Technical University of Munich

Department of Mathematics

Master Thesis

The Stable Set Problem and Graph Decompositions

Sophia Keip

Examiner: Prof. Dr. Stefan Weltge

Advisor: Prof. Dr. Stefan Weltge

Submission date: 27.08.2019

I hereby declare that this thesis is my own work and that no other sources have been used except those clearly indicated and referenced.

Garching, 27.08.2019

Zusammenfassung

Diese Arbeit beschäftigt sich mit Techniken, das gewichtete Stabile-Mengen-Problem auf Graphen G zu lösen, die an einer Trennmenge von Knoten in zwei Teile zerlegt werden können. Alle vorgestellten Methoden basieren auf einem polyedrischen Ansatz. Hierbei wird das Problem durch die Optimierung einer linearen Funktion auf dem Stabile-Mengen-Polytop $P(G)$ beschrieben. Das Stabile-Mengen-Polytop ist die konvexe Hülle aller charakteristischen Vektoren, die einer stabilen Menge in G zugeordnet werden können. Um die Methoden der linearen Programmierung auf dieses Optimierungsproblem anzuwenden, ist man an einer geeigneten Beschreibung von $P(G)$ interessiert. $P(G)$ kann entweder durch eine Menge an Ungleichungen im entsprechenden Raum oder durch eine erweiterte Formulierung beschrieben werden. Eine erweiterte Formulierung ist die Ungleichungsbeschreibung eines Polytopes, welches auf $P(G)$ projiziert werden kann. Bei beiden Arten $P(G)$ zu beschreiben ist die Anzahl an Ungleichungen entscheidend. Um eine Ungleichungsbeschreibung oder eine erweiterte Formulierung von $P(G)$ zu erhalten, nutzen wir die Tatsache dass G zerlegbar ist. Das Ziel ist es, eine Beschreibung von $P(G)$ auf Basis der Beschreibungen der Stabile-Mengen-Polytope der beiden Teile von G zu erhalten. Solche Techniken ermöglichen es, das Stabile-Mengen-Problem auf einem Graphen zu lösen, der rekursiv in einfachere Graphen zerlegt werden kann, bei denen eine kleine Beschreibung des Stabile-Mengen-Problems bekannt ist. Klein bedeutet hier polynomiell relativ zur Eingabegröße. Entscheidend für unser Vorhaben ist die Struktur der Trennmenge, an der die beiden Komponenten verbunden sind.

Zunächst werden wir uns den Fall ansehen, in dem die Trennmenge eine Clique bildet. $P(G)$ kann in diesem Fall einfach durch die Vereinigung der Ungleichungsbeschreibungen der Stabile-Mengen-Polytope der Komponenten beschrieben werden. Wir werden zeigen, dass dieser Ansatz nicht funktioniert, wenn die Trennmenge keine Clique bildet. In diesem Fall bleibt das Stabile-Mengen-Problem $\mathcal{NP}$-schwer, auch wenn die einzelnen Komponenten sehr einfach sind. Wir werden dies durch die Reduktion eines $\mathcal{NP}$-schweren Problems zeigen.

In dieser Arbeit werden zwei Methoden für beliebige Arten von Trennmengen beleuchtet. Die Idee ist es, Modifizierungen von G zu betrachten. Ziel dieser Modifizierungen ist die Konstruktion einer geeigneten Trennmenge. Haben wir eine Ungleichungsbeschreibung für das Stabile-Mengen-Polytop des modifizierten Graphen, ist es sehr einfach eine erweiterte Formulierung von $P(G)$ zu erhalten.

Neben der Beschreibung der einzelnen Ansätze werden wir uns den Einfluss der Ergebnisse auf die Anzahl an Ungleichungen ansehen, die benötigt werden um $P(G)$ zu beschreiben.

Summary

This thesis studies techniques to solve the weighted stable set problem on graphs G, which can be decomposed into two components by a node cut set. The considered methods are based on a polyhedral approach. Here, the problem is expressed by the optimization of a linear function over the stable set polytope $P(G)$, which is the convex hull of all characteristic vectors of stable sets in G. In order to apply tools of linear programming to this optimization, one is interested in a suitable description of $P(G)$. $P(G)$ can either be described by a set of linear inequalities in the ambient space or by an extended formulation, which is an inequality description of a polytope that can be projected onto $P(G)$. In both cases, the number of describing inequalities is crucial. In order to get an inequality description or an extended formulation of the stable set polytope, we exploit the fact that the considered graphs are decomposable. Our goal is to find techniques to obtain a description of $P(G)$ that is based on the descriptions of the stable set polytopes of the two components. Such techniques provide a way to solve the stable set problem on a graph which is recursively decomposable into simpler graphs that have a known description of their stable set polytopes of small size. Small means in this context polynomial in the input size. The structure of the cut set which connects the single components is crucial for our intention.

We first consider the case where the cut set forms a clique. Here, the inequality description of $P(G)$ is simply the union of the inequality descriptions of the stable set polytopes associated with the components. We show that this does not work when the cut set is not a clique. In this case, the stable set problem is still $\mathcal{NP}$-hard even if the single components are very simple. This is shown by the reduction of an $\mathcal{NP}$-hard problem.

This work studies two methods for arbitrary kinds of cut sets. The idea is to look at modifications of G. The goal of these modifications is the construction of a suitable cut set. Once we have an inequality description of the stable set polytope of the modified graph, it is easy to obtain an extended formulation of $P(G)$.

Beyond just describing the various techniques, the impact of the results on the number of inequalities that are needed to describe $P(G)$ is studied.

Contents

1 Introduction

Let $G = (V, E)$ be an undirected graph. A set $S \subseteq V$ is called **stable** if it does not contain a pair of connected nodes. The **stable set problem** looks for the maximum of

$$\sum_{v \in S} c_v$$

over all stable sets $S \subseteq V$ of G. Here, c_v denotes a weight assigned to every node v in G. It is an $\mathcal{NP}$-hard optimization problem. The polyhedral approach to solve this problem bases on the identification of every stable set S in G with a characteristic vector $\chi^S \in \{0, 1\}^{|V|}$ where

$$\chi_v^S = 1 \Leftrightarrow v \in S.$$

This allows us to reformulate the problem as

$$\max\{c^T \chi^S \mid S \text{ stable in } G\},$$

where the vector c contains the weights of the nodes corresponding to the single entries. Since we optimize a linear function, we can look at the convex hull of the characteristic vectors instead of looking at single points. This leads to

$$\max\{c^T x \mid x \in P(G)\},$$

where

$$P(G) = \text{conv}\big\{\chi^S \in \mathbb{R}^{|V|} \mid S \text{ stable set of } G\big\}.$$

We call $P(G)$ the **stable set polytope**. This polytope can be described as the set of solutions of a system of linear inequalities. Finding the right set of inequalities that describe $P(G)$ is in general not easy. Another approach to obtain $P(G)$ is to get it as a projection of another polytope. If a polytope can be mapped onto the stable set polytope by some affine map, it is called **extension** of the stable set polytope. The system of inequalities and equations that describe an extension of $P(G)$ is called **extended formulation** of $P(G)$. Usually extensions use extra variables and are therefore of higher dimension than $P(G)$. We will see more about that in the first chapter.

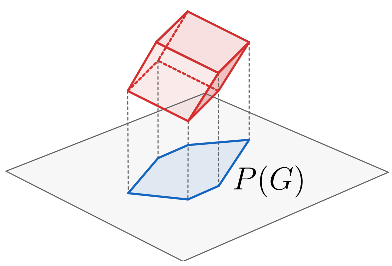


Figure 1.1: The stable set polytope $P(G)$ as projection of a higher dimensional polytope.

The minimum number of inequalities that are needed to describe an extension of a polytope is called **extension complexity** of the polytope. Finding a defining inequality system or an extended formulation of $P(G)$ that is of small size, i.e. polynomial on the input size, leads to the applicability of the tools of linear programming which offer a wide range of algorithms. In this work we will look at graphs G that can be decomposed into two graphs G_1 and G_2, e.g. like in figure 1.2.

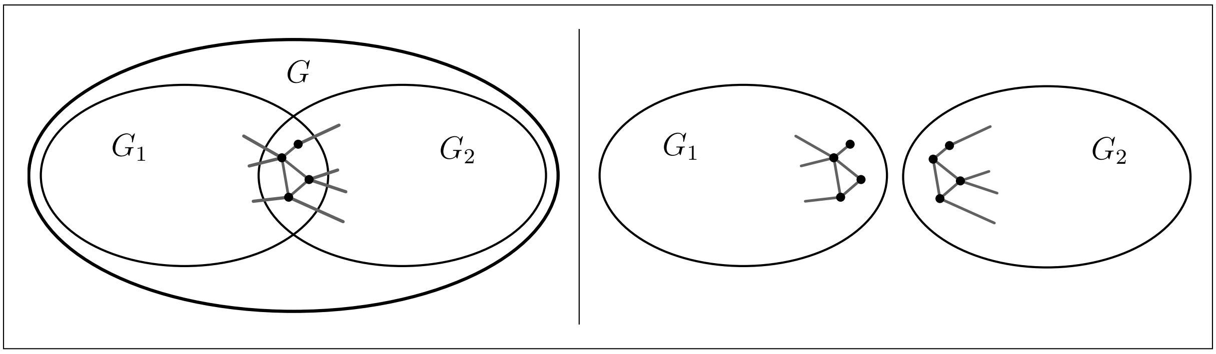


Figure 1.2: A graph G that is decomposable into the two graphs G_1 and G_2.

The set of nodes that decomposes the graph is called **(node) cut set**. We are now interested in techniques that allow us to describe the stable set polytope of G based on the stable set polytopes associated with the two pieces G_1 and G_2. Such techniques would allow us to describe the stable set polytope of a graph, which is recursively decomposable into graphs which have a known description. We also want to know how the extension complexity of $P(G)$ depends on the extension complexities of $P(G_1)$ and $P(G_2)$. A small extension complexity of $P(G)$ means that the stable set problem on G can be formulated by a linear program with a small number of inequalities. Crucial for our intention is the structure of the cut set which connects G_1 and G_2.
If the cut set is a clique, i.e. a complete graph, then the inequality description of $P(G)$ is just the union of the inequality descriptions of $P(G_1)$ and $P(G_2)$ [1]. So in this case a small description of $P(G_1)$ and $P(G_2)$ leads to a small description of $P(G)$. We will see that the strategy of just taking the union of the two inequality descriptions fails already in the case where we have a cut set that consists of two non-connected nodes. Now the question comes up if we can expect to find a similar simple way to handle this case. We will reduce the $\mathcal{NP}$-hard MAX-2-SAT-problem to the stable set problem on a graph, which is decomposable via such two node cut sets. That shows that this case is also $\mathcal{NP}$-hard. So for non-clique cut sets another approach is needed.
Here, extended formulations are getting involved. The idea is to look at a modification of G and obtain an extended formulation of $P(G)$ based on the stable set polytope of this modification. The main aim of such a modification is the construction of a suitable cut set.
One approach is to modify the cut set of G in such a way that it forms a clique [2]. We will call the modified graph G^+. As the cut set of G^+ now forms clique, we already know how to obtain the stable set polytope of G^+ from the polytopes associated with G_1^+ and G_2^+. $P(G)$ is then obtained by a projection of $P(G^+)$. A disadvantage of this method is that it adds a lot of edges between the nodes of the cut set and the rest of the graph. This could damage nice graph structures.
We will present an approach where the modification only affects the nodes in the cut set. We will call the modified graph $\tilde{G}$. Here, the stable set polytope of the modified graph

$P(\tilde{G})$ does not directly provide an extension of $P(G)$. We will find a face of $P(\tilde{G})$ that provides our wanted extension. Our construction will allow us to obtain a description of this face by the union of the descriptions of corresponding faces of $P(\tilde{G}_1)$ and $P(\tilde{G}_2)$. Our approach works with various modifications, where each of them adds a certain number of nodes. We will see that there is a minimum number of nodes that have to be added. We will observe that it is possible to find cases where this minimum number is achievable but we will also see an example where it is not possible.
The idea of taking a face as an extension is inspired by the work of Baharona and Mahjoub [3]. They applied this approach to the case where the cut set consists of two non connected nodes. Furthermore, they did not only find an extension of $P(G)$, they even found a way to describe it in the original space. We will show that their approach can be simplified in the case where one of the two components is bipartite.
Besides describing different techniques we will look at their impact on the extension complexity of $P(G)$.

2 Preliminaries

2.1 Graphs

We will use the basic notation of graph theory, i.e. $G = (V, E)$ denotes an undirected, loopless graph with nodeset V and edgeset E. An edge between u and $v \in V$ is denoted by uv. For a set of nodes $A \subset V$ we denote by $G \setminus A$ the graph G where the nodes in A together with the corresponding edges are deleted. $E_A \subset E$ denotes the edges between the nodes of A. $G_A = (A, E_A)$ is called the induced subgraph of A in G.

Definition 2.1. *Let $G = (V, E)$ be a graph. A **(node) cut set** $U \subset V$ is a set of nodes such that*

i. $G \setminus U$ disconnects.

ii. $G \setminus u$ does not disconnect for all $u \subsetneq U$.

Definition 2.2. *We say that a graph $G = (V, E)$ **decomposes** into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ if there is a cut set $U \in V$ such that*

i. $V_1 \cap V_2 = U$, $V_1 \cup V_2 = V$

ii. $(G_1)_U = (G_2)_U$

iii. $E_1 \cup E_2 = E$.

This means that the nodes and edges of an identical induced subgraph in G_1 and G_2 can be identified with each other such that it results in G.

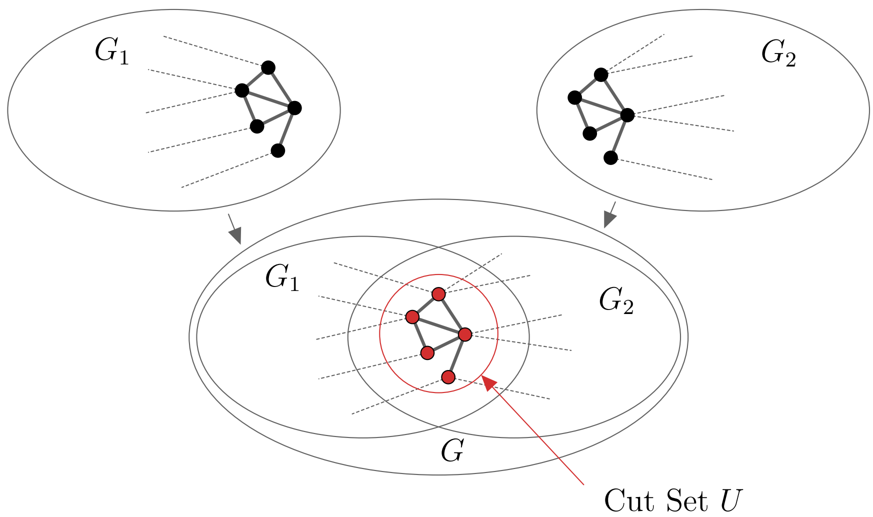


Figure 2.1: A graph G which is decomposable into the graphs G_1 and G_2.

If U is a clique, i.e. G_U is complete, we call U a **clique cut set**. The **neighbors** of a set of nodes $A \in V$ are all nodes of $V \setminus A$ that are connected to one of the nodes of A.

2.2 About the Stable Set Polytope

In this chapter we will take a closer look at the stable set polytope and different classes of its defining inequalities. An overview can be found in chapter 9 of [4]. Remember that the stable set polytope $P(G)$ is defined by

$$P(G) = \text{conv}\{\chi^S \in \mathbb{R}^{|V|} \mid S \text{ stable set of } G\},$$

Let $x \in \mathbb{R}^{|V|}$. Since no two neighbored nodes can be in a stable set and all characteristic vectors are non-negative, the following **edge** and **non-negativity constraints** are valid.

$$x_u + x_v \leq 1 \;\; \forall\, uv \in E,\; u, v \in V \tag{2.1}$$

$$x_v \geq 0 \;\; \forall\, v \in V \tag{2.2}$$

It can be shown that these inequalities are sufficient to describe the stable set polytope of a graph if and only if the graph is bipartite.
The simplest case in which (2.1) and (2.2) are not enough to describe the stable set polytope are odd cycles. For example for a cycle of five nodes the vector $x = (1/2, 1/2, 1/2, 1/2, 1/2)$ fulfills (2.1) and (2.2). An optimization in the direction of $c = (1, 1, 1, 1, 1)$ over the stable set polytope of the cycle should give us the maximum cardinality of a stable set in this cycle. This is 2. But the multiplication with x gives 2.5. Therefore x it is not in the stable set polytope of the cycle.
For this reason we introduce a new class of inequalities, the so called **odd-cycle inequalities**

$$\sum_{v \in C} x_v \leq \frac{|C| - 1}{2} \quad \forall\, C \text{ odd cycle of } G. \tag{2.3}$$

A graph is called **t-perfect** if (2.1), (2.2) and (2.3) suffice to define its stable set polytope. One example of this class of graphs are **almost bipartite graphs**. These are graphs that have a vertex v such that $G \setminus \{v\}$ is bipartite. We will see this class of graphs later again. In t-perfect and bipartite graphs the maximum weighted stable set can be found in polynomial time. The proofs can be found in [4].
Now we look at some notations. Let $x_1 \in P(G_1) \subset \mathbb{R}^{|V_1|}$ and $x_2 \in P(G_2) \subset \mathbb{R}^{|V_2|}$ and let x_1 and x_2 have the same values on the entries assigned to $U = V_1 \cap V_2$. Then $x = x_1 \cup x_2 \in \mathbb{R}^{|V|}$ denotes the vector where the entries belonging to U of x_1 and x_2 are identified with each other i.e.

