Matchings, covers, and stars

by

Heather Heier

B.S., Kansas State University, 2018

A REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Mathematics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2019

Approved by:

Major Professor
Nathan Albin

Copyright

Abstract

This report relates four important problems of combinatorial optimization on graphs: the maximum matching problem, the minimum edge cover problem, the maximum perfect matching problem, and the minimum perfect matching problem. Although polynomial time algorithms exist for each of these problems in general, they all become much simpler on bipartite graphs due to an interesting relationship to the stars on such a graph.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Matchings, perfect matchings, covers, and stars

This report is built on a few key concepts in graph theory. Matchings, perfect matchings, covers, and stars are objects of specific characteristics on a graph.

Definition 1.1.1. Let $G = (V, E)$ be a connected graph. A *matching* on G is a subset $M \subseteq E$ with the property that each vertex in V meets at most one edge in M.

Example 1.1.1. Consider the triangle graph, all of the possible matchings are illustrated in Figure 1.1

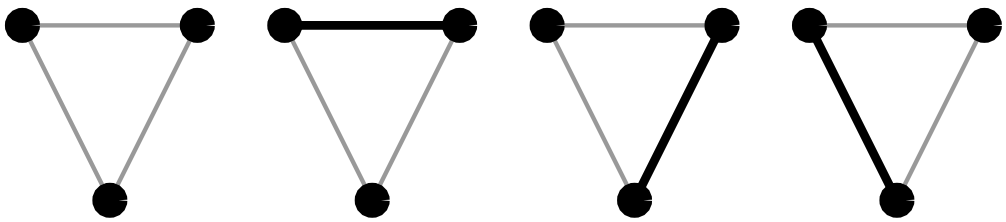


Figure 1.1: Triangle graph matchings

The set of all possible matchings of a graph is usually very large. An example of a smaller, more specific set of matchings is the set of perfect matchings.

Definition 1.1.2. A *perfect matching* is a subset $M \subseteq E$ with the property that each vertex in V meets exactly one edge in M.

Example 1.1.2. For instance, if we let G be a square graph, then G has two perfect matchings as shown in Figure 1.2.

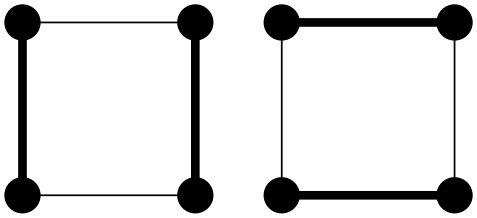


Figure 1.2: Perfect matching for a square graph.

Every connected graph has at least one matching because $M = \emptyset$ satisfies the definition of matching, but not every connected graph has a perfect matching. For instance, any graph with an odd number of vertices cannot have a perfect matching. The triangle graph is a simple example of a graph with an odd number of vertices, and all possible matchings result in at least one disconnected vertex, failing to create a perfect matching (Figure 1.1).

Definition 1.1.3. Let $G = (V, E)$ be a graph and let $F \subseteq E$ be a subset of edges. If every vertex of G meets at least one edge of F, then F is called an *edge cover* (or *cover* for short). In what follows, we shall use the notation Γ_{match}, $\Gamma_{\text{p-match}}$, and Γ_{cover} for the families of matchings, perfect matchings, and covers on G respectively.

Example 1.1.3. The family of covers for the square graph are illustrated in Figure 1.3.

Remark 1.1.1. Note that F is a perfect matching if and only if it is both a matching and a cover. In other words,

$$\Gamma_{\text{p-match}} = \Gamma_{\text{match}} \cap \Gamma_{\text{cover}}.$$

The three families above can be characterized using the family of stars.

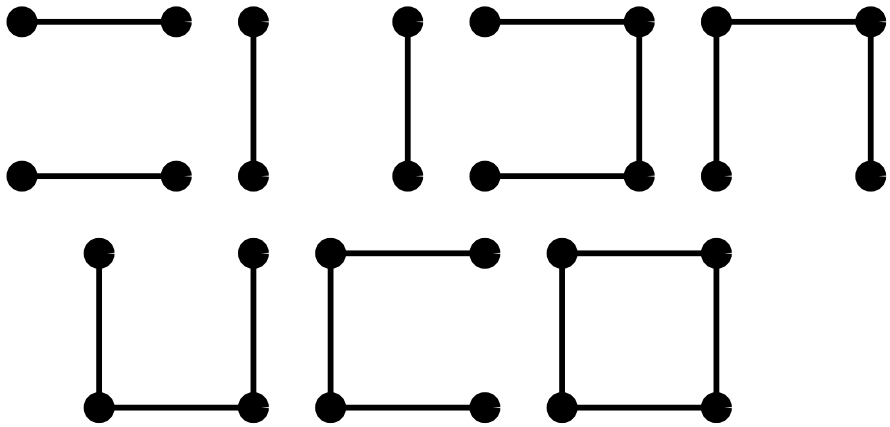


Figure 1.3: Γ_{cover} for square graph

Definition 1.1.4. Let v be any vertex of $G = (V, E)$. The *star*, S_v, centered at v is defined to be the set of all edges in E that meet v. We shall use the notation Γ_{star} to represent the family of all stars on G.

Therefore, stars are made from all edges attached directly to a vertex as shown in Figure 1.4.

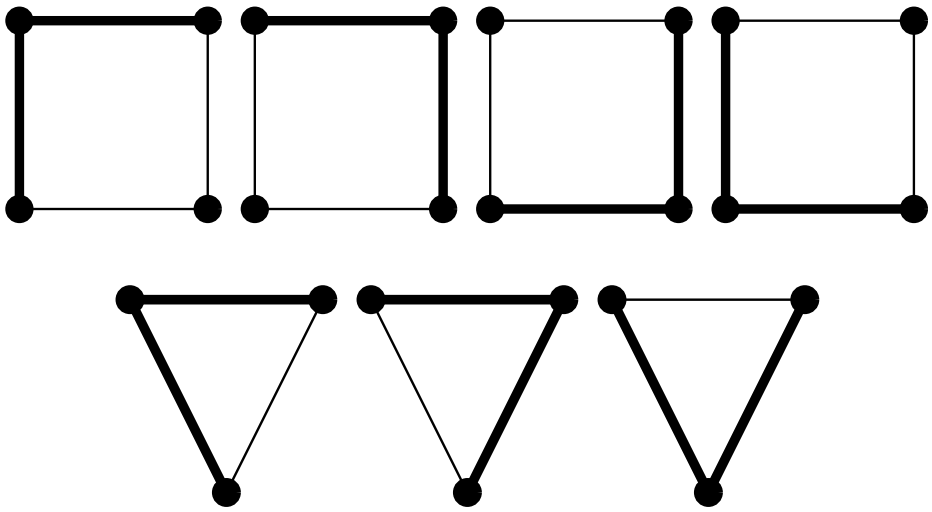


Figure 1.4: Stars for square and triangle.

The stars provide a convenient alternative to describing the families Γ_{match}, Γ_{cover}, and $\Gamma_{\text{p-match}}$.

Theorem 1.1.1. *Let $G = (V, E)$ be a graph and let $F \subseteq E$. Then*

- $F \in \Gamma_{match}$ *if and only if* $|F \cap S| \leq 1$ *for every* $S \in \Gamma_{star}$.
- $F \in \Gamma_{cover}$ *if and only if* $|F \cap S| \geq 1$ *for every* $S \in \Gamma_{star}$.
- $F \in \Gamma_{p\text{-}match}$ *if and only if* $|F \cap S| = 1$ *for every* $S \in \Gamma_{star}$.

Proof. Since $|F \cap S_v|$ counts the number of edges in F that meet v, this is an immediate consequence of the definitions. □

For each of these families Γ, there exists one or more optimal $F \in \Gamma$. A few examples of these are the maximum matchings and maximum cardinality matchings, or the minimal covers and minimum cardinality covers.

