Lecture 1: Matchings

Ahmad Abdi

January 9, 2023

Abstract

Let $G = (V, E)$ be a graph. A *matching* is a subset of E consisting of pairwise vertex-disjoint edges. A matching is *perfect* if it *saturates* every vertex, i.e. if every vertex is covered by some edge of the matching. Does G have a perfect matching? If so, can we find it *efficiently*? Given prescribed edge weights, what is the minimum total weight of a perfect matching?

We address the first question in §2 by providing an elegant min-max formula for the maximum cardinality of a matching of a graph. The min-max formula does not lead lead to an efficient algorithm, so in §3, we provide a polynomial time algorithm for finding a perfect matching of a graph (if one exists), answering the second question. We then address the third question in §4 by providing a facet description for the incidence vectors of perfect matchings of a graph.

1 Kőnig's Theorem and Hall's Marriage Theorem

Let us start with two classical results in Graph Theory about matchings in bipartite graphs.

Let $G = (V, E)$ be a graph. Denote by $\nu(G)$ the maximum cardinality of a matching. What is $\nu(G)$? A simple upper bound on $\nu(G)$ comes from "vertex covers". A *vertex cover* is a subset of vertices incident with every edge of the graph, that is, a vertex cover is the complement of a stable set. Denote by $\tau(G)$ the minimum cardinality of a vertex cover. Every vertex cover contains a distinct vertex from every edge of a matching, so $\nu(G) \leq \tau(G)$. These two parameters are not always equal. For example, for a triangle, $\nu = 1$ while $\tau = 2$. However, for bipartite graphs, the two parameters are equal!

Theorem 1.1 (Kőnig's Theorem)**.** Let $G = (V, E)$ be a bipartite graph. Then the minimum cardinality of a vertex cover is equal to the maximum cardinality of a matching.

For the proof, we need a notion that will be useful throughout this lecture. Given a matching M, an M-*alternating circuit* is a circuit whose edges are alternately in and out of M. An M-*alternating path* is a path whose edges are alternately in and out of M. An M-*augmenting path* is an alternating path whose end vertices are unmatched. We shall drop the prefix M- from these two notions whenever there is no ambiguity. Observe that if P is an augmenting path, then the symmetric difference $M \triangle P$ is a matching of cardinality one larger than M.

Proof. Let $L \cup R$ be a bipartition of G. Pick a maximum matching M.

Claim. There exists a vertex cover K that consists only of matched vertices, and intersects every edge of M exactly once.

Proof of Claim. Let U be the set of unmatched vertices in L. Consider all M-alternating paths with an end in U. By construction, the inner nodes of all such paths are matched, and since M is a maximum matching, each such path has only one unmatched end, which by definition belongs to U. Let S_R (resp. S_L) be the set of all the nodes in R (resp. $L - U$) that belong to such an M-alternating path. Then S_L, S_R are comprised of matched vertices. By definition, $U \cup S_L$ has no neighbour in $R - S_R$ (or else, S_R would have been larger), and there is no matching edge between S_R and $L - (U \cup S_L)$ (or else, S_L would have been larger). This has two implications:

- $K := S_R \cup (L - (U \cup S_L))$ is a vertex cover,
- K intersects every edge of M at most, and therefore exactly, once.

By definition of U, $L - (U \cup S_L)$ consists of matched vertices, so K consists of matched vertices. Thus, K is the desired set. $\Diamond$

Since K consists only of matched vertices, and intersects every edge of M exactly once, it follows that $|K| = |M|$, so the minimum cardinality of a vertex cover is equal to the maximum cardinality of a matching. $\square$

As a consequence, we get the following classical result, which implies when a bipartite graph has a perfect matching:

Theorem 1.2 (Hall's Marriage Theorem)**.** Let G be a bipartite graph with bipartition $L \cup R$. Then exactly one of the following two statements holds:

1. G has a matching that saturates every vertex in L,
2. there exists a subset $A \subseteq L$ such that $|A| > |N(A)|$, where $N(A)$ denotes the set of neighbours of A.

In particular, if $|L| = |R|$, then G does not have a perfect matching if, and only if, (2) holds.

