Networks and Combinatorial Optimization

(A)Math 514 — Autumn 2020

Thomas Rothvoss

UNIVERSITY *of* WASHINGTON

Compiled on Tuesday 8th December, 2020, 21:29

Lecture I

Organization

+

Chapter 1.4 — Minimum Spanning Trees

Organization

Lecturer: Thomas Rothvoss (`rothvoss@uw.edu`)
TA / grader: Andrew Pryhuber (`pryhuber@uw.edu`)

- Webpage: `https://canvas.uw.edu/courses/1395403`
- Lecture video posted MW.
  Each video equivalent to 80min whiteboard lecture (on average)
- Lecture notes of Lex Schrijver (see webpage)
- Weekly homework
  - posted Friday's, due following Friday on GradeScope
  - 1st homework posted Friday Oct 2, due Friday Oct 9
  - Recommended: Submission in groups of 2-3 students
- Office hours on Zoom
  - Monday 10am-11am
  - Wednesday 11am-12pm
- No exam

What is Combinatorial Optimization

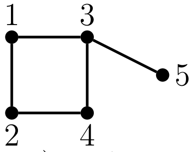
- **Combinatorial optimization:** finding the **best solution** out of **finite** number of possibilities in a computationally **efficient way**.
- Need to understand **problem structure** in order to succeed.

Chapter 1.4

Minimum Spanning Trees

Graph Theory (1)

- An **undirected graph** $G = (V, E)$ is a pair of sets.
  $V =$ **vertices** (finite set)
  $E =$ **edges** (unordered pairs of vertices)

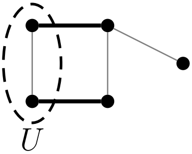


Graph $G = (V, E)$ with $V = \{1, 2, 3, 4, 5\}$
and $E = \{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 4\}, \{3, 5\}\}$

- We also write $G = (V(G), E(G))$

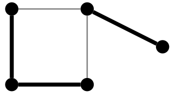
Graph Theory (2)

- A subset $U \subseteq V$ induces a **cut** $\delta(U) = \{\{u,v\} \in E \mid |\{u,v\} \cap U| = 1\}$



Graph Theory (3)

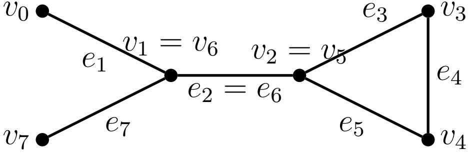
- A **subgraph** of $G = (V(G), E(G))$ is a graph $H = (V(H), E(H))$ where $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$ with the restriction that if $\{i, j\} \in E(H)$ then $i, j \in V(H)$.
- If $V' \subseteq V(G)$, then the subgraph **induced** by V' is the graph $(V', E(V'))$ where $E(V')$ is the set of all edges in G for which both vertices are in V'.
- A subgraph H of G is a **spanning subgraph** of G if $V(H) = V(G)$.



spanning subgraph

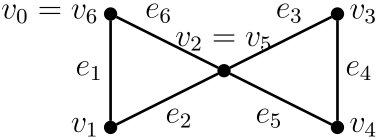
Graph Theory (4)

- A **walk** in a graph $G = (V, E)$ is a sequence of vertices and edges $v_0, e_1, v_1, e_2, v_2, e_3, \ldots, e_k, v_k$, such that for $i = 0, \ldots, k$, $v_i \in V$, $e_i \in E$ where $e_i = \{v_{i-1}, v_i\}$.



a walk of length 7

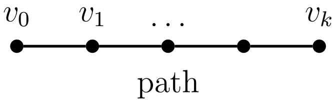
- If $v_0 = v_k$, then this is a **closed walk**.



closed walk of length 6

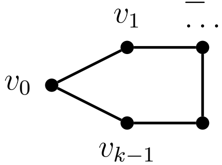
Graph Theory (5)

- A **path** is a graph $P = (V, E)$ where $V = \{v_0, v_1, \ldots, v_k\}$ and $E = \{\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_{k-1}, v_k\}\}$ and all $v_0, \ldots, v_k$ are distinct.
- The **length** of the path is the number of edges in the path which equals k.
- Called (v_0, v_k)**-path** if we want to emphasize the endpoints.



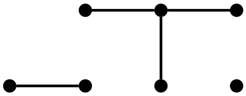
Graph Theory (6)

- A **cycle** is a graph $G = (V, E)$ with $V = \{v_0, v_1, \ldots, v_{k-1}\}$ and $E = \{\{v_0, v_1\}, \{v_1, v_2\}, \ldots, \{v_{k-1}, v_0\}\}$ where $v_0, \ldots, v_{k-1}$ are distinct and $k \geq 3$.

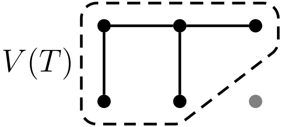


cycle with $k = 5$ vertices and edges

- A graph G is **acyclic** if it contains no cycle as subgraphs.
- An acyclic graph is called a **forest**. A connected forest $T = (V(T), E(T))$ is a **tree**.



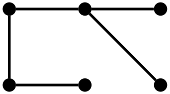
forest



tree T

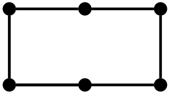
Graph Theory (7)

- $T = (V(T), E(T))$ is a **spanning tree** of G, if T is spanning, connected and acyclic.



spanning tree

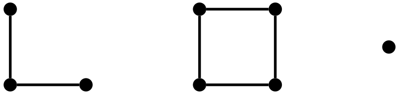
- A **Hamiltonian circuit** of G is a subgraph that is a spanning cycle.



Hamiltonian circuit/tour

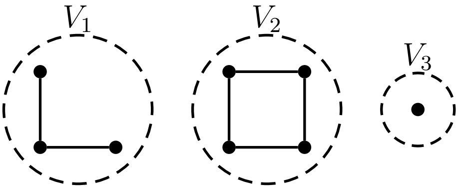
Graph Theory (8)

- Define **equivalence relation** $u \sim v$ if there exists a u-v path in G
- Call the equivalence classes $V_1, \ldots, V_k$.
- Then the induced subgraphs $G[V_1], \ldots, G[V_k]$ are called **(connected) components**.



Graph Theory (8)

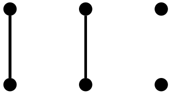
- Define **equivalence relation** $u \sim v$ if there exists a u-v path in G
- Call the equivalence classes $V_1, \ldots, V_k$.
- Then the induced subgraphs $G[V_1], \ldots, G[V_k]$ are called **(connected) components**.



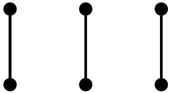
(connected) components

Graph Theory (9)

- A set $M \subseteq E$ of edges with degree ≤ 1 for each vertex is called **matching**.
- A set $M \subseteq E$ of edges with degree exactly 1 for each vertex is called **perfect matching**.



matching

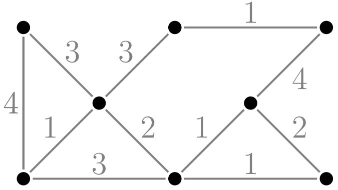


perfect matching

- **Convention:** Paths / trees / spanning trees / cycles / Hamiltonian circuits formally defined as **graphs** $H = (V(H), E(H))$. Often we call the edges $E(H)$ paths / trees etc.

Minimum Spanning Trees (1)

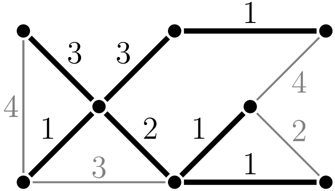
> MINIMUM SPANNING TREE
> **Input:** Undirected graph $G = (V, E)$, length function $\ell : E \to \mathbb{R}$
> **Goal:** A spanning tree T of G (i.e. $E(T) \subseteq E(G)$) minimizing $\ell(T) := \sum_{e \in E(T)} \ell(e)$.



Applications: Designing road systems, electrical power lines, telephone lines

Minimum Spanning Trees (1)

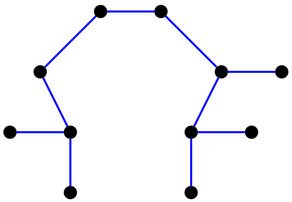
MINIMUM SPANNING TREE
Input: Undirected graph $G = (V, E)$, length function $\ell : E \to \mathbb{R}$
Goal: A spanning tree T of G (i.e. $E(T) \subseteq E(G)$) minimizing $\ell(T) := \sum_{e \in E(T)} \ell(e)$.



Applications: Designing road systems, electrical power lines, telephone lines

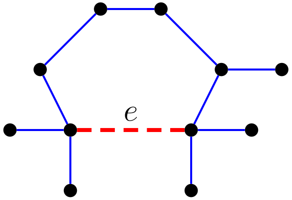
Useful facts about Spanning Trees

- **Fact 1.** Let T be a spanning connected subgraph of G. The following conditions are equivalent
  - T is a spanning tree (i.e. acyclic)
  - $|E(T)| = |V(T)| - 1$.
  - $\forall e = \{u, v\} \in E$ there exists a unique u-v path in T



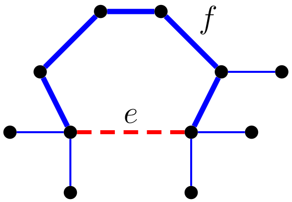
Useful facts about Spanning Trees

- **Fact 1.** Let T be a spanning connected subgraph of G. The following conditions are equivalent
  - T is a spanning tree (i.e. acyclic)
  - $|E(T)| = |V(T)| - 1$.
  - $\forall e = \{u, v\} \in E$ there exists a unique u-v path in T
- **Fact 2.** Let T be a spanning tree in G. Suppose $e \notin E(T)$ and f is any edge on the unique path in T between the end points of e. Then $(V(T), E(T) \setminus \{f\} \cup \{e\})$ is again a spanning tree.



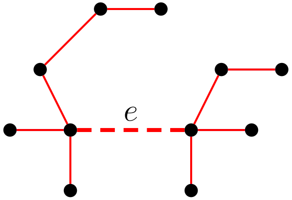
Useful facts about Spanning Trees

- **Fact 1.** Let T be a spanning connected subgraph of G. The following conditions are equivalent
  - T is a spanning tree (i.e. acyclic)
  - $|E(T)| = |V(T)| - 1$.
  - $\forall e = \{u, v\} \in E$ there exists a unique u-v path in T
- **Fact 2.** Let T be a spanning tree in G. Suppose $e \notin E(T)$ and f is any edge on the unique path in T between the end points of e. Then $(V(T), E(T) \setminus \{f\} \cup \{e\})$ is again a spanning tree.



Useful facts about Spanning Trees

- **Fact 1.** Let T be a spanning connected subgraph of G. The following conditions are equivalent
  - T is a spanning tree (i.e. acyclic)
  - $|E(T)| = |V(T)| - 1$.
  - $\forall e = \{u, v\} \in E$ there exists a unique u-v path in T
- **Fact 2.** Let T be a spanning tree in G. Suppose $e \notin E(T)$ and f is any edge on the unique path in T between the end points of e. Then $(V(T), E(T) \setminus \{f\} \cup \{e\})$ is again a spanning tree.



Dijkstra-Prim Algorithm

Dijkstra-Prim Algorithm

Input: A connected graph G with edge costs $\ell : E \to \mathbb{R}$.

Output: A MST T of G.

(1) Choose any $v \in V$ and set $T := (\{v\}, \emptyset)$

(2) WHILE $V(T) \neq V$

  (3) Choose $e \in \delta(T)$ of minimal length

  (4) Add e to $E(T)$ (and include endpoint of e)

Dijkstra-Prim Algorithm

Dijkstra-Prim Algorithm

Input: A connected graph G with edge costs $\ell : E \to \mathbb{R}$.

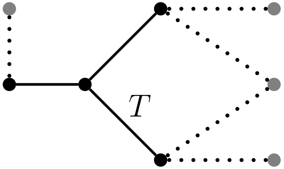
Output: A MST T of G.

(1) Choose any $v \in V$ and set $T := (\{v\}, \emptyset)$

(2) WHILE $V(T) \neq V$

  (3) Choose $e \in \delta(T)$ of minimal length

  (4) Add e to $E(T)$ (and include endpoint of e)



Dijkstra-Prim Algorithm

Dijkstra-Prim Algorithm

Input: A connected graph G with edge costs $\ell : E \to \mathbb{R}$.

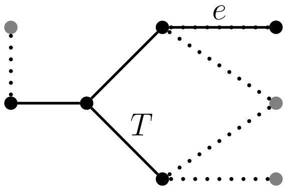
Output: A MST T of G.

(1) Choose any $v \in V$ and set $T := (\{v\}, \emptyset)$

(2) WHILE $V(T) \neq V$

  (3) Choose $e \in \delta(T)$ of minimal length

  (4) Add e to $E(T)$ (and include endpoint of e)



Dijkstra-Prim Algorithm

Dijkstra-Prim Algorithm

Input: A connected graph G with edge costs $\ell : E \to \mathbb{R}$.

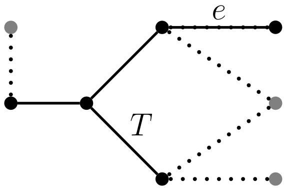
Output: A MST T of G.

(1) Choose any $v \in V$ and set $T := (\{v\}, \emptyset)$

(2) WHILE $V(T) \neq V$

  (3) Choose $e \in \delta(T)$ of minimal length

  (4) Add e to $E(T)$ (and include endpoint of e)



Note:

- This is a "Greedy algorithm"
- Earliest algorithm for finding MST due to Boruvka (1926). Variants by Dijkstra (1959), Prim (1957)

Proof of correctness

Definition

A forest F is called **greedy** if $\exists$ minimum spanning tree T with $E(F) \subseteq E(T)$.

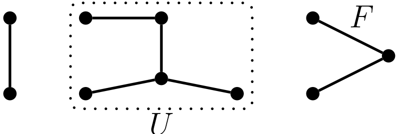
Theorem

Let F be a greedy forest, U be one of the connected components. If $e \in \delta(U)$ is an edge of minimum length in $\delta(U)$, then $F \cup \{e\}$ is again a greedy forest.

Proof of correctness (2)

Proof.

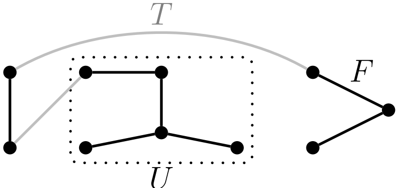
- Let F be a greedy forest with component U.



Proof of correctness (2)

Proof.

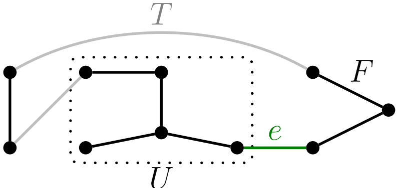
- Let F be a greedy forest with component U.
- Let T be a MST with $E(T) \supseteq E(F)$.



Proof of correctness (2)

Proof.

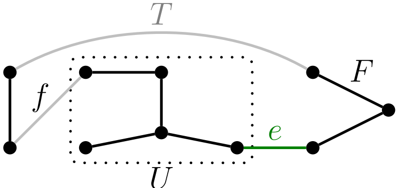
- Let F be a greedy forest with component U.
- Let T be a MST with $E(T) \supseteq E(F)$.
- Select $e \in \delta(U)$ of min. length



Proof of correctness (2)

Proof.

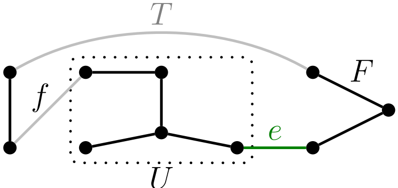
- Let F be a greedy forest with component U.
- Let T be a MST with $E(T) \supseteq E(F)$.
- Select $e \in \delta(U)$ of min. length
- Let P be unique path in T between endpoints of e
- Select $f \in E(P) \cap \delta(U)$



Proof of correctness (2)

Proof.

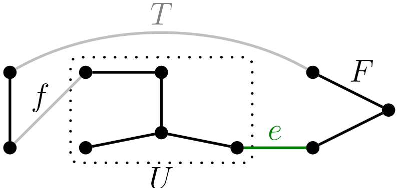
- Let F be a greedy forest with component U.
- Let T be a MST with $E(T) \supseteq E(F)$.
- Select $e \in \delta(U)$ of min. length
- Let P be unique path in T between endpoints of e
- Select $f \in E(P) \cap \delta(U)$
- We have $\ell(e) \leq \ell(f)$ by assumption



Proof of correctness (2)

Proof.

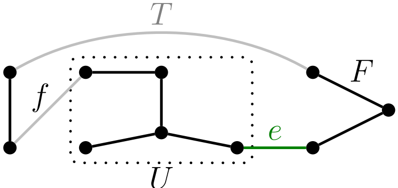
- Let F be a greedy forest with component U.
- Let T be a MST with $E(T) \supseteq E(F)$.
- Select $e \in \delta(U)$ of min. length
- Let P be unique path in T between endpoints of e
- Select $f \in E(P) \cap \delta(U)$
- We have $\ell(e) \leq \ell(f)$ by assumption
- $T' := (V, (E(T) \setminus \{f\}) \cup \{e\})$ is a spanning tree. Moreover $\ell(T') = \ell(T) - \ell(f) + \ell(e) \leq \ell(T)$.



Proof of correctness (2)

Proof.

- Let F be a greedy forest with component U.
- Let T be a MST with $E(T) \supseteq E(F)$.
- Select $e \in \delta(U)$ of min. length
- Let P be unique path in T between endpoints of e
- Select $f \in E(P) \cap \delta(U)$
- We have $\ell(e) \leq \ell(f)$ by assumption
- $T' := (V, (E(T) \setminus \{f\}) \cup \{e\})$ is a spanning tree. Moreover $\ell(T') = \ell(T) - \ell(f) + \ell(e) \leq \ell(T)$.
- So T' is an MST that includes $F \cup \{e\}$



Proof of correctness (3)

Corollary

The Dijkstra-Prim algorithm yields a MST of G.

Proof of correctness (3)

Corollary

The Dijkstra-Prim algorithm yields a MST of G.

Proof.

- At start $T = (\{v\}, \emptyset)$ is a greedy forest
- By Theorem, in every step T remains a greedy forest
- At end, T is a spanning tree that is greedy.
- By def. $\exists$MST $T^* : E(T) \subseteq E(T^*)$. Must have $T^* = T$.

Kruskal's algorithm

Kruskal's Algorithm

Input: A connected graph G with edge costs $\ell : E \to \mathbb{R}$.
Output: A MST T of G.

(1) Sort the edges such that $\ell(e_1) \leq \ell(e_2) \leq \ldots \leq \ell(e_m)$.

(2) Set $T = (V, \emptyset)$

(3) For i from 1 to m do
If $T \cup \{e_i\}$ is acyclic then update $T := T \cup \{e_i\}$.

Kruskal's algorithm

Kruskal's Algorithm

Input: A connected graph G with edge costs $\ell : E \to \mathbb{R}$.
Output: A MST T of G.

(1) Sort the edges such that $\ell(e_1) \leq \ell(e_2) \leq \ldots \leq \ell(e_m)$.

(2) Set $T = (V, \emptyset)$

(3) For i from 1 to m do
    If $T \cup \{e_i\}$ is acyclic then update $T := T \cup \{e_i\}$.

Theorem

Kruskal's algorithm computes an MST.

Kruskal's algorithm

Kruskal's Algorithm

Input: A connected graph G with edge costs $\ell : E \to \mathbb{R}$.
Output: A MST T of G.

(1) Sort the edges such that $\ell(e_1) \leq \ell(e_2) \leq \ldots \leq \ell(e_m)$.

(2) Set $T = (V, \emptyset)$

(3) For i from 1 to m do
If $T \cup \{e_i\}$ is acyclic then update $T := T \cup \{e_i\}$.

Theorem

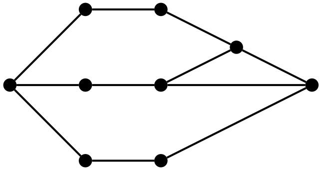
Kruskal's algorithm computes an MST.

Proof.

- $(V, \emptyset)$ is a greedy forest
- In every step we add a cheapest edge crossing one of the connected components
- So we terminate with a connected greedy forest $\to$ MST

The Maximum Reliability Problem

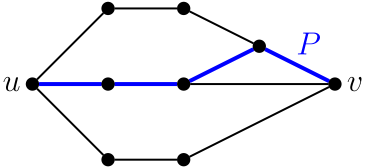
- Given graph $G = (V, E)$, function $s : E \to \mathbb{R}_{\geq 0}$ ($s(e)$ gives **strength** of an edge)



The Maximum Reliability Problem

- Given graph $G = (V, E)$, function $s : E \to \mathbb{R}_{\geq 0}$ ($s(e)$ gives **strength** of an edge)
- For a path P in G, define **reliability**

$$r(P) := \min_{e \in E(P)} s(e)$$



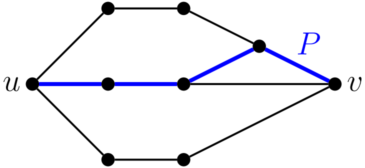
The Maximum Reliability Problem

- Given graph $G = (V, E)$, function $s : E \to \mathbb{R}_{\geq 0}$ ($s(e)$ gives **strength** of an edge)
- For a path P in G, define **reliability**

$$r(P) := \min_{e \in E(P)} s(e)$$

- For a pair $u, v \in V$, define the **reliability**

$$r_G(u, v) := \max\{r(P) : P \text{ is } u\text{-}v \text{ path in } G\}$$



The Maximum Reliability Problem (2)

Theorem

Let T be a spanning tree in G maximizing $\sum_{e \in E(T)} s(e)$. Then $r_T(u,v) = r_G(u,v)\ \forall u, v \in V$.

The Maximum Reliability Problem (2)

Theorem

Let T be a spanning tree in G maximizing $\sum_{e \in E(T)} s(e)$. Then $r_T(u, v) = r_G(u, v)\ \forall u, v \in V$.

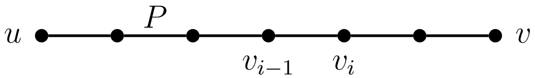
- Suffices to prove that $r_T(u, v) \geq r_G(u, v)$.

The Maximum Reliability Problem (2)

Theorem

Let T be a spanning tree in G maximizing $\sum_{e \in E(T)} s(e)$. Then $r_T(u,v) = r_G(u,v)\ \forall u,v \in V$.

- Suffices to prove that $r_T(u,v) \geq r_G(u,v)$.
- Fix u-v path P in G. Let $v_0, \ldots, v_m$ be nodes of P with $\{v_{i-1}, v_i\} \in E(P)$

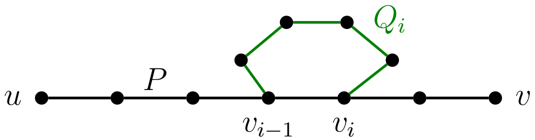


The Maximum Reliability Problem (2)

Theorem

Let T be a spanning tree in G maximizing $\sum_{e \in E(T)} s(e)$. Then $r_T(u,v) = r_G(u,v)\ \forall u,v \in V$.

- Suffices to prove that $r_T(u,v) \geq r_G(u,v)$.
- Fix u-v path P in G. Let $v_0, \ldots, v_m$ be nodes of P with $\{v_{i-1}, v_i\} \in E(P)$
- Let Q_i be unique v_{i-1}-v_i path in T
- We know $s(e) \geq s(v_{i-1}, v_i)\ \forall e \in E(Q_i)$ (by opt. of T)

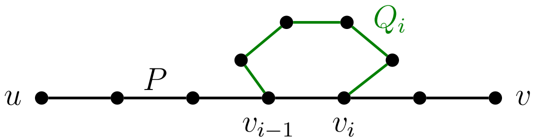


The Maximum Reliability Problem (2)

Theorem

Let T be a spanning tree in G maximizing $\sum_{e \in E(T)} s(e)$. Then $r_T(u,v) = r_G(u,v)\ \forall u,v \in V$.

- Suffices to prove that $r_T(u,v) \geq r_G(u,v)$.
- Fix u-v path P in G. Let $v_0, \ldots, v_m$ be nodes of P with $\{v_{i-1}, v_i\} \in E(P)$
- Let Q_i be unique v_{i-1}-v_i path in T
- We know $s(e) \geq s(v_{i-1}, v_i)\ \forall e \in E(Q_i)$ (by opt. of T)
- Let Q be concatenation of $Q_1, \ldots, Q_m$. Then Q is a u-v walk in T with $r(Q) \geq r(P)$. □



Lecture 2

Chapter 10.1 — Matroids and the greedy Algorithm

Kruskal's algorithm

Kruskal's algorithm to find MST in $G = (V, E)$ with $|E| = m$ and lengths $\ell : E \to \mathbb{R}$:

(1) Sort the edges such that $\ell(e_1) \leq \ell(e_2) \leq \ldots \leq \ell(e_m)$.

(2) Set $T = (V, \emptyset)$

(3) For i from 1 to m do

If $T \cup \{e_i\}$ is acyclic then update $T := T \cup \{e_i\}$.

Kruskal's algorithm

Kruskal's algorithm to find MST in $G = (V, E)$ with $|E| = m$ and lengths $\ell : E \to \mathbb{R}$:

(1) Sort the edges such that $\ell(e_1) \leq \ell(e_2) \leq \ldots \leq \ell(e_m)$.

(2) Set $T = (V, \emptyset)$

(3) For i from 1 to m do
If $T \cup \{e_i\}$ is acyclic then update $T := T \cup \{e_i\}$.

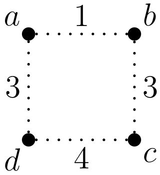
Theorem

Kruskal's algorithm finds an MST.

- Kruskal's algorithm is a **greedy algorithm**.

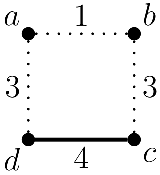
Greedy algorithm are not always optimal

Example: Find a maximum weight matching in $G = (V, E)$



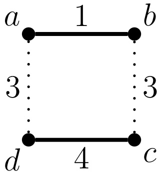
Greedy algorithm are not always optimal

Example: Find a maximum weight matching in $G = (V, E)$



Greedy algorithm are not always optimal

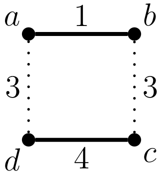
Example: Find a maximum weight matching in $G = (V, E)$



- Greedy matching has value 5
- Optimum value is 6

Greedy algorithm are not always optimal

Example: Find a maximum weight matching in $G = (V, E)$



- Greedy matching has value 5
- Optimum value is 6

Matroids are exactly the structures for which the greedy algorithm works!

Definition of Matroids (1)

Definition

A **matroid** is a pair $M = (X, \mathcal{I})$ where X is a finite set, $\mathcal{I} \subseteq 2^X$ s.t.

(i) $\emptyset \in \mathcal{I}$

(i) If $Y \in \mathcal{I}$ and $Z \subseteq Y$ then $Z \in \mathcal{I}$

(iii) If $Y, Z \in \mathcal{I}$ and $|Y| < |Z|$ then for some $x \in Z \setminus Y$ one has $Y \cup \{x\} \in \mathcal{I}$

Definition of Matroids (1)

Definition

A **matroid** is a pair $M = (X, \mathcal{I})$ where X is a finite set, $\mathcal{I} \subseteq 2^X$ s.t.

(i) $\emptyset \in \mathcal{I}$

(i) If $Y \in \mathcal{I}$ and $Z \subseteq Y$ then $Z \in \mathcal{I}$

(iii) If $Y, Z \in \mathcal{I}$ and $|Y| < |Z|$ then for some $x \in Z \setminus Y$ one has $Y \cup \{x\} \in \mathcal{I}$

- The set X is called the **ground set**

Definition of Matroids (1)

Definition

A **matroid** is a pair $M = (X, \mathcal{I})$ where X is a finite set, $\mathcal{I} \subseteq 2^X$ s.t.

(i) $\emptyset \in \mathcal{I}$

(i) If $Y \in \mathcal{I}$ and $Z \subseteq Y$ then $Z \in \mathcal{I}$

(iii) If $Y, Z \in \mathcal{I}$ and $|Y| < |Z|$ then for some $x \in Z \setminus Y$ one has $Y \cup \{x\} \in \mathcal{I}$

- The set X is called the **ground set**
- $\mathcal{I}$ are called the **independent** sets

Definition of Matroids (1)

Definition

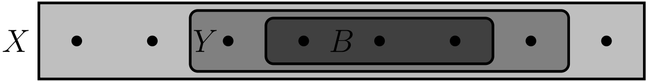
A **matroid** is a pair $M = (X, \mathcal{I})$ where X is a finite set, $\mathcal{I} \subseteq 2^X$ s.t.

(i) $\emptyset \in \mathcal{I}$

(i) If $Y \in \mathcal{I}$ and $Z \subseteq Y$ then $Z \in \mathcal{I}$

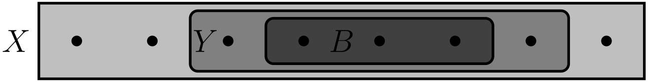
(iii) If $Y, Z \in \mathcal{I}$ and $|Y| < |Z|$ then for some $x \in Z \setminus Y$ one has $Y \cup \{x\} \in \mathcal{I}$

- The set X is called the **ground set**
- $\mathcal{I}$ are called the **independent** sets
- Let $Y \subseteq X$. A set $B \subseteq Y$ is called a **basis of Y** if $B \in \mathcal{I}$ and for all $x \in Y \setminus B$ one has $B \cup \{x\} \notin \mathcal{I}$



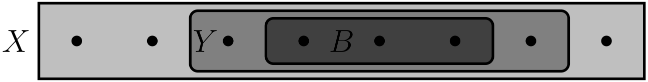
Definition of Matroids (2)

- Let $Y \subseteq X$. A set $B \subseteq Y$ is called a **basis of Y**, if $B \in \mathcal{I}$ and for all $x \in Y \setminus B$ one has $B \cup \{x\} \notin \mathcal{I}$



Definition of Matroids (2)

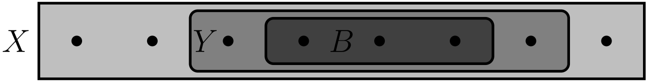
- Let $Y \subseteq X$. A set $B \subseteq Y$ is called a **basis of Y**, if $B \in \mathcal{I}$ and for all $x \in Y \setminus B$ one has $B \cup \{x\} \notin \mathcal{I}$



- **Fact.** For any two bases B_1, B_2 of Y have the same cardinality, i.e. $|B_1| = |B_2|$

Definition of Matroids (2)

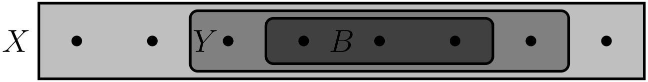
- Let $Y \subseteq X$. A set $B \subseteq Y$ is called a **basis of Y**, if $B \in \mathcal{I}$ and for all $x \in Y \setminus B$ one has $B \cup \{x\} \notin \mathcal{I}$



- **Fact.** For any two bases B_1, B_2 of Y have the same cardinality, i.e. $|B_1| = |B_2|$
- The **rank** of Y is $r_M(Y) = \max\{|B| : B \subseteq Y \text{ and } B \in \mathcal{I}\}$
- Define **rank of the matroid M** as $r_M(X)$

Definition of Matroids (2)

- Let $Y \subseteq X$. A set $B \subseteq Y$ is called a **basis of Y**, if $B \in \mathcal{I}$ and for all $x \in Y \setminus B$ one has $B \cup \{x\} \notin \mathcal{I}$



- **Fact.** For any two bases B_1, B_2 of Y have the same cardinality, i.e. $|B_1| = |B_2|$
- The **rank** of Y is $r_M(Y) = \max\{|B| : B \subseteq Y \text{ and } B \in \mathcal{I}\}$
- Define **rank of the matroid M** as $r_M(X)$

Lemma

Let $M = (X, \mathcal{I})$. Then M is a matroid if and only if

(i) $\emptyset \in \mathcal{I}$

(i) *If $Y \in \mathcal{I}$ and $Z \subseteq Y$ then $Z \in \mathcal{I}$*

(iii') *For all $Y \subseteq X$, all maximally independent subsets of Y have the same cardinality.*

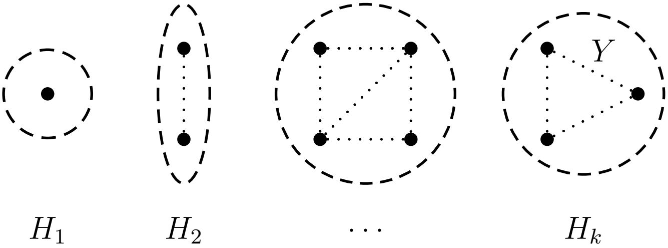
Example: Graphic matroid

- Let $G = (V, E)$ be an undirected graph. Then $(E, \mathcal{I})$ with $\mathcal{I} := \{F \subseteq E \mid F \text{ acyclic}\}$ is a **matroid**!
- That matroid is called the **graphic matroid**.

Example: Graphic matroid

- Let $G = (V, E)$ be an undirected graph. Then $(E, \mathcal{I})$ with $\mathcal{I} := \{F \subseteq E \mid F \text{ acyclic}\}$ is a **matroid**!
- That matroid is called the **graphic matroid**.
- For $Y \subseteq E$, let $H_1, \ldots, H_k$ be connected components of (V, Y)

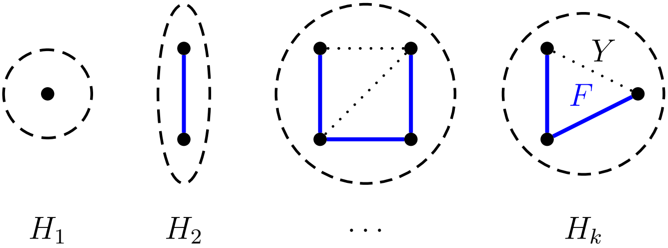
$$\mathrm{rk}_M(Y) = |V| - k = \sum_{i=1}^{k} (|V(H_i)| - 1)$$



Example: Graphic matroid

- Let $G = (V, E)$ be an undirected graph. Then $(E, \mathcal{I})$ with $\mathcal{I} := \{F \subseteq E \mid F \text{ acyclic}\}$ is a **matroid**!
- That matroid is called the **graphic matroid**.
- For $Y \subseteq E$, let $H_1, \ldots, H_k$ be connected components of (V, Y)

$$\mathrm{rk}_M(Y) = |V| - k = \sum_{i=1}^{k}(|V(H_i)| - 1)$$



Example: The linear matroid

- Let V be a vectorspace (for example $\mathbb{R}^n$). Pick $X := \{v_1, \dots, v_n\} \subseteq V$ and

  $$\mathcal{I} := \{Y \subseteq X \mid Y \text{ linearly independent}\}$$

  Then $M = (X, \mathcal{I})$ is a **matroid** (called a **linear matroid**)!

Example: The linear matroid

- Let V be a vectorspace (for example $\mathbb{R}^n$). Pick $X := \{v_1, \dots, v_n\} \subseteq V$ and

  $$\mathcal{I} := \{Y \subseteq X \mid Y \text{ linearly independent}\}$$

  Then $M = (X, \mathcal{I})$ is a **matroid** (called a **linear matroid**)!
- In particular for $Y \subseteq X$ one has

  $$\mathrm{rk}_M(Y) = \dim(\mathrm{span}(Y))$$

The Matroid Greedy Algorithm

The Matroid Greedy Algorithm

Input: Matroid $M = (X, \mathcal{I})$ and weight function $w : X \to \mathbb{R}$.
Output: A basis Y maximizing $w(Y) := \sum_{x \in Y} w(x)$.

(1) Sort the elements in $X = \{e_1, \ldots, e_n\}$ such that $w(e_1) \geq w(e_2) \geq \ldots \geq w(e_n)$.

(2) Set $Y := \emptyset$

(3) For i from 1 to n do
If $Y \cup \{e_i\} \in \mathcal{I}$ then update $Y := Y \cup \{e_i\}$.

The Matroid Greedy Algorithm

The Matroid Greedy Algorithm

Input: Matroid $M = (X, \mathcal{I})$ and weight function $w : X \to \mathbb{R}$.
Output: A basis Y maximizing $w(Y) := \sum_{x \in Y} w(x)$.

(1) Sort the elements in $X = \{e_1, \ldots, e_n\}$ such that $w(e_1) \geq w(e_2) \geq \ldots \geq w(e_n)$.

(2) Set $Y := \emptyset$

(3) For i from 1 to n do
If $Y \cup \{e_i\} \in \mathcal{I}$ then update $Y := Y \cup \{e_i\}$.

Theorem

Suppose $M = (X, \mathcal{I})$ satisfies conditions (i) and (ii). Then M is a matroid $\Leftrightarrow$ for any weight function $w : X \to \mathbb{R}$, the greedy algorithm finds a maximum weight basis.

The Matroid Greedy Algorithm (2)

Claim I. M matroid $\Rightarrow$ greedy algorithm finds optimum

The Matroid Greedy Algorithm (2)

Claim I. M matroid $\Rightarrow$ greedy algorithm finds optimum

Definition

We say $Y \in \mathcal{I}$ is **greedy** if there is a maximum weight basis B with $Y \subseteq B$.

The Matroid Greedy Algorithm (2)

Claim I. M matroid $\Rightarrow$ greedy algorithm finds optimum

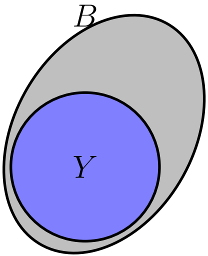
Definition

We say $Y \in \mathcal{I}$ is **greedy** if there is a maximum weight basis B with $Y \subseteq B$.

Claim II. Suppose $Y \in \mathcal{I}$ is greedy and $x \in X \setminus Y$ so that $Y \cup \{x\} \in \mathcal{I}$ and $w(x)$ maximal. Then $Y \cup \{x\}$ is greedy.

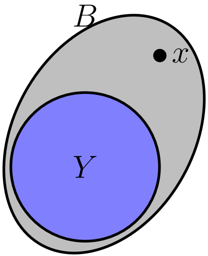
Claim II. Suppose $Y \in \mathcal{I}$ is greedy and $x \in X \setminus Y$ so that $Y \cup \{x\} \in \mathcal{I}$ and $w(x)$ maximal. Then $Y \cup \{x\}$ is greedy.

- Let B be maximum weight basis with $B \supseteq Y$.



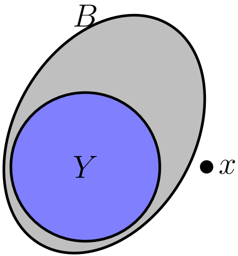
Claim II. Suppose $Y \in \mathcal{I}$ is greedy and $x \in X \setminus Y$ so that $Y \cup \{x\} \in \mathcal{I}$ and $w(x)$ maximal. Then $Y \cup \{x\}$ is greedy.

- Let B be maximum weight basis with $B \supseteq Y$.
- If $x \in B$, then $Y \cup \{x\} \subseteq B$ and we are done!



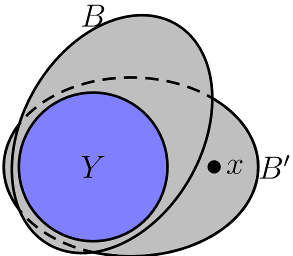
Claim II. Suppose $Y \in \mathcal{I}$ is greedy and $x \in X \setminus Y$ so that $Y \cup \{x\} \in \mathcal{I}$ and $w(x)$ maximal. Then $Y \cup \{x\}$ is greedy.

- Let B be maximum weight basis with $B \supseteq Y$.
- If $x \in B$, then $Y \cup \{x\} \subseteq B$ and we are done!
  So suppose $x \notin B$.



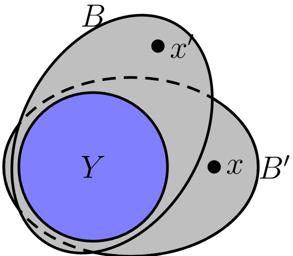
Claim II. Suppose $Y \in \mathcal{I}$ is greedy and $x \in X \setminus Y$ so that $Y \cup \{x\} \in \mathcal{I}$ and $w(x)$ maximal. Then $Y \cup \{x\}$ is greedy.

- Let B be maximum weight basis with $B \supseteq Y$.
- If $x \in B$, then $Y \cup \{x\} \subseteq B$ and we are done!
  So suppose $x \notin B$.
- Construct basis B' with $Y \cup \{x\} \subseteq B' \subseteq B \cup \{x\}$ (start with $Y \cup \{x\}$, then keep adding elements from B)



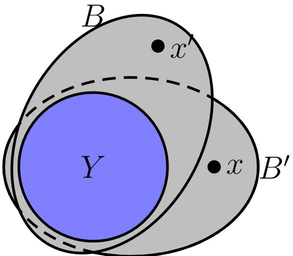
Claim II. Suppose $Y \in \mathcal{I}$ is greedy and $x \in X \setminus Y$ so that $Y \cup \{x\} \in \mathcal{I}$ and $w(x)$ maximal. Then $Y \cup \{x\}$ is greedy.

- Let B be maximum weight basis with $B \supseteq Y$.
- If $x \in B$, then $Y \cup \{x\} \subseteq B$ and we are done! So suppose $x \notin B$.
- Construct basis B' with $Y \cup \{x\} \subseteq B' \subseteq B \cup \{x\}$ (start with $Y \cup \{x\}$, then keep adding elements from B)
- As $|B'| = |B|$ there is a unique element $x' \in B \setminus B'$.



Claim II. Suppose $Y \in \mathcal{I}$ is greedy and $x \in X \setminus Y$ so that $Y \cup \{x\} \in \mathcal{I}$ and $w(x)$ maximal. Then $Y \cup \{x\}$ is greedy.

- Let B be maximum weight basis with $B \supseteq Y$.
- If $x \in B$, then $Y \cup \{x\} \subseteq B$ and we are done!
  So suppose $x \notin B$.
- Construct basis B' with $Y \cup \{x\} \subseteq B' \subseteq B \cup \{x\}$ (start with $Y \cup \{x\}$, then keep adding elements from B)
- As $|B'| = |B|$ there is a unique element $x' \in B \setminus B'$.
- By choice of x, $w(x) \geq w(x')$ and so $w(B') \geq w(B)$. Then $Y \cup \{x\}$ is greedy! □



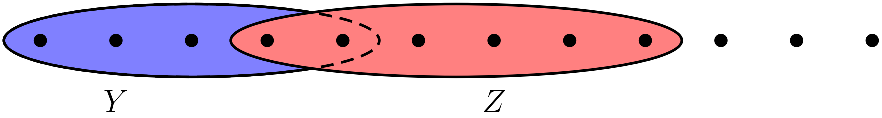
The Matroid Greedy Algorithm (3)

Claim III. (iii) not satisfied $\Rightarrow$ greedy fails for some w

The Matroid Greedy Algorithm (3)

Claim III. (iii) not satisfied $\Rightarrow$ greedy fails for some w

- Let $Y, Z \in \mathcal{I}$ with $k := |Y| < |Z|$ and $Y \cup \{z\} \notin \mathcal{I}$ for all $z \in Z \setminus Y$

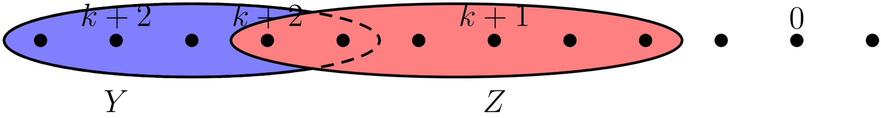


The Matroid Greedy Algorithm (3)

Claim III. (iii) not satisfied $\Rightarrow$ greedy fails for some w

- Let $Y, Z \in \mathcal{I}$ with $k := |Y| < |Z|$ and $Y \cup \{z\} \notin \mathcal{I}$ for all $z \in Z \setminus Y$
- Define

$$w(x) := \begin{cases} k+2 & \text{if } x \in Y \\ k+1 & \text{if } x \in Z \setminus Y \\ 0 & \text{if } x \in X \setminus (Y \cup Z) \end{cases}$$



The Matroid Greedy Algorithm (3)

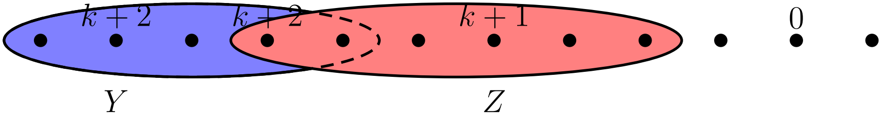
Claim III. (iii) not satisfied $\Rightarrow$ greedy fails for some w

- Let $Y, Z \in \mathcal{I}$ with $k := |Y| < |Z|$ and $Y \cup \{z\} \notin \mathcal{I}$ for all $z \in Z \setminus Y$
- Define

$$w(x) := \begin{cases} k+2 & \text{if } x \in Y \\ k+1 & \text{if } x \in Z \setminus Y \\ 0 & \text{if } x \in X \setminus (Y \cup Z) \end{cases}$$

- Greedy will pick Y (plus potentially some elements in $X \setminus (Y \cup Z)$). Greedy solution has value

$$w(Y) = k(k+2) < (k+1)^2 = w(Z) \qquad \square$$



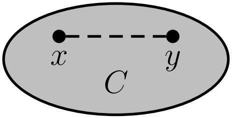
Lecture 3

Chapter 2 — Polytopes, polyhedra, Farkas' Lemma and Linear Programming — Part 1/3

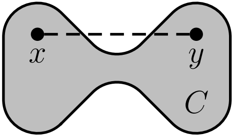
Convexity

Definition

A set $C \subseteq \mathbb{R}^n$ is **convex** if for all $x, y \in C$ and $0 \leq \lambda \leq 1$ one has $\lambda x + (1 - \lambda) y \in C$



convex



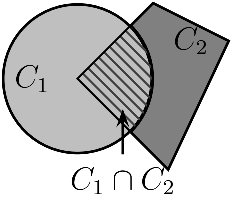
not convex

- Intuitively: For any pair of points $x, y \in C$, the line segment connecting them must lie inside C
- The point $\lambda x + (1 - \lambda) y$ is called a **convex combination** of x and y.

Convexity (2)

Lemma

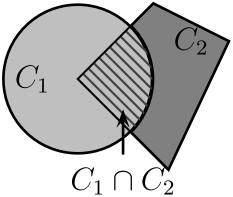
Let $C_i \subseteq \mathbb{R}^n$ be convex for $i \in I$. Then $\bigcap_{i \in I} C_i$ is convex.



Convexity (2)

Lemma

Let $C_i \subseteq \mathbb{R}^n$ be convex for $i \in I$. Then $\bigcap_{i\in I} C_i$ is convex.



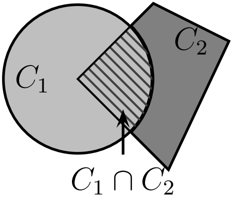
Proof.

- Let $x, y \in \bigcap_{i\in I} C_i$ and $0 \leq \lambda \leq 1$.
- For any $i \in I$,
  $x, y \in C_i \Rightarrow \lambda x + (1-\lambda)y \in C_i \Rightarrow \lambda x + (1-\lambda)y \in \bigcap_i C_i$

Convexity (2)

Lemma

Let $C_i \subseteq \mathbb{R}^n$ be convex for $i \in I$. Then $\bigcap_{i\in I} C_i$ is convex.



Proof.

- Let $x, y \in \bigcap_{i\in I} C_i$ and $0 \le \lambda \le 1$.
- For any $i \in I$,
  $x, y \in C_i \Rightarrow \lambda x + (1-\lambda)y \in C_i \Rightarrow \lambda x + (1-\lambda)y \in \bigcap_i C_i$

Conclusion: For any set $X \subseteq \mathbb{R}^n$ there is a unique smallest set containing X,

$$\operatorname{conv}(X) := \bigcap_{C \supseteq X : C \text{ is convex}} C$$

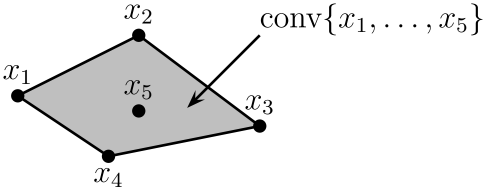
Convexity (2)

A more intuitive characterization:

Lemma

For any $X \subseteq \mathbb{R}^n$ one has

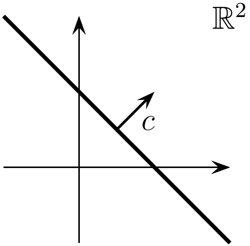
$$\operatorname{conv}(X) = \left\{ \sum_{i=1}^{t} \lambda_i x_i \;\middle|\; \begin{array}{c} x_1, \ldots, x_t \in X, \ \textit{and } \lambda_i \geq 0 \ \forall i \\ \textit{and } \sum_{i=1}^{t} \lambda_i = 1 \textit{ for some } t \end{array} \right\}$$



Hyperplanes

Definition

For $c \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ and $\delta \in \mathbb{R}$, the set $H = \{x \in \mathbb{R}^n \mid c^T x = \delta\}$ is called an **affine hyperplane**.

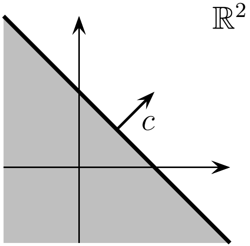


Hyperplanes

Definition

For $c \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ and $\delta \in \mathbb{R}$, the set $H = \{x \in \mathbb{R}^n \mid c^T x = \delta\}$ is called an **affine hyperplane**.

- $H_{\leq} := \{x \in \mathbb{R}^n \mid c^T x \leq \delta\}$ is a **(closed) half-space**

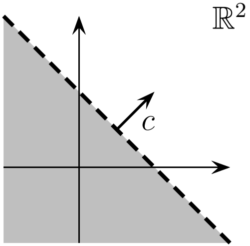


Hyperplanes

Definition

For $c \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ and $\delta \in \mathbb{R}$, the set $H = \{x \in \mathbb{R}^n \mid c^T x = \delta\}$ is called an **affine hyperplane**.

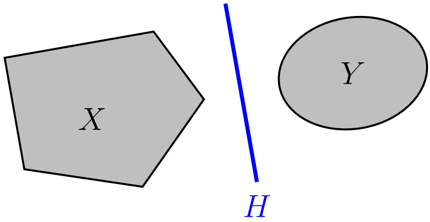
- $H_{\leq} := \{x \in \mathbb{R}^n \mid c^T x \leq \delta\}$ is a **(closed) half-space**
- $H_{<} := \{x \in \mathbb{R}^n \mid c^T x < \delta\}$ is a **(open) half-space**



Hyperplanes (2)

Definition

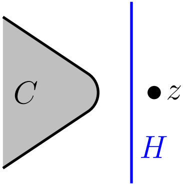
We say that a hyperplane H **separates** $X \subseteq \mathbb{R}^n$ and $Y \subseteq \mathbb{R}^n$ if X and Y lie in different open halfspaces of H.



Separating Hyperplane Theorem

Theorem

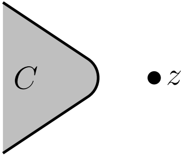
Let $C \subseteq \mathbb{R}^n$ be a closed convex set and $z \in \mathbb{R}^n \setminus C$. Then there is a hyperplane separating z and C



Separating Hyperplane Theorem

Theorem

Let $C \subseteq \mathbb{R}^n$ be a closed convex set and $z \in \mathbb{R}^n \setminus C$. Then there is a hyperplane separating z and C

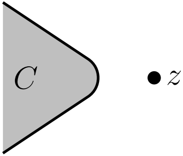


Proof. True if $C = \emptyset$. Suppose $C \neq \emptyset$.

Separating Hyperplane Theorem

Theorem

Let $C \subseteq \mathbb{R}^n$ be a closed convex set and $z \in \mathbb{R}^n \setminus C$. Then there is a hyperplane separating z and C

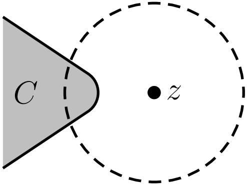


Proof. True if $C = \emptyset$. Suppose $C \neq \emptyset$.
Claim. The minimum $\min\{\|z - y\|_2 : y \in C\}$ is attained.

Separating Hyperplane Theorem

Theorem

Let $C \subseteq \mathbb{R}^n$ be a closed convex set and $z \in \mathbb{R}^n \setminus C$. Then there is a hyperplane separating z and C



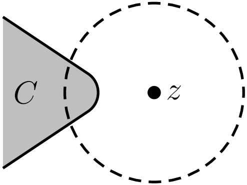
Proof. True if $C = \emptyset$. Suppose $C \neq \emptyset$.
Claim. The minimum $\min\{\|z - y\|_2 : y \in C\}$ is attained.

- Fix $r > 0$ with $B(z, r) \cap C \neq \emptyset$. Then $\min\{\|z - y\|_2 : y \in C\} = \min\{\|z - y\|_2 : y \in B(z, r) \cap C\}$.

Separating Hyperplane Theorem

Theorem

Let $C \subseteq \mathbb{R}^n$ be a closed convex set and $z \in \mathbb{R}^n \setminus C$. Then there is a hyperplane separating z and C



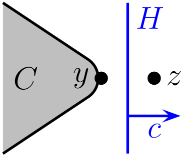
Proof. True if $C = \emptyset$. Suppose $C \neq \emptyset$.

Claim. The minimum $\min\{\|z-y\|_2 : y \in C\}$ is attained.

- Fix $r > 0$ with $B(z,r) \cap C \neq \emptyset$. Then $\min\{\|z-y\|_2 : y \in C\} = \min\{\|z-y\|_2 : y \in B(z,r) \cap C\}$.
- Moreover $B(z,r) \cap C$ is **compact** and the map $y \mapsto \|z-y\|_2$ is **continuous**. Claim follows. $\square$

Separating Hyperplane Theorem (2)

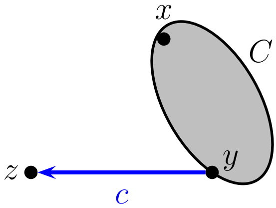
- Fix $y \in C$ minimizing $\|z - y\|_2$.
- Choose $H := \{x \in \mathbb{R}^n \mid c^T x = \delta\}$ with $c := z - y$ and $\delta := c^T(\frac{z+y}{2})$.