- $x_v = (x_1)_v$ for $v \in V_1 \setminus U$
- $x_v = (x_2)_v$ for $v \in V_2 \setminus U$
- $x_v = (x_1)_v = (x_2)_v$ for $v \in U$.

The part of a vector $x \in P(G)$, that belongs to the nodes of G_i, $i = 1, 2$, is denoted by x_{G_i}. Similarly, we denote the part of x that belongs to the nodes of a set $S \subset V$ by x_S.
Let $ax \leq \alpha$, $x \in \mathbb{R}^{|V|}$ be an inequality. Then $V_a = \{v \in V \mid a_v \neq 0\}$ is called the **support** of the inequality. Note that a_v denotes the coefficient in the inequality before the variable that is assigned to v.

2.3 Extended Formulations

We have already seen the geometrical meaning of extended formulations in the introduction. Let's take a closer look at them.

Definition 2.3 (Extended Formulation [2]). *An extended formulation for a polytope $P \subseteq \mathbb{R}^n$ is a system of inequalities $Bx + Cy \leq d$, $y \in \mathbb{R}^m$ such that*

$$P = \{x \in \mathbb{R}^n : \exists y \in \mathbb{R}^m : \ Bx + Cy \leq d\} \tag{2.4}$$

The polytope that is defined by an extended formulation is called **extension**. In order to show that a polytope $\tilde{P}$ is an extension of a polytope P we will use the following two steps:

- The projection of every vertex of $\tilde{P}$ lies in P.
- For every vertex of P there exists a vector in $\tilde{P}$ that is projected to this vertex.

During this work we will hear the term "projection along variables". The projection of a vector $(x, y) \in \mathbb{R}^{n+m}$ along the variables $y \in \mathbb{R}^m$ is $x \in \mathbb{R}^n$. It is a projection onto the space of the x variables.

Definition 2.4 (Size of an extension [5]). *The **size** of an extension is the number of inequalities that are needed to describe it.*

Note that equations are not taken into account, because they can be eliminated by a reduction of variables.

Definition 2.5 (Extension Complexity [5]). *The **extension complexity** of a polytope P is the smallest size of any of its extensions. It is denoted by $xc(P)$.*

Balas proved the following extended formulation for the convex hull of a union of polytopes.

Theorem 2.6 (Balas [6]). *Let $P_i := \{x \in \mathbb{R}^n : A_i x \leq b_i\} \subseteq \mathbb{R}^n$, $i = 1, \ldots, k$, be nonempty polytopes. Then*

$$\begin{aligned} conv\left(\bigcup P_i\right) = \Big\{ & x \in R^n : \exists (x_1, \ldots, x_k, \lambda),\ x_i \in \mathbb{R}^n,\ \lambda \in \mathbb{R}^k s.t. \\ & x = \sum_{i=1}^{k} x_i;\ A_i x_i \leq \lambda_i b_i;\ \sum_{i=1}^{k} \lambda_i = 1, \lambda_i \geq 0\ ,\ i = 1 \ldots k \Big\}. \end{aligned} \tag{2.5}$$

The proof can be found in [6]. This theorem builds the basis for the following theorem. Here, we replace the inequality descriptions of the P_i's with extended formulations.

Theorem 2.7. *Let $B_i x + C_i y \leq b_i$ be an extended formulation of P_i, $i = 1, \ldots, k$. Then Balas formulation yields*

$$\begin{aligned} conv\left(\bigcup P_i\right) = \Big\{ & x \in R^n : \exists (x_1, \ldots, x_k, y_1, \ldots, y_k, \lambda),\ x_i \in \mathbb{R}^n,\ y_i \in \mathbb{R}^{m_i},\ \lambda \in \mathbb{R}^k \\ & s.t.\ x = \sum_{i=1}^{k} x_i;\ B_i x_i + C_i y_i \leq \lambda_i b_i;\ \sum_{i=1}^{k} \lambda_i = 1, \lambda_i \geq 0\ ,\ i = 1 \ldots k \Big\}. \end{aligned} \tag{2.6}$$

Proof. Description (2.5) has to give us the same polytope as the description (2.6). Therefore we have to show that x fulfills the inequalities of one description if and only if it fulfills the inequalities of the other description. So let

$$P = \{x \in \mathbb{R}^n : \ Ax \leq b\} = \{x \in \mathbb{R}^n : \ \exists y \in \mathbb{R}^m : Bx + Cy \leq d\} \tag{2.7}$$

be a polytope. We have to show that

$$Ax \leq \lambda b \Leftrightarrow Bx + Cy \leq \lambda d \tag{2.8}$$

for $\lambda \geq 0$ and $x \in \mathbb{R}^n$. If this is valid for an arbitrary polytope it is valid for every polytope P_i, $i = 1, \ldots, k$ and therefore (2.5) and (2.6) result in the same polytope.
" $\Rightarrow$ ": Let $Ax \leq \lambda b$. First we look at the case $\lambda > 0$. We know that

$$A \frac{1}{\lambda} x \leq b. \tag{2.9}$$

This means $\frac{1}{\lambda} x \in P$ and therefore, (2.7) yields

$$\exists y' : B \frac{1}{\lambda} x + Cy' \leq d \Leftrightarrow Bx + C\lambda y' \leq \lambda d. \tag{2.10}$$

So with setting $y = \lambda y'$ we found a y such that

$$Bx + Cy \leq \lambda d. \tag{2.11}$$

Now let $\lambda = 0$. Then we have

$$Ax \leq \mathbb{O}. \tag{2.12}$$

Here, $\mathbb{O}$ denotes the zero vector. Since the recession cone of P is $\{u \in \mathbb{R}^n : Au \leq \mathbb{O}\}$, we know that x is in this cone. This means that $\mu x \in P \ \forall \mu > 0$. We also know that P is bounded, so we have

$$x = \mathbb{O}. \tag{2.13}$$

If we now also choose $y = \mathbb{O}$ we have

$$Bx + Cy \leq \lambda \mathbb{O}. \tag{2.14}$$

" $\Leftarrow$ " Let $Bx + Cy \leq \lambda d$. We start again with the case $\lambda > 0$. We have

$$Bx + Cy \leq \lambda d \Leftrightarrow B \frac{1}{\lambda} x + C \frac{1}{\lambda} y \leq d \tag{2.15}$$

From (2.7) we have

$$A \frac{1}{\lambda} x \leq b \Leftrightarrow Ax \leq \lambda b. \tag{2.16}$$

Now let $\lambda = 0$. Then we have

$$Bx + Cy \leq \mathbb{O} \tag{2.17}$$

We now take some $\bar{x} \in P$ together with a $\bar{y}$ such that

$$B\bar{x} + C\bar{y} \leq d. \tag{2.18}$$

Adding a positive multiple of (2.17) to (2.18) gives us

$$B(\bar{x} + \mu x) + C(\bar{y} + \mu y) \leq d \text{ for } \mu \geq 0, \tag{2.19}$$

which means that $\bar{x} + \mu x \in P$ for all $\mu \geq 0$. We conclude again $x = \mathbb{O}$ since P is bounded and have

$$Ax \leq \mathbb{O}. \tag{2.20}$$

□

Assume now that the extended formulations used in theorem 2.7 are minimal, i.e. contain $xc(P_i)$ inequalities for each P_i, $i = 1, \dots, k$. Let's look at the number of inequalities that is used in the obtained formulation of theorem 2.7. The formulation contains $xc(P_i)$ inequalities plus the non-negativity constraint for λ_i for every P_i in the union. Therefore we can conclude that

$$xc\left(\bigcup P_i\right) \leq \sum_{i=1}^{k} (xc(P_i) + 1). \tag{2.21}$$

Furthermore, it could be shown in [5] that for polytopes of dimension of at least one this bound can be improved to

$$xc\left(\bigcup P_i\right) \leq \sum_{i=1}^{k} (xc(P_i)). \tag{2.22}$$

Since a stable set polytope of a non-empty graph always has dimension of at least 1, we can use (2.22) when we look at a union of stable set polytopes. So let's formulate $P(G)$ as a union of polytopes. What we do is going over all possible stable sets in the cut set and obtaining $P(G)$ by the union over these possibilities. This results in

$$P(G) = conv\left(\bigcup_{S \in U} \left\{x \in \mathbb{R}^{|V|} : x_{G_1} \in P(G_1),\ x_{G_2} \in P(G_2), x_U = \chi^S\right\}\right) \tag{2.23}$$

where S is stable in U. The set

$$\left\{x \in \mathbb{R}^{|V|} : x_{G_1} \in P(G_1),\ x_{G_2} \in P(G_2), x_U = \chi^S\right\} \tag{2.24}$$

has extension complexity $(xc(P(G_1)) + xc(P(G_2)))$, since the equations are not counted. If we now define

$$s = |\{S \subseteq U \mid S \text{ stable in } U\}| \tag{2.25}$$

we can use (2.22) and get

$$xc(P(G)) \leq s\,(xc(P(G_1)) + xc(P(G_2))). \tag{2.26}$$

3 Composed Graphs with Clique Cut Sets

In this chapter we will look at graphs that can be decomposed by a clique cut set. Chvátal [1] proved that in this case the inequality description of the composed graph is simply the union of the inequality descriptions of the stable set polytopes of its components.

Theorem 3.1. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $Q = V_1 \cap V_2$ be a clique cut set. Let $P(G_1)$, $P(G_2)$ and $P(G)$ be the stable set polytopes of G_1, G_2 and G. Then we have*

$$x \in P(G) \Leftrightarrow x_{G_1} \in P(G_1) \text{ and } x_{G_2} \in P(G_2).$$

We will give a different proof than Chvátal. For that proof we need the following lemma.

Lemma 3.2. *Let $\lambda_1, \lambda_2, \dots \lambda_n \in \mathbb{R}_{\geq 0}$, $\mu_1, \mu_2, \dots, \mu_m \in \mathbb{R}_{\geq 0}$ and*

$$K := \sum_{i=1}^{n} \lambda_i = \sum_{i=1}^{m} \mu_i.$$

Then there exist numbers $\eta_1, \eta_2, \dots, \eta_l \in \mathbb{R}_{\geq 0}$ with the following properties:

i. $\sum_{i=1}^{l} \eta_i = K$

ii. $\exists\ p_1 \leq p_2 \leq \dots \leq p_n \in \{1, \dots, l\} : \sum_{i=1}^{k} \lambda_i = \sum_{i=1}^{p_k} \eta_i \ \ \forall\ \ k = 1, \dots, n$

iii. $\exists\ q_1 \leq q_2 \leq \dots \leq q_m \in \{1, \dots, l\} : \sum_{i=1}^{k} \mu_i = \sum_{i=1}^{q_k} \eta_i \ \ \forall\ \ k = 1, \dots, m$

Proof. Let $\lambda_1, \lambda_2, \dots \lambda_n \in \mathbb{R}_{\geq 0}$, $\mu_1, \mu_2, \dots, \mu_m \in \mathbb{R}_{\geq 0}$ and

$$K =: \sum_{i=1}^{n} \lambda_i = \sum_{i=1}^{m} \mu_i. \tag{3.1}$$

Now look at the set

$$\{s_1, \dots, s_l\} = \Big\{\sum_{i=1}^{k} \lambda_i | k \in \{1, \dots, n\}\Big\} \cup \Big\{\sum_{i=1}^{k} \mu_i | k \in \{1, \dots, m\}\Big\} \tag{3.2}$$

with $s_1 < s_2 < \dots < s_l = K$ and define

$$\begin{aligned} \eta_1 &= s_1 \\ \eta_2 &= s_2 - s_1 \\ &\vdots \\ \eta_i &= s_i - s_{i-1} \text{ for } i = 2, \dots, l \\ &\vdots \\ \eta_l &= s_l - s_{l-1}. \end{aligned}$$

Then $\sum_{i=1}^{l} \eta_i = s_l = K$ since it is a telescoping sum. Furthermore, for every sum $\sum_{i=1}^{k} \lambda_i$ there exists an $r \in \{1, \dots, l\}$ such that $s_r = \sum_{i=1}^{k} \lambda_i$. Now set $p_k = r$. Then we have

$$\sum_{i=1}^{k} \lambda_i = s_r = \sum_{i=1}^{r} \eta_i = \sum_{i=1}^{p_k} \eta_i. \tag{3.3}$$

Since k is arbitrary statement *ii.* is true. Statement *iii.* can be obtained analogue. $\square$

Now we can prove theorem 3.1.

Proof. "$\Rightarrow$": This implication follows almost directly. We denote by $x_i \in \{0,1\}^{|V|}$, $i = 1,\dots,r$ characteristic vectors of stable sets in G. Then we have

$$x \in P(G) \tag{3.4}$$
$$\Rightarrow x = \sum_{i=1}^{r} \lambda_i x_i,\ \lambda \geq 0,\ \sum_{i=1}^{r} \lambda_i = 1 \tag{3.5}$$
$$\Rightarrow x_{G_k} = \sum_{i=1}^{r} \lambda_i (x_i)_{G_k},\ \lambda \geq 0,\ \sum_{i=1}^{r} \lambda_i = 1, k = 1,2 \tag{3.6}$$
$$\Rightarrow x_{G_k} \in P(G_k). \tag{3.7}$$

The last step follows since $(x_i)_{G_k} \in \{0,1\}^{|V_k|}$ is a characteristic vector of a stable set in G_k for $k = 1,2$, $i = 1,\dots,r$.

"$\Leftarrow$": Let now $x_{G_1} \in P(G_1)$ and $x_{G_2} \in P(G_2)$. Our goal is to show that $x = \{x_{G_1} \cup x_{G_2}\} \in \mathbb{R}^{|V|}$ is in $P(G)$. We note that x_{G_1} and x_{G_2} are both convex combinations

$$x_{G_1} = \sum_{i=1}^{n} \lambda_i a_i,\quad \lambda_i \geq 0 \ \text{ for } \ i = 1,\dots,n,\ \sum_{i=1}^{n} \lambda_i = 1 \tag{3.8}$$
$$x_{G_2} = \sum_{i=1}^{m} \mu_i b_i,\quad \mu_i \geq 0 \ \text{ for } \ i = 1,\dots,m,\ \sum_{i=1}^{m} \mu_i = 1, \tag{3.9}$$

where $a_i,\ i = 1,\dots,n$ and $b_i,\ i = 1,\dots,m$ are characteristic vectors of stable sets in G_1 and G_2, respectively. Since at most one node of the clique cut set Q can be in a stable set, the entries of a_i and b_i corresponding to the clique Q have at most one 1-entry, so

$$(a_i)_Q, (b_i)_Q \in \{e_1, e_2, \dots, e_{|Q|}, \mathbb{O}\}. \tag{3.10}$$

Here, $e_i \in \mathbb{R}^{|Q|}$ denotes the i'th unit vector for $i \in \{1,\dots,|Q|\}$. (3.10) tells us a lot about the structure of our convex combination. We know

$$(x_{G_1})_Q = (x_{G_2})_Q = \sum_{j=1}^{n} \lambda_i (a_i)_Q = \sum_{i=1}^{m} \mu_i (b_i)_Q \tag{3.11}$$

Note that $(x_{G_1})_Q$ and $(x_{G_2})_Q$ have a unique representation as a linear combination of unit vectors, since these are linearly independent. Therefore, the sum of coefficients λ_i that correspond to a certain unit vector is unique and equals the sum of coefficients μ_i which can be associated with the same unit vector. Formally this means for all $j = 1,\dots,|Q|$

$$\sum \{\lambda_i \mid (a_i)_Q = e_j\} = \sum \{\mu_i \mid (b_i)_Q = e_j\} \tag{3.12}$$

or, equivalently, for all $v \in Q$

$$\sum \{\lambda_i \mid (a_i)_v = 1\} = \sum \{\mu_i \mid (b_i)_v = 1\}. \tag{3.13}$$

Since we know that the sum of the coefficients equals 1, we can also conclude that

$$\begin{aligned}\sum\{\lambda_i \mid (a_i)_v = 0 \;\forall\; v \in Q\} &= 1 - \sum\{\lambda_i \mid (a_i)_v = 1 \text{ for some } v \in Q\}\\ &= 1 - \sum\{\mu_i \mid (b_i)_v = 1 \text{ for some } v \in Q\}\\ &= \sum\{\mu_i \mid (b_i)_v = 0 \;\forall\; v \in Q\}.\end{aligned} \tag{3.14}$$