Proof. Clearly both statements cannot hold simultaneously. Assume that (1) does not hold. Let M be a maximum matching, and let K be a minimum vertex cover. By Kőnig's Theorem, $|M| = |K|$. In particular, every vertex of K is saturated, and K intersects every edge of M exactly once. Let $A := L - K$. As K is a vertex cover, its complement is a stable set, so $N(A) \subseteq K \cap R$. Every edge of M with an end in $K \cap R$ must have its other end in $L - K = A$, so $|A| \geq |K \cap R| \geq |N(A)|$. Moreover, since M does not saturate every vertex in L, it does not saturate some vertex in $L - K = A$, so $|A| > |K \cap R| \geq |N(A)|$, implying that (2) holds. $\square$

2 The Tutte-Berge formula

In the previous section, we gave a min-max formula for the maximum cardinality of a matching in bipartite graphs, which in turn led to a characterisation of when a bipartite graph has a perfect matching. In this section, we extend both of these results to arbitrary graphs.

Let $G = (V, E)$ be a graph. Given a subset $U \subseteq V$, define

$$\mathrm{def}(U) := \mathrm{oc}(G \setminus U) - |U|$$

where $\mathrm{oc}(H)$ denotes the number of odd (connected) components of a graph H, where "odd" refers to the parity of the number of vertices. The key idea is that if a matching saturates every vertex of an odd component of $G \setminus U$, then at least one of the vertices of the component must be matched with a vertex of U. Consequently, if $\mathrm{def}(U) > 0$, then no matching can ever saturate every vertex that belongs to an odd component of $G \setminus U$, implying in turn that there is no perfect matching. What's more, maximising the deficiency tells us exactly how many vertices a maximum matching is shy of for being a perfect matching. To elaborate, recall that $\nu(G)$ denotes the maximum cardinality of a matching of G. Let

$$\mathrm{def}(G) := \max\{\mathrm{def}(U) : U \subseteq V\}.$$

Lemma 2.1. Let $G = (V, E)$ be a graph, and choose $U \subseteq V$ such that $\mathrm{def}(U) = \mathrm{def}(G)$. Then the number of unsaturated vertices in every matching is at least $\mathrm{def}(G)$. Moreover, if equality holds, then the matching is maximum, and every vertex of U is saturated in every maximum matching of G.

Proof. Let M be a matching, and let $V_1, \ldots, V_k$ be the vertex sets of the odd components of $G \setminus U$. Then, for each $i \in [k]$, some vertex of V_i is either not saturated, or matched with a vertex of U. Subsequently, the number of unsaturated vertices of M is at least $k - |U|$, which is $\mathrm{def}(G)$. Moreover, if we have equality here, then M is a maximum matching, and every vertex of $V - (V_1 \cup \cdots \cup V_k)$, which includes U, is saturated, as required. $\square$

We are now ready to prove the following important result in Matching Theory:

Theorem 2.2 (Tutte-Berge Formula)**.** Let $G = (V, E)$ be a graph. Then for every maximum matching, the number of unsaturated vertices is equal to $\mathrm{def}(G)$. That is,

$$\nu(G) = \frac{1}{2}\left(|V| - \mathrm{def}(G)\right).$$

In particular, G has a perfect matching if, and only if, $\mathrm{oc}(G \setminus U) \leq |U|$ for all $U \subseteq V$.

Proof. We proceed by induction on $|E|$. The base case $|E| = 0$ is obvious. For the induction step, assume that $|E| \geq 1$. It can be readily checked that the number of unsaturated vertices in every matching is at least $\mathrm{def}(G)$. To prove the other direction, pick an edge e with ends u, v.

If an end of e, say u, is saturated in every maximum matching, then $\nu(G \setminus u) = \nu(G) - 1$. By the induction hypothesis, for a maximum matching M' of $G \setminus u$, and a subset $U' \subseteq V - \{u\}$, the number of M'-unsaturated vertices in $G \setminus u$ is $\mathrm{oc}(G \setminus u \setminus U') - |U'|$. Let $U := U' \cup \{u\}$, and let M be a maximum matching of G. By assumption, $|M| = |M'| + 1$, so the number of M-unsaturated vertices of G is

$$\mathrm{oc}(G \setminus u \setminus U') - |U'| - 1 = \mathrm{oc}(G \setminus U) - |U| = \mathrm{def}(U) \le \mathrm{def}(G),$$

thereby proving the other direction.

