such features of this polytope which may allow us to deduce isomorphism directly. We observed that for non-isomorphic graphs G_1, G_2, the Tinhofer polytope is contained inside the Birkhoff polytope (the convex hull of all possible integer solutions), whereas our polytope in the extended space, $\mathcal{P}_{G_1G_2}$, lies completely outside $\mathcal{B}^{[2]}$ (the convex hull of all possible integer solutions) and sandwiched between two polytopes, one of them being $\mathcal{B}^{[2]}$. The other polytope, referred to as $\mathcal{P}$ in the thesis, can be thought of as the superset of $\mathcal{P}_{G_1G_2}$ for all G_1, G_2 for a given n. So we have for non-isomorphic G_1, G_2, $\mathcal{P}_{G_1G_2} \subseteq \mathcal{P} \setminus \mathcal{B}^{[2]}$.

The highlights of the thesis are essentially Chapters 3 and 4. In Chapter 3 we identified two new classes of facets of the QAP polytope (referred to as $\mathcal{B}^{[2]}$ in the thesis) and gave a general inequality that includes all the known facets of $\mathcal{B}^{[2]}$. In chapter 4 we defined a partial ordering on the exponential sized families of facets, such that the minimal faces in the ordering are never violated by any point in $\mathcal{P}_{G_1G_2}$. This allowed us to introduce the concept of a *minimal violated inequality*, an inequality X violated by some point p in $\mathcal{P}_{G_1G_2}$ such that all inequalities less than X in the partial ordering, are satisified by p. Then we showed that if $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ is separated from $\mathcal{B}^{[2]}$ by a single such inequality, then it is easy (means poly-time) to determine if the given pair of graphs are isomorphic or not. Later in the same chapter, we studied the general case when the region $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ is separated from $\mathcal{B}^{[2]}$ by k such inequalities. For this case, we gave an algorithm that takes roughly $O(n^k)$ time to determine if the given pair of graphs are isomorphic or not. This shows that the complexity of the graph isomorphism problem depends on the number of facet planes of $\mathcal{B}^{[2]}$ that separate $\mathcal{P}_{G_1G_2} \setminus \mathcal{B}^{[2]}$ from $\mathcal{B}^{[2]}$.

We conclude by describing a couple of open problems that we feel are worth pursuing.

8.1 Open Problems

8.1.1 GI belongs to co-NP?

It is easy to see that GI belongs to the class NP. To verify that two graphs are isomorphic, all we need is a permutation σ and verification takes $O(n^2)$ time. However, does a certificate exist that can be used to verify *quickly* (i.e., in poly-time) that the given graphs are non-isomorphic, remains an open problem.

In this thesis we have seen that the feasible region of LP-GI, $\mathcal{P}_{G_1G_2}$, is sandwiched between the polytopes $\mathcal{P}$ and $\mathcal{B}^{[2]}$, for non-isomorphic graphs. So there exists a minimal set of facet defining inequalities of $\mathcal{B}^{[2]}$ that separates $\mathcal{P}_{G_1G_2}$ from $\mathcal{B}^{[2]}$. Adding these to LP-GI would make it infeasible. So our certificate for a given pair of non-isomorphic graphs G_1, G_2, is a description of these inequalities. Clearly, if we can show that the resulting linear program can be solved in polynomial time by the ellipsoid method, then we can establish that GI also belongs to the class co-NP.

8.1.2 Geometry of the Feasible Region

Consider the polytope $\mathcal{P}$ for a given n. In the thesis we have established that the feasible region of LP-GI for graphs G_1, G_2 corresponds to a unique face of $\mathcal{P}$ (upto taking complements of G_1 and G_2). So we can view our linear program as a function mapping a pair of graphs to some face of $\mathcal{P}$. So for any pair of graphs on n vertices each, the problem of detecting if they are isomorphic reduces to that of identifying if the corresponding face of $\mathcal{P}$ has an integral vertex. The question we are asking is that does the geometry of a face of $\mathcal{P}$ having one or more integral vertices, *differ* significantly from that of a face having only fractional vertices. More importantly can we provide a characterization that can be efficiently checked. In the case of rigid graphs, the geometry looks interesting. Here we need to determine if the feasible face touches the polytope $\mathcal{B}^{[2]}$ at exactly one point which is the vertex of $\mathcal{B}^{[2]}$ corresponding to the identity permutation.

Appendix A

Dimension of $\mathcal{B}^{[2]}$

A.1 Introduction

Symmetric vectorization is the row-major vectorization of a $r \times r$ symmetric matrix, ignoring the lower triangular entries, resulting in a $(r^2 + r)/2$ dimensional vector. Define an $(n^4 + n^2)/2 \times n!$ matrix A in which each column is the symmetric vectorization of a $P_\sigma^{[2]}$ for all $\sigma \in S_n$.

We will consider the space $\mathbb{R}^{(n^4+n^2)/2}$ in which the axes of a basis are indexed as $(ab)(xy)$ where $a, b, x, y \in [n]$, (ab) is an unordered pair, and (xy) is an ordered pair if $a \neq b$ otherwise it is also unordered. In case of unordered pairs we adopt the convention that the first element is smaller. Therefore there are $\binom{n}{2}.n^2+n.(\binom{n}{2}+n) = (n^4 + n^2)/2$ axes. If v denotes the symmetric vectorization of some $P_\sigma^{[2]}$, then an element in the upper triangular $P_\sigma^{[2]}$ matrix, $(P_\sigma^{[2]})_{ax,by}$ will be denoted by $v(ab, xy)$ in $\mathbb{R}^{(n^4+n^2)/2}$.

Define the $\frac{n^4+n^2}{2} \times \frac{n^4+n^2}{2}$ positive semidefinite matrix $B = AA^T$. It is easy to see that $B_{(ij)(kl),(ab)(xy)}$ is equal to the number of permutations, σ, such that $\sigma(\min\{i, j\}) = k, \sigma(\max\{i, j\}) = l,\sigma(\min\{a, b\}) = x, \sigma(\max\{a, b\}) = y$. Clearly possible values of entries in B are $(n - 1)!, (n - 2)!, (n - 3)!, (n - 4)!$, and 0.

In section A.2 we identify four eigenvalues of B, namely, $(3/2)n!$, $n(n - 3)!$, $(n-1)!/(n-3)$, $2n(n-2)!$. We also determine 1, $\binom{n-1}{2}^2$, $(\binom{n-1}{2}-1)^2$, $(n-1)^2$ linearly independent eigenvectors for these eigenvalues respectively. Therefore $rank(A) =$

$rank(B) \geq 1 + \binom{n-1}{2}^2 + (\binom{n-1}{2} - 1)^2 + (n-1)^2 = n!/(2(n-4)!) + (n-1)^2 + 2.$

The dimension of the linear space spanned by $P_\sigma^{[2]}$s is equal to the rank of A which, in turn, is equal to the rank of B. So the dimension of the affine plane spanned by $P_\sigma^{[2]}$s is $rank(B) - 1$.

In order to compute $rank(B)$ we will first establish that B has no other eigenvalues and the dimensions of the eigenspaces are equal to (not greater than) 1, $\binom{n-1}{2}^2$, $(\binom{n-1}{2} - 1)^2$, $(n-1)^2$ respectively.

Let D denote the diagonalized B matrix. Then the contribution of these eigenvectors to the trace of D is $(3/2)n! \times 1 + n(n-3)! \times \binom{n-1}{2}^2 + (n-1)!/(n-3) \times (\binom{n-1}{2} - 1)^2 + 2n(n-2)! \times (n-1)^2 = \binom{n+1}{2}n!$. If the rank of D is larger than $n!/(2(n-4)!) + (n-1)^2 + 2$, then the trace must be strictly greater than $\binom{n+1}{2}n!$ because B is a positive semidefinite matrix.