Definition 1.1.5. Let $M \in \Gamma_{\text{match}}$. M is called a *maximal matching* if it is not a proper subset of any other matching. M is called a *maximum cardinality matching* if

$$|M| = \max_{M' \in \Gamma_{\text{match}}} |M'|.$$

Definition 1.1.6. Let $C \in \Gamma_{\text{cover}}$. C is called a *minimal cover* if it does not contain any other cover as a proper subset. C is called a *minimum cardinality cover* if

$$|C| = \min_{C' \in \Gamma_{\text{cover}}} |C'|.$$

For example, the last three matchings in Figure 1.1 are maximal matchings and the first two covers in Figure 1.3 are the minimal covers.

Theorem 1.1.2. *If $M \subseteq E$ is a perfect matching, then it is both a maximum cardinality matching and a minimum cardinality cover.*

Proof. Let $M \subseteq E$ such that M is a perfect matching. As a result, each vertex $v \in V$ meets exactly one edge $M \in E$. Since each vertex meets an edge exactly once, the perfect matching

contains the minimum or most efficient number of edges to cover the vertices, making the perfect matchings the minimal cardinality covers. The perfect matching is also a maximal cardinality matching because a matching is a set of edges where each vertex meets at most one edge, and in a perfect matching each vertex meets the maximum value of one edge. □

1.2 Bipartite graphs

Definition 1.2.1. A bipartite graph is a graph whose vertices can be partitioned into two disjoint sets, $V = V_1 \cup V_2$, so that no two vertices in the same set have an edge between them.

For instance, the square graph used in Figures 1.2, 1.3, and 1.4, is bipartite (the disjoint sets consist of opposite corner vertices), but the triangle graph is not. A useful characterization of bipartite graphs is found in the following theorem.

Theorem 1.2.1. *A graph G with at least two vertices is bipartite if and only if it contains no odd cycles (cycles of odd length) [5].*

Proof. (Refer to page 14 of [5])

Let G be a bipartite graph consisting of partite sets V_1 and V_2. Let C denote a cycle in G where C is a list of visited vertices $v_1, v_2, ..., v_k, v_1$. Without loss of generality, let the cycle C start at some vertex in V_1 so every v_i for odd i is in V_1 and v_i for even i is in V_2. Since a cycle ends where it starts, the second to last vertex in C, v_k, must be in V_2 implying k is even, making C an even cycle.

For the opposite direction, let G be a graph with at least two vertices containing no odd cycles. Without loss of generality, assume G is connected, but if it is not, consider handling each of its connected components separately. Let v and x be vertices in G, and define V_1 to be the set vertices x with shortest paths from x to v having even length. Therefore, V_2 is the set of remaining vertices not in V_1.

Now, let $x \in V_1$ and $x' \in V_1$, and suppose they are adjacent. If x is v, then the shortest path from v to x' has length one since x and x' are adjacent, implying $x' \in V_2$, contradicting $x' \in V_1$. Without loss of generality, x and x' cannot equal v.

Let P_1 denote a shortest path from v to x and P_2 be the same from v to x'. P_1 and P_2 share the starting point v, so let v' be a vertex in both paths and let P_1' and P_2' be shortest paths from v' to x or x' correspondingly. Therefore, v' is the last common vertex in both paths to ensure P_1' and P_2' are the shortest paths, and $v' = v_i = w_i$ for some i. Since x and x' are adjacent, the cycle created using $P_1' + P_2' = v_i + v_{i+1}, \ldots, v_{2k}, w_{2t}, w_{2t-1}, \ldots, w_i$ is of odd length $(2k - i) + (2t - i) + 1$, creating a contradiction to the assumption there are no odd cycles in G.

In conclusion, no two vertices in V_1 or V_2 are adjacent, and therefore, G is a bipartite graph composed of two partite sets V_1 and V_2. □

A necessary condition for a connected bipartite graph to have a perfect matching is that $|V_1| = |V_2|$.

Remark 1.2.1. If we then consider a bipartite graph where $\Gamma_{\text{p-match}} \neq \emptyset$, the number of vertices in each of the disjoint sets must be the same, because every vertex must be paired with one other node. Since there are only edges from one disjoint set to the other, this forces all of the vertices in one set to be paired to vertices in the other set. In order to have a perfect matching, all vertices must be paired off, forcing the number of vertices to be equal for both sets.

However, this condition is not sufficient. An example is illustrated in Figure 1.5. The condition is satisfied, but there is no possible perfect matching since node e must meet two edges for a and c to meet one edge.

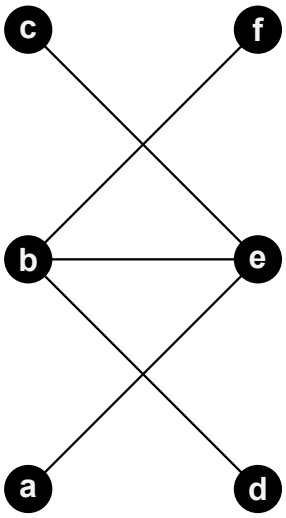


Figure 1.5: Bipartite counter example

In addition to $|V_1| = |V_2|$, we must also consider the neighbors of each vertex to find perfect matchings.

Definition 1.2.2. The *neighborhood* of a vertex v, denoted $N(v)$, is the set of vertices (neighbors) adjacent to v:

$$N(v) = \{x \in V \mid \{v, x\} \in E\}$$

For $F \subseteq V$,

$$N(F) = \left(\bigcup_{v \in F} N(v) \right) \setminus F$$

Definition 1.2.3. Let G be a bipartite graph with partite sets V_1 and V_2. If there exists a matching on G that covers V_1, we say that V_1 can be *matched into* V_2.

Thus, there exists a perfect matching for G if and only if $|V_1| = |V_2|$ and V_1 can be matched into V_2. Necessary and sufficient conditions for this last criterion are provided by Hall's Marriage Theorem.

Theorem 1.2.2 (Hall's Marriage Theorem [5])**.** *Let G be a bipartite graph with partite sets V_1 and V_2. V_1 can be matched into V_2 if and only if $|N(F)| \geq |F|$ for all subsets F of V_1.*

Idea of proof. Suppose that V_1 can be matched into V_2, and let F be some subset of V_1. The same matching $M \subseteq E$ that matches V_1 into V_2 also matches F into V_2. Thus, for every $v \in F$, there is a unique $\phi(v) \in V_2$ so that $\{v, \phi(v)\} \in M$. So,

$$|N(F)| \geq |\{\phi(v) : v \in F\}| = |F|.$$

The converse relies on a famous lemma called Berge's lemma and can be found on pages 106–107 in [5]. □

Chapter 2

Combinatorial optimization and linear programming relaxation

2.1 Optimization on graphs in a general framework

Consider a common class of combinatorial optimization problems on graphs. Start with a family of subsets of edges $\Gamma \subseteq 2^E$ (for example, Γ could be any of Γ_{match}, Γ_{cover} or $\Gamma_{\text{p-match}}$) and a set of non-negative edge weights $\sigma \in \mathbb{R}^E_{\geq 0}$. The weights σ provide a natural way to measure subsets $F \subseteq E$. Namely,

$$\sigma(F) := \sum_{e \in F} \sigma(e). \tag{2.1}$$

The quantity $\sigma(F)$, called the σ-*weight* or simply the *weight* of F, is just the accumulated values of σ on the edges included in F. With the family Γ and weights σ given, it is common to study one or both of the optimization problems

$$\begin{aligned} \text{minimize} \quad & \sigma(\gamma) \\ \text{subject to} \quad & \gamma \in \Gamma, \end{aligned} \tag{2.2}$$

and

$$\begin{aligned} \text{maximize} \quad & \sigma(\gamma) \\ \text{subject to} \quad & \gamma \in \Gamma. \end{aligned} \tag{2.3}$$

Example 2.1.1. The minimum weight and maximum weight problems for Γ_{match}, Γ_{cover} and $\Gamma_{\text{p-match}}$ are famous, widely studied problems with a variety of applications.

- When $\Gamma = \Gamma_{\text{match}}$, (2.3) is known as the *maximum weight matching problem*. (The minimum matching problem is not interesting since $F = \emptyset$ is a matching.) If, in addition, $\sigma \equiv 1$, the maximum weight matching problem is equivalent to the maximum cardinality matching problem.

