Combinatorial Optimization

Dozent: Stephan Held

December 1, 2022

Contents

0 Organization

- Prerequisites
  - Basic knowledge of graph algorithms
  - Linear Programming (LP Duality)
  - Programming skills in C++
- Exam
  - Qualification requires 50% of the points in theoretical & programming exercises
  - Oral exam
- Books
  - "Combinatorial Optimization", Korte & Vygen
  - "Understanding & Using Linear Programming", B. Gärtner, J. Matouset
  - Skript (theorems & definitions)
  - Further book recommendations are on the website

1 Matchings

1.1 Introduction

Definition 1.1.

1. A *matching* M in a graph $G = (V, E)$ is a set of pairwise disjointed edges, i.e. they don't have a common endpoint.

   $\nu(G) :=$ max. cardinality of a matching in G

2. An *edge cover* C of a graph $G = (V, E)$ is a subset of E s.t. $V = \bigcup_{e \in C} e$.

   $\zeta(G) :=$ min. cardinality of an edge cover in G

3. A matching is called *perfect* (or *1-factor*) if it is an edge cover

4. $v \in V$ with $v \in e \in M$ is called *M-covered*

5. $v \in V$ is called *M-exposed* if it is not M-covered

Definition 1.2.

1. A *stable set* (independent set) S is a set of pairwise non-adjacent vertices.

   $\alpha(G) :=$ max. cardinality of a stable set

2. A *vertex cover* C is a subset of V s.t. $E = \bigcup_{\{x,y\}\in E,\ x\in G}\{x,y\}$
   $\tau(G) :=$ min. cardinality of a vertex cover

Lemma 1.3.

1. $\alpha(G) + \tau(G) = |V|$
2. $\nu(G) + \zeta(G) = |V|$ *if G has no isolated vertices*
3. $\zeta(G) = \alpha(G)$ *if G is bipartite and has no isolated vertices*

Problem. Cardinalty Matching Problem
Input: Graph $G = (V, E)$
Task: Find a maximum cardinality matching

Problem. Maximum Weight Matching Problem (MWMP)
Input: Graph G, $c : E \to \mathbb{R}$
Task: Find a matching M maximizing $c(M)$

Problem. Minimum Weight Perfect Matching (MWPMP)
Input: Graph G, $c : E \to \mathbb{R}$
Task: Find a perfect matching of minimum weight or decide that no perfect matching exists in G

Lemma 1.4. *The MWMP is equivalent to the MWPMP (i.e. there exists a transformation with linear complexity)*

Proof. Given a MWPMP instance (G, c), define $c' := K - c$ ($K := 1 + \sum_{e\in E} |c(e)|$).
$\Rightarrow$ Any maximum weight matching is a maximum cardinality matching

Given a MVMP instance (G, c), define G' as 2 copies of G where the 2 copies of a vertex are joined by an edge.
$\Rightarrow$ G' has a perfect matching. Define:

$$c'(e) := \begin{cases} -c(e) & \text{if } e \text{ is in the first copy} \\ 0 & \text{else} \end{cases}$$

A minimum weight perfect matching in G' gives us a maximum weight matching in G. □

Definition 1.5. Let $G = (V, E)$ be a graph and $M \subseteq E$ a matching in G. A path P is *M-alternating* if its edges are alternatingly in and not in M. If both end points of this path are M-exposed, P is an *M-augmenting* path.

Lemma 1.6. *Given a matching M in G and an inclusion-wise maximal M-alternating path P,*

$$M\Delta P := M \setminus P \cup P \setminus M$$

is a matching. If P is M-augmenting, then $|M\Delta P| = |M| + 1$.

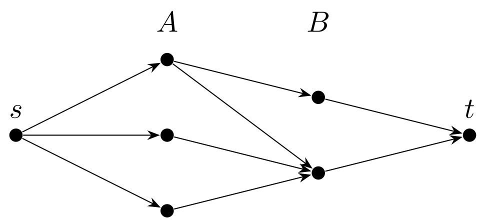


Figure 1: Example of the construction in Theorem 1.8

Theorem 1.7 (Petersen 1891, Berge 1957)**.** *Augmenting Path Theorem Given a graph $G = (V, E)$ and a matching M in G:*

$$|M| = \nu(G) \Leftrightarrow \nexists M\text{-augmenting path } P \text{ in } G$$

Proof.

"$\Rightarrow$": Clear

"$\Leftarrow$": Assume there exists a matching M' with $|M'| > |M|$. Let $G' := (V, M\Delta M')$.
$\Rightarrow |\delta_{G'}(v)| \leq 2\ \forall v \in V$
$\Rightarrow G'$ is the union of disjoint circuits and paths
$\Rightarrow$ all circuits are even and have the same number of edges from M and M'
$\Rightarrow \exists$ a path P in G' starting and ending with an edge in M'
$\Rightarrow P$ is an alternating path

□

1.2 Bipartite Matching

Theorem 1.8 (König 1931)**.** *If G is bipartite, then $\nu(G) = \tau(G)$*

Proof. Add vertices s and t edges between them to all vertices of the respective partition. Direct all edges from s to t. Then $\nu(G)$ is maximum number of disjoint s-t-paths. Menger $\Rightarrow$ This is equal to the minimum number of vertices that disconnect t from s. □

Theorem 1.9 (Hall 1935)**.** *Let $G = (A \dot\cup B, E)$ be a bipartite graph. Then:*

$$G \text{ has a matching covering } A \Leftrightarrow |\Gamma(X)| \geq |X| \quad \forall X \subseteq A$$

Corollary 1.10. *Marriage Theorem*

$$|\Gamma(X)| \geq |X|\ \ \forall X \subseteq A \text{ and } |A| = |B| \Leftrightarrow G \text{ has a perfect matching}$$

Definition 1.12. The MWPMP for bipartite graphs is called *Assignment Problem.*

Theorem 1.13. *The Assignment Problem can be solved in time $O(nm + n^2 \log m)$.*

Proof. Use the Successive Shortest Paths algorithm in an auxiliary graph. □

1.3 The Tutte Matrix & Randomized Matching

Definition 1.14. Let G be a simple, undirected graph. Let G' be an orientation of G and $(X_e)_{e \in E(G)}$. The *Tutte matrix* is defined as

$$T_G(X) \coloneqq (T^*_{vw})_{v,w \in V(G)}$$

where

$$t^*_{vw} \coloneqq \begin{cases} X_{\{v,w\}} & \text{if } (v,w) \in E(G) \\ -X_{\{v,w\}} & \text{if } -(v,w) \in E(G) \\ 0 & \text{else} \end{cases}$$

Remark 1.15. $T_G(X)$ is shew-symmetric (i.e. $T_G(X) = -(T_G(X))^t$). $\operatorname{rank}(T_G(X))$ is independent of the orientation of G. $\det(T_G(X))$ is a polyomial in X.

Theorem 1.16 (Tutte)**.** *A simple graph G has a perfect matching* $\Leftrightarrow \det(T_G(X)) \neq 0$

Proof. Let $V(G) = \{v_1, \dots, v_n\}$ and S_n be the permutation group.

$$\det T_G(X) = \sum_{\pi \in S_n} \operatorname{sgn}\pi \cdot \prod_{i=1}^{n} t^*_{v_i, v_{\pi(i)}}$$

Let $S'_n \coloneqq \{\pi \in S_n \mid \prod_{i=1}^n t^*_{v_i, v_{\pi_i}} \neq 0\}$. Each $\pi \in S_n$ corresponds to a digraph $H_\pi \coloneqq (V(G), \{(v_i, v_{\pi(i)}) \mid i \in [n]$[1]$\})$. We have $|\delta^+(v) = 1 = |\delta^-(v)|| \ \forall v \in V(H_\pi) \Rightarrow H_\pi$ is the union of disjoint circuits. If $\pi \in S'_n$, then $H_\pi \subset \overleftrightarrow{G'}$.

If there exists $\pi \in S'_n$ s.t. H_π is a collection of even circuits, then this immediately yields a perfect matching in G (using every second edge of each circuit).

Otherwise, $\forall \pi \in S'_n$, H_π contains an odd circuit. Let $r(\pi) \in S'_n$ arise from π by reversing edges on the unique odd circuit containing a vertex with minimum index $\Rightarrow r(r(\pi)) = \pi$ and $\operatorname{sgn}(\pi) = \operatorname{sgn}(r(\pi))$. The second part is true since for reversing an odd cycle, we need an even number of swaps. Let $v_{i_1}, \dots, v_{i_{2k+1}}$ be the "first" odd circuit. Then $r(\pi)$ is attained by $2k$ swaps: For $j = 1, \dots, k$ swap $(\pi(i_{2j-1}), \pi(i_{2k}))$ and $(\pi(i_{2j}), \pi(i_{2k+1}))$.

[1] This is an abbreviation for $\{1, \dots, n\}$.

$\prod_{i=1}^{n} t^*_{v_i v_{\pi(i)}} = -\prod_{i=1}^{n} t^*_{v_i v_{r(\pi(i))}}$ since there is an odd number of sign changes to t^*. $\Rightarrow \det(T_G(X)) = 0$. We have shown that if G has no perfect matching, then $\det T_G(X) = 0$.

Assume that G has a perfect matching M. Define π as $\pi(i) = j, \pi(j) = i$ where $\{i, j\} \in M$. $\Rightarrow \prod_{i=1}^{n} t^*_{v_i v_{\pi(i)}} = \prod_{e \in M} -X_e^2$ cannot be canceled out. In particular, $\det T_G(X) \neq 0$. $\square$

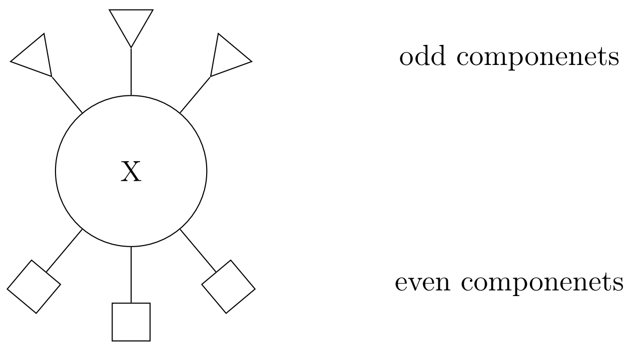
Remark 1.17. Picking $X' \in [0,1]^{E(G)}$ at random, we almost surely have (since the zero set of a non-zero polynomial is a set of measure zero):

$$\det T_G(X') \neq 0 \Leftrightarrow G \text{ has a perfect matching}$$

Theorem 1.18 (Lovász 1979)**.** *Let G be a simple graph and $X \in [0,1]^{E(G)}$ chosen randomly. Then almost surely* $\operatorname{rank}(T_G(X)) = 2\nu(G)$.

1.4 Tutte's Matching Theorem

Let $X \subseteq V(G)$. $G - X$ consists of even and odd (in terms of the number of vertices) connected components. We define $q_G(X)$ to be the number of odd components in $G - X$.



Definition 1.19. A graph G satisfies the *Tutte Condition* if $q_G(X) \leq |X|$ for all $X \subseteq V(G)$. $\emptyset \neq X \subseteq V(G)$ is called *barrier* if $q_G(X) = |X|$.