Now we compute the trace of B, which is equal to that of D. The rank of D is same as that of B so we can directly compute it. Let us define a notation $X(i_1 \to j_1, i_2 \to j_2, \ldots, i_k \to j_k)$ which denotes the total number of permutations in which i_1 maps to j_1, i_2 maps to j_2, ..., i_k maps to j_k. To determine the trace of B, recall that $B_{(ij)(kl),(ab)(xy)}$ is equal to $X(i \to k, j \to l, a \to x, b \to y)$ where we assume that $i \leq j$ and $a \leq b$. Then the trace of B is equal to $\sum_{(ab)(xy)} B_{(ab)(xy),(ab)(xy)} = \sum_{a,x} X(a \to x) + \sum_{a<b, x \neq y} X(a \to x, b \to y) = n^2(n-1)! + 2\binom{n}{2}^2(n-2)! = \binom{n+1}{2}n!$. This implies that $rank(B)$ and hence the dimension of the linear space spanned by $P_\sigma^{[2]}$s is $n!/(2(n-4)!) + (n-1)^2 + 2$. So the dimension of the affine space spanned by $P_\sigma^{[2]}$s is one less, i.e., $n!/(2(n-4)!) + (n-1)^2 + 1$, which is nothing but the dimension of $\mathcal{B}^{[2]}$.

A.2 Eigenvalues and eigenvectors of the matrix of permutation counts

A.2.1 First Eigenvalue

The first eigenvalue has only one eigenvector v which is given by $v_{(aa)(xx)} = n - 1$ for all a, x and $v_{(ab)(xy)} = 1$ for all $a \neq b$ and all $x \neq y$. All other entries of v are zero. Let $B_{(ij)(kl)}$ denote the row of B with index $(ij)(kl)$. From the definition of B it is obvious that row vectors $B_{(ij)(kl)}$ are non-zero only if either $i = j$ and $k = l$ or $i \neq j$ and $k \neq l$. We will consider the inner product of $B_{(ij)(kl)}$ and v for these two cases. Case (1): the details of various terms in $B_{(ii)(kk)} \cdot v$ are as follows:

term index	term value	no. of terms	total contribution
$(ii)(kk)$	$(n-1)(n-1)!$	1	$(n-1)(n-1)!$
$(ib)(ky)$	$1(n-2)!$	$(n-1)^2$	$(n-1)^2(n-2)!$
$(aa)(xx)$	$(n-1)(n-2)!$	$(n-1)^2$	$(n-1)^2(n-1)!$
$(ab)(xy)$	$1(n-3)!$	$2\binom{n-1}{2}^2$	$2(n-3)!\binom{n-1}{2}^2$

So the inner product is the sum total of all the contributions which turns out to be $(n-1)3n!/2$ or $v_{(ii)(kk)}3n!/2$.

Case (2): same details in $B_{(ij)(kl)} \cdot v$ are as follows:

term index	term value	no. of terms	total contribution
$(ij)(kl)$	$1(n-2)!$	1	$(n-2)!$
$(ii)(kk)$	$(n-1)(n-2)!$	1	$(n-1)(n-2)!$
$(jj)(ll)$	$(n-1)(n-2)!$	1	$(n-1)(n-2)!$
$(ib)(ky)$	$1(n-3)!$	$(n-2)^2$	$(n-2)^2(n-3)!$
$(jb)(ly)$	$1(n-3)!$	$(n-2)^2$	$(n-2)^2(n-3)!$
$(aa)(xx)$	$(n-1)(n-3)!$	$(n-2)^2$	$(n-1)(n-2)^2(n-3)!$
$(ab)(xy)$	$1(n-4)!$	$2\binom{n-2}{2}^2$	$2(n-4)!\binom{n-2}{2}^2$

In this case the inner product reduces to $3n!/2$ which is same as $v_{(ij)(kl)}3n!/2$. Hence v is an eigenvector with eigenvalue $3n!/2$.

A.2.2 Second Eigenvalue

Let $G_1 = (V_1, E_1, w_1)$ be an undirected and $G_2 = (V_2, E_2, w_2)$ be a directed edge-weighted graphs, where V_1 and V_2 are subsets of $[n]$. Neither graph has loop-edges and $(x, y) \in E_2$ if and only if $(y, x) \in E_2$. For each $e_1 = (ab) \in E_1$ and $e_2 = (xy) \in E_2$ we associate (e_1, e_2) with axis (ab, xy). Note (ab, xy) is different from (ab, yx). We define a vector $v(G_1, G_2)$ or simply v as follows: if $(ab) \in E_1$ and $(xy) \in E_2$ then $v_{(ab)(xy)} = w_1(ab).w_2(xy)$, otherwise $v_{(ab)(xy)} = 0$. In this subsection and in the following subsection we will show that $v(G_1, G_2)$ are eigenvectors for various instances of G_1 and G_2.

We will use following notations regarding G_1 and G_2. $\delta(a)$ will denote the edges incident on vertex $a \in V_1$ and $N(a)$ will denote the neighbors of vertex $a \in V_1$. $N[a]$

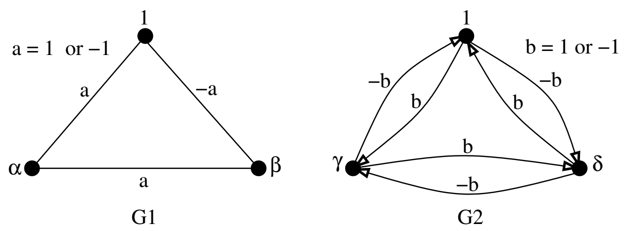

Figure A.1: Index Graphs for the Eigenvectors for Second Eigenvalue

will denote the closed neighborhood of a which is $N(a) \cup \{a\}$. Graph G_2 is directed, so $\delta^{\rightarrow}(x)$ denotes the set of outgoing edges and $\delta^{\leftarrow}(x)$ denotes the incoming edges incident on $x \in V_2$. Similarly neighborhoods will be denoted by $N^{\rightarrow}(x)$ and $N^{\leftarrow}(x)$ respectively.

In this subsection we assume following properties of the weight functions: (i) all weights are 1 or -1; (ii) $\sum_{b \in N(a), a<b} w_1(a,b) - \sum_{b \in N(a), a>b} w_1(a,b) = 0$; (iii) w_2 be such that $w_2(x,y) = -w_2(y,x)$ for all $(x,y) \in E_2$, and (iv) $\sum_{y \in N^{\rightarrow}[x]} w_2(x,y) = \sum_{y \in N^{\leftarrow}[x]} w_2(y,x) = 0$ for each $x \in V_2$.

Lemma A.2.1. *The inner product $B_{(ij)(kl)} \cdot v(G_1, G_2)$ is zero unless $i, j \in V_1$ and $k, l \in V_2$.*

Proof. If k, l both are out of V_2 then for each (a, b) we will get contributions from (x, y) and (y, x) which will cancel each other.

Next suppose $k \in V_2$ and $l \notin V_2$. Here we consider two cases: $i, j \notin V_1$ and $i \in V_1, j \notin V_1$. In the first case for every non-zero term with index $(a,b)(x,y)$, both x and y will be different from k because k is the image of i so neither x nor y can be image of i. Hence contributions from (x, y) and (y, x) will cancel each other. If $i \in V_1$ and a and b are different from i, then the situation will be same as above. On the other hand if $a = i$, then we will get a total sum as $\sum_{y \in N^{\rightarrow}(k)} w_1(a,b) w_2(x,y)$ if $a < b$. In case $a > b$, then the total will be $\sum_{y \in N^{\rightarrow}(k)} w_1(a,b) w_2(y,x)$. In both cases the sum is zero.