Or goal now is to construct $x = \{x_1 \cup x_2\}$ as a convex combination

$$x = \eta_1 c_1 + \eta_2 c_2 + \cdots + \eta_r c_r, \;\; \sum_{k=1}^{r} \eta_k = 1, \;\; \eta_k \geq 0 \;\forall\; k = 1, \ldots, r \tag{3.15}$$

of points $c_k = \{a_i \cup b_j\},\; i \in \{1, \ldots, n\},\; j \in \{1, \ldots, m\},\; k = 1, \ldots, r$, which are characteristic vectors of stable sets in G. It is only possible to match the two vectors a_i and b_j if they can be assigned to the same vector of $\{e_1, e_2, \ldots, e_{|Q|}, \mathbb{O}\}$. Let's fix such a vector and say w.l.o.g. $a_1, a_2, \ldots, a_s$ and $b_1, b_2, \ldots, b_t$ are all vectors which belong to this vector and $\lambda_1, \ldots, \lambda_s$ and $\mu_1, \ldots, \mu_t$ are the coefficients in front of these vectors. From (3.13) and (3.14) we know that

$$\sum_{i=1}^{s} \lambda_i = \sum_{j=1}^{t} \mu_i =: K. \tag{3.16}$$

We can use lemma 3.2 and know that there are numbers $\eta_1, \ldots, \eta_l \in \mathbb{R}_{\geq 0}$ and integers $p_1, \ldots, p_s \in \{1, \ldots, l\}$, $q_1, \ldots, q_t \in \{1, \ldots, l\}$ such that

i. $\sum_{i=1}^{l} \eta_i = K$

ii. $\sum_{i=1}^{k} \lambda_i = \sum_{i=1}^{p_k} \eta_i$ for $k = 1, \ldots, s$

iii. $\sum_{i=1}^{k} \mu_i = \sum_{i=1}^{q_k} \eta_i$ for $k = 1, \ldots, t$

Therefore, one can rewrite

$$\begin{aligned}\sum_{i=1}^{s} \lambda_i a_i &= \sum_{i=1}^{1} \lambda_i a_1 + (\sum_{i=1}^{2} \lambda_i - \sum_{i=1}^{1} \lambda_i) a_2 + \cdots + (\sum_{i=1}^{s} \lambda_i - \sum_{i=1}^{s-1} \lambda_i) a_s &(3.17)\\ &= \sum_{i=1}^{p_1} \eta_i a_1 + (\sum_{i=1}^{p_2} \eta_i - \sum_{i=1}^{p_1} \eta_i) a_2 + \cdots + (\sum_{i=1}^{p_s} \eta_i - \sum_{i=1}^{p_{s-1}} \eta_i) a_s &(3.18)\\ &= \sum_{i=1}^{p_1} \eta_i a_1 + \sum_{i=p_1+1}^{p_2} \eta_i a_2 + \cdots + \sum_{i=p_{(s-1)}+1}^{p_s} \eta_i a_s &(3.19)\end{aligned}$$

and analogously

$$\sum_{j=1}^{t} \mu_j b_j = \sum_{i=1}^{q_1} \eta_i b_1 + \sum_{i=q_1+1}^{q_2} \eta_i b_2 + \cdots + \sum_{i=q_{(t-1)}+1}^{q_t} \eta_i b_t. \tag{3.20}$$

Note that $p_s = q_t = l$. Therefore, each η_k, $k \in \{1, \ldots, l\}$ is the coefficient of exactly one a_i and one b_j. For each η_k we can match a_i and b_j, since they belong to the same vector of $\{e_1, e_2, \ldots, e_{|Q|}, \mathbb{O}\}$. We define $c_k = a_i \cup b_j$. Note that c_k is the characteristic vector of a stable set of G. If we repeat this procedure for every vector in $\{e_1, e_2, \ldots, e_{|Q|}, \mathbb{O}\}$ we get all coefficients η_k and all vectors c_k to obtain the desired convex combination (3.15), which proves $x \in P(G)$. □

Now we proved that the inequality description of $P(G)$ is given by the union of the inequalities defining $P(G_1)$ and $P(G_2)$. Let us see what that means for the extension complexity of $P(G)$.

Corollary 3.3. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $Q = V_1 \cap V_2$ be a clique cut set. Let $P(G_1)$, $P(G_2)$ and $P(G)$ be the stable set polytopes of G_1,G_2 and G. Then we have*

$$xc(P(G)) \leq xc(P(G_1)) + xc(P(G_2)) \tag{3.21}$$

Proof. Let $B_1 x_{G_1} + C_1 y_1 \leq d_1$ and $B_2 x_{G_2} + C_2 y_2 \leq d_2$ be extended formulations of $P(G_1)$ and $P(G_2)$ that have minimum size.

$$\begin{aligned} P(G) &\overset{3.1}{=} \{x \in \mathbb{R}^V : x_{G_1} \in P(G_1) \text{ and } x_{G_2} \in P(G_2)\} \\ &= \{x \in \mathbb{R}^V : \exists y_1 : B_1 x_{G_1} + C_1 y_1 \leq d_1 \text{ and } \exists y_2 : B_2 x_{G_2} + C_2 y_2 \leq d_2\} \end{aligned}$$

Therefore the inequalities

$$\begin{aligned} B_1 x_{G_1} + C_1 y_1 &\leq d_1 \\ B_2 x_{G_2} + C_2 y_2 &\leq d_2 \end{aligned}$$

provide an extended formulation of $P(G)$ and the size of the minimum extended formulation has to be smaller or equal. □

Remark 3.4. *The condition of having a clique cut set cannot be dropped, which can be shown by a very simple example: Let's assume the graph G is a cycle of five nodes. G is decomposable into a path of three and a path of four nodes. We call them G_1 and G_2. This decomposition is shown in figure 3.1. Obviously the cut set $\{v_1, v_2\}$ is not a clique.*

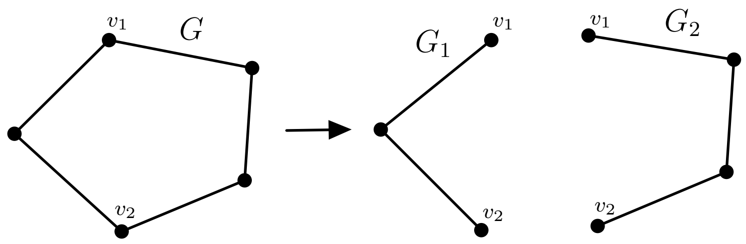


Figure 3.1: The graph G that decomposes into G_1 and G_2 on a cut set which consists of two disconnected nodes.

Now consider the vectors $x_1 = (1/2, 1/2, 1/2)^T$ and $x_2 = (1/2, 1/2, 1/2, 1/2)^T$. Since

$$\frac{1}{2}\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \end{pmatrix} \tag{3.22}$$

and

$$\frac{1}{2}\begin{pmatrix}1\\0\\1\\0\end{pmatrix}+\frac{1}{2}\begin{pmatrix}0\\1\\0\\1\end{pmatrix}=\begin{pmatrix}1/2\\1/2\\1/2\\1/2\end{pmatrix} \tag{3.23}$$

we have $x_1 \in P(G_1)$ and $x_2 \in P(G_2)$. The corresponding vector in $P(G)$ would be $x = x_1 \cup x_2 = (1/2, 1/2, 1/2, 1/2, 1/2)^T$. Let's optimize over $P(G)$ with the objective vector $c = (1, 1, 1, 1, 1)^T$. That should give us the cardinality of the maximum stable set in G. This would be a set of two nodes, so the value of our objective function would be 2. As $c^T x = 2.5$, x cannot be in $P(G)$. That shows that Chvátals theorem does not hold here.

We have seen that in the case of a two not cut set, where the nodes are not connected, it is not sufficient to just take the union of the inequality descriptions of the graph's components. In the next chapter we will check if we can expect an easy method to handle this case.

4 Hardness of the Stable Set Problem on a composed graph

In this chapter we will look at the a case where Chvátals theorem is not applicable, namely the case when the cut set consists of two disconnected nodes. If there was a way to obtain a result similar to 3.1 for this case, we could solve the stable set problem efficiently on the following graph: Let G be a graph that can be decomposed into a "main graph" $\bar{G}$ and some other graphs G_1, G_2, G_3 ... G_k. Assume that the cut set between $\bar{G}$ and each G_i, $i = 1, \ldots, n$ is a pair of non-connected nodes. Assume furthermore that the stable set polytope of all subgraphs can be easily described.

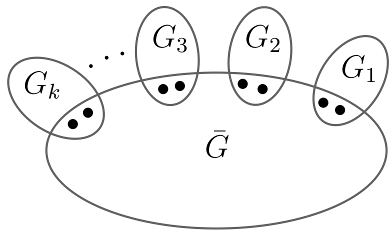


Figure 4.1: Scheme of a graph G which decomposes into a "main graph" $\bar{G}$ and the graphs G_i, $i = 1, \ldots, k$. The cut set between $\bar{G}$ and each G_i consists of two disconnected nodes.

First of all let's apply (2.23) to the two graphs $\bar{G}$ and G_1. Let's call the nodes of the corresponding cut set v_1 and v_2. Applying (2.23) to this case gives us the following:

$$\begin{aligned} P(G) = \mathrm{conv}\Big(& \Big\{x \in \mathbb{R}^{|V|} : x_{\bar{G}} \in P(\bar{G}), x_{G_1} \in P(G_1), x_{v_1} = 1, x_{v_2} = 1\Big\} \\ & \cup \Big\{x \in \mathbb{R}^{|V|} : x_{\bar{G}} \in P(\bar{G}), x_{G_1} \in P(G_1), x_{v_1} = 0, x_{v_2} = 1\Big\} \\ & \cup \Big\{x \in \mathbb{R}^{|V|} : x_{\bar{G}} \in P(\bar{G}), x_{G_1} \in P(G_1), x_{v_1} = 1, x_{v_2} = 0\Big\} \\ & \cup \Big\{x \in \mathbb{R}^{|V|} : x_{\bar{G}} \in P(\bar{G}), x_{G_1} \in P(G_1), x_{v_1} = 0, x_{v_2} = 0\Big\}\Big) \end{aligned}$$

In this case we can conclude

$$xc(P(G)) \leq 4\,(xc(P(\bar{G})) + xc(P(G_1)). \tag{4.1}$$

If we now apply this approach to the whole graph in figure 4.1, we get

$$xc(P(G)) \leq 4^k \left(\sum_{i=1}^{k} xc(P(G_i)) + xc(P(\bar{G})) \right) \tag{4.2}$$

So even if the extension complexity of all components was very small, we would just get an exponential bound for $xc(P(G))$. But can we still expect to find a way to handle the problem in polynomial time? We will show that this cannot be the case, by reducing the $\mathcal{NP}$-hard MAX-2-SAT problem to it. We start with the explanation of the latter.

Definition 4.1 (Conjunctive normal form (CNF))**.** *In boolean algebra a* ***conjunctive normal form*** *is a conjunction of clauses where each of these clauses is a disjunction of literals. A* ***literal*** *is a logic variable or its negation.*

Informally we can talk of an 'and' connection of 'ors'. We denote our logic variables as $x_i \in \{0,1\}$, $i = 1, \dots, n$.

Example 4.2. *An example of a conjunctive normal form is the following*

$$(x_0 \vee x_1) \wedge (\neg x_0 \vee x_1) \wedge (\neg x_0 \vee \neg x_1) \wedge (x_0 \vee \neg x_1)$$

In general satisfiability problems (SAT problems) one looks for an allocation of 1 and 0 values on the n logic variables, such that a given CNF is true. A special case is the 2-SAT problem. Here, we have the restriction that every clause of the CNF contains at most two literals. This problem is in $\mathcal{P}$. When we look at example 4.2 we see that no 1-0-allocation on x_0 and x_1 fulfills all clauses. This leads us to the MAX-2-SAT problem. Here, we look for the maximum number of clauses that can be satisfied in a given CNF. This optimization problem is $\mathcal{NP}$-hard. To show that the stable set problem on our composed graph G is also $\mathcal{NP}$-hard, we will reduce the MAX-2-SAT problem to it. We will use the decision version of the MAX-2-SAT problem for this matter, i.e. the question if one can fulfill at least a certain number of clauses in a given CNF. This decision version is $\mathcal{NP}$-complete.

The Reduction

Given a case of the MAX-2-SAT problem with k clauses on n variables, we construct our composed graph like in the following:

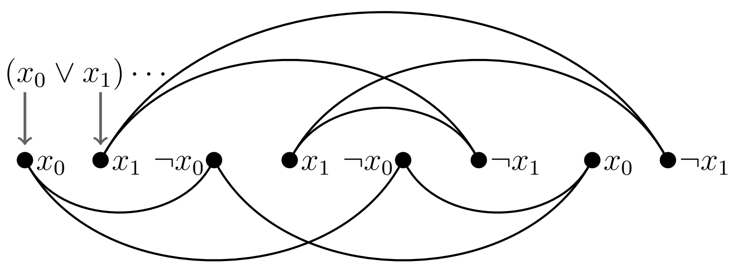
- $\bar{G}$: For each clause we introduce two nodes and label them corresponding to the two literals in the clause. We add edges to all pairs of nodes which correspond to a $\{x_j, \neg x_j\}$, $j = 1, \dots, n$ combination. We give all nodes the weight 1.
- G_i : Every graph G_i, $i = 1, \dots, k$ corresponds to one clause in the formula. Each graph is a path of four nodes, where the first and the last node are labeled with the two literals of the clause. These two nodes get the weight 1, the ones in the middle of the path stay unlabeled and are assigned with the weight $2k + 1$.

We compose these graphs by connecting the first and the last nodes of the paths G_i with the two nodes belonging to the same clause in $\bar{G}$. The whole procedure can be done in polynomial time. Let's demonstrate this construction using the CNF of example 4.2.

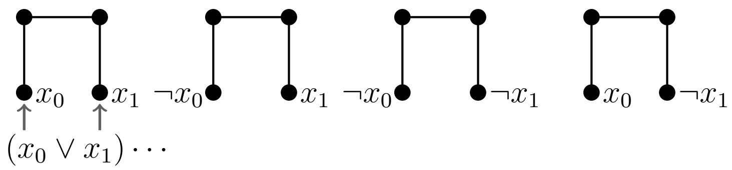
Example 4.3. *We look again at*

$$(x_0 \vee x_1) \wedge (\neg x_0 \vee x_1) \wedge (\neg x_0 \vee \neg x_1) \wedge (x_0 \vee \neg x_1) \tag{4.3}$$

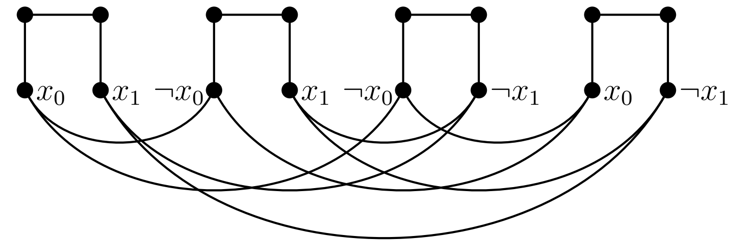
First we construct our graph $\bar{G}$.



Then we construct the G_i's. In this case we have four of them.



Together we get the decomposable graph G:



As demonstrated in the example we now have exactly what we wanted: a big main graph $\bar{G}$ which has connections consisting of two non-connected nodes to a k smaller graphs $G_1, \ldots, G_k$. These smaller graphs are even identical. Now we show that the used graphs are all bipartite, which means that the weighted stable set problem can be solved in polynomial time on each of them. When we look at the partition of $\bar{G}$ which divides the nodes in negated and not negated variables, i.e. $I_1 = \{v \mid \exists x_j, j = 1, \ldots, n : v \mathrel{\widehat{=}} x_j\}$, $I_2 = \{v \mid \exists x_j, j = 1, \ldots, n : v \mathrel{\widehat{=}} \neg x_j\}$, we see that no two nodes within I_1 or I_2 are connected. That shows that $\bar{G}$ is bipartite. Also all G_i, $i = 1, \ldots, k$ are bipartite, as they are just paths.

Proof that the Reduction is correct

Theorem 4.4. *Given an instance of the MAX-2-SAT problem and a graph G, constructed like above. Then the following holds:*

$$\text{In a given } CNF\ K \text{ clauses can be fulfilled.}$$
$$\Leftrightarrow$$
$$\text{There is a stable set in } G \text{ which has the weight } K + k(2k+1).$$

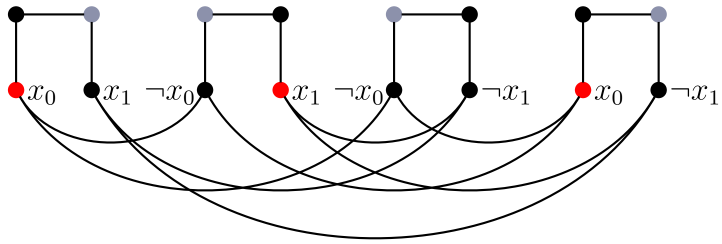
Proof. " $\Rightarrow$ " Given an instance of the MAX-2-SAT problem where K clauses of the corresponding CNF can be satisfied.
That means it is possible to select one true literal in K clauses. This choice is consistent in the sense that no complementary pair $\{x_j, \neg x_j\}$, $j = 1, \ldots, n$ is contained in the chosen set.

In our example we have e.g. $K = 3$. By setting $x_0 = 1$, $x_1 = 1$ the corresponding choice of variables looks like

$$(\color{red}{x_0} \vee x_1) \wedge (\neg x_0 \vee \color{red}{x_1}) \wedge (\neg x_0 \vee \neg x_1) \wedge (\color{red}{x_0} \vee \neg x_1).$$

Now we pick all nodes in G that correspond to the chosen literals. We get a stable set, since the only edges between these nodes connect a literal and its complement, both of which could not have been selected. At the moment our stable set has the weight K. We have selected at most one node in each subgraph G_i, $i = 1, \ldots, k$. Therefore, we can also pick one of the unlabeled nodes in each G_i and add them to the stable set without destroying the stability. That updates the weight of the stable set to $K + k(2k+1)$, since we added k nodes of weight $2k + 1$.

Let's visualize this in our example:



The red nodes are the ones that were chosen corresponding to the choice of literals in the CNF. The grey ones are the unlabeled nodes that were added.