Claim. $c^T z > \delta$ and $c^T x < \delta \; \forall x \in C$

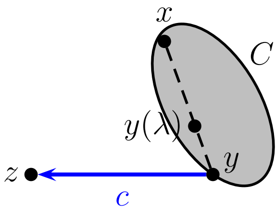
- We can verify that $c^T z = \delta + \frac{1}{2}\|c\|_2^2 > \delta$.
- Suppose for sake of contradiction that there is a $x \in C$ with $c^T x \geq \delta$. In particular $c^T x > c^T y$.

Separating Hyperplane Theorem (3)



Separating Hyperplane Theorem (3)

- Consider $y(\lambda) := (1-\lambda)y + \lambda x$ (recall $c = z - y$)

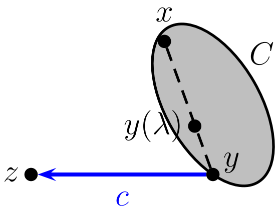


Separating Hyperplane Theorem (3)

- Consider $y(\lambda) := (1-\lambda)y + \lambda x$ (recall $c = z - y$)
- Then

$$\begin{aligned} &\|z - y(\lambda)\|_2^2 = \|c + \lambda(y-x)\|_2^2 \\ = \quad &\|c\|_2^2 + 2\lambda \underbrace{c^T(y-x)}_{<0} + \lambda^2 \|y-x\|_2^2 \overset{!}{<} \|c\|_2^2 = \|z-y\|_2^2 \end{aligned}$$

- Contradiction if we pick $\lambda > 0$ small enough. $\square$



Affine independence

Definition

Vectors $x_1, \ldots, x_m \in \mathbb{R}^n$ are **affinely independent** if

$$\left(\sum_{i=1}^{m} \lambda_i x_i = 0 \text{ and } \sum_{i=1}^{m} \lambda_i = 0\right) \Rightarrow \left(\lambda_1 = \ldots = \lambda_m = 0\right)$$

$\mathbb{R}^2$:

affinely indep. affinely dep.

Affine independence

Definition

Vectors $x_1, \ldots, x_m \in \mathbb{R}^n$ are **affinely independent** if

$$\left(\sum_{i=1}^{m} \lambda_i x_i = 0 \text{ and } \sum_{i=1}^{m} \lambda_i = 0 \right) \Rightarrow \left(\lambda_1 = \ldots = \lambda_m = 0 \right)$$

$\mathbb{R}^2$:

affinely indep. affinely dep.

Lemma

$x_1, \ldots, x_m \in \mathbb{R}^n$ affinely independent $\Leftrightarrow \binom{x_1}{1}, \ldots, \binom{x_m}{1}$ are linearly independent

Affine independence (2)

- **Fact 1.** At most $n + 1$ points in $\mathbb{R}^n$ are affinely independent

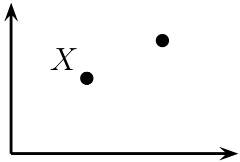
Affine independence (2)

- **Fact 1.** At most $n+1$ points in $\mathbb{R}^n$ are affinely independent
- **Fact 2.** Affine invariance is invariant under translation.

Affine independence (2)

- **Fact 1.** At most $n+1$ points in $\mathbb{R}^n$ are affinely independent
- **Fact 2.** Affine invariance is invariant under translation.
- We define

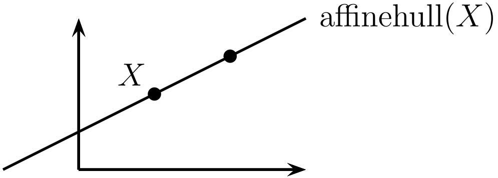
$$\text{affinehull}(X) := \Big\{ \sum_{i=1}^{t} \lambda_i x_i \mid x_1, \ldots, x_t \in X \text{ and } \sum_{i=1}^{t} \lambda_i = 1 \Big\}$$



Affine independence (2)

- **Fact 1.** At most $n+1$ points in $\mathbb{R}^n$ are affinely independent
- **Fact 2.** Affine invariance is invariant under translation.
- We define

$$\text{affinehull}(X) := \Big\{ \sum_{i=1}^{t} \lambda_i x_i \mid x_1, \ldots, x_t \in X \text{ and } \sum_{i=1}^{t} \lambda_i = 1 \Big\}$$

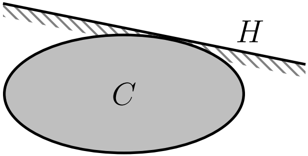


Polyhedra

Lemma

For any closed convex set $C \subseteq \mathbb{R}^n$ one has $C = \bigcap_{C \subseteq H_\leq} H_\leq$

Proof. Exercise.

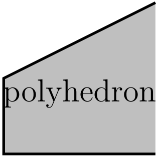


- Possibly an infinite number of halfspaces is needed.

Polyhedra (2)

Definition

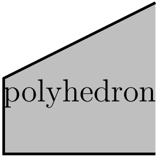
The intersection of a finite number of closed half-spaces is called a **polyhedron**.



Polyhedra (2)

Definition

The intersection of a finite number of closed half-spaces is called a **polyhedron**.

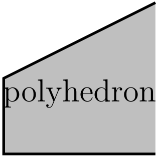


- **Fact.** Polyhedra are closed and convex

Polyhedra (2)

Definition

The intersection of a finite number of closed half-spaces is called a **polyhedron**.



- **Fact.** Polyhedra are closed and convex
- Each polyhedron P can be represented as

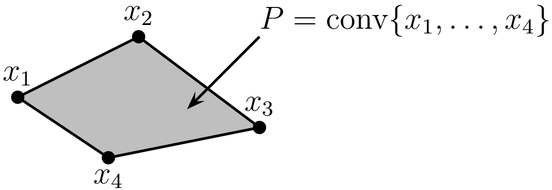
$$P = \{x \in \mathbb{R}^n \mid Ax \le b\} = \left\{ x \in \mathbb{R}^n \mid \begin{array}{rcl} A_1^T x & \le & b_1 \\ A_2^T x & \le & b_2 \\ & \vdots & \\ A_m^T x & \le & b_m \end{array} \right\}$$

for a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $b \in \mathbb{R}^m$ (A_i^T is the ith row of matrix A).

Polytopes and polyhedra (3)

Definition

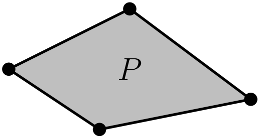
$P \subseteq \mathbb{R}^n$ is a **polytope** if $P = \mathrm{conv}\{x_1, \ldots, x_t\}$ for a finite number of points $x_1, \ldots, x_t \in \mathbb{R}^n$.



Polytopes and polyhedra (4)

Theorem

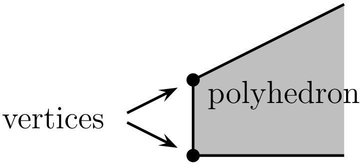
$P \subseteq \mathbb{R}^n$ is a polytope $\Leftrightarrow$ P is a bounded polyhedron.



Extreme points

Definition

Let $C \subseteq \mathbb{R}^n$ be a convex set. A point $z \in C$ is called **vertex / extreme point** if there are no $x, y \in C, 0 < \lambda < 1$ with $x \neq y$ so that $z = \lambda x + (1 - \lambda)y$



- Phrased differently: If $z \in C$ is the strict convex combination of two different points in C, then z is NOT a vertex.

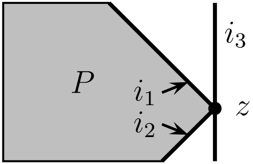
Lecture 4

Chapter 2 — Polytopes, polyhedra, Farkas' Lemma and Linear Programming — Part 2/3

Characterization of vertices

Lemma

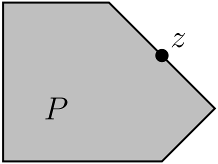
Suppose $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ *be a polyhedron with* $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ *and let* $z \in P$. *Let* A_z *be the submatrix of* A *consisting of those rows* i *s.t.* $A_i^T z = b_i$.
Then z *is a vertex of* $P \Leftrightarrow rank(A_z) = n$.



$$A_z = \begin{pmatrix} A_{i_1}^T \\ A_{i_2}^T \\ A_{i_3}^T \end{pmatrix}$$

Characterization of opt. LP sol. (2)

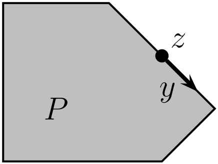
Claim I. $\mathrm{rank}(A_z) < n \Rightarrow z$ not a vertex



Characterization of opt. LP sol. (2)

Claim I. $\mathrm{rank}(A_z) < n \Rightarrow z$ not a vertex

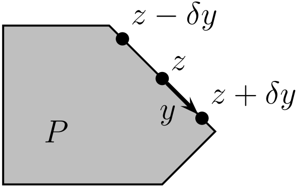
- There is a direction $y \in \ker(A_z) \setminus \{\mathbf{0}\}$ (meaning $A_z y = \mathbf{0}$).



Characterization of opt. LP sol. (2)

Claim I. $\mathrm{rank}(A_z) < n \Rightarrow z$ not a vertex

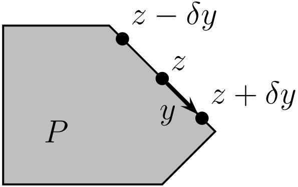
- There is a direction $y \in \ker(A_z) \setminus \{\mathbf{0}\}$ (meaning $A_z y = \mathbf{0}$).
- For some $\delta > 0$, $A_i^T(z + \delta y) \leq b_i$ and $A_i^T(z - \delta y) \leq b_i$ for any non-tight constraint



Characterization of opt. LP sol. (2)

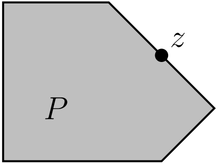
Claim I. $\text{rank}(A_z) < n \Rightarrow z$ not a vertex

- There is a direction $y \in \ker(A_z) \setminus \{\mathbf{0}\}$ (meaning $A_z y = \mathbf{0}$).
- For some $\delta > 0$, $A_i^T(z + \delta y) \leq b_i$ and $A_i^T(z - \delta y) \leq b_i$ for any non-tight constraint
- Then $z + \delta y, z - \delta y \in P \Rightarrow z$ is not a vertex $\square$



Characterization of opt. LP sol. (3)

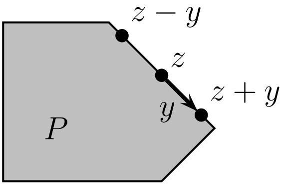
Claim II. z not a vertex $\Rightarrow \text{rank}(A_z) < n$



Characterization of opt. LP sol. (3)

Claim II. z not a vertex $\Rightarrow \operatorname{rank}(A_z) < n$

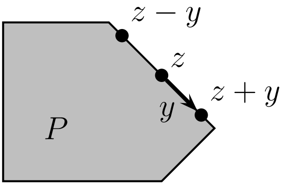
- If not, then by definition (and convexity) $z + y \in P$ and $z - y \in P$ for some $y \in \mathbb{R}^n \setminus \{\mathbf{0}\}$.



Characterization of opt. LP sol. (3)

Claim II. z not a vertex $\Rightarrow \text{rank}(A_z) < n$

- If not, then by definition (and convexity) $z + y \in P$ and $z - y \in P$ for some $y \in \mathbb{R}^n \setminus \{\mathbf{0}\}$.
- Consider index i with $A_i^T z = b_i$. Then $(A_i^T(z+y) \leq b_i \quad \& \quad A_i^T(z-y) \leq b_i) \Rightarrow A_i^T y = 0$
- Hence $y \in \ker(A_z)$ and so $\text{rank}(A_z) < n$.



Number of extreme points

Corollary

A polyhedron has finitely many vertices.

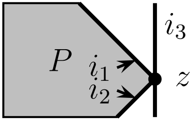
Number of extreme points

Corollary

A polyhedron has finitely many vertices.

Proof.

- Consider $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ with $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$
- Each vertex z is the **unique** solution to the linear system $A_z x = b_z$



$$A_z = \begin{pmatrix} A_{i_1}^T \\ A_{i_2}^T \\ A_{i_3}^T \end{pmatrix}$$

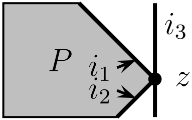
Number of extreme points

Corollary

A polyhedron has finitely many vertices.

Proof.

- Consider $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ with $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$
- Each vertex z is the **unique** solution to the linear system $A_z x = b_z$



$$A_z = \begin{pmatrix} A_{i_1}^T \\ A_{i_2}^T \\ A_{i_3}^T \end{pmatrix}$$

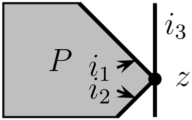
- There are at most 2^m many submatrices (formed by taking a subset of the rows). $\square$

Number of extreme points

> **Corollary**
>
> *A polyhedron has finitely many vertices.*

Proof.

- Consider $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ with $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$
- Each vertex z is the **unique** solution to the linear system $A_z x = b_z$



$$A_z = \begin{pmatrix} A_{i_1}^T \\ A_{i_2}^T \\ A_{i_3}^T \end{pmatrix}$$

- There are at most 2^m many submatrices (formed by taking a subset of the rows). $\square$

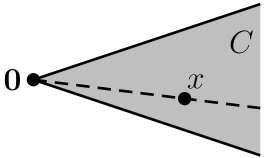
- Simple improvement: at most $\binom{m}{n}$ vertices
- McMullen (1970): Number of vertices is $\leq O(m^{\lfloor n/2 \rfloor})$.

Convex cones

Definition

A set $C \subseteq \mathbb{R}^n$ is a **convex cone** if

$$\lambda x + \mu y \in C \quad \forall x, y \in C \;\; \forall \lambda, \mu \geq 0$$

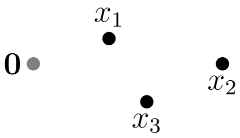


Convex cones (2)

Definition

For $X \subseteq \mathbb{R}^n$ we define

$$\begin{aligned}
\mathrm{cone}(X) \quad &:= \quad \text{unique minimal convex cone containing } X \\
&= \quad \Big\{ \underbrace{\sum_{i=1}^{t} \lambda_i x_i}_{\substack{\text{conical combination} \\ \text{of } x_1,\ldots,x_t}} \;\Big|\; x_1,\ldots,x_t \in X;\ \lambda_1,\ldots,\lambda_t \geq 0 \Big\}
\end{aligned}$$

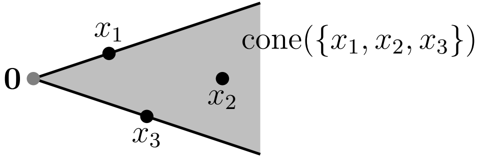


Convex cones (2)

Definition

For $X \subseteq \mathbb{R}^n$ we define

$$\begin{aligned} \operatorname{cone}(X) &:= \text{unique minimal convex cone containing } X \\ &= \Big\{ \underbrace{\sum_{i=1}^{t} \lambda_i x_i}_{\substack{\text{conical combination} \\ \text{of } x_1,\ldots,x_t}} \mid x_1,\ldots,x_t \in X;\ \lambda_1,\ldots,\lambda_t \geq 0 \Big\} \end{aligned}$$



Farkas Lemma

Lemma (Farkas' Lemma 1902)

One has $\big(\exists x \geq \mathbf{0} : Ax = b\big) \quad \dot\vee \quad \big(\exists y : y^T A \geq \mathbf{0} \text{ and } y^T b < 0\big)$.

Claim I. Impossible that both systems have a solution.

- Suppose for sake of contradiction that there are solutions x, y to both systems.
- Then

$$0 \leq \underbrace{(y^T A)}_{\geq \mathbf{0}} \underbrace{x}_{\geq \mathbf{0}} = y^T \underbrace{(Ax)}_{=b} = y^T b < 0$$

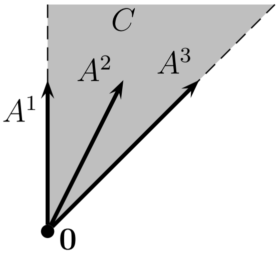
Farkas Lemma (2)

Claim II. Assume there is no $x \geq \mathbf{0}$ with $Ax = b$. Then there is a $y^T A \geq \mathbf{0}$ and $y^T b < 0$.

Farkas Lemma (2)

Claim II. Assume there is no $x \geq \mathbf{0}$ with $Ax = b$. Then there is a $y^T A \geq \mathbf{0}$ and $y^T b < 0$.

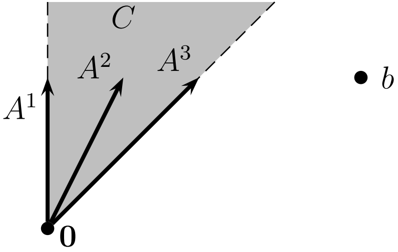
- Let $A_1, \ldots, A_n$ be the columns of A.
- Consider the cone $C := \text{cone}(\{A^1, \ldots, A^n\})$
- C is convex (clear) and closed (exercise).



Farkas Lemma (2)

Claim II. Assume there is no $x \geq \mathbf{0}$ with $Ax = b$. Then there is a $y^T A \geq \mathbf{0}$ and $y^T b < 0$.

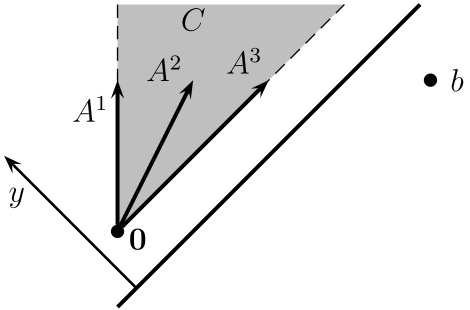
- Let $A_1, \ldots, A_n$ be the columns of A.
- Consider the cone $C := \text{cone}(\{A^1, \ldots, A^n\})$
- C is convex (clear) and closed (exercise).
- By assumption we know that $b \notin C$.



Farkas Lemma (2)

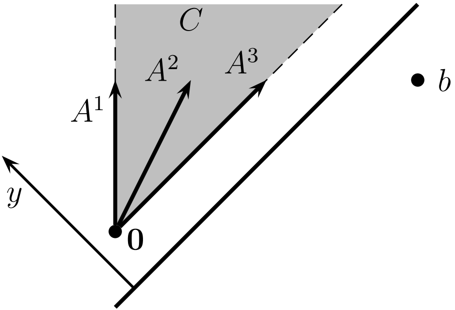
Claim II. Assume there is no $x \geq \mathbf{0}$ with $Ax = b$. Then there is a $y^T A \geq \mathbf{0}$ and $y^T b < 0$.

- Let $A_1, \ldots, A_n$ be the columns of A.
- Consider the cone $C := \text{cone}(\{A^1, \ldots, A^n\})$
- C is convex (clear) and closed (exercise).
- By assumption we know that $b \notin C$.



- Then there is a hyperplane $y^T c = \gamma$ separating C and b.

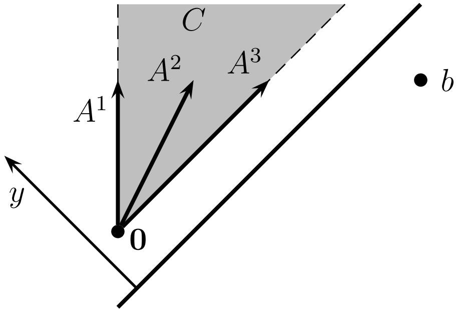
Farkas Lemma (3)



- We have

$$\forall c \in C : \quad y^T c > \gamma > y^T b$$

Farkas Lemma (3)

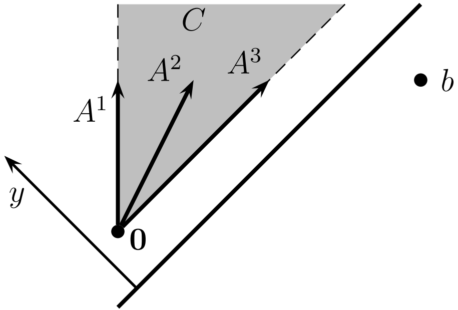


- We have

$$\forall c \in C: \quad y^T c > \gamma > y^T b$$

- As $\mathbf{0} \in C$ we must have $\gamma < 0$.

Farkas Lemma (3)



- We have
$$\forall c \in C: \quad y^T c > \gamma > y^T b$$
- As $\mathbf{0} \in C$ we must have $\gamma < 0$.
- For $x_i \geq 0$, $x_i A^i \in C$ and so $x_i \cdot y^T A^i > \gamma$.
- Then $y^T A^i \geq 0$ for each $i \in [n]$.
- More compactly $y^T A \geq \mathbf{0}$. $\square$

Farkas variants

Theorem

Let $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. Then

(I) $\exists x : Ax \leq b \quad \Leftrightarrow \quad \nexists y \geq \mathbf{0} : y^T A = \mathbf{0},\ y^T b < 0$

(II) $\exists x \geq \mathbf{0} : Ax = b \quad \Leftrightarrow \quad \nexists y : y^T A \geq \mathbf{0},\ y^T b < 0$

(III) $\exists x \geq \mathbf{0} : Ax \leq b \quad \Leftrightarrow \quad \nexists y \geq \mathbf{0} : y^T A \geq \mathbf{0},\ y^T b < 0$

We now prove $(II) \Rightarrow (I)$.

Farkas variants

$$\text{(I)} \quad \exists x : Ax \leq b \quad \Leftrightarrow \quad \nexists y \geq \mathbf{0} : y^T A = \mathbf{0}, y^T b < 0$$

$$\text{(II)} \quad \exists x \geq \mathbf{0} : Ax = b \quad \Leftrightarrow \quad \nexists y : y^T A \geq \mathbf{0}, y^T b < 0$$

One has

$$\exists x : Ax \leq b$$

Farkas variants

(I) $\exists x : Ax \leq b \quad \Leftrightarrow \quad \nexists y \geq \mathbf{0} : y^T A = \mathbf{0}, y^T b < 0$

(II) $\exists x \geq \mathbf{0} : Ax = b \quad \Leftrightarrow \quad \nexists y : y^T A \geq \mathbf{0}, y^T b < 0$

One has

$$\begin{aligned} & \exists x : Ax \leq b \\ \Leftrightarrow \quad & \exists x, s \geq \mathbf{0} : Ax + Is = b \end{aligned}$$

Farkas variants

$$
\begin{array}{rll}
\text{(I)} & \exists x : Ax \le b & \Leftrightarrow \quad \nexists y \ge \mathbf{0} : y^T A = \mathbf{0}, y^T b < 0 \\
\text{(II)} & \exists x \ge \mathbf{0} : Ax = b & \Leftrightarrow \quad \nexists y : y^T A \ge \mathbf{0}, y^T b < 0
\end{array}
$$

One has

$$
\begin{array}{rl}
& \exists x : Ax \le b \\
\Leftrightarrow & \exists x, s \ge \mathbf{0} : Ax + Is = b \\
\Leftrightarrow & \exists x' \ge \mathbf{0}, x'' \ge 0, s \ge \mathbf{0} : Ax' - Ax'' + Is = b
\end{array}
$$

Farkas variants

$$\text{(I)} \quad \exists x : Ax \le b \quad \Leftrightarrow \quad \nexists y \ge \mathbf{0} : y^T A = \mathbf{0}, y^T b < 0$$

$$\text{(II)} \quad \exists x \ge \mathbf{0} : Ax = b \quad \Leftrightarrow \quad \nexists y : y^T A \ge \mathbf{0}, y^T b < 0$$

One has

$$\begin{aligned}
& \exists x : Ax \le b \\
\Leftrightarrow \quad & \exists x, s \ge \mathbf{0} : Ax + Is = b \\
\Leftrightarrow \quad & \exists x' \ge \mathbf{0}, x'' \ge 0, s \ge \mathbf{0} : Ax' - Ax'' + Is = b \\
\Leftrightarrow \quad & \exists \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} \ge \mathbf{0} : [A, -A, I] \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} = b
\end{aligned}$$

Farkas variants

$$
\begin{array}{rlcl}
\text{(I)} & \exists x : Ax \leq b & \Leftrightarrow & \nexists y \geq \mathbf{0} : y^T A = \mathbf{0}, y^T b < 0 \\
\text{(II)} & \exists x \geq \mathbf{0} : Ax = b & \Leftrightarrow & \nexists y : y^T A \geq \mathbf{0}, y^T b < 0
\end{array}
$$

One has

$$
\begin{aligned}
& \exists x : Ax \leq b \\
\Leftrightarrow \quad & \exists x, s \geq \mathbf{0} : Ax + Is = b \\
\Leftrightarrow \quad & \exists x' \geq \mathbf{0}, x'' \geq 0, s \geq \mathbf{0} : Ax' - Ax'' + Is = b \\
\Leftrightarrow \quad & \exists \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} \geq \mathbf{0} : [A, -A, I] \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} = b \\
\overset{(II)}{\Leftrightarrow} \quad & \nexists y : y^T [A, -A, I] \geq \mathbf{0},\ y^T b < 0
\end{aligned}
$$

Farkas variants

$$\text{(I)}\quad \exists x : Ax \leq b \quad \Leftrightarrow \quad \nexists y \geq \mathbf{0} : y^T A = \mathbf{0}, y^T b < 0$$

$$\text{(II)}\quad \exists x \geq \mathbf{0} : Ax = b \quad \Leftrightarrow \quad \nexists y : y^T A \geq \mathbf{0}, y^T b < 0$$

One has

$$\begin{aligned}
& \exists x : Ax \leq b \\
\Leftrightarrow \quad & \exists x, s \geq \mathbf{0} : Ax + Is = b \\
\Leftrightarrow \quad & \exists x' \geq \mathbf{0}, x'' \geq 0, s \geq \mathbf{0} : Ax' - Ax'' + Is = b \\
\Leftrightarrow \quad & \exists \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} \geq \mathbf{0} : [A, -A, I] \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} = b \\
\overset{(II)}{\Leftrightarrow} \quad & \nexists y : y^T [A, -A, I] \geq \mathbf{0},\ y^T b < 0 \\
\Leftrightarrow \quad & \nexists y : y^T A \geq \mathbf{0},\ y^T (-A) \geq \mathbf{0},\ y^T I \geq \mathbf{0},\ y^T b < 0
\end{aligned}$$

Farkas variants

$$\text{(I)}\quad \exists x : Ax \leq b \quad \Leftrightarrow \quad \nexists y \geq \mathbf{0} : y^T A = \mathbf{0}, y^T b < 0$$

$$\text{(II)}\quad \exists x \geq \mathbf{0} : Ax = b \quad \Leftrightarrow \quad \nexists y : y^T A \geq \mathbf{0}, y^T b < 0$$

One has

$$\begin{aligned}
& \exists x : Ax \leq b \\
\Leftrightarrow \quad & \exists x, s \geq \mathbf{0} : Ax + Is = b \\
\Leftrightarrow \quad & \exists x' \geq \mathbf{0}, x'' \geq 0, s \geq \mathbf{0} : Ax' - Ax'' + Is = b \\
\Leftrightarrow \quad & \exists \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} \geq \mathbf{0} : [A, -A, I] \begin{pmatrix} x' \\ x'' \\ s \end{pmatrix} = b \\
\overset{(II)}{\Leftrightarrow} \quad & \nexists y : y^T [A, -A, I] \geq \mathbf{0},\ y^T b < 0 \\
\Leftrightarrow \quad & \nexists y : y^T A \geq \mathbf{0},\ y^T (-A) \geq \mathbf{0},\ y^T I \geq \mathbf{0},\ y^T b < 0 \\
\Leftrightarrow \quad & \nexists y : y^T A = \mathbf{0},\ y \geq \mathbf{0},\ y^T b < 0 \qquad \square
\end{aligned}$$

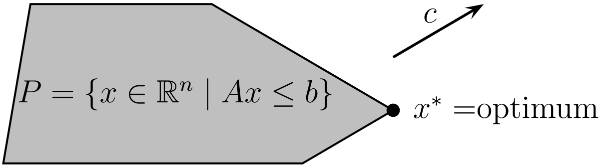
Lecture 5

Chapter 2 — Polytopes, polyhedra, Farkas' Lemma and Linear Programming — Part 3/3

Linear programs

Definition

The optimization problem $\max\{c^T x \mid Ax \leq b\}$ is called **linear program** where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.



Optimum for LPs is attained

Lemma

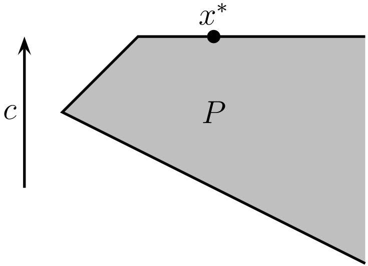
Let $P \subseteq \mathbb{R}^n$ *be a polyhedron and* $c \in \mathbb{R}^n$. *If* $\sup\{c^T x \mid x \in P\} < \infty$ *then* $\max\{c^T x \mid x \in P\}$ *is attained.*

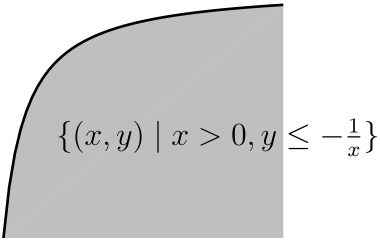
Optimum for LPs is attained

Lemma

Let $P \subseteq \mathbb{R}^n$ be a polyhedron and $c \in \mathbb{R}^n$. If $\sup\{c^T x \mid x \in P\} < \infty$ *then* $\max\{c^T x \mid x \in P\}$ *is attained.*

- Trivial for polytopes.
- Not true for all closed convex sets!





Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$\nexists x \in P \text{ with } c^T x \geq \delta$$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$
\begin{aligned}
& \nexists x \in P \text{ with } c^T x \geq \delta \\
\Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix}
\end{aligned}
$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$
\begin{aligned}
& \nexists x \in P \text{ with } c^T x \geq \delta \\
\Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix} \\
\Leftrightarrow \quad & \exists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A \\ -c^T \end{pmatrix} = \mathbf{0}, (y^T, \lambda) \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0
\end{aligned}
$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$
\begin{aligned}
& \nexists x \in P \text{ with } c^T x \geq \delta \\
\Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix} \\
\Leftrightarrow \quad & \exists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A \\ -c^T \end{pmatrix} = \mathbf{0}, (y^T, \lambda) \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0 \\
\Leftrightarrow \quad & \exists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = \lambda c^T, y^T b < \lambda \delta
\end{aligned}
$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$
\begin{aligned}
& \nexists x \in P \text{ with } c^T x \geq \delta \\
\Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix} \\
\Leftrightarrow \quad & \exists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A \\ -c^T \end{pmatrix} = \mathbf{0}, (y^T, \lambda) \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0 \\
\Leftrightarrow \quad & \exists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = \lambda c^T, y^T b < \lambda \delta
\end{aligned}
$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$
- Then

$$\sup\{y^T A x \mid x \in P\}$$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$
\begin{aligned}
& \nexists x \in P \text{ with } c^T x \geq \delta \\
\Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix} \\
\Leftrightarrow \quad & \exists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A \\ -c^T \end{pmatrix} = \mathbf{0}, (y^T, \lambda) \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0 \\
\Leftrightarrow \quad & \exists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = \lambda c^T, y^T b < \lambda \delta
\end{aligned}
$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$
- Then

$$\sup\{y^T A x \mid x \in P\} \leq y^T b$$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$\begin{aligned} & \nexists x \in P \text{ with } c^T x \geq \delta \\ \Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix} \\ \Leftrightarrow \quad & \exists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A \\ -c^T \end{pmatrix} = \mathbf{0}, (y^T, \lambda) \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0 \\ \Leftrightarrow \quad & \exists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = \lambda c^T, y^T b < \lambda \delta \end{aligned}$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$
- Then

$$\sup\{y^T A x \mid x \in P\} \leq y^T b < \lambda \delta$$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$
\begin{aligned}
& \nexists x \in P \text{ with } c^T x \geq \delta \\
\Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix} \\
\Leftrightarrow \quad & \exists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A \\ -c^T \end{pmatrix} = \mathbf{0}, (y^T, \lambda) \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0 \\
\Leftrightarrow \quad & \exists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = \lambda c^T, y^T b < \lambda \delta
\end{aligned}
$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$
- Then

$$\lambda \cdot \sup\{c^T x \mid x \in P\} = \sup\{y^T A x \mid x \in P\} \leq y^T b < \lambda \delta$$

Optimum for LPs is attained (2)

- Set $\delta := \sup\{c^T x \mid x \in P\} < \infty$.
- Suppose for sake of contradiction:

$$
\begin{aligned}
& \nexists x \in P \text{ with } c^T x \geq \delta \\
\Leftrightarrow \quad & \nexists x : \begin{pmatrix} A \\ -c^T \end{pmatrix} x \leq \begin{pmatrix} b \\ -\delta \end{pmatrix} \\
\Leftrightarrow \quad & \exists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A \\ -c^T \end{pmatrix} = \mathbf{0}, (y^T, \lambda) \begin{pmatrix} b \\ -\delta \end{pmatrix} < 0 \\
\Leftrightarrow \quad & \exists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = \lambda c^T, y^T b < \lambda \delta
\end{aligned}
$$

- **Farkas I:** $(\exists \tilde{x} : \tilde{A}\tilde{x} \leq \tilde{b}) \dot{\vee} (\exists \tilde{y} \geq \mathbf{0}, \tilde{y}^T \tilde{A} = \mathbf{0}, \tilde{y}^T \tilde{b} < 0)$
- Then

$$\lambda\delta = \lambda \cdot \sup\{c^T x \mid x \in P\} = \sup\{y^T A x \mid x \in P\} \leq y^T b < \lambda\delta$$

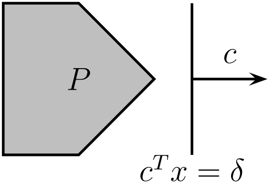
- Contradiction! $\square$

Valid inequalities

Lemma

Let $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ *and assume* $P := \{x \in \mathbb{R}^n \mid Ax \leq b\}$ *is non-empty. Then*

$$\left(c^T x \leq \delta \;\; \forall x \in P\right) \Leftrightarrow \left(\exists y \geq \mathbf{0} : y^T A = c^T \text{ and } c^T b \leq \delta\right)$$

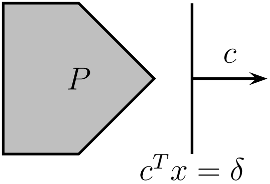


Valid inequalities

Lemma

Let $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ and assume $P := \{x \in \mathbb{R}^n \mid Ax \le b\}$ is non-empty. Then

$$\left(c^T x \le \delta \;\; \forall x \in P\right) \Leftrightarrow \left(\exists y \ge \mathbf{0} : y^T A = c^T \text{ and } c^T b \le \delta\right)$$



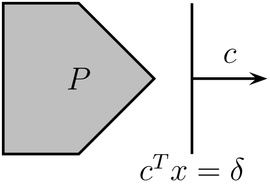
Claim I. Suppose there is a $y \ge \mathbf{0} : y^T A = c^T$ and $y^T b \le \delta$. Then for $x \in P$ one has $c^T x \le \delta$.

Valid inequalities

Lemma

Let $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ and assume $P := \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is non-empty. Then

$$\left(c^T x \leq \delta \ \ \forall x \in P\right) \Leftrightarrow \left(\exists y \geq \mathbf{0} : y^T A = c^T \text{ and } c^T b \leq \delta\right)$$



Claim I. Suppose there is a $y \geq \mathbf{0} : y^T A = c^T$ and $y^T b \leq \delta$. Then for $x \in P$ one has $c^T x \leq \delta$.
Proof. We verify that

$$c^T x = y^T A x \overset{y \geq \mathbf{0}, Ax \leq b}{\leq} y^T b \leq \delta$$

Valid inequalities (2)

Claim II. $(c^T x \leq \delta \; \forall x \in P) \Rightarrow (\exists y \geq \mathbf{0} : y^T A = c^T, c^T b \leq \delta)$.

Valid inequalities (2)

Claim II. $(c^T x \leq \delta \; \forall x \in P) \Rightarrow (\exists y \geq \mathbf{0} : y^T A = c^T, c^T b \leq \delta)$.

- Suppose such a y does not exist.

$$\nexists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = c^T,\; y^T b + \lambda = \delta$$

Valid inequalities (2)

Claim II. $(c^T x \leq \delta \; \forall x \in P) \Rightarrow (\exists y \geq \mathbf{0} : y^T A = c^T, c^T b \leq \delta)$.

- Suppose such a y does not exist.

$$\nexists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = c^T,\; y^T b + \lambda = \delta$$

$$\nexists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} = (c^T, \delta)$$

- **Farkas:** $(\exists x \geq \mathbf{0} : Ax = b) \dot{\vee} (\exists y : y^T A \geq \mathbf{0} \;\&\; y^T b < 0)$

Valid inequalities (2)

Claim II. $(c^T x \leq \delta \; \forall x \in P) \Rightarrow (\exists y \geq \mathbf{0} : y^T A = c^T, c^T b \leq \delta)$.

- Suppose such a y does not exist.

$$\nexists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = c^T, \; y^T b + \lambda = \delta$$

$$\nexists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} = (c^T, \delta)$$

$$\Leftrightarrow \quad \exists \begin{pmatrix} z \\ u \end{pmatrix} : \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} z \\ u \end{pmatrix} \geq \begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix} \& (c^T, \delta) \begin{pmatrix} z \\ u \end{pmatrix} < 0$$

- **Farkas:** $(\exists x \geq \mathbf{0} : Ax = b) \dot{\vee} (\exists y : y^T A \geq \mathbf{0} \; \& \; y^T b < 0)$

Valid inequalities (2)

Claim II. $(c^T x \leq \delta \; \forall x \in P) \Rightarrow (\exists y \geq \mathbf{0} : y^T A = c^T, c^T b \leq \delta)$.

- Suppose such a y does not exist.

$$\begin{aligned}
& \nexists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = c^T,\; y^T b + \lambda = \delta \\
& \nexists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} = (c^T, \delta) \\
\Leftrightarrow \quad & \exists \begin{pmatrix} z \\ u \end{pmatrix} : \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} z \\ u \end{pmatrix} \geq \begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix} \& (c^T, \delta) \begin{pmatrix} z \\ u \end{pmatrix} < 0 \\
\Leftrightarrow \quad & \exists z, u : Az + bu \geq \mathbf{0}, u \geq 0, c^T z + \delta u < 0
\end{aligned}$$

- **Farkas:** $(\exists x \geq \mathbf{0} : Ax = b) \dot{\vee} (\exists y : y^T A \geq \mathbf{0} \;\&\; y^T b < 0)$

Valid inequalities (2)

Claim II. $(c^T x \leq \delta \ \forall x \in P) \Rightarrow (\exists y \geq \mathbf{0} : y^T A = c^T, c^T b \leq \delta)$.

- Suppose such a y does not exist.

$$\begin{aligned}
&& \nexists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = c^T,\ y^T b + \lambda = \delta \\
&& \nexists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} = (c^T, \delta) \\
\Leftrightarrow && \exists \begin{pmatrix} z \\ u \end{pmatrix} : \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} z \\ u \end{pmatrix} \geq \begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix} \& (c^T, \delta) \begin{pmatrix} z \\ u \end{pmatrix} < 0 \\
\Leftrightarrow && \exists z, u : Az + bu \geq \mathbf{0}, u \geq 0, c^T z + \delta u < 0 \\
\overset{\text{flip sign}}{\Leftrightarrow} && \exists z, u \geq 0 : Az \leq bu, c^T z > \delta u
\end{aligned}$$

- **Farkas:** $(\exists x \geq \mathbf{0} : Ax = b) \dot{\vee} (\exists y : y^T A \geq \mathbf{0} \ \& \ y^T b < 0)$
- **Case $u = 0$:** Then $Az \leq \mathbf{0}$, $c^T z > 0$. Fix $x_0 \in P$. Then for τ large enough: $A(x_0 + \tau z) \leq b$, $c^T (x_0 + \tau z) > \delta$. Contradiction!

Valid inequalities (2)

Claim II. $(c^T x \leq \delta \; \forall x \in P) \Rightarrow (\exists y \geq \mathbf{0} : y^T A = c^T, c^T b \leq \delta)$.

- Suppose such a y does not exist.

$$
\begin{aligned}
& \nexists y \geq \mathbf{0}, \lambda \geq 0 : y^T A = c^T,\; y^T b + \lambda = \delta \\
& \nexists (y, \lambda) \geq \mathbf{0} : (y^T, \lambda) \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} = (c^T, \delta) \\
\Leftrightarrow \quad & \exists \begin{pmatrix} z \\ u \end{pmatrix} : \begin{pmatrix} A & b \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} z \\ u \end{pmatrix} \geq \begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix} \& (c^T, \delta) \begin{pmatrix} z \\ u \end{pmatrix} < 0 \\
\Leftrightarrow \quad & \exists z, u : Az + bu \geq \mathbf{0}, u \geq 0, c^T z + \delta u < 0 \\
\overset{\text{flip sign}}{\Leftrightarrow} \quad & \exists z, u \geq 0 : Az \leq bu, c^T z > \delta u
\end{aligned}
$$

- **Farkas:** $(\exists x \geq \mathbf{0} : Ax = b) \dot{\vee} (\exists y : y^T A \geq \mathbf{0} \;\&\; y^T b < 0)$
- **Case $u = 0$:** Then $Az \leq \mathbf{0}$, $c^T z > 0$. Fix $x_0 \in P$. Then for τ large enough: $A(x_0 + \tau z) \leq b$, $c^T(x_0 + \tau z) > \delta$. Contradiction!
- **Case $u > 0$:** Scale until $u = 1$. Then $Az \leq b$, $c^T z > \delta$. Contradiction!

Duality Theorem for Linear programming

Theorem

Let $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$. Then

$$\underbrace{\max\{c^T x \mid Ax \leq b\}}_{\textit{primal LP}} = \underbrace{\min\{y^T b \mid y^T A = c^T,\ y \geq 0\}}_{\textit{dual LP}}$$

provided both LPs are feasible.

Duality Theorem for Linear programming

Theorem

Let $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$. Then

$$\underbrace{\max\{c^T x \mid Ax \leq b\}}_{\text{primal LP}} = \underbrace{\min\{y^T b \mid y^T A = c^T,\ y \geq 0\}}_{\text{dual LP}}$$

provided both LPs are feasible.

Weak duality:

- Suppose x and y are feasible solutions. Then

$$c^T x = (y^T A)x = y^T(Ax) \overset{y\geq \mathbf{0}, Ax\leq b}{\leq} y^T b$$

Duality Theorem for Linear programming (2)

Strong duality:

- Set $\delta := \sup\{c^T x \mid Ax \leq b\}$

Duality Theorem for Linear programming (2)

Strong duality:

- Set $\delta := \sup\{c^T x \mid Ax \leq b\}$
- If $\delta = \infty$, then dual is infeasible (by weak duality)
- Suppose $\delta < \infty$ (i.e. primal is bounded).

Duality Theorem for Linear programming (2)

Strong duality:

- Set $\delta := \sup\{c^T x \mid Ax \leq b\}$
- If $\delta = \infty$, then dual is infeasible (by weak duality)
- Suppose $\delta < \infty$ (i.e. primal is bounded).
- Set $P := \{x \in \mathbb{R}^n \mid Ax \leq b\}$. Then $c^T x \leq \delta$ is feasible inequality for P.

Duality Theorem for Linear programming (2)

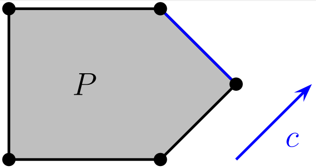
Strong duality:

- Set $\delta := \sup\{c^T x \mid Ax \leq b\}$
- If $\delta = \infty$, then dual is infeasible (by weak duality)
- Suppose $\delta < \infty$ (i.e. primal is bounded).
- Set $P := \{x \in \mathbb{R}^n \mid Ax \leq b\}$. Then $c^T x \leq \delta$ is feasible inequality for P.
- Then by last Cor. $\exists y \geq \mathbf{0} : y^T A = c^T$ and $y^T b \leq \delta$.
- This is a solution for dual with objective value δ! $\square$

Optimum LP sol for polytopes

Lemma

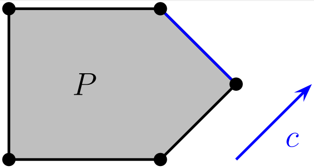
Let $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ be a polytope. Then $\max\{c^T x \mid x \in P\}$ is attained at a vertex of P.



Optimum LP sol for polytopes

Lemma

Let $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ *be a polytope. Then* $\max\{c^T x \mid x \in P\}$ *is attained at a vertex of* P.



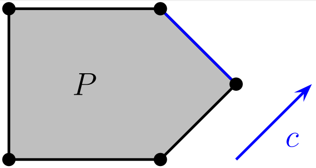
Proof:

- Let $x \in P$ and $c \in \mathbb{R}^n$. Let $P = \mathrm{conv}\{v_1, \ldots, v_m\}$ where v_i is vertex of P.

Optimum LP sol for polytopes

Lemma

Let $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ be a polytope. Then $\max\{c^T x \mid x \in P\}$ is attained at a vertex of P.

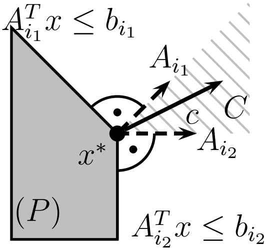


Proof:

- Let $x \in P$ and $c \in \mathbb{R}^n$. Let $P = \mathrm{conv}\{v_1, \ldots, v_m\}$ where v_i is vertex of P.
- Then

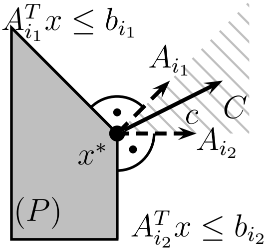
$$\begin{aligned} \max\{c^T x \mid x \in P\} &= \max\Big\{\sum_{i=1}^m \lambda_i \cdot c^T v_i \mid \sum_{i=1}^m \lambda_i = 1, \lambda_i \geq 0\Big\} \\ &= \max\{c^T v_1, \ldots, c^T v_m\} \end{aligned}$$

Geometry of LPs



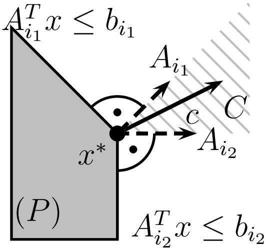
- Let x^* be **optimum solution** to $\max\{c^T x \mid Ax \leq b\}$.

Geometry of LPs



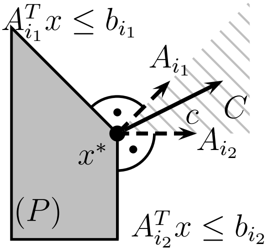
- Let x^* be **optimum solution** to $\max\{c^T x \mid Ax \leq b\}$.
- $I := \{i \mid A_i^T x^* = b_i\}$ be the **tight** inequalities
- Consider $C := \{\sum_{i \in I} A_i y_i \mid y_i \geq 0 \; \forall i \in I\}$

Geometry of LPs



- Let x^* be **optimum solution** to $\max\{c^T x \mid Ax \leq b\}$.
- $I := \{i \mid A_i^T x^* = b_i\}$ be the **tight** inequalities
- Consider $C := \{\sum_{i \in I} A_i y_i \mid y_i \geq 0 \; \forall i \in I\}$
- Then $c \in C$ otherwise x^* not optimal
- I.e. $\exists y \geq \mathbf{0}$ with $y^T A = c^T$ and $y_i = 0 \; \forall i \notin I$.

Geometry of LPs



- Let x^* be **optimum solution** to $\max\{c^T x \mid Ax \leq b\}$.
- $I := \{i \mid A_i^T x^* = b_i\}$ be the **tight** inequalities
- Consider $C := \{\sum_{i \in I} A_i y_i \mid y_i \geq 0 \ \forall i \in I\}$
- Then $c \in C$ otherwise x^* not optimal
- I.e. $\exists y \geq \mathbf{0}$ with $y^T A = c^T$ and $y_i = 0 \ \forall i \notin I$.
- We claim the **duality gap** is 0:

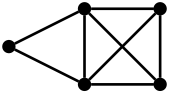
$$y^T b - c^T x^* \;=\; y^T b - \underbrace{y^T A}_{=c^T} x^* = \sum_{i=1}^{m} \underbrace{y_i}_{=0 \text{ if } i \notin I} \cdot \underbrace{(b_i - A_i^T x^*)}_{=0 \text{ if } i \in I} = 0$$

Lecture 6

Chapter 3 — Matchings and Covers in Bipartite Graphs — Part 1/2

Stable sets and vertex covers

Let $G = (V, E)$ be an undirected graph.

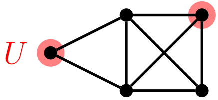


Stable sets and vertex covers

Let $G = (V, E)$ be an undirected graph.

Definition

$U \subseteq V$ is a **stable set / independent set** if for all $i, j \in U$ one has $\{i, j\} \notin E$.

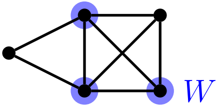


Stable sets and vertex covers

Let $G = (V, E)$ be an undirected graph.

Definition

$U \subseteq V$ is a **stable set / independent set** if for all $i, j \in U$ one has $\{i, j\} \notin E$.



Definition

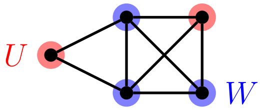
$W \subseteq V$ is a **vertex cover** if $e \cap W \neq \emptyset$ for all $e \in E$

Stable sets and vertex covers

Let $G = (V, E)$ be an undirected graph.

Definition

$U \subseteq V$ is a **stable set / independent set** if for all $i, j \in U$ one has $\{i, j\} \notin E$.



Definition

$W \subseteq V$ is a **vertex cover** if $e \cap W \neq \emptyset$ for all $e \in E$

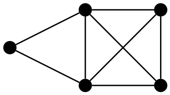
Lemma

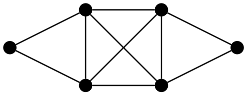
U is stable set in $G \Leftrightarrow V \setminus U$ is vertex cover

Matchings and edge covers

Definition

$M \subseteq E$ is a **matching** in G if $e \cap e' = \emptyset$ for any distinct edges $e, e' \in M$.

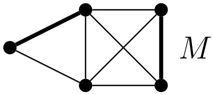


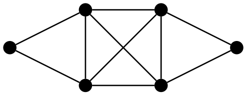


Matchings and edge covers

Definition

$M \subseteq E$ is a **matching** in G if $e \cap e' = \emptyset$ for any distinct edges $e, e' \in M$.



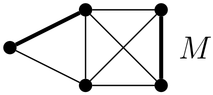


non-perfect matching

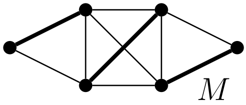
Matchings and edge covers

Definition

$M \subseteq E$ is a **matching** in G if $e \cap e' = \emptyset$ for any distinct edges $e, e' \in M$. A matching is **perfect** if $|M| = \frac{1}{2}|V|$.



non-perfect matching

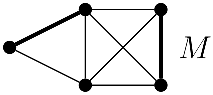


perfect matching

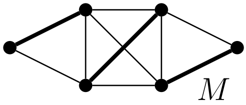
Matchings and edge covers

Definition

$M \subseteq E$ is a **matching** in G if $e \cap e' = \emptyset$ for any distinct edges $e, e' \in M$. A matching is **perfect** if $|M| = \frac{1}{2}|V|$.



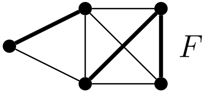
non-perfect matching



perfect matching

Definition

$F \subseteq E$ is an **edge cover** if $V(F) = V$.



edge cover

Optimum stable sets, VC, matchings,..

We define

$$
\begin{aligned}
\alpha(G) &:= \max\{|C| : C \text{ stable set in } G\} = \text{stability } \# \\
\tau(G) &:= \min\{|W| : W \text{ vertex cover}\} = \text{vertex cover } \# \\
\nu(G) &:= \max\{|M| : M \text{ matching}\} = \text{matching } \# \\
\rho(G) &:= \min\{|F| : F \text{ edge cover}\} = \text{ edge cover } \#
\end{aligned}
$$

Optimum stable sets, VC, matchings,..

We define

$$\begin{aligned}
\alpha(G) &:= \max\{|C| : C \text{ stable set in } G\} = \text{stability } \# \\
\tau(G) &:= \min\{|W| : W \text{ vertex cover}\} = \text{vertex cover } \# \\
\nu(G) &:= \max\{|M| : M \text{ matching}\} = \text{matching } \# \\
\rho(G) &:= \min\{|F| : F \text{ edge cover}\} = \text{ edge cover } \#
\end{aligned}$$

Lemma

(i): $\alpha(G) \leq \rho(G)$ *and (ii):* $\nu(G) \leq \tau(G)$

Optimum stable sets, VC, matchings,..

We define

$$
\begin{aligned}
\alpha(G) &:= \max\{|C| : C \text{ stable set in } G\} = \text{stability } \# \\
\tau(G) &:= \min\{|W| : W \text{ vertex cover}\} = \text{vertex cover } \# \\
\nu(G) &:= \max\{|M| : M \text{ matching}\} = \text{matching } \# \\
\rho(G) &:= \min\{|F| : F \text{ edge cover}\} = \text{ edge cover } \#
\end{aligned}
$$

Lemma

(i): $\alpha(G) \leq \rho(G)$ and (ii): $\nu(G) \leq \tau(G)$

Proof of (i).