Proposition 1.20. *For any graph G and any $X \subseteq V(G)$:*

$$q_G(X) - |X| \equiv |V(G)| \mod 2$$

Definition 1.21. A graph G is *factor-critical* if $G - v$ has a perfect matching for all $v \in V(G)$. A matching is called *near-perfect* if it covers $|V(G)| - 1$ vertices.

Proposition 1.22. *If G is factor-critical, then it is connected.*

Theorem 1.23 (Tutte 1947)**.** *A graph G has a perfect matching $\Leftrightarrow$ Tutte Condition holds (i.e. $q_G(X) \leq |X|\ \forall X \subseteq V(G)$)*

Proof.

"$\Rightarrow$": Clear

"$\Leftarrow$": We proceed by induction on $|V(G)|$. The case $|V(G)| = 2$ is clear.

Generally, if the Tutte Condition holds, then $|V(G)|$ must be even (pick $X = \emptyset$). Proposition 1.20 $\Rightarrow$ $q_G(X) - |X|$ is even. Every $x \in V(G)$ induces a barrier $\{x\}$. Let X be a maximum barrier. Then $G - X$ doesn't have any even components (since otherwise a single vertex of such a component could be added to X).

Claim: Each odd component is factor-critical.
Let C be an odd component in $G - X$, $v \in V(C)$. Assume that $C - v$ does not have a perfect matching. Induction Hypothesis $\Rightarrow$ $C - v$ violates Tutte Condition.
$\Rightarrow \exists Y \subseteq V(C - v) : q_{C-v}(Y) > |Y|$
$\overset{1.20}{\Rightarrow} q_{C-v}(Y) \geq |Y| + 2$
Observe $X \cap \{v\} = Y \cap \{v\} = X \cap Y = \emptyset$:

$$\begin{aligned}
q_G(X \cup Y \cup \{v\}) &= q_G(X) - 1 + q_C(Y \cup \{v\}) \\
&= |X| - 1 + q_{C-v}(Y) \\
&\geq |X| - 1 + |Y| + 2 \\
&= |X \cup Y| + 1 \\
&= |X \cup Y \cup \{v\}|
\end{aligned}$$

$\Rightarrow X \cup Y \cup \{v\}$ is a barrier
$\Rightarrow$ Claim

Let G' arise from G by contracting each odd component into a single vertex. We have $V(G') = X \dot{\cup} Z$ and G' is bipartite. We have to show that G' has a perfect matching. If not, then $\exists A \subseteq Z : |\Gamma_{G'}(A)| < |A|$ $\Rightarrow q_G(\Gamma_{G'}(A)) \geq |A| > |\Gamma_{G'}(A)|$ which contradicts the Tutte Condition.

$\square$

Theorem 1.24 (Berge 1958)**.**

$$|V(G)| = 2\nu(G) + \max_{X \subseteq V(G)} (q_G(X) - |X|)$$

Proof. For $X \subseteq V(G)$, any matching has at least $q_G(X) - |X|$ uncovered vertices, so "$\geq$" holds.

For the other inequality, add $k := \max_{X \subseteq V(G)}(q_G(X) - |X|)$ new vertices and connect them to all existing vertices, yielding a new graph H.

We claim that H has a perfect matching. This then implies:

$$2\nu(G) + k \geq 2\nu(H) - k = |V(H)| - k = |V(G)|$$

Assume that H does not have a perfect matching. Then by Tutte's Theorem, there exists $Y \subseteq V(H)$ with $q_H(Y) > |Y|$. By 1.20, $k \equiv |V(G)| \mod 2$, therefore $|V(H)|$ is even, so $Y \neq \emptyset$. Y must contain all new vertices, otherwise $H - Y$ would be connected[2] and $q_H(Y) \leq 1 \leq |Y|$.

$$\Rightarrow q_G(Y \cap |V(G)|) = q_H(Y) > |Y| = |Y \cap V(G)| + k$$

which is a contradiction to the choice of k. □

1.5 Ear Decompositions of Factor-Critical Graphs

Definition 1.25. Let G be a graph. An *ear decomposition* of G is a sequence $r, P_1, \ldots, P_k$ with $G = (r, \emptyset) + P_1 + \ldots + P_k$ such that each P_i is either a path with exactly the endpoints located in $\{r\} \cup \bigcup_{j \in [i-1]} V(P_j)$ or a circuit where exactly one of the vertices belongs to $\{r\} \cup \bigcup_{j \in [i-1]} V(P_j)$.

$P_1, \ldots, P_k$ are called *ears.* If $|V(P_1)| \geq 3$ and $P_2, \ldots, P_k$ are paths we call it a *proper* ear decomposition.

Theorem 1.27 (Whitney 1932)**.** *Let G be an undirected graph. Then:*

$$G \text{ 2-connected} \Leftrightarrow G \text{ has a proper ear decomposition}$$

Definition 1.28. An ear decomposition is *odd* if every ear has odd length (in terms of the number of edges).

Theorem 1.29. *Let G be an undirected graph. Then*

$$G \text{ factor-critical} \Leftrightarrow G \text{ has an odd ear decomposition}$$

The first vertex r of the ear decomposition can be chosen arbitrarily.

Proof.

"$\Leftarrow$": Let G be a graph with an odd ear decomposition $r, P_1, \ldots, P_k$. P_1 is an odd circuit, so it is factor-critical. We use induction on the number of ears. Let P be the last ear and G' be G before adding P. By the induction hypothesis, G' is factor-critical. Given $v \in V(G)$, we have to show that $G - v$ has a perfect matching.

Case 1: $v \in V(G')$. Then $G' - v$ has a perfect matching. Adding every second edge of P (excluding the endpoints) to it, yields a perfect matching of $G - v$.

Case 2: $v \in V(G) \setminus V(G')$. Let x, y be the endpoints of P. Without loss of generality let $P_{[v,x]}$ be even. There exists a perfect matching in $G' - x$. Together with every second edge of $P_{[v,y]}$ and $P_{[v,x]}$ this is a perfect matching in $G - v$.

[2]Note that Y cannot contain all old vertices, since otherwise $q_H(Y) < |Y|$.

"$\Rightarrow$": Let $r \in V(G)$ be any vertex. Let M be a perfect matching in $G - r$. Suppose we have an odd ear decomposition for $G' \subseteq G$ with $r \in V(G')$ and $M \cap E(G')$ is a near-perfect matching in G' (i.e. all vertices in G' except for r are matched with other vertices in G').

If $G' \neq G$, there is an edge $\{x, y\} \in E(G) \setminus E(G')$ with $x \in V(G')$ (by Proposition 1.22). If $y \in V(G')$, then $\{x, y\}$ can be chosen as the next ear. Otherwise, construct an M-alternating odd ear, starting with $\{x, y\}$. Let N be a matching in $G - y$. $M \Delta N$ contains a y-r-path P. Let w be the first vertex in $P \cap V(G')$. w is M-exposed in $P_{[y,w]}$, y is N-exposed in $P_{[y,w]}$. Therefore $P_{[y,w]}$ is even and together with $\{x, y\}$ it forms an M-alternating odd ear.

Inductively, this argument yields an odd ear decomposition.

□

Definition 1.30. Let G be factor-critical and M a near-perfect matching. An M-alternating ear decomposition is an odd ear decomposition such that each ear is an M-alternating path or circuit C with:

$$|E(C) \cap M| = |E(C) \setminus M| - 1$$

Corollary 1.31. *For any factor-critical graph G and any near-perfect matching M in G, there exists in M-alternating ear decomposition of G.*

Definition 1.32. Let G be factor-critical, M a near-perfect matching and $r, P_1, \ldots, P_k$ an M-alternating ear decomposition of G. $\mu, \varphi : V(G) \to V(G)$ are *associated with the ear decomposition* if:

- $\{x, y\} \in M \Rightarrow \mu(x) = y$
- $\{x, y\} \in E(P_i) \setminus M$ and $x \notin \{r\} \cup \bigcup_{j \in [i]} V(P_j)$
  $\Rightarrow \varphi(x) = y$
- $\mu(r) = \varphi(r) = r$

Proposition 1.33. *Let G be a factor-critical graph and μ, φ functions associated with an M-alternating ear decomposition. Then this ear decomposition is unique up to the order of the ears. The Ear-Decomposition-Algorithm correctly determines an explicit list of the ears in linear time.*

Proof. Step 3 determines ears uniquely. The algorithm clearly runs in linear time. □

Lemma 1.34. *Let G be factor-critical and μ, φ associated with an M-alternating ear decomposition. Then the maximal path given by the initial sequence*

$$x, \mu(x), \varphi(\mu(x)), \mu(\varphi(\mu(x))), \ldots \tag{1}$$

defines an M-alternating x-r-path of even length.

Algorithm 1: Ear Decomposition Algorithm

Input: Factor-critical graph G, functions μ, φ associated with an M-alternating ear decomposition

Output: An M-alternating ear decomposition $r, P_1, \ldots, P_k$

$X := \{r\}$ where r is the vertex with $\mu(r) = r$
$k := 0$, $S :=$ empty stack
while $X \neq V(G)$ **do**
  if S *is non-empty* **then**
    Let $v \in V(G) \setminus X$ be an endpoint of the topmost element of the stack
  else
    Choose $v \in V(G) \setminus X$ arbitrarily
  $x := v,\ y := \mu(v),\ P := (\{x, y\}, \{\{x, y\}\})$
  while $\varphi(\varphi(x)) = x$ **do**
    $P := P + \{x, \varphi(x)\} + \{\varphi(x), \mu(\varphi(x))\}$
    $x := \mu(\varphi(x))$
  while $\varphi(\varphi(y)) = y$ **do**
    $P := P + \{y, \varphi(y)\} + \{\varphi(y), \mu(\varphi(y))\}$
    $y := \mu(\varphi(y))$
  $P := P + \{x, \varphi(x)\} + \{y, \varphi(y)\}$
  P is the ear containing y as an inner vertex. Put P on S.
  while *Both endpoints of the topmost element P of the stack S are in X* **do**
    Delete P from S
    $k := k + 1,\ P_k := P,\ X := X \cup V(P)$
forall $\{y, z\} \in E(G) \setminus (E(P_1) \cup \ldots \cup E(P_k))$ **do**
  $k := k + 1,\ P_k := (\{y, z\}, \{\{y, z\}\})$
return $r, P_1, \ldots, P_k$

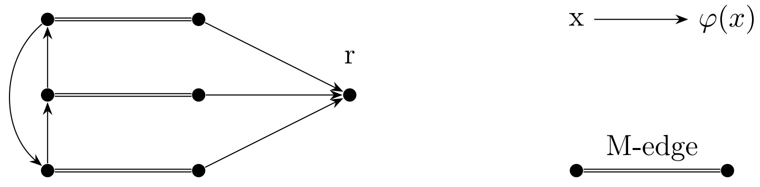


Figure 2: Counter example for the reverse implication of lemma 1.34

Proof. We proceed by induction on the number of ears. Let $x \in V(G) \setminus \{r\}$ and P_i be the ear containing x. A subsequence of (1) is a subpath Q of P_i from x to $y \in \{r\} \cup \bigcup_{j \in [i-1]} V(P_j)$. Q starts with a matching edge and ends with a non-matching edge, so it has even length. If $y = r$, we are done, otherwise the statement follows from the induction hypothesis. □

1.6 Edmond's Matching Algorithm

Definition 1.35. Let G be a graph, M a matching in G. A *blossom* in G with respect to M is a factor-critical subgraph of B of G such that $|M \cap E(B)| = \frac{|V(B)|-1}{2}$. The vertex $r \in V(B)$ that is exposed by M is called the *base* of B.