" $\Leftarrow$ " Assume there is a stable set in G with the weight $K + k(2k + 1)$.
Since the weight of all labeled nodes together is $2k$, the weight of $k(2k + 1)$ has to come from k unlabeled nodes. Otherwise such a high value would not be achievable. So in every subgraph G_i, $i = 1, \ldots, k$ exactly one of the unlabeled nodes has to be in this stable set. There cannot be more of the unlabeled nodes in the stable set, since this would destroy stability. It follows that K of the literal corresponding nodes have to be in the stable set. They all belong to different clauses, since the one unlabeled node of each subgraph G_i, which has to be in our stable set, prevents that both nodes corresponding to a clause are together in the stable set. Also no complementary pair $\{x_j, \neg x_j\}$, $j = 1, \ldots, n$, of variables is in the stable set, since the edges of $\bar{G}$ prevent this. So if we pick the literals of the CNF corresponding to the labeled nodes in the stable set of G, this choice is consistent and covers K clauses. So a true assignment to this literals leads to the satisfiability of K clauses in our given CNF. $\square$

Thus we now know that our reduction is correct, we can conclude that the stable set problem on our composed graph G is $\mathcal{NP}$-complete in its decision version and $\mathcal{NP}$-hard in its optimization version, even if we can solve it on the single components. That means that we cannot expect to find a way that is as simple as Chvátals approach to handle the case of a two node cut set where the nodes are disconnected.

5 An Extension with a Clique Cut Set

Since Chvátal's theorem is only applicable to a special case, we are looking for a more general method. Conforti, Gerards and Pashkovich present an approach in [2] where the graph is modified in such a way that Chvátals theorem is applicable. The stable set polytope of this modified graph then provides an extension of $P(G)$. The modification takes place via so-called **clique lifting**.

Definition 5.1 (Clique Lifting)**.** *Let $G = (V, E)$ be an undirected graph and let $U \subseteq V$. A clique lift G^+ of U from G is the replacement of the nodes in U by a clique of the nodes*

$$\{w_S \mid S \text{ non-empty stable set of } U\}. \tag{5.1}$$

Each of these new nodes is connected to the neighbors of the stable set the node represents. We call the set of added nodes U^+.

Let's look at this procedure in the example of a two node cut set where the nodes are disconnected.

Example 5.2 (Clique lift of a graph with a cut set consisting of two disconnected nodes)**.** *Let's assume we have a graph $G = (V, E)$ which is decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ where $V_1 \cap V_2 = U = \{v_1, v_2\}$ and v_1, v_2 disconnected. The non-empty stable sets in U are $\{v_1\}$, $\{v_2\}$ and $\{v_1, v_2\}$. Therefore, we substitute U with three nodes, as shown in the following figure.*

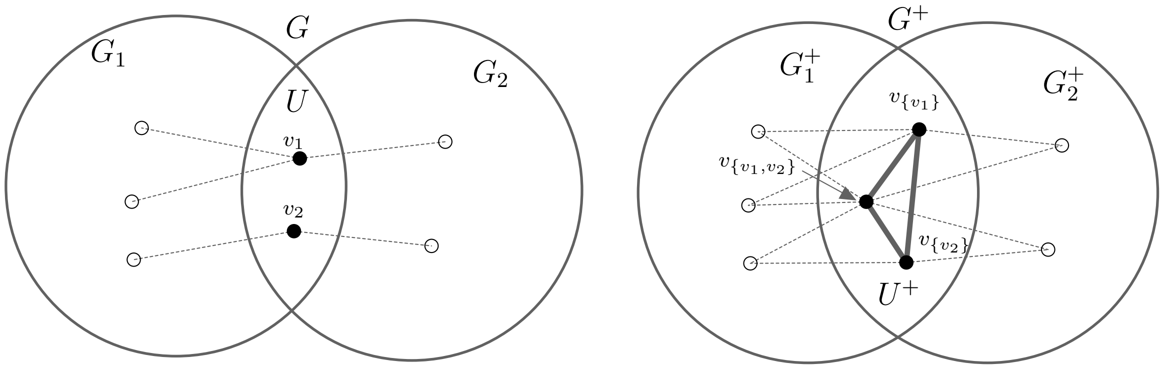


Figure 5.1: A graph G which decomposes into G_1 and G_2 on a cut set that consists of two disconnected nodes before and after the clique lifting.

Now G^+ has a clique cutset and we can use Chvátals Theorem.

The applicability of Chvátals theorem leads to the following corollary.

Corollary 5.3. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $U = V_1 \cap V_2$. Moreover, let G^+, G_1^+ and G_2^+ be the clique lifts of U from G, G_1 and G_2. Then we have*

$$x \in P(G^+) \Leftrightarrow x_{G_1^+} \in P(G_1^+) \text{ and } x_{G_2^+} \in P(G_2^+). \tag{5.2}$$

For the extension complexity this means the following.

Corollary 5.4. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $U = V_1 \cap V_2$. Let G^+, G_1^+ and G_2^+ be the clique lifts of U from G, G_1 and G_2. Then we have*

$$xc(P(G)) \leq xc(P(G^+)) \leq xc(P(G_1^+)) + xc(P(G_2^+)) \tag{5.3}$$

Since corollary 5.3 only gives us an inequality description of $P(G^+)$ we are interested in the projection which gives us $P(G)$.

Theorem 5.5. *Let G^+ be the clique lift of U in G. Then the stable set polytope $P(G)$ is the image of $P(G^+)$ under the projection defined by*

$$p_v(x) = \begin{cases} \sum\limits_{S \text{ in } U: v \in S} x_{v_S} & \text{if } v \in U \\ \\ x_v & \text{otherwise} \end{cases} \tag{5.4}$$

Proof. We will show that every projected vertex of $P(G^+)$ lies in $P(G)$ and that every vertex of $P(G)$ has a preimage in $P(G^+)$. Note that at most one node v_S of U^+ can be in a stable set of G^+. Let $S_{G \setminus U}$ be a stable set on the nodes of $G \setminus U$ and S be a non empty stable set in U. From the construction of G^+ we have

$$S_{G\setminus U} \cup \{v_S\} \text{ is stable in } G^+ \quad \Leftrightarrow \quad S_{G\setminus U} \cup S \text{ is stable in } G \tag{5.5}$$

$$S_{G\setminus U} \text{ is stable in } G^+ \quad \Leftrightarrow \quad S_{G\setminus U} \text{ is stable in } G. \tag{5.6}$$

The projection of the characteristic vector of $S_{G\setminus U} \cup \{v_S\}$ in $P(G^+)$ is the characteristic vector of $S_{G\setminus U} \cup S$ in $P(G)$ (analogue for 5.6). Since $S_{G\setminus U}$, v_S and S were chosen arbitrary, we see that every vertex of $P(G^+)$ is projected to an element of $P(G)$ and that every vertex of $P(G)$ has a preimage in $P(G^+)$. That proves that $P(G)$ is the image of $P(G^+)$ which means that $P(G^+)$ is an extension of $P(G)$. □

To obtain $P(G^+)$ one needs to know the inequality description of $P(G_1^+)$ and $P(G_2^+)$. If G_1 and G_2 have a nice structure, the clique lift could destroy this structure as a lot of new edges within the whole graph are added. In the next chapter we will look at a method where the modification only affects the node of the cut set U.

6 Faces as Extension

6.1 A general Approach

Before we start this chapter we look at a short recap about faces of polyhedra.

Remark 6.1 (Faces of polytopes)**.** *Let $ax \le \alpha$ be a valid inequality of a polytope P, i.e. $ax \le \alpha \;\forall\, x \in P$. Then the set $F = \{x \in P : ax = \alpha\}$ is a called a face of P. By definition F is also a polytope. All vertices of a polytope are faces. A set $F' \subseteq F$ is a face of F if and only if it is a face P. That means that $x \in F$ is a vertex of F if and only if it is a vertex of P. Therefore, all faces of an integer polytope are again integer polytopes. As all the vertices of the stable set polytope are characteristic vectors of stable sets, all vertices of its faces are also characteristic vectors of stable sets. For more information about faces of polytopes refer to chapter 3.1 of [7].*

We start again with a graph $G = (V, E)$ which is decomposable into two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Let $U = V_1 \cap V_2$ be the cut set. The goal is again to find an extended formulation for $P(G)$. The idea for this approach comes from Barahona and Mahjoub [3], who applied it to a cut set which consists of two disconnected nodes. We will have a look at their work later. Let us outline our intention. We want to start again with a modified version of G. We will modify our graph in such a way that all edges we add only affect the nodes in the modified cut set. The modified graph is called $\tilde{G} = (\tilde{V}, \tilde{E})$ and the two graphs $\tilde{G}_1 = (\tilde{V}_1, \tilde{E}_1)$ and $\tilde{G}_2 = (\tilde{V}_2, \tilde{E}_2)$ are the corresponding modified versions of G_1 and G_2. In opposite to the clique lifting approach that we have seen in the last chapter, the nodes of U should not be removed. The aim is now to find a face $F(\tilde{G})$ of $P(\tilde{G})$ such that $P(G)$ is the projection of $F(\tilde{G})$ along the added variables, like visualized in figure 6.1.

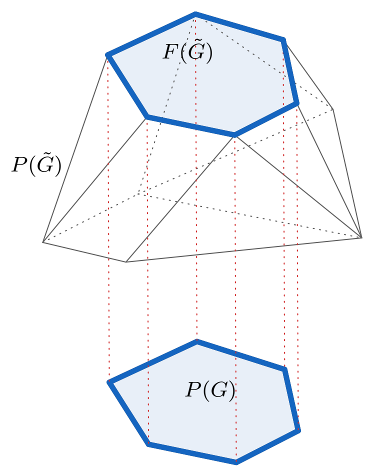


Figure 6.1: $P(G)$ as projection of a face $F(\tilde{G})$ of a higher dimensional stable set polytope $P(\tilde{G})$.

We start by analyzing the assumptions that ensure that such a face exist. Furthermore, we want to know how to obtain an inequality description of that face.

Theorem 6.2. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $U = V_1 \cap V_2$ be the cut set. Let $\tilde{G} = (\tilde{V}, \tilde{E})$ be the graph that evolved out of G by enlarging the cut set to $\tilde{U} = U \cup \{w_1, w_2, \ldots, w_k\}$ together with edges within the new cut set. Let $\tilde{G}_1 = (\tilde{V}_1, \tilde{E}_1)$ and $\tilde{G}_2 = (\tilde{V}_2, \tilde{E}_2)$ be the corresponding modifications of G_1 and G_2. Let $a\tilde{x} \leq \alpha$, $\tilde{x} \in \mathbb{R}^{|V|}$ be an inequality with the following properties:*

i. The support of the inequality lies in $\tilde{U}$.

ii. $a\tilde{x} \leq \alpha$ defines a face of $P(\tilde{G}_1)$, $P(\tilde{G}_2)$ and $P(\tilde{G})$.

iii. For all stable sets $S \in U$ there exists a stable set $\tilde{S} \in \tilde{U}$ with $a\chi^{\tilde{S}} = \alpha$ such that $U \cap \tilde{S} = S$.

iv. The set $\{\chi^{\tilde{S}}_{\tilde{U}} \mid a\chi^{\tilde{S}} = \alpha\ ,\ \tilde{S} \neq \emptyset \text{ stable in } \tilde{U}\}$ is linearly independent.

Then $P(G)$ is the projection of $F(\tilde{G}) = \{\tilde{x} \in P(\tilde{G}) : a\tilde{x} = \alpha\}$ along the variables corresponding to $w_1, w_2, \ldots, w_k$ and the following holds:

$$\tilde{x} \in F(\tilde{G}) \Leftrightarrow \tilde{x}_{\tilde{G}_1} \in F(\tilde{G}_1) \text{ and } \tilde{x}_{\tilde{G}_2} \in F(\tilde{G}_2) \tag{6.1}$$

Where $F(\tilde{G}_1) = \{\tilde{x} \in P(\tilde{G}_1) \mid a\tilde{x} = \alpha\}$ and $F(\tilde{G}_2) = \{\tilde{x} \in P(\tilde{G}_1 \mid a\tilde{x} = \alpha\}$.

That means that $F(\tilde{G})$ is described by the union of inequalities that define $F(\tilde{G}_1)$ and $F(\tilde{G}_2)$. Let's prove this.

Proof. First we prove that $F(\tilde{G})$ provides the wanted extended formulation. We have to show that the projection of all vertices of $F(\tilde{G})$ lies in $P(G)$. Remember that this projection just "deletes" the entries assigned to the nodes $w_1, w_2, \ldots, w_k$ of a vector. Let $\tilde{x}$ be a vertex of $F(\tilde{G})$. As $F(\tilde{G})$ is a face of $P(\tilde{G})$ all its vertices are integer and correspond to stable sets in $\tilde{G}$. Deleting the components of the added nodes in $\tilde{x}$ gives us a vector $x \in \mathbb{R}^{|V|}$ which corresponds to a stable set of G and therefore $x \in P(G)$.
Now we show that there exists a corresponding $\tilde{x} \in F(\tilde{G})$ that is projected onto x for every vertex $x \in P(G)$. So let $x \in P(G)$ be a vertex of $P(G)$. Since x is a characteristic vector of a stable set in G, x_U belongs to a stable set $S \in U$. From the assumptions we made, we know that there exists a stable set $\tilde{S} \in \tilde{U}$ with $a\chi^{\tilde{S}} = \alpha$ such that $U \cap \tilde{S} = S$. Let us now add the corresponding components $\tilde{x}_{w_1}, \tilde{x}_{w_2}, \ldots, \tilde{x}_{w_k}$ of this $\tilde{S}$ to x. We get a vector $\tilde{x}$ which corresponds to a stable set in $\tilde{G}$. Furthermore we have $a\tilde{x} = \alpha$, so $\tilde{x}$ is in $F(\tilde{G})$. We conclude

$$P(G) = \{x \mid \exists \tilde{x}_{w_1}, \tilde{x}_{w_2}, \ldots, \tilde{x}_{w_k} : (x, \tilde{x}_{w_1}, \tilde{x}_{w_2}, \ldots, \tilde{x}_{w_k}) \in F(\tilde{G})\}, \tag{6.2}$$

so $F(\tilde{G})$ is an extension of $P(G)$.
Now we show that

$$\tilde{x} \in F(\tilde{G}) \Leftrightarrow \tilde{x}_{\tilde{G}_1} \in F(\tilde{G}_1) \text{ and } \tilde{x}_{\tilde{G}_2} \in F(\tilde{G}_2) \tag{6.3}$$

holds. Here we use a similar proof to the one of theorem 3.1.
" $\Rightarrow$ " This direction is analogue to the proof of theorem 3.1.
" $\Leftarrow$ " Let $\tilde{x}_{\tilde{G}_1} \in F(\tilde{G}_1)$ and $\tilde{x}_{\tilde{G}_2} \in F(\tilde{G}_2)$. We can write these vectors as convex

combinations of characteristic vectors of stable sets in $F(\tilde{G}_1)$ and $F(\tilde{G}_2)$, respectively. From the assumptions we know that the cut set's corresponding parts of these vectors are linearly independent. Now we can use the same argumentation as in the proof of theorem 3.1 and match the characteristic vectors of $\tilde{G}_1$ and $\tilde{G}_2$ with suitable coefficients such that

$$\tilde{x} = \tilde{x}_{\tilde{G}_1} \cup \tilde{x}_{\tilde{G}_1}$$

can be represented as convex combinations of characteristic stable set vectors of $\tilde{G}$. That shows $\tilde{x} \in F(\tilde{G})$. □

The procedure that is used to obtain $P(G)$ in theorem 6.2 is shown in the following figure.