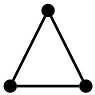
- Let C be independent set, $F \subseteq E$ edge cover
- "assign" $v \in C$ to edge $e \in F$ with $v \in e$.
- Each edge in F gets ≤ 1 node assigned $\Rightarrow |C| \leq |F|$

Optimum stable sets, VC, matchings,..

$$\begin{aligned}
\alpha(G) &:= \max\{|C| : C \text{ stable set in } G\} = \text{stability } \# \\
\tau(G) &:= \min\{|W| : W \text{ vertex cover}\} = \text{vertex cover } \# \\
\nu(G) &:= \max\{|M| : M \text{ matching}\} = \text{matching } \# \\
\rho(G) &:= \min\{|F| : F \text{ edge cover}\} = \text{ edge cover } \#
\end{aligned}$$

Lemma

(i): $\alpha(G) \leq \rho(G)$ and (ii): $\nu(G) \leq \tau(G)$



$\alpha(G) = 1 < 2 = \rho(G)$

$\nu(G) = 1 < 2 = \tau(G)$

Gallai's Theorem

Theorem (Gallai's Theorem)

In any graph without isolated vertices,
$\alpha(G) + \tau(G) = |V| = \nu(G) + \rho(G)$.

Gallai's Theorem

Theorem (Gallai's Theorem)

In any graph without isolated vertices,
$\alpha(G) + \tau(G) = |V| = \nu(G) + \rho(G)$.

First equation:

- We know: C stable set $\Leftrightarrow V \setminus C$ vertex cover

Gallai's Theorem

Theorem (Gallai's Theorem)

In any graph without isolated vertices,
$\alpha(G) + \tau(G) = |V| = \nu(G) + \rho(G)$.

First equation:

- We know: C stable set $\Leftrightarrow V \setminus C$ vertex cover

Second equation: Claim. $\rho(G) = \nu(G) + (|V| - 2\nu(G))$

- Let F be minimum edge cover.
- Let $M \subseteq F$ be inclusion wise maximal matching in F.
- Each $e \in M$ covers 2 new nodes. Each $e \in F \setminus M$ covers 1 new node
- So $|F| = |M| + (|V| - 2|M|)$

Gallai's Theorem

Theorem (Gallai's Theorem)

$\alpha(G) + \tau(G) = |V| = \nu(G) + \rho(G)$.

Corollary

$\alpha(G) = \rho(G) \Leftrightarrow \nu(G) = \tau(G)$

- Recall that always $\alpha(G) \leq \rho(G)$ and $\nu(G) \leq \tau(G)$

M-augmenting paths

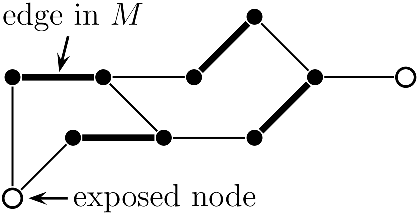
Definition

Let M be a matching in $G = (V, E)$. A path $P = (v_0, \dots, v_t)$ in G is **M-augmenting** if

(i) t is odd

(ii) $\{v_1, v_2\}, \{v_3, v_4\}, \dots, \{v_{t-2}, v_{t-1}\} \in M$

(iii) $v_0, v_t \notin V(M)$



M-augmenting paths

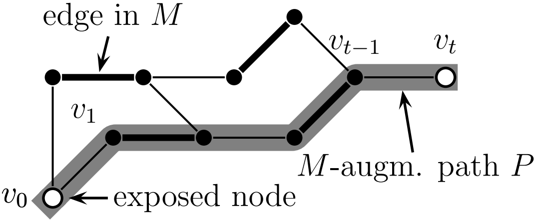
Definition

Let M be a matching in $G = (V, E)$. A path $P = (v_0, \dots, v_t)$ in G is **M-augmenting** if

(i) t is odd

(ii) $\{v_1, v_2\}, \{v_3, v_4\}, \dots, \{v_{t-2}, v_{t-1}\} \in M$

(iii) $v_0, v_t \notin V(M)$



M-augmenting paths (2)

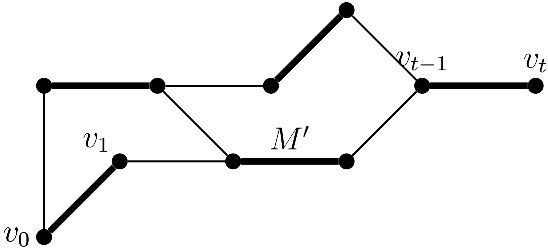
For set A, B, let $A \Delta B := (A \setminus B) \cup (B \setminus A)$ be the **symmetric difference**.

M-augmenting paths (2)

For set A, B, let $A \Delta B := (A \setminus B) \cup (B \setminus A)$ be the **symmetric difference**.

Observation

If P is an M-augmenting path in G, then $M' := M \Delta E(P)$ is a matching in G with $|M'| = |M| + 1$.



M-augmenting paths (3)

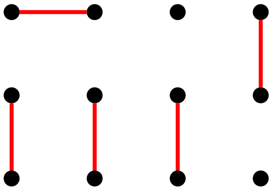
Theorem

Let $G = (V, E)$ be an undirected graph with matching $M \subseteq E$. Either M is a matching of maximum cardinality, or there exists an M-augmenting path.

- Clear: If $\exists$ M-augmenting path $\Rightarrow$ M not optimal

M-augmenting paths (4)

Claim. M not maximal $\Rightarrow \exists M$-augmenting path

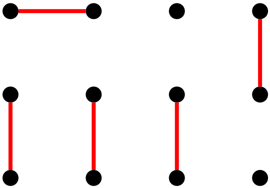


matching M

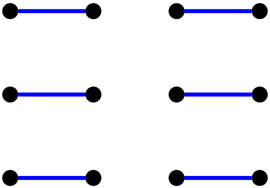
M-augmenting paths (4)

Claim. M not maximal $\Rightarrow \exists M$-augmenting path

- Let $M' \subseteq E$ matching with $|M'| > |M|$.



matching M

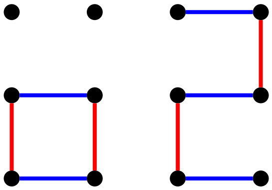


matching M'

M-augmenting paths (4)

Claim. M not maximal $\Rightarrow \exists M$-augmenting path

- Let $M' \subseteq E$ matching with $|M'| > |M|$.
- Consider $G' := (V, M \Delta M')$. Vertices in G' have degrees $\{0, 1, 2\}$

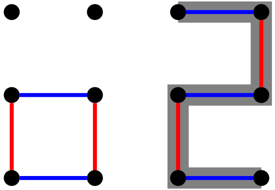


difference $M \Delta M'$

M-augmenting paths (4)

Claim. M not maximal $\Rightarrow \exists M$-augmenting path

- Let $M' \subseteq E$ matching with $|M'| > |M|$.
- Consider $G' := (V, M \Delta M')$. Vertices in G' have degrees $\{0, 1, 2\}$
- Connected components of G' are paths or circuit
- As $|M'| > |M|$, there is a component with more edges from M' than from M
- This component has to be a path with endpoints in M' $\rightarrow$ M-augmenting path $\square$



difference $M \Delta M'$

Kőnig's Theorem

Theorem (Kőnig 1931)

In any bipartite graph, $\nu(G) = \tau(G)$.

Kőnig's Theorem

Theorem (Kőnig 1931)

In any bipartite graph, $\nu(G) = \tau(G)$.

- We know that $\nu(G) \leq \tau(G)$ in any graph.
  Suffices to prove $\nu(G) \geq \tau(G)$.

Kőnig's Theorem

Theorem (Kőnig 1931)

In any bipartite graph, $\nu(G) = \tau(G)$.

- We know that $\nu(G) \le \tau(G)$ in any graph.
  Suffices to prove $\nu(G) \ge \tau(G)$.

Claim I. Let $G = (V, E)$ be bipartite with $|E| \ge 1$. Then G has a vertex that is covered by **every** maximum matching.

Kőnig's Theorem

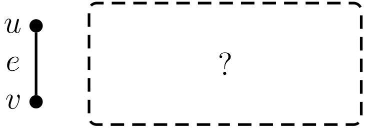
Theorem (Kőnig 1931)

In any bipartite graph, $\nu(G) = \tau(G)$.

- We know that $\nu(G) \leq \tau(G)$ in any graph.
  Suffices to prove $\nu(G) \geq \tau(G)$.

Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

- Suppose not. Fix an edge $e = \{u, v\} \in E$.



Kőnig's Theorem

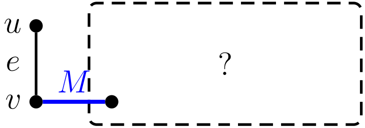
Theorem (Kőnig 1931)

In any bipartite graph, $\nu(G) = \tau(G)$.

- We know that $\nu(G) \leq \tau(G)$ in any graph.
  Suffices to prove $\nu(G) \geq \tau(G)$.

Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

- Suppose not. Fix an edge $e = \{u, v\} \in E$.
- Let M be a maxi. matching with $u \notin V(M)$, $v \in V(M)$.



Kőnig's Theorem

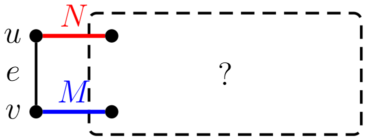
Theorem (Kőnig 1931)

In any bipartite graph, $\nu(G) = \tau(G)$.

- We know that $\nu(G) \leq \tau(G)$ in any graph. Suffices to prove $\nu(G) \geq \tau(G)$.

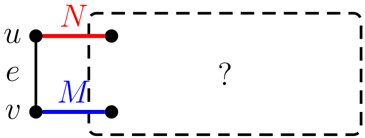
Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

- Suppose not. Fix an edge $e = \{u, v\} \in E$.
- Let M be a maxi. matching with $u \notin V(M)$, $v \in V(M)$.
- Let N be a max. matching with $v \notin V(N)$, $u \in V(N)$.



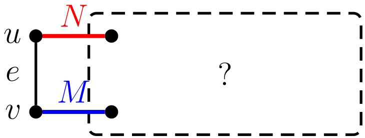
Kőnig's Theorem (2)

- Let P be the component of $(V, M\Delta N)$ containing u



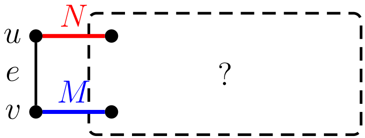
Kőnig's Theorem (2)

- Let P be the component of $(V, M \Delta N)$ containing u
- P is a path of even length (cannot be circuit, cannot have odd length)



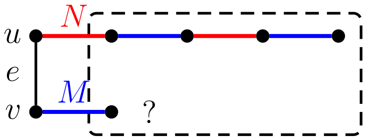
Kőnig's Theorem (2)

- Let P be the component of $(V, M \Delta N)$ containing u
- P is a path of even length (cannot be circuit, cannot have odd length)
- Can P contain v?



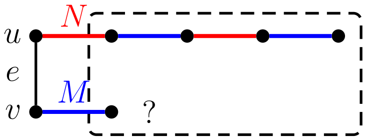
Kőnig's Theorem (2)

- Let P be the component of $(V, M \Delta N)$ containing u
- P is a path of even length (cannot be circuit, cannot have odd length)
- Can P contain v? No, if it did it would end in v and $P + e$ be close odd length cycle in G.



Kőnig's Theorem (2)

- Let P be the component of $(V, M \Delta N)$ containing u
- P is a path of even length (cannot be circuit, cannot have odd length)
- Can P contain v? No, if it did it would end in v and $P + e$ be close odd length cycle in G.
- Then $P + \{u, v\}$ is N-augmenting



Kőnig's Theorem (3)

We know:

Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

Claim II: Any bipartite graph $G = (V, E)$ contains a vertex cover of size $\nu(G)$.

Kőnig's Theorem (3)

We know:

Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

Claim II: Any bipartite graph $G = (V, E)$ contains a vertex cover of size $\nu(G)$.

- Prove by induction on $|V|$

Kőnig's Theorem (3)

We know:

Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

Claim II: Any bipartite graph $G = (V, E)$ contains a vertex cover of size $\nu(G)$.

- Prove by induction on $|V|$
- Let u be vertex from Claim I contained in every matching.

Kőnig's Theorem (3)

We know:

Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

Claim II: Any bipartite graph $G = (V, E)$ contains a vertex cover of size $\nu(G)$.

- Prove by induction on $|V|$
- Let u be vertex from Claim I contained in every matching.
- $G' := G \setminus u$ has $\nu(G') = \nu(G) - 1$

Kőnig's Theorem (3)

We know:

Claim I. Let $G = (V, E)$ be bipartite with $|E| \geq 1$. Then G has a vertex that is covered by **every** maximum matching.

Claim II: Any bipartite graph $G = (V, E)$ contains a vertex cover of size $\nu(G)$.

- Prove by induction on $|V|$
- Let u be vertex from Claim I contained in every matching.
- $G' := G \setminus u$ has $\nu(G') = \nu(G) - 1$
- Let U' be vertex cover for G'. Then $U' \cup \{v\}$ is vertex cover for G. □

Consequence of Kőnig's Theorem

Corollary

Let $G = (V, E)$ be bipartite without isolated vertices. Then $\alpha(G) = \rho(G)$.

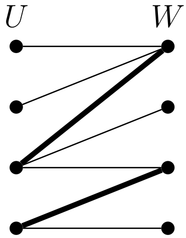
- König: $\nu(G) = \tau(G)$
- Earlier Cor: $\alpha(G) = \rho(G) \Leftrightarrow \nu(G) = \tau(G)$ $\square$

Lecture 7

Chapter 3 — Matchings and Covers in Bipartite Graphs — Part 2/2

Maximum cardinality matching in a bipartite graph

- **Input:** Bipartite graph $G = (V, E)$ with $V = U \dot{\cup} W$ and matching $M \subseteq E$
- **Output:** Matching $M' \subseteq E$ w. $|M'| > |M|$ if there is one

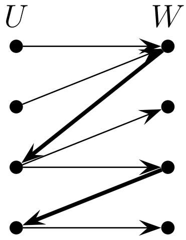


Maximum cardinality matching in a bipartite graph

- **Input:** Bipartite graph $G = (V, E)$ with $V = U \dot{\cup} W$ and matching $M \subseteq E$
- **Output:** Matching $M' \subseteq E$ w. $|M'| > |M|$ if there is one

(1) Define a directed graph $D = (U \cup W, E')$ with

$$E' := \{(w,u) \in W \times U \mid \{u,w\} \in M\} \cup \{(u,w) \in U \times W \mid \{u,w\} \in E \setminus M$$



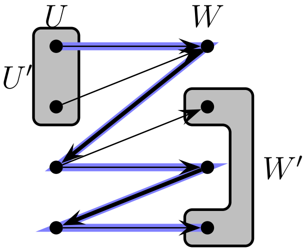
Maximum cardinality matching in a bipartite graph

- **Input:** Bipartite graph $G = (V, E)$ with $V = U \dot{\cup} W$ and matching $M \subseteq E$
- **Output:** Matching $M' \subseteq E$ w. $|M'| > |M|$ if there is one

(1) Define a directed graph $D = (U \cup W, E')$ with

$$E' := \{(w, u) \in W \times U \mid \{u, w\} \in M\} \cup \{(u, w) \in U \times W \mid \{u, w\} \in E \backslash M$$

(2) Set $U' := U \setminus V(M)$, $W' := W \setminus V(M)$. Return any path from U' to W' in D



Analysis

Theorem

The algorithm is correct. Moreover, a maximum-size matching in a bipartite graph can be found in time $O(|V| \cdot |E|)$.

Analysis

Theorem

The algorithm is correct. Moreover, a maximum-size matching in a bipartite graph can be found in time $O(|V| \cdot |E|)$.

Correctness:

- An M-augmenting path in G has to go between a node in U' and a node in W' (as it is an odd length path in a bipartite graph). Hence it corresponds to a directed path in D.
- In reverse, a directed path corresponds to an M-augmenting path

Analysis

Theorem

The algorithm is correct. Moreover, a maximum-size matching in a bipartite graph can be found in time $O(|V| \cdot |E|)$.

Correctness:

- An M-augmenting path in G has to go between a node in U' and a node in W' (as it is an odd length path in a bipartite graph). Hence it corresponds to a directed path in D.
- In reverse, a directed path corresponds to an M-augmenting path

Running time:

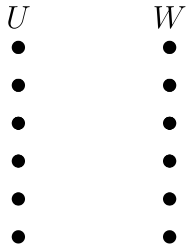
- At most $|V|/2$ augmentations needed. Each augmentation takes time $O(|E|)$.

Computing Min Vertex Covers

Lemma

In a bipartite graph $G = (V, E)$ ($V = U \cup W$) one can find a minimum vertex cover in polynomial time

- Not surprising as $\nu(G) = \tau(G)$ in bipartite graphs

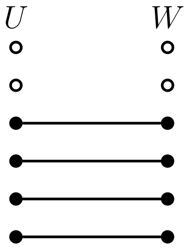


Computing Min Vertex Covers

Lemma

In a bipartite graph $G = (V, E)$ ($V = U \cup W$) one can find a minimum vertex cover in polynomial time

- Not surprising as $\nu(G) = \tau(G)$ in bipartite graphs
- Fix a maximum matching M^*.

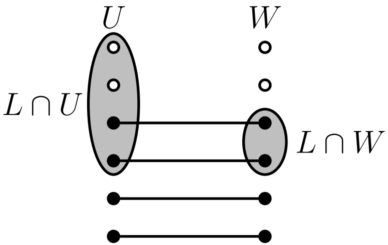


Computing Min Vertex Covers

Lemma

In a bipartite graph $G = (V, E)$ ($V = U \cup W$) one can find a minimum vertex cover in polynomial time

- Not surprising as $\nu(G) = \tau(G)$ in bipartite graphs
- Fix a maximum matching M^*.
- Consider the directed graph $D = (V, E')$ from earlier. Let $L \subseteq V$ be nodes reachable from M^*-exposed vertices in U.

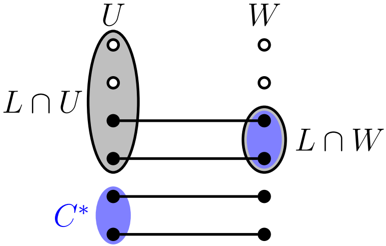


Computing Min Vertex Covers

Lemma

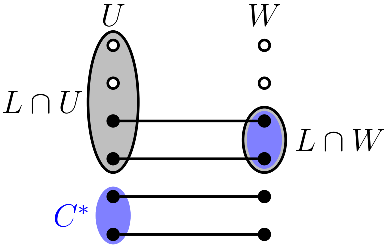
In a bipartite graph $G = (V, E)$ ($V = U \cup W$) one can find a minimum vertex cover in polynomial time

- Not surprising as $\nu(G) = \tau(G)$ in bipartite graphs
- Fix a maximum matching M^*.
- Consider the directed graph $D = (V, E')$ from earlier. Let $L \subseteq V$ be nodes reachable from M^*-exposed vertices in U.
- Set $C^* := (U \setminus L) \cup (W \cap L)$



Computing Min Vertex Covers (2)

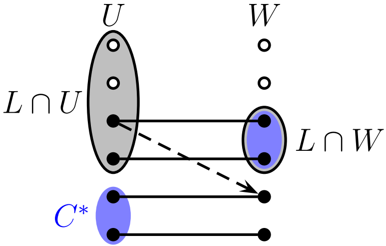
Claim I. C^* is vertex cover.



Computing Min Vertex Covers (2)

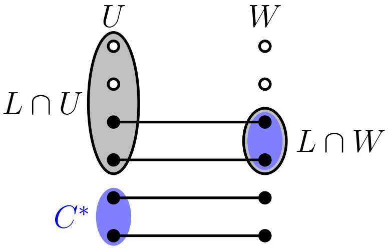
Claim I. C^* is vertex cover.

- Suppose there is an edge $\{u, w\} \in E$ with $u \in L \cap U$ and $w \in W \setminus L$
- But u was reachable in D from an M^*-exposed node in U and w was not. Contradiction!



Computing Min Vertex Covers (3)

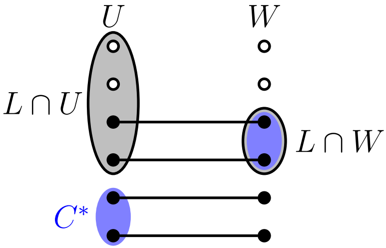
Claim II. $|C^*| \leq |M^*|$



Computing Min Vertex Covers (3)

Claim II. $|C^*| \leq |M^*|$

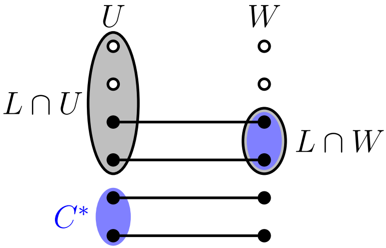
- **Observation 1:** $C^* \subseteq V(M)$
  Reason: M^*-exposed nodes in U reachable by construction. M^*-exposed nodes in W **not** reachable since then there would be an M^*-augmenting path



Computing Min Vertex Covers (3)

Claim II. $|C^*| \leq |M^*|$

- **Observation 1:** $C^* \subseteq V(M)$
  Reason: M^*-exposed nodes in U reachable by construction. M^*-exposed nodes in W **not** reachable since then there would be an M^*-augmenting path
- **Observation 2:** Each $e \in M^*$ has $|e \cap C^*| \leq 1$
  Reason: Otherwise both nodes e reachable, which is a contradiction!



Application: Assignment problem

Setting: We have machines $m_1, \ldots, m_k$ and jobs $j_1, \ldots, j_s$. Each machine is suitable only for a certain subset of jobs. Moreover each machine can only process one job per day.

Goal: Maximize the number of processed jobs

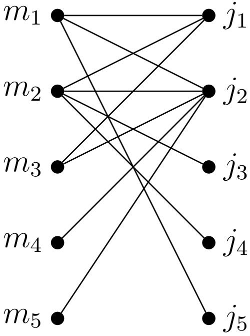
	j_1	j_2	j_3	j_4	j_5
m_1	X	X			X
m_2	X	X	X	X	
m_3	X	X			
m_4		X			
m_5		X			

Application: Assignment problem

Setting: We have machines $m_1, \dots, m_k$ and jobs $j_1, \dots, j_s$. Each machine is suitable only for a certain subset of jobs. Moreover each machine can only process one job per day.

Goal: Maximize the number of processed jobs

	j_1	j_2	j_3	j_4	j_5
m_1	X	X			X
m_2	X	X	X	X	
m_3	X	X			
m_4		X			
m_5		X			



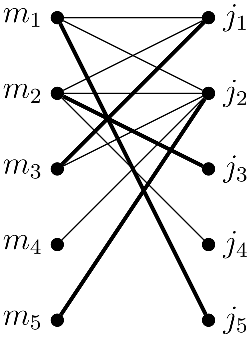
- Model as bipartite graph. Then compute max cardinality matching

Application: Assignment problem

Setting: We have machines $m_1, \ldots, m_k$ and jobs $j_1, \ldots, j_s$. Each machine is suitable only for a certain subset of jobs. Moreover each machine can only process one job per day.

Goal: Maximize the number of processed jobs

	j_1	j_2	j_3	j_4	j_5
m_1	X	X			**X**
m_2	X	X	**X**	X	
m_3	**X**	X			
m_4		X			
m_5		**X**			



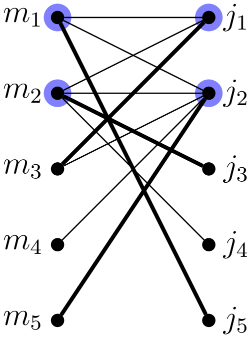
- Model as bipartite graph. Then compute max cardinality matching

Application: Assignment problem

Setting: We have machines $m_1, \ldots, m_k$ and jobs $j_1, \ldots, j_s$. Each machine is suitable only for a certain subset of jobs. Moreover each machine can only process one job per day.

Goal: Maximize the number of processed jobs

	j_1	j_2	j_3	j_4	j_5
m_1	X	X			**X**
m_2	X	X	**X**	X	
m_3	**X**	X			
m_4		X			
m_5		**X**			

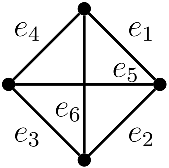


- Model as bipartite graph. Then compute max cardinality matching
- $\{m_1, m_2, j_1, j_2\}$ vertex cover $\Rightarrow \nu(G) \le 4$

The Matching Polytope of a Graph

Fix a graph $G = (V, E)$. For $M \subseteq E$ we define the **characteristic vector / incidence vector** as $\chi_M \in \mathbb{R}^E$ with

$$\chi^M(e) = \begin{cases} 1 & \text{if } e \in M \\ 0 & \text{if } e \notin M \end{cases}$$



$$\chi^{\{e_1,e_3\}} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \chi^{\{e_5,e_6\}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \quad \chi^{\emptyset} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

The Matching Polytope of a Graph (2)

Definition

For an undirected graph G, the **matching polytope** is

$$P_{\text{matching}}(G) := \text{conv}\{\chi^M \in \mathbb{R}^E \mid M \subseteq E \text{ is matching}\}$$

and the **perfect matching polytope** is

$$P_{\text{perfectmatching}}(G) := \text{conv}\{\chi^M \in \mathbb{R}^E \mid M \subseteq E \text{ is perfect matching}\}$$

The Matching Polytope of a Graph (2)

Definition

For an undirected graph G, the **matching polytope** is

$$P_{\text{matching}}(G) := \operatorname{conv}\{\chi^M \in \mathbb{R}^E \mid M \subseteq E \text{ is matching}\}$$

and the **perfect matching polytope** is

$$P_{\text{perfectmatching}}(G) := \operatorname{conv}\{\chi^M \in \mathbb{R}^E \mid M \subseteq E \text{ is perfect matching}\}$$

- Both are polytopes
- Both only have vertices in $\{0, 1\}^E$

The Matching Polytope of a Graph (2)

Definition

For an undirected graph G, the **matching polytope** is

$$P_{\text{matching}}(G) := \operatorname{conv}\{\chi^M \in \mathbb{R}^E \mid M \subseteq E \text{ is matching}\}$$

and the **perfect matching polytope** is

$$P_{\text{perfectmatching}}(G) := \operatorname{conv}\{\chi^M \in \mathbb{R}^E \mid M \subseteq E \text{ is perfect matching}\}$$

- Both are polytopes
- Both only have vertices in $\{0,1\}^E$
- What are the inequalities defining both?

The Matching Polytope of a Graph (3)

Define

$$Q_{\text{matching}}(G) := \left\{ x \in \mathbb{R}^E \mid \begin{array}{rcll} \sum_{e:v\in e} x_e & \leq & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\}$$

$$Q_{\text{perfectmatching}}(G) := \left\{ x \in \mathbb{R}^E \mid \begin{array}{rcll} \sum_{e:v\in e} x_e & = & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\}$$

The Matching Polytope of a Graph (3)

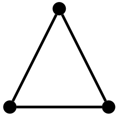
Define

$$\begin{array}{rcl} Q_{\text{matching}}(G) & := & \left\{ x \in \mathbb{R}^E \;\middle|\; \begin{array}{rcll} \sum_{e:v\in e} x_e & \leq & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\} \\ Q_{\text{perfectmatching}}(G) & := & \left\{ x \in \mathbb{R}^E \;\middle|\; \begin{array}{rcll} \sum_{e:v\in e} x_e & = & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\} \end{array}$$

Observe:

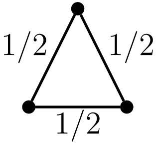
$$\begin{array}{rcl} Q_{\text{matching}}(G) & = & \text{conv}\{Q_{\text{matching}}(G) \cap \mathbb{Z}^E\} \\ Q_{\text{perfectmatching}}(G) & = & \text{conv}\{Q_{\text{perfectmatching}}(G) \cap \mathbb{Z}^E\} \end{array}$$

The Matching Polytope of a Graph (4)



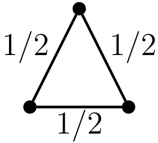
- Let G be triangle.

The Matching Polytope of a Graph (4)



- Let G be triangle. Then $x^* := \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \end{pmatrix}$ lies in $Q_{\text{matching}}(G)$ but not in $P_{\text{matching}}(G)$.

The Matching Polytope of a Graph (4)



- Let G be triangle. Then $x^* := \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \end{pmatrix}$ lies in $Q_{\text{matching}}(G)$ but not in $P_{\text{matching}}(G)$.
- Recall

$$P_{\text{matching}}(G) = \text{conv}\left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}$$

The Matching Polytope of a Graph (5)

Theorem

If G is bipartite, then $P_{matching}(G) = Q_{matching}(G)$ and $P_{perfectmatching}(G) = Q_{perfectmatching}(G)$.

- False in non-bipartite graphs!

The Matching Polytope of a Graph (5)

Theorem

If G is bipartite, then $P_{matching}(G) = Q_{matching}(G)$ and $P_{perfectmatching}(G) = Q_{perfectmatching}(G)$.

- False in non-bipartite graphs!

The following algorithm finds a matching M in a bipartite graph maximizing $\sum_{e \in M} w_e$:

(1) Find an optimum vertex solution x^* for the LP

$$\max \left\{ \sum_{e \in E} w_e x_e \mid \sum_{e: v \in e} x_e \leq 1 \ \forall v \in V, \ \ x_e \geq 0 \ \forall e \in E \right\}$$

(2) Return $\{e \in E \mid x_e^* = 1\}$

Lecture 8

Chapter 8 — Integer Linear Programming and Totally Unimodular Matrices — Part 1/2

Integer linear programming

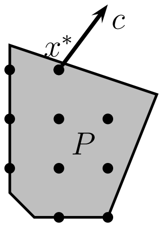
Definition

A problem of the form

$$\max\left\{c^T x \mid Ax \leq b; x \in \mathbb{Z}^n\right\}$$

is called an **integer linear program**.

- In general such a problem is **NP**-hard to solve!



Integer linear programming (2)

Observation: One always has

$$\max\{c^T x \mid Ax \leq b, x \in \mathbb{Z}^n\} \leq \max\{c^T x \mid Ax \leq b\}$$

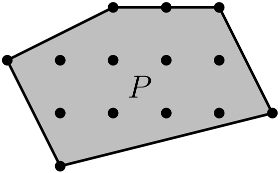
Integer linear programming (2)

Observation: One always has

$$\max\{c^T x \mid Ax \leq b, x \in \mathbb{Z}^n\} \leq \max\{c^T x \mid Ax \leq b\}$$

Definition

A polytope P is **integer / integral**, if all vertices are integer vectors.



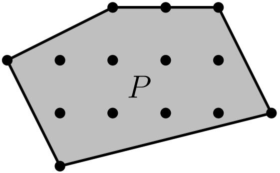
Integer linear programming (2)

Observation: One always has

$$\max\{c^T x \mid Ax \leq b, x \in \mathbb{Z}^n\} \leq \max\{c^T x \mid Ax \leq b\}$$

Definition

A polytope P is **integer / integral**, if all vertices are integer vectors.

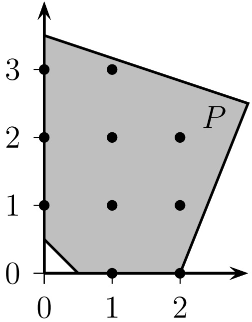


- For example $P_{\text{matching}}(G)$ and $P_{\text{perfectmatching}}(G)$
- **Exercise:** For a polytope P one has: P integer $\Leftrightarrow \forall c$ one has $\max\{c^T x \mid x \in P\} = \max\{c^T x \mid x \in P \cap \mathbb{Z}^n\}$.

Integer linear programming (3)

Definition

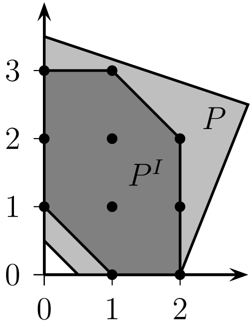
For a polyhedron $P \subseteq \mathbb{R}^n$ we define the **integer hull** as $P^I := \text{conv}(P \cap \mathbb{Z}^n)$.



Integer linear programming (3)

Definition

For a polyhedron $P \subseteq \mathbb{R}^n$ we define the **integer hull** as $P^I := \mathrm{conv}(P \cap \mathbb{Z}^n)$.



- Note that for a polytope one has: P integral $\Leftrightarrow P = P^I$

Totally unimodular matrices

Definition

A matrix $A \in \mathbb{R}^{m \times n}$ is called **totally unimodular** (TU) if each square submatrix of A has determinant in $\{-1, 0, 1\}$.

Totally unimodular matrices

Definition

A matrix $A \in \mathbb{R}^{m \times n}$ is called **totally unimodular** (TU) if each square submatrix of A has determinant in $\{-1, 0, 1\}$.

- If A is TU, then $A \in \{-1, 0, 1\}^{m \times n}$
- For example I_n is totally unimodular

Totally unimodular matrices

Definition

A matrix $A \in \mathbb{R}^{m \times n}$ is called **totally unimodular** (TU) if each square submatrix of A has determinant in $\{-1, 0, 1\}$.

- If A is TU, then $A \in \{-1, 0, 1\}^{m \times n}$
- For example I_n is totally unimodular

Cramer's Rule

Let $B \in \mathbb{Z}^{n \times n}$ be invertible. Then $B^{-1} = \frac{1}{\det(B)} C$ where $C \in \mathbb{Z}^{n \times n}$.

Totally unimodular matrices (2)

Theorem

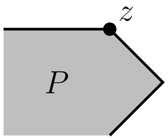
If $A \in \mathbb{R}^{m \times n}$ is TU and $b \in \mathbb{Z}^m$, then every vertex of the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

Totally unimodular matrices (2)

Theorem

If $A \in \mathbb{R}^{m \times n}$ is TU and $b \in \mathbb{Z}^m$, then every vertex of the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Let z be a vertex of P. Then $\text{rank}(A_z) = n$ for the matrix with $A_z z = b_z$.

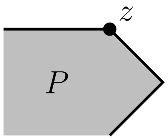


Totally unimodular matrices (2)

Theorem

If $A \in \mathbb{R}^{m \times n}$ is TU and $b \in \mathbb{Z}^m$, then every vertex of the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Let z be a vertex of P. Then $\text{rank}(A_z) = n$ for the matrix with $A_z z = b_z$.
- Let $Bz = b'$ be a subsystem of n linear independent equations (i.e. $B \in \mathbb{Z}^{n \times n}$)

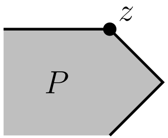


Totally unimodular matrices (2)

Theorem

If $A \in \mathbb{R}^{m \times n}$ is TU and $b \in \mathbb{Z}^m$, then every vertex of the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Let z be a vertex of P. Then $\text{rank}(A_z) = n$ for the matrix with $A_z z = b_z$.
- Let $Bz = b'$ be a subsystem of n linear independent equations (i.e. $B \in \mathbb{Z}^{n \times n}$)
- Then $\det(B) \in \{-1, 1\}$ (since A is TU) and so $B^{-1} \in \mathbb{Z}^{n \times n}$
- Then $z = B^{-1} b' \in \mathbb{Z}^n$ as $b' \in \mathbb{Z}^n$. $\square$

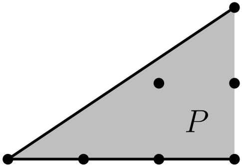


Polyhedra from TU matrices

- Not every polyhedron P has vertices

Definition

A polyhedron P is **integer** if for all $c \in \mathbb{R}^n$ where $\max\{c^T x \mid x \in P\} < \infty$, the maximum is attained by some integer vector.



Polyhedra from TU matrices (2)

Corollary

Let $A \in \mathbb{R}^{m \times n}$ *be TU and* $b \in \mathbb{Z}^m$*. Then the polyhedron* $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ *is integral.*

Polyhedra from TU matrices (2)

Corollary

Let $A \in \mathbb{R}^{m \times n}$ be TU and $b \in \mathbb{Z}^m$. Then the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Fix a $c \in \mathbb{R}^n$ so that $\max\{c^T x \mid x \in P\} < \infty$ and let x^* be an optimum solution (possibly fractional).

Polyhedra from TU matrices (2)

Corollary

Let $A \in \mathbb{R}^{m \times n}$ be TU and $b \in \mathbb{Z}^m$. Then the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Fix a $c \in \mathbb{R}^n$ so that $\max\{c^T x \mid x \in P\} < \infty$ and let x^* be an optimum solution (possibly fractional).
- Choose $d', d'' \in \mathbb{Z}^n$ so that $d' \leq x^* \leq d''$

Polyhedra from TU matrices (2)

Corollary

Let $A \in \mathbb{R}^{m \times n}$ be TU and $b \in \mathbb{Z}^m$. Then the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Fix a $c \in \mathbb{R}^n$ so that $\max\{c^T x \mid x \in P\} < \infty$ and let x^* be an optimum solution (possibly fractional).
- Choose $d', d'' \in \mathbb{Z}^n$ so that $d' \leq x^* \leq d''$
- Consider the polytope

$$\begin{aligned} Q &:= \{x \in \mathbb{R}^n \mid Ax \leq b, d' \leq x \leq d''\} \\ &= \left\{ x \in \mathbb{R}^n \mid \begin{pmatrix} A \\ -I_n \\ I_n \end{pmatrix} x \leq \begin{pmatrix} b \\ -d' \\ d'' \end{pmatrix} \right\} \end{aligned}$$

Polyhedra from TU matrices (2)

Corollary

Let $A \in \mathbb{R}^{m \times n}$ be TU and $b \in \mathbb{Z}^m$. Then the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Fix a $c \in \mathbb{R}^n$ so that $\max\{c^T x \mid x \in P\} < \infty$ and let x^* be an optimum solution (possibly fractional).
- Choose $d', d'' \in \mathbb{Z}^n$ so that $d' \leq x^* \leq d''$
- Consider the polytope

$$\begin{aligned} Q &:= \{x \in \mathbb{R}^n \mid Ax \leq b, d' \leq x \leq d''\} \\ &= \left\{ x \in \mathbb{R}^n \mid \begin{pmatrix} A \\ -I_n \\ I_n \end{pmatrix} x \leq \begin{pmatrix} b \\ -d' \\ d'' \end{pmatrix} \right\} \end{aligned}$$

- Easy to check that A TU $\Rightarrow \begin{pmatrix} A \\ -I_n \\ I_n \end{pmatrix}$ TU, RHS is integral

Polyhedra from TU matrices (2)

Corollary

Let $A \in \mathbb{R}^{m \times n}$ be TU and $b \in \mathbb{Z}^m$. Then the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Fix a $c \in \mathbb{R}^n$ so that $\max\{c^T x \mid x \in P\} < \infty$ and let x^* be an optimum solution (possibly fractional).
- Choose $d', d'' \in \mathbb{Z}^n$ so that $d' \leq x^* \leq d''$
- Consider the polytope

$$\begin{aligned} Q &:= \{x \in \mathbb{R}^n \mid Ax \leq b, d' \leq x \leq d''\} \\ &= \left\{ x \in \mathbb{R}^n \mid \begin{pmatrix} A \\ -I_n \\ I_n \end{pmatrix} x \leq \begin{pmatrix} b \\ -d' \\ d'' \end{pmatrix} \right\} \end{aligned}$$

- Easy to check that A TU $\Rightarrow \left(\begin{smallmatrix} A \\ -I_n \\ I_n \end{smallmatrix}\right)$ TU, RHS is integral
- So there is an integral optimum $\tilde{x}$ for $\max\{c^T x \mid x \in Q\}$.

Polyhedra from TU matrices (2)

Corollary

Let $A \in \mathbb{R}^{m \times n}$ be TU and $b \in \mathbb{Z}^m$. Then the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is integral.

- Fix a $c \in \mathbb{R}^n$ so that $\max\{c^T x \mid x \in P\} < \infty$ and let x^* be an optimum solution (possibly fractional).
- Choose $d', d'' \in \mathbb{Z}^n$ so that $d' \leq x^* \leq d''$
- Consider the polytope

$$
\begin{aligned}
Q \;&:=\; \{x \in \mathbb{R}^n \mid Ax \leq b, d' \leq x \leq d''\} \\
&= \left\{ x \in \mathbb{R}^n \mid \begin{pmatrix} A \\ -I_n \\ I_n \end{pmatrix} x \leq \begin{pmatrix} b \\ -d' \\ d'' \end{pmatrix} \right\}
\end{aligned}
$$

- Easy to check that A TU $\Rightarrow \left(\begin{smallmatrix} A \\ -I_n \\ I_n \end{smallmatrix}\right)$ TU, RHS is integral
- So there is an integral optimum $\tilde{x}$ for $\max\{c^T x \mid x \in Q\}$.
- As $x^* \in Q$, one has $c^T \tilde{x} \geq c^T x^*$. $\square$

LP duality and TUness

Corollary

Let $A \in \mathbb{R}^{m\times n}$ TU, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$. Both LPs

$$\max\{c^T x \mid Ax \leq b\} = \min\{y^T b \mid y^T A = c^T,\ y \geq \mathbf{0}\}$$

have integral optimum solutions (assuming the LPs are feasible).

LP duality and TUness

Corollary

Let $A \in \mathbb{R}^{m \times n}$ TU, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$. Both LPs

$$\max\{c^T x \mid Ax \leq b\} = \min\{y^T b \mid y^T A = c^T,\ y \geq \mathbf{0}\}$$

have integral optimum solutions (assuming the LPs are feasible).

- By prev. Cor., primal has integral opt.

LP duality and TUness

Corollary

Let $A \in \mathbb{R}^{m \times n}$ *TU,* $b \in \mathbb{Z}^m$*,* $c \in \mathbb{Z}^n$*. Both LPs*

$$\max\{c^T x \mid Ax \leq b\} = \min\{y^T b \mid y^T A = c^T,\ y \geq \mathbf{0}\}$$

have integral optimum solutions (assuming the LPs are feasible).

- By prev. Cor., primal has integral opt.
- The dual is

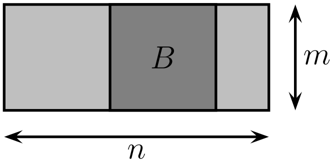
$$\min\left\{y^T b \mid \begin{pmatrix} A^T \\ A^T \\ -I_m \end{pmatrix} y \leq \begin{pmatrix} c^T \\ -c^T \\ \mathbf{0} \end{pmatrix}\right\}$$

- Observe A TU $\Rightarrow \begin{pmatrix} A^T \\ A^T \\ -I_m \end{pmatrix}$ TU and RHS is integral. $\square$

Unimodularity

Definition

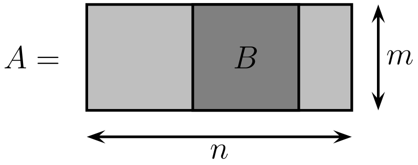
A matrix $A \in \mathbb{R}^{m \times n}$ is **unimodular**, if $\operatorname{rank}(A) = m$ and any $m \times m$ submatrix B of A has determinant in $-1, 0, 1$.

$A =$ 

Unimodularity

Definition

A matrix $A \in \mathbb{R}^{m \times n}$ is **unimodular**, if $\text{rank}(A) = m$ and any $m \times m$ submatrix B of A has determinant in $-1, 0, 1$.



Example:

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 8 & 1 & -1 \end{pmatrix}$$

is unimodular but not TU.

Unimodularity (2)

Lemma

Let $A \in \mathbb{R}^{m \times n}$. Then A TU $\Leftrightarrow [I_m, A]$ unimodular.

- "$\Rightarrow$" clear as $\text{rank}([I_m, A]) = m$

Unimodularity (2)

Lemma

Let $A \in \mathbb{R}^{m \times n}$. Then A TU $\Leftrightarrow [I_m, A]$ unimodular.

- "$\Rightarrow$" clear as $\operatorname{rank}([I_m, A]) = m$
- Now $\Leftarrow$. Let $B \in \mathbb{R}^{k \times k}$ be a submatrix of A.
- Then after permutation of columns and rows,

$$\begin{pmatrix} I_{m-k} & 0 \\ 0 & B \end{pmatrix}$$

is a submatrix of $[I_m, A]$.

Unimodularity (2)

Lemma

Let $A \in \mathbb{R}^{m \times n}$. Then A TU $\Leftrightarrow [I_m, A]$ unimodular.

- "$\Rightarrow$" clear as $\operatorname{rank}([I_m, A]) = m$
- Now $\Leftarrow$. Let $B \in \mathbb{R}^{k \times k}$ be a submatrix of A.
- Then after permutation of columns and rows,

$$\begin{pmatrix} I_{m-k} & 0 \\ 0 & B \end{pmatrix}$$

  is a submatrix of $[I_m, A]$.
- Then

$$\det(B) = \pm \det \begin{pmatrix} I_{m-k} & 0 \\ 0 & B \end{pmatrix} \in \{-1, 0, 1\}$$

$\square$

Unimodularity vs. integrality

Theorem

Let $A \in \mathbb{Z}^{m \times n}$ *and* $rank(A) = m$. *Then* A *is unimodular* $\Leftrightarrow \forall b \in \mathbb{Z}^m : P_b := \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ *is integral.*

Unimodularity vs. integrality

Theorem

Let $A \in \mathbb{Z}^{m \times n}$ and $\mathrm{rank}(A) = m$. Then A is unimodular $\Leftrightarrow \forall b \in \mathbb{Z}^m : P_b := \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ is integral.

Claim 1. A unimodular $\Rightarrow P_b$ integral.

Unimodularity vs. integrality

Theorem

Let $A \in \mathbb{Z}^{m \times n}$ and $\text{rank}(A) = m$. Then A is unimodular $\Leftrightarrow \forall b \in \mathbb{Z}^m : P_b := \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ is integral.

Claim 1. A unimodular $\Rightarrow P_b$ integral.

- We write

$$P_b = \{x \mid Ax = b, x \geq 0\} = \left\{ x \in \mathbb{R}^n \mid \underbrace{\begin{pmatrix} A \\ -A \\ -I_n \end{pmatrix}}_{=:D} x \leq \underbrace{\begin{pmatrix} b \\ -b \\ \mathbf{0} \end{pmatrix}}_{=:f} \right\}$$

Unimodularity vs. integrality

Theorem

Let $A \in \mathbb{Z}^{m \times n}$ and $\mathrm{rank}(A) = m$. Then A is unimodular $\Leftrightarrow \forall b \in \mathbb{Z}^m : P_b := \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ is integral.

Claim 1. A unimodular $\Rightarrow P_b$ integral.

- We write

$$P_b = \{x \mid Ax = b, x \geq 0\} = \left\{ x \in \mathbb{R}^n \mid \underbrace{\begin{pmatrix} A \\ -A \\ -I_n \end{pmatrix}}_{=:D} x \leq \underbrace{\begin{pmatrix} b \\ -b \\ \mathbf{0} \end{pmatrix}}_{=:f} \right\}$$

- We have $\mathrm{rank}(A) = n \Rightarrow$ optimum for $\max\{c^T x \mid Ax = b, x \geq \mathbf{0}\}$ (if finite) attained by some vertex z

Unimodularity vs. integrality

Theorem

Let $A \in \mathbb{Z}^{m\times n}$ and $\mathrm{rank}(A) = m$. Then A is unimodular $\Leftrightarrow \forall b \in \mathbb{Z}^m : P_b := \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ is integral.

Claim 1. A unimodular $\Rightarrow P_b$ integral.

- We write

$$P_b = \{x \mid Ax = b, x \geq 0\} = \left\{ x \in \mathbb{R}^n \mid \underbrace{\begin{pmatrix} A \\ -A \\ -I_n \end{pmatrix}}_{=:D} x \leq \underbrace{\begin{pmatrix} b \\ -b \\ \mathbf{0} \end{pmatrix}}_{=:f} \right\}$$

- We have $\mathrm{rank}(A) = n \Rightarrow$ optimum for $\max\{c^T x \mid Ax = b, x \geq \mathbf{0}\}$ (if finite) attained by some vertex z
- D_z contains all rows of $A, -A$ and some rows of $-I_n$

Unimodularity vs. integrality (2)

- There is a matrix $B \in \mathbb{R}^{n\times n}$ containing all rows of A plus $n-m$ rows of $-I_n$ so that $Bz = f'$ has unique solution of z.

Unimodularity vs. integrality (2)

- There is a matrix $B \in \mathbb{R}^{n \times n}$ containing all rows of A plus $n - m$ rows of $-I_n$ so that $Bz = f'$ has unique solution of z.
- Up to permuting columns, the picture looks like this:

$$B = \left[\begin{array}{c|c} \multicolumn{2}{c}{A} \\ \hline -I_{n-m} & \mathbf{0} \end{array}\right]$$

Unimodularity vs. integrality (2)

- There is a matrix $B \in \mathbb{R}^{n \times n}$ containing all rows of A plus $n - m$ rows of $-I_n$ so that $Bz = f'$ has unique solution of z.
- Up to permuting columns, the picture looks like this:

$$B = \left[\begin{array}{c|c} A & \tilde{A} \\ \hline -I_{n-m} & \mathbf{0} \end{array}\right]$$

- Then for some $m \times m$ submatrix $\tilde{A}$ of A one has $\det(B) = \pm \det(\tilde{A}) \in \{-1, 0, 1\}$ as A is unimodular. $\square$

Direction "$\Leftarrow$" in the next lecture!

Lecture 9

Chapter 8 — Integer Linear Programming and Totally Unimodular Matrices — Part 2/2

Unimodularity vs. integrality (3)

Claim 2. $P_b = \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ integral for all $b \in \mathbb{Z}^m \Rightarrow A$ unimodular.

Unimodularity vs. integrality (3)

Claim 2. $P_b = \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ integral for all $b \in \mathbb{Z}^m \Rightarrow A$ unimodular.

- Let B be non-singular $m \times m$ submatrix of A. Need to prove that $\det(B) \in \{-1, 1\}$

Unimodularity vs. integrality (3)

Claim 2. $P_b = \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ integral for all $b \in \mathbb{Z}^m \Rightarrow A$ unimodular.

- Let B be non-singular $m \times m$ submatrix of A. Need to prove that $\det(B) \in \{-1, 1\}$
- Showing Claim 3 suffices.

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

Unimodularity vs. integrality (3)

Claim 2. $P_b = \{x \in \mathbb{R}^n \mid Ax = b, x \geq \mathbf{0}\}$ integral for all $b \in \mathbb{Z}^m \Rightarrow A$ unimodular.

- Let B be non-singular $m \times m$ submatrix of A. Need to prove that $\det(B) \in \{-1, 1\}$
- Showing Claim 3 suffices. To see this:

$$\text{Claim 3} \quad \Rightarrow \quad B^{-1} \in \mathbb{Z}^{m \times m} \quad \Rightarrow \quad \det(B^{-1}) \in \mathbb{Z},$$

  then from $\det(B) \cdot \det(B^{-1}) = 1$ and $\det(B) \in \mathbb{Z}$, we get $\det(B) \in \{-1, 1\}$.

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

Unimodularity vs. integrality (3)

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

- Pick a $u \in \mathbb{Z}^m$ so that $z := u + B^{-1}v > \mathbf{0}$. Suffices to prove that $z \in \mathbb{Z}^m$!

Unimodularity vs. integrality (3)

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

- Pick a $u \in \mathbb{Z}^m$ so that $z := u + B^{-1}v > \mathbf{0}$. Suffices to prove that $z \in \mathbb{Z}^m$!
- Set $b := Bz$. Then $b = Bz = Bu + BB^{-1}v = Bu + v \in \mathbb{Z}^m$

Unimodularity vs. integrality (3)

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

- Pick a $u \in \mathbb{Z}^m$ so that $z := u + B^{-1}v > \mathbf{0}$. Suffices to prove that $z \in \mathbb{Z}^m$!
- Set $b := Bz$. Then $b = Bz = Bu + BB^{-1}v = Bu + v \in \mathbb{Z}^m$
- Note that $\binom{z}{\mathbf{0}} \in P_b$.

Unimodularity vs. integrality (3)

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

- Pick a $u \in \mathbb{Z}^m$ so that $z := u + B^{-1}v > \mathbf{0}$. Suffices to prove that $z \in \mathbb{Z}^m$!
- Set $b := Bz$. Then $b = Bz = Bu + BB^{-1}v = Bu + v \in \mathbb{Z}^m$
- Note that $\binom{z}{\mathbf{0}} \in P_b$.
- Tight constraints for $\binom{z}{\mathbf{0}}$ have rank n:

$$\begin{pmatrix} B & * \\ \mathbf{0} & -I_{n-m} \end{pmatrix} \begin{pmatrix} z \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} b \\ \mathbf{0} \end{pmatrix}$$

Unimodularity vs. integrality (3)

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

- Pick a $u \in \mathbb{Z}^m$ so that $z := u + B^{-1}v > \mathbf{0}$. Suffices to prove that $z \in \mathbb{Z}^m$!
- Set $b := Bz$. Then $b = Bz = Bu + BB^{-1}v = Bu + v \in \mathbb{Z}^m$
- Note that $\binom{z}{\mathbf{0}} \in P_b$.
- Tight constraints for $\binom{z}{\mathbf{0}}$ have rank n:

$$A \rightarrow \begin{pmatrix} B & * \\ \mathbf{0} & -I_{n-m} \end{pmatrix} \begin{pmatrix} z \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} b \\ \mathbf{0} \end{pmatrix}$$

Unimodularity vs. integrality (3)

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

- Pick a $u \in \mathbb{Z}^m$ so that $z := u + B^{-1}v > \mathbf{0}$. Suffices to prove that $z \in \mathbb{Z}^m$!
- Set $b := Bz$. Then $b = Bz = Bu + BB^{-1}v = Bu + v \in \mathbb{Z}^m$
- Note that $\binom{z}{\mathbf{0}} \in P_b$.
- Tight constraints for $\binom{z}{\mathbf{0}}$ have rank n:

$$A \quad \underbrace{\begin{pmatrix} B & * \\ \mathbf{0} & -I_{n-m} \end{pmatrix}}_{\text{rank} = n} \begin{pmatrix} z \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} b \\ \mathbf{0} \end{pmatrix}$$

Unimodularity vs. integrality (3)

Claim 3. For each $v \in \mathbb{Z}^m$ one has $B^{-1}v \in \mathbb{Z}^m$

- Pick a $u \in \mathbb{Z}^m$ so that $z := u + B^{-1}v > \mathbf{0}$. Suffices to prove that $z \in \mathbb{Z}^m$!
- Set $b := Bz$. Then $b = Bz = Bu + BB^{-1}v = Bu + v \in \mathbb{Z}^m$
- Note that $\binom{z}{\mathbf{0}} \in P_b$.
- Tight constraints for $\binom{z}{\mathbf{0}}$ have rank n:

$$A \quad \underbrace{\begin{pmatrix} B & * \\ \mathbf{0} & -I_{n-m} \end{pmatrix}}_{\text{rank} = n} \begin{pmatrix} z \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} b \\ \mathbf{0} \end{pmatrix}$$

- Then $\binom{z}{\mathbf{0}}$ is a vertex of P_b
- By assumption $\binom{z}{\mathbf{0}}$ is integral. $\square$

Unimodularity vs. integrality (4)

Theorem (Hoffman-Kruskal Theorem)

Let $A \in \mathbb{Z}^{m\times n}$*. Then* A *is* $TU \Leftrightarrow \forall b \in \mathbb{Z}^m$,
$P_b = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq \mathbf{0}\}$ *is integer.*

- Suffices to prove "$\Leftarrow$"

Unimodularity vs. integrality (4)

Theorem (Hoffman-Kruskal Theorem)

Let $A \in \mathbb{Z}^{m \times n}$. Then A is TU $\Leftrightarrow \forall b \in \mathbb{Z}^m$, $P_b = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq \mathbf{0}\}$ is integer.