Definition 1.36. Let G be a graph, M a matching in G, B a blossom and Q a M-alternating v-r-path of even length from $v \in V(G)$ that is M-exposed to the base r of B. Additionally, let $E(Q) \cap E(B) = \emptyset$. $B + Q$ is called a M-flower.

Lemma 1.37. *Let G be a graph, M a matching in G. Suppose there is a M-flower $B+Q$. Let G', M' result from G and M by contracting $V(B)$ into a single vertex. Then:*

$$M \text{ maximum matching in } G \Leftrightarrow M \text{ maximum matching in } G'$$

Proof.

"$\Leftarrow$": Assume that M is not maximum in G. $N := M \Delta E(Q)$ is a matching with $|N| = |M|$.
$\Rightarrow \exists N$-augmenting path P in G. At least one endpoint x of P is in $V(B)$. If P and B are disjoint, let y be the other endpoint of P. Otherwise, let y be the first vertex on P in B. $P' := P_{[x,y]}$ is an N'-augmenting path in G', so $|N'| = |M'| < \mu(G')$.

"$\Rightarrow$": Assume that M' is not maximum in G', so there exists a matching N' in G' with $|N'| > |M'|$. Let N_0 arise from N' in G, then N_0 contains ≤ 1 vertex from $V(B)$. Since B is factor-critical, N_0 can be extended by $k := \frac{|V(G)|-1}{2}$ edges to a matching N in G. We have

$$|N| = |N_0| + k = \left|N'\right| + k > \left|M'\right| + k = |M|$$

so M is not maximum.

□

Lemma 1.39. *Let G be a graph, M a matching in G. $X \subseteq V(G)$ is the set of exposed vertices. We can find a shortest M-alternating X-X-walk of positive length in $O(|E(G)|)$ time.*

Proof. Define $D := (V(G), A)$ where:

$$A := \{(u, v) \mid \exists x \in V(G) : \{u, x\} \in E(G), \{x, v\} \in M\}$$

A shortest $X - \Gamma_G(X)$-path in D corresponds to a shortest X-X-walk in G. □

Theorem 1.40. *Let $P = v_0, \dots, v_t$ be a shortest M-alternating X-X-walk in G. Then either*

- *P is an M-augmenting path* or
- *$v_0, \dots, v_j$ is an M-flower for some $j \leq t$.*

Proof. If P is not a path, choose $i < j$ such that $v_i = v_j$ and j minimal. Then $v_0, \dots, v_{j-1}$ are distinct vertices. If $j - i$ is even, deleting $v_{i-1}, \dots, v_j$ from P yields a shorter walk, so $j - i$ is odd.

Case 1: j is even. Then i is odd and therefore $v_{i+1} = v_{j-1}$ must be the matching mate of $V_i = v_j$ which contradicts the minimality of j.

Case 2: j is odd. Then i is even, so $v_0, \dots, v_i$ is an M-alternating path of even length and $v_i, \dots, v_j$ is an M-alternating odd circuit, i.e. a blossom.

□

Algorithm 2: Edmond's Augmenting Path Search

Input: Graph G, matching M
Output: An M-augmenting path (if one exists)

$X :=$ set of exposed vertices
if *$\exists M$-alternating X-X-walk of positive length* **then**
    $P = v_0, \dots, v_t :=$ a shortest such walk
    if *P is a path* **then**
        return P
    else
        Choose j as in Theorem 1.40
        $v_0, \dots, v_j$ is an M-flower with blossom B
        Recurse on G/B
        Augment an M/B-augmenting path in G/B to an M-augmenting path P' in G
        return P'
else
    $\nexists M$-augmenting path

Theorem 1.41. *Given a graph G, a maximum cardinality matching can be found in time $O(n^2m)$ where $n := |V(G)|, m := |E(G)|$*

Proof. Start with $M = \emptyset$ and iteratively find M-augmenting path P, set $M := M\Delta E(P)$. If no such path exists, then M is maximum. P can be found in time $O(mn)$[3]. Since M is maximum after at most $\frac{n}{2}$ augmentation, we have total running time $O(n^2m)$. □

1.6.1 Growing forest - $O(n^3)$

Definition 1.42. Let G be a graph, M a matching in G. An *alternating forest* with respect to M in G is a forest F in G where:

- $V(F)$ contains all M-exposed vertices, each tree of F contains exactly one exposed vertex, its *root.*
- We call $v \in V(G)$ an outer (inner) vertex if it has even (odd) distance from the root of its component.
- $\forall v \in V(F)$ the unique path from v to the root of its component is M-alternating.
- $v \in V(G) \setminus V(F)$ is called *out-of-forest.*

Clearly, inner vertices always have degree 2 (we always assume that there are no matching edges that can immediately be added to F).

Proposition 1.43. *In any alternating forest, the number of outer vertices that are not the root equals the number of inner vertices.*

Proof. For all outer vertices, there exists exactly one inner vertex on its path to the root. □

Lemma 1.44. *Given a graph G, a matching M, an alternating forest F with respect to M in G. Then, either M is a maximum matching or $\exists$ outer vertex $x \in V(F)$, an edge $\{x, y\} \notin E(F)$ such that one of the following holds:*

- *Grow: $y \notin V(F)$ and therefore $\{y, z\} \in M$ with $z \notin V(F)$. In this case, y, z and $\{x, y\}, \{y, z\}$ can be added to F.*
- *Augment: y is an outer vertex in a different connected component in F. In this case, M can be augmented along $P(x) \cup \{x, y\} \cup P(y)$ where $P(z)$ denotes the unique path from $z \in V(F)$ to the root of its connected component.*

[3]Here, m is the time required for finding a walk and the recursion depth is bounded by n.

- *Shrink: y is an outer vertex in the same component as x. Let r be the first vertex on $P(x)$ that is also on $P(y)$. Then $|\delta_F(r)| \geq 3$, so y is an outer vertex and $\left|E(F_{[x,r]})\right|, \left|E(F_{[y,r]})\right|$ are even. Together with $\{x,y\}$ these paths form a blossom with ≥ 3 vertices.*

Proof. We show that if none of these cases apply, M is maximum. Let X be the set of inner vertices, $s := |X|$ and t be the number of outer vertices. All outer vertices are isolated in $G - X$, so $G - X$ and $q_G(X) - |X| = t - s$. By Berge's formula (1.24), $t - s$ vertices are exposed by any matching, so M is maximum. □

Definition 1.45. Let G be a graph, M a matching in G. A subgraph F of G is a *general blossom forest* with respect to M if there exists a partition $V(F) = V_1 \dot\cup \ldots \dot\cup V_k$ such that $F_i = F[V_i]$ is a maximal factor-critical subgraph of F with $|M \cap E(F_i)| = \frac{|V_i|-1}{2}$ $(i \in [k])$ and after contracting each V_i, we obtain an M-alternating forest F'. F_i is called an outer (inner) blossom if V_i is an outer (inner) vertex in F'.

A *special blossom forest* is a general blossom forest where each inner blossom is a single vertex.

Store a special blossom forest with 3 functions $\mu, \varphi, \rho : V(G) \to V(G)$:

$$\mu(x) := \begin{cases} x & \text{if } x \text{ is exposed in } M \\ y & \text{if } \{x,y\} \in M \end{cases}$$

$$\varphi(x) := \begin{cases} x & \text{if } x \text{ is the base of an outer blossom or } x \text{ is out-of-forest} \\ y & \text{if } x \text{ is an inner vertex and } \{x,y\} \in E(F) \setminus M \\ y & \text{if } x \text{ is an outer vertex (i.e. in an outer blossom) and } \mu, \varphi \text{ are associated with an } M\text{-alternating ear decomposition of } x\text{'s blossom, } \{x,y\} \in E(F) \setminus M \end{cases}$$

$$\rho(x) := \begin{cases} x & \text{if } x \text{ is an inner vertex or out-of-forest} \\ y & \text{if } x \text{ is an outer vertex and } y \text{ is the base of the outer blossom containing } x \ (y = x \text{ is possible).} \end{cases}$$

Proposition 1.46. *Let F be a special blossom forest with respect to M and μ, φ, ρ as above. Then:*

1. *For all outer vertices x, $P(x) :=$ maximal path given by subsequence of $x, \mu(x), \varphi(\mu(x)), \mu(\varphi(\mu(x))), \ldots$ is an M-alternating path from x to q where q is the root of the component containing x.*

2. *A vertex x is*

    - *an outer vertex $\Leftrightarrow \mu(x) = x \vee \varphi(\mu(x)) \neq \mu(x)$*

- *an inner vertex* $\Leftrightarrow \varphi(\mu(x)) = \mu(x) \wedge \varphi(x) \neq x$
- *out-of-forest* $\Leftrightarrow \mu(x) \neq x \wedge \varphi(x) = x \wedge \varphi(\mu(x)) = \mu(x)$

Proof.

1. By definition of μ, φ and lemma 1.33 some initial subsequence of $P(x)$ ends at the base r of the blossom containing x. If $r = q$, we are done. Otherwise $\mu(r), \varphi(\mu(r))$ are next elements in a sequence leading to outer vertex $\varphi(\mu(r))$. This can be iterated.

2. Since the conditions are mutually exclusive, it suffices to show one implication for all the statements.

   - If x is outer, it is a root ($\mu(x) = x$) or $P(x)$ is a path of length at least 2, so $\varphi(\mu(x)) \neq \mu(x)$.
   - If x is inner, then $\mu(x)$ is the base of an outer blossom. Therefore $\varphi(\mu(x)) = \mu(x)$. $P(\mu(x))$ is a path of length at least 2, so $\varphi(x) \neq x$.
   - If x is out-of-forest, then x is covered by M so $\mu(x) \neq x$. By definition of φ, $\varphi(x) = x$. $\mu(x)$ is out-of-forest as well, so $\varphi(\mu(x)) = \mu(x)$.

□

Lemma 1.47. *Following invariants hold:*

a) $\{\{x, \mu(x)\} \mid x \in V(G), \mu(x) \neq x\}$ *is a matching*

b) $\{\{x, \mu(x)\} \mid \underbrace{x \in V(G), \varphi(\mu(x)) = \mu(x) \wedge \varphi(x) \neq x}_{\textit{inner vertices}}\} \cup \{\{x, \varphi(x)\} \mid x \in V(G), \varphi(x) \neq x\}$ *forms the edge set of a special blossom forest.*

c) μ, φ, ρ *satisfy the conditions in definition 1.45 (special blossom forest).*

Proof. a) holds as μ only changes in *Augment.* b) is correct after initialization and after the reset in the *Augment* step. It is preserved by *Grow* steps.

In a *Shrink* step, r (the first vertex that the paths from x, y to the root share) is a root or has $|\delta(r)| = 3$ (i.e. it is the base of a blossom), so it is an outer vertex. We define a blossom $B := \{v \in V(G) \mid \varphi(v) \in V(P(x)_{[x,r]}) \cup V(P(y)_{[y,r]})\}$. Consider $\{u, v\} \in F$ with $u \in B, v \notin B$. If $\{u, v\} \in M$, we have $u = r, v = \mu(r)$ (since $F[B]$ contains a near-perfect matching). u was an outer vertex before shrinking and $F[B]$ being factor-critical follows from c) and the characterization by ear-decompositions.