Otherwise, there exists a maximum matching M_u that does not saturate u, and a maximum matching M_v that does not saturate v. The maximality of the matchings implies that M_u saturates v, and M_v saturates u. (In particular, M_u, M_v are distinct matchings.)

Consider the symmetric difference $F := M_u \triangle M_v$. Then F is the vertex-disjoint union of alternating paths and alternating circuits, where each alternating path starts from an M_u-unsaturated vertex and ends at an M_v-unsaturated vertex, because both M_u, M_v are maximum matchings. Since u is M_v-saturated and M_u-unsaturated, it is the end of an alternating path P_u. Similarly, since v is M_u-saturated and M_v-unsaturated, it is the end of an alternating path P_v. These two alternating paths must be the same, for if not, then $(M_u \triangle P_v) \cup \{e\}$ would be a matching of larger cardinality than M_u, which is a contradiction.

We just showed $P_u = P_v$. In particular, $C := \{e\} \cup P_u$ is an odd circuit that includes $\frac{|C|-1}{2}$ many edges from each of M_u, M_v. Let G' be the graph obtained by *shrinking* C, i.e. by contracting the edges of C. Then $M_u - C$ is a matching of G', so $\nu(G') \ge \nu(G) - \frac{|C|-1}{2}$. In fact, we must have equality here, because every matching M' of G' can be extended to a matching of G by adding $\frac{|C|-1}{2}$ appropriate edges from C to M', so

$$\nu(G') = \nu(G) - \frac{|C|-1}{2}.$$

In particular, $M_u - C$ is a maximum matching of G', one that does not saturate the shrunken vertex – call it w.

Observe that the number of M_u-unsaturated vertices of G is equal to the number of $(M_u - C)$-unsaturated vertices of G'. We know what the latter is: By the induction hypothesis, there exists a subset $U \subseteq (V - V(C)) \cup \{w\}$ such that the number of $(M_u - C)$-unsaturated vertices of G' is $\mathrm{oc}(G' \setminus U) - |U|$. By Lemma 2.1, U cannot contain $(M_u - C)$-unsaturated vertices of G', so $w \notin U$, implying in turn that $U \subseteq V$. Consider now the components of $G \setminus U$: There are those that correspond identically to the components of $G' \setminus U$ not containing w. There is one more component, which arises from the component of $G' \setminus U$ containing w after de-contracting w to the odd circuit C; notice that these two components have the same number of vertices modulo 2, because $|C|$ is odd. Subsequently, $\mathrm{oc}(G \setminus U) = \mathrm{oc}(G' \setminus U)$. Consequently, the number of M_u-unsaturated vertices G is equal to $\mathrm{oc}(G \setminus U) - |U| = \mathrm{def}(U) \le \mathrm{def}(G)$, thereby proving the other direction.

In both cases, we proved that the number of unsaturated vertices in a maximum matching was less than or equal to $\mathrm{def}(G)$. This completes the induction step. □

An important consequence of Theorem 2.2 is that the problem of deciding whether a graph G has a perfect matching belongs to $\mathrm{NP} \cap \mathrm{co-NP}$: If $G = (V, E)$ has a perfect matching, then a polynomial certificate can be provided, namely the perfect matching itself. Otherwise, if G does not have a perfect matching, then another polynomial certificate can be provided, namely, a subset $U \subseteq V$ for which $\mathrm{def}(U) > 0$. This complexity result *suggests* that deciding whether a graph has a perfect matching should not be NP-complete, which as we will see is indeed the case. In fact, we will prove that this problem belongs to P.

3 Edmonds' Blossom Algorithm

In this section, we provide an elegant algorithm that, given a graph, runs in time polynomial in the size of the graph and outputs either a perfect matching or certifies that none exists. The utter simplicity of the algorithm allows it to be extended for other purposes. For example, a slight extension of it gives a polynomial algorithm for finding a maximum matching (Exercise 6). One can also combine it with a primal-dual linear programming technique and obtain a polynomial algorithm for finding a minimum weight perfect matching (see §4 for more). It also gives an algorithmic characterisation of Theorem 2.2 for the existence of a perfect matching.