- Suffices to prove "$\Leftarrow$"
- Fix $Q_b := \{\binom{z}{s} \in \mathbb{R}^{n+m} \mid [A, I_m]\binom{z}{s} = b, \binom{z}{s} \geq \mathbf{0}\}$

Unimodularity vs. integrality (4)

Theorem (Hoffman-Kruskal Theorem)

Let $A \in \mathbb{Z}^{m \times n}$. Then A is TU $\Leftrightarrow \forall b \in \mathbb{Z}^m$, $P_b = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq \mathbf{0}\}$ is integer.

- Suffices to prove "$\Leftarrow$"
- Fix $Q_b := \{\binom{z}{s} \in \mathbb{R}^{n+m} \mid [A, I_m]\binom{z}{s} = b, \binom{z}{s} \geq \mathbf{0}\}$
- All vectors in Q_b are of the form $\binom{z}{b-Az}$

Claim. $(z, b - Az)$ vertex of $Q_b \Rightarrow z$ is vertex of P_b

Unimodularity vs. integrality (4)

Theorem (Hoffman-Kruskal Theorem)

Let $A \in \mathbb{Z}^{m \times n}$. Then A is TU $\Leftrightarrow \forall b \in \mathbb{Z}^m$, $P_b = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq \mathbf{0}\}$ is integer.

- Suffices to prove "$\Leftarrow$"
- Fix $Q_b := \{\binom{z}{s} \in \mathbb{R}^{n+m} \mid [A, I_m]\binom{z}{s} = b, \binom{z}{s} \geq \mathbf{0}\}$
- All vectors in Q_b are of the form $\binom{z}{b-Az}$
- Vertices of P_b integral $\Rightarrow$ vertices of Q_b integral

Claim. $(z, b - Az)$ vertex of $Q_b \Rightarrow z$ is vertex of P_b

- Suppose z not a vertex, i.e. $\exists u \neq \mathbf{0}$: $z + u, z - u \in P_b$. Then $(z + u, b - A(z + u)), (z - u, b - A(z - u)) \in Q_b$

Unimodularity vs. integrality (4)

Theorem (Hoffman-Kruskal Theorem)

Let $A \in \mathbb{Z}^{m \times n}$. Then A is TU $\Leftrightarrow \forall b \in \mathbb{Z}^m$, $P_b = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq \mathbf{0}\}$ is integer.

- Suffices to prove "$\Leftarrow$"
- Fix $Q_b := \{\binom{z}{s} \in \mathbb{R}^{n+m} \mid [A, I_m]\binom{z}{s} = b, \binom{z}{s} \geq \mathbf{0}\}$
- All vectors in Q_b are of the form $\binom{z}{b-Az}$
- Vertices of P_b integral $\Rightarrow$ vertices of Q_b integral
- Then $[A, I_m]$ unimodular

Claim. $(z, b - Az)$ vertex of $Q_b \Rightarrow z$ is vertex of P_b

- Suppose z not a vertex, i.e. $\exists u \neq \mathbf{0}$: $z + u, z - u \in P_b$.
  Then $(z + u, b - A(z + u)), (z - u, b - A(z - u)) \in Q_b$

Unimodularity vs. integrality (4)

Theorem (Hoffman-Kruskal Theorem)

Let $A \in \mathbb{Z}^{m \times n}$. Then A is TU $\Leftrightarrow \forall b \in \mathbb{Z}^m$, $P_b = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq \mathbf{0}\}$ is integer.

- Suffices to prove "$\Leftarrow$"
- Fix $Q_b := \{\binom{z}{s} \in \mathbb{R}^{n+m} \mid [A, I_m]\binom{z}{s} = b, \binom{z}{s} \geq \mathbf{0}\}$
- All vectors in Q_b are of the form $\binom{z}{b-Az}$
- Vertices of P_b integral $\Rightarrow$ vertices of Q_b integral
- Then $[A, I_m]$ unimodular
- Hence A TU

Claim. $(z, b - Az)$ vertex of $Q_b \Rightarrow z$ is vertex of P_b

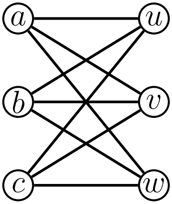
- Suppose z not a vertex, i.e. $\exists u \neq \mathbf{0}$: $z + u, z - u \in P_b$.
  Then $(z + u, b - A(z + u)), (z - u, b - A(z - u)) \in Q_b$

TU matrices from bipartite graphs

Definition

The **node-edge incidence matrix** of graph G is a matrix $A \in \{0,1\}^{V \times E}$ with

$$A_{v,e} = \begin{cases} 1 & v \text{ incident to } e \\ 0 & \text{otherwise} \end{cases}$$



graph G

edges

$$\textbf{nodes}\quad \begin{array}{c|ccccccccc|} & (a,u) & (a,v) & & & & & & & \\ a & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ b & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ c & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ u & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ v & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ w & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{array}$$

node edge incidence matrix A

TU matrices from bipartite graphs (2)

Theorem

Let $G = (V, E)$ be a graph with incidence matrix A_G. Then G is bipartite $\Leftrightarrow$ A_G is TU

TU matrices from bipartite graphs (2)

Theorem

Let $G = (V, E)$ be a graph with incidence matrix A_G. Then G is bipartite $\Leftrightarrow$ A_G is TU

Claim I. G not bipartite $\Rightarrow$ A_G not TU

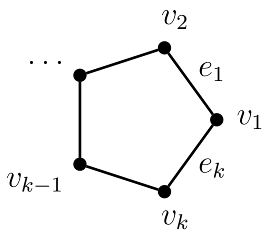
TU matrices from bipartite graphs (2)

Theorem

Let $G = (V, E)$ be a graph with incidence matrix A_G. Then G is bipartite $\Leftrightarrow$ A_G is TU

Claim I. G not bipartite $\Rightarrow$ A_G not TU

- Consider an odd cyle H with vertices $v_1, \ldots, v_k$ and edges $e_1, \ldots, e_k$. A_H is a square submatrix of A_G.



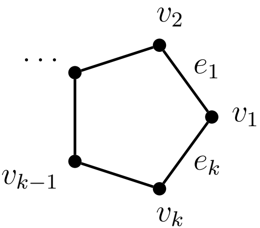
TU matrices from bipartite graphs (2)

Theorem

Let $G = (V, E)$ be a graph with incidence matrix A_G. Then G is bipartite $\Leftrightarrow$ A_G is TU

Claim I. G not bipartite $\Rightarrow$ A_G not TU

- Consider an odd cyle H with vertices $v_1, \dots, v_k$ and edges $e_1, \dots, e_k$. A_H is a square submatrix of A_G.
- After permuting rows and columns A_H is:



$$\begin{array}{c c} & \begin{array}{ccccc} e_1 & \cdots & & & e_k \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ \vdots \\ \\ v_k \end{array} & \begin{bmatrix} 1 & & & & 1 \\ 1 & 1 & & & \\ & 1 & 1 & & \\ & & 1 & 1 & \\ & & & 1 & 1 \end{bmatrix} \end{array} = A_H$$

- One may check that $|\det(B)| = 2$.

TU matrices from bipartite graphs (3)

Claim I. G bipartite $\Rightarrow A_G$ TU

TU matrices from bipartite graphs (3)

Claim I. G bipartite $\Rightarrow A_G$ TU

- Let B be a $k \times k$ submatrix of A_G. We prove by induction over k that $\det(B) \in \{-1, 0, 1\}$.

TU matrices from bipartite graphs (3)

Claim I. G bipartite $\Rightarrow A_G$ TU

- Let B be a $k \times k$ submatrix of A_G. We prove by induction over k that $\det(B) \in \{-1, 0, 1\}$.
- Case $k = 1$ is trivial. Consider $k \geq 2$. We do a case split!

TU matrices from bipartite graphs (3)

Claim I. G bipartite $\Rightarrow A_G$ TU

- Let B be a $k \times k$ submatrix of A_G. We prove by induction over k that $\det(B) \in \{-1, 0, 1\}$.
- Case $k = 1$ is trivial. Consider $k \geq 2$. We do a case split!

Case: B has a column with only 0's:

- Then $\det(B) = 0$

TU matrices from bipartite graphs (3)

Claim I. G bipartite $\Rightarrow A_G$ TU

- Let B be a $k \times k$ submatrix of A_G. We prove by induction over k that $\det(B) \in \{-1, 0, 1\}$.
- Case $k = 1$ is trivial. Consider $k \geq 2$. We do a case split!

Case: B has a column with only 0's:

- Then $\det(B) = 0$

Case: B has a column with exactly one 1:

- After permuting rows and columns,

$$B = \begin{pmatrix} 1 & * \\ \mathbf{0} & B' \end{pmatrix}$$

  with $\det(B') \in \{-1, 0, 1\}$ by IH.
- Then $\det(B) = 1 \cdot \det(B') \in \{-1, 0, 1\}$

TU matrices from bipartite graphs (4)

Case: Every column of B has exactly 2 ones

$$\begin{bmatrix} 1 & & 1 & \\ & 1 & & 1 \\ 1 & 1 & & 1 \\ & & 1 & \end{bmatrix}$$

TU matrices from bipartite graphs (4)

Case: Every column of B has exactly 2 ones

- Partition the row indices $I \cup J$ so that each column has exactly one row in I and one row in J (uses bipartiteness!)

$$\begin{array}{c} I \\ \\ J \\ \\ \end{array} \left[\begin{array}{cccc} 1 & & 1 & \\ & 1 & & 1 \\ \hdashline 1 & 1 & & 1 \\ & & 1 & \end{array}\right]$$

TU matrices from bipartite graphs (4)

Case: Every column of B has exactly 2 ones

- Partition the row indices $I \cup J$ so that each column has exactly one row in I and one row in J (uses bipartiteness!)

$$\begin{array}{c} I \\ \\ J \\ \\ \end{array} \left[\begin{array}{cccc} 1 & & 1 & \\ & 1 & & 1 \\ \hdashline 1 & 1 & & 1 \\ & & 1 & \end{array}\right]$$

- Then $\sum_{i \in I} B_i - \sum_{i \in J} B_j = \mathbf{0}$. Hence $\det(B) = 0$. $\square$

The matching polytope

Theorem

If G is bipartite, then $P_{matching}(G) = Q_{matching}(G)$ and $P_{perfectmatching}(G) = Q_{perfectmatching}(G)$.

$$Q_{\text{matching}}(G) \quad := \quad \left\{ x \in \mathbb{R}^E \mid \begin{array}{rcll} \sum_{e: v \in e} x_e & \leq & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\}$$

The matching polytope

Theorem

If G is bipartite, then $P_{matching}(G) = Q_{matching}(G)$ and $P_{perfectmatching}(G) = Q_{perfectmatching}(G)$.

$$\begin{aligned} Q_{\text{matching}}(G) &:= \left\{ x \in \mathbb{R}^E \mid \begin{array}{rcll} \sum_{e:v\in e} x_e & \leq & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\} \\ &= \left\{ x \in \mathbb{R}^E \mid \begin{pmatrix} A_G \\ -I_n \end{pmatrix} x \leq \begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix} \right\} \end{aligned}$$

Proof:

The matching polytope

Theorem

If G is bipartite, then $P_{matching}(G) = Q_{matching}(G)$ and $P_{perfectmatching}(G) = Q_{perfectmatching}(G)$.

$$
\begin{aligned}
Q_{\text{matching}}(G) &:= \left\{ x \in \mathbb{R}^E \mid \begin{array}{rcll} \sum_{e:v\in e} x_e & \leq & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\} \\
&= \left\{ x \in \mathbb{R}^E \mid \begin{pmatrix} A_G \\ -I_n \end{pmatrix} x \leq \begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix} \right\}
\end{aligned}
$$

Proof:

- $\binom{A_G}{-I_n}$ is TU. So every vertex of $Q_{\text{matching}}(G)$ is integral

The matching polytope

Theorem

If G is bipartite, then $P_{matching}(G) = Q_{matching}(G)$ and $P_{perfectmatching}(G) = Q_{perfectmatching}(G)$.

$$Q_{\text{matching}}(G) \;:=\; \left\{ x \in \mathbb{R}^E \mid \begin{array}{rcll} \sum_{e:v\in e} x_e & \leq & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \end{array} \right\}$$

$$= \left\{ x \in \mathbb{R}^E \mid \begin{pmatrix} A_G \\ -I_n \end{pmatrix} x \leq \begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix} \right\}$$

Proof:

- $\binom{A_G}{-I_n}$ is TU. So every vertex of $Q_{\text{matching}}(G)$ is integral
- Then

$$Q_{\text{matching}}(G) = \text{conv}\{Q_{\text{matching}} \cap \mathbb{Z}^E\} = P_{\text{matching}}(G)$$

- Similar for $Q_{\text{perfectmatching}}(G)$ $\square$

König's Theorem via TUness

Theorem (Kőnig's Theorem)

Let G be a bipartite graph. Then $\nu(G) = \tau(G)$.

König's Theorem via TUness

Theorem (Kőnig's Theorem)

Let G be a bipartite graph. Then $\nu(G) = \tau(G)$.

$$\nu(G) \;=\; \max\{\mathbf{1}^T x \mid A_G x \leq \mathbf{1},\; x \geq \mathbf{0}\}$$

König's Theorem via TUness

Theorem (Kőnig's Theorem)

Let G be a bipartite graph. Then $\nu(G) = \tau(G)$.

$$\begin{aligned} \nu(G) &= \max\{\mathbf{1}^T x \mid A_G x \le \mathbf{1},\ x \ge \mathbf{0}\} \\ &= \min\{y^T \mathbf{1} \mid y^T A_G \ge \mathbf{1},\ y \ge \mathbf{0}\} \end{aligned}$$

König's Theorem via TUness

Theorem (Kőnig's Theorem)

Let G be a bipartite graph. Then $\nu(G) = \tau(G)$.

$$\begin{aligned}
\nu(G) &= \max\{\mathbf{1}^T x \mid A_G x \leq \mathbf{1},\ x \geq \mathbf{0}\} \\
&= \min\{y^T \mathbf{1} \mid y^T A_G \geq \mathbf{1},\ y \geq \mathbf{0}\} \\
&= \min\{y^T \mathbf{1} \mid y^T A_G \geq \mathbf{1},\ y \in \mathbb{Z}^V_{\geq 0}\}
\end{aligned}$$

- Both LPs have optimum solutions that are integral

König's Theorem via TUness

Theorem (Kőnig's Theorem)

Let G be a bipartite graph. Then $\nu(G) = \tau(G)$.

$$\begin{aligned}
\nu(G) &= \max\{\mathbf{1}^T x \mid A_G x \leq \mathbf{1},\ x \geq \mathbf{0}\} \\
&= \min\{y^T \mathbf{1} \mid y^T A_G \geq \mathbf{1},\ y \geq \mathbf{0}\} \\
&= \min\{y^T \mathbf{1} \mid y^T A_G \geq \mathbf{1},\ y \in \mathbb{Z}^V_{\geq 0}\} \\
&= \min\left\{\sum_{i \in V} y_i \mid y^T A_G \geq \mathbf{1},\ y \in \{0,1\}^V\right\}
\end{aligned}$$

- Both LPs have optimum solutions that are integral
- Optimum solution would have $y_i \leq 1$

König's Theorem via TUness

Theorem (Kőnig's Theorem)

Let G be a bipartite graph. Then $\nu(G) = \tau(G)$.

$$
\begin{aligned}
\nu(G) &= \max\{\mathbf{1}^T x \mid A_G x \leq \mathbf{1},\ x \geq \mathbf{0}\} \\
&= \min\{y^T \mathbf{1} \mid y^T A_G \geq \mathbf{1},\ y \geq \mathbf{0}\} \\
&= \min\{y^T \mathbf{1} \mid y^T A_G \geq \mathbf{1},\ y \in \mathbb{Z}^V_{\geq 0}\} \\
&= \min\left\{\sum_{i \in V} y_i \mid y^T A_G \geq \mathbf{1},\ y \in \{0,1\}^V\right\} = \tau(G)
\end{aligned}
$$

- Both LPs have optimum solutions that are integral
- Optimum solution would have $y_i \leq 1$
- Problem selects minimum number of rows of A_G to cover $\mathbf{1} \in \mathbb{R}^E \rightarrow$ min. vertex cover $\square$

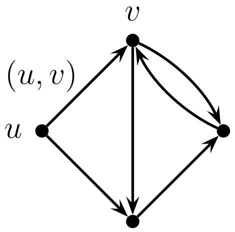
Lecture 10

Chapter 4 — Flows and circulations — Part 1/3

Directed graphs

Definition

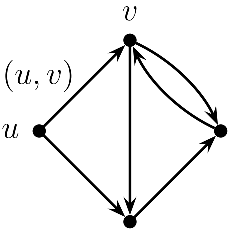
A **directed graph** $D = (V, A)$ is a pair where V is finite and A consists of pairs (u, v) with $u, v \in V$ and $u \neq v$.

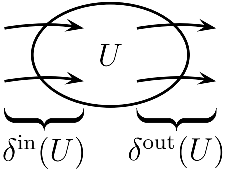


Directed graphs

Definition

A **directed graph** $D = (V, A)$ is a pair where V is finite and A consists of pairs (u, v) with $u, v \in V$ and $u \neq v$.





Define cuts

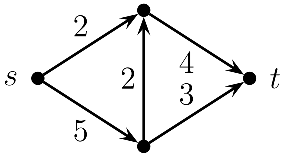
$$\delta^{\text{in}}(U) := \{(u, v) \in A \mid u \notin U, v \in U\}$$
$$\delta^{\text{out}}(U) := \{(u, v) \in A \mid u \in U, v \notin U\}$$

Flows in networks

Definition

Let $D = (V, A)$ be a directed graph with $s, t \in V$. A function $f : A \to \mathbb{R}$ is an *s-t* **flow** if

- $f(a) \geq 0 \;\; \forall a \in A$
- $\sum_{a \in \delta^{\text{in}}(v)} f(a) = \sum_{a \in \delta^{\text{out}}(v)} f(a) \;\; \forall v \in V \setminus \{s, t\}$



Flows in networks

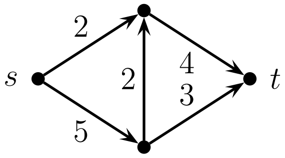
Definition

Let $D = (V, A)$ be a directed graph with $s, t \in V$. A function $f : A \to \mathbb{R}$ is an *s-t* **flow** if

- $f(a) \geq 0 \ \ \forall a \in A$
- $\sum_{a \in \delta^{\text{in}}(v)} f(a) = \sum_{a \in \delta^{\text{out}}(v)} f(a) \ \ \forall v \in V \setminus \{s, t\}$

The **value** of a flow is the net amount of flow leaving s (=net amount of flow entering t):

$$\text{value}(f) := \sum_{a \in \delta^{\text{out}}(s)} f(a) - \sum_{a \in \delta^{\text{in}}(s)} f(a)$$



Flows in networks (2)

- Given a function $c : A \to \mathbb{R}_{\geq 0}$, a flow f is a **flow under** c if $f(a) \leq c(a)\ \forall a \in A$.

Flows in networks (2)

- Given a function $c : A \to \mathbb{R}_{\geq 0}$, a flow f is a **flow under** c if $f(a) \leq c(a)$ $\forall a \in A$.
- **MaxFlow problem:** Given $D = (V, A)$, $s, t \in V$ and $c : A \to \mathbb{R}_{\geq 0}$, find an s-t flow f under c of maximum value.

Flows in networks (2)

- Given a function $c : A \to \mathbb{R}_{\geq 0}$, a flow f is a **flow under** c if $f(a) \leq c(a)\ \forall a \in A$.
- **MaxFlow problem:** Given $D = (V, A)$, $s, t \in V$ and $c : A \to \mathbb{R}_{\geq 0}$, find an s-t flow f under c of maximum value.
- For $U \subseteq V$, $\delta^{\text{out}}(U)$ is an **s-t cut** if $s \in U$, $t \notin U$

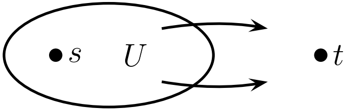
Flows in networks (2)

- Given a function $c : A \to \mathbb{R}_{\geq 0}$, a flow f is a **flow under c** if $f(a) \leq c(a)\ \forall a \in A$.
- **MaxFlow problem:** Given $D = (V, A)$, $s, t \in V$ and $c : A \to \mathbb{R}_{\geq 0}$, find an s-t flow f under c of maximum value.
- For $U \subseteq V$, $\delta^{\text{out}}(U)$ is an **s-t cut** if $s \in U$, $t \notin U$
- We call $c(\delta^{\text{out}}(U)) := \sum_{a \in \delta^{\text{out}}(U)} c(a)$ the **capacity** of a cut

Flows vs cuts

Lemma

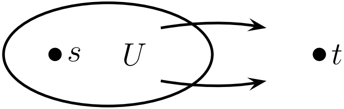
Let f be an s-t flow under c and let $\delta^{out}(U)$ be an s-t cut. Then $value(f) \leq c(\delta^{out}(U))$.



Flows vs cuts

Lemma

Let f be an s-t flow under c and let $\delta^{out}(U)$ be an s-t cut. Then value$(f) \leq c(\delta^{out}(U))$.



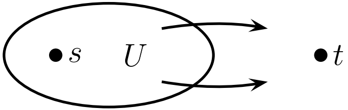
Proof.

$$\text{value}(f) \;=\; \sum_{a\in\delta^{\text{out}}(s)} f(a) - \sum_{a\in\delta^{\text{in}}(s)} f(a)$$

Flows vs cuts

Lemma

Let f be an s-t flow under c and let $\delta^{out}(U)$ be an s-t cut. Then $value(f) \leq c(\delta^{out}(U))$.



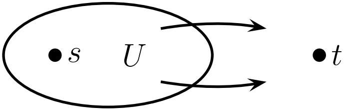
Proof.

$$\begin{aligned}
\text{value}(f) &= \sum_{a \in \delta^{\text{out}}(s)} f(a) - \sum_{a \in \delta^{\text{in}}(s)} f(a) \\
&= \sum_{v \in U} \Big(\sum_{a \in \delta^{\text{out}}(v)} f(a) - \sum_{a \in \delta^{\text{in}}(v)} f(a) \Big)
\end{aligned}$$

Flows vs cuts

Lemma

Let f be an s-t flow under c and let $\delta^{out}(U)$ be an s-t cut. Then $value(f) \leq c(\delta^{out}(U))$.



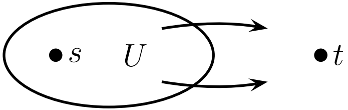
Proof.

$$\begin{aligned}
\text{value}(f) &= \sum_{a\in\delta^{\text{out}}(s)} f(a) - \sum_{a\in\delta^{\text{in}}(s)} f(a) \\
&= \sum_{v\in U}\Big(\sum_{a\in\delta^{\text{out}}(v)} f(a) - \sum_{a\in\delta^{\text{in}}(v)} f(a)\Big) \\
&= \sum_{a\in\delta^{\text{out}}(U)} \underbrace{f(a)}_{\leq c(a)\ (*)} - \sum_{a\in\delta^{\text{in}}(U)} \underbrace{f(a)}_{\geq 0\ (**)}
\end{aligned}$$

Flows vs cuts

Lemma

Let f be an s-t flow under c and let $\delta^{out}(U)$ be an s-t cut. Then $value(f) \leq c(\delta^{out}(U))$.



Proof.

$$\begin{aligned}
\text{value}(f) &= \sum_{a\in\delta^{\text{out}}(s)} f(a) - \sum_{a\in\delta^{\text{in}}(s)} f(a) \\
&= \sum_{v\in U}\Big(\sum_{a\in\delta^{\text{out}}(v)} f(a) - \sum_{a\in\delta^{\text{in}}(v)} f(a)\Big) \\
&= \sum_{a\in\delta^{\text{out}}(U)} \underbrace{f(a)}_{\leq c(a)\ (*)} - \sum_{a\in\delta^{\text{in}}(U)} \underbrace{f(a)}_{\geq 0\ (**)} \leq \sum_{a\in\delta^{\text{out}}(U)} c(a)
\end{aligned}$$

- **Note:** We have equality iff (*) and (**) are equalities!

Residual graphs

- For an edge $a = (u, v)$ we denote $a^{-1} := (v, u)$ as the inverse edge.

Definition

Let $f : A \to \mathbb{R}$ be an s-t flow under c. The **residual graph** is the graph $D_f = (V, A_f)$ with

$$\begin{aligned} f(a) < c(a) \quad &\Rightarrow \quad a \in A_f \\ f(a) > 0 \quad &\Rightarrow \quad a^{-1} \in A_f \end{aligned}$$

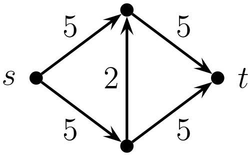
Residual graphs

- For an edge $a = (u, v)$ we denote $a^{-1} := (v, u)$ as the inverse edge.

Definition

Let $f : A \to \mathbb{R}$ be an s-t flow under c. The **residual graph** is the graph $D_f = (V, A_f)$ with

$$\begin{aligned} f(a) < c(a) \quad &\Rightarrow \quad a \in A_f \\ f(a) > 0 \quad &\Rightarrow \quad a^{-1} \in A_f \end{aligned}$$



capacities c

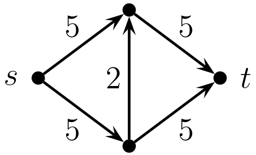
Residual graphs

- For an edge $a = (u, v)$ we denote $a^{-1} := (v, u)$ as the inverse edge.

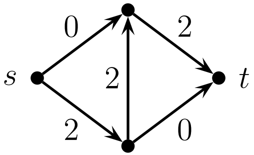
Definition

Let $f : A \to \mathbb{R}$ be an s-t flow under c. The **residual graph** is the graph $D_f = (V, A_f)$ with

$$\begin{aligned} f(a) < c(a) \quad &\Rightarrow \quad a \in A_f \\ f(a) > 0 \quad &\Rightarrow \quad a^{-1} \in A_f \end{aligned}$$



capacities c



flow f

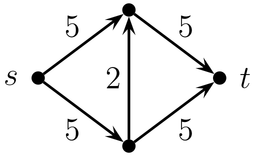
Residual graphs

- For an edge $a = (u, v)$ we denote $a^{-1} := (v, u)$ as the inverse edge.

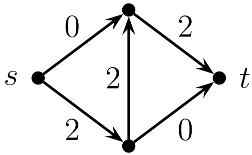
Definition

Let $f : A \to \mathbb{R}$ be an s-t flow under c. The **residual graph** is the graph $D_f = (V, A_f)$ with

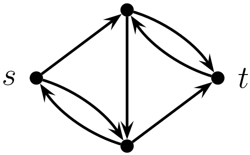
$$\begin{aligned} f(a) < c(a) \quad &\Rightarrow \quad a \in A_f \\ f(a) > 0 \quad &\Rightarrow \quad a^{-1} \in A_f \end{aligned}$$



capacities c



flow f



graph D_f

Flow augmentation

- Consider $D = (V, A)$, flow f under c

Flow augmentation

- Consider $D = (V, A)$, flow f under c
- Suppose there is an s-t path P in D_f with edges $a_1, \ldots, a_k$

Flow augmentation

- Consider $D = (V, A)$, flow f under c
- Suppose there is an s-t path P in D_f with edges $a_1, \ldots, a_k$
- Define
$$\sigma_i := \begin{cases} c(a_i) - f(a_i) > 0 & \text{if } a_i \in A \\ f(a_i^{-1}) & \text{if } a_i^{-1} \in A \end{cases}$$
- Define $\alpha := \min\{\sigma_1, \ldots, \sigma_k\} > 0$

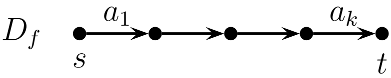
Flow augmentation

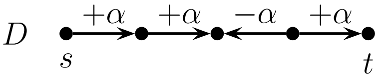
- Consider $D = (V, A)$, flow f under c
- Suppose there is an s-t path P in D_f with edges $a_1, \ldots, a_k$
- Define

$$\sigma_i := \begin{cases} c(a_i) - f(a_i) > 0 & \text{if } a_i \in A \\ f(a_i^{-1}) & \text{if } a_i^{-1} \in A \end{cases}$$

- Define $\alpha := \min\{\sigma_1, \ldots, \sigma_k\} > 0$
- Define $f' : A \to \mathbb{R}_{\geq 0}$ with

$$f'(a) := \begin{cases} f(a) + \alpha & \text{if } a = a_i \text{ for some } i \\ f(a) - \alpha & \text{if } a = a_i^{-1} \text{ for some } i \\ f(a) & \text{otherwise} \end{cases}$$





Lemma

f' is an s-t flow under c with value(f') = value$(f) + \alpha$.

MaxFlow=MinCut

Theorem (MaxFlow=MinCut; Ford Fulkerson 1956)

For any $D = (V, A)$ and $c : A \to \mathbb{R}_{\geq 0}$,

$$\max\{\mathit{value}(f) \mid f \text{ under } c\} = \min\{c(\delta^{out}(U)) \mid \{s\} \subseteq U \subseteq V \setminus \{t\}\}$$

MaxFlow=MinCut

Theorem (MaxFlow=MinCut; Ford Fulkerson 1956)

For any $D = (V, A)$ *and* $c : A \to \mathbb{R}_{\geq 0}$,

$$\max\{value(f) \mid f \text{ under } c\} = \min\{c(\delta^{out}(U)) \mid \{s\} \subseteq U \subseteq V \backslash \{t\}\}$$

Proof:

- Fix a flow f of maximum value.

MaxFlow=MinCut

Theorem (MaxFlow=MinCut; Ford Fulkerson 1956)

For any $D = (V, A)$ *and* $c : A \to \mathbb{R}_{\geq 0}$,

$$\max\{\mathit{value}(f) \mid f \ \mathit{under}\ c\} = \min\{c(\delta^{out}(U)) \mid \{s\} \subseteq U \subseteq V \setminus \{t\}\}$$

Proof:

- Fix a flow f of maximum value.

Case 1: There is an s-t path in D_f.

- We can augment f along some s-t path to some flow f' with value$(f) >$ value(f'). Contradiction!

MaxFlow=MinCut

Theorem (MaxFlow=MinCut; Ford Fulkerson 1956)

For any $D = (V, A)$ *and* $c : A \to \mathbb{R}_{\geq 0}$,

$$\max\{\mathit{value}(f) \mid f \ \mathit{under}\ c\} = \min\{c(\delta^{out}(U)) \mid \{s\} \subseteq U \subseteq V \setminus \{t\}\}$$

Proof:

- Fix a flow f of maximum value.

Case 1: There is an s-t path in D_f.

- We can augment f along some s-t path to some flow f' with value$(f) >$ value(f'). Contradiction!

Case 2: There is no s-t path in D_f.

MaxFlow=MinCut

Theorem (MaxFlow=MinCut; Ford Fulkerson 1956)

For any $D = (V, A)$ *and* $c : A \to \mathbb{R}_{\geq 0}$,

$$\max\{value(f) \mid f \text{ } under \text{ } c\} = \min\{c(\delta^{out}(U)) \mid \{s\} \subseteq U \subseteq V \setminus \{t\}\}$$

Proof:

- Fix a flow f of maximum value.

Case 1: There is an s-t path in D_f.

- We can augment f along some s-t path to some flow f' with value$(f) >$ value(f'). Contradiction!

Case 2: There is no s-t path in D_f.

- Let $U := \{u \in V \mid \exists \text{path in } D_f \text{ from } s \text{ to } u\}$
- Then $\delta^{\text{out}}(U)$ is an s-t cut.

MaxFlow=MinCut

Theorem (MaxFlow=MinCut; Ford Fulkerson 1956)

For any $D = (V, A)$ *and* $c : A \to \mathbb{R}_{\geq 0}$,

$$\max\{value(f) \mid f \ \ under\ c\} = \min\{c(\delta^{out}(U)) \mid \{s\} \subseteq U \subseteq V \setminus \{t\}\}$$

Proof:

- Fix a flow f of maximum value.

Case 1: There is an s-t path in D_f.

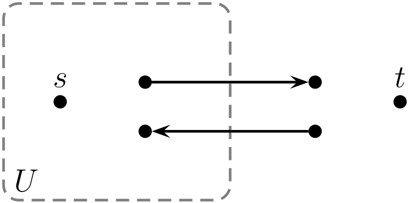
- We can augment f along some s-t path to some flow f' with value$(f) >$ value(f'). Contradiction!

Case 2: There is no s-t path in D_f.

- Let $U := \{u \in V \mid \exists \text{path in } D_f \text{ from } s \text{ to } u\}$
- Then $\delta^{\text{out}}(U)$ is an s-t cut.
- **Claim:** value$(f) = c(\delta^{\text{out}}(U))$

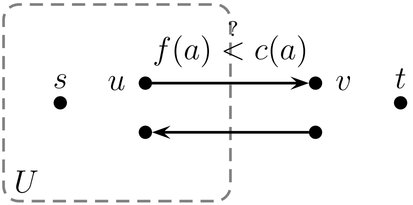
MaxFlow=MinCut (2)

- Recall: $U := \{u \in V \mid \exists \text{path in } D_f \text{ from } s \text{ to } u\}$
- We verify that $(*)$ and $(**)$ are tight:
  - **An outgoing edge** $a = (u, v) \in \delta^{\mathbf{out}}(U)$ **has** $f(a) = c(a)$. Otherwise $f(a) < c(a)$, $a \in A_f$ and v would be reachable via u.



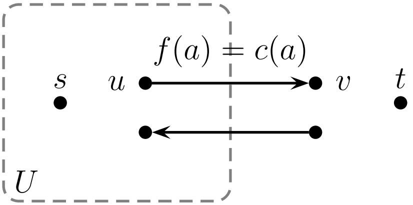
MaxFlow=MinCut (2)

- Recall: $U := \{u \in V \mid \exists \text{path in } D_f \text{ from } s \text{ to } u\}$
- We verify that $(*)$ and $(**)$ are tight:
  - **An outgoing edge** $a = (u, v) \in \delta^{\mathbf{out}}(U)$ **has** $f(a) = c(a)$. Otherwise $f(a) < c(a)$, $a \in A_f$ and v would be reachable via u.



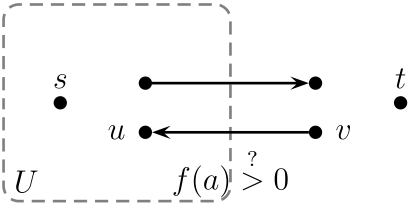
MaxFlow=MinCut (2)

- Recall: $U := \{u \in V \mid \exists \text{path in } D_f \text{ from } s \text{ to } u\}$
- We verify that $(*)$ and $(**)$ are tight:
  - **An outgoing edge $a = (u, v) \in \delta^{\mathbf{out}}(U)$ has** $f(a) = c(a)$. Otherwise $f(a) < c(a)$, $a \in A_f$ and v would be reachable via u.



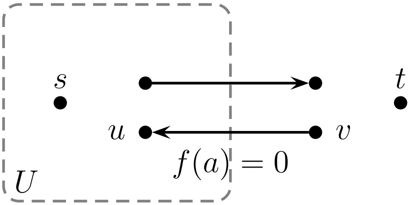
MaxFlow=MinCut (2)

- Recall: $U := \{u \in V \mid \exists \text{path in } D_f \text{ from } s \text{ to } u\}$
- We verify that $(*)$ and $(**)$ are tight:
  - **An outgoing edge** $a = (u, v) \in \delta^{\mathbf{out}}(U)$ **has** $f(a) = c(a)$. Otherwise $f(a) < c(a)$, $a \in A_f$ and v would be reachable via u.
  - **An incoming edge** $a = (v, u) \in \delta^{\mathbf{in}}(U)$ **has** $f(a) = 0$. If $f(a) > 0$, then $a^{-1} \in A_f$ and again v was reachable via u $\square$



MaxFlow=MinCut (2)

- Recall: $U := \{u \in V \mid \exists \text{path in } D_f \text{ from } s \text{ to } u\}$
- We verify that $(*)$ and $(**)$ are tight:
  - **An outgoing edge $a = (u, v) \in \delta^{\mathbf{out}}(U)$ has** $f(a) = c(a)$. Otherwise $f(a) < c(a)$, $a \in A_f$ and v would be reachable via u.
  - **An incoming edge $a = (v, u) \in \delta^{\mathbf{in}}(U)$ has $f(a) = 0$.** If $f(a) > 0$, then $a^{-1} \in A_f$ and again v was reachable via u $\square$

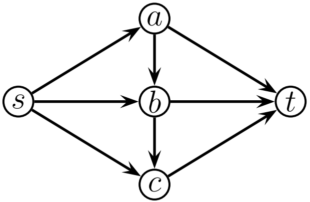


The incidence matrix of a directed graph

Definition

The **(node-edge) incidence matrix** of a directed graph $D = (V, A)$ is the matrix $A \in \{-1, 0, 1\}^{V \times A}$ defined by

$$A_{v,a} = \begin{cases} -1 & a \in \delta^{\text{in}}(v) \\ 1 & e \in \delta^{\text{out}}(v) \\ 0 & \text{otherwise.} \end{cases}$$



graph G

edges

nodes	(s,a)	(s,b)	(s,c)	(a,b)	(b,c)	(a,t)	(b,t)	(c,t)
s	1	1	1	0	0	0	0	0
a	-1	0	0	1	0	1	0	0
b	0	-1	0	-1	1	0	1	0
c	0	0	-1	0	-1	0	0	1
t	0	0	0	0	0	-1	-1	-1

The incidence matrix of a directed graph (2)

Lemma

The node edge incidence matrix M of any directed graph $D = (V, A)$ is TU.

Proof:

- Almost same as for incidence matrix of bipartite undirected graphs.

The incidence matrix of a directed graph (2)

Lemma

The node edge incidence matrix M of any directed graph $D = (V, A)$ is TU.

Proof:

- Almost same as for incidence matrix of bipartite undirected graphs.

Notation:

- For $U \subseteq V$, let M_U be submatrix with rows indexed by $u \in U$.

MaxFlow=MinCut

Alternative proof for the MaxFlow=MinCut Theorem:

$$\max\{\text{value}(f) \mid f \text{ flow under } c\}$$

MaxFlow=MinCut

Alternative proof for the MaxFlow=MinCut Theorem:

$$\begin{aligned} & \max\{\text{value}(f) \mid f \text{ flow under } c\} \\ = \; & \max\{M_s x \mid \mathbf{0} \leq x \leq c, M_{V\setminus\{s,v\}} x = \mathbf{0}, x \in \mathbb{R}^E\} \end{aligned}$$

MaxFlow=MinCut

Alternative proof for the MaxFlow=MinCut Theorem:

$$
\begin{aligned}
& \max\{\text{value}(f) \mid f \text{ flow under } c\} \\
= \; & \max\{M_s x \mid \mathbf{0} \le x \le c, M_{V\setminus\{s,v\}} x = \mathbf{0}, x \in \mathbb{R}^E\} \\
= \; & \min\{y^T c \mid y^T + z^T M_{V\setminus\{s,t\}} \ge M_s^T,\ y \in \mathbb{R}^E_{\ge 0},\ z \in \mathbb{R}^{V\setminus\{s,t\}}\}
\end{aligned}
$$

- Dual must have integral optimum

MaxFlow=MinCut

Alternative proof for the MaxFlow=MinCut Theorem:

$$
\begin{aligned}
& \max\{\text{value}(f) \mid f \text{ flow under } c\} \\
= \; & \max\{M_s x \mid \mathbf{0} \leq x \leq c, M_{V\setminus\{s,v\}} x = \mathbf{0}, x \in \mathbb{R}^E\} \\
= \; & \min\{y^T c \mid y^T + z^T M_{V\setminus\{s,t\}} \geq M_s^T,\ y \in \mathbb{R}^E_{\geq 0},\ z \in \mathbb{R}^{V\setminus\{s,t\}}\} \\
= \; & \text{...(some work)...} \\
= \; & \min\{c(\delta^{\text{out}}(U)) \mid \{s\} \subseteq U \subseteq V \setminus \{t\}\}
\end{aligned}
$$

- Dual must have integral optimum

The Ford-Fulkerson Algorithm

Ford and Fulkerson's algorithm for Max Flows

Input: $D = (V, A)$, $s, t \in V$, $c : A \to \mathbb{R}_{\geq 0}$.
Output: A maximum s-t-flow under c

(1) Set $f(a) = 0$ for all $a \in A$.

(2) REPEAT

   (3) Find an s-t path $P = (a_1, \ldots, a_k)$ in D_f. If none exists then stop.

   (4) Set $\sigma_i := c(a_i) - f(a_i)$ if $a_i \in A$, $\sigma_i := f(a_i^{-1})$ if $a_i^{-1} \in A$

   (5) Compute $\alpha := \min\{\sigma_1, \ldots, \sigma_k\}$.

   (6) Augment f along P by α.

Finiteness of Ford Fulkerson

Theorem

Let $D = (V, A)$ and $c : A \to \mathbb{Q}_{\geq 0}$. Then Ford Fulkerson finds a maximum flow in finite time.

Finiteness of Ford Fulkerson

Theorem

Let $D = (V, A)$ and $c : A \to \mathbb{Q}_{\geq 0}$. Then Ford Fulkerson finds a maximum flow in finite time.

Proof:

- If algo terminates, it has an optimum solution (see proof of MaxFlow=MinCut)
- For some $K \in \mathbb{N}$, $K \cdot c(a) \in \mathbb{Z}$ for all $a \in A$
- Then value(f) increases in each iteration by at least $\frac{1}{K}$

Finiteness of Ford Fulkerson

Theorem

Let $D = (V, A)$ and $c : A \to \mathbb{Q}_{\geq 0}$. Then Ford Fulkerson finds a maximum flow in finite time.

Proof:

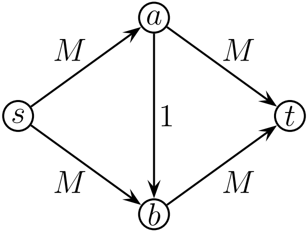
- If algo terminates, it has an optimum solution (see proof of MaxFlow=MinCut)
- For some $K \in \mathbb{N}$, $K \cdot c(a) \in \mathbb{Z}$ for all $a \in A$
- Then value(f) increases in each iteration by at least $\frac{1}{K}$

Remark:

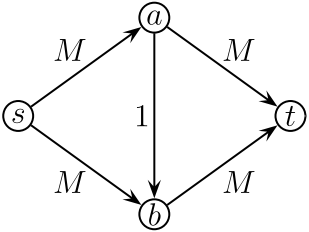
- If $c(a) \in \mathbb{R}$, then this is false (sequence of flows might not even converge)

A pathological instance for Ford Fulkerson

Example: Consider Ford-Fulkerson on this instance



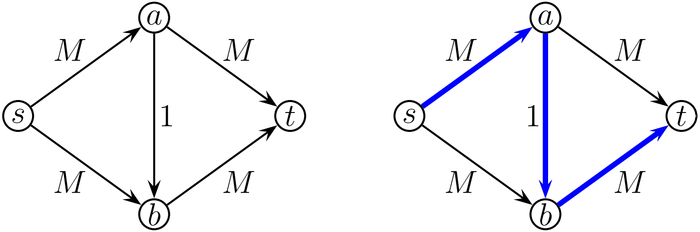
G with cap c



residual graph D_f
with residual cap.

A pathological instance for Ford Fulkerson

Example: Consider Ford-Fulkerson on this instance



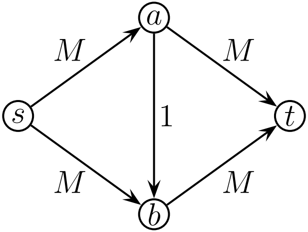
G with cap c

residual graph D_f

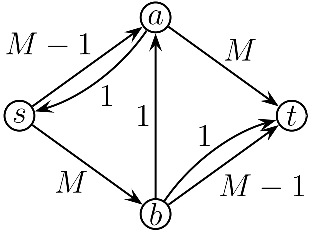
with residual cap.

A pathological instance for Ford Fulkerson

Example: Consider Ford-Fulkerson on this instance



G with cap c

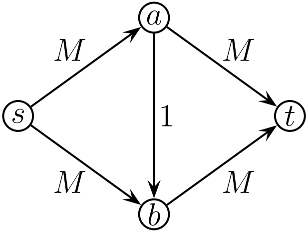


residual graph D_f

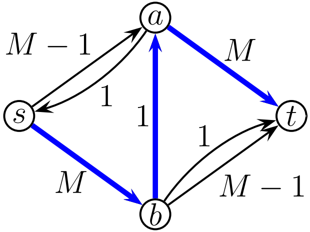
with residual cap.

A pathological instance for Ford Fulkerson

Example: Consider Ford-Fulkerson on this instance



G with cap c

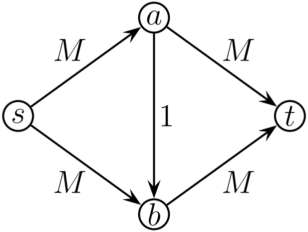


residual graph D_f

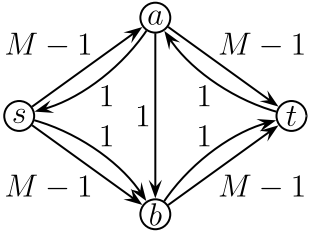
with residual cap.

A pathological instance for Ford Fulkerson

Example: Consider Ford-Fulkerson on this instance



G with cap c

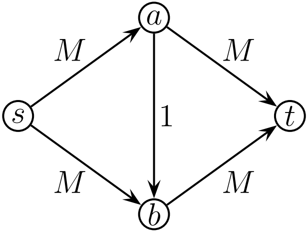


residual graph D_f

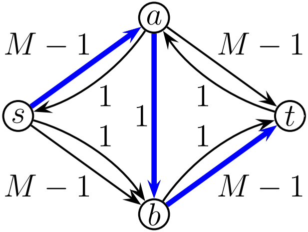
with residual cap.

A pathological instance for Ford Fulkerson

Example: Consider Ford-Fulkerson on this instance



G with cap c



residual graph D_f
with residual cap.

- Ford Fulkerson takes $2M$ iterations

Lecture 11

Chapter 4 — Flows and circulations — Part 2/3

The Transportation Problem

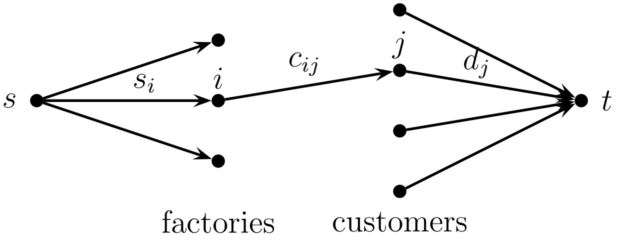
Transportation problem:

- m factories, n customers. Factory i can produce s_i tons of a product.
- Customer j needs d_j tons of the product.
- Factory i can transport at most c_{ij} tons to customer j
- **Question:** Can the demand be satisfied?

The Transportation Problem

Transportation problem:

- m factories, n customers. Factory i can produce s_i tons of a product.
- Customer j needs d_j tons of the product.
- Factory i can transport at most c_{ij} tons to customer j
- **Question:** Can the demand be satisfied?



- This is possible $\Leftrightarrow$ maxflow has value $\sum_{j=1}^{n} d_j$

Elimination of sports teams

- Consider a **sport league**. A team gets 1 point for a win, 0 points for loosing (no ties).
- We are in the middle of the season and wonder out beloved team can still become **champion** (=having the maximum number of points, possibly in a tie)

Comment: This is not in Schrijver's notes!

Elimination of sports teams

- Consider a **sport league**. A team gets 1 point for a win, 0 points for loosing (no ties).
- We are in the middle of the season and wonder out beloved team can still become **champion** (=having the maximum number of points, possibly in a tie)

Example 1:

Team	Wins	To play
A	33	8
B	28	4
$\vdots$		

Comment: This is not in Schrijver's notes!

Elimination of sports teams

- Consider a **sport league**. A team gets 1 point for a win, 0 points for loosing (no ties).
- We are in the middle of the season and wonder out beloved team can still become **champion** (=having the maximum number of points, possibly in a tie)

Example 1:

Team	Wins	To play
A	33	8
B	28	4
$\vdots$		

- Can team B still become champion?
  No! (A will dominate)

Comment: This is not in Schrijver's notes!

Elimination of sports teams (2)

Example 2:

Team	wins	To play	A	B	C	D
A	33	8	-	1	6	1
B	29	4	1	-	0	3
C	28	7	6	0	-	1
D	27	5	1	3	1	-

- Can team B still become chamption?

Elimination of sports teams (2)

Example 2:

Team	wins	To play	A	B	C	D
A	33	8	-	1	6	1
B	29	4	1	-	0	3
C	28	7	6	0	-	1
D	27	5	1	3	1	-

- Can team B still become chamption?
- **No.** B would need to win all games. A would have to loose all games. But then C wins at least another 6 games, getting $34 > 33$ points.

Elimination of sports teams (2)

Example 2:

Team	wins	To play	A	B	C	D
A	33	8	-	1	6	1
B	29	4	1	-	0	3
C	28	7	6	0	-	1
D	27	5	1	3	1	-

- Can team B still become chamption?
- **No.** B would need to win all games. A would have to loose all games. But then C wins at least another 6 games, getting $34 > 33$ points.

We prove: If B cannot become champion, there is a simple reason why not!

Elimination of sports teams (3)

Assumption: Our team B wins all remaining games!

Elimination of sports teams (3)

Assumption: Our team B wins all remaining games!

Some notation:

$$
\begin{array}{rcll}
M & := & \text{\#wins for } B \text{ at end of seasons} & \\
 & & \text{assuming they win all remaining games} & \\
T & := & \{\text{teams other than } B\} & \\
w_i & := & \text{\#wins of team } i \text{ currently} & \forall i \in T \\
r_{ij} & := & \text{\#games remaining between } i \text{ and } j & \forall i, j \in T, i \neq j \\
P & := & \{\{i, j\} \subseteq T \mid i \neq j, r_{ij} > 0\} &
\end{array}
$$

Elimination of sports teams (3)

Assumption: Our team B wins all remaining games!

Some notation:

$$
\begin{array}{rcl}
M & := & \#\text{wins for } B \text{ at end of seasons} \\
  &   & \text{assuming they win all remaining games} \\
T & := & \{\text{teams other than } B\} \\
w_i & := & \#\text{wins of team } i \text{ currently} \quad \forall i \in T \\
r_{ij} & := & \#\text{games remaining between } i \text{ and } j \quad \forall i,j \in T, i \neq j \\
P & := & \{\{i,j\} \subseteq T \mid i \neq j, r_{ij} > 0\}
\end{array}
$$

Goal: Find outcome of the games so that all teams $i \in T$ get $\leq M - w_i$ additional wins!

A simple elimination criterion

Lemma

If there is a subset $\tilde{A} \subseteq T$ with

$$\sum_{i \in \tilde{A}} w_i + \sum_{\{i,j\} \subseteq \tilde{A}, \{i,j\} \in P} r_{ij} > M \cdot |\tilde{A}|$$

then B cannot become champion!

A simple elimination criterion

Lemma

If there is a subset $\tilde{A} \subseteq T$ with

$$\sum_{i \in \tilde{A}} w_i + \sum_{\{i,j\} \subseteq \tilde{A}, \{i,j\} \in P} r_{ij} > M \cdot |\tilde{A}|$$

then B cannot become champion!

Proof:

- Note that

$$\begin{array}{c}\text{average \#points} \\ \text{of teams in } \tilde{A} \\ \text{at end of season}\end{array} \geq \frac{1}{|\tilde{A}|}\Bigg(\sum_{i \in \tilde{A}} w_i + \sum_{\{i,j\} \subseteq \tilde{A}, \{i,j\} \in P} r_{ij}\Bigg) > M$$

- No matter the outcome, some team will get $> M$ points

A simple elimination criterion (2)

Example 1:

Team	wins	To play
A	33	8
B	28	4
⋮		

A simple elimination criterion (2)

Example 1:

Team	wins	To play
A	33	8
B	28	4
$\vdots$		

- Elimination criterion with $\tilde{A} := \{A\}$ applies!