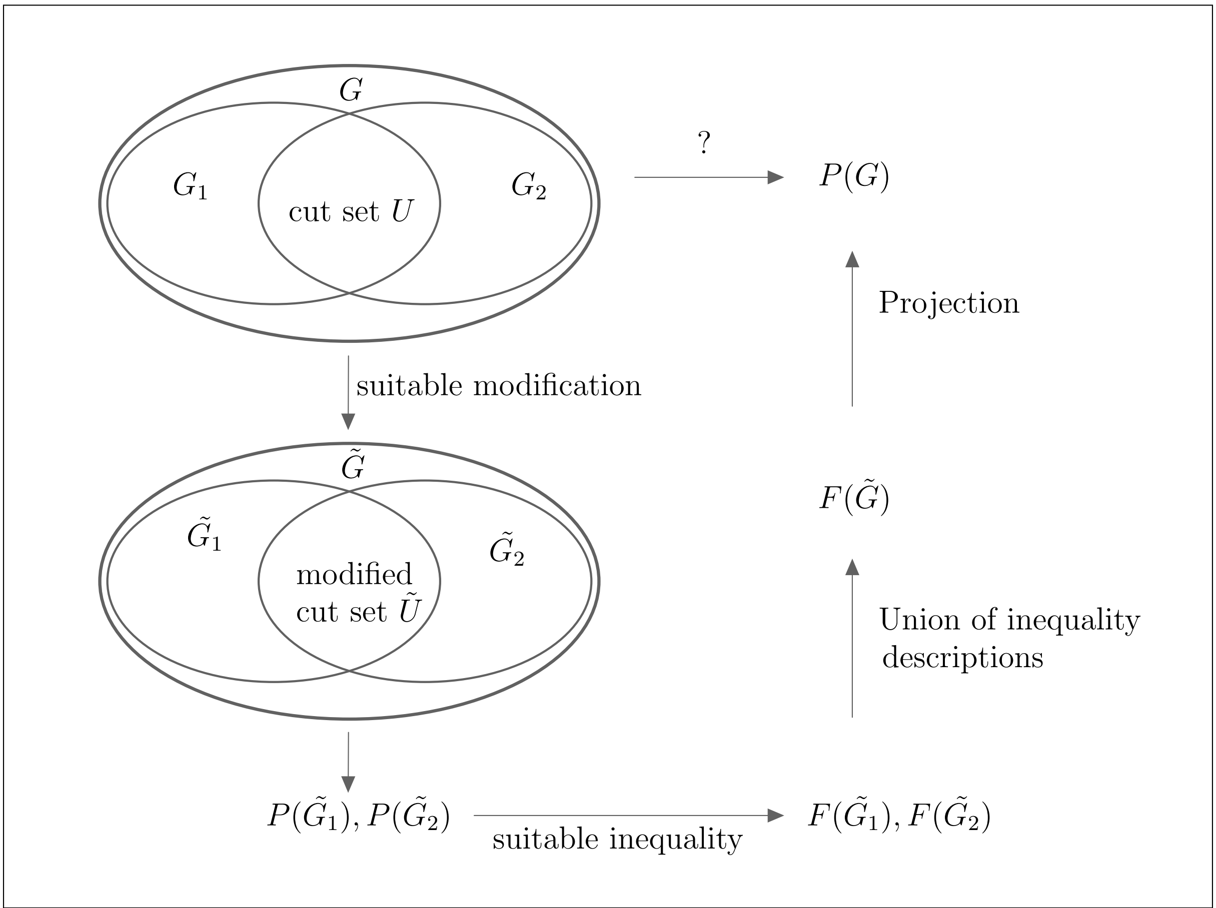
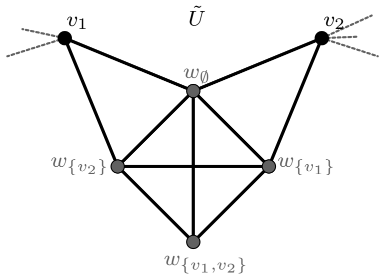


Figure 6.2: The way how $P(G)$ is obtained by the projection of a face of $P(\tilde{G})$ in the sense of theorem 6.2.

This theorem gives us information about the assumptions on the existence of such a face. Now the question comes up if we can find a modification for any arbitrary cut set such that the assumptions of theorem 6.2 are fulfilled. We will present an approach, which makes it possible to construct a suitable modification and a suitable inequality for every kind of cut set. The idea is shown in the following example.

Example 6.3. *Let G be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. We assume that $U = V_1 \cap V_2 = \{v_1, v_2\}$ and that v_1 and v_2 are not connected. In this cut set we have four possibilities of stable sets, namely $\{v_1, v_2\}$, $\{v_1\}$, $\{v_2\}$ and $\emptyset$.*
Now we have to construct our modified graphs $\tilde{G}_1$ and $\tilde{G}_2$. We add a node for every possible stable set in U, i.e. $\tilde{U} = \{v_1, v_2, w_{\{v_1,v_2\}}, w_{\{v_1\}}, w_{\{v_2\}}, w_\emptyset\}$, and complete them with edges to a clique. We add edges between a new node and all nodes of U which are not in the stable set the new node belongs to. This means we add the edges $w_{\{v_1\}}v_2, w_{\{v_2\}}v_1, w_\emptyset v_1$ and $w_\emptyset v_2$. Let's look at the resulting graph.



Now we have to choose a suitable inequality $a\tilde{x} \leq \alpha$. For α we choose the maximum cardinality of a stable set in U plus one. In our case this is 3. As the support of this inequality has to be in $\tilde{U}$, we only need coefficients for the variables assigned to the nodes of $\tilde{U}$. The variables that belong to the nodes in U get the coefficient 1. A variable that belongs to a node which is assigned to a stable set of U gets the coefficient α minus the cardinality of the assigned stable set. In our example this leads to the inequality

$$\tilde{x}_{v_1} + \tilde{x}_{v_2} + \tilde{x}_{w_{\{v_1,v_2\}}} + 2\tilde{x}_{w_{\{v_1\}}} + 2\tilde{x}_{w_{\{v_2\}}} + 3\tilde{x}_{w_\emptyset} \leq 3. \tag{6.4}$$

Now we have defined the graphs $\tilde{G}_1$, $\tilde{G}_2$ and $\tilde{G}$ and we have an inequality. As desired the added nodes are only connected with each other and the nodes in U. We now check if theorem 6.2 is applicable.

i. *The support of (6.4) lies in $\tilde{U}$*

ii. *(6.4) is valid, i.e. it defines a face.*

iii. *Within $\{v_1, v_2, w_{\{v_1,v_2\}}\}$, $\{v_1, w_{\{v_1\}}\}, \{v_2, w_{\{v_2\}}\}$ and $\{w_\emptyset\}$ we can find a stable set $\tilde{S} \in \tilde{U}$ with $a\chi^{\tilde{S}} = \alpha$ for all stable sets $S \in U$ such that $U \cap \tilde{S} = S$.*

iv. *The characteristic vectors of all stable sets $\tilde{S}$ in $\tilde{U}$ with $a\chi^{\tilde{S}} = \alpha$ are the columns of the following matrix*

$$\begin{array}{c} v_1 \\ v_2 \\ w_{\{v_1,v_2\}} \\ w_{\{v_1\}} \\ w_{\{v_2\}} \\ w_\emptyset \end{array} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{6.5}$$

As the part of the matrix that belongs to the added nodes builds the unit matrix these columns are clearly linearly independent.

The conditions of the theorem are fulfilled and we can conclude that

$$F(\tilde{G}) = \{\tilde{x} \in P(\tilde{G}) \mid \tilde{x}_{v_1} + \tilde{x}_{v_2} + \tilde{x}_{w_{\{v_1,v_2\}}} + 2\tilde{x}_{w_{\{v_1\}}} + 2\tilde{x}_{w_{\{v_2\}}} + 3\tilde{x}_{w_\emptyset} = 3\} \tag{6.6}$$

provides an extended formulation of $P(G)$ and

$$\tilde{x} \in F(\tilde{G}) \Leftrightarrow \tilde{x}_{\tilde{G}_1} \in F(\tilde{G}_1) \ \text{ and } \ \tilde{x}_{\tilde{G}_2} \in F(\tilde{G}_2) \tag{6.7}$$

The procedure from example 6.3 can be done for any kind of cut set. We formalize it in the following theorem.

Theorem 6.4. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $U = V_1 \cap V_2$. Let $\tilde{G}_1$, $\tilde{G}_2$ and $\tilde{G}$ be constructed as follows:*

1. *For every stable set $S \in U$ add a node w_S.*
2. *Connect the set $\{w_S \mid S \text{ stable in } U\}$ to a clique.*
3. *Add the edges $\{w_S v,\ v \in \{U \setminus S\}\}$ for every node in $\{w_S \mid S \text{ stable set of } U\}$.*

Let $a\tilde{x} \le \alpha$ be an inequality with the following properties

1. $a_v = 0 \;\forall\, v \in \tilde{V}_1 \cup \tilde{V}_2 \setminus \tilde{U}$
2. $\alpha = max\{|S| + 1,\ S \text{ stable set of } U\}$
3. $a_v = 1 \;\forall\, v \in U$
4. $a_{w_S} = \alpha - |S|\ ,\ S \text{ stable in } U$

Then theorem 6.2 is applicable.

Proof. Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $U = V_1 \cap V_2$. Let $\tilde{G}_1$, $\tilde{G}_2$ and $\tilde{G}$ and $a\tilde{x} \le \alpha$ be defined like in theorem 6.4. First recognize that the added nodes are only connected with each other and the nodes in U. We have to show that all assumtions of theorem 6.2 are satisfied.

i. The support of the inequality lies in $\tilde{U}$.

This fact follows from the definition of $a\tilde{x} \le \alpha$.

ii. The inequality $a\tilde{x} \le \alpha$ defines a face of $P(\tilde{G}_1)$, $P(\tilde{G}_2)$ and $P(\tilde{G})$.

Since the support of $a\tilde{x} \le \alpha$ lies in $\tilde{U}$, we only have to check that all characteristic vectors of stable sets $\tilde{S}$ in $\tilde{U}$ fulfill the inequality. If $\tilde{S}$ only contains nodes in U, the inequality is fulfilled since α is larger than the maximum weight of such a stable set. So let $\tilde{S}$ be a stable set that contains a node $w_S \in \tilde{U} \setminus U$. Note that it is not possible that it contains more than one such node. The only nodes that can be in a stable set with w_S are the nodes in the corresponding stable set S, so the maximum stable set which contains a certain w_S is $w_S \cup S$. Inserting $\chi^{(w_S \cup S)}$ into our inequality gives us

$$(\alpha - |S|) + |S| = \alpha. \tag{6.8}$$

Note that $(\alpha - |S|)$ is the coefficient of x_{w_S} and $|S|$ is the sum of all coefficients of x_v, $v \in S \subseteq U$. We see that our inequality is valid and defines a face of $P(\tilde{G})$. Since the support lies in $\tilde{U}$ it is also valid for $P(\tilde{G}_1)$ and $P(\tilde{G}_2)$ and defines a face there.

iii. For all stable sets $S \in U$ exists a stable set $\tilde{S} \in \tilde{U}$ with $a\chi^{\tilde{S}} = \alpha$ such that $S \cap \tilde{S} = S$.

Let S be a stable set in U. If we look a the set $\tilde{S} = S \cup w_S \in \tilde{U}$, we find that it is still stable, since no node in S is connected to w_S. First notice that $U \cap \tilde{S} = S$. Now we insert the characteristic vector of this set into $a\tilde{x} \leq \alpha$ and get

$$(\alpha - |S|) + |S| = \alpha, \tag{6.9}$$

i.e. $a\chi^{\tilde{S}} = \alpha$. Therefore this assumption is fulfilled.

iv. The set $\{\chi_{\tilde{U}}^{\tilde{S}} \mid a\chi^{\tilde{S}} = \alpha \ , \ \tilde{S} \text{ stable in } \tilde{U}\}$ is linearly independent.

All stable sets $\tilde{S} \in \tilde{U}$ with $a\chi^{\tilde{S}} = \alpha$ are of the form $\tilde{S} = S \cup w_S$. The corresponding characteristic vectors look like the columns of the following matrix

$$\begin{array}{c} U \\ w_{S_1} \\ w_{S_2} \\ \vdots \\ w_{S_k} \end{array} \begin{pmatrix} x_{S_1} & x_{S_2} & \dots & x_{S_k} \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \tag{6.10}$$

where $S_1, S_2, \dots, S_k$ are all possible stable sets in U. Since the components corresponding to $w_{S_1}, w_{S_2}, \dots, w_{S_k}$ are clearly linear independent, all columns are linear independent.

We showed that all assumptions of theorem 6.4 are fulfilled and it is therefore applicable. □

With this approach we have something that works for every kind of cut set. The number of nodes that have to be added is the number of all possible stable sets in U. We call this number again

$$s = |\{S \mid S \text{ stable in } U\}|.$$

This can be a lot. But what is the minimum number of nodes that we have to add to use theorem 6.2? Let's look at the set

$$\{\chi_{\tilde{U}}^{\tilde{S}} \mid a\chi^{\tilde{S}} = \alpha \ , \ \tilde{S} \text{ stable in } \tilde{U}\}. \tag{6.11}$$

We need a stable set $\tilde{S} \in \tilde{U}$ for **all** stable sets $S \in U$ such that

$$a\chi^{\tilde{S}} = a\chi_{\tilde{U}}^{\tilde{S}} = \alpha \tag{6.12}$$

and $S \cap \tilde{S} = S$. That means that (6.11) has to contain at least s vectors. The first equality of (6.12) holds since the support of $ax \leq \alpha$ lies in $\tilde{U}$. We need that all vectors in (6.11) are linear independent. A set of s vectors can only be linearly independent when every vector has at least s entries. That means that we need s nodes in $\tilde{U}$. Since $U \subseteq \tilde{U}$, we have to add at least $s - |U|$ new nodes. In the next chapter we will check if this minimum number of added nodes is always achievable.

6.2 Minimum Dimension of a Face as Extension

In the last chapter we saw that we have to add at least $s - |U|$ nodes in order to use theorem 6.2. In this chapter we will check if this number is achievable and furthermore if it is always achievable. For this matter we will look at two examples where the cut sets consist of three nodes.

The minimum number of added nodes is achievable

We will now present an example where the minimum number of added nodes can be achieved. So let's look at a graph $G = (V, E)$ which is decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Let $V_1 \cap V_2 = U = \{v_1, v_2, v_3\}$ and let v_1 and v_2 are connected. We look at a three node cut set of the following form:

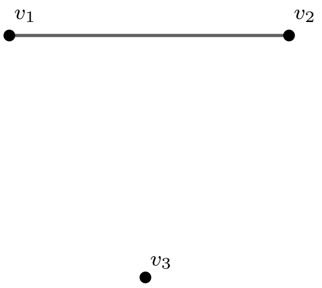


Figure 6.3: Three node cut set before modification.

Within this cut set we have six possible stable sets:

$$\{v_1, v_3\},\ \{v_2, v_3\},\ \{v_1\},\ \{v_2\},\ \{v_3\},\ \emptyset \tag{6.13}$$

That means that the minimum number of nodes we have to add is three, so let's set $\tilde{U} = \{v_1, v_2, v_3, w_1, w_2, w_3\}$. We connect the nodes with the edges w_1v_1, w_1v_3, w_1, w_2, w_2v_1, w_2v_2, w_2w_3, w_3v_2 and w_3v_3.

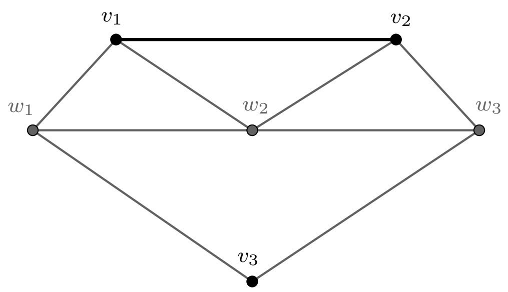


Figure 6.4: Three node cut set after modification.

Now we need a suitable inequality. We take

$$x_{v_1} + x_{v_2} + x_{v_3} + x_{w_1} + x_{w_2} + x_{w_3} \leq 2. \tag{6.14}$$

The following matrix contains the characteristic vectors of all stable sets in the modified cut set that fulfill (6.14), i.e $\{\chi^{\tilde{S}}_{\tilde{U}} \mid a\chi^{\tilde{S}} = \alpha \ , \ \tilde{S} \text{ stable in } \tilde{U}\}$.

$$\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ w_1 \\ w_2 \\ w_3 \end{array} \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \tag{6.15}$$

Now let's check if theorem 6.2 is applicable. We see immediately that the support of (6.14) is in the modified cut set and that (6.14) is valid and therefore face defining on $P(\tilde{G})$. Furthermore, the first three rows of (6.15) show that we have a stable set $\tilde{S}$, $a\chi^{\tilde{S}} = \alpha$ for every stable set S of (6.13) such that $\tilde{S} \cap U = S$. The determinant of (6.15) is $-2 \neq 0$, so the set $\{\chi^{\tilde{S}}_{\tilde{U}} \mid a\chi^{\tilde{S}} = \alpha \ , \ \tilde{S} \text{ stable in } \tilde{U}\}$ is linearly independent. All conditions of theorem 6.2 are fulfilled, so we found an example where it is possible to obtain the minimum number of added nodes.

The minimum number of added nodes is not always achievable

Now we will see an example where the minimum number of added nodes is not achievable. Let $G = (V, E)$ be again decomposable into $G_1 = (V_2, E_2)$ and $G_2 = (V_2, E_2)$. We stay in the three node cut set case, i.e. $U = V_1 \cap V_2 = \{v_1, v_2, v_3\}$, but this time with no edges between the nodes. There are eight possible stable sets within this cut set:

$$\{v_1, v_2, v_3\},\ \{v_1, v_2\},\ \{v_1, v_3\},\ \{v_2, v_3\},\ \{v_1\},\ \{v_2\},\ \{v_3\},\ \emptyset \tag{6.16}$$

So the minimum number of added nodes is five, i.e. $\tilde{U} = \{v_1, v_2, v_3, w_1, w_2, w_3, w_4, w_5\}$. In order to apply theorem 6.2, there has to be a possibility to add edges between the nodes of $\tilde{U}$ such that the resulting graph $\tilde{G}_{\tilde{U}} = (\tilde{U}, E_{\tilde{U}})$ has the following properties:

- The nodes v_1, v_2 and v_3 are pairwise disconnected.
- There exists a face defining inequality $ax \leq \alpha$ of $P(\tilde{G}_{\tilde{U}})$ such that
  - $\{\chi^{\tilde{S}} \mid a\chi^{\tilde{S}} = \alpha \ , \ \tilde{S} \text{ stable in } \tilde{U}\}$ is linearly independent.
  - for all stable sets $S \in U$ exists a stable set $\tilde{S} \in \tilde{U}$ with $a\chi^{\tilde{S}} = \alpha$ such that $S \cap \tilde{S} = S$.