- When $\Gamma = \Gamma_{\text{cover}}$, (2.2) is known as the *minimum weight edge cover problem*. (The maximum edge cover problem is not interesting since $F = E$ is an edge cover.) If, in addition, $\sigma \equiv 1$, the minimum weight cover problem is equivalent to the minimum cardinality cover problem.

- When $\Gamma = \Gamma_{\text{p-match}}$ (2.2) is known as the *minimum weight perfect matching problem* and (2.3) is known as the *maximum weight perfect matching problem*.

The minimum weight and maximum weight problems are *combinatorial optimization problems* because they involve optimization over a finite, usually large set. To get a sense of the size, suppose $G = K_{n,n}$, the complete bipartite graph with $|V_1| = |V_2| = n$ and with an edge connecting each pair $(x, y) \in V_1 \times V_2$. In that case, $|\Gamma_{\text{p-match}}| = n!$. Therefore, solving either the minimum weight or maximum weight problem by directly finding all perfect matchings and computing their weights is not computationally feasible for even moderately large graphs. Fortunately, these problems can be solved without doing this through a linear relaxation.

Remark 2.1.1. By convention, if $\Gamma_{\text{p-match}} = \emptyset$, the value of the minimum perfect matching

problem is said to be $+\infty$. Assuming $\Gamma_{\text{p-match}}$ is not empty, a minimum perfect matching always exists, but may not be unique.

For instance, if the square graph has edge weights as shown in Figure 2.1, then the minimum perfect matching is $\min\{a+c, b+d\}$ and it is only unique if $a+c \neq b+d$.

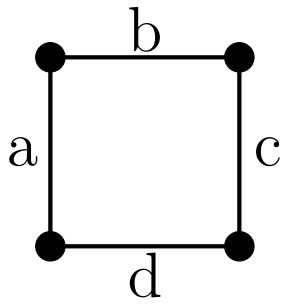


Figure 2.1: Non-unique perfect matching

2.2 Polytopes for matchings and covers

Consider again an arbitrary family of subsets of edges $\Gamma \subseteq 2^E$. Every element $\gamma \in \Gamma$ is a subset of edges ($\gamma \subseteq E$) and, hence, can be associated with its incidence vector $\mathbb{1}_\gamma \in \mathbb{R}^E$ defined as

$$\mathbb{1}_\gamma(e) = \begin{cases} 1 & \text{if } e \in \gamma, \\ 0 & \text{otherwise.} \end{cases}$$

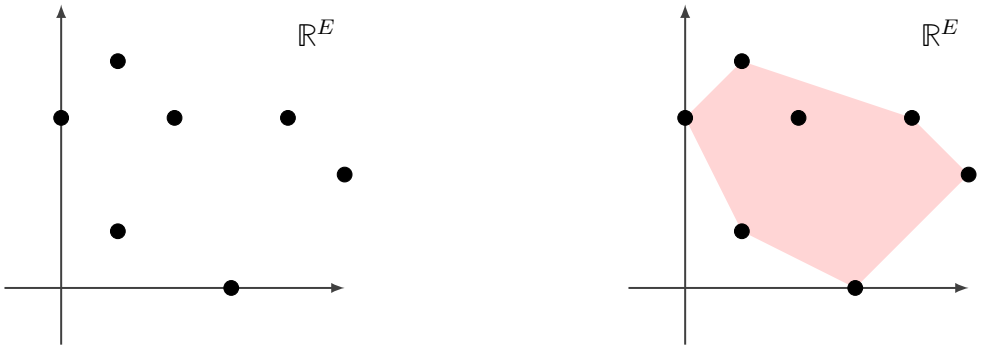
In this way, Γ can be represented as a cloud of points in $\mathbb{R}^E$, where each point corresponds to an element of Γ as shown in Figure 2.2a. Frequently, Γ shall be associated with the set of incidence vectors so that Γ is treated as a finite subset of $\mathbb{R}^E$. The convex hull of this set,

$$\text{conv}(\Gamma) := \text{conv}\left\{\mathbb{1}_\gamma : \gamma \in \Gamma\right\},$$

is called the *polytope* for the family Γ. As a consequence of this definition, it follows that

$$\text{ext}(\text{conv}(\Gamma)) \subseteq \Gamma, \tag{2.4}$$

where ext(conv(Γ)) is the set of extreme points (i.e., vertices) of the polytope conv(Γ) (see Figure 2.2b). In what follows, the notation P_{match}, P_{cover} and $P_{\text{p-match}}$ is used to denote the polytopes for Γ_{match}, Γ_{cover} and $\Gamma_{\text{p-match}}$ respectively.



(a) (b)

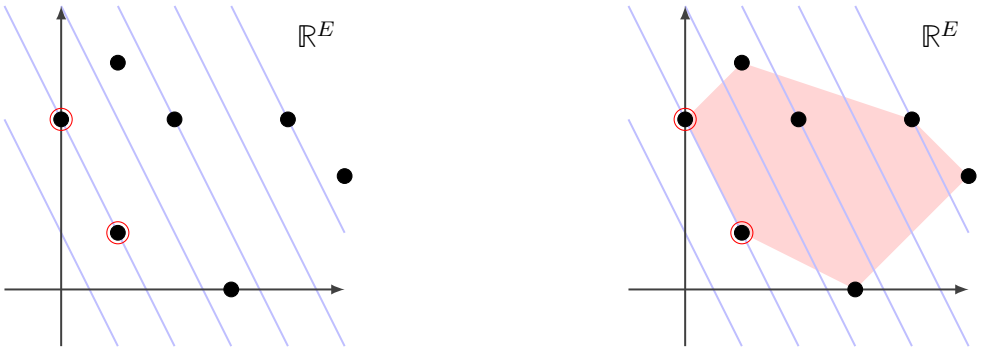
Figure 2.2: (a) A diagram representing a family $\Gamma \subseteq 2^E$ in $\mathbb{R}^E$. (b) The same diagram with the convex hull conv(Γ) shaded.

With the new representation, the quantity $\sigma(F)$ from (2.1) can be rewritten as

$$\sigma(F) = \sum_{e \in F} \sigma(e) = \sum_{e \in E} \sigma(e) \mathbb{1}_F(e) = \sigma^T \mathbb{1}_F,$$

which is a linear function. Thus, for example, the minimum perfect matching problem can be viewed as a problem of minimizing a linear function over a finite set of points in $\mathbb{R}^E$ (see Figure 2.3a).

The fundamental theorem of linear programming states that the minimum and maximum values of a linear function over a convex polytope are attained on the polytope's vertices. Due to convexity, if the minimum (or maximum) is attained at more than one vertex, it is necessarily attained on the entire line segment connecting them (see Figure 2.3b).



(a) (b)

Figure 2.3: (a) A diagram representing the minimum perfect matching problem visualized in $\mathbb{R}^E$. Blue lines represent level sets of the linear function $x \mapsto \sigma^T x$. In this figure, the minimum is attained at two points, circled in red. (b) The same diagram with the perfect matching polytope shaded. Since there are two minimum perfect matchings, the polytope includes a line segment of minimizers.

Therefore, (2.2) is equivalent to

$$\begin{aligned} \text{minimize} \quad & \sigma^T x \\ \text{subject to} \quad & x \in \operatorname{conv}(\Gamma), \end{aligned} \tag{2.5}$$

in the sense that (2.2) and (2.5) have the same value and every minimizer of (2.2) corresponds to a minimizer of (2.5). (Note, however, that (2.5) may have additional minimizers.) In a similar way, (2.3) is equivalent to

$$\begin{aligned} \text{maximize} \quad & \sigma^T x \\ \text{subject to} \quad & x \in \operatorname{conv}(\Gamma). \end{aligned} \tag{2.6}$$

2.3 Polytope inequalities

Although (2.5) and (2.6) provide interesting interpretations of the minimum weight and maximum weight problems, it is not yet clear that they simplify the combinatorial problems. The most direct way to find $\operatorname{conv}(\Gamma)$, after all, is to find all of Γ and then form the convex hull. Often, this difficulty can be addressed by using an alternative description of the polytope.