Next consider the case when $i \in V_1, j \notin V_1, k \in V_2, l \in V_2$. If a, b are both different from i, then contribution of (x, y) and of (y, x) will cancel each other. Now, let $a = i$. In this case also we consider two sub-cases: $(kl) \notin E_2$ and $(k, l) \in E_2$. In the first sub case the contribution will be $(n-3)! \sum_{y \in N^{\rightarrow}(k)} w_1(i,b) w_2(k,y)$ or $(n-3)! \sum_{y \in N^{\rightarrow}(k)} w_1(i,b) w_2(y,k)$ depending on $i < b$ or $i > b$. In both cases the sum is zero. In the second sub case case, assume $i < b$. Then the sum of the contributions is $\sum_{y \in N^{\rightarrow}(k) \setminus \{l\}} w_1(i,b) w_2(k,y)$. This simplifies to $-w_1(i,b) w_2(k,l)$. If $i > b$, then we get $-w_1(i,b) w_2(l,k) = w_1(i,b) w_2(k,l)$. In this case if we add up the contributions for all $b \in N(i)$, then we get $(n-3)!(\sum_{b \in N(i): b<i} w_1(i,b) w_2(k,l) - \sum_{b \in N(i): b>i} w_1(i,b) w_2(k,l))$. This reduces to zero because it is given that $\sum_{b \in N(i): b<i} w_1(i,b) - \sum_{b \in N(i): b>i} w_1(i,b) = 0$.

Finally let $i \in V_1, j \in V_1, k \in V_2, l \notin V_2$. If $a \neq i$, then contribution of (x, y) will cancel that of (y, x). If $a = i$, then the contribution will be $(n-3)! \sum_{y \in N^{\rightarrow}(k)} w_1(i,b) w_2(k,y)$ or $(n-3)! \sum_{y \in N^{\rightarrow}(k)} w_1(i,b) w_2(y,k)$ depending on $i < b$ or $i > b$. As observed earlier, in both cases the sum is zero. □

Theorem A.2.2. *For graphs G_1 and G_2 given in figure A.1 where $\alpha < \beta$ and $\gamma < \delta$, $v(G_1, G_2)$ is an eigenvector with eigenvalue $n(n-3)!$.*

Proof. We have seen from Lemma A.2.1 that in the inner product $B_{(ij)(kl)} \cdot v$ the only terms, $(ab)(xy)$, that have non-zero value are those in which $a, b \in V_1$ and $x, y \in V_2$. Without loss of generality assume that $i < j$. Denote the third vertex in V_1 apart from i, j by p and the third vertex in V_2 apart from k, l by q.

The non-zero contribution is possible for three terms: (i) $(ab)(xy) = (ij)(kl)$, (ii) $(ab)(xy) = (ip)(kq)$, (iii) $(ab)(xy) = (jp)(lq)$. The values of these terms, whose sum is the inner product, are as follows:

(i) $(n-2)!w_1(ij)w_2(kl)$;

(ii) if $i < p$, then $(n-3)!w_1(ip)w_2(kq)$, else $(n-3)!w_1(ip)w_2(qk)$;

(iii) if $j < p$, then $(n-3)!w_1(jp)w_2(lq)$, else $(n-3)!w_1(jp)w_2(ql)$.

Next, we consider the three cases one by one.

Case $p < i < j$: Recall that for $V_1 = \{L, M, H\}$ with $L < M < H$, properties of w_1 implies $w_1(LM) - w_1(MH) = 0$, $w_1(LM) + w_1(LH) = 0$, and $w_1(LH) + w_1(MH) = 0$. So the total is $(n-2)!w_1(ij)w_2(kl) + (n-3)!(w_1(ip)w_2(qk) - w_1(jp)w_2(ql))$. The w_1-equations given above simplifies the expression to $(n-2)!w_1(ij)w_2(kl) + (n-3)!(w_1(ij)w_2(kl) - w_1(jp)(-w_2(kl)))$. This expression simplifies to $n(n-3)!w_1(ij)w_2(kl)$.

Case $i < p < j$: Here the total is $(n-2)!w_1(ij)w_2(kl) + (n-3)!(w_1(ip)w_2(kq) + w_1(jp)w_2(ql))$ which is equal to $(n-2)!w_1(ij)w_2(kl) + (n-3)!((-w_1(ij))(-w_2(kl)) + (-w_1(jp))(-w_2(kl)))$. This also simplifies to $n(n-3)!w_1(ij)w_2(kl)$.

Case $i < j < p$: In this case total is $(n-2)!w_1(ij)w_2(kl) + (n-3)!(w_1(ip)w_2(kq) + w_1(jp)w_2(lq))$ which is equal to $(n-2)!w_1(ij)w_2(kl) + (n-3)!((-w_1(ij))(-w_2(kl)) + w_1(jp)w_2(kl))$. This too simplifies to $n(n-3)!w_1(ij)w_2(kl)$.

Hence in each the inner-product is $n(n-3)!v[(ij)(kl)]$. So $B \cdot v = n(n-3)!v$, i.e., v is an eigenvector with eigenvalue $n(n-3)!$. □

Theorem A.2.3. *The eigenspace of B corresponding to eigenvalue $n(n-3)!$ has dimension at least $\binom{n-1}{2}^2$.*

Proof. We can define one vector $v(G_1, G_2)$ for each choice of (α, β) and each choice of (γ, δ) from $2, \ldots, n$. Let v and v' be such a vector due to $(\alpha, \beta, \gamma, \delta)$. Then $v_{(\alpha,\beta),(\gamma,\delta)}$ is non-zero but the same component of all other eigenvectors of this type is zero. Hence these vectors are independent. □

A.2.3 Third Eigenvalue

The eigenvectors are defined in the same way as in the previous section. In this subsection we assume following properties of the weight-functions associated with G_1 and G_2: (i) all weights are 1 or -1; (ii) $\sum_{b \in N(a)} w_1(a,b) = 0$ for all $a \in V_1$; (iii) $w_2(x,y) = w_2(y,x)$ for all $(xy) \in E_2$; and (iv) for all x, $\sum_{y \in N^{\rightarrow}}(x) w_2(x,y) = \sum_{y \in N^{\leftarrow}}(x) w_2(y,x) = 0$.

Lemma A.2.4. *The inner product $B_{(ij)(kl)} \cdot v(G_1, G_2)$ is non-zero only if $(ij) \in E_1$ and $(kl) \in E_2$.*

Proof. **Case** $k \notin V_2$, $l \notin V_2$

Clearly if $a \in \{i,j\}$ or $b \in \{i,j\}$, then $B_{(ij)(kl)}[(ab)(xy)] = 0$. Otherwise $\sum_{(xy) \in E_2} B_{(ij)(kl)}[(ab)(xy)].v[(ab)(xy)] = (n-4)! \sum_{x \in V_2} \sum_{y \in N^{\rightarrow}(x)} v[(ab)(xy)] = 0$. So $B_{(ij)(kl)} \cdot v = 0$.

Case $i \notin V_1$, $j \notin V_1$

As in the above case $B_{(ij)(kl)} \cdot v = 0$.

Case $i \in V_1$, $j \notin V_1$

Assume that $i < j$.

Sub-case $x \neq k$

First let $a \notin N[i]$.

$\sum_{b \in N[a]} B_{(ij)(kl)}[(ab)(xy)].v[(ab)(xy)] = (n-4)! \sum_{b \in N[a]} v[(ab)(xy)] = 0$.

Next $a \in N(i)$.

$\sum_{a \in N(i)} \sum_{b \in N(a)} B_{(ij)(kl)}[(ab)(xy)].v[(ab)(xy)]$

$= (n-4)! \sum_{a \in N(i)} (-v[(ai)(xy)]) = 0$.

Sub-case $x = k$

$\sum_{b \in N(i)} B_{(ij)(kl)}[(ab)(xy)].v[(ab)(ky)] = (n-3)! \sum_{b \in N(i)} v[(ab)(ky)] = 0$.

Similar argument works for $j < i$.

Case $k \in V_2$, $l \notin V_2$ or $k \notin V_2$, $l \in V_2$

This case is similar to the previous case.

At this stage in the proof we have shown that the inner product can be non-zero only if $i \in V_1, j \in V_1, k \in V_2$ and $l \in V_2$. Now we will show that even in these cases the inner product will be zero if $(ij) \notin E_1$ or $(kl) \in E_2$.

For the remainder of the proof $i \in V_1, j \in V_1, k \in V_2$ and $l \in V_2$. For the remaining three cases we assume that $(i,j) \notin E_1$.

Case $a \notin N[i] \cup N[j]$

Let $i < j$.