A simple elimination criterion (2)

Example 1:

Team	wins	To play
A	33	8
B	28	4
⋮		

- Elimination criterion with $\tilde{A} := \{A\}$ applies!

Example 2:

Team	wins	To play	A	B	C	D
A	33	8	-	1	6	1
B	29	4	1	-	0	3
C	28	7	6	0	-	1
D	27	5	1	3	1	-

A simple elimination criterion (2)

Example 1:

Team	wins	To play
A	33	8
B	28	4
$\vdots$		

- Elimination criterion with $\tilde{A} := \{A\}$ applies!

Example 2:

Team	wins	To play	A	B	C	D
A	33	8	-	1	6	1
B	29	4	1	-	0	3
C	28	7	6	0	-	1
D	27	5	1	3	1	-

- Elimination criterion with $\tilde{A} := \{A, C\}$ applies!

Equivalence of criterion

Theorem

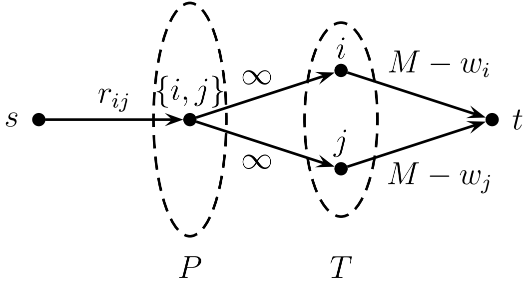
If team B cannot become champion, then there is a subset $\tilde{A} \subseteq T$ with $\sum_{i \in \tilde{A}} w_i + \sum_{\{i,j\} \subseteq \tilde{A}, \{i,j\} \in P} r_{ij} > M \cdot |\tilde{A}|$.

Equivalence of criterion

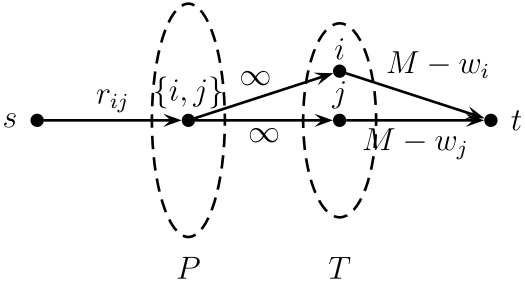
Theorem

If team B cannot become champion, then there is a subset $\tilde{A} \subseteq T$ with $\sum_{i\in\tilde{A}} w_i + \sum_{\{i,j\}\subseteq\tilde{A},\{i,j\}\in P} r_{ij} > M \cdot |\tilde{A}|$.

- Create a graph $D = (V, A)$ as with $V := T \cup P \cup \{s,t\}$:



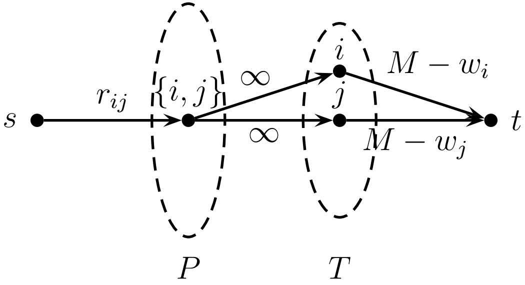
Equivalence of criterion(2)



- Then

$$\begin{aligned} & B \text{ can become champion} \\ \Leftrightarrow \quad & \exists \text{ integral flow of value } \textstyle\sum_{\{i,j\} \in P} r_{ij} \end{aligned}$$

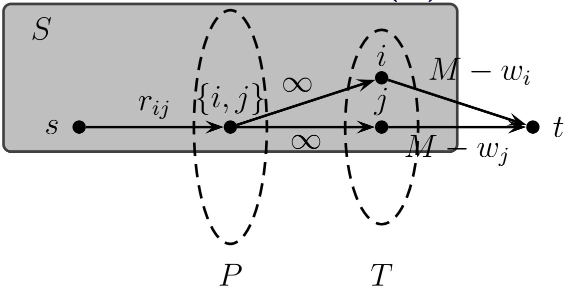
Equivalence of criterion(2)



- Then

$$\begin{aligned} & B \text{ can become champion} \\ \Leftrightarrow \quad & \exists \text{ integral flow of value } \textstyle\sum_{\{i,j\} \in P} r_{ij} \\ \Leftrightarrow \quad & \exists \text{ flow of value } \textstyle\sum_{\{i,j\} \in P} r_{ij} \end{aligned}$$

Equivalence of criterion(2)



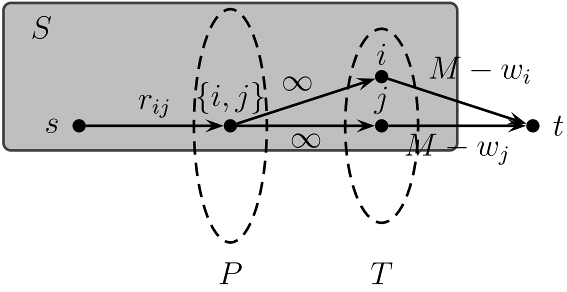
- Then

$$
\begin{aligned}
& B \text{ can become champion} \\
\Leftrightarrow \;& \exists \text{ integral flow of value } \textstyle\sum_{\{i,j\}\in P} r_{ij} \\
\Leftrightarrow \;& \exists \text{ flow of value } \textstyle\sum_{\{i,j\}\in P} r_{ij}
\end{aligned}
$$

- By **MaxFlow=MinCut Theorem**

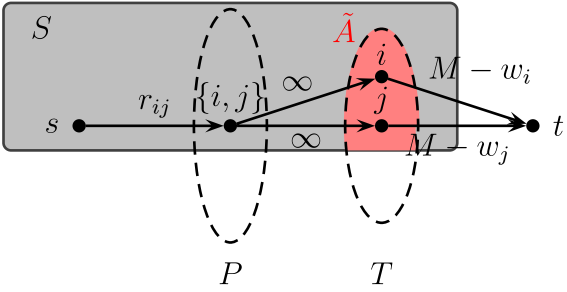
$$B \text{ cannot become champion } \Leftrightarrow \exists \text{ cut of value } < \sum_{\{i,j\}\in P} r_{ij}$$

Equivalence of criterion (3)



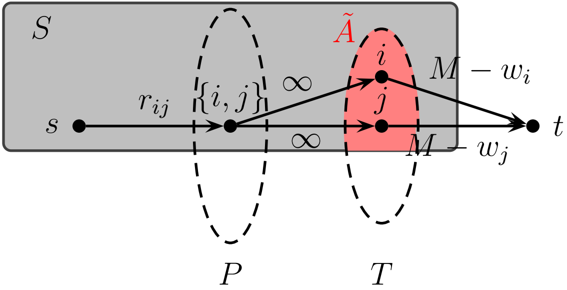
- Let S be a minimum cut with $c(\delta^{\text{out}}(S)) < \sum_{\{i,j\} \in P} r_{ij}$

Equivalence of criterion (3)



- Let S be a minimum cut with $c(\delta^{\text{out}}(S)) < \sum_{\{i,j\} \in P} r_{ij}$
- Set $\tilde{A} := S \cap T$

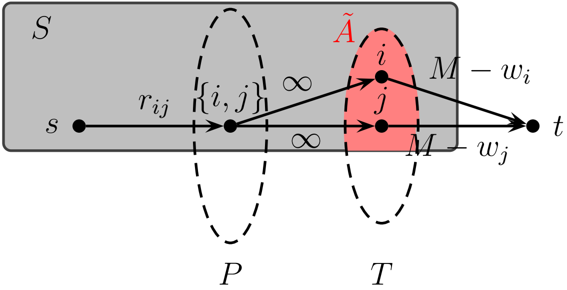
Equivalence of criterion (3)



- Let S be a minimum cut with $c(\delta^{\text{out}}(S)) < \sum_{\{i,j\} \in P} r_{ij}$
- Set $\tilde{A} := S \cap T$

Claim. $S = \{s\} \cup \{\{i,j\} \in P : |\{i,j\} \cap \tilde{A}| = 2\} \cup \tilde{A}$.

Equivalence of criterion (3)



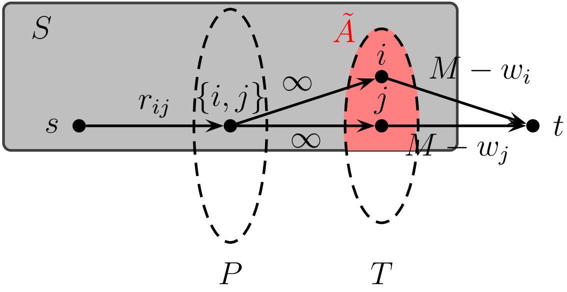
- Let S be a minimum cut with $c(\delta^{\text{out}}(S)) < \sum_{\{i,j\} \in P} r_{ij}$
- Set $\tilde{A} := S \cap T$

Claim. $S = \{s\} \cup \{\{i,j\} \in P : |\{i,j\} \cap \tilde{A}| = 2\} \cup \tilde{A}$.

Proof:

- If $\{i,j\} \in S, |\{i,j\} \cap \tilde{A}| < 2 \Rightarrow$ edge with cap ∞ is cut

Equivalence of criterion (3)



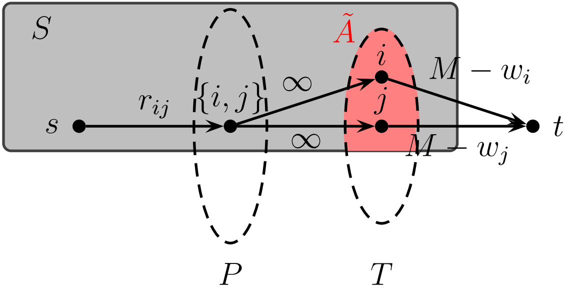
- Let S be a minimum cut with $c(\delta^{\text{out}}(S)) < \sum_{\{i,j\} \in P} r_{ij}$
- Set $\tilde{A} := S \cap T$

Claim. $S = \{s\} \cup \{\{i,j\} \in P : |\{i,j\} \cap \tilde{A}| = 2\} \cup \tilde{A}$.
Proof:

- If $\{i,j\} \in S, |\{i,j\} \cap \tilde{A}| < 2 \Rightarrow$ edge with cap ∞ is cut
- If $\{i,j\} \notin S$, $|\{i,j\} \cap \tilde{A}| = 2 \Rightarrow$ moving $\{i,j\}$ into S decreases cut value by r_{ij}

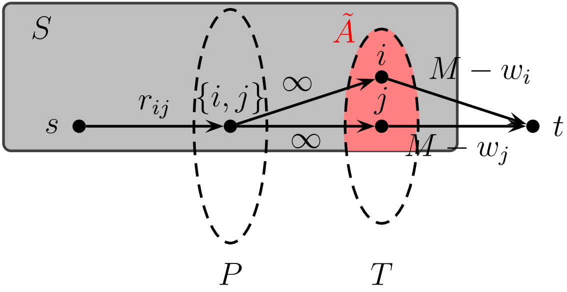
$\square$

Equivalence of criterion(4)



$$c(\delta^{\text{out}}(S)) \quad = \quad \sum_{\{i,j\}\in P : |\{i,j\}\cap\tilde{A}|\leq 1} r_{ij} + \sum_{i\in\tilde{A}}(M - w_i) < \sum_{\{i,j\}\in P} r_{ij}$$

Equivalence of criterion(4)



$$c(\delta^{\text{out}}(S)) = \sum_{\{i,j\} \in P : |\{i,j\} \cap \tilde{A}| \leq 1} r_{ij} + \sum_{i \in \tilde{A}} (M - w_i) < \sum_{\{i,j\} \in P} r_{ij}$$

$$\Rightarrow \quad M \cdot |\tilde{A}| < \sum_{i \in \tilde{A}} w_i + \sum_{\{i,j\} \in P : |\{i,j\} \cap \tilde{A}| = 2} r_{ij} \quad \square$$

Lecture 12

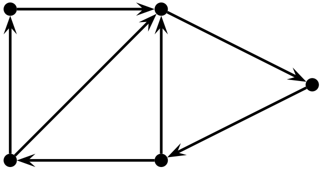
Chapter 4 — Flows and circulations — Part 3/3

Circulation

Definition

Let $D = (V, A)$ be a directed graph. A function $f : A \to \mathbb{R}$ is a **circulation** if

$$\sum_{a \in \delta^{\text{out}}(v)} f(a) = \sum_{a \in \delta^{\text{in}}(v)} f(a) \quad \forall v \in V$$

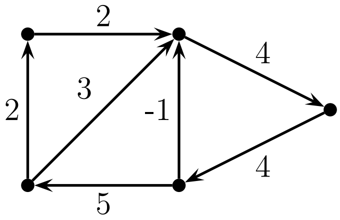


Circulation

Definition

Let $D = (V, A)$ be a directed graph. A function $f : A \to \mathbb{R}$ is a **circulation** if

$$\sum_{a \in \delta^{\text{out}}(v)} f(a) = \sum_{a \in \delta^{\text{in}}(v)} f(a) \quad \forall v \in V$$



Hoffman's Circulation Theorem

Theorem

Let $D = (V, A)$ be a directed graph and let $d, c : A \to \mathbb{R}$ with $d(a) \leq c(a)$ for all $a \in A$. The following is equivalent:

(A) *There exists a circulation $f : A \to \mathbb{R}$ with $d(a) \leq f(a) \leq c(a)$ for all $a \in A$.*

(B) *One has $d(\delta^{in}(U)) \leq c(\delta^{out}(U))$ for every $U \subseteq V$.*

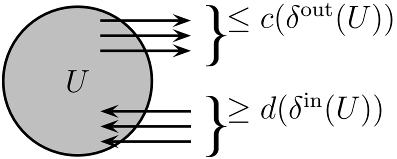
Hoffman's Circulation Theorem

Theorem

Let $D = (V, A)$ be a directed graph and let $d, c : A \to \mathbb{R}$ with $d(a) \leq c(a)$ for all $a \in A$. The following is equivalent:

(A) *There exists a circulation $f : A \to \mathbb{R}$ with $d(a) \leq f(a) \leq c(a)$ for all $a \in A$.*

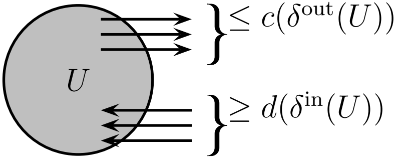
(B) *One has $d(\delta^{in}(U)) \leq c(\delta^{out}(U))$ for every $U \subseteq V$.*



Hoffman's Circulation Theorem (2)

Claim $(A) \Rightarrow (B)$. We have

$$d(\delta^{\text{in}}(U)) \leq f(\delta^{\text{in}}(U)) \overset{\text{circulation}}{=} f(\delta^{\text{out}}(U)) \leq c(\delta^{\text{out}}(U))$$



Hoffman's Circulation Theorem (2)

Claim $(B) \Rightarrow (A)$**.**

Hoffman's Circulation Theorem (2)

Claim $(B) \Rightarrow (A)$**.**

- For a function $f : A \to \mathbb{R}$ (not necessarily a circulation) define $\text{loss}_f \in \mathbb{R}^V$ with $\text{loss}_f(v) := f(\delta^{\text{out}}(v)) - f(\delta^{\text{in}}(v))$.

Hoffman's Circulation Theorem (2)

Claim $(B) \Rightarrow (A)$.

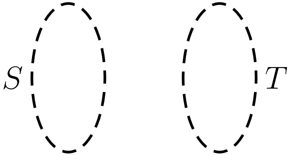
- For a function $f : A \to \mathbb{R}$ (not necessarily a circulation) define $\text{loss}_f \in \mathbb{R}^V$ with $\text{loss}_f(v) := f(\delta^{\text{out}}(v)) - f(\delta^{\text{in}}(v))$.
- Choose a function $f : A \to \mathbb{R}$ with $d \leq f \leq c$ min. $\|\text{loss}_f\|_1$

Hoffman's Circulation Theorem (2)

Claim $(B) \Rightarrow (A)$**.**

- For a function $f : A \to \mathbb{R}$ (not necessarily a circulation) define $\text{loss}_f \in \mathbb{R}^V$ with $\text{loss}_f(v) := f(\delta^{\text{out}}(v)) - f(\delta^{\text{in}}(v))$.
- Choose a function $f : A \to \mathbb{R}$ with $d \leq f \leq c$ min. $\|\text{loss}_f\|_1$
- If $\text{loss}_f = \mathbf{0}$, then f circulation. Suppose $\text{loss}_f \neq \mathbf{0}$.
- Set

$$S := \{v \in V \mid \text{loss}_f(v) < 0\} \quad \text{and} \quad T := \{v \in V \mid \text{loss}_f(v) > 0\}$$

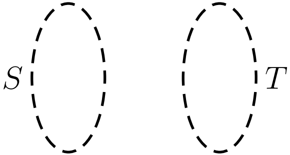


Hoffman's Circulation Theorem (2)

Claim $(B) \Rightarrow (A)$.

- For a function $f : A \to \mathbb{R}$ (not necessarily a circulation) define $\text{loss}_f \in \mathbb{R}^V$ with $\text{loss}_f(v) := f(\delta^{\text{out}}(v)) - f(\delta^{\text{in}}(v))$.
- Choose a function $f : A \to \mathbb{R}$ with $d \leq f \leq c$ min. $\|\text{loss}_f\|_1$
- If $\text{loss}_f = \mathbf{0}$, then f circulation. Suppose $\text{loss}_f \neq \mathbf{0}$.
- Set

$$S := \{v \in V \mid \text{loss}_f(v) < 0\} \quad \text{and} \quad T := \{v \in V \mid \text{loss}_f(v) > 0\}$$

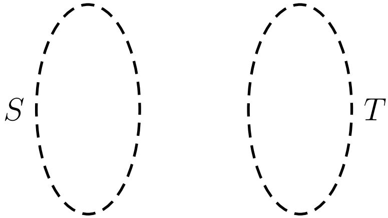


- We define **residual graph** is the graph $D_f = (V, A_f)$ with

$$\begin{aligned} f(a) < c(a) \quad &\Rightarrow \quad a \in A_f \\ f(a) > d(a) \quad &\Rightarrow \quad a^{-1} \in A_f \end{aligned}$$

Hoffman's Circulation Theorem (3)

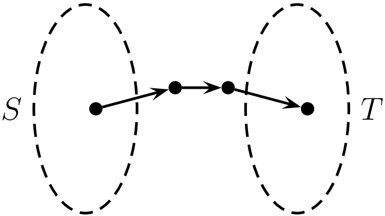
Case 1: There is a S-T path in D_f



Hoffman's Circulation Theorem (3)

Case 1: There is a S-T path in D_f

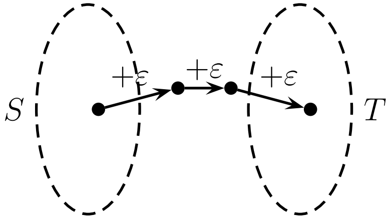
- Fix any S-T path P in D_f



Hoffman's Circulation Theorem (3)

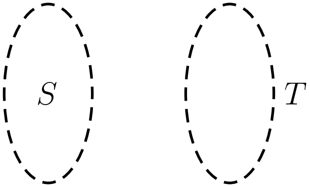
Case 1: There is a S-T path in D_f

- Fix any S-T path P in D_f
- Augment flow f along P by some by $\varepsilon > 0$. Then $\|\text{loss}_f\|_1$ decreases! Contradiction!



Hoffman's Circulation Theorem (4)

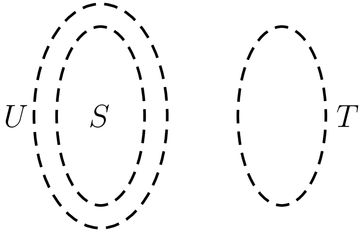
Case 2: There is no S-T path in D_f.



Hoffman's Circulation Theorem (4)

Case 2: There is no S-T path in D_f.

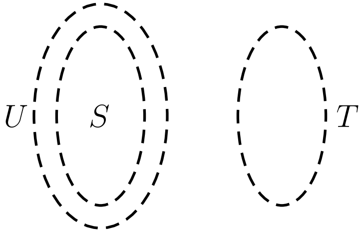
- Let $U := \{u \in V \mid u \text{ reachable from } S \text{ in } D_f\}$



Hoffman's Circulation Theorem (4)

Case 2: There is no S-T path in D_f.

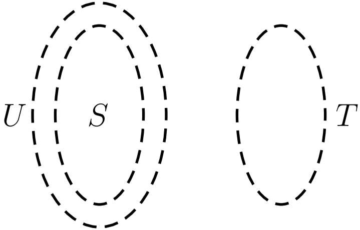
- Let $U := \{u \in V \mid u \text{ reachable from } S \text{ in } D_f\}$
- For each $a \in \delta^{\text{out}}(U)$, one has $f(a) = c(a)$
- For each $a \in \delta^{\text{in}}(U)$, one has $f(a) = d(a)$
  (same argument as in MaxFlow=MinCut)



Hoffman's Circulation Theorem (4)

Case 2: There is no S-T path in D_f.

- Let $U := \{u \in V \mid u \text{ reachable from } S \text{ in } D_f\}$
- For each $a \in \delta^{\text{out}}(U)$, one has $f(a) = c(a)$
- For each $a \in \delta^{\text{in}}(U)$, one has $f(a) = d(a)$
  (same argument as in MaxFlow=MinCut)



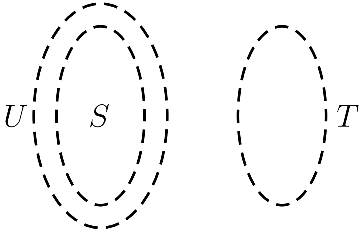
Then

$$0 \quad > \quad \text{loss}_f(S)$$

Hoffman's Circulation Theorem (4)

Case 2: There is no S-T path in D_f.

- Let $U := \{u \in V \mid u \text{ reachable from } S \text{ in } D_f\}$
- For each $a \in \delta^{\text{out}}(U)$, one has $f(a) = c(a)$
- For each $a \in \delta^{\text{in}}(U)$, one has $f(a) = d(a)$
  (same argument as in MaxFlow=MinCut)



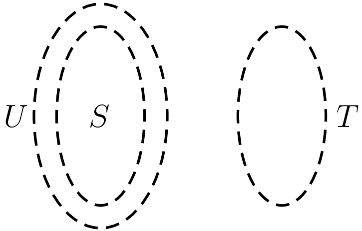
Then

$$0 \quad > \quad \text{loss}_f(S) = \text{loss}_f(U)$$

Hoffman's Circulation Theorem (4)

Case 2: There is no S-T path in D_f.

- Let $U := \{u \in V \mid u \text{ reachable from } S \text{ in } D_f\}$
- For each $a \in \delta^{\text{out}}(U)$, one has $f(a) = c(a)$
- For each $a \in \delta^{\text{in}}(U)$, one has $f(a) = d(a)$
  (same argument as in MaxFlow=MinCut)



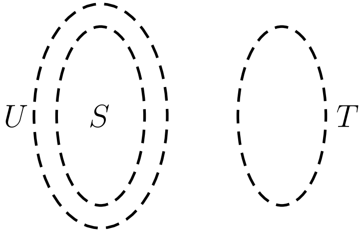
Then

$$0 \quad > \quad \text{loss}_f(S) = \text{loss}_f(U) = f(\delta^{\text{out}}(U)) - f(\delta^{\text{in}}(U))$$

Hoffman's Circulation Theorem (4)

Case 2: There is no S-T path in D_f.

- Let $U := \{u \in V \mid u \text{ reachable from } S \text{ in } D_f\}$
- For each $a \in \delta^{\text{out}}(U)$, one has $f(a) = c(a)$
- For each $a \in \delta^{\text{in}}(U)$, one has $f(a) = d(a)$
  (same argument as in MaxFlow=MinCut)



Then

$$\begin{aligned} 0 &> \text{loss}_f(S) = \text{loss}_f(U) = f(\delta^{\text{out}}(U)) - f(\delta^{\text{in}}(U)) \\ &= c(\delta^{\text{out}}(U)) - d(\delta^{\text{in}}(U)) \quad \square \end{aligned}$$

Min Cost Flows – Variant 1

- **Given:** Directed graph $D = (V, A)$, capacities $c : A \to \mathbb{R}_{\geq 0}$, cost $k : A \to \mathbb{R}$, demands $b : V \to \mathbb{R}$.
- **Goal:** Find a flow f under c, respecting demands b, minimizing $\text{cost}(f) := \sum_{a \in A} f(a) \cdot k(a)$

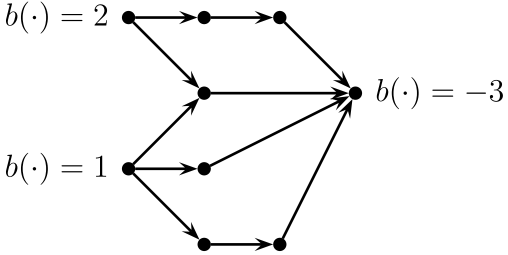
Min Cost Flows – Variant 1

- **Given:** Directed graph $D = (V, A)$, capacities $c : A \to \mathbb{R}_{\geq 0}$, cost $k : A \to \mathbb{R}$, demands $b : V \to \mathbb{R}$.
- **Goal:** Find a flow f under c, respecting demands b, minimizing $\mathrm{cost}(f) := \sum_{a \in A} f(a) \cdot k(a)$

$$\begin{array}{rcll} \min \sum_{a \in A} k(a) \cdot f(a) & & & (*) \\ \sum_{a \in \delta^{\mathrm{out}}(v)} f(a) - \sum_{a \in \delta^{\mathrm{in}}(v)} f(a) & = & b(v) & \forall v \in V \\ 0 \leq f(a) & \leq & c(a) & \forall a \in A \end{array}$$

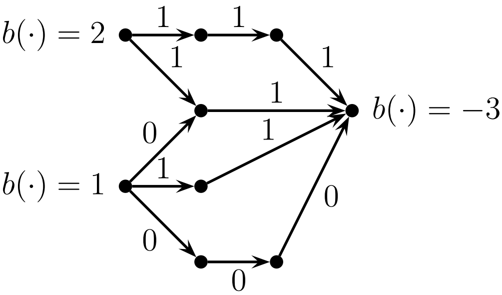
Min Cost Flows (2)

Example: All edges have $c(a) := 1$, $k(a) := 1$.



Min Cost Flows (2)

Example: All edges have $c(a) := 1$, $k(a) := 1$.



Note: Necessary that $\sum_{v \in V} b(v) = 0$

Min Cost Flows (2)

Observation: Min Cost Flow LP $(*)$ can be rewritten to

$$\begin{aligned} \min k^T f \\ \begin{pmatrix} M & \mathbf{0} \\ \boldsymbol{I}_A & \boldsymbol{I}_A \end{pmatrix} \begin{pmatrix} f \\ s \end{pmatrix} &= \begin{pmatrix} b \\ c \end{pmatrix} \\ (f, s) &\geq \mathbf{0} \end{aligned}$$

- Here M be the node edge incident matrix for D

Min Cost Flows (2)

Observation: Min Cost Flow LP $(*)$ can be rewritten to

$$\begin{aligned} \min k^T f & \\ \begin{pmatrix} M & \mathbf{0} \\ \boldsymbol{I}_A & \boldsymbol{I}_A \end{pmatrix} \begin{pmatrix} f \\ s \end{pmatrix} &= \begin{pmatrix} b \\ c \end{pmatrix} \\ (f, s) &\geq \mathbf{0} \end{aligned}$$

- Here M be the node edge incident matrix for D
- M is TU $\Rightarrow$ constraint matrix of $(*)$ is TU
- If b and c are integral, then there is integral optimum

Application 1 - Shortest path

Application: Shortest s-t path in D

- Shortest path problem can be modeled as min cost flow:

$$\min \sum_{a \in A} k(a) \cdot f(a) \qquad (*)$$

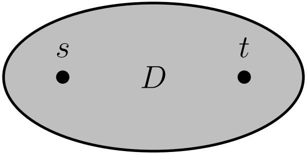
$$\sum_{a \in \delta^{\text{out}}(v)} f(a) - \sum_{a \in \delta^{\text{in}}(v)} f(a) = \begin{cases} 1 & \text{if } v = s \\ -1 & \text{if } v = t \\ 0 & \text{otherwise} \end{cases} \qquad \forall v \in V$$

$$0 \leq f(a) \leq 1 \quad \forall a \in A$$

Application 2 - Max Flow

Application: Maximum s-t Flow

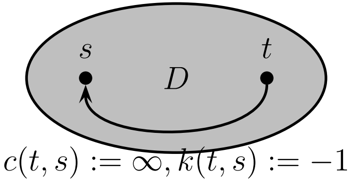
- Consider $D = (V, A)$, capacities $c : A \to \mathbb{R}_{\geq 0}$, $s, t \in V$



Application 2 - Max Flow

Application: Maximum s-t Flow

- Consider $D = (V, A)$, capacities $c : A \to \mathbb{R}_{\geq 0}$, $s, t \in V$



Reduction to MinCost Flows:

- Set $k(a) := 0$ for all $a \in A$.
- $b(v) := 0$ for all $v \in V$.
- Add arc (t, s) with cost $k(t, s) := -1$

Min Cost Flows – Variant 2

- **Given:** Directed graph $D = (V, A)$, $s, t \in V$, capacities $c : A \to \mathbb{R}_{\geq 0}$, cost $k : A \to \mathbb{R}_{\geq 0}$.
- **Goal:** Find a <u>maximum</u> flow f under c that minimizes $\text{cost}(f) := \sum_{a \in A} f(a) \cdot k(a)$

Towards a Min Cost Flow algorithm

- Let $D = (V, A)$ be a graph with capacities $c : A \to \mathbb{R}$ and cost $k : A \to \mathbb{R}_{\geq 0}$.

Definition

Let $f : A \to \mathbb{R}$ be an s-t flow under c. The **residual graph** is the graph $D_f = (V, A_f)$ with

$$\begin{aligned} f(a) < c(a) \quad &\Rightarrow \quad a \in A_f \\ f(a) > 0 \quad &\Rightarrow \quad a^{-1} \in A_f \end{aligned}$$

Towards a Min Cost Flow algorithm

- Let $D = (V, A)$ be a graph with capacities $c : A \to \mathbb{R}$ and cost $k : A \to \mathbb{R}_{\geq 0}$.

Definition

Let $f : A \to \mathbb{R}$ be an s-t flow under c. The **residual graph** is the graph $D_f = (V, A_f)$ with

$$\begin{aligned} f(a) < c(a) \quad &\Rightarrow \quad a \in A_f,\ \ \ell(a) := k(a) \\ f(a) > 0 \quad &\Rightarrow \quad a^{-1} \in A_f,\ \ \ell(a^{-1}) := -k(a) \end{aligned}$$

where $\ell : A_f \to \mathbb{R}$ is a length function.

Towards a Min Cost Flow algorithm (2)

Definition

Let $D = (V, A)$, $s, t \in V$, $c, k : A \to \mathbb{R}_{\geq 0}$. An s-t flow f with $0 \leq f \leq c$ is called **extreme** if

$$\operatorname{cost}(f) \leq \operatorname{cost}(g) \qquad \forall s\text{-}t \text{ flow } g \text{ with } 0 \leq g \leq c$$
$$\text{and } \operatorname{value}(g) = \operatorname{value}(f)$$

Towards a Min Cost Flow algorithm (2)

Definition

Let $D = (V, A)$, $s, t \in V$, $c, k : A \to \mathbb{R}_{\geq 0}$. An s-t flow f with $0 \leq f \leq c$ is called **extreme** if

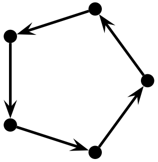
$$\text{cost}(f) \leq \text{cost}(g) \qquad \forall s\text{-}t \text{ flow } g \text{ with } 0 \leq g \leq c \text{ and } \text{value}(g) = \text{value}(f)$$

Lemma

Let f be an s-t flow in D with $0 \leq f \leq c$. Then f is an extreme flow $\Leftrightarrow$ D_f has no directed circuits C in D_f with $\ell(C) < 0$.

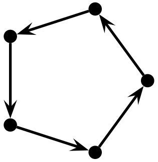
Towards a Min Cost Flow algorithm (3)

Claim 1. $\exists$ **directed circuit** C **in** D_f **with** $\ell(C) < 0 \Rightarrow f$ **is not extreme.**



Towards a Min Cost Flow algorithm (3)

Claim 1. **$\exists$ directed circuit C in D_f with $\ell(C) < 0 \Rightarrow f$ is not extreme.**

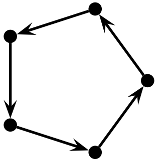


- For $\varepsilon > 0$ small enough define

$$g(a) := \begin{cases} f(a) + \varepsilon & \text{if } a \in C \\ f(a) - \varepsilon & \text{if } a^{-1} \in C \\ f(a) & \text{otherwise} \end{cases}$$

Towards a Min Cost Flow algorithm (3)

Claim 1. $\exists$ directed circuit C in D_f with $\ell(C) < 0 \Rightarrow f$ is not extreme.



- For $\varepsilon > 0$ small enough define

$$g(a) := \begin{cases} f(a) + \varepsilon & \text{if } a \in C \\ f(a) - \varepsilon & \text{if } a^{-1} \in C \\ f(a) & \text{otherwise} \end{cases}$$

- Then g is a s-t flow with $0 \le g \le c$, $\text{value}(g) = \text{value}(f)$ and $\text{cost}(g) = \text{cost}(f) + \varepsilon \cdot \ell(C) < \text{cost}(f)$.

Towards a Min Cost Flow algorithm (4)

Claim 2. If f is not extreme $\Rightarrow \exists$ directed circuit C in D_f with $\ell(C) < 0$.

Towards a Min Cost Flow algorithm (4)

Claim 2. If f is not extreme $\Rightarrow \exists$ directed circuit C in D_f with $\ell(C) < 0$.

- Let g be an s-t flow with $\mathrm{value}(g) = \mathrm{value}(f)$ and $\mathrm{cost}(g) < \mathrm{cost}(f)$.

Towards a Min Cost Flow algorithm (4)

Claim 2. If f is not extreme $\Rightarrow \exists$ directed circuit C in D_f with $\ell(C) < 0$.

- Let g be an s-t flow with $\text{value}(g) = \text{value}(f)$ and $\text{cost}(g) < \text{cost}(f)$.
- Define $h : A_f \to \mathbb{R}_{\geq 0}$ by

$$\begin{aligned} h(a) &:= g(a) - f(a) \quad \text{if } g(a) > f(a) \\ h(a^{-1}) &:= f(a) - g(a) \quad \text{if } g(a) < f(a) \end{aligned}$$

  (and $h(a) = 0$ otherwise).
- **Obs.:** h is a circulation and $\ell(h) = \text{cost}(g) - \text{cost}(f) < 0$

Towards a Min Cost Flow algorithm (4)

Claim 2. If f is not extreme $\Rightarrow \exists$ directed circuit C in D_f with $\ell(C) < 0$.

- Let g be an s-t flow with $\text{value}(g) = \text{value}(f)$ and $\text{cost}(g) < \text{cost}(f)$.
- Define $h : A_f \to \mathbb{R}_{\geq 0}$ by

$$\begin{array}{rcll} h(a) & := & g(a) - f(a) & \text{if } g(a) > f(a) \\ h(a^{-1}) & := & f(a) - g(a) & \text{if } g(a) < f(a) \end{array}$$

  (and $h(a) = 0$ otherwise).
- **Obs.:** h is a circulation and $\ell(h) = \text{cost}(g) - \text{cost}(f) < 0$
- **Exercise:** h is conic combination of circulations on single circuit
- Some circuit C in A_f has $\ell(C) < 0$ □

A Min Cost Flow algorithm

Min cost flow algorithm

Input: $D = (V, A)$, $s, t \in V$, $c : A \to \mathbb{R}_{\geq 0}$.
Output: A maximum s-t-flow under c

(1) Set $f_0(a) = 0$ for all $a \in A$.

(2) FOR $k = 0$ TO ∞

  (3) Find an s-t path $P = (a_1, \ldots, a_q)$ in D_{f_k} minimizing $\ell(P)$ (def. w.r.t. to D_{f_k}). If none exists then stop.

  (4) Set $\sigma_i := c(a_i) - f(a_i)$ if $a_i \in A$, $\sigma_i := f(a_i^{-1})$ if $a_i^{-1} \in A$

  (5) Compute $\alpha := \min\{\sigma_1, \ldots, \sigma_q\}$.

  (6) Augment f_k along P by α and call the result f_{k+1}.

A Min Cost Flow algorithm (2)

Lemma

Let f be an extreme flow. Suppose f' arises by augmenting along a minimum length path in D_f (w.r.t. ℓ). Then f' is extreme.

A Min Cost Flow algorithm (2)

Lemma

Let f be an extreme flow. Suppose f' arises by augmenting along a minimum length path in D_f (w.r.t. ℓ). Then f' is extreme.

- Let P be the path chosen to augment from f' to f

A Min Cost Flow algorithm (2)

Lemma

Let f be an extreme flow. Suppose f' arises by augmenting along a minimum length path in D_f (w.r.t. ℓ). Then f' is extreme.

- Let P be the path chosen to augment from f' to f
- Suppose f' is not extreme
- Then there is a directed circuit C with $\ell'(C) < 0$ (ℓ' is w.r.t. $D_{f'}$)

A Min Cost Flow algorithm (2)

Lemma

Let f be an extreme flow. Suppose f' arises by augmenting along a minimum length path in D_f (w.r.t. ℓ). Then f' is extreme.

- Let P be the path chosen to augment from f' to f
- Suppose f' is not extreme
- Then there is a directed circuit C with $\ell'(C) < 0$ (ℓ' is w.r.t. $D_{f'}$)

Case 1: P does not touch C.

- Then C was already contained in D_f. Contradiction!

A Min Cost Flow algorithm (2)

Lemma

Let f be an extreme flow. Suppose f' arises by augmenting along a minimum length path in D_f (w.r.t. ℓ). Then f' is extreme.

- Let P be the path chosen to augment from f' to f
- Suppose f' is not extreme
- Then there is a directed circuit C with $\ell'(C) < 0$ (ℓ' is w.r.t. $D_{f'}$)

Case 1: P does not touch C.

- Then C was already contained in D_f. Contradiction!

Case 2: P does touch C.

- Then P plus C is a shorter path $\square$

Lecture 13

Chapter 5 — Matchings in non-bipartite graphs — Part 1/2

M-augmenting paths

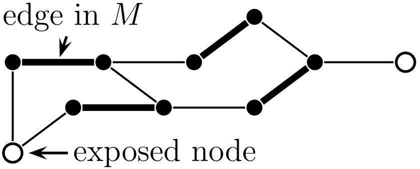
Definition

Let M be a matching in $G = (V, E)$. A path $P = (v_0, \ldots, v_t)$ in G is M-**augmenting** if

(i) t is odd

(ii) $\{v_1, v_2\}, \{v_3, v_4\}, \ldots, \{v_{t-2}, v_{t-1}\} \in M$

(iii) $v_0, v_t \notin V(M)$



M-augmenting paths

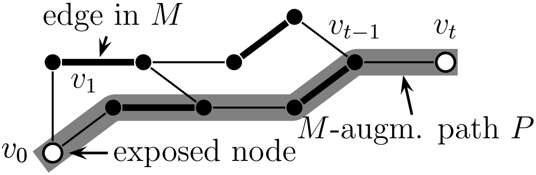
Definition

Let M be a matching in $G = (V, E)$. A path $P = (v_0, \ldots, v_t)$ in G is **M-augmenting** if

(i) t is odd

(ii) $\{v_1, v_2\}, \{v_3, v_4\}, \ldots, \{v_{t-2}, v_{t-1}\} \in M$

(iii) $v_0, v_t \notin V(M)$



M-augmenting paths

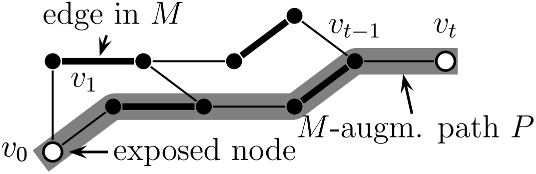
Definition

Let M be a matching in $G = (V, E)$. A path $P = (v_0, \ldots, v_t)$ in G is **M-augmenting** if

(i) t is odd

(ii) $\{v_1, v_2\}, \{v_3, v_4\}, \ldots, \{v_{t-2}, v_{t-1}\} \in M$

(iii) $v_0, v_t \notin V(M)$



Theorem

A matching M in $G = (V, E)$ is maximal $\Leftrightarrow \nexists$ any M-augmenting path in G.

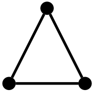
What we know from Chapter 3

- **Fact 1:** In any graph G one has $\nu(G) \leq \tau(G)$

What we know from Chapter 3

- **Fact 1:** In any graph G one has $\nu(G) \leq \tau(G)$
- **Fact 2:** In a bipartite graph G, $\nu(G) = \tau(G)$.

  Not true in general graphs!

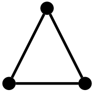


$$\nu(G) = 1 < 2 = \tau(G)$$

What we know from Chapter 3

- **Fact 1:** In any graph G one has $\nu(G) \leq \tau(G)$
- **Fact 2:** In a bipartite graph G, $\nu(G) = \tau(G)$.

  Not true in general graphs!



$\nu(G) = 1 < 2 = \tau(G)$

- **Fact 3:** Call a node v is **critical** in G if it is covered by every maximum matching. A bipartite graph without isolated vertices has at least one critical node.

  Not true in general graphs!

Towards the Tutte-Berge Formula

- Denote the minimum number of exposed vertices as

$$\begin{aligned}\mathrm{ex}(G) &:= \min\{\#M\text{-exposed nodes} \mid M \text{ matching in } G\} \\ &= |V| - 2\nu(G)\end{aligned}$$

Towards the Tutte-Berge Formula

- Denote the minimum number of exposed vertices as

$$\begin{aligned}\mathrm{ex}(G) &:= \min\{\#M\text{-exposed nodes} \mid M \text{ matching in } G\}\\ &= |V| - 2\nu(G)\end{aligned}$$

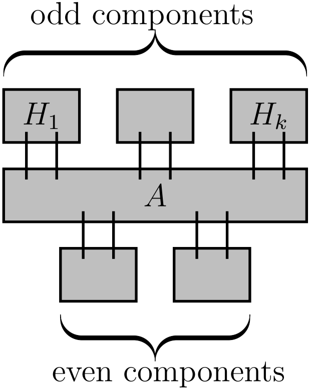
- A connected component in a graph is called **odd** if it has an odd number of vertices.
- We define $\mathbf{odd}(G)$ as the number of odd components in graph G.

Towards the Tutte-Berge Formula (2)

- Consider graph $G = (V, E)$ with matching M and $A \subseteq V$.

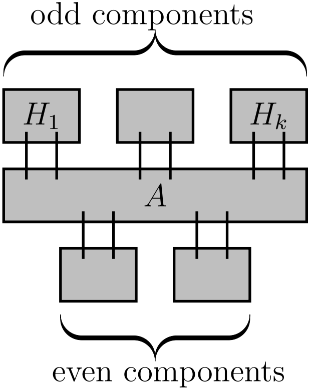
Towards the Tutte-Berge Formula (2)

- Consider graph $G = (V, E)$ with matching M and $A \subseteq V$.
- Let $H_1, \ldots, H_k$ w. $k := \text{odd}(G \setminus A)$ be odd comp. in $G \setminus A$.



Towards the Tutte-Berge Formula (2)

- Consider graph $G = (V, E)$ with matching M and $A \subseteq V$.
- Let $H_1, \ldots, H_k$ w. $k := \text{odd}(G \setminus A)$ be odd comp. in $G \setminus A$.



- Then $\forall i \in [k]$, M either leaves a node in H_i exposed, or it contains an edge between a node in C_i and A. Hence

$$\text{ex}(G) \geq k - |A|$$

The Tutte-Berge Formula

Theorem (Tutte-Berge Formula)

For every graph $G = (V, E)$ one has

$$\nu(G) = \min_{A \subseteq V} \left\{ \frac{1}{2} \big(|V| + |A| - odd(G \setminus A) \big) \right\}$$

- Equivalent to

$$\mathrm{ex}(G) = \max\{\mathrm{odd}(G \setminus A) - |A| \mid A \subseteq V\}$$

since $2\nu(G) + \mathrm{ex}(G) = |V|$.

The Tutte-Berge Formula (2)

Lemma

Let $G = (V, E)$ be a connected graph with $E \neq \emptyset$ and no critical node. Then $ex(G) = 1$.

The Tutte-Berge Formula (2)

Lemma

Let $G = (V, E)$ be a connected graph with $E \neq \emptyset$ and no critical node. Then $ex(G) = 1$.

Proof:

- If $\mathrm{ex}(G) = 0$, then every node is critical.

The Tutte-Berge Formula (2)

Lemma

Let $G = (V, E)$ be a connected graph with $E \neq \emptyset$ and no critical node. Then $ex(G) = 1$.

Proof:

- If $\mathrm{ex}(G) = 0$, then every node is critical.
- Hence suppose for the sake of contradiction that $\mathrm{ex}(G) \geq 2 \rightarrow$ every maximal matching M leaves at least two nodes exposed.

The Tutte-Berge Formula (2)

Lemma

Let $G = (V, E)$ be a connected graph with $E \neq \emptyset$ and no critical node. Then $ex(G) = 1$.

Proof:

- If $\mathrm{ex}(G) = 0$, then every node is critical.
- Hence suppose for the sake of contradiction that $\mathrm{ex}(G) \geq 2 \to$ every maximal matching M leaves at least two nodes exposed.
- For $u, v \in V$, let $d(u, v)$ be the distance in G (in terms of the number of edges; note that here we use connectedness).

The Tutte-Berge Formula (2)

Lemma

Let $G = (V, E)$ be a connected graph with $E \neq \emptyset$ and no critical node. Then $ex(G) = 1$.

Proof:

- If $\mathrm{ex}(G) = 0$, then every node is critical.
- Hence suppose for the sake of contradiction that $\mathrm{ex}(G) \geq 2 \rightarrow$ every maximal matching M leaves at least two nodes exposed.
- For $u, v \in V$, let $d(u, v)$ be the distance in G (in terms of the number of edges; note that here we use connectedness).
- Fix a maximal matching M that minimizes $d(u, v)$ for a pair of M-exposed nodes.

The Tutte-Berge Formula (2)

Lemma

Let $G = (V, E)$ be a connected graph with $E \neq \emptyset$ and no critical node. Then $ex(G) = 1$.

Proof:

- If $\mathrm{ex}(G) = 0$, then every node is critical.
- Hence suppose for the sake of contradiction that $\mathrm{ex}(G) \geq 2 \to$ every maximal matching M leaves at least two nodes exposed.
- For $u, v \in V$, let $d(u, v)$ be the distance in G (in terms of the number of edges; note that here we use connectedness).
- Fix a maximal matching M that minimizes $d(u, v)$ for a pair of M-exposed nodes.

Case $d(u, v) = 1$.

- Then $M \cup \{u, v\}$ is a bigger matching $\to$ contradiction!

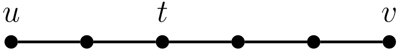
The Tutte-Berge Formula (3)

Case $d(u, v) \geq 2$:

The Tutte-Berge Formula (3)

Case $d(u,v) \geq 2$:

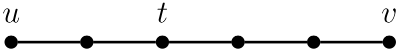
- Fix $t \in V$ with $d(u,t), d(v,t) < d(u,v)$



The Tutte-Berge Formula (3)

Case $d(u, v) \geq 2$:

- Fix $t \in V$ with $d(u, t), d(v, t) < d(u, v)$

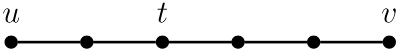


- Consider a maximal matching N that leaves t exposed. If there are several, choose the matching that maximizes the number $|M \cap N|$ of joint edges.

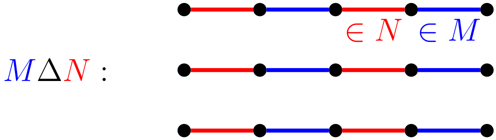
The Tutte-Berge Formula (3)

Case $d(u, v) \geq 2$:

- Fix $t \in V$ with $d(u, t), d(v, t) < d(u, v)$

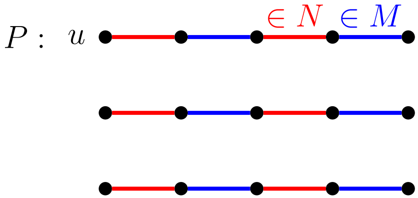


- Consider a maximal matching N that leaves t exposed. If there are several, choose the matching that maximizes the number $|M \cap N|$ of joint edges.
- Next, consider the symmetric difference $M \Delta N$. Each of the nodes u, v, t is exposed in either M or N, so they are all endpoints of some paths in $M \Delta N$. Since M and N are maximal and we maximized $|M \cap N|$, we know that $M \Delta N$ consists only of even length paths.



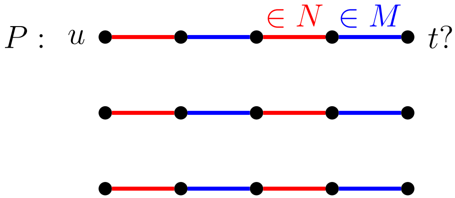
The Tutte-Berge Formula (4)

- Consider the even length path P containing u as an endpoint.



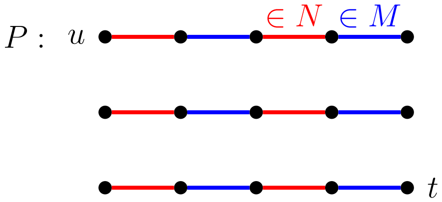
The Tutte-Berge Formula (4)

- Consider the even length path P containing u as an endpoint.
- If the other endpoint is t, then $M \Delta P$ has the exposed nodes v and t which contradicts the choice of M.



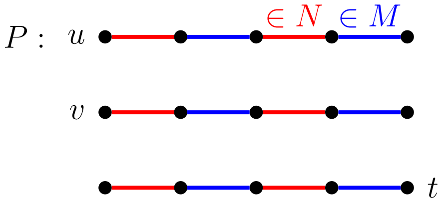
The Tutte-Berge Formula (4)

- Consider the even length path P containing u as an endpoint.
- If the other endpoint is t, then $M \Delta P$ has the exposed nodes v and t which contradicts the choice of M.



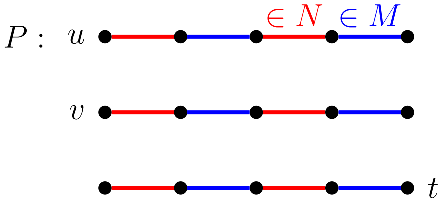
The Tutte-Berge Formula (4)

- Consider the even length path P containing u as an endpoint.
- If the other endpoint is t, then $M \Delta P$ has the exposed nodes v and t which contradicts the choice of M.
- That means P has u as one endpoint and the other one is neither v nor t.



The Tutte-Berge Formula (4)

- Consider the even length path P containing u as an endpoint.
- If the other endpoint is t, then $M \Delta P$ has the exposed nodes v and t which contradicts the choice of M.
- That means P has u as one endpoint and the other one is neither v nor t.



- Then the matching $N \Delta P$ still has t exposed but has more edges in common with M, which is a contradiction. $\square$

The Tutte-Berge Formula (5)

Proof of the Tutte-Berge Formula

- We prove the equivalent statement:

$$\mathrm{ex}(G) = \max\{\mathrm{odd}(G \setminus A) - |A| \mid A \subseteq V\}$$

The Tutte-Berge Formula (5)

Proof of the Tutte-Berge Formula

- We prove the equivalent statement:

$$\mathrm{ex}(G) = \max\{\mathrm{odd}(G \setminus A) - |A| \mid A \subseteq V\}$$

- We already argued the direction "$\geq$". We prove "$\leq$" by induction.

The Tutte-Berge Formula (6)

Case: G is disconnected:

The Tutte-Berge Formula (6)

Case: G is disconnected:

- Let $G_1, \ldots, G_k$ are the connected components of G (doesn't matter whether these are even or odd) with $k \geq 2$.

The Tutte-Berge Formula (6)

Case: G is disconnected:

- Let $G_1, \ldots, G_k$ are the connected components of G (doesn't matter whether these are even or odd) with $k \geq 2$.
- We apply induction to each of the connected components to find $A_i \subseteq V(G_i)$ with $\text{ex}(G_i) = \text{odd}(G_i \setminus A_i) - |A_i|$.

The Tutte-Berge Formula (6)

Case: G is disconnected:

- Let $G_1, \ldots, G_k$ are the connected components of G (doesn't matter whether these are even or odd) with $k \geq 2$.
- We apply induction to each of the connected components to find $A_i \subseteq V(G_i)$ with $\mathrm{ex}(G_i) = \mathrm{odd}(G_i \setminus A_i) - |A_i|$.
- Then $A := \bigcup_{i=1}^{k} A_i$ gives

$$
\begin{aligned}
\mathrm{ex}(G) &= \sum_{i=1}^{k} \mathrm{ex}(G_i) \\
&= \underbrace{\sum_{i=1}^{k} \mathrm{odd}(G_i \setminus A_i)}_{=\mathrm{odd}(G \setminus A)} - \underbrace{\sum_{i=1}^{k} |A_i|}_{=|A|} \\
&= \mathrm{odd}(G \setminus A) - |A|
\end{aligned}
$$

The Tutte-Berge Formula (7)

Case: G connected and there is no critical node.

The Tutte-Berge Formula (7)

Case: G connected and there is no critical node.