For c), it's clear that μ always represents a matching. $\varphi(x) = x$ if x is not an outer vertex. Therefore, $\mu + \varphi$ represent an M-alternating ear decomposition of B. During *Shrink*, $\varphi(v)$ is not changed if $\varphi(v) = r$. Therefore, the odd ear decomposition for $B' :=$ blossom containing r, is the correct starting point. The next ear is $P(x)_{[x,x']} \cup P(y)_{[y,y']} + \{x, y\}$, where x' (y') is the first vertex in B' on $P(x)_{[x,r]}$ ($P(y)_{[y,r]}$).

For each ear Q of a former blossom $B'' \subseteq B$, $Q \setminus (E(P(x)) \cup E(P(y)))$ form a new ear (since it is created by removing an even path). φ, μ represent this ear-decomposition. □

Theorem 1.48. *Edmond's cardinality matching algorithm correctly determines a maximum matching in $O(n^3)$ time, where $n := |V(G)|$.*

Proof. By lemma 1.47 and proposition 1.46, the algorithm maintains a special blossom forest. Let M, F be the final matching and forest. x an outer vertex implies that $\forall y \in \Gamma(x) : y$ is inner and $\varphi(y) = \varphi(x)$. Define:

$$B := \text{set of inner vertices}$$
$$B := \text{set of bases of (outer) blossoms}$$

Then every unmatched vertex is in B. Matched vertices in B have matching mates in X and $|B| = |X| + |V(G)| - 2\,|M|$. (Outer) blossoms are odd connected components in $G - X$, so by Berge's theorem (1.24), at least $|B| - |X|$ vertices remain uncovered by any matching, so M is maximum.

We now consider the running time: The status (outer, inner, out-of-forest) for a given vertex can be checked in constant time (proposition 1.46). Therefore, *Grow*, *Augment* and *Shrink* can be implemented in $O(n)$ time. There are at most n calls to *Grow* and *Shrink* per augment and at most $\frac{n}{2}$ *Augments*. This implies the running time $O(n^3)$. □

Remark 1.49. The time for *Shrink* can be reduced to $O(\log n)$ using a binary tree, leading to a running time of $O(nm \log n)$ in total. Tarjan (1974), Gabow & Tarjan (1983) proved a running time of $O(nm\alpha(m, n))$ (where α is the inverse Ackermann function) or $O(nm)$.

Remark 1.50. It's not necessary to reset everything after augmenting. It suffices to reset the 2 trees that were changed by the augmentation. Gabow & Tarjan (1983) showed that it's possible to augment all paths of the same length in $O(m)$ time. There are $2\sqrt{\nu(G)} + 2$ different path lengths, so in total this results in a running time of $O(\sqrt{n}m)$.

Remark 1.51 (Skew-symmetric flows). Goldberg & Karzanov (2003) (and Fremuth-Pagen & Jungnickel (2003)) used *Generalized Max-Flow* to achieve a running time of $O(\sqrt{n}m\frac{\log \frac{m}{n}}{\log n})$.

1.7 Gallai-Edmonds Decomposition

Proposition 1.52. *Let G be a graph, $X \subseteq V(G)$ with $|V(G)| - 2\nu(G) = q_G(X) - |X|$. Then any maximum matching of G*

Algorithm 3: Edmond's Cardinality Matching Algorithm

Input: A graph G

Output: A maximum matching M (defined by $\{x, \mu(x)\}$)

$\mu(v) := v,\ \varphi(v) := v,\ \rho(v) := v,\ \mathit{scanned}(v) :=$ false for all $v \in V(G)$

`// Outer Vertex Scan:`

while $\exists$ *outer vertex* x *with* $\mathit{scanned}(x) =$ false **do**

    Let y be a neighbor of x such that y is either out-of-forest or y is outer with $\rho(y) \neq \rho(x)$

    if *such a* y *does not exist* **then**

        $\mathit{scanned}(x)$ =true, **continue**

    `// Grow:`

    if y *is out-of-forest* **then**

        $\varphi(y) := x$, **continue**

    `// Augment:`

    else if $P(x)$ *and* $P(y)$ *are vertex-disjoint* **then**

        $\mu(\varphi(v)) = v,\ \mu(v) = \varphi(v)$ for all $v \in V(P(x) \cap P(y))$ with odd distance from x or y on $P(x)$ or $P(y)$, respectively

        $\mu(x) := y,\ \mu(y) := x$

        $\varphi(v) := v, \rho(v) := v, \mathit{scanned}(v) :=$ false for all $v \in V(G)$

    `// Shrink:`

    else

        Let r be the first vertex on $V(P(x)) \cap V(P(y))$ with $\rho(r) = r$

        forall $v \in V(P(x)_{[x,r]}) \cup V(P(y)_{y,r})$ *with odd distance from* x *or* y *on* $P(x)_{[x,r]}$ *or* $P(y)_{[y,r]}$, *respectively and* $\rho(\varphi(v)) \neq r$ **do**

            $\varphi(\varphi(v)) := v$

        if $\rho(x) \neq r$ **then**

            $\varphi(x) := y$

        if $\rho(y) \neq r$ **then**

            $\varphi(y) := x$

        forall $v \in V(G)$ *with* $\rho(v) \in V(P(x)_{[x,r]}) \cup V(P(y)_{[y,r]})$ **do**

            $\rho(v) := r$

return μ

- *contains a perfect matching in the even components of $G - X$.*
- *contains a near-perfect matching in odd components of $G - X$.*
- *matches all $x \in X$ to distinct odd components.*

Proof. Follows directly from Berge's theorem (1.24). □

Theorem 1.53. *Let G be a graph and:*

$$Y := \{v \in V(G) \mid \exists \text{ maximum matching that exposes } v\}$$

Define $X := \Gamma(Y)$ and $W := V(G) \setminus (X \cup Y)$. Then:

1. *X attains $\max_{X' \subseteq V(G)} q_G(X') - |X'|$.*
2. *$G[Y]$ is the union of factor-critical subgraphs and $G[W]$ is the union of even connected components.*
3. *Any maximum matching in G*
   - *contains a perfect matching in $G[W]$.*
   - *contains a near-perfect matching in each component of $G[Y]$.*
   - *matches all $x \in X$ to distinct connected components*

Y, X, W is called Gallai-Edmonds decomposition *of G.*

Proof. Consider the matching M and special blossom forest F at the end of the algorithm. Let X' (Y') be the set of inner (outer) vertices and W' the set of out-of-forest vertices.

Claim. *X', Y', W' satisfy 1., 2. and 3.*

(Proof of theorem 1.48).

Proposition 1.52 implies that any maximum matching covers all vertices in $V(G) \setminus Y'$, so $Y \subseteq Y'$. For the other inclusion, let $v \in Y'$. Then $M \Delta P(v)$ is a maximum matching exposing v, so $v \in Y$ and $Y' = Y$. By definition, $X = X'$ and $W = W'$. □

Corollary 1.54. *A graph G has a perfect matching $\Leftrightarrow \forall U \subseteq V(G)$, $G - U$ has at most $|U|$ factor-critical components.*

1.8 Minimum Weight Perfect Matching

We use the following Integer Programming formulation:

$$\begin{aligned} \min \sum_{e \in E(G)} & c_e x_e \\ \text{s.t.} \quad \sum_{e \in \delta(v)} & x_e = 1 \qquad v \in V(G) \\ & x_e \in \{0,1\} \end{aligned}$$

and the corresponding relaxation where we only require $x_e \geq 0$. The dual problem of this is:

$$\begin{aligned} \max \sum_{v \in V(G)} & z_v \\ \text{s.t.} \quad & z_v + z_w \leq c_e \qquad \{v, w\} \in E(G) \end{aligned}$$

Proposition 1.55 (Hungarian Method)**.** *Let G be a graph, $c \in \mathbb{R}^{E(G)}$ and $z \in \mathbb{R}^{V(G)}$ with $z_v + z_w \leq c_e$ for all $e = \{v, w\} \in E(G)$. Define:*

$$G_z := (V(G), \{e = \{v, w\} \mid z_v + z_w = c_e\})$$

Let M be a matching in G_z, F a maximal alternating forest in G_z with respect to M. Let X/Y be the set of inner/outer vertices. Then:

1. *If M is a perfect matching in G_z, then it is a minimum-weight perfect matching in G.*

2. *If $\Gamma_G(y) \subseteq X$ for all $y \in Y$, then M is a maximum matching.*

3. *If neither 1. nor 2. hold, define:*

$$\epsilon := \min\{\min_{e=\{v,w \in E(G[Y])\}} \frac{c_e - z_v - z_w}{2}, \min_{e \in \delta(Y) \cap \delta(V(F))} c_e - z_v - z_w\}$$

Set $z'_v := z_v - \epsilon$ for all $v \in X$, $z'_v := z_v + \epsilon$ for all $v \in Y$ and $z'_v := z_v$ for all $v \in V(G) \setminus (X \cup Y)$. Then z' is a feasible dual solution and $M \cup E(F) \subseteq E(G_{z'})$. Additionally, $\Gamma_{G_{z'}}(y) \setminus X \neq \emptyset$ for some $y \in Y$.

Proof. 1. Let M' be a minimum-weight perfect matching.

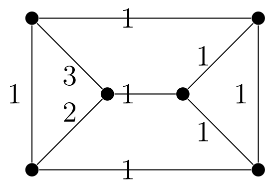
$$\begin{aligned} \sum_{e \in M'} c_e &= \sum_{v \in V(G)} z_v + \sum_{e=\{v,w\} \in M'} (c_e - z_v - z_w) \\ &\geq \sum_{v \in V(G)} z_v \\ &= \sum_{v \in V(G)} z_v + \sum_{e=\{v,w\} \in M} (c_e - z_v - z_w) \\ &= \sum_{e \in M} c_e \end{aligned}$$

2. Each outer vertex is an odd blossom (singleton) of $G - x$. By Berge (1.24), at least $|Y| - |X|$ vertices remain uncovered.

3. z' stays feasible by the choice of ϵ. Edges in $E(F), M$ remain tight. By 1. and 2., $\exists y \in Y : \Gamma_{G_{z'}}(y) \setminus X \neq \emptyset$.

□

Remark 1.56. For bipartite graphs, the adjacency matrix is totally unimodular, so the LP has integral vertices.



We define $\mathcal{A} := \{X \subseteq V(G) \text{ odd}\}$ and add the blossom inequalities

$$\sum_{e \in \delta(X)} x_e \geq 1 \qquad \forall X \in \mathcal{A}$$

to the LP relaxation. The new dual problem is then:

$$\begin{aligned} \max \sum_{A \in \mathcal{A}} z_A & \\ \text{s.t.} \quad \sum_{A \in \mathcal{A}, e \in \delta(A)} z_A &\leq c_e \\ z_A \geq 0 \qquad & (A \in \mathcal{A}, |A| \geq 3) \end{aligned}$$

Edmond's Algorithm starts with an empty matching $x = 0$ and dual feasible solution:

$$z_A := \begin{cases} \frac{1}{2} \min\{c(e) \mid e \in \delta(A)\} & |A| = 1 \\ 0 & \text{else} \end{cases}$$

We always ensure that z is dual feasible and that (x, z) satisfy complementary slackness:

$$x_e > 0 \Rightarrow \sum_{A \in \mathcal{A}, e \in \delta(A)} z_A = c_e$$

$$z_A > 0, |A| > 1 \Rightarrow \sum_{e \in \delta(A)} x_e = 1$$

Definition 1.57. $c_z(e) := c(e) - \sum_{A \in \mathcal{A}, e \in \delta(A)} z_A$ is the *reduced cost* of e.