$\sum_{b \in N(a)} B_{(ij)(kl)}[(ab)(xy)].v[(ab)(xy)] = (n-4)! \sum_{b \in N(a)} v[(ab)(xy)] = 0.$

Case $a \in (N(i) \cup N(j)) \setminus \{i,j\}$

Recall that i and j are not adjacent.

$\sum_{b \in N(a) \setminus \{i\}} B_{(ij)(kl)}[(ab)(xy)].v[(ab)(xy)]$ is equal to $(n-4)!(-v[(ai)(xy)])$ if $a \in N(i) \setminus N(j)$; $(n-4)!(-v[(aj)(xy)])$ if $a \in N(j) \setminus N(i)$; and $(n-4)!(-v[(ai)(xy)] - v[(aj)(xy)])$ if $a \in N(i) \cap N(j)$;

So $\sum_{a \in (N(i) \cup N(j)) \setminus \{i,j\}} \sum_{b \in N(a) \setminus \{i\}} B_{(ij)(kl)}[(ab)(xy)].v[(ab)(xy)]$ is equal to $|N(i) \cap N(j)|(n-4)! \sum_{a \in N(i)} (-v[(ai)(xy)]) + (n-4)! \sum_{a \in N(j)} (-v[(aj)(xy)]) = 0$.

Case $a = i$ or $a = j$

$\sum_{b \in N(i)} B_{(ij)(kl)}[(ab)(xy)] v[(ab)(xy)] = (n-3)! \sum_{b \in N(i)} v[(ib)(xy)] = 0$. Similarly for $a = j$.

From the last three cases we conclude that if (ij) is not an edge in G_1 then $B_{(ij)(kl)} \cdot v(G_1, G_2) = 0$. Similarly we can show that $B_{(ij)(kl)} \cdot v(G_1, G_2) = 0$ if (kl) is not an edge in G_2. $\square$

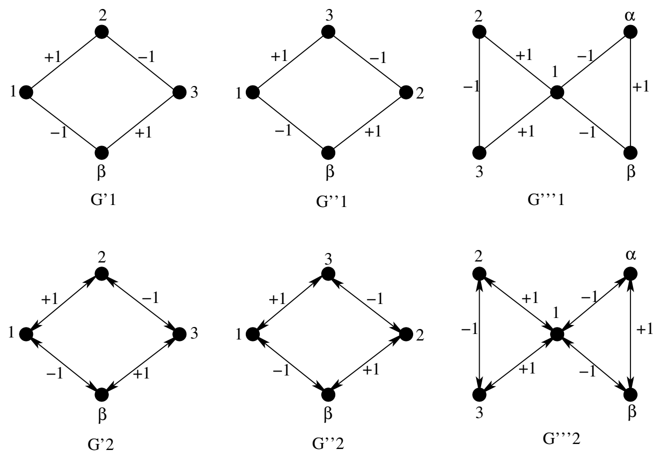

Figure A.2: Index Graphs for the Eigenvectors for Third Eigenvalue

In this case G_1 is any graph among G_1', G_1'', and G_1''' and G_2 is any graph among G_2', G_2'', and G_2'''. These graphs are given in figure A.2.

Theorem A.2.5. *$v(G_1, G_2)$ is an eigenvector of B for each $G_1 \in \{G_1', G_1'', G_1'''\}$ and $G_2 \in \{G_2', G_2'', G_2'''\}$ with eigenvalue $(n-1)!/(n-3)$.*

Proof. From lemma A.2.4 we have seen that $B_{(ij)(kl)} \cdot v(G_1, G_2)$ is zero if $(ij) \notin E_1$ or $(kl) \notin E_2$. For $(ij) \in E_1$ and $(kl) \in E_2$ there are $2 \times 14 \times 14$ cases. Each of these cases is equivalent to one of the following 15 cases: (1) $B_{(12)(12)} \cdot v(G_1', G_2')$, (2) $B_{(12)(23)} \cdot v(G_1', G_2')$, (3) $B_{(12)(12)} \cdot v(G_1', G_2''')$, (4) $B_{(12)(21)} \cdot v(G_1', G_2''')$, (5) $B_{(12)(23)} \cdot v(G_1', G_2''')$, (6) $B_{(12)(12)} \cdot v(G_1''', G_2')$, (7) $B_{(12)(23)} \cdot v(G_1''', G_2')$, (8) $B_{(12)(12)} \cdot v(G_1''', G_2''')$, (9) $B_{(12)(21)} \cdot v(G_1''', G_2''')$, (10) $B_{(12)(23)} \cdot v(G_1''', G_2''')$, (11) $B_{(23)(12)} \cdot v(G_1''', G_2')$, (12) $B_{(23)(23)} \cdot v(G_1''', G_2')$, (13) $B_{(23)(12)} \cdot v(G_1''', G_2''')$, (14) $B_{(23)(21)} \cdot v(G_1''', G_2''')$. (15) $B_{(23)(23)} \cdot v(G_1''', G_2''')$.

The theorem can be established by showing $B_{(ij)(kl)} \cdot v = ((n-1)!/(n-3))v[(ij)(kl)]$ for each of the cases enumerated above. We illustrate the same for cases 1 and 13.

Case (1): $B_{(12)(12)} \cdot v[G_1', G_2']$ is equal to $B_{(12)(12)}[(12)(12)]v[(12)(12)]$ $+B_{(12)(12)}$

$[(1\beta)(1\delta)]v[(1\beta)(1\delta)]$ $+B_{(12)(12)}[(23)(23)]v[(23)(23)]$ $+B_{(12)(12)}[(3\beta)(3\delta)]v[(3\beta)(3\delta)]+$ $B_{(12)(12)}[(3\beta)(\delta 3)]$. This is equal to $(n-2)!+2(n-3)!+2(n-4)! = (n-1)!/(n-3)$. Since $v[(12)(12) = 1$, it is also equal to $((n-1)!/(n-3))v[(12)(12)]$.

Case (10): $B_{(12)(23)} \cdot v(G_1''', G_2''')$ is equal to $B_{(12),(23)}[(12),(23)]v[(12),(23)] +$ $B_{(12),(23)}[(13),(21)]v[(13),(21)] + B_{(12),(23)}[(1\alpha),(21)]v[(1\alpha),(21)]$ $+ B_{(12),(23)}[(1\beta),(21)]v[(1\beta),(21)] + B_{(12),(23)}[(23),(31)]v[(23),(31)]$ $+ B_{(12),(23)}[(\alpha\beta),(1\gamma)]v[(\alpha\beta),(1\gamma)] + B_{(12),(23)}[(\alpha\beta),(\gamma 1)]v[(\alpha\beta),(\gamma 1)]$ $+ B_{(12),(23)}[(\alpha\beta),(1\delta)]v[(\alpha\beta),(1\delta)] + B_{(12),(23)}[(\alpha\beta),(\delta 1)]v[(\alpha\beta),(\delta 1)]$ $+B_{(12),(23)}[(\alpha\beta),(\gamma\delta)]v[(\alpha\beta),(\gamma\delta)]+B_{(12),(23)}[(\alpha\beta),(\delta\gamma)]v[(\alpha\beta),(\delta\gamma)]$ This expression is equal to $-(n-2)! - 2(n-3)! - 2(n-4)!$ which is equal to $-(n-1)!/(n-3)$. Since $v[(12)(23)] = -1$, $B_{(12)(23)} \cdot v(G_1''', G_2''') = ((n-1)!/(n-3))v[(12)(23)]$. □

Theorem A.2.6. *The dimension of the eigenspace of eigenvalue* $(n-1)!/(n-3)$ *is* $(\binom{n-1}{2} - 1)^2$.