- Using the last Lemma $\mathrm{ex}(G) = 1$ and so G is odd.
- Hence for $A := \emptyset$ we have $\mathrm{ex}(G) = 1 = \mathrm{odd}(G \setminus A) - |A|$.

The Tutte-Berge Formula (7)

Case: G connected and there is no critical node.

- Using the last Lemma $\text{ex}(G) = 1$ and so G is odd.
- Hence for $A := \emptyset$ we have $\text{ex}(G) = 1 = \text{odd}(G \setminus A) - |A|$.

Case: G connected and there is a critical node $u \in V$.

The Tutte-Berge Formula (7)

Case: G connected and there is no critical node.

- Using the last Lemma $\mathrm{ex}(G) = 1$ and so G is odd.
- Hence for $A := \emptyset$ we have $\mathrm{ex}(G) = 1 = \mathrm{odd}(G \setminus A) - |A|$.

Case: G connected and there is a critical node $u \in V$.

- We apply induction to $G \setminus \{u\}$ and obtain a set $A \subseteq V \setminus \{u\}$ with

$$
\begin{aligned}
\mathrm{ex}(G) &\overset{u \text{ critical}}{=} \mathrm{ex}(G \setminus \{u\}) - 1 \\
&\overset{\text{induction}}{=} \mathrm{odd}((G \setminus \{u\}) \setminus A) - |A| - 1 \\
&= \mathrm{odd}(G \setminus (A \cup \{u\})) - |A \cup \{u\}|
\end{aligned}
$$

- Here we use that deleting a critical node increases the minimum number of exposed nodes by 1. That means $A \cup \{u\}$ satisfies the claim. $\square$

Tutte's 1-factor theorem

Corollary (Tutte's 1-factor theorem)

A graph $G = (V, E)$ has a perfect matching if and only if $odd(G \setminus A) \leq |A|$ for all $A \subseteq V$.

Edmonds Gallai Decomposition

Theorem (Edmonds Gallai decomposition)

Let $G = (V, E)$ be an undirected graph. Let

$$\begin{aligned} D &:= \{v \in V \mid \exists \textit{ some max matching that leaves } v \textit{ uncovered}\} \\ A &:= \{\textit{neighbors of } D\} \\ C &:= \textit{remaining vertices} \end{aligned}$$

Then $ex(G) = odd(G \setminus A) - |A|$.

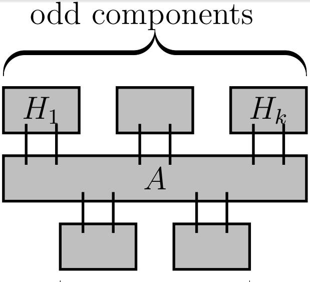
Edmonds Gallai Decomposition

Theorem (Edmonds Gallai decomposition)

Let $G = (V, E)$ be an undirected graph. Let

$$\begin{aligned} D &:= \{v \in V \mid \exists \text{ some max matching that leaves } v \text{ uncovered}\} \\ A &:= \{\text{neighbors of } D\} \\ C &:= \text{remaining vertices} \end{aligned}$$

Then $ex(G) = odd(G \setminus A) - |A|$.



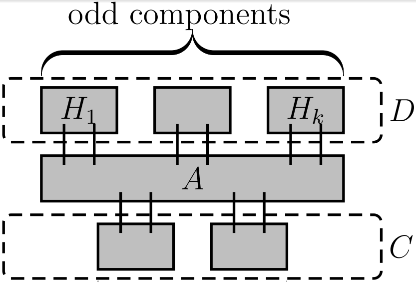
Edmonds Gallai Decomposition

Theorem (Edmonds Gallai decomposition)

Let $G = (V, E)$ be an undirected graph. Let

$$\begin{array}{lcl} D & := & \{v \in V \mid \exists \text{ some max matching that leaves } v \text{ uncovered}\} \\ A & := & \{\text{neighbors of } D\} \\ C & := & \text{remaining vertices} \end{array}$$

Then $ex(G) = odd(G \setminus A) - |A|$.



Lecture 14

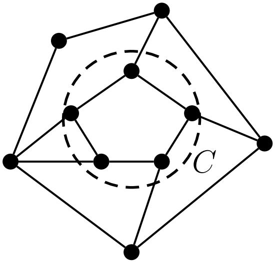
Chapter 5 — Matchings in non-bipartite graphs — Part 2/2

Contraction

Definition

Let $G = (V, E)$ be an undirected graph and $C \subseteq V$. Then **contracting** C gives the graph $G/C = (V/C, E/C)$ where $V := V \setminus C \cup \{C\}$ and edges E/C defined by

$$\begin{aligned} \{u,v\} \in E, \{u,v\} \cap C = \emptyset \quad &\Rightarrow \quad \{u,v\} \in E/C \\ \{u,v\} \in E, u \notin C, v \in C \quad &\Rightarrow \quad \{u,C\} \in E/C \end{aligned}$$



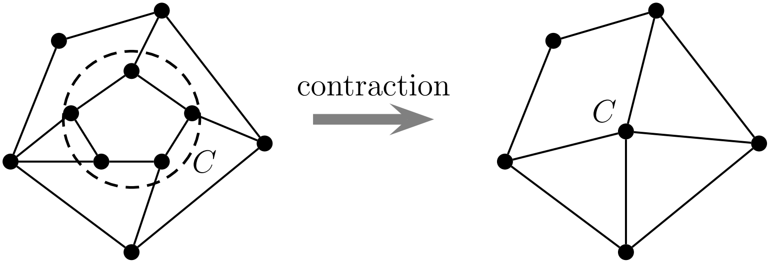
graph G

Contraction

Definition

Let $G = (V, E)$ be an undirected graph and $C \subseteq V$. Then **contracting** C gives the graph $G/C = (V/C, E/C)$ where $V := V \setminus C \cup \{C\}$ and edges E/C defined by

$$\begin{aligned} \{u, v\} \in E, \{u, v\} \cap C = \emptyset \quad &\Rightarrow \quad \{u, v\} \in E/C \\ \{u, v\} \in E, u \notin C, v \in C \quad &\Rightarrow \quad \{u, C\} \in E/C \end{aligned}$$

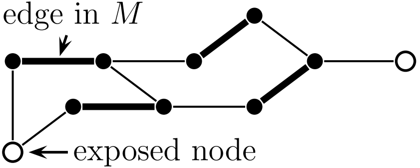


graph G contracted graph G/C

Alternating walks

Definition

A walk $P = (v_0, v_1, \ldots, v_t)$ is called M**-alternating** if for each $i \in \{1, \ldots, t-1\}$ exactly one of the edges $\{v_{i-1}, v_i\}$, $\{v_i, v_{i+1}\}$ lies in M.

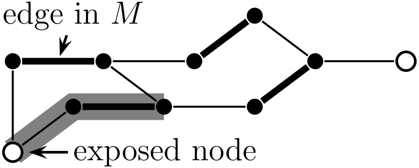


- **Note:** M-augmenting paths $\subseteq$ M-alternating walks

Alternating walks

Definition

A walk $P = (v_0, v_1, \ldots, v_t)$ is called M**-alternating** if for each $i \in \{1, \ldots, t-1\}$ exactly one of the edges $\{v_{i-1}, v_i\}$, $\{v_i, v_{i+1}\}$ lies in M.

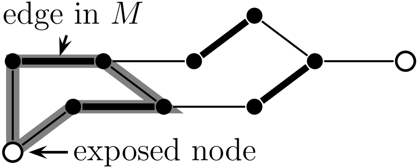


- **Note:** M-augmenting paths $\subseteq$ M-alternating walks

Alternating walks

Definition

A walk $P = (v_0, v_1, \ldots, v_t)$ is called ***M*-alternating** if for each $i \in \{1, \ldots, t-1\}$ exactly one of the edges $\{v_{i-1}, v_i\}$, $\{v_i, v_{i+1}\}$ lies in M.

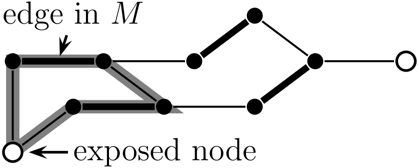


- **Note:** M-augmenting paths $\subseteq$ M-alternating walks

Alternating walks

Definition

A walk $P = (v_0, v_1, \ldots, v_t)$ is called **M-alternating** if for each $i \in \{1, \ldots, t-1\}$ exactly one of the edges $\{v_{i-1}, v_i\}$, $\{v_i, v_{i+1}\}$ lies in M.

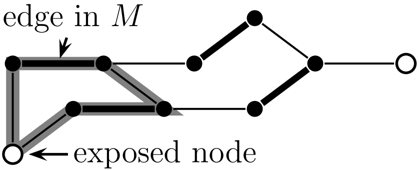


- **Note:** M-augmenting paths $\subseteq$ M-alternating walks
- For $W \subseteq V$, a **W-W alternating walk** is a walk that starts at $w_1 \in W$ and ends at $w_2 \in W$ (possibly $w_1 = w_2$).

Alternating walks

Definition

A walk $P = (v_0, v_1, \dots, v_t)$ is called **M-alternating** if for each $i \in \{1, \dots, t-1\}$ exactly one of the edges $\{v_{i-1}, v_i\}$, $\{v_i, v_{i+1}\}$ lies in M.

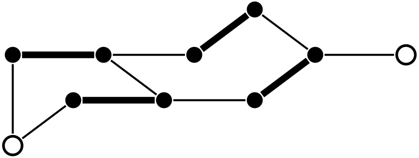


- **Note:** M-augmenting paths $\subseteq$ M-alternating walks
- For $W \subseteq V$, a **W-W alternating walk** is a walk that starts at $w_1 \in W$ and ends at $w_2 \in W$ (possibly $w_1 = w_2$).
- Set $W := \{v \in V \mid v \text{ is } M\text{-exposed}\}$. An M-augmenting path is a W-W alternating walk in which all vertices are distinct.

Finding W-W alternating walks

Lemma

Let $G = (V, E)$ be a graph, M matching, W are the M-exposed vertices. One can find a shortest W-W alternating walk in polynomial time.



Finding W-W alternating walks

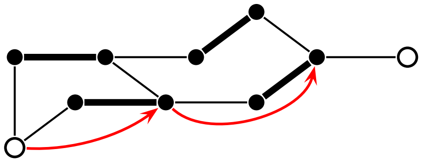
Lemma

Let $G = (V, E)$ be a graph, M matching, W are the M-exposed vertices. One can find a shortest W-W alternating walk in polynomial time.

Proof:

- Define auxiliary directed graph $D = (V, A)$ with

$$A := \{(w_1, w_2) \mid \exists u \in V : \{w_1, u\} \in E, \{u, w_2\} \in M\}$$



Finding W-W alternating walks

Lemma

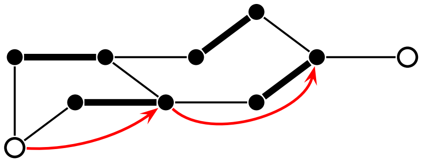
Let $G = (V, E)$ be a graph, M matching, W are the M-exposed vertices. One can find a shortest W-W alternating walk in polynomial time.

Proof:

- Define auxiliary directed graph $D = (V, A)$ with

$$A := \{(w_1, w_2) \mid \exists u \in V : \{w_1, u\} \in E, \{u, w_2\} \in M\}$$

- Take the shortest path in D from W to $N(W)$



M-blossoms

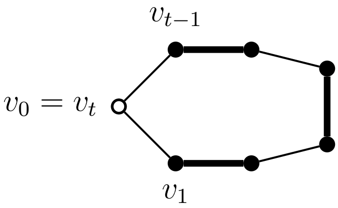
Definition

An M-alternating walk $P = (v_0, \dots, v_t)$ is called an M**-blossom** if

(i) $v_0, \dots, v_{t-1}$ are distinct,

(ii) $v_0 = v_t$,

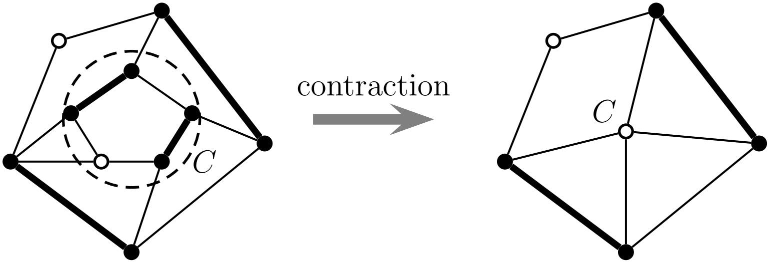
(iii) v_0 is M-exposed.



Augm. P in G/C $\leftrightarrow$ Augm. P in G (1)

Lemma

Let M be a matching in G and C be an M-blossom. Then $\exists M$-augmenting path in G $\Leftrightarrow$ $\exists M/C$-augmenting path in G/C.



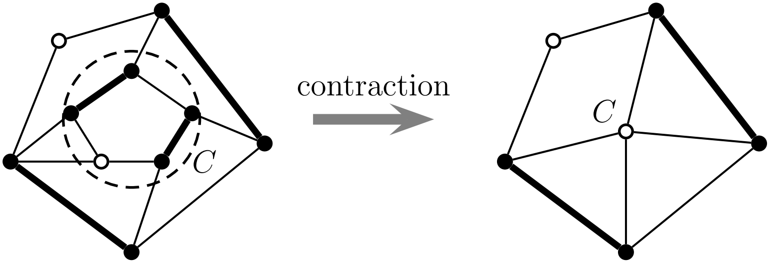
graph G with blossom C

contracted graph G/C

Augm. P in G/C $\leftrightarrow$ Augm. P in G (1)

Lemma

Let M be a matching in G and C be an M-blossom. Then $\exists M$-augmenting path in $G \Leftrightarrow \exists M/C$-augmenting path in G/C.



graph G with blossom C contracted graph G/C

Proof:

- Observation: M/C is indeed a matching

Augm. P in G/C → Augm. P in G (3)

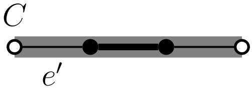
Claim I. M/C-augmenting path in G/C can be extended to an M-augmenting path in G.

Augm. P in G/C $\rightarrow$ Augm. P in G (3)

Claim I. M/C-augmenting path in G/C can be extended to an M-augmenting path in G.

Proof.

- Let P be an M/C augmenting path in G/C.
- If P does not contain C, the claim is clear. Otherwise C is end point of P.

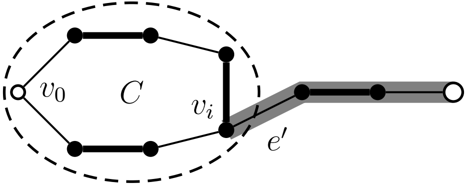


Augm. P in $G/C \rightarrow$ Augm. P in G (3)

Claim I. M/C-augmenting path in G/C can be extended to an M-augmenting path in G.

Proof.

- Let P be an M/C augmenting path in G/C.
- If P does not contain C, the claim is clear. Otherwise C is end point of P.
- After expansion, P will enter C with a non-M edge e' in v'.

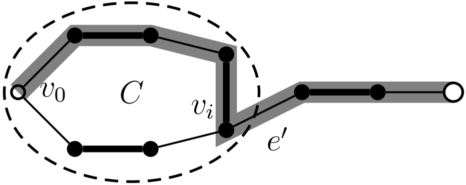


Augm. P in G/C → Augm. P in G (3)

Claim I. M/C-augmenting path in G/C can be extended to an M-augmenting path in G.

Proof.

- Let P be an M/C augmenting path in G/C.
- If P does not contain C, the claim is clear. Otherwise C is end point of P.
- After expansion, P will enter C with a non-M edge e' in v'.
- From v' we extend P **clockwise or counter-clockwise** so that the first edge we take from v' in C is an M-edge.
- This gives an M-augmenting path in G. $\square$



M-augm. paths survive contraction

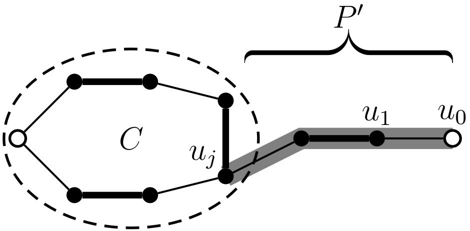
Claim II. $\exists M$-augmenting path in $G \Rightarrow \exists M/C$-augmenting path in G/C.

- Consider M-augm path P that includes some nodes of C

M-augm. paths survive contraction

Claim II. $\exists M$-augmenting path in $G \Rightarrow \exists M/C$-augmenting path in G/C.

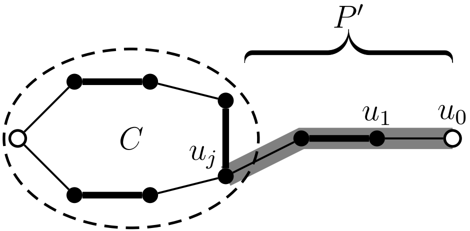
- Consider M-augm path P that includes some nodes of C
- Consider the beginning $P' = (u_0, u_1, \ldots, u_j)$ of P with $u_0 \notin C$ being M-exposed where u_j is the first node in C.



M-augm. paths survive contraction

Claim II. $\exists M$-augmenting path in $G \Rightarrow \exists M/C$-augmenting path in G/C.

- Consider M-augm path P that includes some nodes of C
- Consider the beginning $P' = (u_0, u_1, \ldots, u_j)$ of P with $u_0 \notin C$ being M-exposed where u_j is the first node in C.



- Note that $M \cap \delta(C) = \emptyset$.
- That means C is M/C-exposed in G/C and P' is an augmenting path with u_0 and C as exposed vertices. $\square$

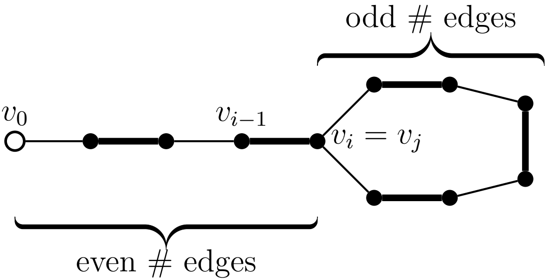
Finding an M-blossom

Theorem

Let M be a matching in G with M-exposed vertices W and let $v \in V$. Let $P = (v_0, v_1, \dots, v_t = v)$ be a shortest even-length M-alternating W-v walk. One of the cases is true:

(i) *P is a path*

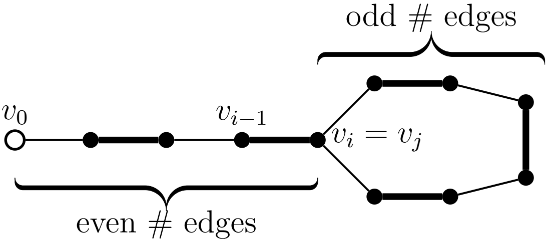
(ii) *There are $i < j$ such that $v_i = v_j$, i is even, j is odd, $v_0, \dots, v_{j-1}$ are distinct*



Finding an M-blossom (2)

Proof:

- Assume $P = (v_1, \ldots, v_t)$ is not simple and v_j is the first revisited node.



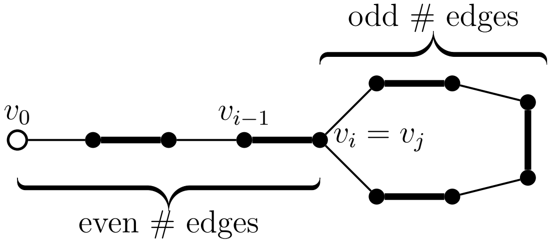
Finding an M-blossom (2)

Proof:

- Assume $P = (v_1, \ldots, v_t)$ is not simple and v_j is the first revisited node.

Case $j - i$ is even.

- Then $P' = (v_1, \ldots, v_i, v_{j+1}, \ldots, v_t)$ would be shorter walk.



Finding an M-blossom (2)

Proof:

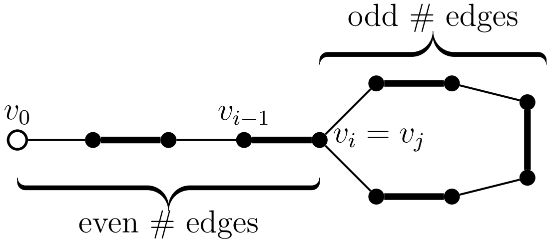
- Assume $P = (v_1, \ldots, v_t)$ is not simple and v_j is the first revisited node.

Case $j - i$ is even.

- Then $P' = (v_1, \ldots, v_i, v_{j+1}, \ldots, v_t)$ would be shorter walk.

Case: j even and i odd.

- We have $\{v_i, v_{i+1}\}, \{v_{j-1}, v_j\} \in M$ and $v_i = v_j$. So $v_{i+1} = v_{j-1}$ and v_{j-1} was revisited previously (note that $j - i \geq 3$).



Finding an M-blossom (2)

Proof:

- Assume $P = (v_1, \ldots, v_t)$ is not simple and v_j is the first revisited node.

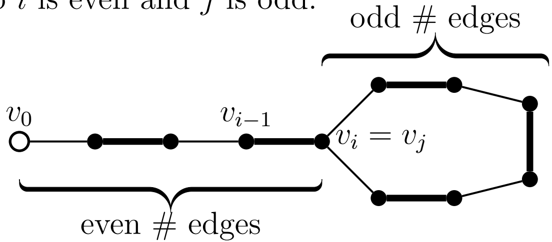
Case $j - i$ is even.

- Then $P' = (v_1, \ldots, v_i, v_{j+1}, \ldots, v_t)$ would be shorter walk.

Case: j even and i odd.

- We have $\{v_i, v_{i+1}\}, \{v_{j-1}, v_j\} \in M$ and $v_i = v_j$. So $v_{i+1} = v_{j-1}$ and v_{j-1} was revisited previously (note that $j - i \geq 3$).

Conclusion: So i is even and j is odd. □



Edmonds algorithm

Edmonds' algorithm

Input: $G = (V, E)$, matching $M \subseteq E$.
Output: A matching N with $|N| = |M| + 1$ or conclusion that M is maximal.

(1) Set $W := M$-exposed vertices.
(2) Compute shortest W-W M-alternating walk $P = (v_0, \dots, v_t)$
(3) **Case 1. There is no such walk**
  (4) Return "M is maximal"
(5) **Case 2. There is such a walk**
  (6) **Case 2a. P is a path**
    (7) Then P is an M-augmenting path.
    (8) Return $N := M \Delta E(P)$
  (9) **Case 2b. P is not a path**
    (10) ..

Edmonds algorithm (2)

⋮

(9) **Case 2b. P is not a path**

- (10) Let v_j be first revisited node with $v_i = v_j$ for $i < j$, i even, j odd
- (11) Set $M' := M \Delta \{\{v_0, v_1\}, \ldots, \{v_{i-1}, v_i\}\}$ which is a matching with $|M'| = |M|$
- (12) Set $C := \{v_i, v_{i+1}, \ldots, v_j\}$ which is an M'-blossom
- (13) Call algorithm recursively for G/C and M'/C
- (14) IF M'/C is maximal, then
  - (15) M' is maximal in $G \to$ return M is maximal
- (16) IF $\exists M'/C$-augmenting path P, then
  - (17) obtain M'-augmenting path P'
  - (18) return $M' \Delta E(P')$

Edmonds algorithm (3)

Theorem

A maximum cardinality matching in $G = (V, E)$ can be found in time $O(|V|^2|E|)$.

Edmonds algorithm (3)

Theorem

A maximum cardinality matching in $G = (V, E)$ can be found in time $O(|V|^2|E|)$.

Proof:

- A shortest M-alternating W-W walk can be found in time $O(|E|)$
- Recursion has depth at most $|V|$
- Augmenting M by 1 takes time at most $O(|V| \cdot |E|)$
- Total time $O(|V|^2 \cdot |E|)$

The Perfect matching polytope

- Recall that

$$P_{\text{perfectmatching}}(G) = \text{conv}\{\chi^M \mid M \subseteq E \text{ is matching}\}$$

The Perfect matching polytope

- Recall that

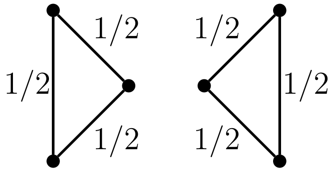
$$P_{\text{perfectmatching}}(G) = \text{conv}\{\chi^M \mid M \subseteq E \text{ is matching}\}$$

- For bipartite graphs,

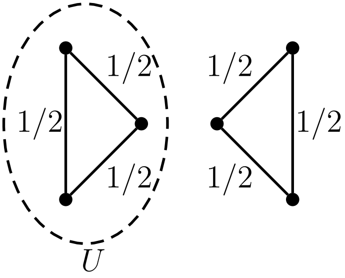
$$P_{\text{perfectmatching}}(G) = \{x \in \mathbb{R}^E \mid x(\delta(v)) = 1 \;\; \forall v \in V, x_e \geq 0 \;\; \forall e \in E\}$$

- False for non-bipartite graphs

The Perfect matching polytope (2)



The Perfect matching polytope (2)

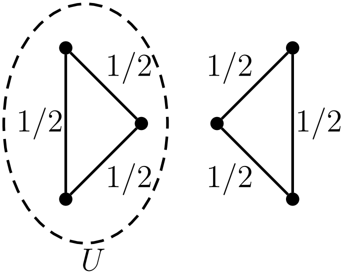


The Perfect matching polytope (2)

Theorem (Edmonds 1965)

In any graph $G = (V, E)$,

$$P_{perfectmat.(G)} = \left\{ x \in \mathbb{R}^E \mid \begin{array}{rcll} x(\delta(v)) & = & 1 & \forall v \in V \\ x_e & \geq & 0 & \forall e \in E \\ x(\delta(U)) & \geq & 1 & \forall U \subseteq V, |U| \ odd \end{array} \right\}$$



Lecture 15

Semidefinite programming — Part 1/2

Positive semi-definite matrices

- A matrix $X \in \mathbb{R}^{n \times n}$ is **symmetric** if $X_{ij} = X_{ji}$ for all $i, j \in [n]$

Positive semi-definite matrices

- A matrix $X \in \mathbb{R}^{n \times n}$ is **symmetric** if $X_{ij} = X_{ji}$ for all $i, j \in [n]$
- **Fact.** For a symmetric matrix, all Eigenvalues are real.

Positive semi-definite matrices

- A matrix $X \in \mathbb{R}^{n \times n}$ is **symmetric** if $X_{ij} = X_{ji}$ for all $i, j \in [n]$
- **Fact.** For a symmetric matrix, all Eigenvalues are real.

Definition

A symmetric matrix $X \in \mathbb{R}^{n \times n}$ is **positive semidefinite** if all its Eigenvalues are non-negative.

- We write $X \succeq 0 \Leftrightarrow X$ is PSD.

Positive semi-definite matrices

- A matrix $X \in \mathbb{R}^{n\times n}$ is **symmetric** if $X_{ij} = X_{ji}$ for all $i, j \in [n]$
- **Fact.** For a symmetric matrix, all Eigenvalues are real.

Definition

A symmetric matrix $X \in \mathbb{R}^{n\times n}$ is **positive semidefinite** if all its Eigenvalues are non-negative.

- We write $X \succeq 0 \Leftrightarrow X$ is PSD.
- For $A, B \in \mathbb{R}^{n\times n}$ we write

$$\langle A, B\rangle := \sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}\cdot B_{ij}$$

as the **Frobenius inner product**.

Positive semi-definite matrices (2)

Lemma

For a symmetric matrix $X \in \mathbb{R}^{n \times n}$, the following is equivalent

a) $a^T X a \geq 0 \; \forall a \in \mathbb{R}^n$.

b) X *is positive semidefinite.*

c) *There exists a matrix* U *so that* $X = UU^T$.

d) *There are* $u_1, \ldots, u_n \in \mathbb{R}^r$ *with* $X_{ij} = \langle u_i, u_j \rangle$ *for* $i, j \in [n]$.

Positive semi-definite matrices (2)

Lemma

For a symmetric matrix $X \in \mathbb{R}^{n \times n}$, the following is equivalent

a) $a^T X a \geq 0 \ \forall a \in \mathbb{R}^n$.

b) *X is positive semidefinite.*

c) *There exists a matrix U so that $X = UU^T$.*

d) *There are $u_1, \dots, u_n \in \mathbb{R}^r$ with $X_{ij} = \langle u_i, u_j \rangle$ for $i, j \in [n]$.*

Proof:

- Any symmetric real matrix is **diagonalizable**, that means $X = WDW^T = \sum_{i=1}^n \lambda_i v_i v_i^T$ for diagonal D, orth. W.

Then

- $a) \Rightarrow b)$. $0 \leq v_i^T X v_i = \lambda_i \|v_i\|_2^2 = \lambda_i$
- $b) \Rightarrow c)$. $X = WDW^T = UU^T$ for $U := W\sqrt{D}$.
- $c) \Leftrightarrow d)$. Choose u_i as ith row of U.
- $c) \Rightarrow a)$. For any $a \in \mathbb{R}^n$, $a^T X a = \|Ua\|_2^2 \geq 0$.

Positive semi-definite matrices 3

Definition

The **cone of PSD matrices** is

$$\begin{aligned}\mathbb{S}^n_{\geq 0} &:= \{X \in \mathbb{R}^{n\times n} \mid X \text{ symmetric}, X \succeq 0\} \\ &= \{X \in \mathbb{R}^{n\times n} \mid X \text{ symmetric}, \ \langle X, aa^T \rangle \geq 0 \ \forall a \in \mathbb{R}^n\}\end{aligned}$$

Positive semi-definite matrices 3

Definition

The **cone of PSD matrices** is

$$\begin{aligned} \mathbb{S}^n_{\geq 0} &:= \{X \in \mathbb{R}^{n \times n} \mid X \text{ symmetric}, X \succeq 0\} \\ &= \{X \in \mathbb{R}^{n \times n} \mid X \text{ symmetric}, \left\langle X, aa^T \right\rangle \geq 0 \; \forall a \in \mathbb{R}^n\} \end{aligned}$$

- **Fact:** $\mathbb{S}^n_{\geq 0}$ is convex.

Positive semi-definite matrices 3

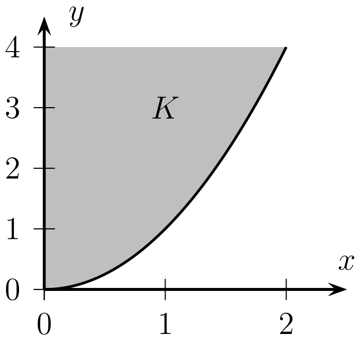
Definition

The **cone of PSD matrices** is

$$\begin{aligned}\mathbb{S}^n_{\geq 0} &:= \{X \in \mathbb{R}^{n\times n} \mid X \text{ symmetric}, X \succeq 0\}\\ &= \{X \in \mathbb{R}^{n\times n} \mid X \text{ symmetric}, \langle X, aa^T\rangle \geq 0 \;\forall a \in \mathbb{R}^n\}\end{aligned}$$

- **Fact:** $\mathbb{S}^n_{\geq 0}$ is convex.

$$K := \left\{(x,y) \mid \begin{pmatrix} 1 & x \\ x & y \end{pmatrix} \succeq 0\right\}$$



A semidefinite program

- A **semidefinite program** is of the form:

$$
\begin{aligned}
\max \langle C, X \rangle & & & \\
\langle A_k, X \rangle & \leq & b_k & \quad \forall k = 1, \ldots, m \\
X & & \text{symmetric} & \\
X & \succeq & 0 &
\end{aligned}
$$

where $C, A_1, \ldots, A_m \in \mathbb{R}^{n \times n}$.

A semidefinite program

- A **semidefinite program** is of the form:

$$\begin{array}{rcl} \max \langle C, X \rangle & & \\ \langle A_k, X \rangle & \leq & b_k \quad \forall k = 1, \ldots, m \\ X & & \text{symmetric} \\ X & \succeq & 0 \end{array}$$

where $C, A_1, \ldots, A_m \in \mathbb{R}^{n \times n}$.

Less well behaved than LPs:

- **Issue 1:** Strong duality might fail.
- **Issue 2:** Possibly all solutions are irrational
- **Issue 3:** Possibly exact solutions have exponential encoding length

Solvability of Semidefinite Programs

Theorem

Given rational input $A_1, \ldots, A_m, b_1, \ldots, b_m, C, R$ *and* $\varepsilon > 0$, *suppose*

$$SDP = \max\{\langle C, X\rangle \mid \langle A_k, X\rangle \leq b_k \,\forall k;\; X\, symmetric;\; X \succeq 0\}$$

is feasible and all feasible points are contained in $B(\mathbf{0}, R)$. *Then one can find a* X^* *with*

$$\langle A_k, X^*\rangle \leq b_k + \varepsilon,\; X^* \; symmetric,\; X^* \succeq 0$$

such that $\langle C, X^*\rangle \geq SDP - \varepsilon$. *The running time is polynomial in the input length,* $\log(R)$ *and* $\log(1/\varepsilon)$ *(in the Turing machine model).*

Vector programs

Idea:

- $Y \succeq 0$ holds iff $Y_{ij} = \langle v_i, v_j \rangle$ for some vectors v_i

Vector programs

Idea:

- $Y \succeq 0$ holds iff $Y_{ij} = \langle v_i, v_j \rangle$ for some vectors v_i

SDP:

$$\begin{aligned} \max \sum_{i,j} C_{ij} Y_{ij} & & & \\ \sum_{i,j} A_{ij}^k \cdot Y_{ij} & \leq & b_k & \quad \forall k \\ Y & & \text{sym.} & \\ Y & \succeq & 0 & \end{aligned}$$

Vector program

$$\begin{aligned} \max \sum_{i,j} C_{ij} \langle v_i, v_j \rangle & & & \\ \sum_{i,j} A_{ij}^k \cdot \langle v_i, v_j \rangle & \leq & b_k & \quad \forall k \\ v_i & \in & \mathbb{R}^r & \quad \forall i \end{aligned}$$

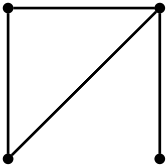
Observation

The SDP and the vector program are equivalent.

MaxCut

MaxCut
Input: An undirected graph $G = (V, E)$
Goal: Find the cut $S \subseteq V$ that maximizes the number $|\delta(S)|$ of cut edges.

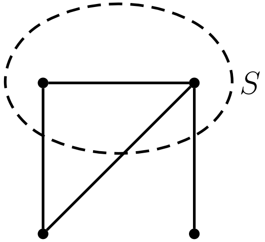
Example:



MaxCut

MaxCut
Input: An undirected graph $G = (V, E)$
Goal: Find the cut $S \subseteq V$ that maximizes the number $|\delta(S)|$ of cut edges.

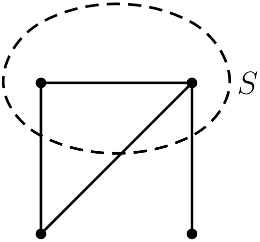
Example:



MaxCut

MaxCut
Input: An undirected graph $G = (V, E)$
Goal: Find the cut $S \subseteq V$ that maximizes the number $|\delta(S)|$ of cut edges.

Example:



- **NP**-hard to find a solution that cuts even 94% of what the optimum cuts [Hastad 1997]
- Simple greedy algorithm cuts at least $|E|/2$ edges.

MaxCut SDP

SDP:

$$\begin{array}{rcl} \max & & \frac{1}{2}\sum_{\{i,j\}\in E}(1 - X_{ij}) \\ X & \succeq & 0 \\ X_{ii} & = & 1 \quad \forall i \in V \\ X & \in & \mathbb{R}^{n\times n} \end{array}$$

Vector program

$$\begin{array}{rcl} \max & & \frac{1}{2}\sum_{\{i,j\}\in E}(1 - \langle u_i, u_j\rangle) \\ \|u_i\|_2 & = & 1 \quad \forall i \in V \\ u_i & \in & \mathbb{R}^r \end{array}$$

MaxCut SDP

SDP:

$$
\begin{aligned}
\max \quad & \frac{1}{2} \sum_{\{i,j\} \in E} (1 - X_{ij}) \\
X &\succeq 0 \\
X_{ii} &= 1 \quad \forall i \in V \\
X &\in \mathbb{R}^{n \times n}
\end{aligned}
$$

Vector program

$$
\begin{aligned}
\max \quad & \frac{1}{2} \sum_{\{i,j\} \in E} (1 - \langle u_i, u_j \rangle) \\
\|u_i\|_2 &= 1 \quad \forall i \in V \\
u_i &\in \mathbb{R}^r
\end{aligned}
$$

Lemma

If $S^* \subseteq V$ is opt. solution for MaxCut, then $SDP \geq |\delta(S^*)|$.

MaxCut SDP

SDP:

$$\begin{array}{rcl} \max & & \frac{1}{2} \sum_{\{i,j\} \in E} (1 - X_{ij}) \\ X & \succeq & 0 \\ X_{ii} & = & 1 \quad \forall i \in V \\ X & \in & \mathbb{R}^{n \times n} \end{array}$$

Vector program

$$\begin{array}{rcl} \max & & \frac{1}{2} \sum_{\{i,j\} \in E} (1 - \langle u_i, u_j \rangle) \\ \|u_i\|_2 & = & 1 \quad \forall i \in V \\ u_i & \in & \mathbb{R}^r \end{array}$$

Lemma

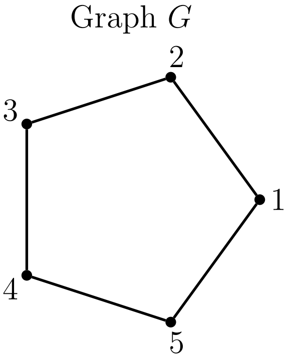
If $S^* \subseteq V$ is opt. solution for MaxCut, then $SDP \geq |\delta(S^*)|$.

Proof:

- We set $r := 1$ and define $u_i \in \mathbb{R}^1$ by

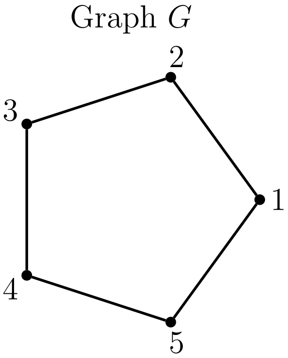
$$u_i := \begin{cases} 1 & \text{if } i \in S^* \\ -1 & \text{if } i \in V \setminus S^* \end{cases}$$

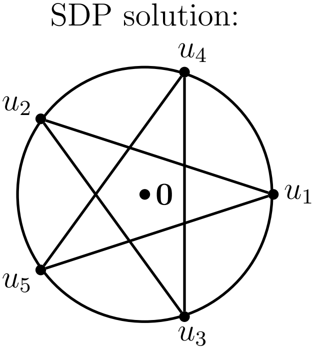
Example MaxCut SDP



- Optimum MaxCut $= 4$

Example MaxCut SDP

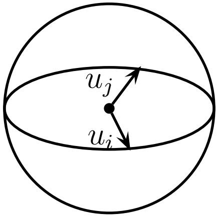




- Optimum MaxCut $= 4$
- Choose $u_i \in \mathbb{R}^2$ with $u_i := (\cos(\frac{4i\pi}{4}), \sin(\frac{4\pi}{5}))$ and we get vector program solution of value $5 \cdot \frac{1}{2}(1 - \cos(\frac{4}{5}\pi) \approx 4.522$

The Hyperplane Rounding algorithm

(1) Solve the SDP
(2) Take a random standard Gaussian $a \in \mathbb{R}^r$
(3) Define $S := \{i \in V \mid \langle a, u_i \rangle \geq 0\}$

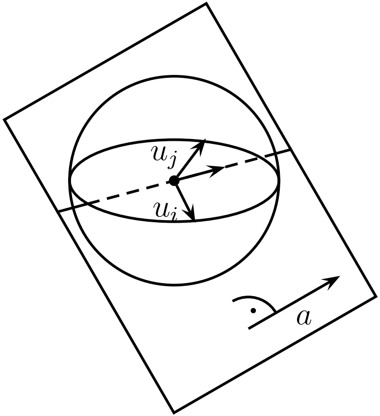


The Hyperplane Rounding algorithm

(1) Solve the SDP

(2) Take a random standard Gaussian $a \in \mathbb{R}^r$

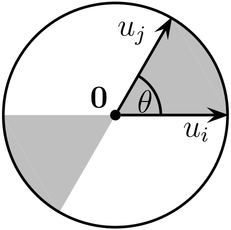
(3) Define $S := \{i \in V \mid \langle a, u_i \rangle \geq 0\}$



The Hyperplane Rounding algorithm (2)

Lemma

For $\{i,j\} \in E$ *one has* $\Pr[\{i,j\} \in \delta(S)] = \frac{1}{\pi} arccos(\langle u_i, u_j \rangle)$.



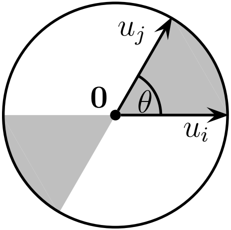
The Hyperplane Rounding algorithm (2)

Lemma

For $\{i,j\} \in E$ *one has* $\Pr[\{i,j\} \in \delta(S)] = \frac{1}{\pi} arccos(\langle u_i, u_j \rangle)$.

Proof.

- The angle between vectors is exactly $\theta := \arccos(\langle u_i, u_j \rangle)$ (as $\langle u_i, u_j \rangle = \cos(\theta)$).
- Only projection of a into $U := \text{span}\{u_i, u_j\}$ matters.



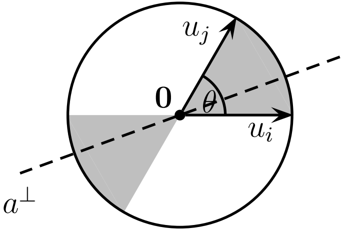
The Hyperplane Rounding algorithm (2)

Lemma

For $\{i,j\} \in E$ *one has* $\Pr[\{i,j\} \in \delta(S)] = \frac{1}{\pi} arccos(\langle u_i, u_j \rangle)$.

Proof.

- The angle between vectors is exactly $\theta := \arccos(\langle u_i, u_j \rangle)$ (as $\langle u_i, u_j \rangle = \cos(\theta)$).
- Only projection of a into $U := \text{span}\{u_i, u_j\}$ matters.
- Then $\Pr[u_i, u_j \text{ separated}] = \frac{2\theta}{2\pi}$.



The Hyperplane Rounding algorithm (3)

Theorem

One has $\mathbb{E}[|\delta(S)|] \geq 0.878 \cdot SDP \geq 0.878 \cdot |\delta(S^*)|$.

The Hyperplane Rounding algorithm (3)

Theorem

One has $\mathbb{E}[|\delta(S)|] \geq 0.878 \cdot SDP \geq 0.878 \cdot |\delta(S^*)|$.

- By **linearity of expectation** it suffices to show that every edge $\{i, j\} \in E$ one has

  $\Pr[\{i, j\} \in \delta(S)] \geq \frac{1}{2}(1 - \langle u_i, u_j \rangle) =$ contribution to SDP obj.fct

The Hyperplane Rounding algorithm (3)

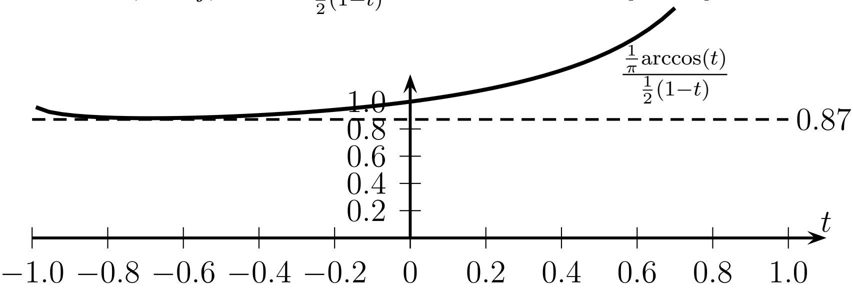
Theorem

One has $\mathbb{E}[|\delta(S)|] \geq 0.878 \cdot SDP \geq 0.878 \cdot |\delta(S^*)|$.

- By **linearity of expectation** it suffices to show that every edge $\{i,j\} \in E$ one has

  $\Pr[\{i,j\} \in \delta(S)] \geq \frac{1}{2}(1 - \langle u_i, u_j \rangle) =$ contribution to SDP obj.fct

- Set $t := \langle u_i, u_j \rangle$ and $\frac{\frac{1}{\pi}\arccos(t)}{\frac{1}{2}(1-t)} \geq 0.878 \quad \forall t \in [-1,1]$



Lecture 16

Semidefinite programming — Part 2/2

Grothendieck's Inequality

For a matrix $A \in \mathbb{R}^{m \times n}$ define

$$\begin{aligned} INT(A) &:= \max\left\{ \sum_{i=1}^{m}\sum_{j=1}^{n} A_{ij} x_i y_j \mid x \in \{-1,1\}^m, y \in \{-1,1\}^n \right\} \\ SDP(A) &:= \max\left\{ \sum_{i=1}^{m}\sum_{j=1}^{n} A_{ij} \langle u_i, v_j \rangle \mid \|u_i\|_2 = \|v_j\|_2 = 1 \right\} \end{aligned}$$

Grothendieck's Inequality

For a matrix $A \in \mathbb{R}^{m \times n}$ define

$$INT(A) \quad := \quad \max\left\{ \sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij} x_i y_j \mid x \in \{-1,1\}^m, y \in \{-1,1\}^n \right\}$$

$$SDP(A) \quad := \quad \max\left\{ \sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij} \langle u_i, v_j \rangle \mid \|u_i\|_2 = \|v_j\|_2 = 1 \right\}$$

Theorem (Grothendieck's Inequality)

For any matrix $A \in \mathbb{R}^{m \times n}$ one has

$$INT(A) \leq SDP(A) \leq C_G \cdot INT(A)$$

where $C_G \leq 1.783$.

- Grothendieck proved that C_G is indeed a constant
- [Krivine 1979] proved that $C_G \leq 1.783$

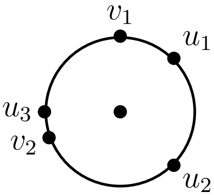
Hyperplane rounding

Random experiment:

(1) Given vectors $u_i, v_j \in \mathbb{R}^r$.

(2) Sample a **Gaussian** g in $\mathbb{R}^r$ and set

$$x_i := \operatorname{sign}(\langle u_i, g\rangle) \quad \text{and} \quad y_j := \operatorname{sign}(\langle v_j, g\rangle)$$



- Recall that

$$\operatorname{sign}(z) := \begin{cases} 1 & \text{if } z \geq 0 \\ -1 & \text{if } z < 0 \end{cases}$$

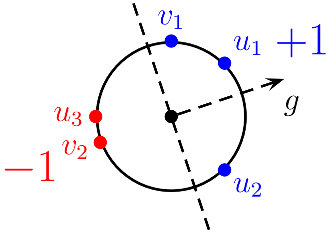
Hyperplane rounding

Random experiment:

(1) Given vectors $u_i, v_j \in \mathbb{R}^r$.

(2) Sample a **Gaussian** g in $\mathbb{R}^r$ and set

$$x_i := \mathrm{sign}(\langle u_i, g\rangle) \quad \text{and} \quad y_j := \mathrm{sign}(\langle v_j, g\rangle)$$



- Recall that

$$\mathrm{sign}(z) := \begin{cases} 1 & \text{if } z \geq 0 \\ -1 & \text{if } z < 0 \end{cases}$$

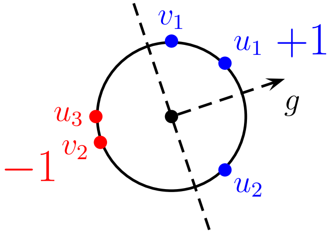
Hyperplane rounding

Random experiment:

(1) Given vectors $u_i, v_j \in \mathbb{R}^r$.

(2) Sample a **Gaussian** g in $\mathbb{R}^r$ and set

$$x_i := \operatorname{sign}(\langle u_i, g\rangle) \quad \text{and} \quad y_j := \operatorname{sign}(\langle v_j, g\rangle)$$



- Recall that

$$\operatorname{sign}(z) := \begin{cases} 1 & \text{if } z \geq 0 \\ -1 & \text{if } z < 0 \end{cases}$$

- **Question:** How does $\mathbb{E}[A_{ij} x_i y_j]$ relate to $A_{ij}\langle u_i, v_j\rangle$?

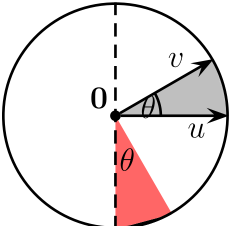
Hyperplane rounding (2)

Lemma

Let $u, v \in \mathbb{R}^r$ with $\|u\|_2 = \|v\|_2 = 1$. Then

$$\mathop{\mathbb{E}}_{g\ Gaussian}\Big[sign(\langle g, u\rangle) \cdot sign(\langle g, v\rangle)\Big] = \frac{2}{\pi} arcsin(\langle u, v\rangle)$$

- In words: Probability that u, v end up on the same side of a hyperplane is exactly $\frac{2}{\pi}\text{arcsin}(\langle u, v\rangle)$



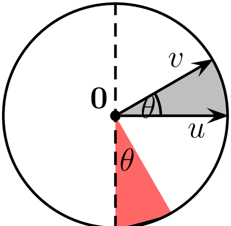
Hyperplane rounding (2)

Lemma

Let $u, v \in \mathbb{R}^r$ with $\|u\|_2 = \|v\|_2 = 1$. Then

$$\mathop{\mathbb{E}}_{g\ Gaussian}\left[sign(\langle g, u\rangle) \cdot sign(\langle g, v\rangle)\right] = \frac{2}{\pi} arcsin(\langle u, v\rangle)$$

- In words: Probability that u, v end up on the same side of a hyperplane is exactly $\frac{2}{\pi}\arcsin(\langle u, v\rangle)$
- Set $\cos(\theta) = \langle u, v\rangle$. Then $\Pr[u, v \text{ separated}] = \frac{\theta}{\pi}$



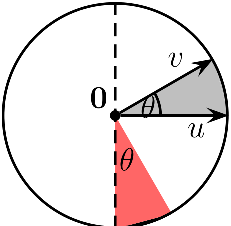
Hyperplane rounding (2)

Lemma

Let $u, v \in \mathbb{R}^r$ with $\|u\|_2 = \|v\|_2 = 1$. Then

$$\mathop{\mathbb{E}}_{g \text{ Gaussian}}\left[sign(\langle g, u\rangle) \cdot sign(\langle g, v\rangle)\right] = \frac{2}{\pi} arcsin(\langle u, v\rangle)$$

- In words: Probability that u, v end up on the same side of a hyperplane is exactly $\frac{2}{\pi}\arcsin(\langle u, v\rangle)$
- Set $\cos(\theta) = \langle u, v\rangle$. Then $\Pr[u, v \text{ separated}] = \frac{\theta}{\pi}$
- $\mathbb{E}[\cdots] = 1 - 2\Pr[u, v \text{ separated}] = 1 - \frac{2\theta}{\pi} = \frac{2}{\pi}\arcsin(\langle u, v\rangle)$
- Recall: $\arccos(t) = \frac{\pi}{2} - \arcsin(t)$

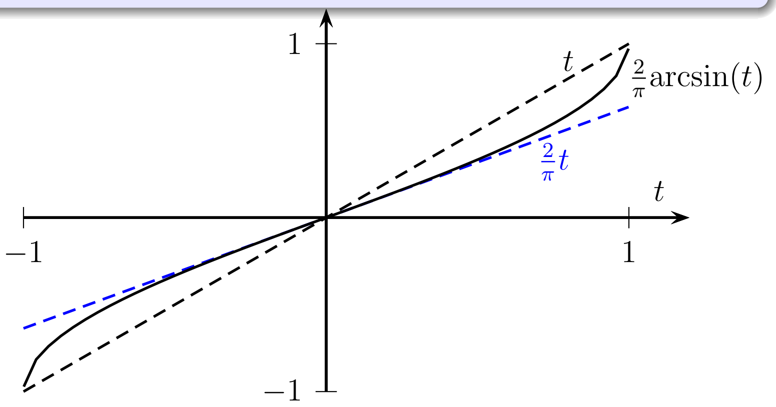


Hyperplane rounding (3)

Lemma

Let $u, v \in \mathbb{R}^r$ with $\|u\|_2 = \|v\|_2 = 1$. Then

$$\mathop{\mathbb{E}}_{g \text{ Gaussian}} \left[sign(\langle g, u \rangle) \cdot sign(\langle g, v \rangle) \right] = \frac{2}{\pi} arcsin(\langle u, v \rangle)$$



For $t \geq 0$, $\frac{2}{\pi}t \leq \frac{2}{\pi}\arcsin(t) \leq t$

Preliminary conclusion

We can conclude that:

- For $A_{ij} \geq 0$ and $\langle u_i, u_j \rangle \geq 0$ one has $\mathbb{E}[A_{ij} x_i y_j] \geq \frac{2}{\pi} \cdot A_{ij} \langle u_i, v_j \rangle$
- For $A_{ij} < 0$ and $\langle u_i, u_j \rangle \geq 0$ one has $\mathbb{E}[A_{ij} x_i y_j] \geq A_{ij} \langle u_i, v_j \rangle$

Preliminary conclusion

We can conclude that:

- For $A_{ij} \geq 0$ and $\langle u_i, u_j \rangle \geq 0$ one has $\mathbb{E}[A_{ij} x_i y_j] \geq \frac{2}{\pi} \cdot A_{ij} \langle u_i, v_j \rangle$
- For $A_{ij} < 0$ and $\langle u_i, u_j \rangle \geq 0$ one has $\mathbb{E}[A_{ij} x_i y_j] \geq A_{ij} \langle u_i, v_j \rangle$

Problem: Due to the non-linearity, this does bound $INT(A)$ in terms of $SDP(A)$!!

Tensors

Definition

A **kth order tensor** $A \in \mathbb{R}^{n_1 \times \ldots \times n_k}$ is a k-dimensional array of numbers; we write $A = (A_{i_1,\ldots,i_k})_{i_1 \in [n_1],\ldots,i_k \in [n_k]}$.