Input. Let $G = (V, E)$ be a graph, and let M be a matching. If M is perfect, then we are done. Otherwise, let r be a vertex not saturated by M, and let T be an *alternating tree* rooted at r.

Output. A perfect matching, or a deficient subset of V certifying that G has no perfect matching.

Alternating Tree. An M*-alternating tree*, or simply an *alternating tree* T is a subgraph of G satisfying the following statements:

1. T is a rooted tree with root r,
2. r is the only vertex of the tree that is not saturated by M,
3. for every vertex v of T, the unique rv-path is an alternating path.

We may then partition the vertices of T into two parts $A(T), B(T)$, where $A(T)$ consists of those vertices v where the rv-path in T has an odd number of edges, and $B(T)$ consists of the remaining vertices. Notice that $r \in B(T)$. We ask that alternating trees satisfy one more condition:

4. every leaf of T belongs to $B(T)$.

Consequently,

5. every vertex in $A(T)$ has a unique child in T, with whom it is joined via an edge in M.

Observe that every vertex of T, other than r, is matched by an edge in $M \cap E(T)$ to another vertex in T. Subsequently,

6. $|B(T)| = |A(T)| + 1$.

Subroutines. There are four subroutines depending on types of edges outside of T incident with a vertex in $B(T)$.

augment matching: There is an edge e incident with a vertex in $B(T)$ whose other end is outside of T and is not saturated by M.

Assume that e has ends $v \in B(T)$ and $u \notin V(T)$. Then the rv-path in T, together with the edge e, gives an augmenting path, which we use to augment M to a matching whose cardinality is larger by one.

In this case, we re-initialise the algorithm with the new matching.

extend tree: There is an edge e incident with a vertex in $B(T)$ whose other end is outside of T and is saturated by M.

Assume that e has ends $v \in B(T)$ and $u \notin V(T)$. Then u is saturated by an edge $f \in M$. In this case, we add e, f and the ends of f to the tree T to get a larger alternating tree.

In this case, we re-initialise the algorithm with the same matching but with the larger alternating tree.

frustrated tree: If every edge in $E - E(T)$ incident with $B(T)$ has its other end in $A(T)$, then we say that the alternating tree is *frustrated*.

Lemma 3.1. If T is frustrated, then $\mathrm{def}(A(T)) > 0$, and in particular, G has no perfect matching.

Proof. Every edge of T incident with $B(T)$ has its other end in $A(T)$, so our assumption implies that every edge of G incident with $B(T)$ has its other end in $A(T)$. In particular, $B(T)$ is a stable set, and each vertex of it forms an odd component of $G \setminus A(T)$. As a result, $\mathrm{def}(A(T)) = \mathrm{oc}(G \setminus A(T)) - |A(T)| \geq |B(T)| - |A(T)| = 1$, as claimed. □

Thus, in this case, we declare that G has no perfect matching, and output $A(T)$ as a deficient subset.

shrink blossom: Otherwise, there is an edge e in $E - E(T)$ incident with $v \in B(T)$ whose other end is $u \in B(T)$. In this case, we get an odd circuit in the tree C called a *blossom*.

We then shrink the blossom C, and re-run the algorithm on $G' := G/C$ with the matching $M' := M - C$ and the tree T/C. Observe that T/C is indeed an M'-alternating tree, and since T satisfies (5), the vertex of G' corresponding to C, call it w, must belong to $B(T/C)$. Unlike the previous subroutines, there are a couple of things to be wary of here.

If at some iteration we augment the matching M' in G', then we stop. The oddness of C allows us to augment M in G accordingly, and therefore re-initialise from G with the new matching.

Otherwise, the tree T/C is extended to a frustrated tree T' of G'. Then Lemma 3.1 tells us that G' has no perfect matching. In fact, since $w \in B(T')$ and therefore $A(T') \subseteq V$, we can argue that G has no perfect matching:

Lemma 3.2. If T' is frustrated in G', then $\mathrm{def}_G(A(T')) > 0$, and in particular, G has no perfect matching.