Proof. Each vector described above has an entry which is non-zero while the same entry is zero among the other vectors. In $v[G_1'(\beta), G_2'(\delta)]$ such an entry is $(3\beta)(3\delta)$; in $v[G_1'(\beta), G_2''(\delta)]$ such an entry is $(3\beta)(2\delta)$;in $v[G_1'(\beta), G_2'''(\gamma\delta)]$ such an entry is $(3\beta)(\gamma\delta)$ so on. Hence all these vectors are linearly independent. Therefore the dimension of this eigenspace is at least equal to the number of these vectors. The number of instances G_1' are $n-3$, that of G_1'' is $n-3$ and that of G_1''' is $\binom{n-3}{2}$. So the total number of instances of G_1 is $\binom{n-1}{2} - 1$. The number of instances of G_2 is the same. Hence the total number of vectors is $(\binom{n-1}{2} - 1)^2$. □

A.2.4 Fourth Eigenvalue

To define the eigenvectors for this eigenvalue consider three $n \times n$ matrices: A_1, A_2, A_3, given below. The symmetric vectorization $v(E)$ of block-matrix E will be shown to an eigenvector of B with eigenvalue $2n(n-2)!$.

$$A_1 = \begin{bmatrix} 1-(n-1)(n-2) & 0 & \dots & 0 & 0 & 0 \\ 0 & n-1 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & n-1 & 0 \\ 0 & 0 & \dots & 0 & 0 & -1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & -n+2+\frac{1}{n-2} & -n+2+\frac{1}{n-2} & \dots & -n+2+\frac{1}{n-2} & -n+2 \\ \frac{1}{n-2} & 0 & \frac{n-1}{n-2} & \dots & \frac{n-1}{n-2} & 1 \\ \frac{1}{n-2} & \frac{n-1}{n-2} & 0 & \dots & \frac{n-1}{n-2} & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{1}{n-2} & \frac{n-1}{n-2} & \frac{n-1}{n-2} & \dots & 0 & 1 \\ -1 & -1 & -1 & \dots & -1 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & -n+2 & -n+2 & -n+2 & \dots & -n+2 & -n+3 \\ n-2 & 0 & 0 & 0 & \dots & 0 & 1 \\ n-2 & 0 & 0 & 0 & \dots & 0 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ n-2 & 0 & 0 & 0 & \dots & 0 & 1 \\ n-3 & -1 & -1 & -1 & \dots & -1 & 0 \end{bmatrix}$$

$$E = \begin{bmatrix} A_1 & A_3 & A_2 & \dots & A_2 \\ A_3^T & -A_1 & -A_2 & \dots & -A_2 \\ A_2^T & -A_2^T & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ A_2^T & -A_2^T & 0 & \dots & 0 \end{bmatrix}$$

Let $X(a_1 \to a_1', a_2 \to a_2', a_3 \to a_3', a_4 \to a_4')$ denote the number of permutations in which a_i maps to a_i' for $i = 1, 2, 3, 4$. Since matrix A_1 is diagonal, $(B \cdot v)_{ik,jl}$ can be expressed as operations on the $n \times n$ blocks as follows: $(B \cdot v)_{ik,jl} = \sum_{pr,qs} B_{ik,jl}(pr, qs)v(pr, qs) = \sum_{q,s} A_1(q, s).X(i \to j, k \to l, 1 \to q, 1 \to s) -$

$\sum_{q,s} A_1(q,s).X(i \to j, k \to l, 2 \to q, 2 \to s) + \sum_{q,s} A_3(q,s).X(i \to j, k \to l, 1 \to q, 2 \to s) + \sum_{r>2}\sum_{q,s} A_2(q,s).X(i \to j, k \to l, 1 \to q, r \to s) - \sum_{r>2}\sum_{q,s} A_2(q,s).X(i \to j, k \to l, 2 \to q, r \to s)$. Denote this expression by $C(ik)_{jl}$. Now we will evaluate the value of this expression for various values of (ik, jl).

Case: $ik = 11$. In this case we need to show that $C(11) = 2n(n-2)!A_1$.

For $jl = 11$ we have $C(11)_{11} = (n-1)!(1-(n-1)(n-2)) - (-1+(n-1)(n-2))(n-2)! + (-(n-2)^2 - (n-3))(n-2)!(n-2)((n-2)((-n+2+(1/(n-2)))) - (n-2))(n-2)! - (n-2)^2(1+(n-3)(n-1)/(n-2))(n-3)!$. This expression simplifies to $2n(n-2)!(1-(n-1)(n-2)) = 2n(n-2)!A_1(1,1)$.

For $jl = 22$ we have $C(11)_{22} = (n-1)(n-1)! - ((1-(n-1)(n-2)) + (n-1)(n-3)-1)(n-2)! + ((n-2)+1)(n-2)! + (n-2)(1/(n-2) + (n-3)(n-1)/(n-2)+1)(n-2)! - (n-2)((n-3)(-n+2-1/(n-2)) - (n-2) + (1/(n-2) + (n-4)(n-1)/(n-2) + 1)(n-3)-1)(n-3)!$. It simplifies to $2n! = (n-1).2n(n-2)! = 2n(n-2)!.A_1(2,2)$.

For $jl = tt$ where $2 \le t \le n-1$ we get the similar expression, i.e., $2n(n-2)!.A_1(t,t)$.

In case $jl = nn$, $C(11)_{nn} = -(n-1)! - (1-(n-1)(n-2) + (n-1)(n-2))(n-2)! + ((n-3)-(n-2))(n-2)! - (n-2)(n-2)! - (n-2)((-n+2+(1/(n-2))(n-2) + ((1/(n-2)) + ((n-3)(n-1)/(n-2))(n-2))(n-3)! = -2n(n-2)! = 2n(n-2)!A_1(n,n)$.

In every case where $j \neq l$, $X(1 \to j, 1 \to l, *, *) = 0$. So $C(11)_{jl} = 0$ for all $j \neq l$. Thus $C(11) = 2n(n-2)!A_1$.

Case: $ik = 22$. In this case we have to show that $C(22) = 2n(n-2)!(-A_1)$.

We observe following equalities. $A_1(q,s).X(2 \to j, 2 \to l, 2 \to q, 2 \to s) = A_1(q,s).X(1 \to j, 1 \to l, 1 \to q, 1 \to s)$, $A_1(q,s).X(2 \to j, 2 \to l, 1 \to q, 1 \to s) = A_1(q,s).X(1 \to j, 1 \to l, 2 \to q, 2 \to s)$, $A_2(q,s).X(2 \to j, 2 \to l, 1 \to q, r \to s) = A_2(q,s).X(1 \to j, 1 \to l, 2 \to q, r \to s)$, $A_2(q,s).X(2 \to j, 2 \to l, 2 \to q, r \to s) = A_2(q,s).X(1 \to j, 1 \to l, 1 \to q, r \to s)$. Further, $A_3(q,s).X(2 \to j, 2 \to l, 1 \to q, 2 \to s) = A_3(q,s).X(1 \to j, 1 \to l, 2 \to q, 1 \to s) = A_3(s,q).X(1 \to j, 1 \to l, 1 \to q, 2 \to s) = -A_3(q,s).X(1 \to j, 1 \to l, 1 \to q, 2 \to s)$.

Substituting these values in the expression of $C(22)$ we get $C(22) = -C(11) = 2n(n-2)!(-A_1)$.

Case $ik = 12$. In this case we have to show that $C(12) = 2n(n-2)!A_3$.

For $jl = 12$, $C(12)_{12} = (1-(n-1)(n-2))(n-2)! - (n-1)(n-2)! + ((n-3)(-n+2+(1/(n-2))) - (n-2))(n-2)(n-3)! - (1+(n-1)(n-3)/(n-2))(n-2)(n-3)! = -2(n-2)n(n-2)! = 2n(n-2)!A_3(1,2)$.

For $jl = 21$, $C(12)_{21} = (n-1)(n-2)! - (1-(n-1)(n-2)) + (n-2)(n-2)! + (n-2)(1+(n-3)(n-1)/(n-2))(n-3)! - (n-2)((n-3)(-n+2+1/(n-2)) - (n-2))(n-3)! = (n-2)!(2n^2-4n) = 2n(n-2)!A_3(2,1)$.

Next for $jl = 1n$, $C(12)_{1n} = (1-(n-1)(n-2))(n-2)! + (n-2)! - (n-3)(n-2)! - (n-2)^2(n-2-(1/(n-2)))(n-3)! = -2n(n-3)(n-2)! = 2n(n-2)!A_3(1,n)$.