It suffices to look at $\tilde{G}_{\tilde{U}}$, since a face defining inequality of $P(\tilde{G}_{\tilde{U}})$ also defines a face of $P(\tilde{G})$. Furthermore, the required linear independence only affects the characteristic vectors of stable sets within $\tilde{U}$, so it doesn't matter if we look at $P(\tilde{G}_{\tilde{U}})$ or $P(\tilde{G})$. Also for our last assumption it does not make a difference: all stable sets $\tilde{S}$ of $\tilde{G}_{\tilde{U}}$ are also stable in $\tilde{G}$ and since the support of our inequality lies in $\tilde{U}$ we have

$$a\chi^{\tilde{S}} = \alpha,\ \chi^{\tilde{S}} \in \{0,1\}^{|\tilde{U}|} \Leftrightarrow a\chi^{\tilde{S}} = \alpha,\ \chi^{\tilde{S}} \in \{0,1\}^{|\tilde{V}|}. \tag{6.17}$$

So if the characteristic vector of a stable set in U has a preimage in a face of $P(\tilde{G}_{\tilde{U}})$ it will also have a preimage in the corresponding face of $P(\tilde{G})$.

Remark 6.5. *Note that we need eight linearly independent characteristic vectors of stable sets in our face. Linearly independent vectors are also affinely independent. A face is a facet of a stable set polytope if and only if it contains $|V|$ affinely independent characteristic stable set vectors [8]. Since in our case $|V| = |\tilde{U}| = 8$, the wanted face has to be a facet.*

What we will do is looking at all possible graphs on the eight nodes and checking if they fulfill the conditions. We will implement this in Python. The code can be found in A.1. The basis is formed by a list of all different undirected graphs on eight nodes (up to isomorphisms), where every graph is given by the upper diagonal part of its adjacency matrix[1]. In the following we go through the procedure that is applied to each graph.

1. We will follow the procedure with the example of a graph having the following adjacency matrix. We will call the eight nodes $u_1, \dots, u_8$ since we cannot know in the beginning which three nodes will correspond to v_1, v_2 and v_3.

$$\begin{array}{cc} & \begin{matrix} u_1 & u_2 & u_3 & u_4 & u_5 & u_6 & u_7 & u_8 \end{matrix} \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

2. In the first step we check each graph to see if it has a stable set of size three. This is needed since we need three nodes in the graph which correspond to the disconnected nodes v_1, v_2 and v_3. In our example graph this condition is fulfilled. We see that u_1, u_2 and u_3 are disconnected.

$$\begin{array}{cc} & \begin{matrix} u_1 & u_2 & u_3 & u_4 & u_5 & u_6 & u_7 & u_8 \end{matrix} \\ \begin{matrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

[1] https://people.cs.umass.edu/~barring/graphs/eightgraphs (16.08.2019)

3. Now we generate the characteristic vectors of all possible stable sets of the graph. This is done by looking at every subset of nodes of the graph and checking if it forms a stable set. The characteristic vectors of all stable sets in our example are the columns of the following matrix. We denote the stable sets by $S_1, \ldots, S_{19}$.

$$
\begin{array}{c}
 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8
\end{array}
\begin{array}{c}
\begin{array}{ccccccccccccccccccc}
S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} & S_{11} & S_{12} & S_{13} & S_{14} & S_{15} & S_{16} & S_{17} & S_{18} & S_{19}
\end{array} \\
\begin{pmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0
\end{pmatrix}
\end{array}
$$

4. Note from remark 6.5 that our wanted face is a facet. In order to get all facet defining inequalities of a graph we use a tool called cddlib[2]. This tool enumerates all facet defining inequalities of a convex hull of vectors. From step 3. we know the vectors whose convex hull gives us the stable set polytope, so the tool is applicable to our case. For our example graph we get the following facet defining inequalities:

$$
\begin{aligned}
1x_1 + 1x_3 + 1x_4 + 1x_5 + 1x_6 + 1x_7 + 1x_8 &\leq 2 \\
1x_1 + 1x_2 + 1x_4 + 1x_5 + 1x_6 + 1x_7 + 1x_8 &\leq 2 \\
2x_3 + 1x_4 + 1x_5 + 1x_6 + 1x_7 + 1x_8 &\leq 2 \\
2x_2 + 1x_4 + 1x_5 + 1x_6 + 1x_7 + 1x_8 &\leq 2 \\
1x_1 + 1x_6 + 1x_7 &\leq 1 \\
&\vdots \\
1x_3 + 1x_6 + 1x_7 &\leq 1 \\
x_i &\geq 0 \quad \text{for } i = 1, 2, 3, 4, 5, 6, 7, 8
\end{aligned}
$$

5. Now we look at every facet defining inequality and check which characteristic vectors lie in the defined facet, i.e. we check which characteristic vectors fulfill the inequality with equality. Since the empty stable set does not play a role, we delete its characteristic vector out of the list. Let's look at the first inequality that defines the stable set polytope of our example graph and check which characteristic vectors of the stable sets $S_1, \ldots, S_{19}$ fulfill it with equality.

[2] https://inf.ethz.ch/personal/fukudak/cdd_home/ (15.08.2019)

The result can be found in the columns of the following matrix.

$$\begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array} \begin{array}{c} \begin{array}{cccccccc} S_{11} & S_{12} & S_{14} & S_{15} & S_{16} & S_{17} & S_{18} & S_{19} \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

6. The next step is to verify the linear independence of the characteristic vectors of non-empty stable sets that fulfill an inequality with equality. We first check if the number of such vectors equals eight. If there are more such vectors they cannot be linearly independent. If there are less it is not possible that every stable set within v_1, v_2 and v_3 has a preimage. Then we check the determinant of the matrix which has these vectors as columns. In our example we have eight such vectors and the determinant of the matrix is $-2 \neq 0$. The facet defined of the first inequality is therefore a candidate for an extension in sense of theorem 6.2.

7. Lastly we have to check if every stable set within v_1, v_2 and v_3 has a preimage in this facet. For this matter we look at all stable sets of size three of a graph. These are candidates to correspond to v_1, v_2 and v_3. Looking back to step 2. we see that in our example the only stable set of size three is the set $\{u_1, u_2, u_3\}$. Let's look at the corresponding entries of the characteristic vectors in our considered facet.

$$\begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array} \begin{array}{c} \begin{array}{cccccccc} S_{11} & S_{12} & S_{14} & S_{15} & S_{16} & S_{17} & S_{18} & S_{19} \end{array} \\ \begin{pmatrix} \boxed{\begin{array}{cccccccc} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}} \\ \begin{array}{cccccccc} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{array} \end{pmatrix} \end{array}$$

Obviously not every stable set of (6.16) has its characteristic vector within these entries. Therefore, we can conclude that this facet is not suitable in the sense of theorem 6.2.

We apply this procedure to all possible graphs and come to the conclusion that none of the graphs has a suitable facet. So in this case we cannot achieve the minimum number of added nodes which means that it is not always possible to obtain this minimum number.

6.3 Two node Cut Sets

6.3.1 General Case

Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. We assume that $U = V_1 \cap V_2 = \{v_1, v_2\}$ and that v_1 and v_2 are not connected. In this chapter we look at an approach of Barahona and Mahjoub [3], who modified the graph such that theorem 6.2 is applicable. In this special case they even found an inequality description of $P(G)$ in the original space. The modification is the following:

- $\tilde{V}_k = V_k \cup \{w_1, w_2, w_3\} \;\; k = 1, 2$
- $\tilde{E}_k = E_k \cup \{v_1w_1, v_2w_1, v_1w_2, w_2w_3, w_3v_2\} \;\; k = 1, 2$
- $\tilde{G}_1 = (\tilde{V}_1, \tilde{E}_1)$, $\tilde{G}_2 = (\tilde{V}_2, \tilde{E}_2)$ and $\tilde{G} = (\tilde{V}_1 \cup \tilde{V}_2, \tilde{E}_1 \cup \tilde{E}_2)$

What we did is completing the nodes v_1 and v_2 to a cycle of length five. We call the cycle C^*.

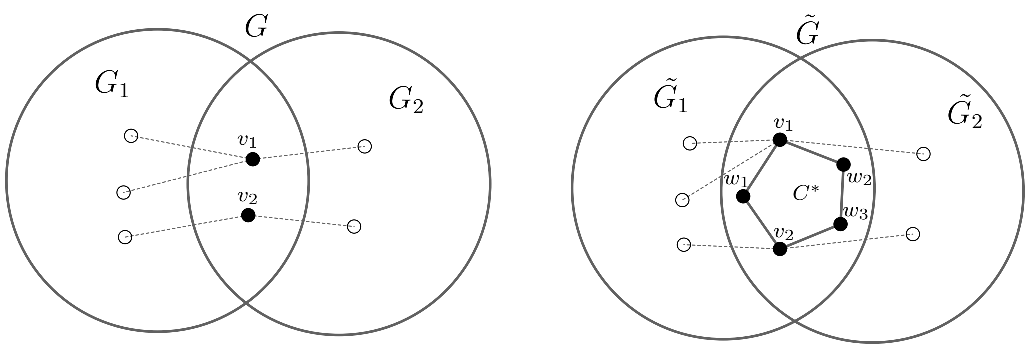


Figure 6.5: A graph G which decomposes into G_1 and G_2 on a cut set that consists of two disconnected nodes before and after the modification of adding the cycle of length five.

In order to use theorem 6.2 we need an inequality. Since at most two nodes in C can be together in a stable set, the inequality

$$\sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} \leq 2 \tag{6.18}$$

is valid. It defines a face of the polytopes $P(\tilde{G}_1)$, $P(\tilde{G}_2)$ and $P(\tilde{G})$. Let us check if theorem 6.2 is applicable. The added nodes are only connected to the nodes of the cut set and the support of the inequality lies in $\tilde{U}$. That means that we only have to check the following two conditions.

- For all stable sets $S \in U$ there exists a stable set $\tilde{S} \in C^*$ with $a\chi^{\tilde{S}} = \alpha$ such that $U \cap \tilde{S} = S$
- The set $\{\chi^{\tilde{S}}_{C^*} \mid a\chi^{\tilde{S}} = \alpha \;,\; \tilde{S} \text{ stable in } C^*\}$ is linearly independent

Let's look at the matrix that has $\{\chi_{C^*}^{\tilde{S}} \mid a\chi^{\tilde{S}} = \alpha\ ,\ \tilde{S} \text{ stable in } C^*\}$ as columns.

$$M = \begin{array}{cc} \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix} & \begin{array}{c} v_1 \\ v_2 \\ w_1 \\ w_2 \\ w_3 \end{array} \end{array} \tag{6.19}$$

Since $det(M) = 2 \neq 0$, we have the required linear independence. Furthermore we can read from the first to rows of M that every stable set within v_1 and v_2 has a preimage in the facet. Theorem 6.2 is therefore applicable and we can formulate the following corollary:

Corollary 6.6. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. We assume that $U = V_1 \cap V_2 = \{v_1, v_2\}$ and that v_1 and v_2 are not connected. Let $\tilde{G}$, $\tilde{G}_1$ and $\tilde{G}_2$ be defined like above. Then*

$$F(\tilde{G}) = \{x \in P(\tilde{G}) \mid \sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2\} \tag{6.20}$$

provides an extension of $P(G)$. Furthermore, we have

$$x \in F(\tilde{G}) \Leftrightarrow x_{\tilde{G}_1} \in F(\tilde{G}_1) \text{ and } x_{\tilde{G}_2} \in F(\tilde{G}_2), \tag{6.21}$$

where

$$F(\tilde{G}_1) = \{x \in P(\tilde{G}_1) \mid \sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2\}$$

and

$$F(\tilde{G}_2) = \{x \in P(\tilde{G}_2) \mid \sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2\}.$$

Barahona and Mahjoub found a description of $P(G)$ in the original space. They start with the inequality descriptions of $P(\tilde{G}_1)$ and $P(\tilde{G}_2)$. To get an idea of the structure of these inequalities we take the following lemmas into account. The proofs can be found in [8]. We denote the subgraph of G that is induced by the support of an inequality $ax \leq \alpha$, by G_a.

Lemma 6.7. *If G_a contains a path with vertices p,u,v,q, where u and v are of degree 2, then $a_u = a_v$.*

Lemma 6.8. *Let G_a be different from an odd cycle (and from K_3) and let u and v be two given nodes. Then G_a does not contain two edge-disjoint paths between u and v such that each node, except u and v, has degree 2.*

Lemma 6.9. *If $ax \leq \alpha$ is not of the form $x_u + x_v \leq 1$ or $0 \leq x_u \leq 1$, then G_a does not contain a node of degree 1.*

So let's apply these lemmata on our case. The goal is to cluster the inequalities that define $P(\tilde{G}_k)$, $k = 1, 2$, according to the occurrence of w_1, w_2 or w_3. First we have the case that w_1, w_2 and w_3 are not in the support of an inequality. We denote these inequalities by the index sets I_1^1 for $\tilde{G}_1$ and I_1^2 for $\tilde{G}_2$. Note that the only inequality where the support only contains nodes of $\{w_1, w_2, w_3\}$ and none of the nodes $\{v_1, v_2\}$ is the edge-constraint for the edge $w_2 w_3$. Furthermore, there cannot be an inequality support (except edge-constraints) where only one node of $\{v_1, v_2\}$ together with nodes of $\{w_1, w_2, w_3\}$ is contained, since this would contradict lemma 6.9. So let's look at the different cases in which v_1 and v_2 together with w_1, w_2 or w_3 and the other nodes of $\tilde{V}_k$, $k = 1, 2$ are in the support of an inequality:

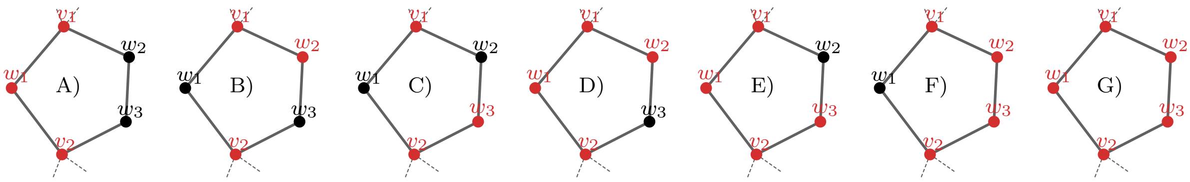


Figure 6.6: Different combinations of w_1, w_2 and w_3 in the support of a defining inequality of $P(\tilde{G}_1)$ respectively $P(\tilde{G}_2)$. ●: in the support, ●: not in the support

In the cases B), C), D) and E) we end up with nodes which have degree 1, so these combinations are not possible for our inequalities according to lemma 6.9. In case A) we don't have this problem, so inequalities whose support intersects $\{v_1, v_2, w_1\}$ but not $\{w_2, w_3\}$ will be taken into account. The corresponding index set will be called I_2^k, $k = 1, 2$. The same holds for case F). Here we just have to be careful with the coefficients of w_2 and w_3. According to lemma 6.7 they have to be equal. We denote the indexset by I_3^k, $k = 1, 2$. In case G) we have to be careful. If the support of an inequality contains more than these five vertices, lemma 6.8 applies and constellation G) would not be possible. But if the support of an inequality only contains $\{v_1, v_2, w_1, w_2, w_3\}$, lemma 6.8 does not apply since we have an odd cycle, so such an inequality would be possible. Again, lemma 6.7 applies and v_1, v_2, w_1, w_2 and w_3 must have the same coefficients. With this achievement we can cluster the inequalities that define $P(\tilde{G}_k)$, $k = 1, 2$ as follows:

a) $\sum_{v \in V_k} a_{iv}^k x_v \leq \alpha_i^k \quad i \in I_1^k$

b) $\sum_{v \in V_k} a_{iv}^k x_v + x_{w_1} \leq \alpha_i^k \quad i \in I_2^k$

c) $\sum_{v \in V_k} a_{iv}^k x_v + x_{w_2} + x_{w_3} \leq \alpha_i^k \quad i \in I_3^k$

d) $x_{v_1} + x_{w_1} \leq 1$

e) $x_{v_1} + x_{w_2} \leq 1$

f) $x_{v_2} + x_{w_1} \leq 1$

g) $x_{v_2} + x_{w_3} \leq 1$

h) $x_{w_2} + x_{w_3} \leq 1$

i) $x_{v_1} + x_{v_2} + x_{w_1} + x_{w_2} + x_{w_3} \leq 2$

j) $x_v \geq 0, \quad v \in \tilde{V}_k$

If we add the inequalitiy

k) $-x_{v_1} - x_{v_2} - x_{w_1} - x_{w_2} - x_{w_3} \leq -2,$

we have a description for $F(\tilde{G}_k)$, $k = 1, 2$. Now we can use corollary 6.6 and obtain an inequality description of $F(\tilde{G})$ by uniting both systems. We need to find a way to get from the inequality description of $F(\tilde{G})$ to the description of $P(G)$. Therefore we use the following theorem of Balas and Pulleybank [9].

Theorem 6.10 (Balas and Pulleybank)**.** *Let $Z = \{(x, y) \mid Ax + By \leq b,\ x \geq 0, y \geq 0\}$. Then the projection along the y variables is given by*

$$X = \{x \mid (vA)x \leq vb, \forall v \in extr\,\psi,\ x \geq 0\} \tag{6.22}$$

where $extr\,\psi$ denotes the set of extreme rays of

$$\psi = \{y \mid yB \geq 0, y \geq 0\} \tag{6.23}$$

In our case the y-variables are the variables x_{w_i}, $i = 1, 2, 3$ and B is the matrix of coefficients of these variables. Therefore, it is given by

$$B = \begin{array}{c} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ -1 & -1 & -1 \end{pmatrix} \end{array} \begin{array}{l} a) \\ b) \\ c) \\ d) \\ e) \\ f) \\ g) \\ h) \\ i) \\ k) \end{array} . \tag{6.24}$$

The letters on the right show from which type of inequality the coefficients come from. Since some of the rows in B are redundant we work with

$$\bar{B} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ -1 & -1 & -1 \end{pmatrix} \begin{array}{l} a) \\ b), d) \text{ or } f) \\ c) \text{ or } h) \\ e) \\ g) \\ i) \\ k) \end{array} . \tag{6.25}$$

Instead of looking at the extreme rays we will look at the extreme points of

$$\bar{\psi} = \{z \mid z\bar{B} \geq 0, \sum_{i=1}^{7} z_i = 1; z \geq 0\}. \tag{6.26}$$

This step is illustrated in the following picture.