Theorem 1.58. *There are at most* $\frac{7}{2}|V(G)|^2$ *of the repeat-until loop in algorithm 4.*

Proof. $\mathcal{B}$ is laminar at any time, i.e. for $X, Y \in \mathcal{B}$ we have $(X \subseteq Y) \vee (Y \subseteq X) \vee (X \cap Y = \emptyset)$. Therefore $|\mathcal{B}| \leq 2|V(G)|$.

Observation. *Any* U *added to* $\mathcal{B}$ *during* Shrink *will not be "unpacked" before the next* Augment.

Proof. After *Shrink*, there exists an even length M-augmenting R-U-path. It remains in G_z until the next *Augment* or until U is included in another blossom $U' \supseteq U$ which is not resolved before an *Augment* (inductively). ☐

Between 2 augments:

- \# *Unpacks* $\leq |\mathcal{B}|$ at beginning of the sequence
- \# *Shrinks* $\leq |\mathcal{B}|$ at the end of the sequence

Therefore, there are at most $4|V(G)|$ *Unpack* and *Shrink* operations between 2 augments. For each dual change without *Unpack*, we have: $z_B > 0 \quad \forall B \in \mathcal{B}$, so ϵ is not determined by z_B. Therefore $\exists e = \{X, Y\}$ with $X \notin \mathcal{X}, Y \in \mathcal{Y}$ where $c_z(e)$ becomes 0.

Case 1: $X \notin \mathcal{Y}$. Then $|V(G_z) \setminus (\mathcal{X} \cup \mathcal{Y})|$ decreases.

Case 2: $X \in \mathcal{Y}$. Then $\exists$ X-Y M-alternating walk in the next iteration.

In particular, such a dual change can occur at most $|V(G)|$ times between 2 augmentations.

In total, there are at most $\frac{1}{2}|V(G)|$ *Augment* steps. Therefore, there are $\frac{1}{2}|V(G)|^2(4 + |V(G)| + 2|V(G)|)$ ☐

Algorithm 4: Minimum-Weight Perfect Matching

Input: Graph G with edge weights $c : E(G) \to \mathbb{R}$

Output: A minimum-weight perfect matching M in (G, c)

Corollary 1.59. *A minimum-weight perfect matching can be computed in* $O(n^2 m)$ *time where* $n := |V(G)|$ *and* $m = |E(G)|$.

Proof. Theorem 1.58 times $O(m)$. ☐

Remark 1.60. To achieve $O(n^3)$ running time, one can modify the algorithm:

1. Use a General Blossom Forest to avoid recomputing the R-R-walks from scratch. We then have mappings $\mu_v, \varphi_v^i, \rho_v^i$ for $1 \leq i \leq k_v$ where k_v is the number of blossoms that contain v.

2. Store all vertices in a heap (ordered by their criticality for dual-feasibility) to speed up the computation of ϵ.

Gabow (1990) showed a running time of $O(n(m+n\log n))$. Gabow & Tarjan (1991) showed a running time of $O(m\log(nW)\sqrt{n\alpha(m,n)\log n})$ where $W := \max_{e\in E(G)} |c(e)|$.

Theorem 1.61. *Let G be a graph. The set of vectors $x \in \mathbb{R}^{E(G)}$ satisfying*

$$\begin{aligned} x_e &\geq 0 && e \in E(G) \\ x(\delta(v)) &= 1 && v \in V(G) \\ x(\delta(A)) &\geq 1 && A \subseteq V(G) \text{ with } |A| \text{ odd} \end{aligned}$$

is the convex hull of all perfect matchings in G. It is called the perfect matching polytope*.*

Proof. For any objective function $c : E(G) \to \mathbb{R}$, the minimum-weight perfect matching algorithm produces an integral primal and a dual solution that satisfy complementary slackness. In particular, all vertices of the polytope are integral. □

Theorem 1.62. *Let G be a graph. The set of vectors $x \in \mathbb{R}^{E(G)}$ satisfying*

$$\begin{aligned} x_e &\geq 0 && e \in E(G) \\ x(\delta(v)) &\leq 1 && v \in V(G) \\ x(E(G[A])) &\leq \frac{|A|-1}{2} && A \subseteq V(G) \text{ with } |A| \text{ odd} \end{aligned}$$

is the convex hull of all matchings in G. It is called the matching polytope*.*

Proof. Any matching solution x satisfies these conditions. Let x be any solution that satisfies the conditions. We have to show that x is a convex combination of matching solutions. Define H by:

$$\begin{aligned} V(H) :=& \{(v,i) \mid v \in V(G), i \in \{1,2\}\} \\ E(H) :=& \{\{(v,i),(w,i)\} \mid \{v,w\} \in E(G), i \in \{1,2\}\} \\ & \cup \{\{(v,1),(v,2)\} \mid v \in V(G)\} \end{aligned}$$

We set $y_{\{(v,i),(w,i)\}} := x_{\{v,w\}}$ for all $\{v,w\} \in E(G), i \in \{1,2\}$ and $y_{\{(v,1),(v,2)\}} := 1 - x(\delta(v))$ for all $v \in V(G)$. Then $y \geq 0$ and $y(\delta_H(x)) = 1$ for all $x \in V(H)$.

Claim. *y satisfies the inequalities of the perfect matching polytope (in particular the blossom inequalities).*

If this is true, by 1.62 y is a convex combination of perfect matchings. $H[\{(v,1) \mid v \in V(G)\}]$ is isomorphic to G, so x is a convex combination of matchings in G.

We now prove the claim: Let $X \subseteq V(G)$ with $|X|$ odd. We have to show that $y(\delta_H(X)) \geq 1$. Define:

$$\begin{aligned} A &:= \{v \in V(G) \mid (v,1) \in X, (v,2) \notin X\} \\ B &:= \{v \in V(G) \mid (v,1) \in X, (v,2) \in X\} \\ C &:= \{v \in V(G) \mid (v,1) \notin X, (v,2) \in X\} \end{aligned}$$

Define $A_i := A \cap (V(G) \times \{i\})$ and $B_i := B \cap (V(G) \times \{i\})$. $|B_1 \cup B_2|$ is even, so (since $|X|$ is odd) $|A|$ or $|C|$ is odd. Without loss of generality, let $|A|$ be odd.

$$\begin{aligned} \sum_{e \in \delta_H(X)} y_e \geq & \sum_{v \in A_1} \underbrace{\sum_{e \in \delta_H(v)} y_e}_{=1} - 2 \cdot \sum_{e \in E(H[A_1])} y_e - \sum_{e \in \delta(A_1) \cap \delta(B_1)} y_e \\ & + \sum_{e \in \delta(A_2) \cap \delta(B_2)} \\ & = |A_1| - 2 \cdot \sum_{e \in E(G[A])} x_e \\ & \geq |A_1| - (|A| - 1) \\ & = 1 \end{aligned}$$

□

Theorem 1.63. *The matching polyhedron is TDI (Totally Dual Integral), i.e. for all $c \in \mathbb{Z}^{E(G)}$ for which the dual program of ($\max c^t x s.t. \dots$) has a finite optimum solution, it has an integral optimum solution.*

Proof. The dual is

$$\begin{aligned} \min & \sum_{v \in V(G)} y_v + \sum_{e \in \mathcal{A},\ |A|>1} \frac{|A|-1}{2} z_A \\ s.t. & \sum_{v \in e} y_v + \sum_{A \in \mathcal{A},\ |A|>1, e \in E(G[A])} z_A \geq c(e) \qquad e \in E(G) \\ & y, z \geq 0 \end{aligned}$$

Let (G, c) be a counterexample such that $|V(G)| + |E(G)| + \sum_{e \in E(G)} |c(e)|$ is minimum. Then:

- $c(e) \geq 1$ for all $e \in E(G)$, since otherwise e could be deleted.
- G has no isolated vertices.

Claim. *In an optimum solution (y, z), $y = 0$.*

Proof. If $y_v > 0$, then $x(\delta(v)) = 1$ for all optimum solutions x. Decreasing $c(e)$ by 1 for all $e \in \delta(v)$ yields a smaller feasible instance (G, c') where the weight of x is decreased by 1 and x remains optimum. By assumption, (G, c') is not a counterexample, so there exists an integral optimum solution (y', z'). Increasing y'_v by one yields some optimum in (G, c) which has optimum integral solution $(y' + \mathbb{1}_v, z')$. □

Let $(y = 0, z)$ be a dual optimum solution such that

$$\sum_{A \in \mathcal{A}, |A|>1} |A|^2 \, z_A$$

is maximum.

Claim. $\mathcal{F} := \{A : z_A > 0\}$ *is laminar.*

If not, there exist $X, Y \in \mathcal{F}: \; X \cap Y, X \setminus Y, Y \setminus X \neq \emptyset$. We proceed by "uncrossing". Let $\epsilon := \{z_X, z_Y\} > 0$.

Case 1: $|X \cap Y|$ is odd. Then $|X \cup Y|$ is odd. Define:

$$\begin{aligned}
z'_X &:= z_X - \epsilon \\
z'_y &:= z_y - \epsilon \\
z'_{X\cap Y} &:= z_{X\cap Y} + \epsilon && \text{(unless } |X \cap Y| = 1\text{)} \\
z'_{X\cup Y} &:= z_{X\cup Y} + \epsilon \\
z'_A &:= z_A && \text{elsewhere}
\end{aligned}$$

Then (y, z') is a dual optimum solution.

Case 2: $|X \cap Y|$ is even. Then $|X \setminus Y|$ and $|Y \setminus X|$ are odd. Define:

$$\begin{aligned}
z'_X &:= z_X - \epsilon \\
z'_Y &:= z_Y - \epsilon \\
z'_{X\setminus Y} &:= z_{X\setminus Y} + \epsilon && \text{unless } |X \setminus Y| = 1 \\
z'_{Y\setminus X} &:= z_{Y\setminus X} + \epsilon && \text{unless } |Y \setminus X| = 1 \\
z'_A &:= z_A && \text{elsewhere} y'_v := \epsilon \qquad \forall v \in X \cap Y \\
y'_v &:= 0 && \forall v \notin X \cap Y
\end{aligned}$$

Then (y', z') is feasible. The objective value is:

$$\begin{aligned}
&\sum_{v \in V(G)} y'_v + \sum_{A \in \mathcal{A},\ |A|>1} z'_A \frac{|A|-1}{2} \\
=&\epsilon \cdot |X \cap Y| + \sum_{A \in \mathcal{A},\ |A|>1} \frac{|A|-1}{2} \\
+&\epsilon \left(\frac{|X \setminus Y|-1}{2} + \frac{|Y \setminus X|-1}{2} - \frac{|X|-1}{2} - \frac{|Y|-1}{2} \right) \\
=&\text{objective}(y, z)
\end{aligned}$$

Therefore (y', z') is an optimum solution with $y' \neq 0$, which is a contradiction to the previous claim.