Every polytope has two natural descriptions; it can be defined through its vertices, or through its faces. The notation $\mathrm{conv}(\Gamma)$ is an example of the former, often called the *V-representation* of the polytope. The alternative is to define the polytope as an intersection of half-spaces (given by linear inequalities), which is called the *H-representation*.

Moving between the two representations is typically a nontrivial task. However, for the polytopes of interest in this report, Theorem 1.1.1 provides a good starting place. Rewritten in the new notation, Theorem 1.1.1 can be stated as follows.

Theorem 2.3.1. *For a graph $G = (V, E)$,*

$$\Gamma_{match} = \left\{ x \in \{0,1\}^E : \sum_{e \in S} x(e) \leq 1 \ \forall S \in \Gamma_{star} \right\},$$
$$\Gamma_{cover} = \left\{ x \in \{0,1\}^E : \sum_{e \in S} x(e) \geq 1 \ \forall S \in \Gamma_{star} \right\},$$
$$\Gamma_{p\text{-}match} = \left\{ x \in \{0,1\}^E : \sum_{e \in S} x(e) = 1 \ \forall S \in \Gamma_{star} \right\}.$$

Now consider the case of $\Gamma = \Gamma_{\mathrm{match}}$. (The cases $\Gamma = \Gamma_{\mathrm{cover}}$ and $\Gamma = \Gamma_{\text{p-match}}$ can be treated similarly.) In this notation, the maximum weight matching problem can be written in the form

$$\begin{aligned} \text{maximize} \quad & \sigma^T x \\ \text{subject to} \quad & x \in \Gamma_{\mathrm{match}}. \end{aligned} \tag{2.7}$$

This is an example of an integer linear program (ILP). Such optimization problems are known to be computationally hard to solve. The difficulty lies in the constraints that $x(e)$ may only take values in a discrete set of points. A standard technique in the study of ILPs is the use of *linear programming relaxation*. In a linear programming relaxation, we replace the integer variables with real-valued variables subject to a bound constraint. For (2.7), the linear programming relaxation is defined by replacing the constraint $x \in \Gamma_{\mathrm{match}}$ by $x \in \tilde{P}_{\mathrm{match}}$,

where

$$\tilde{P}_{\text{match}} = \left\{ x \in [0,1]^E : \sum_{e \in S} x(e) \leq 1 \ \forall S \in \Gamma_{\text{star}} \right\}.$$

The problem

$$\begin{aligned} \text{maximize} \quad & \sigma^T x \\ \text{subject to} \quad & 0 \leq x \leq 1, \\ & \sum_{e \in S} x(e) \leq 1 \ \forall S \in \Gamma_{\text{star}}, \end{aligned} \tag{2.8}$$

is called the *linear programming relaxation* of the maximum weight matching problem. Observe that, since $\Gamma_{\text{match}} \subseteq \tilde{P}_{\text{match}}$, it follows that $P_{\text{match}} \subseteq \tilde{P}_{\text{match}}$. It is generally much easier to solve (2.8) than it is to even enumerate Γ_{match}. The latter is a combinatorial problem while the former is a linear program with $|E|$ variables and $2|E| + |V|$ constraints. However, if $P_{\text{match}} \subsetneq \tilde{P}_{\text{match}}$ then the (incidence vector of) the maximum matching is not necessarily a maximizer of (2.8).

Fortunately, it turns out that $P_{\text{match}} = \tilde{P}_{\text{match}}$ on bipartite graphs. In fact, if $\tilde{P}_{\text{cover}}$ and $\tilde{P}_{\text{p-match}}$ are defined analogously, the following theorem holds.

Theorem 2.3.2. *If $G = (V, E)$ is a bipartite graph then*

1. $P_{cover} = \tilde{P}_{cover}$,
2. $P_{match} = \tilde{P}_{match}$, *and*
3. $P_{p\text{-}match} = \tilde{P}_{p\text{-}match}$.

Proof. The proof of 1) can be found in [7, Theorem 19.6], 2) and 3) are typically attributed to Edmonds [3]; short proofs of these two statements can be found in [6]. □

For the case of non-bipartite graphs, none of the equalities in Theorem 2.3.2 are true in general and the linear programming relaxations of the various minimum- and maximum weight problems are, therefore, not equivalent to the original problem but, instead, provide

lower or upper bounds. However, additional inequalities can be added to $\tilde{P}_{\text{cover}}$, etc. to provide a true H-representation for the polytopes (see Chapter 4).

Chapter 3

Solving the minimum and maximum weight optimization problems on bipartite graphs

In this section, we derive the Karush Kuhn Tucker (KKT) optimality conditions for the minimum weight and maximum weight optimization problems described in Example 2.1.1 on bipartite graphs.

3.1 The maximum weight bipartite matching problem

As previously stated, a consequence of Theorem 2.3.2 is that the value of the maximum weight matching problem on a bipartite graph is equal to the value of its LP relaxation (2.8). For such a problem, a set of conditions called the KKT conditions are both necessary and sufficient for optimality. The KKT conditions for this problem can be derived using the theory of Lagrangian duality as follows.

3.1.1 The KKT conditions

Begin by introducing a set of Lagrangian dual variables $\nu, \eta \in \mathbb{R}^E$, $\lambda \in \mathbb{R}^{\Gamma_{\text{star}}}$, which are used to form the Lagrangian energy

$$\begin{aligned} L(x, \nu, \eta, \lambda) &= \sum_{e \in E} \sigma(e)x(e) + \sum_{e \in E} \nu(e)x(e) + \sum_{e \in E} \eta(e)(1 - x(e)) + \sum_{S \in \Gamma_{\text{star}}} \lambda(S) \left[1 - \sum_{e \in E} x(e) \mathbb{1}_S(e) \right] \\ &= \sum_{S \in \Gamma_{\text{star}}} \lambda(S) + \sum_{e \in E} \eta(e) + \sum_{e \in E} x(e) \left[\sigma(e) + \nu(e) - \eta(e) - \sum_{S \in \Gamma_{\text{star}}} \lambda(S) \mathbb{1}_S(e) \right]. \end{aligned}$$

For a linear program such as (2.8), the KKT conditions provide a set of necessary and sufficient conditions for optimality. That is, a vector $x \in \mathbb{R}^E$ minimizes (2.8) if and only if there exists a set of dual variables (ν, η, λ) that, together with x, satisfy the KKT conditions. The KKT conditions can be divided into four categories: primal feasibility, dual feasibility, complementary slackness and stationarity.

Primal feasibility. Primal feasibility describes the requirement that the primal variable x is feasible for the primal problem. For (2.8), these conditions are

$$0 \leq x \leq 1, \qquad \text{and} \qquad \sum_{e \in S} x(e) \leq 1 \quad \forall S \in \Gamma_{\text{star}}. \tag{3.1}$$

Dual feasibility. Dual feasibility is the requirement that all dual variables associated with inequality constraints are nonnegative. For (2.8), these conditions are

$$\nu, \eta \in \mathbb{R}^E_{\geq 0}, \qquad \lambda \in \mathbb{R}^{\Gamma_{\text{star}}}_{\geq 0} \tag{3.2}$$

Complementary slackness. Complementary slackness conditions provide connections between the primal and dual variables involved in equality constraints. For (3.1), these

conditions are

$$\nu(e)x(e) = 0 \quad \text{and} \quad \eta(e)(1 - x(e)) = 0 \quad \forall e \in E, \tag{3.3}$$

$$\lambda(S)\left[1 - \sum_{e \in S} x(e)\right] = 0 \quad \forall S \in \Gamma_{\text{star}}. \tag{3.4}$$

Stationarity. Stationarity refers to the requirement that the partial derivative of L with respect to every primal variable must be 0 at a minimum point. For (2.8) these conditions can be written as

$$\sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e) - \sigma(e) = \nu(e) - \eta(e). \tag{3.5}$$

The results of Lagrangian duality theory can be summarized for (2.8) as follows.