Proof. Note that every vertex of $B(T')$ forms an odd component of $G' \setminus A(T')$. Thus, every vertex of $B(T') \setminus \{w\}$ forms an odd component of $G \setminus A(T')$, and as $|C|$ is odd, C forms an odd component of $G \setminus A(T')$. Thus, $\mathrm{def}_G(A(T')) = \mathrm{oc}(G \setminus A(T')) - |A(T')| \geq |B(T')| - |A(T')| = 1$, as required. $\square$

Thus, in this case, we declare that G has no perfect matching, and output $A(T')$ as a deficient subset.

We have already shown the correctness of the algorithm. Its running time can be readily checked to be polynomial:

Theorem 3.3. The Blossom Algorithm on $G = (V, E)$ terminates after $O(|V|)$ augmentation subroutines, $O(|V|^2)$ tree extension subroutines, $O(|V|^2)$ shrinking subroutines, and at most one frustrated tree subroutine.

Proof. Clearly, the number of augmentation steps is at most $\frac{|V|}{2}$. In between consecutive augmentation steps, we can have up to $|V| - 1$ tree extensions and $O(|V|)$ shrinking steps, thereby proving the theorem. $\square$

4 Minimum-weight perfect matchings

Let $G = (V, E)$ be a graph that has a perfect matching, and let $w \in \mathbb{R}^E$ be edge weights. In this section, we develop the key ideas of an algorithm for finding a minimum weight perfect matching, where the weight of a matching M is equal to $w(M) := \sum_{e \in M} w_e$.

Any successful algorithm needs a stopping criterion, that is, it must be able to detect whether a given perfect matching has minimum weight. This is where linear programming duality comes to the rescue.

The goal is to formulate the minimum weight perfect matching problem as a linear program. That way, the complementary slackness conditions would give us a certificate of optimality.

A first, good attempt is the following linear program:

$$\begin{array}{lrcll} \min & w^\top x & & & \\ \text{s.t.} & x(\delta(v)) & = & 1 & v \in V \\ & x_e & \geq & 0 & e \in E \end{array}$$

While guaranteed to model the problem for bipartite graphs (see Exercise 11), this linear program is too weak to model the problem in general. For example, consider the graph displayed in Figure 1, which has a perfect matching, whose edge labels represent the edge weights. While the minimum weight of a perfect matching is 1, the linear program has optimal value 0, obtained by assigning $x^\star_e = \frac{1}{2}$ to each edge e of weight 0, and $x^\star_f = 1$ to each edge f of weight 1. We therefore need to strengthen the linear program above by adding new inequalities satisfied by perfect matchings.

Consider the cut $\delta(\{1, 2, 3\})$. Observe that $x^\star$ takes a value of 0 on the edges belonging to this cut, but every perfect matching would have to an edge belonging to this cut, because both shores have an odd number of vertices. This observation motivates the addition of the following *odd cut inequalities* (also called *blossom inequalities*), valid for all perfect matchings, to the linear program:

$$x(\delta(U)) \geq 1 \quad U \subseteq V, |U| \text{ is odd.}$$

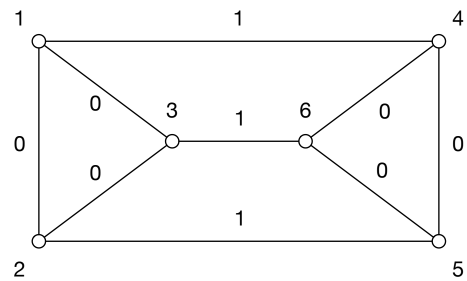

Figure 1: The edge labels represent the edge weights.

It turns out that by adding these odd cut inequalities to the linear program, we correctly model the minimum weight perfect matching problem. To prove this, we need a few ingredients.

4.1 The Edmonds-Johnson Theorem

Let $G = (V, E)$ be an arbitrary graph, and let T be a nonempty even cardinality subset of V. A T-*cut* is a cut of the form $\delta(U)$ where $|U \cap T|$ is odd. A T-*join* is a subset $J \subseteq E$ whose odd-degree vertices coincide with T.