We can conclude that $\mathcal{F}$ is laminar.

Let $A \in \mathcal{F}$ with $z_A \notin \mathbb{Z}$ and $|A|$ is maximal. Define $\epsilon := z_A - \lfloor z_A \rfloor > 0$. Let $A_1, \ldots, A_k$ be the inclusion-wise maximal proper subsets of A in $\mathcal{F}$. Since $\mathcal{F}$ is laminar, $A_i \cap A_j = \emptyset$ for $i \neq j$. Define:

$$\begin{aligned}
z'_A &:= z_A - \epsilon \\
z'_{A_i} &:= z_A + \epsilon && 1 \leq i \leq k \\
z'_D &:= z_D && \text{elsewhere}
\end{aligned}$$

Then (y, z') is dual feasible with objective value:

$$\sum_{B \in \mathcal{A},\ |B|>1} \frac{|B|-1}{2} z'_B < \sum_{B \in \mathcal{A},\ |B|>1} \frac{|B|-1}{2} z_B$$

This contradicts the optimality of (y, z), so there exists no counter example. $\square$

Theorem 1.64. *Let G be a graph.*

$$P := \{x \in \mathbb{R}_{\geq 0}^{E(G)} \mid x(\delta(v)) \leq 1 \quad \forall v \in V(G)\}$$

is the functional matching polytope.

$$Q := \{x \in \mathbb{R}_{\geq 0}^{E(G)} \mid x(\delta(v)) = 1 \quad \forall v \in V(G)\}$$

If G is bipartite, then P and Q are integral.

Proof. The adjacency matrices of bipartite graphs are totally unimodular. $\square$

Theorem 1.65. *Let G be a graph. The vertices of the fractional perfect matching polytope satisfy*

$$x_e = \begin{cases} \frac{1}{2} & \text{if } e \in E(C_1) \cup \ldots \cup E(C_k) \\ 1 & \text{if } e \in M \\ 0 & \text{else} \end{cases}$$

where $C_1, \ldots, C_k$ are vertex-disjoint odd circuits and M is a perfect matching in $G - (V(C_1) \cup \ldots \cup V(C_k))$.

Proof. Exercise 6.3 □

2 T-Joins and b-Matchings

Definition 2.1. Let G be a graph, $T \subseteq V(G)$. A subset $J \subseteq E(G)$ is called *T-join* if T is the set of odd-degree vertices in $(V(G), J)$.

Proposition 2.2. *Let G be a graph, $T, T' \subseteq V(G)$, J a T-join ad J' a T'-join. Then $J \Delta J'$ is a $T \Delta T'$-join.*

Proof. For $v \in V(G)$:

$$\begin{aligned} |\delta_{J \cap J'}(v)| &\equiv |\delta_J(v)| + |\delta_{J'}(v)| \\ &\equiv |\{v\} \cap T| + \left|\{v\} \cap T'\right| \\ &\equiv \left|\{v\} \cap (T \Delta T')\right| \quad \bmod 2 \end{aligned}$$

□

Proposition 2.3. *Let G be a graph, $T \subseteq V(G)$.*

$$\exists\ T\text{-join in } G \Leftrightarrow |V(C) \cap T| \text{ for each connected component } C$$

Proof.

"⇒": Let J be a T-join. For each connected component C:

$$\sum_{v \in V(C)} |J \cap \delta(v)| = 2\,|J \cap E(C)|$$

Therefore $|J \cap \delta(v)|$ is odd for an even number of vertices and $|V(C) \cap T|$ is even.

"⇐": Partition T into pairs $\{v_1, w_1\}, \ldots, \{v_k, w_k\}$ such that v_i and w_i are in the same component for all i. Let P_i be a v_i-w_i-path in G. Define $J := E(P_1) \Delta E(P_2) \Delta \ldots \Delta E(P_k)$. By proposition 2.2, this is a T-join.

□

Theorem 2.4. *Let G be a graph, $c : E(G) \to \mathbb{R}$ and $T \subseteq V(G)$. In strongly polynomial time (e.g. $O(n^2 m)$) we can determine if a T-join exists and if so, compute a minimum-weight T-join.*

Proof. In $O(m)$ ($m := |E(G)|$), we can check if a T-join exists. If so:

1. Eliminate negative weights.

$$\begin{aligned} N &:= \{e \in E(G) \mid c(e) < 0\} \\ U &:= \{v \in V(G) \mid |\delta_N(v)| \text{ odd}\} \\ T' &:= T \Delta U \\ c'(e) &:= |c(e)| && e \in E(G) \end{aligned}$$

   Claim. *If J' is a minimum T'-join with respect to c', then $J' \Delta N$ is a minimum T-join with respect to c.*

   Let $\tilde{J}$ be a T-join. Then $\tilde{J} \Delta N$ is a T'-join, so $c'(\tilde{J}) \leq c'(\tilde{J} \Delta N)$ and

$$c(J) = c'(J') + c(N) \leq c'(\tilde{J} \Delta N) + c(N) = c(\tilde{J})$$

   which proves the claim.

2. We can now assume that $c \geq 0$. A minimum-weight T-join does not have cycles of positive weight. We can eliminate cycles of weight 0 without changing the cost. We can then restrict ourselves to searching for collections of T-T-paths.

   Let K_T be the metric closure of T with respect to G. It can be computed in $O(n \cdot (m + n \log n))$ by using Dijkstra for all vertices. Find a minimum-weight perfect matching M in K_T. Each $e = \{s, t\} \in M$ induces a path $P_{s,t}$. Then the symmetric difference $\Delta_{\{s,t\} \in M} E(P_{s,t})$ is a minimum-weight T-join in G.

□

Corollary 2.6. *A maximum-weight T-join can be computed as fast as a minimum-weight T-join.*

Proof. Set $c' := -c$. □

Corollary 2.7. *Let G be a graph, $c : E(G) \to \mathbb{R}$. We can find a cycle of negative length in G in $O(n^2 m)$ time.*

Proof. Apply theorem 2.4 to $T = \emptyset$. If $c(J) < 0$, $(V(G), J)$ contains a cycle C. If $c(C) = 0$, we can eliminate it and recurse, otherwise return C. □

2.2 T-Join Applications

2.2.1 TSP Approximation

Let (K_n, c) with c metric be an instance of the TSP. Consider the *Double tree algorithm*:

1. Compute a minimum spanning tree T.
2. $T' \coloneqq T + T$ (doubling all edges). Then T' is Eulerian.
3. Walk along T' and add vertices to the TSP tour in the order of their first appearance, yielding a tour T^*. Since c is metric, we have $c(^*) \leq c(T') \leq 2c(T)$. Since the cost of T is a lower bound for the cost of a tour, we have $c(T^*) \leq 2\mathrm{OPT}$ (where OPT is the cost of a shortest TSP tour).

Algorithm 5: Christofides Algorithm (1976)

Input: Complete metric graph (K_n, c)
Output: A TSP-tour T
Find MST T_{MST} in (K_n, c)
$W \coloneqq \{v \in V(K_n) \mid |\delta_{T_{\mathrm{MST}}}(v)| \text{ odd}\}$
$J \coloneqq$ minimum-weight W-Join in (K_n, c)
Add cities to T in the order of first appearance in a Eulerian walk of $T_{\mathrm{MST}} + J$.
return T

Theorem 2.8. *Algorithm 5 is a $\frac{3}{2}$-approximation algorithm for the metric TSP, i.e. for the computed tour T we have:*

$$c(T) \leq \frac{3}{2}\mathrm{OPT}$$

Proof. We have $c(T_{\mathrm{MST}}) \leq \mathrm{OPT}$ and $\mathrm{OPT}(W) \leq \mathrm{OPT}(V(K_n))$ (since c is metric). Any tour through the vertices in W can be decomposed into 2 matchings. Therefore, $c(J) \leq \frac{1}{2}\mathrm{OPT}(W) \leq \frac{1}{2}\mathrm{OPT}$. It follows that $c(T) \leq (1 + \frac{1}{2})\mathrm{OPT}$. □

2.2.2 Shortest Paths in Undirected Graphs

The naive reduction to digraphs requires non-negative weights.

Corollary 2.9. *Given an undirected graph G, $c : E(G) \to \mathbb{R}$ such that each ciruit has length at least 0. Then for $s, t \in V(G)$, a shortest s-t-path can be found in $O(n^2 m)$ time, where $n \coloneqq |V(G)|$, $m \coloneqq |E(G)|$.*

Proof. Choose $T \coloneqq \{s, t\}$. Apply theorem 2.4 to get a minimum-weight T-join J. J can be partitioned into circuits of length 0 and an s-t-path of length $c(J)$. □

2.2.3 Chinese Postman Problem

Definition 2.10. A walk $C = \{v_0, e_1, v_1, \ldots, e_t, v_t\}$ is called a Chinese postman tour if $v_0 = v_t$ and each edge in $E(G)$ is visited at least once. The Chinese Postman Problem is the problem of finding a shortest Chinese postman tour in G with respect ot $c : E(G) \to \mathbb{R}_{\geq 0}$.

Corollary 2.11. *The Chinese postman problem can be solved in $O(n^2 m)$ time, where $n := |V(G)|\,,\ m := |E(G)|$.*

Proof. Set $T := \{v \in V(G)\ \ |\delta(v)| \text{ odd}\}$ and let J be a minimum-weight T-join. Compute a Eulerian tour C in $G + J$. Let C' be a shortest Chinese postman tour. Let $J' :=$ set of edges occuring in C' an even number of times (at least twice). Then J' is a T-join, so $c(J') \geq c(J)$ and:

$$c(C') \geq c(E(G)) + c(J') \geq c(E(G)) + c(J) = c(C)$$

□

2.3 T-Joins and T-Cuts

Definition 2.12. Let G be a graph and $T \subseteq V(G)$. A T-cut is a cut $C = \delta(X)$ with $X \subseteq V(G)$ and $|X \cap T|$ odd.

Proposition 2.13. *Let G be a graph, $T \subseteq V(G)$, $|T|$ even. Then:*

1. *For any T-join J and any T-cut C: $J \cap C \neq \emptyset$.*
2. *The inclusion-wise minimal T-cuts (T-joins) are exactly the inclusion-wise minimal edge sets intersecting all T-joins (all T-cuts).*

Proof. For 1., let $C = \delta(X)$ with $|X \cap T|$ odd be a T-cut. Then the edges in $J \cap C$ either belong to a path passing through X or have an endpoint in T. Therefore $|J \cap C|$ is odd, in particular the set is non-empty.

For 2., we prove in an exercise that each edge set intersecting all T-joins (T-cuts) contains a T-cut (T-join). Therefore minimal such sets are T-cuts (T-joins). Remark: The minimum cardinality of a T-join is at least as large as the maximum number of edge-disjoint T-cuts[4]. □

Theorem 2.14 (Seymour (1981))**.** *Let G be bipartite, $T \subseteq V(G)$ such that there exists a T-join. Then:*

$$\textit{min. cardinality of a } T\textit{-join} = \textit{max. number of edge-disjoint } T\textit{-cuts}$$

The maximum is attained by a crossfree family $\mathcal{C}$ of cuts, i.e.

$$\forall X, Y \in \mathcal{C} : X \subseteq Y \vee Y \subseteq X \vee X \cap Y = \emptyset \vee X \cup Y = V(G)$$

[4] In general, the two numbers are not equal: Consider K_4 and $T = V(K_4)$. A minimum T-join consists of 2 edges but there are no 2 edge-disjoint T-cuts.