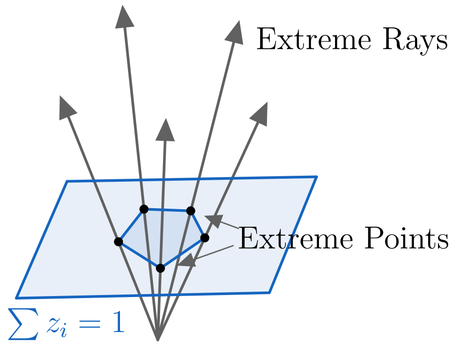


Figure 6.7: Extreme points belonging to extreme rays in $\mathbb{R}^3$.

These extreme points can be enumerated (A.2) and are the columns of the following matrix:

$$\left(\underbrace{\begin{array}{c|c|c|c|c|c|} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}}_{1)} \underbrace{\begin{array}{c|} 0 \\ \frac{1}{3} \\ \frac{1}{3} \\ 0 \\ 0 \\ 0 \\ \frac{1}{3} \end{array}}_{2)} \underbrace{\begin{array}{c|} 0 \\ \frac{1}{4} \\ 0 \\ \frac{1}{4} \\ \frac{1}{4} \\ 0 \\ \frac{1}{4} \end{array}}_{3)} \underbrace{\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{2} \\ \frac{1}{2} \end{array}}_{4)} \right) \tag{6.27}$$

Now we can use Theorem 6.10 to obtain the inequalities of $P(G)$. The theorem says that we have to multiply each extreme point with Ax. In our case Ax is the part of the inequalities a) - k) which belongs to the variables assigned to the nodes of V_1 and V_2. In other words, Ax are the inequalities a) - k) where the variables belonging to w_1, w_2 and w_3 were deleted. If we multiply e.g. the second column of (6.27) with our Ax, we get the inequalities of b) without the variable belonging to w_1. Applying this procedure to all columns gives us the following results. The single steps belong to the different kinds of extreme points, as already clustered in (6.27).

1) Keep the inequalities a) - f).

2) Take the sum of three inequalities, one of type b), d) or f), one of type c) or h) and the inequalitiy k).

3) Take the sum of four inequalities, one of type b), d) or f), one of type e), one of type g) and inequalitiy k).

4) Take the sum of the two inequalities i) and k).

Note that for step 2)-4) it does not matter whether we scale the inequalities or not, since such a scaling leads to an equivalent inequality. For example in step 2) we can neglect the factor $1/3$. Now we have a lot of inequalities which already form an inequality description of $P(G)$. To prove redundancy of some of them we use the following lemma.

Lemma 6.11. *Let $ax \leq \alpha$, $x \in \mathbb{R}^{|V_k|}$ be an inequality that defines a facet of $P(G)$. If $V_a \subseteq V_k$, then this inequality also defines a facet of $P(\tilde{G}_k)$, $k = 1, 2$.*

The proof can be found in [3]. Remember that all facet defining inequalities of $P(\tilde{G}_k)$ where the support only contains nodes of V_k, $k = 1, 2$ are collected in a). Lemma 6.11 says that a multiple of every facet defining inequality in the description of $P(G)$, where the support only contains nodes of V_k, $k = 1, 2$, is covered by the inequalities of a). So all other inequalities with such a support are redundant. Let's look at the support of the inequalities we obtained above and check if we have to keep them. Steps 1)-4) analyze the inequalities that were obtained in the corresponding steps above.

1) The support of b) - f), neglecting variables assigned to the nodes w_1, w_2 and w_3, is a subset of V_k. Therefore, we only keep the inequalities a) and neglect the inequalities b)-f).

2) Here, we only keep the inequalities which are a sum of inequalities of $P(\tilde{G}_1)$ **and** $P(\tilde{G}_2)$. The support of all other such inequalities lies in V_k, $k = 1, 2$.

3) Here, the only inequalities where the support contains nodes, which are not in $\{v_1, v_2\}$, are the inequalities of category b). Their support contains nodes of V_1 **or** V_2. A combination of an inequality of type b) with inequalities where the support only contains nodes of $\{v_1, v_2\}$ cannot have a support that does not only lie in V_k, $k = 1, 2$. So all inequalities obtained here a redundant.

4) This step leads to the equation $0 = 0$, so it is redundant.

What is left are the inequalities of a) together with the mixed inequalities obtained in step 2) that contain defining inequalities of $P(\tilde{G}_1)$ and $P(\tilde{G}_2)$. With this achievement we can formulate the theorem which gives us the wanted description of $P(G)$.

Theorem 6.12. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. We assume that $U = V_1 \cap V_2 = \{v_1, v_2\}$ and that v_1 and v_2 are not connected. Let $\tilde{G}$, $\tilde{G}_1$ and $\tilde{G}_2$ be defined like above. The stable set polytope $P(G)$ is defined by*

i. $\sum_{v \in V_k} a^k_{iv} x_v \leq \alpha^k_i \quad i \in I^k_1$ *(the inequalities a))*

ii. $\sum_{v \in V_k} a^k_{iv} x_v + \sum_{v \in V_l} a^l_{jv} x_v - x_{v_1} - x_{v_2} \leq \alpha^k_i + \alpha^l_j - 2$

for $k = 1, 2;\; l = 1, 2;\; k \neq l;\; i \in I^k_2;\; j \in I^k_3$ *(mixed inequalities from step 2)*

iii. $x_v \geq 0$ *for* $v \in V$

It has also been shown in [3] that this description is minimal.

6.3.2 Simplification in the bipartite Case

In this chapter we apply Barahona and Mahjoubs [3] approach to the special case where one of the two graphs is bipartite. We will see that the modification in this case can be simplified. So let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Let $V_1 \cap V_2 = \{v_1, v_2\}$. Furthermore let w.l.o.g. G_1 be bipartite and let v_1 and v_2 be on the same side of the partition of G_1.

Application of theorem 6.12

Let $\tilde{G}$, $\tilde{G}_1$ and $\tilde{G}_2$ be defined like in the beginning of chapter 6.3.1, i.e. with the added cycle of length five. Since G_1 is bipartite, we know that $P(G_1)$ is described only by edge and non-negativity constraints. But what about $P(\tilde{G}_1)$?

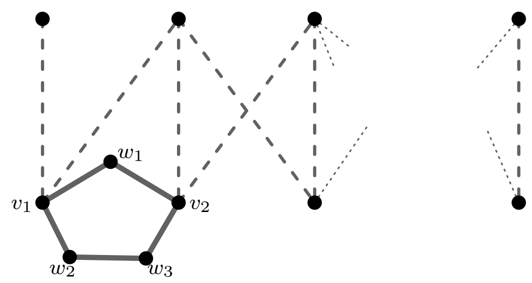


Figure 6.8: The bipartite graph G_1 where the cut set nodes are on the same side of the partition, after adding a cycle of length five.

Since v_1 and v_2 are on different sides of the partition, all v_1-v_2-paths have an odd number of nodes. So all odd cycles, which are caused by adding the cycle of length five, go trough w_2 and w_3. Every cycle that is closed by w_1 is even. We can conclude that $\tilde{G}_1 \setminus w_i$, $i = 2, 3$ is bipartite, so $\tilde{G}_1$ is almost bipartite and therefore t-perfect. This means that the stable set polytope $P(\tilde{G}_1)$ is described by the inequalities

$$x_v \geq 0 \qquad \forall\, v \in \tilde{V}_1 \tag{6.28}$$

$$x_u + x_v \leq 1 \qquad \forall\, uv \in \tilde{E}_1 \tag{6.29}$$

$$\sum_{v \in C} x_v \leq \frac{|C| - 1}{2} \qquad \forall \text{ C odd cycle of } \tilde{G}_1. \tag{6.30}$$

In order to use theorem 6.12, we have to cluster these inequalities into I_1^1, I_2^1 and I_3^1. We start with I_1^1, i.e. the inequalities whose support does not intersect with $\{w_1, w_2, w_3\}$. Here, we only have the edge inequalities for edges in E_1. Now we look at I_2^1. These are the inequalities whose support contains $\{v_1, v_2, w_1\}$ but not $\{w_2, w_3\}$. Since the only odd cycle that goes through w_1 is the added cycle itself, which is treated in a separate inequality, there are no such inequalities, so $I_2^1 = \emptyset$. The set I_3^1, which consists of inequalities whose support intersects $\{v_1, v_2, w_2, w_3\}$ but not $\{w_1\}$, is formed by the odd cycle inequalities without the one corresponding the added cycle. Now we can use theorem 6.12 and obtain the following inequality description for $P(G)$:

i. $x_u + x_v \leq 1 \, \forall \, uv \; in \; E_1$ (Inequalities of I_1^1)

$\sum_{v \in V_k} a_{iv}^k x_v \leq \alpha_i^k \quad i \in I_1^2$

ii. $\sum_{v \in C \cap V_1} x_v + \sum_{v \in V_2} a_{iv}^2 x_v - x_{v_1} - x_{v_2} \leq \frac{|C|-1}{2} + \alpha_i^2 - 2$

for $\quad i \in I_2^2, \;\; C$ odd cycle in $\tilde{G}_1, \; C \neq C^*$

iii. $x_v \geq 0$ for $v \in V$

We see that no defining inequality of $\tilde{G}_1$ whose support intersects $\{w_1\}$ is used. Furthermore no defining inequality of $\tilde{G}_2$ whose support intersects $\{w_2, w_3\}$ is used. This leads to the idea that it is maybe not necessary to add all three nodes to both components of the graphs.

The Simplification

So let's define

- $\hat{V}_1 = V_1 \cup \{w_2, w_3\}$

  $\hat{V}_2 = V_2 \cup \{w_1\}$
- $\hat{E}_1 = E_1 \cup \{v_1 w_2, w_2 w_3, w_3 v_2\}$

  $\hat{E}_2 = E_2 \cup \{v_1 w_1, w_1 v_2\}$
- $\hat{G}_1 = (\hat{V}_1, \hat{E}_1), \; \hat{G}_2 = (\hat{V}_2, \hat{E}_2)$.

The result can be seen in the following figure.

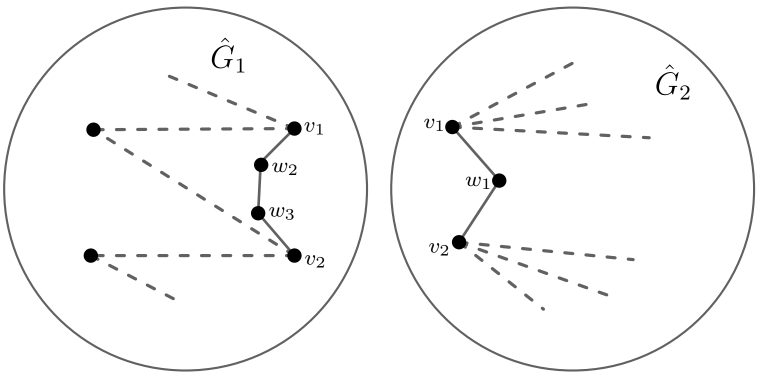


Figure 6.9: The bipartite graph G_1 after adding a path of length four and the graph G_2 after adding a path of length three.

We set up the following theorem:

Theorem 6.13. *Let $\hat{G}_1$ and $\hat{G}_2$ be defined like above. P(G) is the projection of*

$$\hat{F} = \left\{ x \in \mathbb{R}^{\hat{V}_1 \cup \hat{V}_2} : x_{\hat{G}_1} \in P(\hat{G}_1) \text{ and } x_{\hat{G}_2} \in P(\hat{G}_2), \; \sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2 \right\} \tag{6.31}$$

along the variables x_{w_1}, x_{w_2} and x_{w_3}.

Proof. We first have to show that every projected point of $\hat{F}$ lies in $P(G)$. As we already know the inequality description of $P(G)$, we just have to show that all the inequalities are valid for such a projected point. The inequalities in i., whose support does not intersect $\{w_1, w_2, w_3\}$, and the non-negativity constraints in iii. are obviously still valid for $P(\hat{G}_1)$ and $P(\hat{G}_2)$. As the support of these inequalities does not intersect $\{w_1, w_2, w_3\}$, they are also valid for every projected vector. Let's look at the inequalities of ii., i.e. the mixed inequalities. They are a sum of odd cycle inequalities of $P(\tilde{G}_1)$, inequalities from I_2^2 and the inequality $-x_{v_1} - x_{v_2} - x_{w_1} - x_{w_2} - x_{w_3} \leq 2$. As all odd cycles (except $C*$) that existed in $\tilde{G}_1$ still exist in $\hat{G}_1$, all required odd cycle inequalities are valid for $P(\hat{G}_1)$ and therefore also for $\hat{F}$. The inequalities of I_2^2 are valid for $P(\hat{G}_2)$, since their support does not contain w_2 or w_3. So these inequalities are also valid for $\hat{F}$. From the definition of the right hand side of (6.31) we also know that the inequality $-x_{v_1} - x_{v_2} - x_{w_1} - x_{w_2} - x_{w_3} \leq 2$ holds for $\hat{F}$. If we add up these three types of inequalities we end up with

$$\sum_{w \in C \cap V_1} x_w + \sum_{j \in V_2} a_{ij}^2 x_j - x_{v_1} - x_{v_2} \leq \frac{|C| - 1}{2} + \alpha_i^2 - 2 \tag{6.32}$$

for $i \in I_2^2$ and C an odd cycle in $\hat{G}_1$. The variables assigned to w_1, w_2 and w_3 cancel each other out. The result is exactly the mixed type of inequality that defines $P(G)$, so the mixed inequalities are also still valid for $\hat{F}$. As the support of this mixed inequalities does not intersect $\{w_1, w_2, w_3\}$, the inequalities are still valid for all projected vectors. Since we now showed that every projected vector of $\hat{F}$ fulfills all defining inequalities of $P(G)$, we can conclude that every such vector lies in $P(G)$. Now we show that every vertex of $P(G)$ has a preimage in $\hat{F}$. Let $x \in P(G)$ be a vertex of $P(G)$, i.e. a characteristic vector of a stable set in G. For each combination of v_1 and v_2 in such a stable set vector, one finds 0-1-values for x_{w_1}, x_{w_2} and x_{w_3} such that

$$\sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2$$

and the corresponding set of nodes in $\hat{G}_1$ and $\hat{G}_2$ is stable. So every characteristic vector of stable sets in G has a preimage in $\hat{F}$. □

Until now we only considered the case where the nodes v_1 and v_2 are on the same side of the partition. Let's look at the case where they are on different sides.

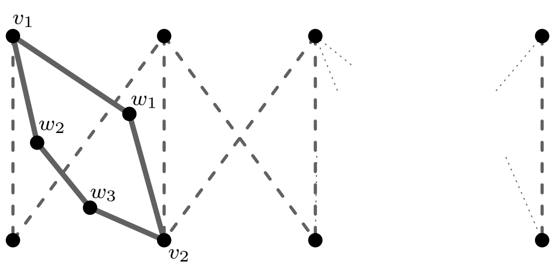


Figure 6.10: The bipartite graph G_1 where the cut set nodes are on different sides of the partition, after adding a cycle of length five.

As in this case all v_1-v_2-paths have an even number of vertices, all the odd cycles that occur after adding the cycle of length five are going trough w_1, which means that $\tilde{G}_1$ is again t-perfect. This time we recognize that in the description of $P(G)$ no defining inequality of $P(\tilde{G}_1)$ occurs where the support uses w_2 or w_3 and no defining inequality of $P(\tilde{G}_2)$ occurs which uses w_1. With these findings we can use the same argumentation as before and come to the following similar result.

Corollary 6.14. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_1, E_2)$. If we define*

- $\hat{V}_1 = V_1 \cup \{w_1\}$

  $\hat{V}_2 = V_2 \cup \{w_2, w_3\}$
- $\hat{E}_1 = E_1 \cup \{v_1w_1, w_1v_2\}$

  $\hat{E}_2 = E_2 \cup \{v_1w_2, w_2w_3, w_3v_2\}$
- $\hat{G}_1 = (\hat{V}_1, \hat{E}_1),\ \hat{G}_2 = (\hat{V}_2, \hat{E}_2),$

then (6.31) provides an extension of $P(G)$ in the case where v_1 and v_2 are on different sides of the partition.

The extension that is obtained by Barahona und Mahjoubs [3] , i.e.

$$F(G) = \Big\{x \in \mathbb{R}^{\tilde{V}_1 \cup \tilde{V}_2} : x_{\tilde{G}_1} \in P(\tilde{G}_1) \text{ and } x_{\tilde{G}_2} \in P(\tilde{G}_2),\ \sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2\Big\}, \quad (6.33)$$

is integer, since it is a face of a stable set polytope. We will show that this is not the case for our new set

$$\hat{F} = \Big\{x \in \mathbb{R}^{\hat{V}_1 \cup \hat{V}_2} : x_{\hat{G}_1} \in P(\hat{G}_1) \text{ and } x_{\hat{G}_2} \in P(\hat{G}_2),\ \sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2\Big\}. \quad (6.34)$$

In order to show this we state an example. Consider two graphs G_1 and G_2 where both are paths of length three. The connection of these two graphs are their endnodes. Clearly G_1 is bipartite. Now we define $\hat{G}_1$ and $\hat{G}_2$ as before and end up like in figure 6.11.