- A vector $a \in \mathbb{R}^n$ is a 1-order tensor
- A matrix $A \in \mathbb{R}^{n_1 \times n_2}$ is a 2-order tensor

Tensors

Definition

A k**th order tensor** $A \in \mathbb{R}^{n_1 \times \ldots \times n_k}$ is a k-dimensional array of numbers; we write $A = (A_{i_1,\ldots,i_k})_{i_1 \in [n_1],\ldots,i_k \in [n_k]}$.

- A vector $a \in \mathbb{R}^n$ is a 1-order tensor
- A matrix $A \in \mathbb{R}^{n_1 \times n_2}$ is a 2-order tensor
- For two tensors $A, B \in \mathbb{R}^{n_1 \times \ldots \times n_k}$ we can define an **inner product**

$$\langle A, B \rangle := \sum_{i_1,\ldots,i_k} A_{i_1,\ldots,i_k} \cdot B_{i_1,\ldots,i_k}$$

Tensors

Definition

A **kth order tensor** $A \in \mathbb{R}^{n_1 \times \ldots \times n_k}$ is a k-dimensional array of numbers; we write $A = (A_{i_1,\ldots,i_k})_{i_1 \in [n_1],\ldots,i_k \in [n_k]}$.

- A vector $a \in \mathbb{R}^n$ is a 1-order tensor
- A matrix $A \in \mathbb{R}^{n_1 \times n_2}$ is a 2-order tensor
- For two tensors $A, B \in \mathbb{R}^{n_1 \times \ldots \times n_k}$ we can define an **inner product**

$$\langle A, B \rangle := \sum_{i_1,\ldots,i_k} A_{i_1,\ldots,i_k} \cdot B_{i_1,\ldots,i_k}$$

- For vector $u \in \mathbb{R}^n$ and $k \in \mathbb{N}$, define the **tensor product**

$$u \otimes \ldots \otimes u := u^{\otimes k} := (u_{i_1} \cdot \ldots \cdot u_{i_k})_{i_1 \in [n],\ldots,i_k \in [n]}$$

Tensors

Definition

A **kth order tensor** $A \in \mathbb{R}^{n_1 \times \ldots \times n_k}$ is a k-dimensional array of numbers; we write $A = (A_{i_1,\ldots,i_k})_{i_1 \in [n_1],\ldots,i_k \in [n_k]}$.

- A vector $a \in \mathbb{R}^n$ is a 1-order tensor
- A matrix $A \in \mathbb{R}^{n_1 \times n_2}$ is a 2-order tensor
- For two tensors $A, B \in \mathbb{R}^{n_1 \times \ldots \times n_k}$ we can define an **inner product**
$$\langle A, B \rangle := \sum_{i_1,\ldots,i_k} A_{i_1,\ldots,i_k} \cdot B_{i_1,\ldots,i_k}$$
- For vector $u \in \mathbb{R}^n$ and $k \in \mathbb{N}$, define the **tensor product**
$$u \otimes \ldots \otimes u := u^{\otimes k} := (u_{i_1} \cdot \ldots \cdot u_{i_k})_{i_1 \in [n],\ldots,i_k \in [n]}$$
- **Fact:** For vectors $u, v \in \mathbb{R}^n$ one has $\langle u^{\otimes k}, v^{\otimes k} \rangle = \langle u, v \rangle^k$.

Tensors

Definition

We call a function $f : \mathbb{R} \to \mathbb{R}$ **(real) analytic** if it can be written as a convergent power series $f(x) = \sum_{k=0}^{\infty} a_k x^k$ for all $x \in \mathbb{R}$.

Tensors

Definition

We call a function $f : \mathbb{R} \to \mathbb{R}$ **(real) analytic** if it can be written as a convergent power series $f(x) = \sum_{k=0}^{\infty} a_k x^k$ for all $x \in \mathbb{R}$.

- For fixed r, we can define a **Hilbert space / infinite-dimensional vector space** of the form

$$H = \{(U^0, U^1, U^2, U^3, \ldots) \mid U^k \text{ is a } k\text{-tensor of size } r^k\}$$

using the natural inner product.

A vector transformation

Now we can "bend" any vectors to give any analytic function that we like:

Lemma

Let $f(x) = \sum_{k=0}^{\infty} a_k x^k$ and fix a dimension $r \in \mathbb{N}$. Then there is a Hilbert space H and maps $F, G : \mathbb{R}^r \to H$ so that

$$\langle F(u), G(v) \rangle = f(\langle u, v \rangle) \quad \forall u, v \in \mathbb{R}^r$$

Moreover the length of the mapped vectors satisfies

$$\|F(u)\|_2^2 = \|G(u)\|_2^2 = \sum_{k=0}^{\infty} |a_k| \cdot \|u\|_2^{2k}$$

A vector transformation (2)

Proof:

- The maps are

$$F(u) := (\sqrt{|a_k|} \cdot u^{\otimes k})_{k \in \mathbb{Z}_{\geq 0}}, \quad G(u) := (\mathrm{sign}(a_k) \cdot \sqrt{|a_k|} \cdot u^{\otimes k})_{k \in \mathbb{Z}_{\geq 0}}$$

A vector transformation (2)

Proof:

- The maps are

$$F(u) := (\sqrt{|a_k|}\cdot u^{\otimes k})_{k\in\mathbb{Z}_{\geq 0}}, \quad G(u) := (\mathrm{sign}(a_k)\cdot\sqrt{|a_k|}\cdot u^{\otimes k})_{k\in\mathbb{Z}_{\geq 0}}$$

- Then for vectors $u, v \in \mathbb{R}^r$ one has

$$\begin{aligned}\langle F(u), G(v)\rangle &= \sum_{k\geq 0} \mathrm{sign}(a_k)\cdot(\sqrt{|a_k|})^2\cdot\left\langle u^{\otimes k}, v^{\otimes k}\right\rangle \\ &= \sum_{k\geq 0} a_k\cdot\langle u, v\rangle^k = f(\langle u, v\rangle).\end{aligned}$$

A vector transformation (2)

Proof:

- The maps are

$$F(u) := (\sqrt{|a_k|} \cdot u^{\otimes k})_{k \in \mathbb{Z}_{\geq 0}}, \quad G(u) := (\mathrm{sign}(a_k) \cdot \sqrt{|a_k|} \cdot u^{\otimes k})_{k \in \mathbb{Z}_{\geq 0}}$$

- Then for vectors $u, v \in \mathbb{R}^r$ one has

$$\begin{aligned} \langle F(u), G(v) \rangle &= \sum_{k \geq 0} \mathrm{sign}(a_k) \cdot (\sqrt{|a_k|})^2 \cdot \left\langle u^{\otimes k}, v^{\otimes k} \right\rangle \\ &= \sum_{k \geq 0} a_k \cdot \langle u, v \rangle^k = f(\langle u, v \rangle). \end{aligned}$$

- We can verify that the lengths are

$$\|F(u)\|_2^2 = \|G(u)\|_2^2 = \sum_{k \geq 0} (\sqrt{|a_k|})^2 \cdot \|u^{\otimes k}\|_2^2 = \sum_{k \geq 0} |a_k| \cdot \|u\|_2^{2k}$$

as claimed. $\square$

Applying the vector transformation

Lemma

Let $r \in \mathbb{N}$. Then there are maps $F, G : \mathbb{R}^r \to H$ so that

$$\langle F(u), G(v) \rangle = \sin\left(\beta \frac{\pi}{2} \langle u, v \rangle \right)$$

where $\beta = \frac{2}{\pi}\ln(1+\sqrt{2}) \approx \frac{1}{1.783}$. Moreover $\|F(u)\|_2^2 = \|G(u)\|_2^2 = 1$ for all $u \in \mathbb{R}^r$ with $\|u\|_2^2 = 1$.

Note that this is equivalent to

$$\frac{2}{\pi}\arcsin(\langle F(u), G(v) \rangle) = \beta \cdot \langle u, v \rangle$$

Applying the vector transformation (2)

Proof.

- Consider $f(x) = \sin(\beta \frac{\pi}{2} x)$.

Applying the vector transformation (2)

Proof.

- Consider $f(x) = \sin(\beta \frac{\pi}{2} x)$.
- Recall that
  $\sin(x) = \sum_{k \geq 0} \frac{(-1)^k}{(2k+1)!} x^{2k+1} = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \dots$
  $\sinh(x) = \sum_{k \geq 0} \frac{1}{(2k+1)!} x^{2k+1}$

Applying the vector transformation (2)

Proof.

- Consider $f(x) = \sin(\beta \frac{\pi}{2} x)$.
- Recall that
  $\sin(x) = \sum_{k \geq 0} \frac{(-1)^k}{(2k+1)!} x^{2k+1} = x - \frac{1}{3!} x^3 + \frac{1}{5!} x^5 - \ldots$
  $\sinh(x) = \sum_{k \geq 0} \frac{1}{(2k+1)!} x^{2k+1}$
- Then for $\|u\|_2 = 1$,

$$\|F(u)\|_2^2 = \sum_{k \geq 0} \left| \frac{(-1)^k}{(2k+1)!} \cdot \left(\beta \frac{\pi}{2}\right)^{2k+1} \right| = \sinh\left(\beta \frac{\pi}{2}\right) \overset{\beta := \frac{2}{\pi} \operatorname{arcsinh}(1)}{=} 1$$

Applying the vector transformation (2)

Proof.

- Consider $f(x) = \sin(\beta \frac{\pi}{2} x)$.
- Recall that
  $\sin(x) = \sum_{k \geq 0} \frac{(-1)^k}{(2k+1)!} x^{2k+1} = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \ldots$
  $\sinh(x) = \sum_{k \geq 0} \frac{1}{(2k+1)!} x^{2k+1}$
- Then for $\|u\|_2 = 1$,

$$\|F(u)\|_2^2 = \sum_{k \geq 0} \left| \frac{(-1)^k}{(2k+1)!} \cdot \left(\beta \frac{\pi}{2} \right)^{2k+1} \right| = \sinh\left(\beta \frac{\pi}{2} \right) \overset{\beta := \frac{2}{\pi} \operatorname{arcsinh}(1)}{=} 1$$

- One can check that

$$\beta = \frac{2}{\pi} \operatorname{arcsinh}(1) = \frac{2}{\pi} \ln(1 + \sqrt{2}) \approx \frac{1}{1.783}.$$

Applying the vector transformation (3)

- Consider $A \in \mathbb{R}^{m \times n}$ and $u_i, v_j \in \mathbb{R}^r$ with $\|u_i\|_2 = 1 = \|v_j\|_2$.
- Sample a **Gaussian** g in H and set

$$x_i := \operatorname{sign}(\langle g, F(u_i) \rangle) \quad \text{and} \quad y_j := \operatorname{sign}(\langle g, G(v_j) \rangle)$$

Applying the vector transformation (3)

- Consider $A \in \mathbb{R}^{m \times n}$ and $u_i, v_j \in \mathbb{R}^r$ with $\|u_i\|_2 = 1 = \|v_j\|_2$.
- Sample a **Gaussian** g in H and set

$$x_i := \text{sign}(\langle g, F(u_i) \rangle) \quad \text{and} \quad y_j := \text{sign}(\langle g, G(v_j) \rangle)$$

- Then

$$\mathbb{E}[x_i y_j] = \frac{2}{\pi} \arcsin(\langle F(u_i), G(v_i) \rangle) = \beta \cdot \langle u_i, v_i \rangle$$

Applying the vector transformation (3)

- Consider $A \in \mathbb{R}^{m \times n}$ and $u_i, v_j \in \mathbb{R}^r$ with $\|u_i\|_2 = 1 = \|v_j\|_2$.
- Sample a **Gaussian** g in H and set

$$x_i := \text{sign}(\langle g, F(u_i) \rangle) \quad \text{and} \quad y_j := \text{sign}(\langle g, G(v_j) \rangle)$$

- Then

$$\mathbb{E}[x_i y_j] = \frac{2}{\pi} \arcsin(\langle F(u_i), G(v_i) \rangle) = \beta \cdot \langle u_i, v_i \rangle$$

- By linearity of expectation

$$\mathbb{E}\left[\sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij} x_i y_j \right] = \underbrace{\beta}_{\approx \frac{1}{1.783}} \sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij} \langle u_i, v_j \rangle \qquad \square$$

Lecture 17

Matroid Intersection — Part 1/2

Recap: Matroids

Definition

Given a **ground set** X and a family of **independent sets** $\mathcal{I} \subseteq 2^X$, $M = (X, \mathcal{I})$ is called a **matroid** if

1. **Non-emptyness:** $\emptyset \in \mathcal{I}$
2. **Monotonicity:** If $Y \in \mathcal{I}$ and $Z \subseteq Y$, then $Z \in \mathcal{I}$
3. **Exchange property:** If $Y, Z \in \mathcal{I}$ with $|Y| < |Z|$, then there is an $x \in Z/Y$ so that $Y \cup \{x\} \in \mathcal{I}$

Recap matroids (2)

Examples of matroids:

- The set of forests in an undirected graph form a **graphical matroid**.

Recap matroids (2)

Examples of matroids:

- The set of forests in an undirected graph form a **graphical matroid**.
- If $v_1, \ldots, v_n$ are vectors in a vector space, and set $\mathcal{I} = \{I \subseteq [n] \mid \{v_i\}_{i \in I} \text{ linearly independent}\}$ then $M = ([n], \mathcal{I})$ is a **linear matroid**.

Recap matroids (2)

Examples of matroids:

- The set of forests in an undirected graph form a **graphical matroid**.
- If $v_1, \ldots, v_n$ are vectors in a vector space, and set $\mathcal{I} = \{I \subseteq [n] \mid \{v_i\}_{i \in I} \text{ linearly independent}\}$ then $M = ([n], \mathcal{I})$ is a **linear matroid**.
- A **partition matroid** with ground set X can be obtained as follows: take any partition $X = B_1 \dot\cup \ldots \dot\cup B_m$ and select numbers $d_i \in \{0, \ldots, |B_i|\}$. Then $M = (X, \mathcal{I})$ with $\mathcal{I} := \{S : |S \cap B_i| \leq d_i \text{ for all } i = 1, \ldots, m\}$ is a matroid.

Recap matroids (3)

We have implicitly seen the following:

Lemma

Let $M = (X, \mathcal{I})$ be a matroid, $Z \subseteq Y \subseteq X$ with $Z \in \mathcal{I}$. Then there is a set S so that $Z \subseteq S \subseteq Y$ and S is a basis of Y.

Matroid intersection

Matroid Intersection

Input: Matroid $M_1 = (X, \mathcal{I}_1)$, $M_2 = (X, \mathcal{I}_2)$ on the same groundset

Goal: Find $\max\{\, |I| : I \in \mathcal{I}_1 \cap \mathcal{I}_2\}$

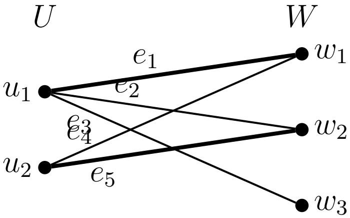
Matroid intersection (2)

Obs: Maximum cardinality bipartite matching is a special case of Matroid intersection!

Matroid intersection (2)

Obs: Maximum cardinality bipartite matching is a special case of Matroid intersection!

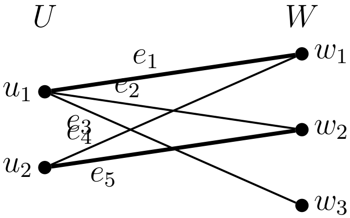
- Consider bipartite graph $G = (V, E)$ with partitions $V = U \cup W = \{u_1, \ldots, u_{|U|}\} \cup \{w_1, \ldots, w_{|W|}\}$.

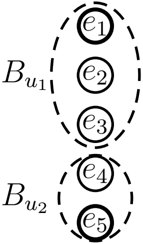


Matroid intersection (2)

Obs: Maximum cardinality bipartite matching is a special case of Matroid intersection!

- Consider bipartite graph $G = (V, E)$ with partitions $V = U \cup W = \{u_1, \dots, u_{|U|}\} \cup \{w_1, \dots, w_{|W|}\}$.
- Define $M_1 = (E, \mathcal{I}_1)$ as **partition matroid** with partitions $\delta(u_1), \dots, \delta(u_{|U|})$ and $d_i := 1$.

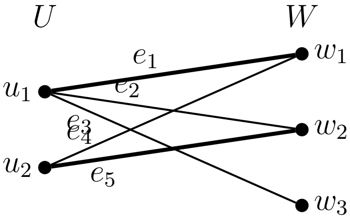


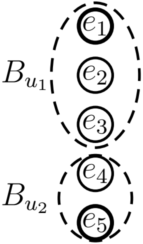


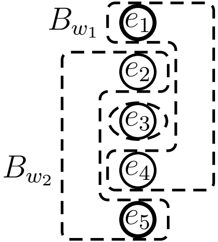
Matroid intersection (2)

Obs: Maximum cardinality bipartite matching is a special case of Matroid intersection!

- Consider bipartite graph $G = (V, E)$ with partitions $V = U \cup W = \{u_1, \ldots, u_{|U|}\} \cup \{w_1, \ldots, w_{|W|}\}$.
- Define $M_1 = (E, \mathcal{I}_1)$ as **partition matroid** with partitions $\delta(u_1), \ldots, \delta(u_{|U|})$ and $d_i := 1$.
- Define $M_2 = (E, \mathcal{I}_2)$ as **partition matroid** with partitions $\delta(w_1), \ldots, \delta(w_{|W|})$, $d_i := 1$.



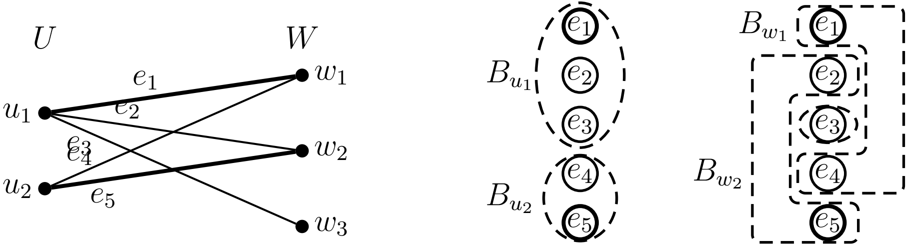




Matroid intersection (2)

Obs: Maximum cardinality bipartite matching is a special case of Matroid intersection!

- Consider bipartite graph $G = (V, E)$ with partitions $V = U \cup W = \{u_1, \dots, u_{|U|}\} \cup \{w_1, \dots, w_{|W|}\}$.
- Define $M_1 = (E, \mathcal{I}_1)$ as **partition matroid** with partitions $\delta(u_1), \dots, \delta(u_{|U|})$ and $d_i := 1$.
- Define $M_2 = (E, \mathcal{I}_2)$ as **partition matroid** with partitions $\delta(w_1), \dots, \delta(w_{|W|})$, $d_i := 1$.
- Matroid intersection = select max # edges, s.t. in each $\delta(u_i)$ and $\delta(w_j)$ we select at most one edge.



The exchange lemma (1)

We know so far:

- For two spanning trees T_1, T_2 in a graph, there is a map $f : E(T_1) \to E(T_2)$ so that $(T_1 \setminus \{e\}) \cup \{f(e)\}$ is a spanning tree for all $e \in E(T_1)$!

The exchange lemma (1)

We know so far:

- For two spanning trees T_1, T_2 in a graph, there is a map $f : E(T_1) \to E(T_2)$ so that $(T_1 \setminus \{e\}) \cup \{f(e)\}$ is a spanning tree for all $e \in E(T_1)$!

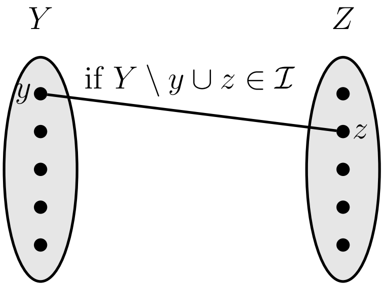
Here we will prove a stronger result:

- For two spanning trees T_1, T_2 in a graph, there is a **bijective** map $f : E(T_1) \to E(T_2)$ so that $(T_1 \setminus \{e\}) \cup \{f(e)\}$ is a spanning tree for all $e \in E(T_1)$!

The exchange lemma (2)

Lemma

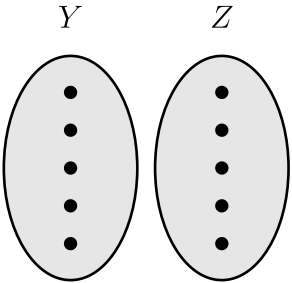
Let $M = (X, \mathcal{I})$ be a matroid and let $Y, Z \in \mathcal{I}$ be disjoint independent sets of the same size. Define a bipartite exchange graph $H = (Y \cup Z, E)$ with $E = \{(y,z) : (Y \setminus y) \cup z \in \mathcal{I}\}$. Then H contains a perfect matching.



The exchange lemma (3)

Proof.

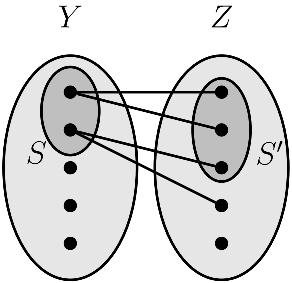
- Suppose by contradiction, H has no perfect matching.



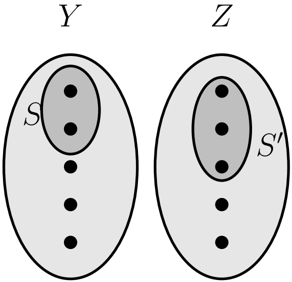
The exchange lemma (3)

Proof.

- Suppose by contradiction, H has no perfect matching.
- By **Hall's condition**, there must be subsets $S \subseteq Y$ and $S' \subseteq Z$ so that $N(S') \subseteq S$ where $|S| < |S'|$.

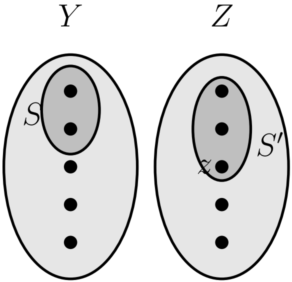


The exchange lemma (4)



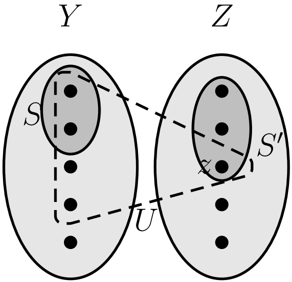
The exchange lemma (4)

- Since $|S| < |S'|$ and $S, S' \in \mathcal{I}$, there is a $z \in S'$ s.t. $S \cup \{z\} \in \mathcal{I}$.



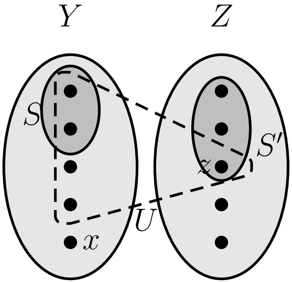
The exchange lemma (4)

- Since $|S| < |S'|$ and $S, S' \in \mathcal{I}$, there is a $z \in S'$ s.t. $S \cup \{z\} \in \mathcal{I}$.
- We can keep adding elements from Y to $S \cup \{z\}$ until we get a set $U \subseteq Y \cup \{z\}$ with $|U| = |Y|$.



The exchange lemma (4)

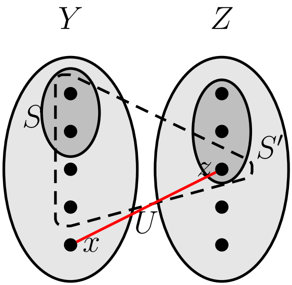
- Since $|S| < |S'|$ and $S, S' \in \mathcal{I}$, there is a $z \in S'$ s.t. $S \cup \{z\} \in \mathcal{I}$.
- We can keep adding elements from Y to $S \cup \{z\}$ until we get a set $U \subseteq Y \cup \{z\}$ with $|U| = |Y|$.



- There is exactly one element in $Y \setminus U$; we call it x.

The exchange lemma (4)

- Since $|S| < |S'|$ and $S, S' \in \mathcal{I}$, there is a $z \in S'$ s.t. $S \cup \{z\} \in \mathcal{I}$.
- We can keep adding elements from Y to $S \cup \{z\}$ until we get a set $U \subseteq Y \cup \{z\}$ with $|U| = |Y|$.



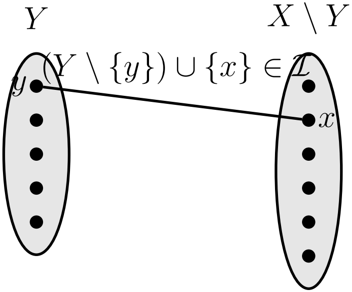
- There is exactly one element in $Y \setminus U$; we call it x.
- Then $(Y/\{x\}) \cup \{z\} = U \in \mathcal{I}$ and $(x, z) \in E$ would be an edge — a contradiction. □

The exchange graph

Definition

For a matroid $M = (X, \mathcal{I})$ and an independent set $Y \in \mathcal{I}$, we define the **exchange graph** $H(M, Y)$ as the bipartite graph with partitions Y and $X \setminus Y$ where we have an edge between $y \in Y$ and $x \in X \setminus Y$ if

$$(Y \setminus \{y\}) \cup \{x\} \in \mathcal{I}.$$



The rank function

Definition

For a matroid $M = (X, \mathcal{I})$ we define the **rank function** $r_M : 2^X \to \mathbb{Z}_{\geq 0}$ by

$$r_M(Y) := \max\{|S| : S \subseteq Y \text{ and } S \in \mathcal{I}\}$$

- Recall that all bases of Y have the same cardinality of $r_M(Y)$.

The rank function (2)

Lemma

Let $M_1 = (X, \mathcal{I}_1)$, $M_2 = (X, \mathcal{I}_2)$ *with rank functions* r_1 *and* r_2. *Then for any independent set* $Y \in \mathcal{I}_1 \cap \mathcal{I}_2$ *and any set* $U \subseteq X$ *one has*

$$|Y| \leq r_1(U) + r_2(X/U).$$

The rank function (2)

Lemma

Let $M_1 = (X, \mathcal{I}_1)$, $M_2 = (X, \mathcal{I}_2)$ with rank functions r_1 and r_2. Then for any independent set $Y \in \mathcal{I}_1 \cap \mathcal{I}_2$ and any set $U \subseteq X$ one has

$$|Y| \leq r_1(U) + r_2(X/U).$$

Proof.

- We have

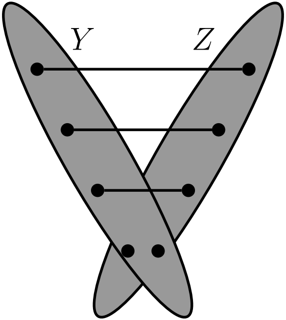
$$|Y| = \underbrace{|U \cap Y|}_{\leq r_1(U)} + \underbrace{|(X/U) \cap Y|}_{\leq r_2(X/U)} \leq r_1(U) + r_2(X/U).$$

  using that Y is an independent set in both matroid.

The reverse exchange lemma

We want a claim of the following type:

- If Y independent and there is a perfect matching between $Y \setminus Z$ and $Z \setminus Y$ in $H(M, Y)$, then Z is independent.



... but some care is needed!

The reverse exchange lemma (2)

Lemma

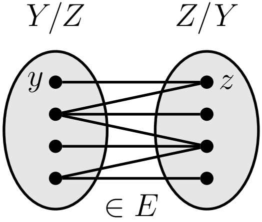
Let $M = (X, \mathcal{I})$ be a matroid and let $Y \in \mathcal{I}$ be an independent set and let $Z \subseteq X$ be any set with $|Z| = |Y|$. Suppose that there exists a <u>unique</u> perfect matching N in $H(M, Y)$ between $Y \Delta Z$. Then $Z \in \mathcal{I}$.

Proof.

- Let $E = \{(y, z) \in (Y \setminus Z) \times (Z \setminus Y) \mid (Y/y) \cup \{z\} \in \mathcal{I}\}$ be all the exchange edges between $Y \setminus Z$ and $Z \setminus Y$.

The reverse exchange lemma (3)

Claim. E has a leaf (= degree-1 node) $y \in Y/Z$.

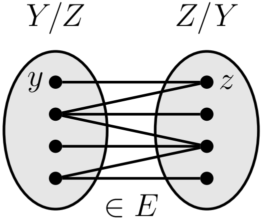


The reverse exchange lemma (3)

Claim. E has a leaf (= degree-1 node) $y \in Y/Z$.

Proof of claim.

- By assumption there is a perfect matching $N \subseteq E$.

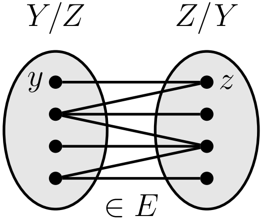


The reverse exchange lemma (3)

Claim. E has a leaf (= degree-1 node) $y \in Y/Z$.

Proof of claim.

- By assumption there is a perfect matching $N \subseteq E$.
- Start at any node $w \in Y \Delta Z$. If on the "right side" $Z \setminus Y$, move along a edge in N; if in $Y \setminus Z$, take a non-N edge.

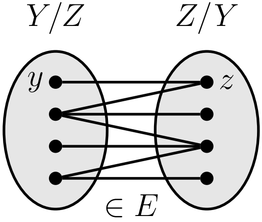


The reverse exchange lemma (3)

Claim. E has a leaf (= degree-1 node) $y \in Y/Z$.

Proof of claim.

- By assumption there is a perfect matching $N \subseteq E$.
- Start at any node $w \in Y \Delta Z$. If on the "right side" $Z \setminus Y$, move along a edge in N; if in $Y \setminus Z$, take a non-N edge.
- If we revisit a node, then we found even length N-alternating cycle $C \subseteq E \Rightarrow N \Delta C$ is also a perfect matching. Contradiction!

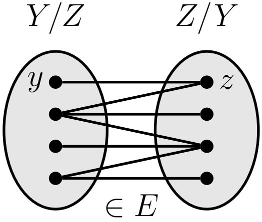


The reverse exchange lemma (3)

Claim. E has a leaf (= degree-1 node) $y \in Y/Z$.

Proof of claim.

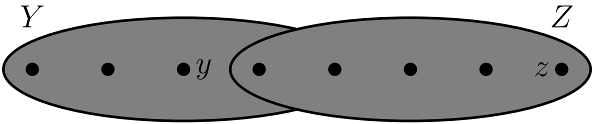
- By assumption there is a perfect matching $N \subseteq E$.
- Start at any node $w \in Y \Delta Z$. If on the "right side" $Z \setminus Y$, move along a edge in N; if in $Y \setminus Z$, take a non-N edge.
- If we revisit a node, then we found even length N-alternating cycle $C \subseteq E \Rightarrow N \Delta C$ is also a perfect matching. Contradiction!
- Walk will end in leaf. Leaf cannot be in Z/Y due to incident matching edge. Must end in leaf $y \in Y/Z$. $\square$



The reverse exchange lemma (4)

Main Proof.

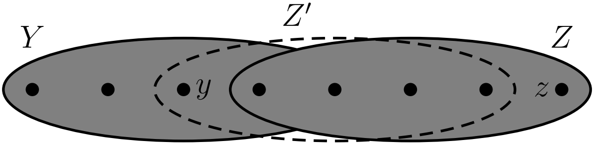
- Let z denote the element with $(y, z) \in N$ (y is leaf)



The reverse exchange lemma (4)

Main Proof.

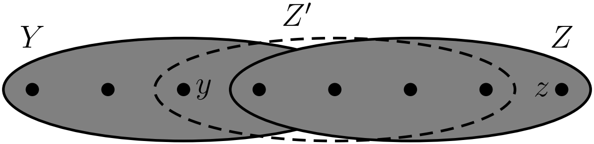
- Let z denote the element with $(y, z) \in N$ (y is leaf)
- Note that $Z' := (Z \setminus \{z\}) \cup \{y\}$ satisfies $|Y \Delta Z'| = |Y \Delta Z| - 2$ and there is still exactly one perfect matching between $Y \Delta Z'$ (which is $N \setminus \{(y, z)\}$).



The reverse exchange lemma (4)

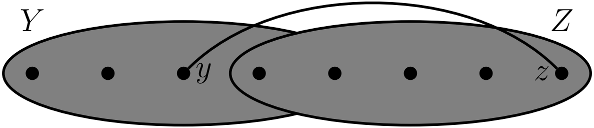
Main Proof.

- Let z denote the element with $(y, z) \in N$ (y is leaf)
- Note that $Z' := (Z \setminus \{z\}) \cup \{y\}$ satisfies $|Y \Delta Z'| = |Y \Delta Z| - 2$ and there is still exactly one perfect matching between $Y \Delta Z'$ (which is $N \setminus \{(y, z)\}$).
- Hence we can apply induction and assume that $Z' \in \mathcal{I}$. Remains to prove that also $Z \in \mathcal{I}$.



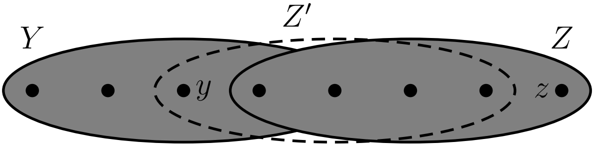
The reverse exchange lemma (4)

- We know that $r((Y \cup Z) \setminus y) \geq r((Y \setminus y) \cup \{z\}) = |Y|$.



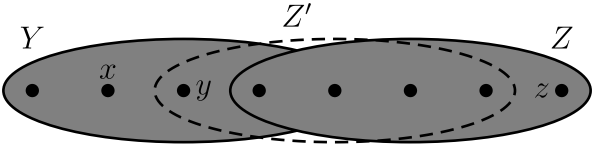
The reverse exchange lemma (4)

- We know that $r((Y \cup Z) \setminus y) \geq r((Y \setminus y) \cup \{z\}) = |Y|$.
- By the matroid exchange property, there is some element $x \in (Y \cup Z)/y$ so that $S := (Z'/y) \cup \{x\}$ is an independent set of size $|Y|$.
- If $x = z$ then $Z = S \in \mathcal{I}$ and we are done.



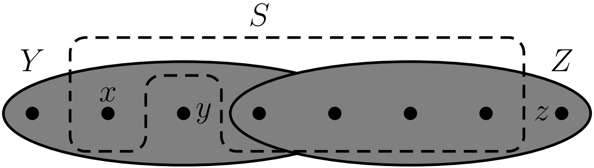
The reverse exchange lemma (4)

- We know that $r((Y \cup Z) \setminus y) \geq r((Y \setminus y) \cup \{z\}) = |Y|$.
- By the matroid exchange property, there is some element $x \in (Y \cup Z)/y$ so that $S := (Z'/y) \cup \{x\}$ is an independent set of size $|Y|$.
- If $x = z$ then $Z = S \in \mathcal{I}$ and we are done. Otherwise, $x \in Y/Z$.



The reverse exchange lemma (4)

- We know that $r((Y \cup Z) \setminus y) \geq r((Y \setminus y) \cup \{z\}) = |Y|$.
- By the matroid exchange property, there is some element $x \in (Y \cup Z)/y$ so that $S := (Z'/y) \cup \{x\}$ is an independent set of size $|Y|$.
- If $x = z$ then $Z = S \in \mathcal{I}$ and we are done. Otherwise, $x \in Y/Z$.



- As $|S| > |Y \setminus y|$, there must be an exchange edge between y and a node in S/Y. That contradicts that y is leaf. $\square$

Lecture 18

Matroid Intersection — Part 2/2

The algorithm

Matroid Intersection Augmentation subroutine:

- **Input:** Two matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ and $Y \in \mathcal{I}_1 \cap \mathcal{I}_2$.
- **Output:** Set $Y' \in \mathcal{I}_1 \cap \mathcal{I}_2$ with $|Y'| = |Y| + 1$ or decide that Y is already optimal.

The algorithm

Matroid Intersection Augmentation subroutine:

- **Input:** Two matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ and $Y \in \mathcal{I}_1 \cap \mathcal{I}_2$.
- **Output:** Set $Y' \in \mathcal{I}_1 \cap \mathcal{I}_2$ with $|Y'| = |Y| + 1$ or decide that Y is already optimal.

Actual algorithm:

(1) Starting at $Y := \emptyset$

(2) Repeat the routine until Y is maximal

The algorithm (2)

- Define

$$
\begin{aligned}
X_1 &:= \{y \in X \setminus Y \mid Y \cup \{y\} \in \mathcal{I}_1\} = \begin{pmatrix}\text{elements that can} \\ \text{be added w.r.t } M_1\end{pmatrix} \\
X_2 &:= \{y \in X \setminus Y \mid Y \cup \{y\} \in \mathcal{I}_2\} = \begin{pmatrix}\text{elements that can} \\ \text{be added w.r.t } M_2\end{pmatrix}
\end{aligned}
$$

The algorithm (2)

- Define

$$
\begin{aligned}
X_1 &:= \{y \in X \setminus Y \mid Y \cup \{y\} \in \mathcal{I}_1\} = \begin{pmatrix}\text{elements that can}\\ \text{be added w.r.t } M_1\end{pmatrix}\\
X_2 &:= \{y \in X \setminus Y \mid Y \cup \{y\} \in \mathcal{I}_2\} = \begin{pmatrix}\text{elements that can}\\ \text{be added w.r.t } M_2\end{pmatrix}
\end{aligned}
$$

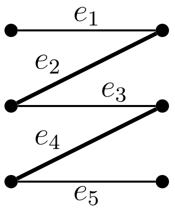
- Define directed graph $D = (X, A)$: for all $y \in Y$ and $x \in X/Y$

$$
\begin{aligned}
(y, x) \in A &\Leftrightarrow (Y/y) \cup \{x\} \in \mathcal{I}_1\\
(x, y) \in A &\Leftrightarrow (Y/y) \cup \{x\} \in \mathcal{I}_2
\end{aligned}
$$

The algorithm (3)

Example:

- M_1, M_2 are partition matroids modelling bipartite matching problem and $Y := \{e_2, e_4\}$

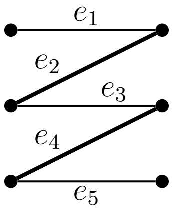


original graph

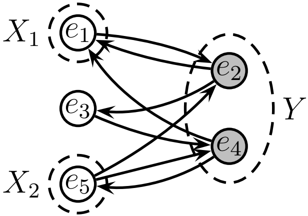
The algorithm (3)

Example:

- M_1, M_2 are partition matroids modelling bipartite matching problem and $Y := \{e_2, e_4\}$



original graph

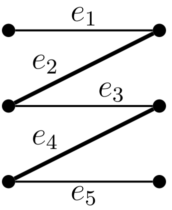


directed exchange graph D

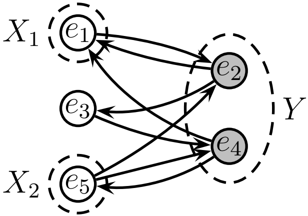
The algorithm (3)

Example:

- M_1, M_2 are partition matroids modelling bipartite matching problem and $Y := \{e_2, e_4\}$



original graph



directed exchange graph D

Observation:

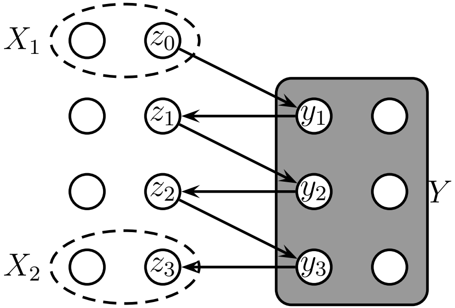
- Matching-Augmenting path corresponds to directed $X_1 - X_2$ path in D

Path existence $\Rightarrow$ augment Y (1)

Lemma

Suppose there exists a directed path $z_0, y_1, z_1, \ldots, y_m, z_m$ in D starting at a vertex $z_0 \in X_1$ and ending at a node $z_m \in X_2$. If that is a <u>shortest</u> path, then

$$Y' := (Y \setminus \{y_1, \ldots, y_m\}) \cup \{z_0, \ldots, z_m\} \in \mathcal{I}_1 \cap \mathcal{I}_2$$

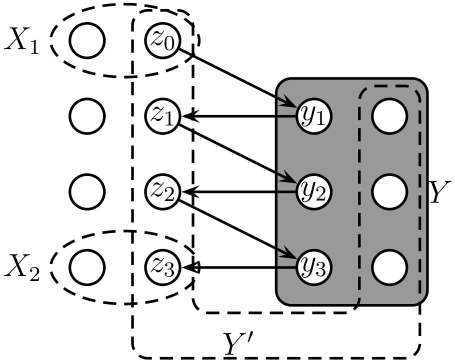


Path existence ⇒ augment Y (1)

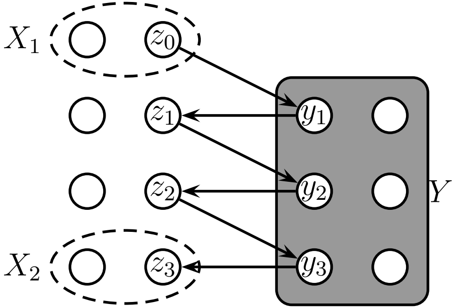
Lemma

Suppose there exists a directed path $z_0, y_1, z_1, \ldots, y_m, z_m$ in D starting at a vertex $z_0 \in X_1$ and ending at a node $z_m \in X_2$. If that is a <u>shortest</u> path, then

$$Y' := (Y \setminus \{y_1, \ldots, y_m\}) \cup \{z_0, \ldots, z_m\} \in \mathcal{I}_1 \cap \mathcal{I}_2$$

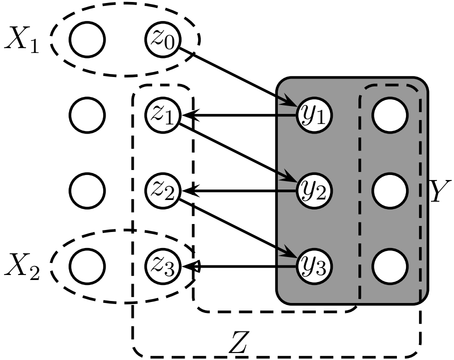


Path existence ⇒ augment Y (2)



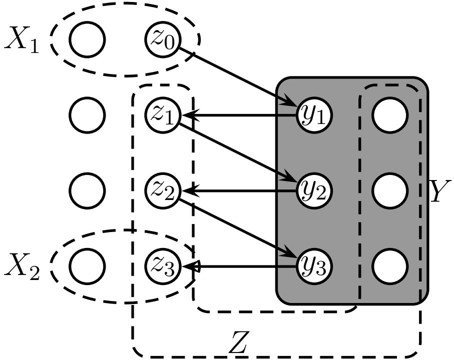
- It suffices to show $Y' \in \mathcal{I}_1$ (direction $Y' \in \mathcal{I}_2$ is similar)

Path existence ⇒ augment Y (2)



- It suffices to show $Y' \in \mathcal{I}_1$ (direction $Y' \in \mathcal{I}_2$ is similar)
- Let $Z := (Y \setminus \{y_1, \ldots, y_m\}) \cup \{z_1, \ldots, z_m\} = Y' \setminus \{z_0\}$.

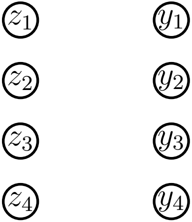
Path existence ⇒ augment Y (2)



- It suffices to show $Y' \in \mathcal{I}_1$ (direction $Y' \in \mathcal{I}_2$ is similar)
- Let $Z := (Y \setminus \{y_1, \ldots, y_m\}) \cup \{z_1, \ldots, z_m\} = Y' \setminus \{z_0\}$.
- Consider the undirected exchange graph $H = ((Y \setminus Z) \dot{\cup} (Z \setminus Y), E)$ w.r.t. M_1 and independent set Y (i.e. $(Y \setminus \{y\}) \cup \{z\} \in \mathcal{I}_1 \Rightarrow \{y, z\} \in E$)

Path existence ⇒ augment Y (3)

Claim. H contains a unique perfect matching.

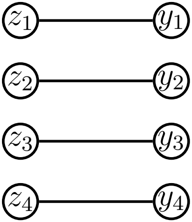


Path existence $\Rightarrow$ augment Y (3)

Claim. H contains a unique perfect matching.

Proof.

- Note that the edges $\{(z_i, y_i) : i = 1, \ldots, m\}$ from the directed path form a perfect matching on $Y \Delta Z$.

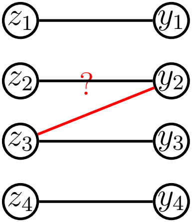


Path existence $\Rightarrow$ augment Y (3)

Claim. H contains a unique perfect matching.

Proof.

- Note that the edges $\{(z_i, y_i) : i = 1, \ldots, m\}$ from the directed path form a perfect matching on $Y \Delta Z$.
- H does not contain a **coord**, which is an edge (y_i, z_j) with $j > i$ (otherwise our X_1-X_2 was not shortest)

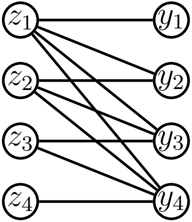


Path existence $\Rightarrow$ augment Y (3)

Claim. H contains a unique perfect matching.

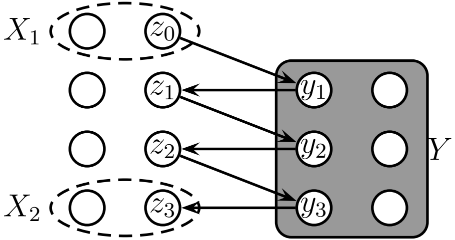
Proof.

- Note that the edges $\{(z_i, y_i) : i = 1, \ldots, m\}$ from the directed path form a perfect matching on $Y \Delta Z$.
- H does not contain a **coord**, which is an edge (y_i, z_j) with $j > i$ (otherwise our X_1-X_2 was not shortest)
- Now, consider the "complete" cordless graph $E^* := \{(y_i, z_j) : i \geq j\}$. Then this graph does have only one perfect matching. In particular, (y_1, z_1) has to be in a matching — then apply induction.



Path existence ⇒ augment Y (4)

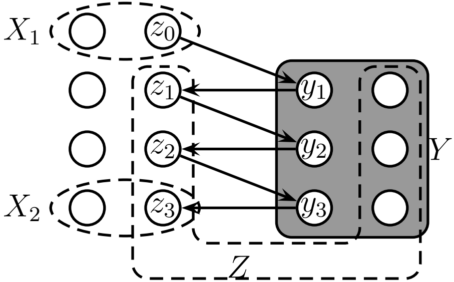
Main proof:



Path existence ⇒ augment Y (4)

Main proof:

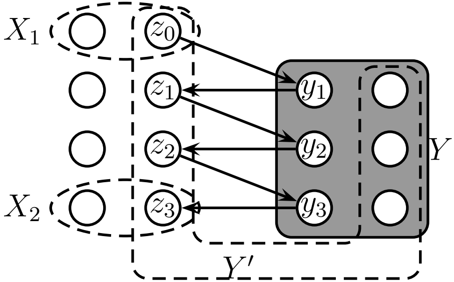
- As matching on $Y \Delta Z$ is unique, by **Reverse Exchange Lemma**, $Z = Y'/\{z_0\} \in \mathcal{I}_1$.



Path existence $\Rightarrow$ augment Y (4)

Main proof:

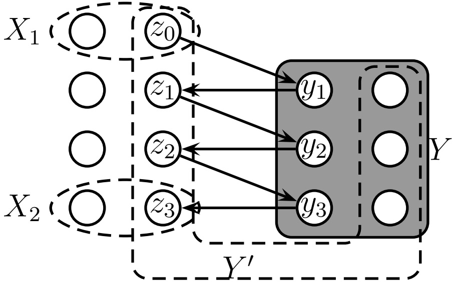
- As matching on $Y \Delta Z$ is unique, by **Reverse Exchange Lemma**, $Z = Y'/\{z_0\} \in \mathcal{I}_1$.
- We know that $r_{M_1}(Y \cup Y') \geq r_{M_1}(Y \cup \{z_0\}) \geq |Y| + 1$ since $z_0 \in X_1$ is one of the "M_1-augmenting" elements.



Path existence $\Rightarrow$ augment Y (4)

Main proof:

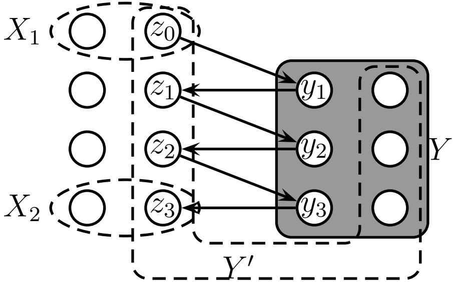
- As matching on $Y \Delta Z$ is unique, by **Reverse Exchange Lemma**, $Z = Y'/\{z_0\} \in \mathcal{I}_1$.
- We know that $r_{M_1}(Y \cup Y') \geq r_{M_1}(Y \cup \{z_0\}) \geq |Y| + 1$ since $z_0 \in X_1$ is one of the "M_1-augmenting" elements.
- One the other hand $r_{M_1}(Y \cup Y'/\{z_0\}) \leq |Y|$ ($Y' \cap X_1 = \{z_0\}$ by shortest path property)



Path existence $\Rightarrow$ augment Y (4)

Main proof:

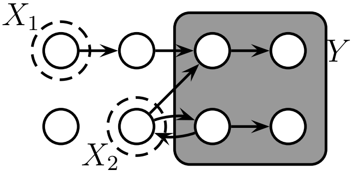
- As matching on $Y \Delta Z$ is unique, by **Reverse Exchange Lemma**, $Z = Y'/\{z_0\} \in \mathcal{I}_1$.
- We know that $r_{M_1}(Y \cup Y') \geq r_{M_1}(Y \cup \{z_0\}) \geq |Y| + 1$ since $z_0 \in X_1$ is one of the "M_1-augmenting" elements.
- One the other hand $r_{M_1}(Y \cup Y'/\{z_0\}) \leq |Y|$ ($Y' \cap X_1 = \{z_0\}$ by shortest path property)
- Hence, the only element that could possibly augment Y'/z_0 to an independent set of size $|Y| + 1$ is z_0 itself. $\square$



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (1)

Lemma

Suppose there is no path from a node in X_1 to a node in X_2 in D. Then Y is optimal. In particular we can find a subset $U \subseteq X$ so that $|Y| = r_{M_1}(U) + r_{M_2}(X \setminus U)$.



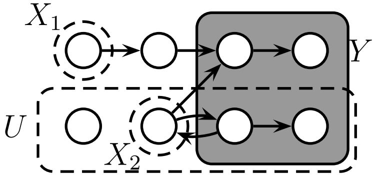
No $X_1 - X_2$ path $\Rightarrow Y$ optimal (1)

Lemma

Suppose there is no path from a node in X_1 to a node in X_2 in D. Then Y is optimal. In particular we can find a subset $U \subseteq X$ so that $|Y| = r_{M_1}(U) + r_{M_2}(X \setminus U)$.

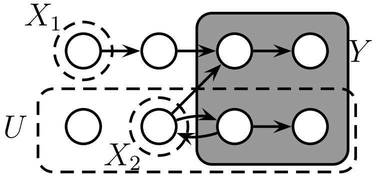
Proof.

- Let $U := \{i \in X : \nexists X_1 - i \text{ path in } H\}$ (or maybe more intuitively, $X \setminus U$ are the nodes that are reachable from X_1).



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (2)

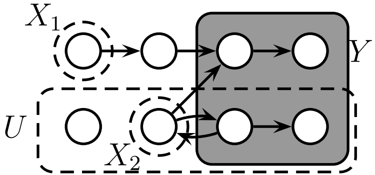
Claim I. $r_{M_1}(U) = |Y \cap U|$.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (2)

Claim I. $r_{M_1}(U) = |Y \cap U|$.

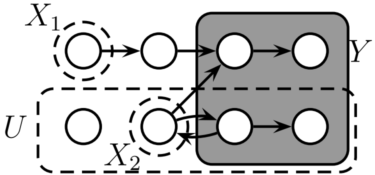
- One direction is easy: $r_{M_1}(U) \geq r_{M_1}(U \cap Y) = |U \cap Y|$.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (2)

Claim I. $r_{M_1}(U) = |Y \cap U|$.

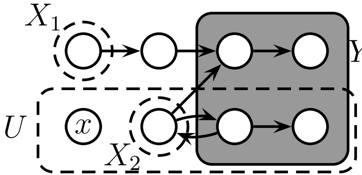
- One direction is easy: $r_{M_1}(U) \geq r_{M_1}(U \cap Y) = |U \cap Y|$.
- Suppose for the sake of contradiction that $r_{M_1}(U) > |Y \cap U|$.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (2)

Claim I. $r_{M_1}(U) = |Y \cap U|$.

- One direction is easy: $r_{M_1}(U) \geq r_{M_1}(U \cap Y) = |U \cap Y|$.
- Suppose for the sake of contradiction that $r_{M_1}(U) > |Y \cap U|$.
- So there is some $x \in U$ so that $(Y \cap U) \cup \{x\} \in \mathcal{I}_1$ is independent set of size $|Y \cap U| + 1$.



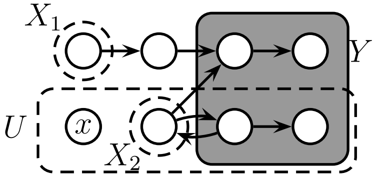
No $X_1 - X_2$ path $\Rightarrow Y$ optimal (2)

Claim I. $r_{M_1}(U) = |Y \cap U|$.

- One direction is easy: $r_{M_1}(U) \geq r_{M_1}(U \cap Y) = |U \cap Y|$.
- Suppose for the sake of contradiction that $r_{M_1}(U) > |Y \cap U|$.
- So there is some $x \in U$ so that $(Y \cap U) \cup \{x\} \in \mathcal{I}_1$ is independent set of size $|Y \cap U| + 1$.