Theorem 3.1.1. *A vector $x \in \mathbb{R}^E$ is a maximizer of (2.8) if and only if there exist dual variables $\nu, \eta \in \mathbb{R}^E$, $\lambda \in \mathbb{R}^{\Gamma_{star}}$ that, together with x, satisfy (3.1)–(3.5).*

3.1.2 Interpreting the KKT conditions

The KKT conditions (3.1)–(3.5) for the maximum weight bipartite matching problem can be better understood through a reinterpretation of these conditions.

To begin, note that there is a one-to-one correspondence between V and Γ_{star} given by the natural mapping $v \mapsto S_v$. Thus, we may think of λ as a vector in $\mathbb{R}^V$. Moreover, since each edge $e = \{u, v\} \in E$ only meets two stars (S_u and S_v), we may rewrite

$$\sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e) = \lambda(u) + \lambda(v).$$

Now, observe that, from (3.3), it is impossible for both $\nu(e)$ and $\eta(e)$ to simultaneously be positive on any given edge e. (Either both are zero, or exactly one of the two is positive and the other is zero.) This provides an interesting interpretation of (3.5). Namely, for all

$e = \{u, v\} \in E$,

$$\nu(e) = [\lambda(u) + \lambda(v) - \sigma(e)]^+ , \tag{3.6}$$

$$\eta(e) = [\lambda(u) + \lambda(v) - \sigma(e)]^- , \tag{3.7}$$

where $[\cdot]^+$ and $[\cdot]^-$ are the *positive part* and *negative part* functions,

$$z^+ := \max\{z, 0\}, \qquad z^- := \max\{-z, 0\}$$

respectively.

Using this reinterpretation, we can rewrite Theorem 3.1.1 as follows.

Theorem 3.1.2. *A vector $x \in \mathbb{R}^E$ is a maximizer of (2.8) if and only if there exist dual variables $\lambda \in \mathbb{R}^V$ that, together with x, satisfy the following conditions.*

$$0 \le x \le 1, \qquad 0 \le \lambda, \qquad \text{and} \qquad \sum_{e \in S_v} x(e) \le 1 \quad \forall v \in V,$$

$$x(e)\,[\lambda(u) + \lambda(v) - \sigma(e)]^+ = 0 \quad \forall e = \{u, v\} \in E,$$

$$(1 - x(e))\,[\lambda(u) + \lambda(v) - \sigma(e)]^- = 0 \quad \forall e = \{u, v\} \in E,$$

$$\lambda(v) \left[1 - \sum_{e \in S_v} x(e)\right] = 0 \quad \forall v \in V.$$

In words, Theorem 3.1.2, can be described as follows. We wish to find a weight $x(e) \in [0, 1]$ for each $e \in E$ and a non-negative weight $\lambda(v)$ for each $v \in V$ that satisfy the following conditions.

- For any vertex v, the sum of x over all edges incident on v is no larger than 1.
- For any edge $e = \{u, v\}$, if $\lambda(u) + \lambda(v) > \sigma(e)$, then $x(e) = 0$.
- For any edge $e = \{u, v\}$, if $\lambda(u) + \lambda(v) < \sigma(e)$, then $x(e) = 1$.

- If on any vertex v, $\lambda(v) > 0$, then the sum of x over all edges incident on v is exactly equal to 1.

3.2 The minimum weight bipartite edge cover problem

Similar to the maximum weight matching problem, the value of the minimum weight edge cover problem on a bipartite graph is equal to the value of its LP relaxation:

$$\begin{aligned} \text{minimize} \quad & \sigma^T x \\ \text{subject to} \quad & 0 \leq x \leq 1, \\ & \sum_{e \in S} x(e) \geq 1 \; \forall S \in \Gamma_{\text{star}}. \end{aligned} \tag{3.8}$$

Again, for such a problem, the KKT conditions are both necessary and sufficient for optimality. The KKT conditions for this problem can be derived using the theory of Lagrangian duality as follows.

3.2.1 The KKT conditions

As in section 3.1, begin by introducing a set of Lagrangian dual variables $\nu, \eta \in \mathbb{R}^E$, $\lambda \in \mathbb{R}^{\Gamma_{\text{star}}}$, which are used to form the Lagrangian energy

$$\begin{aligned} L(x, \nu, \eta, \lambda) &= \sum_{e \in E} \sigma(e)x(e) - \sum_{e \in E} \nu(e)x(e) - \sum_{e \in E} \eta(e)(1 - x(e)) + \sum_{S \in \Gamma_{\text{star}}} \lambda(S) \left[1 - \sum_{e \in E} x(e) \mathbb{1}_S(e) \right] \\ &= \sum_{S \in \Gamma_{\text{star}}} \lambda(S) - \sum_{e \in E} \eta(e) + \sum_{e \in E} x(e) \left[\sigma(e) - \nu(e) + \eta(e) - \sum_{S \in \Gamma_{\text{star}}} \lambda(S) \mathbb{1}_S(e) \right]. \end{aligned}$$

The KKT conditions for this problem divided into four categories, primal feasibility, dual feasibility, complementary slackness, and stationarity are as follows.

Primal feasibility.

$$0 \leq x \leq 1, \qquad \text{and} \qquad \sum_{e \in S} x(e) \geq 1 \quad \forall S \in \Gamma_{\text{star}} \tag{3.9}$$

Dual feasibility.

$$\nu, \eta \in \mathbb{R}^{E}_{\geq 0}, \qquad \lambda \in \mathbb{R}^{\Gamma_{\text{star}}}_{\geq 0} \tag{3.10}$$

Complementary slackness.

$$\nu(e)x(e) = 0 \quad \text{and} \quad \eta(e)(1 - x(e)) = 0 \quad \forall e \in E, \tag{3.11}$$

$$\lambda(S) \left[1 - \sum_{e \in S} x(e) \right] = 0 \quad \forall S \in \Gamma_{\text{star}} \tag{3.12}$$

Stationarity.

$$\sigma(e) - \sum_{S \in \Gamma_{\text{star}}} \lambda(S) \mathbb{1}_S(e) = \nu(e) - \eta(e) \tag{3.13}$$

3.2.2 Interpreting the KKT conditions

As before, the KKT conditions (3.9)–(3.13) for the minimum weight bipartite edge cover problem can be better understood through a reinterpretation of these conditions.

Again, there is a one-to-one correspondence between V and Γ_{star} leading to

$$\sum_{S \in \Gamma_{\text{star}}} \lambda(S) \mathbb{1}_S(e) = \lambda(u) + \lambda(v).$$

Here it is also impossible for both $\nu(e)$ and $\eta(e)$ to simultaneously be positive on any given edge e, so now

$$\nu(e) = \left[\sigma(e) - \lambda(u) - \lambda(v) \right]^{+}, \tag{3.14}$$

$$\eta(e) = \left[\sigma(e) - \lambda(u) - \lambda(v) \right]^{-}. \tag{3.15}$$

Using this reinterpretation, we can rewrite the optimization problem as follows.

$$0 \leq x \leq 1, \qquad 0 \leq \lambda, \qquad \text{and} \qquad \sum_{e \in S_v} x(e) \geq 1 \quad \forall v \in V,$$

$$x(e)\left[\sigma(e) - \lambda(u) - \lambda(v)\right]^{+} = 0 \quad \forall e = \{u, v\} \in E,$$

$$(1 - x(e))\left[\sigma(e) - \lambda(u) - \lambda(v)\right]^{-} = 0 \quad \forall e = \{u, v\} \in E,$$

$$\lambda(v)\left[1 - \sum_{e \in S_v} x(e)\right] = 0 \quad \forall v \in V.$$

In words, we wish to find a weight $x(e) \in [0, 1]$ for each $e \in E$ and a non-negative weight $\lambda(v)$ for each $v \in V$ that satisfy the following conditions.