Lemma 4.1. Let $G = (V, E)$ be an arbitrary graph, and let T be a nonempty even cardinality subset of V. Then the following statements hold:

1. Every T-join and T-cut have an odd number of edges in common. In particular, every T-join and T-cut intersect.

2. If an edge subset J intersects every T-cut, then J contains a T-join.

Proof. Exercise. (**Hint.** For part 2, first solve Exercise 7.) □

Consider the polyhedron $P(G, T) := \{x \in \mathbb{R}^E_+ : x(B) \geq 1, B \text{ is a } T\text{-cut}\}$. Lemma 4.1 (1) implies that the incidence vector of every T-join belongs to $P(G, T)$. Moreover, it follows from Lemma 4.1 (2) implies that every integral vertex of $P(G, T)$ is the incidence vector of an inclusionwise minimal T-join of G (see Exercise 9). We will show that every vertex of $P(G, T)$ is indeed integral, thereby giving a full facet description of the T-join polyhedron.

We will need the following lemma:

Lemma 4.2. Let $G = (V, E)$ be a graph, and let $T \subseteq V$ be nonempty and of even cardinality. Let $\tilde{x}$ be a vertex of the polyhedron $\{x \in \mathbb{R}^E_+ : x(\delta(v)) \geq 1, v \in T\}$, and let $\tilde{G}$ be the subgraph of G induced on the edge set $\{e \in E : \tilde{x}_e > 0\}$. Then every connected component of $\tilde{G}$ is either

i. an odd circuit C whose vertices are in T and edges $\tilde{x}_e = \frac{1}{2}$, or

ii. a star whose vertices, except possibly its centre, are in T and edges $\tilde{x}_e = 1$.

Proof. Exercise. (**Hint.** What can you say about the number of edges of $\tilde{G}$? Use linear algebra to provide an upper bound, and graph connectivity to provide a lower bound.) □

Theorem 4.3 (Edmonds-Johnson Theorem)**.** Let $G = (V, E)$ be a graph, and let $T \subseteq V$ be nonempty and of even cardinality. Then $P(G, T)$ is an integral polyhedron. In particular, every vertex of $P(G, T)$ is the incidence vector of an inclusionwise minimal T-join of G.

Proof. We proceed by induction on $|V| \geq 2$. The induction step $|V| = 2$ is obviously true. For the induction step, assume that $|V| \geq 3$, and let $\tilde{x}$ be a vertex of $P(G, T)$.

If $\tilde{x}$ is also a vertex of $\{x \in \mathbb{R}^E_+ : x(\delta(v)) \geq 1, v \in T\}$, then by Lemma 4.2, then every connected component of the subgraph of G induced on $\{e \in E : \tilde{x}_e > 0\}$ is either type i or type ii. However, the first type of connected components is not possible, because $\tilde{x}$ must satisfy the T-cut inequality $\tilde{x}(V(C)) \geq 1$. This implies that $\tilde{x}$ is an integral vertex, as required.

Otherwise, $\tilde{x}(\delta(U)) = 1$ for a T-cut $\delta(U)$ such that $|U|, |V - U| \neq 1$. Let G_1 be the graph obtained from G after shrinking U to a single vertex u, and let $T_1 := (T - U) \cup \{u\}$. Similarly, let G_2 be the graph obtained from G after shrinking $V - U$ to a single vertex v, and let $T_2 := (T \cap U) \cup \{v\}$. Let $\tilde{x}^1, \tilde{x}^2$ be the restrictions of $\tilde{x}$ to $E(G_1), E(G_2)$, respectively. It can be readily checked that $\tilde{x}^i \in P(G_i, T_i)$, for each $i = 1, 2$. By the induction hypothesis, each $P(G_i, T_i), i = 1, 2$ is an integral polyhedron, so for each $i = 1, 2$,

$$\tilde{x}^i \geq \sum_{J \text{ a } T_i\text{-join of } G_i} \lambda^i_J \chi_J$$

where $\lambda^i \geq \mathbf{0}$ and $\sum_J \lambda^i_J = 1$. Moreover, since $\tilde{x}^i(\delta(U)) = \tilde{x}(\delta(U)) = 1$, and $|J \cap \delta(U)| \geq 1$ for each T_i-join of G_i, we must have equality above for the entries corresponding to the edges in $\delta(U)$. In particular, $|J \cap \delta(U)| = 1$ for all T_i-joins J such that $\lambda^i_J > 0$. Observe that if J_1 is a T_1-join of G_1, J_2 is T_2-join of G_2, and $J_1 \cap \delta(U) = \{e\} = J_2 \cap \delta(U)$, then $J_1 \cup J_2$ is a T-join of G, i.e. we can *glue* the two sets together and obtain a T-join of G. By gluing T_1- and T_2-joins carefully along the edges of $\delta(U)$, we obtain that