Expression for $jl = n1$, $C(12)_{n1} = -1.(n-2)! - (1-(n-1)-(n-2))(n-2)! + (n-3)(n-2)! + 0 + (n-2)^2(n-2-1/(n-2))(n-3)! = 2n(n-3)(n-2)! = 2n(n-2)!A_3(n,1)$.

Expression for $jl = 2n$, $C(12)_{2n} = (n-1)(n-2)! + (n-2)! + (n-2)! + ((n-1)(n-3)/(n-2) + 1/(n-2))(n-2)(n-3)! + (n-2)(n-3)! = 2n(n-2)! = 2n(n-2)!A_3(2,n)$. Similarly it can be seen that $C(12)_{rn} = 2n(n-2)!A_3(r,n)$ for every $2 < r < n-1$.

Expression for $jl = n2$, $C(12)_{n2} = -(n-2)! - (n-1)(n-2)! - (n-2)! + (-1)(n-2)(n-3)! - (n-2)(1/(n-2) + (n-3)(n-1)/(n-2))(n-3)! = -2n(n-2)! = (-1)A_3(n,2)$. Similarly it can be shown that $C(12)_{n,r} = (-1)A_3(n,r)$ for $1 < r < n-1$.

For $j \in \{2,3,\ldots,n-1\}$ and $l \in \{2,3,\ldots,n-1\}$, $C(12)_{jl} = (n-1)(n-2)! - (n-1)(n-2)! + 0 + (1 + 1/(n-2) + (n-1)(n-4)/(n-2))(n-2)(n-3)! - (1 + 1/(n-2) + (n-1)(n-4)/(n-2))(n-2)(n-3)! = 0$.

Putting these together we see that $C(12) = 2n(n-2)!A_3$.

Case $ik = 1k$ with $k \geq 3$: Next we will show that $C(1k) = 2n(n-2)!A_2$, for $k \geq 3$. We will show the details of the computation of $C(1,3)$.

For $jl = 12$, $C(13)_{12} = (1-(n-1)(n-2))(n-2)! - ((n-3)(n-1)-1)(n-3)! - ((n-3)(n-2)+(n-3))(n-3)! - (n-2-1/(n-2))(n-2)! + (n-3)(-(n-2-1/(n-2))(n-3)-(n-2))(n-3)! - ((n-3)(n-1)/(n-2))(n-3)! - (n-3)((n-3)(n-4)(n-1)/(n-2)+(n-3))(n-4)!$. It simplifies to $-(n-2)!(2n^2-7n+5) - (n-3)!(3n^2-13n+10)$ which is equal to $-(n-2-1/(n-2))2n(n-2)! = 2n(n-2)!A_2(1,2)$. Similarly we can show $C(13)_{1l} = 2n(n-2)!A_2(1,l)$, for all $2 \leq l \leq n-1$.

For $jl = 21$, $C(13)_{21} = (n-1)(n-2)! - ((n-3)(n-1)-1)(n-3)! + (n-3)! + (1/(n-2))(n-2)! + (n-3)(1+(n-3)(n-1)/(n-2))(n-3)! - ((n-3)/(n-2)-1)(n-3)! - (n-3)(n-3+(n-3)(n-4)(n-1)/(n-2))(n-4)! = (1/(n-2)).2n(n-2)! = 2n(n-2)!A_2(2,1)$. Similarly we can show $C(13)_{j1} = 2n(n-2)!A_2(j,1)$, for all $2 \leq j \leq n-1$.

For $jl = 1n$, $C(13)_{1n} = (n-2)! - (n-2)(n-1)! - (n-1)! - 2(n-2)(n-2)! - (n-2)(n-3)(n-2)! + (n-3)(n-3)! - (n-2)(n-3)! - (n-1)(n-3)(n-3)! = -2(n-2)n(n-2)! = 2n(n-2)!A_2(1,n)$.

For $jl = n1$, $C(13)_{n1} = -(n-2)! - (n-2)(n-1)(n-3)! - (n-2)(n-3)! - (n-2)! + 0 - 1.(n-3)! - (n-3)(n-2)((n-3)((n-1)/(n-2))(n-4)! = -2n(n-2)! = 2n(n-2)!A_2(n,1)$.

For $jl = 2n$, $C(13)_{2n} = (n-1)(n-2)! - (1-(n-1)(n-2)+(n-3)(n-1))(n-3)! + (n-2)(n-3)! + (n-2)! + (n-3)(1/(n-2)+(n-3)(n-1)/(n-2))(n-3)! - ((-1)(n-3)! - (n-3)((n-3)/(n-2) - (n-3)(n-2-1/(n-2)) + (n-3)(n-4)(n-1)/(n-2))(n-3)! = 1.2n(n-2)! = A_2(2n).2n(n-2)!$. Similarly we can show $C(13)_{jn} = 2n(n-2)!A_2(j,n)$, for all $2 \leq j \leq n-1$.

For $jl = n2$, $C(13)_{n2} = -(n-2)! - (-1-(n-1)(n-2)+(n-1(n-3)))(n-3)! + (n-3-(n-3))(n-3)! + 0 - (n-3)(n-3)! - (-(n-2)+1/(n-2)+(n-3)(n-1)/(n-2))(n-3)! = 0 = 2n(n-2)!.A_2(n,2)$. Similarly it can be shown that $C(13)_{nl} = 0 = 2n(n-2)!.A_2(n,l)$ for all $2 \leq l \leq n-1$.

For $jl = 23$, $C(13)_{23} = (n-1)(n-2)! - (1-(n-1)(n-2)-1+(n-1)(n-4))(n-3)! - (n-2+1)(n-3)! + ((n-1)/(n-2))(n-2)! + (n-3)((n-3)(n-1)/(n-2))(n-3)! - ((n-4)(n-1)/(n-2) - (n-2) + 1/(n-2))(n-3)! - ((n-4)/(n-2) - 1 + (n-4) - (n-2) - (n-4)(n-2-1/(n-2)) + (n-4)(n-5)(n-1)/(n-2))(n-3)! = 2n(n-1)(n-2)!/(n-2) = A_2(2,3).2n(n-2)!$. Similarly we can show that $C(13)_{jl} = 2n(n-2)!A_2(j,l)$, for all $2 \leq j, l \leq n-1$ and $j \neq l$.

For $jl = n2$, $C(13)_{n2} = (-1)(n-2)! - ((n-1)(n-3)-(n-1)(n-2)+1)(n-3)! + 0 + 0 - (n-3).(n-3)! - ((n-3)(n-1)/(n-2) - (n-2) + 1/(n-2))(n-3)! - (-(n-3)(n-2-1/(n-2)) + (n-3)/(n-2) + (n-3)(n-4)(n-1)/(n-2))(n-3)! = 0$. Similarly we can show $C(13)_{nl} = 2n(n-2)!A_2(n,l)$, for all $2 \leq l \leq n-1$.

For $jl = jj$, trivially $C(13)_{jj} = 0$. Hence $C(13)_{jj} = 2n(n-2)!.A_2(j,j)$.

These observations conclude that $C(13) = 2n(n-2)!A_2$. Similarly it can be seen that $C(1,k) = 2n(n-2)!A_2$ for all $k \geq 3$.

Case $ik = 2k$ for $k \geq 3$: Now we will show that $C(2,k) = 2n(n-2)!(-A_2)$ for all $k \geq 3$. We will show the details of the computation of $C(2,3)$.

$C(2,3)_{jl} = \sum_{q,s} A_1(q,s).X(2 \to j, 3 \to l, 1 \to q, 1 \to s) - \sum_{q,s} A_1(q,s).X(2 \to j, 3 \to l, 2 \to q, 2 \to s) + \sum_{q,s} A_3(q,s).X(2 \to j, 3 \to l, 1 \to q, 2 \to s) + \sum_{r>2} \sum_{q,s} A_2(q,s).X(2 \to j, 3 \to l, 1 \to q, r \to s) - \sum_{r>2} \sum_{q,s} A_2(q,s).X(2 \to j, 3 \to l, 2 \to q, r \to s)$.