- For any vertex v, the sum of x over all edges incident on v is no less than 1.
- For any edge $e = \{u, v\}$, if $\lambda(u) + \lambda(v) < \sigma(e)$, then $x(e) = 0$.
- For any edge $e = \{u, v\}$, if $\lambda(u) + \lambda(v) > \sigma(e)$, then $x(e) = 1$.
- If on any vertex v, $\lambda(v) > 0$, then the sum of x over all edges incident on v is exactly equal to 1.

3.3 The minimum and maximum weight bipartite perfect matching problems

Lastly, the minimum and maximum weight bipartite perfect matching problems are nearly identical, but differ in optimizing to a maximum or minimum. The problems are:

$$\begin{aligned} \text{maximize/minimize} \quad & \sigma^T x \\ \text{subject to} \quad & 0 \leq x \leq 1, \\ & \sum_{e \in S} x(e) = 1 \; \forall S \in \Gamma_{\text{star}}, \end{aligned} \tag{3.16}$$

The KKT conditions for these problems can be derived using the theory of Lagrangian duality just like the last two, but with one less dual variable due to the summation constraint now equaling one. For these problems, there are slight differences in the KKT conditions due to the sign differences in the Lagrangian.

3.3.1 The KKT conditions

The Lagrangian energy for the minimization problem is

$$\begin{aligned} L(x,\nu,\eta,\lambda) &= \sum_{e\in E}\sigma(e)x(e) - \sum_{e\in E}\nu(e)x(e) - \sum_{e\in E}\eta(e)(1-x(e)) + \sum_{S\in\Gamma_{\text{star}}}\lambda(S)\left[1-\sum_{e\in E}x(e)\mathbb{1}_S(e)\right] \\ &= \sum_{S\in\Gamma_{\text{star}}}\lambda(S) - \sum_{e\in E}\eta(e) + \sum_{e\in E}x(e)\left[\sigma(e)-\nu(e)+\eta(e)-\sum_{S\in\Gamma_{\text{star}}}\lambda(S)\mathbb{1}_S(e)\right] \end{aligned}$$

while it is the following for the maximization problem

$$\begin{aligned} L(x,\nu,\eta,\lambda) &= \sum_{e\in E}\sigma(e)x(e) + \sum_{e\in E}\nu(e)x(e) + \sum_{e\in E}\eta(e)(1-x(e)) + \sum_{S\in\Gamma_{\text{star}}}\lambda(S)\left[1-\sum_{e\in E}x(e)\mathbb{1}_S(e)\right] \\ &= \sum_{S\in\Gamma_{\text{star}}}\lambda(S) + \sum_{e\in E}\eta(e) + \sum_{e\in E}x(e)\left[\sigma(e)+\nu(e)-\eta(e)-\sum_{S\in\Gamma_{\text{star}}}\lambda(S)\mathbb{1}_S(e)\right] \end{aligned}$$

The KKT conditions for both of these problems will be the same.

Primal feasibility. The primal feasibiltiy conditions are

$$0 \le x \le 1, \qquad \text{and} \qquad \sum_{e\in S}x(e) = 1 \quad \forall S \in \Gamma_{\text{star}}. \tag{3.17}$$

Dual feasibility. Since there are only two inequalities, only two dual variables are necessary.

$$\nu,\eta \in \mathbb{R}^E_{\ge 0} \tag{3.18}$$

Complementary slackness. For both problems the complementary slackness conditions are

$$\nu(e)x(e) = 0 \quad \text{and} \quad \eta(e)(x(e) - 1) = 0 \quad \forall e \in E. \tag{3.19}$$

Stationarity. For the stationarity constraints, there is a sign difference for the minimum and maximum problems. For the maximum problem, the constraint is

$$\sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e) - \sigma(e) = \nu(e) - \eta(e) \quad \forall e \in E. \tag{3.20}$$

while the minimum problem's constraint is

$$\sigma(e) - \sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e) = \nu(e) - \eta(e) \quad \forall e \in E. \tag{3.21}$$

Here it is also impossible for both $\nu(e)$ and $\eta(e)$ to simultaneously be positive on any given edge e, so now for the maximum problem,

$$\nu(e) = \left[\sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e) - \sigma(e)\right]^+, \tag{3.22}$$

$$\eta(e) = \left[\sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e) - \sigma(e)\right]^-, \tag{3.23}$$

and for the minimum problem there is

$$\nu(e) = \left[\sigma(e) - \sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e)\right]^+, \tag{3.24}$$

$$\eta(e) = \left[\sigma(e) - \sum_{S \in \Gamma_{\text{star}}} \lambda(S)\mathbb{1}_S(e)\right]^-. \tag{3.25}$$

3.3.2 Interpreting the KKT conditions

The KKT conditions can be rewritten as follows. Both the maximum and minimum problems share the conditions that

$$0 \leq x \leq 1, \qquad \text{and} \qquad \sum_{e \in S_v} x(e) = 1 \quad \forall v \in V.$$

For the maximum problem we add the conditions that

$$x(e)\left[\lambda(u) + \lambda(v) - \sigma(e)\right]^+ = 0 \qquad \forall e = \{u, v\} \in E$$
$$(1 - x(e))\left[\lambda(u) + \lambda(v) - \sigma(e)\right]^- = 0 \qquad \forall e = \{u, v\} \in E,$$

while for the minimum problem we add the conditions that

$$x(e)\left[\sigma(e) - \lambda(u) - \lambda(v)\right]^+ = 0 \qquad \forall e = \{u, v\} \in E$$
$$(1 - x(e))\left[\sigma(e) - \lambda(u) - \lambda(v)\right]^- = 0 \qquad \forall e = \{u, v\} \in E.$$

In words, to solve these optimization problems, we wish to find a weight $x(e) \in [0, 1]$ for each $e \in E$ and a real weight $\lambda(v)$ for each $v \in V$ that satisfy the following conditions.

- For any vertex v, the sum of x over all edges incident on v is exactly 1.
- For any edge $e = \{u, v\}$ in the maximum problem, if $\lambda(u) + \lambda(v) > \sigma(e)$, then $x(e) = 0$. If $\lambda(u) + \lambda(v) < \sigma(e)$, then $x(e) = 1$.
- For any edge $e = \{u, v\}$ in the minimum problem, if $\lambda(u) + \lambda(v) < \sigma(e)$, then $x(e) = 0$. If $\lambda(u) + \lambda(v) > \sigma(e)$, then $x(e) = 1$.

3.4 Problem comparison

Looking at all four problems at once, it is obvious that there are similarities. The last two problems have one less constraint compared to the first two, due to the complementary slackness constraints only being present for each inequality, and for the perfect matching problems, the sum of the edges on a star must equal 1. The first two problems share that missing complementary slackness constraint to ensure that if $\lambda(v) > 0$ for any vertex v, the sum of the x values over all edges incident to v is exactly one.

Comparing the minimizing and maximizing problems, the signs of the sigma are reversed for the stationarity conditions. Lastly, the primal feasibility constraint involving sums over stars differs depending on the family of objects under consideration. For instance, the matching problem has its own primal feasibility condition that for any vertex v, the sum of x over all edges incident on v is exactly one. Similarly, the edge cover problem has its own primal feasibility condition, and the perfect matching problems share the same primal feasibility conditions.

To demonstrate that these KKT conditions result in a valid optimization for a bipartite graph, consider Figure 3.1. This graph and its assigned edge weights, σ, are optimized according to the KKT conditions for each of the four problems. The optimal λ values are shown on each vertex, and the optimal usage values $x(e)$ are listed on each edge. The edges that are not highlighted with a values listed have $x(e) = 0$. For each graph, the KKT conditions described in this chapter are satisfied. Therefore, the optimal solutions highlighted are valid, and their values are listed for each problem above the corresponding graph.

For non-bipartite graphs, these KKT conditions result in an invalid result. Chapter 4 will further explain the difficulty of solving these optimization problems on a non-bipartite graph.