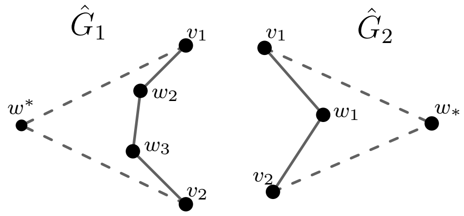


Figure 6.11: The modified versions $\hat{G}_1$ and $\hat{G}_2$ of G_1 and G_2.

We see that

$$\begin{array}{c} w^* \\ v_1 \\ w_2 \\ w_3 \\ v_2 \end{array} \begin{pmatrix} 0 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix} \in P(\hat{G}_1), \quad \begin{array}{c} v_1 \\ w_1 \\ v_2 \\ w_* \end{array} \begin{pmatrix} 1/2 \\ 0 \\ 1/2 \\ 1/2 \end{pmatrix} \in P(\hat{G}_2). \tag{6.35}$$

Since these two vectors together fulfill $\sum_{i=1}^{3} x_{w_i} + x_{v_1} + x_{v_2} = 2$, we have

$$\begin{array}{c} w^* \\ v_1 \\ v_2 \\ w_1 \\ w_2 \\ w_3 \\ w_* \end{array} \begin{pmatrix} 0 \\ 1/2 \\ 1/2 \\ 0 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix} \in \hat{F}. \tag{6.36}$$

If $\hat{F}$ was integer, all its vertices would have to be characteristic vectors of stable sets. Now we optimize over $\hat{F}$ in the direction of $c = (0, 1, 1, 0, 1, 1, 1)$. If $\hat{F}$ was integer, the maximum value of this optimization has to be 2, since a combination of a characteristic vector of a stable set in $\hat{G}_1$ with one of $\hat{G}_2$ cannot have a higher value. If we multiply c with (6.36) we get a value of 2.5. Therefore, our polytope has to be non-integer.

6.4 Extension complexity of modified graphs

Theorem 6.2 gives us the following information about the extension complexity of $P(G)$.

Corollary 6.15. *Let $G = (V, E)$ be decomposable into $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ and let $U = V_1 \cap V_2$. Let $\tilde{G}$, $\tilde{G}_1$ and $\tilde{G}_2$ be modified versions of G, G_1 and G_2 such that theorem 6.2 is applicable. Then we have*

$$xc(P(G)) \leq xc(F(\tilde{G})) \leq xc(F(\tilde{G}_1)) + xc(F(\tilde{G}_2)) \tag{6.37}$$

Now there comes up the question if we can improve the bound from (2.26). Let's see how we can bound $F(\tilde{G}_i)$ in terms of $P(G_i)$, $i = 1, 2$. We know that the nodes that were added to obtain $\tilde{G}_i$ are only connected to the nodes of U. Therefore we can describe $P(\tilde{G}_i)$ by

$$F(\tilde{G}_i) = \operatorname{conv} \bigcup_{S \in \tilde{U}} \left\{ x \in \mathbb{R}^{\tilde{V}_i} : \; x_{G_i} \in P(G_i), \; x_{\tilde{U}} = \chi^S \right\}, \tag{6.38}$$

where $\chi^S \in \{0, 1\}^{\tilde{U}}$ denote all characteristic vectors of stable sets $S \in \tilde{U}$ that fulfill the face defining inequality of $F(\tilde{G}_i)$ with equality. We can conclude that

$$xc(F(\tilde{G}_k)) \leq s' \cdot xc(P(G_k)), \; k = 1, 2 \tag{6.39}$$

where s' denotes the number of stable sets in $\tilde{U}$ that fulfill the face defining inequality with equality. Therefore,

$$xc(P(G)) \leq s' \left(xc(P(G_1)) + xc(P(G_2)) \right). \tag{6.40}$$

We know from theorem 6.4 that we can find a modification for every kind of cut set such that $s' = s$. Remember that s denotes the number of stable sets in U. Since in this case (6.40) equals (2.26), we cannot improve this bound. The results of this chapter can still be helpful, namely in the case when $\tilde{G}_1$ and $\tilde{G}_2$ have a nice structure, i.e. when we can still find a small inequality description of them. In this case we can easily find an extended formulation of $P(G)$ and in the case where we have a two node cut set with disconnected nodes even an inequality description of $P(G)$.

7 Conclusion

Aim of this thesis was to obtain a description of the stable set polytope of a decomposable graph G that bases on the stable set polytopes of the single components G_1 and G_2 of the graph. Let's summarize our findings.

We have two cases where we found an inequality description of $P(G)$ in the original space. One of these is the case where the cut set forms a clique. Here, the inequality description of $P(G)$ is obtained by the union of the inequality descriptions of $P(G_1)$ and $P(G_2)$. In the case of a cut set which consists of two disconnected nodes we also saw an inequality description of the original polytope. Here, we had to go an indirect way which led over extended formulations. The original graph was modified and a face of the stable set polytope of this modified graph provided an extension of $P(G)$. Then the inequality description of $P(G)$ in the original space was obtained by a combination of the inequalities that define this face. The whole procedure is much more difficult than in the clique cut set case. This is not surprising. If there was a way to solve this case similar to the clique cut set case, then one could solve the stable set problem efficiently on a graph which is recursively decomposable into simpler graphs on such two node cut sets. We showed that this is not the case by reducing the $\mathcal{NP}$-hard MAX-2-SAT problem to the weighted stable set problem on such a graph. That also implies that we cannot expect too much if we look for methods that work for any kind of cut sets.

We found techniques to find extended formulations of arbitrary decomposable graphs, but we did not find a description in the original space. One of the approaches we looked at was clique lifting. Here, the nodes of the cut set were replaced by a clique of other nodes. We could then apply the results that we observed for clique cut sets on this modified graph. The obtained stable set polytope then provided an extension of $P(G)$. This method adds edges between the nodes of the modified cut set and the other nodes of the graph. Therefore, we looked for a method that only affects the cut set of G and found a suitable technique. It is a generalization of Barahona and Mahjoub's [3] idea to get a face as an extension. After analyzing the required properties of such a face, we introduced a way to modify an arbitrary cut set in such a way that there exists such a face. We also constructed the inequality that defines this face. With this construction we can conclude the same things as in the approach of Barahona and Mahjoub: the face provides an extension of $P(G)$ and the inequality description of that extension is obtained by the union of the inequality descriptions of the corresponding faces of the stable set polytopes of the two parts of G.

The problem of the studied techniques for arbitrary cut sets is the necessity of a modification of a graph. If the stable set polytopes of G_1 and G_2 can be easily described, it could happen that the modifications change the graph structure of G_1 and G_2 in such a way that it is hard to find the required inequality description. Nevertheless, if the stable set polytopes of the modified versions of G_1 and G_2 are still easy to describe, one gets an extended formulation of $P(G)$ very fast. Therefore, it would be helpful to find ways to obtain the stable set polytope of a modification that base on the original descriptions.

Another goal of this thesis was to look at the dependence of the extension complexity of $P(G)$ on the extension complexities of $P(G_1)$ and $P(G_2)$. In the case of a clique cut set the extension complexity of $P(G)$ could be bounded by the sum of the extension complexities of $P(G_1)$ and $P(G_2)$. So in this case, a small extension complexity of $P(G_1)$ and $P(G_2)$

leads to a small extension complexity of $P(G)$. For arbitrary cut sets we did not find such a small bound in all cases. Here, it depends on the number of stable sets in the single cut set. The extension complexity of $P(G)$ could be bounded by this number times the sum of the extension complexities of $P(G_1)$ and $P(G_2)$. So if this sum and the number of stable sets in the cut set are small, the extension complexity of $P(G)$ is also small.

List of Figures

References

[1] Vašek Chvátal. On certain polytopes associated with graphs. *Journal of Combinatorial Theory, Series B*, 18(2):138–154, 1975.

[2] Michele Conforti, Bert Gerards, and Kanstantsin Pashkovich. Stable sets and graphs with no even holes. *Mathematical Programming*, 153(1):13–39, 2015.

[3] Francisco Barahona and Ali Ridha Mahjoub. Compositions of graphs and polyhedra ii: stable sets. *SIAM Journal on Discrete Mathematics*, 7(3):359–371, 1994.

[4] Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2. Springer Science & Business Media, 2012.

[5] Stefan Weltge. Sizes of linear descriptions in combinatorial optimization. 2015.

[6] Egon Balas. *Disjunctive Programming and Extended Formulations*, pages 69–77. Springer International Publishing, Cham, 2018.

[7] Korte Bernhard and J Vygen. Combinatorial optimization: Theory and algorithms. *Springer, Third Edition, 2005.*, 2008.

[8] Ali Ridha Mahjoub. On the stable set polytope of a series-parallel graph. *Mathematical Programming*, 40(1):53–57, 1988.

[9] Egon Balas and William Pulleyblank. The perfectly matchable subgraph polytope of a bipartite graph. *Networks*, 13(4):495–516, 1983.

A Appendix

A.1 Code Section 6.2

```
import itertools, os, numpy

#
# helper functions to obtain vertex and inequality descriptions of the
    stable set polytopes of all considered graphs
#

def read_graphs():
    file = open("8_nodes.txt", "r")
    graphs = [[[int(a) for a in c] for c in line.split(": ")[1].strip
        ().split(" ")] for line in file]
    file.close()
    return graphs

def is_stable_set(g, nodes):
    for i in range(len(nodes)):
       for j in range(i + 1, len(nodes)):
           if g[nodes[j] - 1][nodes[i]] == 1:
               return False
    return True

def has_stable_set_of_size_three(g):
    n = len(g)
    for i in range(n):
        for j in range(i + 1, n):
            for k in range(j + 1, n):
                if is_stable_set(g, [i, j, k]):
                    return True
    return False

def get_graphs_with_stable_set_of_size_at_least_three(graphs):
    return [g for g in graphs if has_stable_set_of_size_three(g)]


def get_all_stable_sets(g):
    items = range(len(g) + 1)
    powerset = [x for length in range(len(items) + 1) for x in
        itertools.combinations(items, length)]
    stable_sets = []
    for s in powerset:
        if is_stable_set(g,s) == True:
            stable_set = [0, 0, 0, 0, 0, 0, 0, 0]
            for i in range(len(s)):
                stable_set[s[i]] = 1
            stable_sets.append(stable_set)
    return stable_sets

def get_stable_sets_of_graphs(graphs):
    return [get_all_stable_sets(g) for g in graphs]

```

```
def write_v_representations(stable_sets_of_graphs):
    for i in range(len(stable_sets_of_graphs)):
        stable_sets = stable_sets_of_graphs[i]
        m = len(stable_sets)
        f = open('Polytopes/graph' + str(i) + '.ext', 'w')
        f.write('V-representation \nbegin \n' + str(m) + '  ' + str(len(stable_sets[0]) + 1) + '  ' + 'integer\n')
        for j in range(m):
           f.write('  1')
           for k in range(len(stable_sets[0])):
               f.write('  ' + str(stable_sets[j][k]))
           f.write('\n')
        f.write('end')
        f.close()

def create_h_representations():
    for filename in os.listdir("Polytopes"):
        if filename.endswith(".ext"):
            basename = filename.split(".")[0]
            os.system("cat Polytopes/" + basename + ".ext | ./cddexec --rep > Polytopes/" + basename + ".ine 2> /dev/null")

#
# the actual program to obtain vertex and inequality descripton of the stable set polytopes of all considered graphs
#

graphs = read_graphs()
graphs = get_graphs_with_stable_set_of_size_at_least_three(graphs)
print(graphs[11890])
stable_sets_of_graphs = get_stable_sets_of_graphs(graphs)
print(stable_sets_of_graphs[11890])
#write_v_representations(stable_sets_of_graphs)
#create_h_representations()

#
# part which uses vertex and inequality descriptions to check if there exists a facet with the wanted features
#

#Getting for every facet defining inequality of the stable set polytope of a graph all characteristic stable set vectors which fulfill the facet defining inequality with equaliy

vectors_in_facets = [] #saves for every graph and all of it's inequalities all characteristic stable set vectors that fulfill the inequality with equality
for p in range(len(graphs)):
    f = open('Polytopes/graph' + str(p) + '.ine', 'r') #opens the inequality description files
    inequalities = [[a for a in line.strip().split(" ")] for line in f]
    corrected_inequalities = [] #saves the coefficients of the facet defining inequalities
    for i in range(4, len(inequalities)-1):
```

```
        corrected_inequalities.append([])
        for j in range(len(inequalities[i])):
            if inequalities[i][j] != '':
                corrected_inequalities[i-4].append(int(inequalities[i][j]))
    satisfy_equalitiy = [] #saves the vectors that fulfill an inequality with equality for one graph
    for i in range(len(corrected_inequalities)):
        satisfy_equalitiy.append([])
        for j in range(len(stable_sets_of_graphs[p])):
            eingesetzt_in_inequalities = sum(-corrected_inequalities[i][k] * stable_sets_of_graphs[p][j][k - 1] for k in range(1, len(corrected_inequalities[i])))
            if eingesetzt_in_inequalities == corrected_inequalities[i][0]:
                satisfy_equalitiy[i].append(stable_sets_of_graphs[p][j])
    vectors_in_facets.append(satisfy_equalitiy)
f.close()

print(vectors_in_facets[11890])

#Since the zero vector doesn't play a role when linear independecy is checkt we delete all of them from the list

vectors_in_facets_without_zerovector = []  #save the vectors which fulfill the inequalities with equality without the zerovector
for p in range(len(graphs)):
    vectors_in_facet_without_zerovector_per_graph = []
    for i in range(len(vectors_in_facets[p])):
        vectors_in_facet_without_zerovector = []
        for j in range(len(vectors_in_facets[p][i])):
            if vectors_in_facets[p][i][j] != [0, 0, 0, 0, 0, 0, 0, 0]:
                vectors_in_facet_without_zerovector.append(vectors_in_facets[p][i][j])
        vectors_in_facet_without_zerovector_per_graph.append(vectors_in_facet_without_zerovector)
    vectors_in_facets_without_zerovector.append(vectors_in_facet_without_zerovector_per_graph)

print(vectors_in_facets_without_zerovector[11890])

#The next part checks if there is an inequality where the corresponding vectors fulfill all assumtions

items = range(8)
powerset = [x for length in range(len(items) + 1) for x in itertools.combinations(items, length)]
needed_stable_sets = [[0, 0, 0], [0, 0, 1], [0, 1, 0], [0, 1, 1], [1, 0, 0], [1, 0, 1], [1, 1, 0], [1, 1, 1]] #preimages of all stable sets in the cut set
for p in range(len(graphs)):
    for i in range(len(vectors_in_facets_without_zerovector[p])):
        if len(vectors_in_facets_without_zerovector[p][i]) == 8 and numpy.linalg.det(vectors_in_facets_without_zerovector[p][i
```

```
            ]) != 0: #checks first if the number of vectors that fulfill a inequality with equality is eight and checks then the determinant for the linear independence
            if p == 11890:
                print(i)
            for s in powerset:
                if len(s) == 3 and graphs[p][s[2]-1][s[1]] == 0 and graphs[p][s[2]-1][s[0]] == 0 and graphs[p][s[1]-1][s[0]] == 0: #the nodes that correspond to the preimages are not allowed to have edges between each other, since they correspond to the three desconnected nodes from the original cut set
                    if p == 11890:
                        print(s)
                    all_tripel = [] #saves the parts of the characteristic vectors which belong to the nodes that are candidates to represent the nodes of the original cut set
                    for j in range(len(vectors_in_facets_without_zerovector[p][i])):
                        new_tripel = [vectors_in_facets_without_zerovector[p][i][j][s[0]], vectors_in_facets_without_zerovector[p][i][j][s[1]], vectors_in_facets_without_zerovector[p][i][j][s[2]]]
                        all_tripel.append(new_tripel)
                    all_tripel.sort()
                    if p == 11890:
                        print(all_tripel)
                    if all_tripel == needed_stable_sets: #checks if all stable sets within the original cut set have a preimage within the stable sets that lie in a facet
                        print(p)
                        print(i)
```

A.2 Code Section 6.3.1

```
A = [0 -1 0 0 0 -1 1;
     0 0 -1 -1 0 -1 1;
     0 0 -1 0 -1 -1 1; %columns of B bar
     -1 0 0 0 0 0 0;
     0 -1 0 0 0 0 0;
     0 0 -1 0 0 0 0;
     0 0 0 -1 0 0 0;
     0 0 0 0 -1 0 0;
     0 0 0 0 0 -1 0;
     0 0 0 0 0 0 -1] %non-negativity constraints
```

```
b = [0; 0; 0; 0; 0; 0; 0; 0; 0; 0]

Aeq = [1,1,1,1,1,1,1] %equation that has to hold for z

beq = [1]

V=lcon2vert(A,b,Aeq,beq)
%This function was downloaded on
%https://de.mathworks.com/matlabcentral/fileexchange/30892-
    analyze-n-dimensional-polyhedra-in-terms-of-vertices-or-
    in-equalities
```