We will show that the right hand side is equal to $-C(2,3)_{jl}$. We state the following facts.

$A_1(q,s).X(2 \to j, 3 \to l, 1 \to q, 1 \to s) = A_1(q,s).X(1 \to j, 3 \to l, 2 \to q, 2 \to$

s). $A_1(q,s).X(2 \to j, 3 \to l, 2 \to q, 2 \to s) = A_1(q,s).X(1 \to j, 3 \to l, 1 \to q, 1 \to s)a$. $A_3(q,s).X(2 \to j, 3 \to l, 1 \to q, 2 \to s)A_3(q,s).X(1 \to j, 3 \to l, 2 \to q, 1 \to s) = A_3(s,q).X(1 \to j, 3 \to l, 1 \to q, 2 \to s) = -A_3(q,s).X(1 \to j, 3 \to l, 1 \to q, 2 \to s)$. The last equation is due to the fact that A_3 is anti-symmetric. $A_2(q,s).X(2 \to j, 3 \to l, 1 \to q, r \to s) = A_2(q,s).X(1 \to j, 3 \to l, 2 \to q, r \to s)$ for all $r > 2$. $A_2(q,s).X(2 \to j, 3 \to l, 2 \to q, r \to s) = A_2(q,s).X(1 \to j, 3 \to l, 1 \to q, r \to s)$ for all $r > 2$.

Plugging the right hand side expressions into the expression of $C(2,3)_{jl}$ we get $C(2,3)_{jl} = -C(1,3)_{jl}$. Hence $C(2,3) = -A_2.2.n.(n-2)!$. Similarly we can show that $C(2,k) = -A_2.2n(n-2)!$ for all $k \geq 3$.

Case $i > 2, k > 2$: Finally we have to show that $C(ik) = 0$ for $2 < i$ and $2 < k$.

If i and k are both greater than 2, we have $A_1(q,s).X(i \to j, k \to l, 1 \to q, 1 \to s) = A_1(q,s).X(i \to j, k \to 2, 1 \to q, 2 \to s)$ and $A_2(q,s).X(i \to j, k \to l, 1 \to q, r \to s) = A_2(q,s).X(i \to j, k \to l, 2 \to q, r \to s)$. Further, A_3 is anti-symmetric so $\sum_{q,s} A_3(q,s).X(i \to j, k \to l, 1 \to q, 2 \to s) = \sum_{q<s}(A_3(q,s) + A_3(s,q)).X(i \to j, k \to l, 1 \to q, 2 \to s) = 0$. Thus $C(i,k) = 0 = 0.2n(n-2)!$ for all $i > 2, k > 2$.

All these results combine to show that $v(E)$ is an eigenvector of B with eigenvalue $2n(n-2)!$.

To see that there are several linearly independent vectors which are eigen vectors for the same eigenvalue consider the following modifications in E. Begin with the observation that in blocks A_1, A_2, and A_3 indices 1 and n have special role, while all other indices are equivalent. That is, for each $k \in \{1,2,3\}$, $A_k(q,s) = A_k(q',s')$ where $q' = q$ if $q \in \{1,n\}$ otherwise q' is any member of $\{2,\ldots,n-1\}$ and $s' = s$ if $s \in \{1,n\}$ otherwise s' is any member of $\{2,\ldots,n-1\}$. So defining A'_1, A'_2 and A'_3 by exchanging the role of n by any other index α in $\{2,\ldots,n\}$ we again get an eigenvector with the same eigenvalue. These $n-1$ eigenvectors are linearly independent because in each vector the entry $B_3(2,n')$ is non-zero in exactly one vector, for each $n' \in \{2,\ldots,n\}$.

There are additional $n-2$ similar sets of eigenvectors. Exchange the role of 2 in E by any index β in the range $2,3,\ldots,n-1$, see matrix E'. Once again we get a set of $n-1$ eigenvectors. Observe that among the diagonal blocks 22 to nn exactly one block is non-zero in each set. Hence these $(n-1)^2$ vectors are linearly independent.

$$E' = \begin{bmatrix} A_1 & A_2 & \dots & A_2 & A_3 & A_2 & \dots & A_2 \\ A_2^T & 0 & \dots & 0 & -A_2^T & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ A_2^T & 0 & \dots & 0 & -A_2^T & 0 & \dots & 0 \\ A_3^T & -A_2 & \dots & -A_2 & -A_1 & -A_2 & \dots & -A_2 \\ A_2^T & 0 & \dots & 0 & -A_2^T & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ A_2^T & 0 & \dots & 0 & -A_2^T & 0 & \dots & 0 \end{bmatrix}$$

Theorem A.2.7. *The dimension of the eigenspace of B corresponding to eigenvalue $2n(n-2)!$ is at least $(n-1)^2$.*

References

[ABB+73] A. P. Ambler, H. G. Barrow, C. M. Brown, R. H. Burstall, and R. J. Popplestone. A versatile computer-controlled assembly system. In *Proceedings of the 3rd international joint conference on Artificial intelligence*, pages 298–307, 1973. 1

[AHU74] Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, 1974. 1, 5

[AM12] Albert Atserias and Elitza N. Maneva. Sherali-adams relaxations and indistinguishability in counting logics. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 367–379, 2012. 9, 10

[AT05] Vikraman Arvind and Jacobo Torán. Isomorphism testing: Perspective and open problems. *Bulletin of the EATCS*, 86:66–84, 2005. 1

[Bab79] László Babai. Monte-carlo algorithms in graph isomorphism testing. 1979. 6

[Bab80] László Babai. On the complexity of canonical labeling of strongly regular graphs. *SIAM J. Comput.*, 9(1):212–216, 1980. 6

[Bab81] László Babai. Moderately exponential bound for graph isomorphism. In *Fundamentals of Computation Theory, FCT'81, Proceedings of the 1981 International FCT-Conference, Szeged, Hungary, August 24-28, 1981*, pages 34–50, 1981. 1, 7

[Bab14] László Babai. On the automorphism groups of strongly regular graphs I. In *Innovations in Theoretical Computer Science, ITCS'14, Princeton, NJ, USA, January 12-14, 2014*, pages 359–368, 2014. 76

[BCS+13] László Babai, Xi Chen, Xiaorui Sun, Shang-Hua Teng, and John Wilmes. Faster canonical forms for strongly regular graphs. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 157–166, 2013. 6, 76

[BGM82] László Babai, D. Yu. Grigoryev, and David M. Mount. Isomorphism of graphs with bounded eigenvalue multiplicity. In *STOC*, pages 310–324, 1982. 2, 7

[BHZ87] Ravi B. Boppana, Johan Håstad, and Stathis Zachos. Does co-np have short interactive proofs? *Inf. Process. Lett.*, 25(2):127–132, 1987. 1

[BK79] László Babai and Ludek Kucera. Canonical labelling of graphs in linear average time. In *FOCS*, pages 39–46, 1979. 3

[BL83] László Babai and Eugene M. Luks. Canonical labeling of graphs. In *STOC*, pages 171–183, 1983. 1, 7

[Bod90] Hans L. Bodlaender. Polynomial algorithms for graph isomorphism and chromatic index on partial k-trees. *J. Algorithms*, 11(4):631–643, 1990. 2

[Bol82] Béla Bollobás. Distinguishing vertices of random graphs. *Annals of Discrete Mathematics*, 13:33–50, 1982. 4

[CB81] Charles J. Colbourn and Kellogg S. Booth. Linear time automorphism algorithms for trees, interval graphs, and planar graphs. *SIAM J. Comput.*, 10(1):203–225, 1981. 1

[CFI92] Jin-yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identifications. *Combinatorica*, 12(4):389–410, 1992. 4, 73

[CM78] D.G. Corneil and R.A. Mathon. Algorithmic techniques for the generation and analysis of strongly regular graphs and other combinatorial configurations. *Annals of Discrete Mathematics*, 2:1–32, 1978. 1

[dKP02] E. de Klerk and D. V. Pasechnik. Approximation of the stability number of a graph via copositive programming. *SIAM Journal on Optimization*, 12(4):875–892, 2002. 60, 62