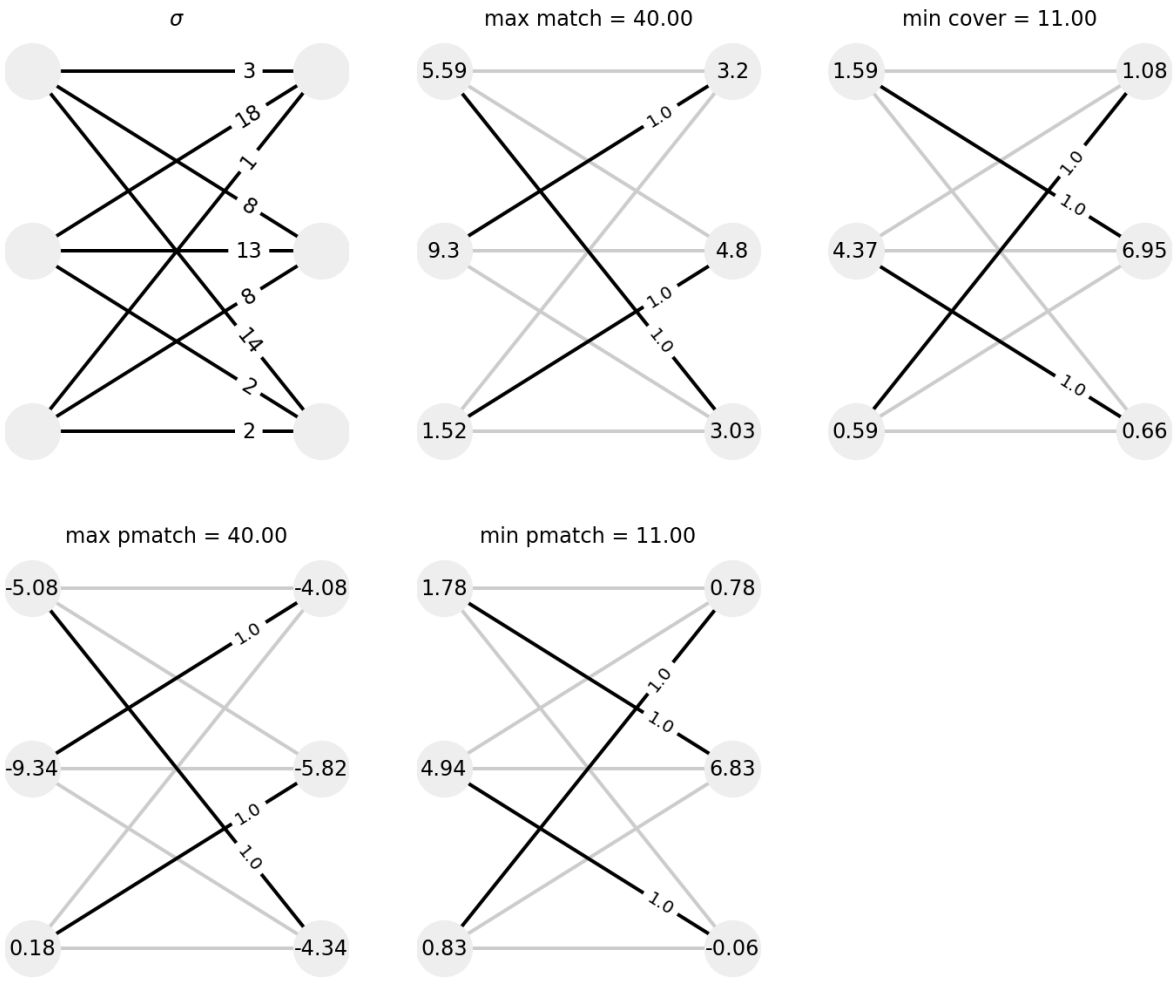


Figure 3.1: Solved bipartite graph using the LP-relaxed constraints

Chapter 4

Extending the theory to non-bipartite graphs

4.1 The problem with non-bipartite graphs

In general, Theorem 2.3.2 is not true for non-bipartite graphs. As a simple example of what goes wrong, consider the triangle graph and the vector $x \equiv \frac{1}{2}$. This point is in each of $\tilde{P}_{\text{cover}}$, $\tilde{P}_{\text{match}}$ and $\tilde{P}_{\text{p-match}}$, but not in any of P_{cover}, P_{match} or $P_{\text{p-match}}$.

In each case, additional inequalities are needed in order to produce the H-representation of the polytopes. The number of additional inequalities is based on the inequalities found in Edmonds [3]. The intersection of the half-spaces given by these inequalities result in the convex polytope. The following definition redefines the necessary notation to better understand the second and third inequalities in Edmonds [3].

Definition 4.1.1. Let $V' \subset V$. The notation $E[V']$ denotes the subset of edges in E that connect two vertices of V', while the notation $\delta(V')$ denotes the subset of edges in E that connect a vertex of V' to a vertex of $V \setminus V'$. Note that if $V' = \{v\}$, then $E[V'] = \emptyset$ and $\delta(V') = S_v$.

Each of the polytopes P_{cover}, P_{match} and $P_{\text{p-match}}$ can be described by adding an additional

set of inequalities to the corresponding polytopes $\tilde{P}_{\text{cover}}$, $\tilde{P}_{\text{match}}$ and $\tilde{P}_{\text{p-match}}$.

Theorem 4.1.1. *A vector $x \in \mathbb{R}^E$ is in P_{cover} if and only if $x \in \tilde{P}_{cover}$ and*

$$x(E[V'] \cup \delta(V')) \geq r + 1 \quad \textit{for all } V' \subseteq V \textit{ with } |V'| = 2r + 1, ; r = 0, 1, 2, \ldots . \tag{4.1}$$

Proof. See [7, Corollary 27.3a]. □

For the polytopes P_{match} and $P_{\text{p-match}}$, Edmonds' matching polyhedron theorem described by Alexander Schrijver [6] proves that the following theorems are true.

Theorem 4.1.2. *A vector $x \in \mathbb{R}^E$ is in P_{match} if and only if $x \in \tilde{P}_{match}$ and*

$$x(E[V']) \leq r \quad \textit{for all } V' \subseteq V \textit{ with } |V'| = 2r + 1, ; r = 0, 1, 2, \ldots . \tag{4.2}$$

Theorem 4.1.3. *A vector $x \in \mathbb{R}^E$ is in $P_{p\text{-}match}$ if and only if $x \in \tilde{P}_{p\text{-}match}$ and*

$$x(\delta(V')) \geq 1 \quad \textit{for all } V' \subset V \textit{ with } |V'| = 2r + 1, ; r = 0, 1, 2, \ldots . \tag{4.3}$$

4.2 Why bipartite graphs are special

The bipartite graphs are special and do not require more inequality constraints because, as proved in Chapter 1, bipartite graphs do not contain odd cycles. As described in Edmonds [3], more inequalities are necessary to create the convex polynomial for a graph with odd cycles. For example, consider the two graphs shown in Figures 3.1 and 4.1. The solutions for each optimization problem in Figure 3.1 satisfy each of the constraints as described in Chapter 3. Therefore, the relaxed problem is equal to the original problem for bipartite graphs, and this again is due to the non-existence of odd cycles. The graph in Figure 4.1 contains odd cycles, so Edmonds [3] says that additional inequalities are necessary to solve the optimization problem correctly. Therefore, the optimization of the non-bipartite graph should not satisfy

the relaxed constraints for every problem, and Figure 4.1 illustrates that. For the maximum matching and the minimum cover problems, the results are valid, but the perfect matching problems are incorrect because, there is no perfect matching for the graph. Therefore, more constraints are needed.