$$\tilde{x} \geq \sum_{J \text{ a } T\text{-join of } G} \lambda_J \chi_J$$

where $\lambda \geq \mathbf{0}$ and $\sum_J \lambda_J = 1$. As $\tilde{x}$ is a vertex of $P(G, T)$, and as $\chi_J \in P(G, T)$ for every T-join J, we must have equality above, and exactly one entry of λ must be nonzero, that is, $\tilde{x}$ is integral, thereby completing the induction step. □

4.2 An optimality certificate

Since every nonempty face of an integral polyhedron is also integral, Theorem 4.3 has the following immediate consequence:

Corollary 4.4. Let $G = (V, E)$ be a graph that has a perfect matching. Then the feasible region of

$$(P) \quad \begin{array}{lrcll} \min & w^\top x & & & \\ \text{s.t.} & x(\delta(v)) & = & 1 & v \in V \\ & x(\delta(U)) & \geq & 1 & U \subseteq V, |U| \text{ is odd} \\ & x_e & \geq & 0 & e \in E \end{array}$$

is an integral polyhedron. That is, for all $w \in \mathbb{R}^E$, the minimum weight of a perfect matching is equal to the optimal value of (P).

Let $G = (V, E)$ be a graph with a perfect matching. An *odd cut* is a cut of the form $\delta(U)$ where $|U|$ is odd, i.e. a V-cut. Consider the dual linear program:

$$(D) \quad \begin{array}{lrcll} \max & \sum (y_v : v \in V) + \sum (y_B : B \text{ is an odd cut}) & & & \\ \text{s.t.} & y_u + y_v + \sum (y_B : B \text{ is an odd cut containing } e) & \leq & w_e & e = uv \in E \\ & y_B & \geq & 0 & \text{for every odd cut } B \end{array}$$

By the Complementary Slackness conditions, a perfect matching M has minimum weight if, and only if, for its incidence vector x, there exists a dual feasible solution y such that

$$x_e > 0 \text{ implies } \sum (y_B : B \text{ is an odd cut containing } e) = w_e - y_u - y_v$$

$$y_B > 0 \text{ implies } |B \cap M| = 1.$$

We have therefore obtained a stopping criterion for a successful algorithm for computing the minimum weight of a perfect matching. One can now bootstrap the Blossom Algorithm with a primal-dual approach for finding a minimum weight perfect matching. The interested reader is referred to an excellent treatment of this algorithm in [1], Chapter 5. Alternatively, one can use the ellipsoid method directly on the linear program (P), but given that we have exponentially many constraints, we would need a polynomial time *separation oracle*, which amounts to finding in polynomial time a minimum weight odd cut of a graph [4]. We shall discuss this idea in more detail in the final lecture.

Exercises

1. Let $G = (V, E)$ be a graph, and let M be a matching. Prove that there are at least $\nu(G) - |M|$ vertex-disjoint augmenting paths.

2. Let $G = (V, E)$ be a graph, and let M be a matching of cardinality at most $\nu(G) - \sqrt{\nu(G)}$. Prove that there exists an augmenting path that picks at most $\sqrt{\nu(G)}$ edges from M.

3. An *edge cover* of a graph $G = (V, E)$ is an edge subset that intersects every $\delta(v), v \in V$. Prove that the minimum cardinality of an edge cover is equal to $|V| - \nu(G)$, where $\nu(G)$ is the maximum cardinality of a matching.

4. Let $r \geq 2$ be an integer. Prove that every r-regular bipartite graph has r disjoint perfect matchings.

5. For an integer $r \geq 2$, an *r-graph* is an r-regular graph with an even number of vertices where every odd cut has at least r edges. Prove that every edge of an r-graph belongs to a perfect matching.

6. Extend Edmonds' Blossom Algorithm to an algorithm that, given a graph, runs in time polynomial in the size of the graph and outputs a maximum matching. As a consequence, give an algorithmic proof of Theorem 2.2.