Proof. If $T = \emptyset$, the statement is clear. Let $T \neq \emptyset$. We proceed by induction on $|V(G)| + |T|$. Let J be a minimum-cardinality T-join. Set:

$$c(e) := \begin{cases} -1 & e \in J \\ 1 & e \in E(G) \setminus J \end{cases}$$

Claim. *Every circuit C has $c(C) \geq 0$.*

$$\begin{aligned} c(C) &= c(C \setminus J) + c(C \cap J) + |J \setminus C| - |J \setminus C| \\ &= \Bigg|\underbrace{C \Delta J}_{T\text{-join}}\Bigg| - |J| \geq 0 \end{aligned}$$

Let P be a minimum length walk in (G, c) traversing no edge more than once such that $|E(P)|$ is minimum. Then P is a path. Let t be the last vertex in P and f the edge entering t. Then $f \in J$, otherwise $c(f) = 1$ and deleting f would yield a shorter path. Furthermore, $|\delta_J(t)| = 1$, otherwise we could add the other edge from $J \cap \delta(t)$ to shorten $c(P)$.

Claim. *Each circuit C that contains t but not f has $c(C) > 0$.*

Case 1: t is the only vertex in $V(C) \cap V(P)$. Let $e \ni t$ be an edge on C incident to t. Then $c(e) = 1$ (since $\delta_J(t) = \{f\}$) and $P' := P + C - e$ yields a shorter walk if $c(C) \leq 0$.

Case 2: $V(C) \cap V(P)$ contains another vertex x. Let u be the last vertex on P before t that is also on C. Define $P' := P_{[u,t]}$. C can be split into 2 u-t-paths C', C''. By minimality of P, $c(P') < 0$. $P' + C', P' + C''$ are circuits (by choice of u). By the first claim, $c(C'), c(C'') > 0$, so also $c(C) > 0$.

Shrink: $\{t\} \cup \Gamma(t)$ to a new vertex v_0. This yields a bipartite graph G'. If $|T \cap (\{t\} \cup \Gamma(t))|$ is odd, set $T' := T \setminus (\{t\} \cup \Gamma(t)) \cup \{v_0\}$. Otherwise, $T' := T \setminus (\{t\} \cup \Gamma(t))$. Define $J := J \setminus \{f\}$.

Claim. *J' is a minimum cardinality T'-join in G'.*

If not, there exists a T'-join J'' with $|J''| < |J'|$. $J'' \Delta J'$ is an $\emptyset$-Join. Therefore, there exists a circuit C' where $|C' \setminus J'| < |C' \setminus J''| = |C' \cap J'|$ (since G is bipartite). If C' results from a circuit C in G not containing T, then $|C \setminus J| < |C \cap J|$. This is a contradiction to the minimality of J.

Therefore C' results from a circuit containing T.

Case 1: C traverses f. Then

$$\begin{aligned}\left|C' \setminus J'\right| - \left|C' \cap J'\right| &= |C \setminus J| - |C \cap J| \\ &> 0\end{aligned}$$

which is a contradiction.

Case 2: By the second claim, $c(C) > 0$, so since G is bipartite $c(C) \geq 2$ and $|C \setminus J| \geq |C \cap J| + 2$. Therefore

$$\begin{aligned}\left|C' \setminus J'\right| &= |C \setminus J| - 2 \\ &\geq |C \cap J| \\ &= \left|C' \cap J'\right|\end{aligned}$$

which is a contradiction to the assumption.

By the induction hypothesis on G', G' has cross-free T'-cuts $D_1, \ldots, D_{|J'|}$. Together with $\delta(t)$, we get $|J'| + 1 = |J|$ T-cuts. Since $\Gamma(t)$ was contracted in G', they are cross-free. $\square$

Corollary 2.15. *Let G be a graph, $c : E(G) \to \mathbb{Z}_{\geq 0}$, $T \subseteq V(G)$ such that a T-join exists. The minimum cost of a T-join equals half the maximum number of T-cuts covering each edge e at most $2 \cdot c(e)$ times. This maximum is attained by a cross-free family of T-cuts.*

Proof. Let $E_0 := \{e \in E(G) \mid c(e) = 0\}$. Contract the connected components in $(V(G), E_0)$ and replace each $e \in E(G)$ by a path of length $2 \cdot c(e) > 0$. The resulting graph G' is bipartite. Let

$T' := \{v \in V(G') \mid v$ corresponds to a connected component X in G with $|X \cap T|$ odd$\}$

Let k be the minimum cost of a T-join in G.

Claim. *The minimum cardinality of a T'-join in G' is $2k$.*

"$\leq$": Every T-join J in J corresponds to a T'-join J' in G' with $|J'| \leq 2c(J)$.

"$\geq$": Let J' be a T'-join in G'. J' corresponds to an edge set $J \subseteq E(G)$. Let $\overline{T} := T \Delta \{v \in V(G) \mid |\delta(v) \cap J| \text{ odd}\}$. For each connected component X in $(V(G), E_0)$:

$$|\delta(X) \cap J| \equiv |X \cap T| \mod 2$$

Therefore $\left|X \cap \overline{T}\right|$ is even, so by proposition 2.3, there exists a $\overline{T}$-join $\overline{J}$ in $(V(G), E_0)$. Then $J \cup \overline{J}$ is a T-join of weight $c(J) = \frac{|J'|}{2}$.

By theorem 2.14, there exist $2k$ pairwise disjoint T'-cuts in G'. In G this yields $2k$ T-cuts such that every edge e is covered by at most $2 \cdot c(e)$ cuts and they can be created cross-free. $\square$

2.3.1 T-join Polytope

We define the T-join polytope:

$$P_{T\text{-join}} := \operatorname{conv}\{x \in \mathbb{R}^{E(G)} \mid x \text{ incidence vector of a } T\text{-join}\}$$
$$P^{\uparrow}_{T\text{-join}} := P_{T\text{-join}} + \mathbb{R}^{E(G)}_{\geq 0}$$

Corollary 2.16. *$P^{\uparrow}_{T\text{-join}}$ is determined by*

$$\begin{aligned} x_e &\geq 0 && e \in E(G) \\ x(\delta(X)) &\geq 1 && \forall T\text{-cuts } \delta(X) \end{aligned}$$

Proof. "$\subseteq$" is clear. Assume that the other inclusion does not hold. Then there exists $w : E(G) \to \mathbb{Q}$ such that the minimum weight of a T-join $\alpha > \min w^t x$ where x satisfies the stated inequalities. Without loss of generality, $w \in \mathbb{Z}^{E(G)}_{\geq 0}$, both cones are identical ($\mathbb{R}^{E(G)}_{\geq 0}$). By corollary 2.15, there exist T-cuts $C_1, \dots, C_{2\alpha}$ such that each edge e is covered at most $2w(e)$ times.

$$y_C := \frac{1}{2}\text{number of times } C \text{ occurs in } C_1, \dots, C_{2\alpha}$$

Then y is a feasible solution to the dual:

$$\begin{aligned} &\max_{C\ T\text{-cut}} y_C \\ \text{s.t.} &\sum_{C\ T\text{-cut},\ e \in C} y_e \leq w(e) && e \in E(G) \\ & y \geq 0 \end{aligned}$$

$\sum_C y_C = \alpha$ is a lower bound for the minimization problem which is a contradiction to the assumed inequality. $\square$

2.4 Excursus: Gomory-Hu Trees

Let G be a graph, $u : E(G) \to \mathbb{R}_{\geq 0}$. Find $\emptyset \subsetneq X \subsetneq V(G)$ minimizing $u(\delta(X))$. One approach: $\binom{|V(G)|}{2}$ s-t-cut computations (this can clearly be reduced to $|V(G)| - 1$ by fixing s).

Definition 2.17. For $s, t \in V(G)$, denote by λ_{st} the minimum capacity of an s-t-cut (or *local edge connectivity* of s, t).

Lemma 2.18. *For all $u, v, w \in V(G)$:*

$$\lambda_{uw} \geq \min\{\lambda_{uv}, \lambda_{vw}\}$$

Proof. Let $\delta(A)$ be a u-w-cut. If $v \in A$, then $\delta(A)$ is a v-w-cut, so $u(\delta(A)) \geq \lambda_{vw}$. Otherwise, $\delta(A)$ is a u-v-cut, so $u(\delta(A)) \geq \lambda_{uv}$. $\square$

Definition 2.19. Let G be a graph, $u : E(G) \to \mathbb{R}_{\geq 0}$. A tree T is a Gomory-Hu tree for (G, u) if $V(T) = V(G)$ and

$$\lambda_{st} = \min_{e \in E(T_{[s,t]})} u(\delta_G(C_e)) \qquad \forall s, t \in V(G)$$

where C_e and $V(G) \setminus C_e$ are the connected components of $T - e$[5].

Lemma 2.20. *Given (G, u) and a tree T with $V(T) = V(G)$:*

$$T \text{ Gomory-Hu tree} \Leftrightarrow \forall e = \{s, t\} \in E(T) \text{ is a minimum capacity } s\text{-}t\text{-cut}$$

Proof. "$\Rightarrow$" follows directly from the definition. For the other direction, let $s, t \in V(G)$ and $e = \{u, v\} \in \arg\min_{e \in E(T_{s,t})} \lambda_{uv}$. Without loss of generality, $s \in C_e$, $t \in V(G) \setminus C_e$, so $\delta(C_e)$ is an s-t-cut. Therefore: $\lambda_{st} \leq u(\delta(C_e)) = \lambda_e$ (with $\lambda_e \coloneqq \lambda_{uv}$). By lemma 2.20 and induction, $\lambda_{st} \geq \min\{\lambda_{v'w'} \mid \{v', w'\} \in E(T_{[s,t]})\} = \lambda_{uv}$. Therefore $\lambda_{st} = \lambda_{uv}$. □

Idea: Choose $r, s \in V(G)$ and compute a minimum capacity r-s-cut $\delta(R)$. Without loss of generality $r \in R$. Construct a graph G_R by shrinking $S \coloneqq V(G) \setminus R$ into a single vertex. Find a minimum capacity p-q-cut (where $p, q \in R$ are chosen arbitrarily) in G_R. This partitions R into 2 parts. Continue this process until $V(G)$ is partitioned into singletons.

Lemma 2.21. *Let (G, u) as above, $s, t \in V(G)$, $\delta(A)$ a minimum capacity s-t-cut in G and $s', t' \in V(G) \setminus A$. Let (G', u') arise from (G, u) by contracting A into a single vertex. Then for any minimum capacity s'-t'-cut $\delta_{G'}(K \cup \{A\})$ in (G', u'), $\delta_G(K \cup A)$ is a minimum capacity s'-t'-cut in (G, u).*

Proof. Without loss of generality, $s \in A$. We show: $\exists$ min. capacity s'-t'-cut $\delta(A')$ in (G, u) such that $A \subseteq A'$. Let $\delta(C)$ be any s'-t'-cut in (G, u). Without loss of generality, $s \in C$. $u(\delta(\cdot))$ is a submodular function, i.e. $u(\delta(A)) + u(\delta(B)) \geq u(\delta(A \cap B)) + u(\delta(A \cup B))$ [6].