[FHL80] Merrick L. Furst, John E. Hopcroft, and Eugene M. Luks. Polynomial-time algorithms for permutation groups. In *21st Annual Symposium on Foundations of Computer Science, Syracuse, New York, USA, 13-15 October 1980*, pages 36–41, 1980. 6

[FM80] I. S. Filotti and Jack N. Mayer. A polynomial-time algorithm for determining the isomorphism of graphs of fixed genus (working paper). In *STOC*, pages 236–243, 1980. 2, 5

[FMR79] I. S. Filotti, Gary L. Miller, and John H. Reif. On determining the genus of a graph in o(vˆo(g)) steps. In *Proceedings of the 11h Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1979, Atlanta, Georgia, USA*, pages 27–37, 1979. 5

[GJ79] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979. 1

[GM12] Martin Grohe and Dániel Marx. Structure theorem and isomorphism test for graphs with excluded topological subgraphs. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 173–192, 2012. 2

[GO12] Martin Grohe and Martin Otto. Pebble games and linear equations. In *Computer Science Logic (CSL'12) - 26th International Workshop/21st Annual Conference of the EACSL, CSL 2012, September 3-6, 2012, Fontainebleau, France*, pages 289–304, 2012. 10

[Gro98] Martin Grohe. Fixed-point logics on planar graphs. In *Thirteenth Annual IEEE Symposium on Logic in Computer Science, Indianapolis, Indiana, USA, June 21-24, 1998*, pages 6–15, 1998. 4

[Gro11] Martin Grohe. From polynomial time queries to graph structure theory. *Commun. ACM*, 54(6):104–112, 2011. 4

[HL83] John Hannah and Thomas J. Laffey. Nonnegative factorization of completely positive matrices. *Linear Algebra and its Applications*, 55(0):1 – 9, 1983. 59

[HT72] John E. Hopcroft and Robert Endre Tarjan. Isomorphism of planar graphs. In *Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York.*, pages 131–152, 1972. 1, 4

[HW74] John E. Hopcroft and J. K. Wong. Linear time algorithm for isomorphism of planar graphs (preliminary report). In *STOC*, pages 172–184, 1974. 1, 5

[IL90] Neil Immerman and Eric Lander. Describing graphs: a first-order approach to graph canonization. *Complexity theory retrospective*, pages 59–81, 1990. 3

[JK97] Michael Jünger and Volker Kaibel. Box-inequalities for quadratic assignment polytopes. In *Mathematical Programming*, pages 175–197, 1997. 23, 70

[Jr.65] E. H. Sussenguth Jr. A graph-theoretic algorithm for matching chemical structures. *J. Chem. Doc.*, 5:36–43, 1965. 1

[Kai97] V. Kaibel. *Polyhedral Combinatorics of the Quadratic Assignment Problem.* PhD thesis, Faculty of Mathematics and Natural Sciences, University of Cologne, 1997. 11, 15, 23, 24, 25, 70

[Kar72] Richard M. Karp. Reducibility among combinatorial problems. In *Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York.*, pages 85–103, 1972. 1

[KKLV10] Johannes Köbler, Sebastian Kuhnert, Bastian Laubner, and Oleg Verbitsky. Interval graphs: Canonical representation in logspace. In *Automata, Languages and Programming, 37th International Colloquium, ICALP 2010, Bordeaux, France, July 6-10, 2010, Proceedings, Part I*, pages 384–395, 2010. 2

[Kle96] Philip N. Klein. Efficient parallel algorithms for chordal graphs. *SIAM J. Comput.*, 25(4):797–827, 1996. 2

[Koz78] Dexter Kozen. A clique problem equivalent to graph isomorphism. *SIGACT News*, 10(2):50–52, July 1978. 60

[Kuc87] Ludek Kucera. Canonical labeling of regular graphs in linear average time. In *28th Annual Symposium on Foundations of Computer Science, Los Angeles, California, USA, 27-29 October 1987*, pages 271–279, 1987. 4

[Las01] Jean B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817, 2001. 10

[Lau10] Bastian Laubner. Capturing polynomial time on interval graphs. In *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*, pages 199–208, 2010. 4

[LB79] George S. Lueker and Kellogg S. Booth. A linear time algorithm for deciding interval graph isomorphism. *J. ACM*, 26(2):183–195, April 1979. 2, 5

[Luk82] Eugene M. Luks. Isomorphism of graphs of bounded valence can be tested in polynomial time. *J. Comput. Syst. Sci.*, 25(1):42–65, 1982. 2, 7

[Mal14] Peter N. Malkin. Sherali-adams relaxations of graph isomorphism polytopes. *Discrete Optimization*, 12:73–97, 2014. 3, 4, 9, 10, 17, 74

[Mat79] Rudolf Mathon. A note on the graph isomorphism counting problem. *Inf. Process. Lett.*, 8(3):131–132, 1979. 1, 6

[Mil80] Gary L. Miller. Isomorphism testing for graphs of bounded genus. In *STOC*, pages 225–235, 1980. 2, 5

[Mil83] Gary L. Miller. Isomorphism of k-contractible graphs. A generalization of bounded valence and bounded genus. *Information and Control*, 56(1/2):1–20, 1983. 7

[MP14] Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, II. *J. Symb. Comput.*, 60:94–112, 2014. 2

[MPRW09] Jérôme Malick, Janez Povh, Franz Rendl, and Angelika Wiegele. Regularization methods for semidefinite programming. *SIAM Journal on Optimization*, 20(1):336–356, 2009. 80

[MPRW11] Jérôme Malick, Janez Povh, Franz Rendl, and Angelika Wiegele. Sdp solver mprw2.m. `https://www.math.aau.at/or/Software/`, 2011. 80

[OWWZ14] Ryan O'Donnell, John Wright, Chenggang Wu, and Yuan Zhou. Hardness of robust graph isomorphism, lasserre gaps, and asymmetry of random graphs. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1659–1677, 2014. 10

[Pon88] Ilia Nikolaevich Ponomarenko. Isomorphism problem for classes of graphs closed under contractions. *Zapiski Nauchnykh Seminarov POMI*, 174:147–177, 1988. 2, 7

[PR09] Janez Povh and Franz Rendl. Copositive and semidefinite relaxations of the quadratic assignment problem. *Discrete Optimization*, 6(3):231–241, 2009. 14, 23, 63

[SA90] Hanif D. Sherali and Warren P. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM J. Discrete Math.*, 3(3):411–430, 1990. 8, 9, 17

[Sch88] Uwe Schöning. Graph isomorphism is in the low hierarchy. *J. Comput. Syst. Sci.*, 37(3):312–323, 1988. 1

[Spe] E Spence. Strongly regular graphs. http://www.maths.gla.ac.uk/~es/srgraphs.php. Accessed: 17/06/2015. 78, 80, 83

[Spi96] Daniel A. Spielman. Faster isomorphism testing of strongly regular graphs. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 576–584, 1996. 6, 76

[SSC14] Aaron Snook, Grant Schoenebeck, and Paolo Codenotti. Graph isomorphism and the lasserre hierarchy. *CoRR*, abs/1401.0758, 2014. 10

[Tin91] Gottfried Tinhofer. A note on compact graphs. *Discrete Applied Mathematics*, 30(2-3):253–264, 1991. 9

[Ueh08] Ryuhei Uehara. Simple geometrical intersection graphs. In *WALCOM: Algorithms and Computation, Second International Workshop, WALCOM 2008, Dhaka, Bangladesh, February 7-8, 2008.*, pages 25–33, 2008. 2

[WL68] B. Weisfeiler and A. A. Lehman. A reduction of a graph to a canonical form and an algebra arising during this reduction (in russian). *Nauchno-Technicheskaya Informatsia, Seriya*, 2(9):12–16, 1968. 3

[ZKT82] VN Zeml'ahenko, NM Korneyenko, and Regina Iosifovna Tyshkevich. Graph isomorphism problem. *Zapiski Nauchnykh Seminarov POMI*, 118:83–158, 1982. 1, 7