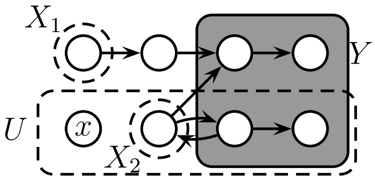
Case $r_{M_1}(Y \cup \{x\}) = |Y| + 1$.

- Then $x \in X_1 \cap U$, contradicting the choice of U.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (3)

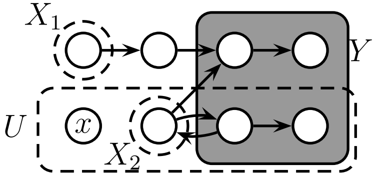
Case: $r_{M_1}(Y \cup \{x\}) = |Y|$.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (3)

Case: $r_{M_1}(Y \cup \{x\}) = |Y|$.

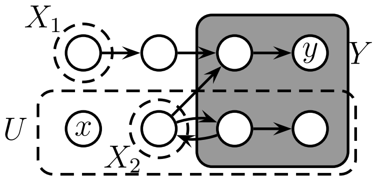
- Take a maximal independent set Z with $(Y \cap U) \cup \{x\} \subseteq Z \subseteq Y \cup \{x\}$.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (3)

Case: $r_{M_1}(Y \cup \{x\}) = |Y|$**.**

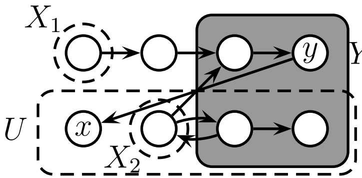
- Take a maximal independent set Z with $(Y \cap U) \cup \{x\} \subseteq Z \subseteq Y \cup \{x\}$.
- Then there is exactly one element $y \in Y/U$, so that $Z = (Y/y) \cup \{x\}$.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (3)

Case: $r_{M_1}(Y \cup \{x\}) = |Y|$.

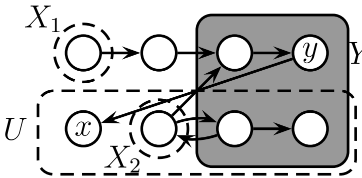
- Take a maximal independent set Z with $(Y \cap U) \cup \{x\} \subseteq Z \subseteq Y \cup \{x\}$.
- Then there is exactly one element $y \in Y/U$, so that $Z = (Y/y) \cup \{x\}$.
- This implies that we have a directed edge (y, x) in D.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (3)

Case: $r_{M_1}(Y \cup \{x\}) = |Y|$**.**

- Take a maximal independent set Z with $(Y \cap U) \cup \{x\} \subseteq Z \subseteq Y \cup \{x\}$.
- Then there is exactly one element $y \in Y/U$, so that $Z = (Y/y) \cup \{x\}$.
- This implies that we have a directed edge (y, x) in D.
- Then the node $x \in U$ is reachable from a element $y \notin U$, which contradicts the definition of U.
- From the contradiction we obtain that indeed $r_{M_1}(U) = |Y \cap U|$.



No $X_1 - X_2$ path $\Rightarrow Y$ optimal (4)

Claim I. $r_{M_1}(U) = |Y \cap U|$.

- Done!

No $X_1 - X_2$ path $\Rightarrow Y$ optimal (4)

Claim I. $r_{M_1}(U) = |Y \cap U|$.

- Done!

Claim II. $r_{M_2}(X/U) = |Y \cap (X/U)|$

- Similar – skipped for symmetry reasons.

No $X_1 - X_2$ path $\Rightarrow Y$ optimal (4)

Claim I. $r_{M_1}(U) = |Y \cap U|$.

- Done!

Claim II. $r_{M_2}(X/U) = |Y \cap (X/U)|$

- Similar – skipped for symmetry reasons.

Conclusion:

- Overall, we have found a set U so that
  $|Y| = |Y \cap U| + |Y \cap (X \setminus U)| = r_{M_1}(U) + r_{M_2}(X \setminus U)$.

Conclusion

Theorem

Matroid intersection can be solved in polynomial time.

Conclusion

Theorem

Matroid intersection can be solved in polynomial time.

Algorithm:

(1) Start with $Y := \emptyset$

(2) REPEAT

    (3) construct the directed exchange graph;

    (4) compute shortest X_1-X_2 path

    (5) IF there is none $\rightarrow Y$ optimal

    (6) ELSE augment Y

Conclusion

Theorem

Matroid intersection can be solved in polynomial time.

Algorithm:

(1) Start with $Y := \emptyset$

(2) REPEAT

  (3) construct the directed exchange graph;

  (4) compute shortest X_1-X_2 path

  (5) IF there is none $\to Y$ optimal

  (6) ELSE augment Y

Requirement for matroid:

- There needs to be an polynomial time algorithm for **independence oracle** — given $Y \subseteq X$, decide whether $Y \in \mathcal{I}$.

Conclusion (2)

Theorem (Edmond's matroid intersection theorem)

For any matroids $M_1 = (X, \mathcal{I}_1)$ and $M_2 = (X, \mathcal{I}_2)$ one has

$$\max\{|S| : S \in \mathcal{I}_1 \cap \mathcal{I}_2\} = \min_{U \subseteq X}\{r_{M_1}(U) + r_{M_2}(X \setminus U)\}$$

Proof.

- When the matroid intersection algorithm terminates, then it has found a set U providing equality.

Lecture 19

A Strongly Polynomial-time Algorithm for Min Cost Circulations — Part 1/2

Source: The book sequence "Combinatorial Optimization" by Schrijver, Part A, Chapter 12.

Min Cost Circulations

Min Cost Circulation
Input: a directed graph $D = (V, A)$, **edge cost** $c : A \to \mathbb{R}$, lower bounds $\ell : A \to \mathbb{R}$ and upper bounds $u : A \to \mathbb{R}$.
Output: a circulation $f : A \to \mathbb{R}$ with $\ell(a) \leq f(a) \leq u(a)$ for all $a \in A$ minimizing $c(f) := \sum_{a \in A} c(a) \cdot f(a)$.

Min Cost Circulations

Min Cost Circulation
Input: a directed graph $D = (V, A)$, **edge cost** $c : A \to \mathbb{R}$, lower bounds $\ell : A \to \mathbb{R}$ and upper bounds $u : A \to \mathbb{R}$.
Output: a circulation $f : A \to \mathbb{R}$ with $\ell(a) \leq f(a) \leq u(a)$ for all $a \in A$ minimizing $c(f) := \sum_{a \in A} c(a) \cdot f(a)$.

- We allow $f(a) \in \mathbb{R}$ to be negative
- Easy to model for example the minimum cost max s-t flow problem

The residual graph

Definition

For a circulation f we define the **residual graph** $D_f = (V, A_f)$ by

$$\ell(u) \leq f(a) \leq u(a),$$

graph D $\quad \bullet \xrightarrow{\color{red}{c(a)}} \bullet \qquad$ graph D_f $\quad \bullet \qquad\qquad \bullet$

The residual graph

Definition

For a circulation f we define the **residual graph** $D_f = (V, A_f)$ by

- For $a \in A$ with $f(a) < u(a)$ we have $a \in A_f$ with **residual capacity** $u_f(a) := u(a) - f(a)$ and residual cost $c(a)$.

$\ell(u) \leq f(a) \leq u(a),$
$c(a)$

graph D $\bullet \longrightarrow \bullet$

$u_f(a) = u(a) - f(a),$
$c_f(a) = c(a)$

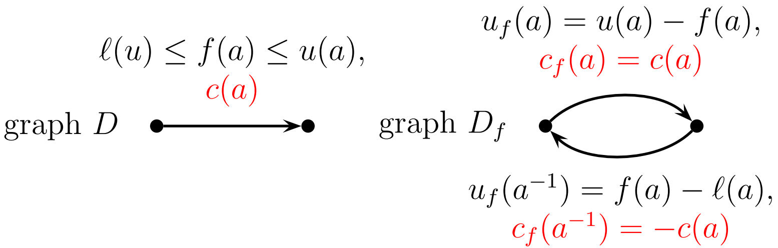
graph D_f $\bullet \longrightarrow \bullet$

The residual graph

Definition

For a circulation f we define the **residual graph** $D_f = (V, A_f)$ by

- For $a \in A$ with $f(a) < u(a)$ we have $a \in A_f$ with **residual capacity** $u_f(a) := u(a) - f(a)$ and residual cost $c(a)$.
- For $a \in A$ with $f(a) > \ell(a)$ we have $a^{-1} \in A_f$ with **residual capacity** $u_f(a^{-1}) := f(a) - \ell(a)$ and residual cost $c(a^{-1}) := -c(a)$.



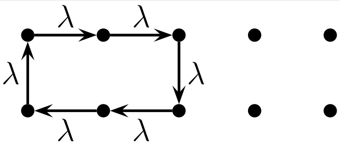
Modifying the circulation

Lemma

Let $f : A \to \mathbb{R}$ *be a circulation with* $\ell(a) \leq f(a) \leq u(a)$ *and let* $C \subseteq A_f$ *be a directed circuit in the residual graph. Set* $\lambda := \min\{u_f(a) : a \in C\}$. *Then* $f' : A \to \mathbb{R}$ *with*

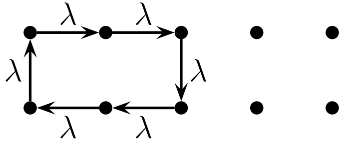
$$f'(a) := \begin{cases} f(a) + \lambda & \text{if } a \in C \\ f(a) - \lambda & \text{if } a^{-1} \in C \\ f(a) & \text{otherwise} \end{cases}$$

is a circulation with $\ell(a) \leq f'(a) \leq u(a)$ *and* $c(f') = c(f) + \lambda \cdot c(C)$.



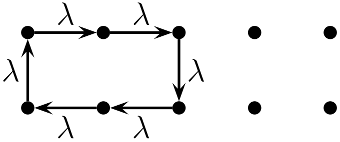
Decompositing a circulation

- Let us call a circulation **atomic** if there is a single directed circuit $C \subseteq A$ so that
$$f(a) = \begin{cases} \lambda & \text{if } a \in C \\ 0 & \text{otherwise} \end{cases}$$
for some $\lambda \geq 0$.



Decompositing a circulation

- Let us call a circulation **atomic** if there is a single directed circuit $C \subseteq A$ so that
$$f(a) = \begin{cases} \lambda & \text{if } a \in C \\ 0 & \text{otherwise} \end{cases}$$
for some $\lambda \geq 0$.



Lemma

Let $f : A \to \mathbb{R}_{\geq 0}$ be a circulation in $D = (V, A)$. Then there are atomic circulations $f_1, \ldots, f_k : A \to \mathbb{R}_{\geq 0}$ with $f = f_1 + \ldots + f_k$ and $k \leq |A|$.

Proof: Past exercise.

Optimality criterion

Lemma

Let f be a circulation in $D = (V, A)$ with $\ell(a) \leq f(a) \leq u(a)$ for all $a \in A$. Then f is not optimal $\Leftrightarrow$ there is a negative cost cycle in D_f.

Proof.

- Done in Chapter 4.

The Minimum Mean Cycle

- **Idea:** We want to use a “good” negative cost circuit in D_f for augmentation

The Minimum Mean Cycle

- **Idea:** We want to use a "good" negative cost circuit in D_f for augmentation

Definition

For a circulation $f : A \to \mathbb{R}$, we define

$$\mu(f) := \min_{C \text{ cycle in } D_f} \left\{ \frac{c(C)}{|C|} \right\}$$

as the cost of the **minimum mean cycle**.

- Note that we allow the empty cycle so that always $\mu(f) \leq 0$.

The Minimum Mean Cycle (2)

Lemma

Given a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$. A minimum mean cycle can be found in time $O(|V|^3 \cdot |A|)$.

The Minimum Mean Cycle (2)

Lemma

Given a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$. A minimum mean cycle can be found in time $O(|V|^3 \cdot |A|)$.

Proof.

- Use the **dynamic program**

$$d_0(u,v) := \begin{cases} 0 & \text{if } u = v \\ \infty & \text{if } u \neq v \end{cases} \quad \text{and} \quad d_k(u,v) := \min_{(w,v)\in A} \{d_{k-1}(u,w) + c(w,v)\}$$

  for $k = 1, \ldots, |V|$ and $u, v \in V$.
- Then $d_k(u,v) =$ min cost of u-v walk with exactly k arcs.

The Minimum Mean Cycle (2)

Lemma

Given a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$. A minimum mean cycle can be found in time $O(|V|^3 \cdot |A|)$.

Proof.

- Use the **dynamic program**

$$d_0(u,v) := \begin{cases} 0 & \text{if } u = v \\ \infty & \text{if } u \neq v \end{cases} \quad \text{and} \quad d_k(u,v) := \min_{(w,v) \in A} \{d_{k-1}(u,w) + c(w,v)\}$$

  for $k = 1, \ldots, |V|$ and $u, v \in V$.
- Then $d_k(u,v) =$ min cost of u-v walk with exactly k arcs.
- Computing all entries takes time $O(|V|^3 \cdot |A|)$.

The Minimum Mean Cycle (2)

Lemma

Given a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$. A minimum mean cycle can be found in time $O(|V|^3 \cdot |A|)$.

Proof.

- Use the **dynamic program**

$$d_0(u, v) := \begin{cases} 0 & \text{if } u = v \\ \infty & \text{if } u \neq v \end{cases} \quad \text{and} \quad d_k(u, v) := \min_{(w,v) \in A} \{d_{k-1}(u, w) + c(w, v)\}$$

  for $k = 1, \ldots, |V|$ and $u, v \in V$.
- Then $d_k(u, v) =$ min cost of u-v walk with exactly k arcs.
- Computing all entries takes time $O(|V|^3 \cdot |A|)$.
- Then the cost of the minimum mean cycle is

$$\min_{k=0,\ldots,|V|;\ u \in V} \left\{ \frac{d_k(u, u)}{k} \right\} \qquad \square$$

The Minimum Mean Cycle (2)

Lemma

Given a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$. A minimum mean cycle can be found in time $O(|V|^3 \cdot |A|)$.

Proof.

- Use the **dynamic program**

$$d_0(u,v) := \begin{cases} 0 & \text{if } u = v \\ \infty & \text{if } u \neq v \end{cases} \quad \text{and} \quad d_k(u,v) := \min_{(w,v)\in A} \{d_{k-1}(u,w) + c(w,v)\}$$

  for $k = 1, \ldots, |V|$ and $u, v \in V$.
- Then $d_k(u,v) =$ min cost of u-v walk with exactly k arcs.
- Computing all entries takes time $O(|V|^3 \cdot |A|)$.
- Then the cost of the minimum mean cycle is

$$\min_{k=0,\ldots,|V|;\ u\in V} \left\{ \frac{d_k(u,u)}{k} \right\} \qquad \square$$

Note: Possibly to improve running time to $O(|V| \cdot |A|)$.

The algorithm

Minimum mean cycle canceling algorithm

(1) Compute any feasible circulation f with $\ell(a) \leq f(a) \leq u(a)$ for all $a \in A$.

(2) WHILE $\exists$ cycle in D_f with negative cost DO

  (3) Compute minimum mean cycle $C \subseteq A_f$

  (4) Augment f along C by $\min\{u_f(a) : a \in C\}$

The algorithm

Minimum mean cycle canceling algorithm

(1) Compute any feasible circulation f with $\ell(a) \leq f(a) \leq u(a)$ for all $a \in A$.

(2) WHILE $\exists$ cycle in D_f with negative cost DO

  (3) Compute minimum mean cycle $C \subseteq A_f$

  (4) Augment f along C by $\min\{u_f(a) : a \in C\}$

Implement (1) as follows:

- Def. $D' = (V, A')$ which for each $a \in A$ with $\ell(a) > 0$, contains two parallel arcs a', a''

$$\begin{array}{lll} c(a') = -1, & \ell(a'') = 0, & u(a') = \ell(a) \\ c(a'') = 0, & \ell(a'') = 0, & u(a'') = u(a) - \ell(a) \end{array}$$

(and keep arcs with $\ell(a) \leq 0$).

The algorithm

Minimum mean cycle canceling algorithm

(1) Compute any feasible circulation f with $\ell(a) \leq f(a) \leq u(a)$ for all $a \in A$.

(2) WHILE $\exists$ cycle in D_f with negative cost DO

  (3) Compute minimum mean cycle $C \subseteq A_f$

  (4) Augment f along C by $\min\{u_f(a) : a \in C\}$

Implement (1) as follows:

- Def. $D' = (V, A')$ which for each $a \in A$ with $\ell(a) > 0$, contains two parallel arcs a', a''

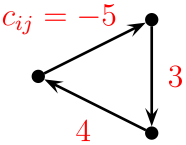
$$\begin{array}{lll} c(a') = -1, & \ell(a'') = 0, & u(a') = \ell(a) \\ c(a'') = 0, & \ell(a'') = 0, & u(a'') = u(a) - \ell(a) \end{array}$$

  (and keep arcs with $\ell(a) \leq 0$).
- $f = 0$ is a feasible circulation in D'
- Mincost circulation in D' gives feasible circulation in D.

Node potentials

Definition

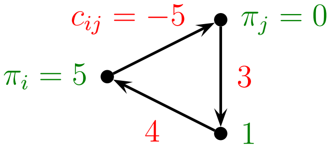
For a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$, a function $\pi : V \to \mathbb{R}$ are called **node potentials**. These induce **reduced costs** $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j$.



Node potentials

Definition

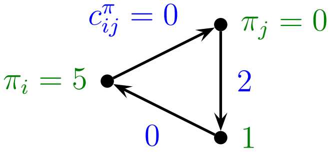
For a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$, a function $\pi : V \to \mathbb{R}$ are called **node potentials**. These induce **reduced costs** $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j$.



Node potentials

Definition

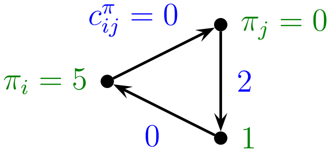
For a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$, a function $\pi : V \to \mathbb{R}$ are called **node potentials**. These induce **reduced costs** $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j$.



Node potentials

Definition

For a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$, a function $\pi : V \to \mathbb{R}$ are called **node potentials**. These induce **reduced costs** $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j$.



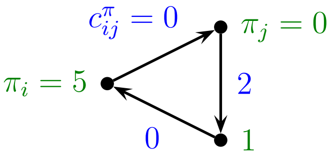
Lemma

Let $D = (V, A)$ be a directed graph with cost $c : A \to \mathbb{R}$ and node potentials π. Then any cycle $C \subseteq A$ has $c^{\pi}(C) = c(C)$.

Node potentials

Definition

For a directed graph $D = (V, A)$ with edge cost $c : A \to \mathbb{R}$, a function $\pi : V \to \mathbb{R}$ are called **node potentials**. These induce **reduced costs** $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j$.



Lemma

Let $D = (V, A)$ be a directed graph with cost $c : A \to \mathbb{R}$ and node potentials π. Then any cycle $C \subseteq A$ has $c^{\pi}(C) = c(C)$.

Proof.

- Clear as the node potentials cancel out.

Node potentials (2)

Lemma

Let $D = (V, A)$ be a directed graph with arc cost $c : A \to \mathbb{R}$. Then D has no negative cost cycle $\Leftrightarrow$ there are node potentials with $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j \geq 0 \quad \forall (i, j) \in A$.

Node potentials (2)

Lemma

Let $D = (V, A)$ be a directed graph with arc cost $c : A \to \mathbb{R}$. Then D has no negative cost cycle $\Leftrightarrow$ there are node potentials with $c^\pi_{i,j} := c_{ij} + \pi_i - \pi_j \geq 0 \quad \forall (i,j) \in A$.

Proof of "$\Leftarrow$":

- Any cycle C has $c(C) = c^\pi(C) \geq 0$ by prev. Lemma.

Node potentials (2)

Lemma

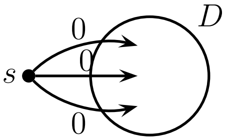
Let $D = (V, A)$ be a directed graph with arc cost $c : A \to \mathbb{R}$. Then D has no negative cost cycle $\Leftrightarrow$ there are node potentials with $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j \geq 0 \quad \forall (i,j) \in A$.

Proof of "$\Leftarrow$":

- Any cycle C has $c(C) = c^{\pi}(C) \geq 0$ by prev. Lemma.

Proof of "$\Rightarrow$":

- Add an new node s and (s, v) with $c(s, v) := 0 \; \forall v \in V$.



Node potentials (2)

Lemma

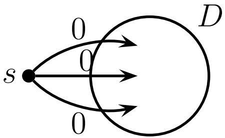
Let $D = (V, A)$ be a directed graph with arc cost $c : A \to \mathbb{R}$. Then D has no negative cost cycle $\Leftrightarrow$ there are node potentials with $c^{\pi}_{i,j} := c_{ij} + \pi_i - \pi_j \geq 0 \quad \forall (i, j) \in A$.

Proof of "$\Leftarrow$":

- Any cycle C has $c(C) = c^{\pi}(C) \geq 0$ by prev. Lemma.

Proof of "$\Rightarrow$":

- Add an new node s and (s, v) with $c(s, v) := 0 \; \forall v \in V$.



- Let $\pi(i) := d(s, i) := s - i$ distance w.r.t. c
- So $c^{\pi}_{i,j} = c_{ij} + d(s, i) - d(s, j) \geq 0 \Leftrightarrow d(s, j) \leq d(s, i) + c_{ij}$ which is the triangle inequality.

ε-optimality

Definition

A circulation $f : A \to \mathbb{R}$ is **ε-optimal** for $\varepsilon \geq 0$ if there are node potentials $\pi : V \to \mathbb{R}$ so that $c_{ij}^{\pi} \geq -\varepsilon$ for all $(i, j) \in A_f$.

ε-optimality

Definition

A circulation $f : A \to \mathbb{R}$ is **ε-optimal** for $\varepsilon \geq 0$ if there are node potentials $\pi : V \to \mathbb{R}$ so that $c_{ij}^{\pi} \geq -\varepsilon$ for all $(i, j) \in A_f$.

Definition

For a circulation $f : A \to \mathbb{R}$, let

$$\varepsilon(f) := \min\{\varepsilon \geq 0 : f \text{ is } \varepsilon\text{-optimal}\}$$

ε-optimality

Definition

A circulation $f : A \to \mathbb{R}$ is **ε-optimal** for $\varepsilon \geq 0$ if there are node potentials $\pi : V \to \mathbb{R}$ so that $c_{ij}^{\pi} \geq -\varepsilon$ for all $(i,j) \in A_f$.

Definition

For a circulation $f : A \to \mathbb{R}$, let

$$\varepsilon(f) := \min\{\varepsilon \geq 0 : f \text{ is } \varepsilon\text{-optimal}\}$$

Intuition:

- $\varepsilon(f)$ is the smallest amount that has to be added to the cost of the arcs in the residual graph to eliminate all negative cost cycles

ε-optimality (2)

Lemma

For any circulation f, $\mu(f) = -\varepsilon(f)$.

ε-optimality (2)

Lemma

For any circulation f, $\mu(f) = -\varepsilon(f)$.

Proof of "$\geq$" (:

- Let π be the node potentials valid for $\varepsilon(f)$. The minimum mean cycle C has $|C| \cdot \mu(f) = c(C) = c^\pi(C) \geq -\varepsilon(f) \cdot |C|$.

ε-optimality (2)

Lemma

For any circulation f, $\mu(f) = -\varepsilon(f)$.

Proof of "$\geq$" (:

- Let π be the node potentials valid for $\varepsilon(f)$. The minimum mean cycle C has $|C| \cdot \mu(f) = c(C) = c^{\pi}(C) \geq -\varepsilon(f) \cdot |C|$.

Proof of "$\leq$":

- Let use define a cost function $\tilde{c}(u,v) := c(u,v) - \mu(f)$.
- Now there is no negative cost cycle w.r.t. $\tilde{c}$ and there are node potentials π with $\tilde{c}(i,j) + \pi_i - \pi_j \geq 0$, which is the same as $c(i,j) + \pi_i - \pi_j \geq \mu(f)$.

Lecture 20

A Strongly Polynomial-time Algorithm for Min Cost Circulations — Part 2/2

Source: The book sequence “Combinatorial Optimization” by Schrijver, Part A, Chapter 12.

Monotonicty of $\varepsilon(f)$

Lemma

Update f to f' by augmenting along a minimum mean cost cycle. Then $\varepsilon(f') \leq \varepsilon(f)$.

- **Remark 1:** Equivalently this means that $|\mu(f')| \leq |\mu(f)|$
- **Remark 2:** False if arbitrary cycle is chosen.

Monotonicty of $\varepsilon(f)$

Lemma

Update f to f' by augmenting along a minimum mean cost cycle. Then $\varepsilon(f') \leq \varepsilon(f)$.

- **Remark 1:** Equivalently this means that $|\mu(f')| \leq |\mu(f)|$
- **Remark 2:** False if arbitrary cycle is chosen.

Proof.

- Set $\varepsilon := \varepsilon(f)$. Let $\pi : V \to \mathbb{R}$ be the node potentials with $c^\pi(a) \geq -\varepsilon$ for every arc $a \in A_f$.
- We will show that the **same** node potentials are still feasible for the updated graph $D_{f'}$.

Monotonicty of $\varepsilon(f)$

Lemma

Update f to f' by augmenting along a minimum mean cost cycle. Then $\varepsilon(f') \leq \varepsilon(f)$.

- **Remark 1:** Equivalently this means that $|\mu(f')| \leq |\mu(f)|$
- **Remark 2:** False if arbitrary cycle is chosen.

Proof.

- Set $\varepsilon := \varepsilon(f)$. Let $\pi : V \to \mathbb{R}$ be the node potentials with $c^\pi(a) \geq -\varepsilon$ for every arc $a \in A_f$.
- We will show that the **same** node potentials are still feasible for the updated graph $D_{f'}$.
- Let $C \subseteq A_f$ be the **minimum mean cycle**.
- Then $c^\pi(C) = -|C| \cdot \varepsilon$ and $c^\pi(a) = -\varepsilon$ for every $a \in C$.

Monotonicty of $\varepsilon(f)$

Lemma

Update f to f' by augmenting along a minimum mean cost cycle. Then $\varepsilon(f') \leq \varepsilon(f)$.

- **Remark 1:** Equivalently this means that $|\mu(f')| \leq |\mu(f)|$
- **Remark 2:** False if arbitrary cycle is chosen.

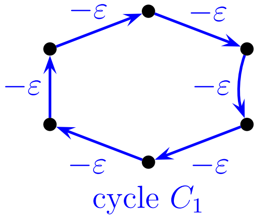
Proof.

- Set $\varepsilon := \varepsilon(f)$. Let $\pi : V \to \mathbb{R}$ be the node potentials with $c^\pi(a) \geq -\varepsilon$ for every arc $a \in A_f$.
- We will show that the **same** node potentials are still feasible for the updated graph $D_{f'}$.
- Let $C \subseteq A_f$ be the **minimum mean cycle**.
- Then $c^\pi(C) = -|C| \cdot \varepsilon$ and $c^\pi(a) = -\varepsilon$ for every $a \in C$.
- The only new arcs $(i,j) \in A_{f'} \setminus A_f$ have $(j,i) \in C$. So the reduced cost are $c^\pi_{ij} = -c_{ji} + \pi_i - \pi_j = -c^\pi_{ji} = \varepsilon \geq 0$. $\square$

Monotonicty of $\varepsilon(f)$

Consider the following scenario:

- Suppose C_1 is minimum mean cycle used in augmentation of f

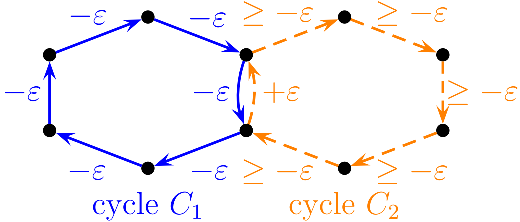


(edges are labelled with reduced cost $c^{\pi}(i, j)$)

Monotonicty of $\varepsilon(f)$

Consider the following scenario:

- Suppose C_1 is minimum mean cycle used in augmentation of f
- In next iteration we augment using a minimum mean cycle C_2 that contains a reverse arc of C_1.

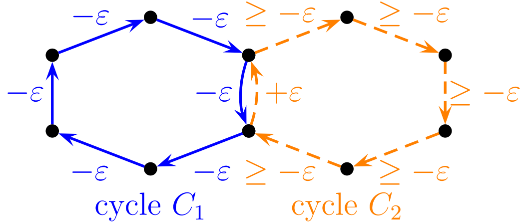


(edges are labelled with reduced cost $c^{\pi}(i,j)$)

Monotonicty of $\varepsilon(f)$

Consider the following scenario:

- Suppose C_1 is minimum mean cycle used in augmentation of f
- In next iteration we augment using a minimum mean cycle C_2 that contains a reverse arc of C_1.
- Then: mean cost of C_2 > mean cost of C_1



(edges are labelled with reduced cost $c^{\pi}(i,j)$)

$\varepsilon(f)$ is decreasing every $|A|$ iterations

Lemma

Consider a sequence $\{f_i\}_{i\geq 0}$ of circulations where f_{i+1} emerges from f_i by augmenting along the minimum mean cycle in D_{f_i}. Then $\varepsilon(f_{|A|+1}) \leq (1 - \frac{1}{|V|}) \cdot \varepsilon(f_0)$.

$\varepsilon(f)$ is decreasing every $|A|$ iterations

Lemma

Consider a sequence $\{f_i\}_{i\geq 0}$ of circulations where f_{i+1} emerges from f_i by augmenting along the minimum mean cycle in D_{f_i}. Then $\varepsilon(f_{|A|+1}) \leq (1 - \frac{1}{|V|}) \cdot \varepsilon(f_0)$.

Proof.

- Let $\varepsilon := \varepsilon(f_0)$ and fix potentials $\pi : V \to \mathbb{R}$ with $c^\pi(a) \geq -\varepsilon$ for all $a \in A_{f_0}$.

$\varepsilon(f)$ is decreasing every $|A|$ iterations (2)

Claim I. There is a $k \in \{0, \ldots, |A|+1\}$ so that the minimum mean cost cycle $C \subseteq D_{f_k}$ in that iteration contains an arc $a \in C$ with $c^{\pi}(a) \geq 0$.

$\varepsilon(f)$ is decreasing every $|A|$ iterations (2)

Claim I. There is a $k \in \{0, \ldots, |A|+1\}$ so that the minimum mean cost cycle $C \subseteq D_{f_k}$ in that iteration contains an arc $a \in C$ with $c^\pi(a) \geq 0$.

Proof of claim I.

- As long as the current iteration k uses a minimum mean cost cycle $C \subseteq D_{f_k}$ with $c^\pi(a) < 0$, every arc that appears new in the residual graph will non-negative reduced cost.

$\varepsilon(f)$ is decreasing every $|A|$ iterations (2)

Claim I. There is a $k \in \{0, \ldots, |A| + 1\}$ so that the minimum mean cost cycle $C \subseteq D_{f_k}$ in that iteration contains an arc $a \in C$ with $c^\pi(a) \geq 0$.

Proof of claim I.

- As long as the current iteration k uses a minimum mean cost cycle $C \subseteq D_{f_k}$ with $c^\pi(a) < 0$, every arc that appears new in the residual graph will non-negative reduced cost.
- Moreover, in every iteration at least one arc is **bottleneck arc** and will **not** appear in the residual graph of the next iteration. This can only go on for at most $|A|$ iterations.

$\varepsilon(f)$ is decreasing every $|A|$ iterations (3)

Claim II. Consider the minimal such k from Claim I. Then $\mu(f_k) \geq -(1 - \frac{1}{|V|}) \cdot \varepsilon$.

$\varepsilon(f)$ is decreasing every $|A|$ iterations (3)

Claim II. Consider the minimal such k from Claim I. Then $\mu(f_k) \geq -(1 - \frac{1}{|V|}) \cdot \varepsilon$.

Proof of claim II.

- Let $C \subseteq A_{f_k}$ be the minimum mean cycle.

$\varepsilon(f)$ is decreasing every $|A|$ iterations (3)

Claim II. Consider the minimal such k from Claim I. Then $\mu(f_k) \geq -(1 - \frac{1}{|V|}) \cdot \varepsilon$.

Proof of claim II.

- Let $C \subseteq A_{f_k}$ be the minimum mean cycle.
- We know that $c^\pi(a) \geq -\varepsilon \; \forall a \in A_{f_k}$.
  WARNING: Might not be true anymore <u>after</u> iteration k!
- Moreover C contains at least one arc with $c^\pi(a) \geq 0$.

$\varepsilon(f)$ is decreasing every $|A|$ iterations (3)

Claim II. Consider the minimal such k from Claim I. Then $\mu(f_k) \geq -(1 - \frac{1}{|V|}) \cdot \varepsilon$.

Proof of claim II.

- Let $C \subseteq A_{f_k}$ be the minimum mean cycle.
- We know that $c^\pi(a) \geq -\varepsilon \; \forall a \in A_{f_k}$.
  WARNING: Might not be true anymore <u>after</u> iteration k!
- Moreover C contains at least one arc with $c^\pi(a) \geq 0$.
- Then $c(C) = c^\pi(C) \geq (|C| - 1) \cdot (-\varepsilon)$ and hence $\mu(f_k) = \frac{c(C)}{|C|} \geq (1 - \frac{1}{|C|}) \cdot (-\varepsilon)$.
- The claim follows from $|C| \leq |V|$. $\square$

A 1st bound for Min Mean Cycle algo

Theorem

Suppose the cost function is $c : A \to \{-c_{\max}, \ldots, +c_{\max}\}$. Then the minimum mean cycle cancelling algorithm terminates after $|V| \cdot (|A| + 1) \cdot \ln(2|V| \cdot c_{\max}))$ iterations.

A 1st bound for Min Mean Cycle algo

Theorem

Suppose the cost function is $c : A \to \{-c_{\max}, \ldots, +c_{\max}\}$. Then the minimum mean cycle cancelling algorithm terminates after $|V| \cdot (|A| + 1) \cdot \ln(2|V| \cdot c_{\max}))$ iterations.

Proof.

- Let f be the flow before the 1st iteration and let f' be the circulation after $|V| \cdot (|A| + 1) \cdot \ln(2|V| \cdot c_{\max})$ iterations.

A 1st bound for Min Mean Cycle algo

Theorem

Suppose the cost function is $c : A \to \{-c_{\max}, \dots, +c_{\max}\}$. Then the minimum mean cycle cancelling algorithm terminates after $|V| \cdot (|A| + 1) \cdot \ln(2|V| \cdot c_{\max}))$ iterations.

Proof.

- Let f be the flow before the 1st iteration and let f' be the circulation after $|V| \cdot (|A| + 1) \cdot \ln(2|V| \cdot c_{\max})$ iterations.
- Then f is $c_{\max}$-optimal.

A 1st bound for Min Mean Cycle algo

Theorem

Suppose the cost function is $c : A \to \{-c_{\max}, \dots, +c_{\max}\}$. Then the minimum mean cycle cancelling algorithm terminates after $|V| \cdot (|A| + 1) \cdot \ln(2|V| \cdot c_{\max}))$ iterations.

Proof.

- Let f be the flow before the 1st iteration and let f' be the circulation after $|V| \cdot (|A| + 1) \cdot \ln(2|V| \cdot c_{\max})$ iterations.
- Then f is $c_{\max}$-optimal.
- Moreover,

$$\begin{aligned} \varepsilon(f') &\leq c_{\max} \cdot \left(1 - \frac{1}{|V|}\right)^{|V| \cdot \ln(2|V| \cdot c_{\max})} \\ &\leq c_{\max} \cdot \exp(-\ln(2|V| c_{\max})) = \frac{1}{2|V|}. \end{aligned}$$

- By integral, this implies that actually $\varepsilon(f') = 0$.

The analysis of Tardos

Theorem (Tardos 1985)

The minimum mean cost cycle algorithm terminates after $O(m^2 \cdot n \cdot \ln(n))$ *iterations where* $n := |V|, m := |A|$.

The analysis of Tardos

Theorem (Tardos 1985)

The minimum mean cost cycle algorithm terminates after $O(m^2 \cdot n \cdot \ln(n))$ iterations where $n := |V|, m := |A|$.

- Let $f_0, f_1, \ldots$ be the **circulations** appearing in minimum mean cycle algorithm
- Let C_t be the **minimum mean cycle** in D_{f_t} used to augment
- Let $\tau := 2nm\lceil\ln(n)\rceil$

The analysis of Tardos

Theorem (Tardos 1985)

The minimum mean cost cycle algorithm terminates after $O(m^2 \cdot n \cdot \ln(n))$ iterations where $n := |V|, m := |A|$.

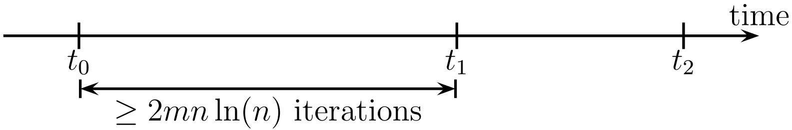
- Let $f_0, f_1, \ldots$ be the **circulations** appearing in minimum mean cycle algorithm
- Let C_t be the **minimum mean cycle** in D_{f_t} used to augment
- Let $\tau := 2nm\lceil \ln(n) \rceil$

Claim. For each t_0, there exists at least one arc $a \in C_{t_0}$ so that $f_{t_1}(a_0) = f_{t_2}(a_0)$ for all $t_2 \geq t_1 := t_0 + \tau$.

- If proven, then we terminate after at most $2m\tau = O(m^2 n \ln(n))$ iterations.

The analysis of Tardos (2)

Claim. For each t_0, there exists at least one arc $a \in C_{t_0}$ so that $f_{t_1}(a_0) = f_{t_2}(a_0)$ for all $t_2 \geq t_1 := t_0 + \tau$.

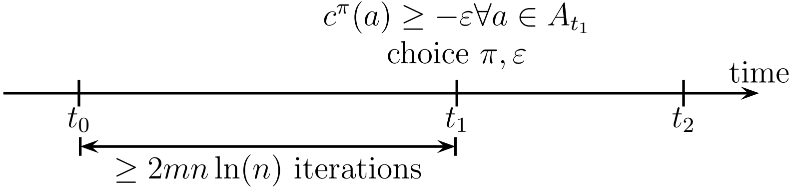


The analysis of Tardos (2)

Claim. For each t_0, there exists at least one arc $a \in C_{t_0}$ so that $f_{t_1}(a_0) = f_{t_2}(a_0)$ for all $t_2 \geq t_1 := t_0 + \tau$.

Proof.

- Set $\varepsilon := \varepsilon(t_1)$
- Let π be **potential** for iteration t_1, i.e. $c^\pi(a) := c(a) + \pi(u) - \pi(v)$ and $c^\pi(a) \geq -\varepsilon$ for every $a = (u, v) \in A_{t_1}$

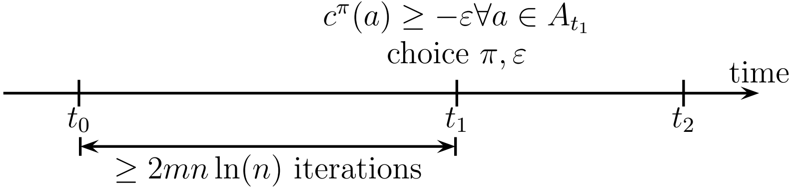


The analysis of Tardos (2)

Claim. For each t_0, there exists at least one arc $a \in C_{t_0}$ so that $f_{t_1}(a_0) = f_{t_2}(a_0)$ for all $t_2 \geq t_1 := t_0 + \tau$.

Proof.

- Set $\varepsilon := \varepsilon(t_1)$
- Let π be **potential** for iteration t_1, i.e. $c^\pi(a) := c(a) + \pi(u) - \pi(v)$ and $c^\pi(a) \geq -\varepsilon$ for every $a = (u, v) \in A_{t_1}$
- Then $\varepsilon \leq \varepsilon(t_0) \cdot (1 - \frac{1}{n})^{\tau/m} \leq \varepsilon(t_0) \exp(-2\lceil \ln(n) \rceil) \leq \frac{\varepsilon(t_0)}{2n}$

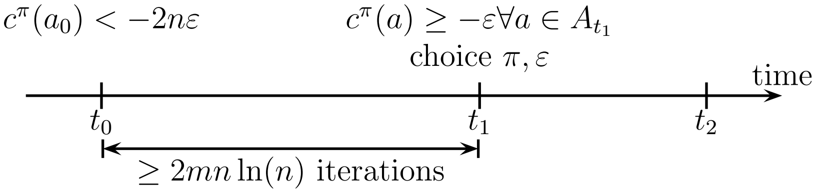


The analysis of Tardos (2)

Claim. For each t_0, there exists at least one arc $a \in C_{t_0}$ so that $f_{t_1}(a_0) = f_{t_2}(a_0)$ for all $t_2 \geq t_1 := t_0 + \tau$.

Proof.

- Set $\varepsilon := \varepsilon(t_1)$
- Let π be **potential** for iteration t_1, i.e. $c^\pi(a) := c(a) + \pi(u) - \pi(v)$ and $c^\pi(a) \geq -\varepsilon$ for every $a = (u, v) \in A_{t_1}$
- Then $\varepsilon \leq \varepsilon(t_0) \cdot (1 - \frac{1}{n})^{\tau/m} \leq \varepsilon(t_0) \exp(-2\lceil \ln(n) \rceil) \leq \frac{\varepsilon(t_0)}{2n}$
- Fix any arc $a_0 \in C_{t_0}$ with $c^\pi(a_0) \leq -\varepsilon(f_{t_0}) < -2n\varepsilon$

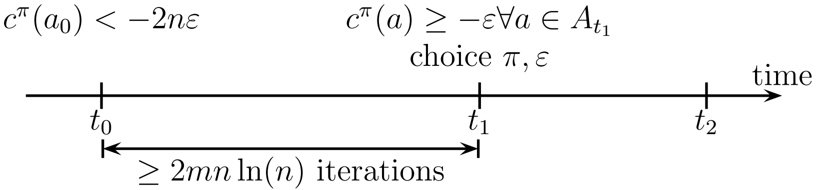


The analysis of Tardos (2)

Claim. For each t_0, there exists at least one arc $a \in C_{t_0}$ so that $f_{t_1}(a_0) = f_{t_2}(a_0)$ for all $t_2 \geq t_1 := t_0 + \tau$.

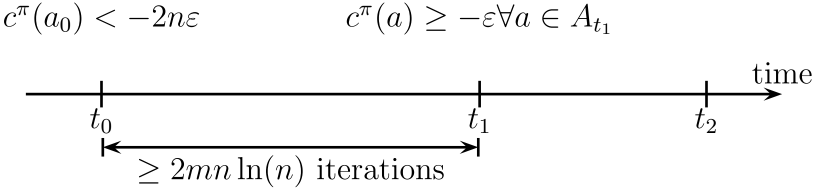
Proof.

- Set $\varepsilon := \varepsilon(t_1)$
- Let π be **potential** for iteration t_1, i.e. $c^\pi(a) := c(a) + \pi(u) - \pi(v)$ and $c^\pi(a) \geq -\varepsilon$ for every $a = (u, v) \in A_{t_1}$
- Then $\varepsilon \leq \varepsilon(t_0) \cdot (1 - \frac{1}{n})^{\tau/m} \leq \varepsilon(t_0) \exp(-2\lceil \ln(n) \rceil) \leq \frac{\varepsilon(t_0)}{2n}$
- Fix any arc $a_0 \in C_{t_0}$ with $c^\pi(a_0) \leq -\varepsilon(f_{t_0}) < -2n\varepsilon$
- Assume by symmetry that $a_0 \in A$ **(=forward arc)**



The analysis of Tardos (2)

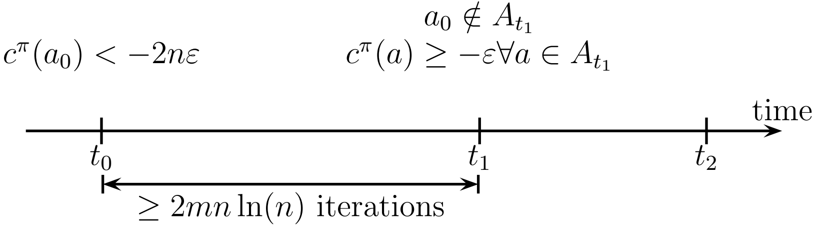
- Assume for sake of contradition that $f_{t_2}(a_0) \neq f_{t_1}(a_0)$.



The analysis of Tardos (2)

- Assume for sake of contradition that $f_{t_2}(a_0) \neq f_{t_1}(a_0)$.

Subclaim. One has $a_0 \notin A_{t_1}$

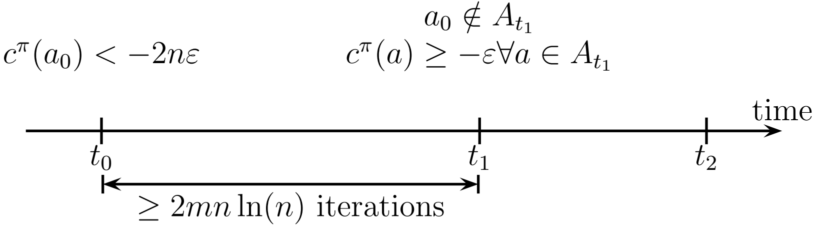


The analysis of Tardos (2)

- Assume for sake of contradition that $f_{t_2}(a_0) \neq f_{t_1}(a_0)$.

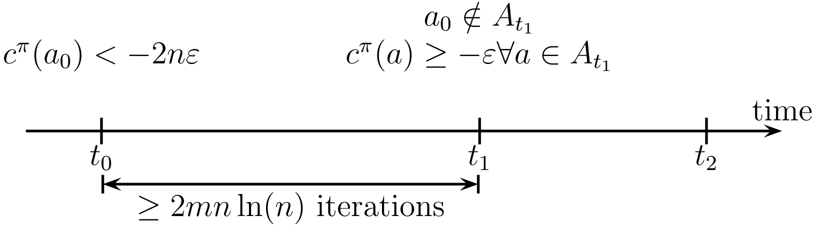
Subclaim. One has $a_0 \notin A_{t_1}$

- If $a_0 \in A_{t_1}$, then $c^\pi(a_0) \geq -\varepsilon$ while at the same time $c^\pi(a_0) < -2n\varepsilon$. Contradiction!



The analysis of Tardos (3)

Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

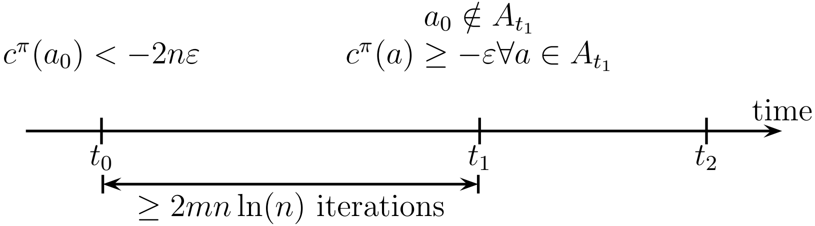


The analysis of Tardos (3)

Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Proof of Subclaim II.

- By previous claim $u(a_0) = f_{t_1}(a_0) > f_{t_2}(a_0)$

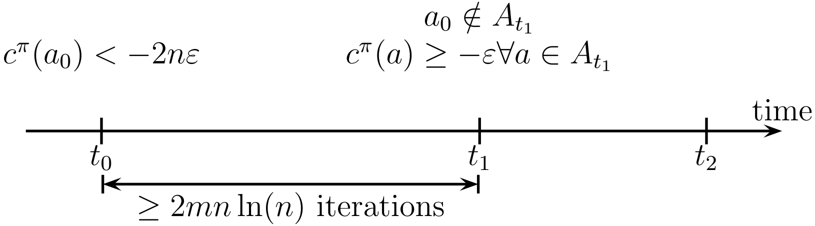


The analysis of Tardos (3)

Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Proof of Subclaim II.

- By previous claim $u(a_0) = f_{t_1}(a_0) > f_{t_2}(a_0)$
- Then the difference $h := f_{t_1} - f_{t_2}$ is a circulation in D_{t_2} with $h(a_0) > 0$

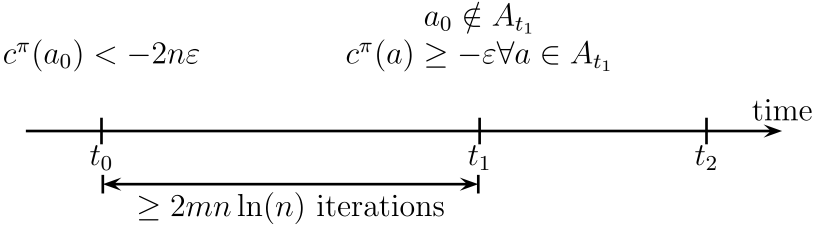


The analysis of Tardos (3)

Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Proof of Subclaim II.

- By previous claim $u(a_0) = f_{t_1}(a_0) > f_{t_2}(a_0)$
- Then the difference $h := f_{t_1} - f_{t_2}$ is a circulation in D_{t_2} with $h(a_0) > 0$
- Choose a circuit $C \subseteq \{a \mid h(a) > 0\}$ containing a_0



The analysis of Tardos (3)

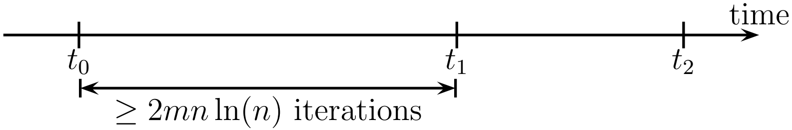
Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Proof of Subclaim II.

- By previous claim $u(a_0) = f_{t_1}(a_0) > f_{t_2}(a_0)$
- Then the difference $h := f_{t_1} - f_{t_2}$ is a circulation in D_{t_2} with $h(a_0) > 0$
- Choose a circuit $C \subseteq \{a \mid h(a) > 0\}$ containing a_0
- Then $C \subseteq A_{t_2}$ and $C^{-1} \subseteq A_{t_1}$

$a_0 \notin A_{t_1}$

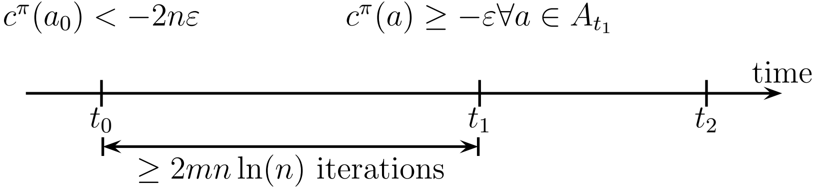
$c^\pi(a_0) < -2n\varepsilon$ $\qquad$ $c^\pi(a) \geq -\varepsilon \forall a \in A_{t_1}$



The analysis of Tardos (3)

Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Continuation of main proof.

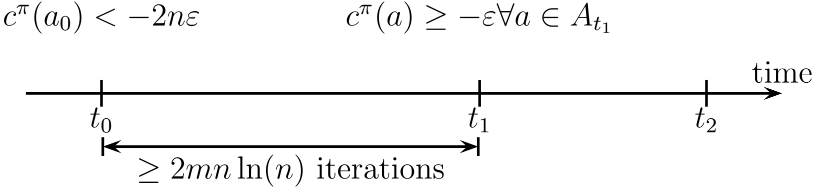


The analysis of Tardos (3)

Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Continuation of main proof.

- Since $C^{-1} \subseteq A_{t_1}$, $c^\pi(a^{-1}) \geq -\varepsilon \Rightarrow c^\pi(a) \leq \varepsilon \ \forall a \in C$



The analysis of Tardos (3)

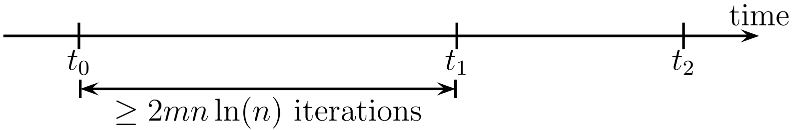
Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Continuation of main proof.

- Since $C^{-1} \subseteq A_{t_1}$, $c^\pi(a^{-1}) \geq -\varepsilon \Rightarrow c^\pi(a) \leq \varepsilon \ \forall a \in C$
- Then

$$c(C) = c^\pi(C) = \underbrace{c^\pi(a_0)}_{<-2n\varepsilon} + \underbrace{c^\pi(C \setminus \{a_0\})}_{\leq(|C|-1)\cdot\varepsilon} < -n\varepsilon \leq -|C| \cdot \varepsilon(t_2)$$

$c^\pi(a_0) < -2n\varepsilon$ $\qquad\qquad$ $c^\pi(a) \geq -\varepsilon \forall a \in A_{t_1}$



The analysis of Tardos (3)

Subclaim II. There exists a directed circuit C so that (i) $C \subseteq A_{t_2}$, (ii) $C^{-1} \subseteq A_{t_1}$, (iii) $a_0 \in C$.

Continuation of main proof.

- Since $C^{-1} \subseteq A_{t_1}$, $c^\pi(a^{-1}) \geq -\varepsilon \Rightarrow c^\pi(a) \leq \varepsilon \ \forall a \in C$
- Then

$$c(C) = c^\pi(C) = \underbrace{c^\pi(a_0)}_{<-2n\varepsilon} + \underbrace{c^\pi(C \setminus \{a_0\})}_{\leq (|C|-1)\cdot\varepsilon} < -n\varepsilon \leq -|C| \cdot \varepsilon(t_2)$$

- So there is a circuit C in D_{t_2} with mean cost $< -\varepsilon(t_2)$.
- Contradiction! $\square$

$c^\pi(a_0) < -2n\varepsilon$ $\qquad$ $c^\pi(a) \geq -\varepsilon \forall a \in A_{t_1}$