$\delta(A \cap C)$ is an s-t-cut, so $u(\delta(A \cap C)) \geq \lambda_{st} = u(\delta(A))$. Therefore, $u(\delta(A \cup C)) \leq u(\delta(C)) = \lambda_{s't'}$. Since $s' \in A \cup C$, $A \cup C$ is a minimum capacity s'-t'-cut. □

In general, we now choose a component X wih $|X| \geq 2$. Contract connected components in $T - \{X\}$, yielding a graph (G', u'). Choose $s, t \in X$, minimum s-t-cut $\delta(A')$ in (G', u'). $X = (X \cap A') \dot\cup (X \cap (V(G') \setminus A'))$.

Lemma 2.22. *At the end of MinCut:*

1. *$A \dot\cup B = V(G)$*
2. *$E(A, B)$ is a minimum s-t-cut in (G, u)*

[5] $\delta(C_e)$ is called *fundamental cut* induced by e

[6] This holds with equality, if we add $2u(E(A, B))$ to the right side

Proof. Elements of $V(T)$ are non-empty subsets of $V(G)$ and $V(T)$ form a partition of $V(G)$. Therefore $A \dot{\cup} B$ is a partition of $V(G)$. 2. follows from successive application of lemma 2.21 to each connected component of $T - X$. □

Lemma 2.23. *At any time before FinishTree: $w(e) = u(\delta_G(\bigcup_{Z \in C_e} Z))$ for all $e \in E(T)$. Moreover, $\forall e = \{P, Q\} \in E(T)$ there exist $p \in P, q \in Q$: $w(e) = \lambda_{pq}$.*

Proof. At the start, $E(T) = \emptyset$. We show that both properties are always satisfied. Let X, s, t, A', B', A, B as determined by ChooseComponents, Contract and MinCut. Edges in $E(T) \setminus \delta(X)$ are not affected. For new edges both conditions are true after ModifyTree.

Consider an edge $e \in \{Y, X\}$ that is replaced by e' in ModifyTree. Without loss of generality $Y \subseteq A$, so $e' = \{X \cap A, Y\}$. We show that both statements hold for e'. $w(e) = w(e') = u(\delta_G(\bigcup_{Z \in C_e} Z)) = u(\delta_G(\bigcup_{Z \in C_{e'}} Z'))$ so 1. holds. Assume $p \in X, q \in Y$: $\lambda_{pq} = w(e)$. If $p \in X \cap A$, we are done.

If $p \in X \cap B$, we claim: $\lambda_{sq} = \lambda_{pq}$. This then implies $w(e') = w(e) = \lambda_{pq} = \lambda_{sq}$. By lemma 2.20, $\lambda_{sq} \geq \min\{\lambda_{st}, \lambda_{tp}, \lambda_{pq}\}$. By lemma 2.22, $E(A, B)$ is a minimum s-t-cut. By lemma 2.21 and since $s, q \in A$, λ_{sq} does not change when contracting B. Adding $\{t, p\}$ with sufficiently high capacity does not change λ_{sq}. Therefore $\lambda_{sq} \geq \min\{\lambda_{st}, \lambda_{pq}\} = \lambda_{pq}$ because $E(A, B)$ is also a p-q-cut. $w(e)$ is the capacity of a cut separating s, q, so $\lambda_{sq} \leq w(e) = \lambda_{pq}$. □

Theorem 2.24 (Min Cut, Gomory & Hu (1961))**.** *Every undirected graph G with edge capacities $e : E(G) \to \mathbb{R}_{\geq 0}$ has a Gomory-Hu-tree. It can be computed using $n - 1$ Min-s-t-cut computations, e.g. in $O(n^3\sqrt{m})$ time (using the Push-Relabel algorithm for computing the minimum cuts) where $n := |V(G)|$ and $m := |E(G)|$.*

Proof. Algorithm-Hu-Algorithm computes a Gomory-Hu-tree (lemma 2.23). It uses $n-1$ iterations in each of which we need $O(n^2\sqrt{m})$ for Push-Relabel. Everything else can be handled in $O(\min\{n^3, n^2 m\})$ time. □

2.5 Finding Minimum-Capacity T-Cuts

Theorem 2.25 (Padberg & Rao (1987))**.** *Given a graph $G, u : E(G) \to \mathbb{R}_{\geq 0}$, a Gomory-Hu-tree H for (G, u), $T \subseteq V(G)$ ($|T| \geq 2$ even), a minimum capacity T-cut can be found among the fundamental cuts of H. A minimum capacity T-cut can be computed in $O(n^3\sqrt{m})$ time.*

Proof. Let $\delta_G(X)$ be a minimum capacity T-cut in G. Let J be the set of edges in $E(H)$ for where $|C_e \cap T|$ is odd (where C_e is a connected component of $H - e$). For all $x \in V(G)$:

$$\begin{aligned} |\delta_J(x)| &\equiv \sum_{e \in \delta_H(x)} |C_e \cap T| \\ &\overset{T \text{ even}}{\equiv} |\{x\} \cap T| \qquad \mod 2 \end{aligned}$$

Therefore J is a T-join in H. Since T-cuts and T-joins intersect, there is $f \in J \cap \delta_H(X)$.

$$\begin{aligned} u(\delta_G(X)) &\geq \min\{u(\delta_G(Y)) \mid |Y \cap f| = 1\} \\ &= u(\delta_G(C_f)) \end{aligned}$$

We conclude that $\delta_G(C_f)$ is a minimum-capacity T-cut. □

2.6 b-Matchings

Definition 2.26. Let G be a graph, $u : E(G) \to \mathbb{N}_0 \cup \{\infty\}$ and $b : V(G) \to \mathbb{N}_0$. A *$b$-matching* is a function $f : E(G) \to \mathbb{N}_0$ such that $f(e) \leq u(e)$ and $f(\delta(v)) \leq b(v)$ for all $e \in E(G)$ and $v \in V(G)$.

- If $u \equiv 1$, the instance is called *simple*.
- If $b \equiv 1$, this is equivalent to a matching.
- If $f(\delta(v)) = b(v)$ for all $v \in V(G)$, it is called *perfect*.
- Simple perfect b-matchings are called *b-factors*.

Example. A TSP-tour is a 2-factor. Therefore valid inequalities for 2-factors are valid for TSP.

Theorem 2.27 (Edmonds (1965))**.** *Let G be a graph, $b : V(G) \to \mathbb{N}$. The b-matching polytope of (G, ∞) is the set of vectors $x \in \mathbb{R}^{E(G)}_{\geq 0}$ satisfying:*

$$\begin{aligned} x_e &\geq 0 && e \in E(G) \\ x(\delta(v)) &\leq b(c) && v \in V(G) \\ \sum_{e \in E(G[X])} x_e &\leq \lfloor \frac{1}{2} \sum_{v \in X} b(v) \rfloor && X \subseteq V(G) \end{aligned}$$

Proof. Clearly, any b-matching satisfies these inequalities. Let x satisfy the inequalities. Without loss of generality $b \geq 1$. Define H by splitting each $v \in V(G)$ into $b(v)$ copies. Define:

$$\begin{aligned} X_v &:= \{(v, i) \mid i \in [b(v)]\} && v \in V(G) \\ V(H) &:= \bigcup_{v \in V(G)} X_v \\ E(H) &:= \{\{v', w'\} \mid \{v, w\} \in E(G), v' \in X_v, w' \in X_w\} \\ y_{e'} &:= \frac{1}{b(v) \cdot b(w)} x_{\{v,w\}} && e' = \{v', w'\} \in E(H), v' \in X_v, w' \in X_w \end{aligned}$$

Claim. *y is a convex combination of matchings in H. Contracting all X_v ($v \in V(G)$) yields a convex combination of b-matchings for x.*

We show that Y is contained in the matching polytope, i.e.:

$$\begin{aligned} y_e &\geq 0 && e \in E(H) \\ \sum_{e \in E(H[A])} y_2 &\leq \frac{|A|-1}{2} && A \subseteq V(H),\ |A| \text{ odd} \end{aligned}$$

If $\forall v \in V(H)$: $X_v \subseteq A$ or $X_v \cap A = \emptyset$, this follows directly from the given inequalities. Otherwise, let $a, b \in X_v$ such that $a \in A, b \notin A$.

$$\begin{aligned} 2\sum_{e \in E(H[A])} y_e &= \sum_{c \in A \setminus \{a\}} \sum_{e \in E(\{c\}, A \setminus \{c\})} y_e + \sum_{e \in E(\{a\}, A \setminus \{a\})} y_e \\ &\leq \sum_{c \in A \setminus \{a\}} \sum_{e \in \delta_H(c) \setminus \{\{c,b\}\}} + \sum_{e \in E(\{a\}, A \setminus \{a\})} y_e \\ &= \sum_{c \in A \setminus \{a\}} \sum_{e \in \delta_H(c)} y_e \underbrace{- \sum_{e \in E(\{b\}, A \setminus \{a\})} y_e + \sum_{e \in E(\{a\}, A \setminus \{a\})} y_e}_{=0} \\ &\leq |A| - 1 \end{aligned}$$

□

Theorem 2.28 (Edmonds & Johnson (1970))**.** *Let G be a graph, $u : E(G) \to \mathbb{N} \cup \{\infty\}$, $b : V(G) \to \mathbb{N}$. The b-matching polytope is given by:*

$$\begin{aligned} x &\geq 0 \\ x &\leq u \\ x(\delta(v)) &\leq b(v) && v \in V(G) \\ \sum_{e \in E(G[X])} x_e + \sum_{e \in F} x_e &\leq \underbrace{\lfloor \frac{1}{2} \left(\sum_{v \in X} b(v) + \sum_{e \in F} u(e) \right) \rfloor}_{\text{Gomory-Chvátal-Cut}} && X \subseteq V(G), F \subseteq \delta(X) \end{aligned}$$

Proof.

"$\subseteq$": Let x be an incidence vector of b-matchings. Then $x \leq u$ and $x(\delta(v)) \leq b(v)$ for all $v \in V(G)$.

$$\begin{aligned} \sum_{e \in E(G[X])} x_e + \sum_{e \in F} x_e &= \frac{1}{2} \left(\sum_{v \in X} \sum_{e \in \delta(x)} x_e + \sum_{e \in F} x_e - \sum_{e \in \delta(X) \setminus F} x_e \right) \\ &\leq \frac{1}{2} \left(\sum_{v \in X} b(v) + \sum_{e \in F} u(e) \right) \end{aligned}$$

Since the left hand side is integral, the right hand side can be rounded down.

"$\supseteq$": Let x satisfy all the inequalities. We have to show that x is a convex combinations of b-matchings. Let H arise from G by subdividing each edge $e = \{v, w\}$ with $u(e) \neq \infty$ by 2 new vertices $(e, v), (e, w)$ and a path v-(e, v)-(e, w)-w, where $b((e, v)) = u(e) = b((e, w))$. Set $y_{\{v,(e,v)\}} := x_e =: y_{\{(e,w),w\}}$ and $y_{\{(e,v),(e,w)\}} := u(e) - x_e$. If $u(e) = \infty$, $y_e := x_e$.

Claim. *y is in the b-matching polytope of (H, ∞). This then implies that x is contained in the capacitated b-matching polytope of (G, u).*

$y(\delta_H(v)) \leq b(v)$ clearly holds for all $v \in V(H)$.

$\square$