7. Let $G = (V, E)$ be a connected graph, and let T be a nonempty even cardinality subset of V. Prove that every spanning tree of G contains a T-join.

8. Prove Lemma 4.1 parts (1) and (2).

9. Let $G = (V, E)$ be a graph, and let T be a nonempty even cardinality subset of V. Prove that the (integral) vertices of $P(G, T)$ are in correspondence with $\{\chi_J : J \text{ is an inclusionwise minimal } T\text{-join}\}$.

10. Prove Lemma 4.2.

11. Let $G = (V, E)$ be a bipartite graph, and let $w \in \mathbb{R}^E$. Consider the linear program

$$\begin{array}{lrcll} \min & w^\top x & & & \\ \text{s.t.} & x(\delta(v)) & = & 1 & v \in V \\ & x_e & \geq & 0 & e \in E. \end{array}$$

Prove the following statements:

(a) Prove that G has a perfect matching if, and only if, the linear program has a feasible solution.

(b) Prove that the minimum weight of a perfect matching is equal to the optimal value of the linear program.

12. Let $G = (V, E)$ be a graph, let $w \in \mathbb{R}^E_+$, and let T be a nonempty even cardinality subset of V. Let $\delta(U), \delta(W)$ be minimum weight T-cuts, where U, W intersect and neither one is contained in the other. Prove that either $\delta(U \cap W), \delta(U \cup W)$ or $\delta(U - W), \delta(W - U)$ are minimum weight T-cuts.

13. Consider the graph G with edge weights w (which are possibly negative). We are interested in solving the *minimum weight matching problem* on the pair (G, w). Create a second copy $G^\star$ of G with the same edge weights, and add an edge of weight 0 between $u, u^\star$ for every vertex u of G. Explain how solving the minimum weight *perfect* matching problem on this new instance can lead to a solution of our problem.

14. Let $G = (V, E)$ be a graph, let $s, t \in V$ be distinct vertices, and let $w \in \mathbb{R}^E_+$. An *odd st-path* is an st-path with an odd number of edges. Similarly, we can define an *even st-path*.

(a) Find a minimum weight odd st-path by reducing it to the minimum weight perfect matching problem. (**Hint.** Start with the construction in Exercise 13, then delete $s^\star, t^\star$.)

(b) Find a minimum weight even st-path by a reducing it to the minimum weight odd path problem between a designated terminal pair.

15. Let $G = (V, E)$ be a plane 3-graph (see Exercise 5 for the definition of an r-graph). Then the following statements are equivalent:

   (i) G has three disjoint perfect matchings,

   (ii) G has two disjoint V-joins,

   (iii) G has a 4-face-colouring, that is, one can colour each face using one of four colours so that faces that share an edge get different colours.

16. Let $G = (V, E)$ be a bipartite graph with bipartition $L \cup R$, and let $b \in \mathbb{Z}^V_{\geq 0}$ such that $b(L) = b(R)$. A *perfect b-matching* is a vector $x \in \mathbb{Z}^E_{\geq 0}$ such that $x(\delta(v)) = b_v$ for every vertex $v \in V$. Prove that exactly one of the following statements holds:

   (i) G has a perfect b-matching,

   (ii) there exists a subset $A \subseteq L$ such that $b(A) > b(N(A))$.

17. A *simple perfect b-matching* is a subset $J \subseteq E$ whose incidence vector is a perfect b-matching. Give an example of a bipartite instance (G, b) with a perfect b-matching but without a simple one. Conclude that Exercise 16 does not characterise the existence of a simple perfect b-matching.

Acknowledgements

The original description of Edmonds' Blossom Algorithm can be found in [3]. The presentation of §4.1 follows [2], Chapter 2. Further references and citations can be found in [1, 2].

References

[1] W. Cook, W. Cunningham, W. Pulleyblank, and A. Schrijver. *Combinatorial Optimization*. Wiley, 1997.

[2] G. Cornuéjols. *Combinatorial Optimization: Packing and Covering*. Society for Industrial and Applied Mathematics, 2001.

[3] J. Edmonds. Paths, trees, and flowers. *Can. J. Math.*, 17(449–467), 1965.

[4] M. W. Padberg and M. R. Rao. Odd minimum cut-sets and b-matchings. 7(1):67–80, 2021/02/14/ 1982.