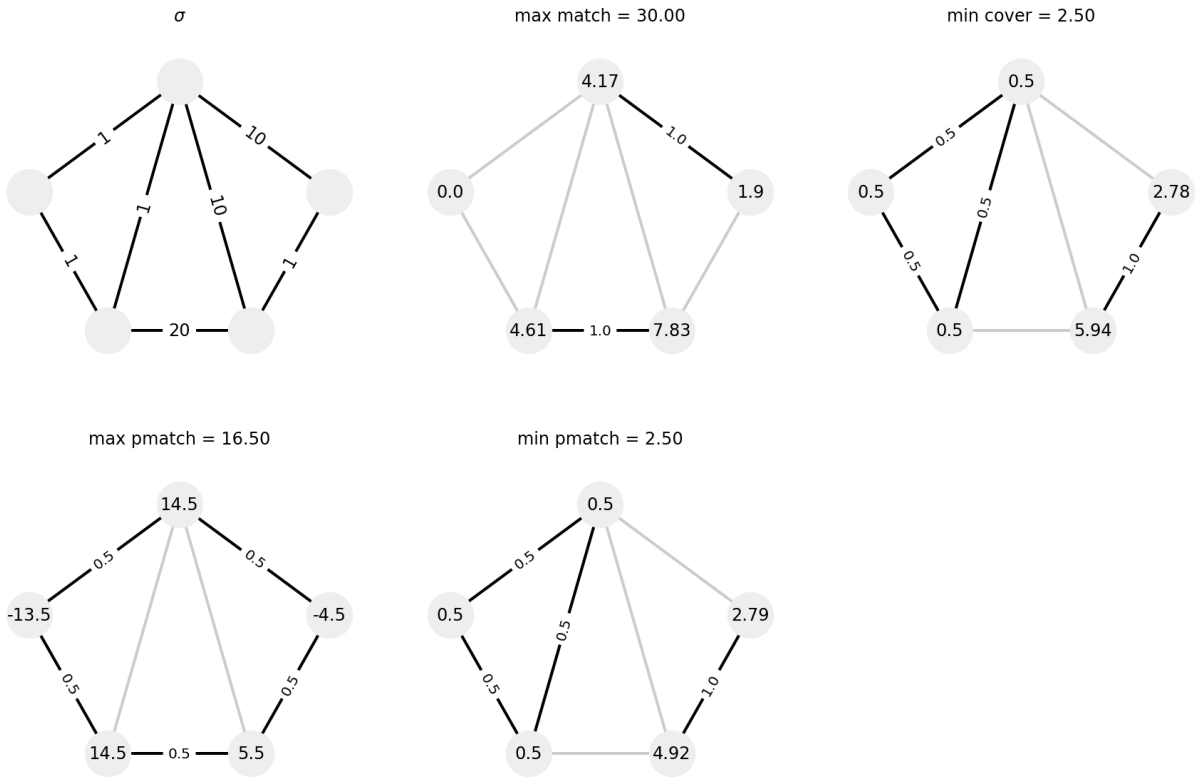


Figure 4.1: Non-bipartite graph solved using the LP-relaxed constraints.

4.3 Computational complexity

For a non-bipartite graph, the additional inequalities for the odd cardinality sets are the *blossom inequalities*. There are exponentially many of these inequalities, since their quantity is directly correlated with the size of a graph. In addition, some of them may be redundant, making non-bipartite graph problems even more complex for large graphs.

Using Theorems 4.1, 4.2, and 4.3 to find the blossom inequalities for the graph in Figure 4.1 is computationally reasonable since the number of vertices is low. Applying those

blossom inequalities in addition to the KKT conditions from Chapter 3 further constrains the problem to correctly show there is no possible perfect matching for the graph (illustrated in Figure 4.2).

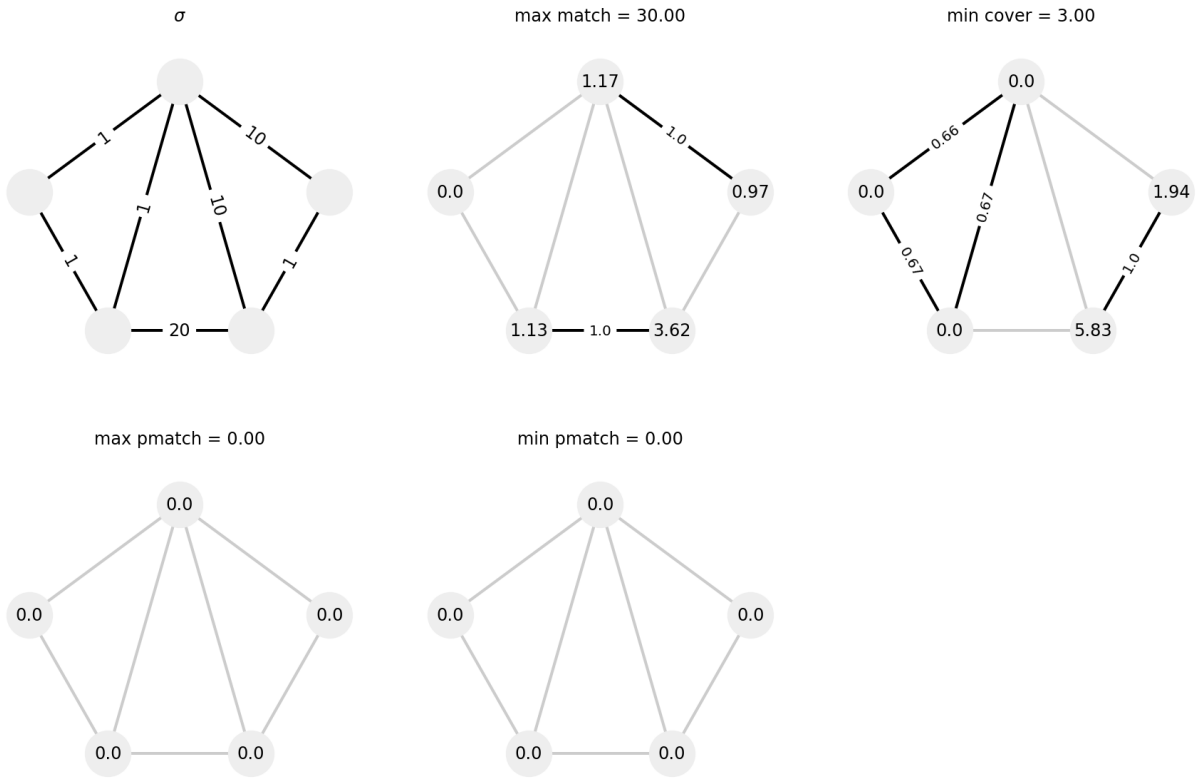


Figure 4.2: Non-bipartite graph solved using the LP-relaxed constraints and the blossom inequalities

Therefore, this example demonstrates how crucial the blossom inequalities are for non-bipartite graph optimization problems.

In general, adding all of the blossom inequalities to solve the non-bipartite cases make this problem hard again. Even trying to identify the most important set of the blossom inequalities is difficult since there are so many of them. Therefore, computations on non-bipartite graphs are still complex, but the bipartite graphs are tractable because they do not require the blossom inequalities and are equal to their linear programming relaxations.

Bibliography

[1] Behrend, R. E. Fractional perfect b-matching polytopes I: General theory. *Linear Algebra and its Applications 439*, 12 (2013), 3822–3858.

[2] Cook, William; Rohe, A. Computing minimum-weight perfect matchings. *INFORMS Journal on Computing 11*, 2 (Spring 1999), 138–148. Accepted by journal in September 1998.

[3] Edmonds, J. Maximum matching and a polyhedron with 0,1-vertices. *JOURNAL OF RESEARCH of the National Bureau of Standards - B. Mathematics and Mathematical Physics 69B*, 1 and 2 (January - June 1965), 125–130. Created December 1, 1964.

[4] Edmonds, J. Paths, trees, and flowers. *Canadian Journal of Mathematics 17* (1965), 449467.

[5] Harris, J. M., Hirst, J. L., and Mossinghoff, M. J. *Combinatorics and Graph Theory*, 2 ed. Undergraduate Texts in Mathematics. Springer, 2008.

[6] Schrijver, A. Short proofs on the matching polyhedron. *Journal of Combinatorial Theory 34*, 1 (1983), 104–108.

[7] Schrijver, A. *Combinatorial Optimization: Polyhedra and Efficiency*, vol. A of *Algorithms and Combinatorics*. Springer, 2003, pp. 318, 319, 462–464.

[8] Sinha, M. Lower bounds for approximating the matching polytope. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms* (Philadelphia, PA, USA, 2018), SODA '18, Society for Industrial and Applied Mathematics, pp. 1585–1604.