

Diameter of Polytopes: Algorithmic and Combinatorial Aspects

Laura Sanità

Department of Mathematics and Computer Science
TU Eindhoven (Netherlands)

Department of Combinatorics and Optimization
University of Waterloo (Canada)

IPCO Summer School, 2020

Linear Programming

Linear Programming

- **Linear Programming** is concerned with the problem of
 - ▶ minimize/maximize a **linear** function on d **continuous** variables
 - ▶ subject to a finite set of **linear constraints**
- Example:

$$\begin{array}{rcll} \max & 5x_1 & -3x_2 & \\ & 2x_1 & +3x_2 & \leq 2 \\ & -x_1 & +4x_2 & \leq 3 \\ & & -3x_2 & \leq 0 \end{array}$$

Linear Programming

- **Linear Programming** is concerned with the problem of
 - ▶ minimize/maximize a **linear** function on d **continuous** variables
 - ▶ subject to a finite set of **linear constraints**
- Example:

$$\begin{array}{rcll} \max & 5x_1 & -3x_2 & \\ & 2x_1 & +3x_2 & \leq 2 \\ & -x_1 & +4x_2 & \leq 3 \\ & & -3x_2 & \leq 0 \end{array}$$

$$\begin{array}{rcl} \max & c^T x & \\ & Ax & \leq b \end{array}$$

- ▶ $x \in \mathbb{R}^d$ is the vector of **variables**
- ▶ $c \in \mathbb{R}^d, b \in \mathbb{R}^n, A \in \mathbb{R}^{n \times d}$ are given

- The above problem instances are called **Linear Programs (LP)**.

Is Linear Programming useful?

Is Linear Programming useful?

- LPs can be used to model several **optimization** problems:
 - ▶ shortest path in a graph
 - ▶ network flows
 - ▶ assignment
 - ▶ ...

Is Linear Programming useful?

- LPs can be used to model several **optimization** problems:
 - ▶ shortest path in a graph
 - ▶ network flows
 - ▶ assignment
 - ▶ ...
- LPs are a fundamental tool for solving harder problems.

Is Linear Programming useful?

- LPs can be used to model several **optimization** problems:
 - ▶ shortest path in a graph
 - ▶ network flows
 - ▶ assignment
 - ▶ ...
- LPs are a fundamental tool for solving harder problems. For example:
 - ▶ Optimization problems with **integer variables** (via Branch&Bound, Cutting planes,...)
 - ▶ **Approximation algorithms** for NP-hard problems.
 - ▶ Commercial **solvers** (CPLEX, GUROBI, XPRESS, ...), Operations Research Industry, **Data Science**.

Algorithms for solving LPs?

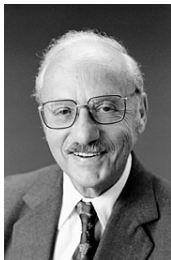
Algorithms for solving LPs?

- The development of algorithms for solving LPs started in the 40's. Some pioneers: Kantorovich&Koopmans, Dantzig, Von Neumann, Ford&Fulkerson...

Algorithms for solving LPs?

- The development of algorithms for solving LPs started in the 40's. Some pioneers: Kantorovich&Koopmans, Dantzig, Von Neumann, Ford&Fulkerson...

- ▶ **George Dantzig**: published the **Simplex Algorithm** for solving LPs in 1947

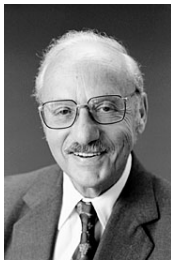


- Nowadays, the simplex algorithm is **extremely** popular and used in practice,

Algorithms for solving LPs?

- The development of algorithms for solving LPs started in the 40's. Some pioneers: Kantorovich&Koopmans, Dantzig, Von Neumann, Ford&Fulkerson...

- ▶ **George Dantzig:** published the **Simplex Algorithm** for solving LPs in 1947



- Nowadays, the simplex algorithm is **extremely** popular and used in practice, named as one of the **"top 10 algorithms"** of the 20th century.

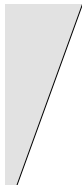
The Simplex Algorithm

The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure:

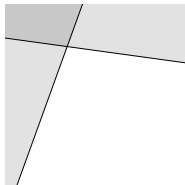
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure:



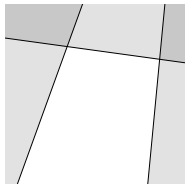
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure:



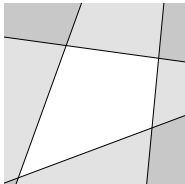
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure:



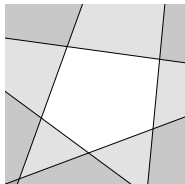
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure:



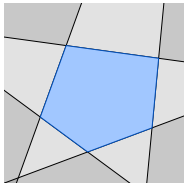
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure:



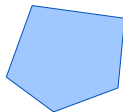
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure:



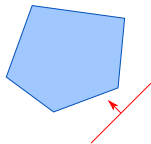
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



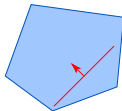
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



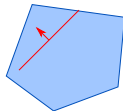
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



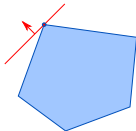
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



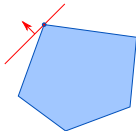
The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



The Simplex Algorithm

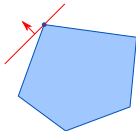
- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



- It is not difficult to realize that an **optimal** solution of such an LP can be found at one of the **extreme points** of the feasible region.

The Simplex Algorithm

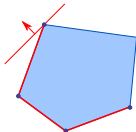
- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



- It is not difficult to realize that an **optimal** solution of such an LP can be found at one of the **extreme points** of the feasible region.
- **Simplex Algorithm's idea:** move from an extreme point to an improving **adjacent** one, until the optimum is found!

The Simplex Algorithm

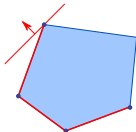
- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



- It is not difficult to realize that an **optimal** solution of such an LP can be found at one of the **extreme points** of the feasible region.
- **Simplex Algorithm's idea:** move from an extreme point to an improving **adjacent** one, until the optimum is found!

The Simplex Algorithm

- The set of possible solutions of an LP has a very nice structure: it is a **convex** set called a **polyhedron** (or a **polytope**, if bounded)



- It is not difficult to realize that an **optimal** solution of such an LP can be found at one of the **extreme points** of the feasible region.
- **Simplex Algorithm's idea:** move from an extreme point to an improving **adjacent** one, until the optimum is found!
- The operation of moving from one extreme point to the next is called **pivoting**

Pivoting

Pivoting

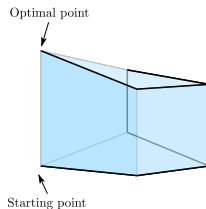
- Clearly, the path followed by the algorithm depends on the **pivoting rule**:
How do we choose the next (improving) extreme point?

Pivoting

- Clearly, the path followed by the algorithm depends on the **pivoting rule**:
How do we choose the next (improving) extreme point?
- **Dantzig's pivoting rule**: move along the edge that “seems” more promising in term of cost-function improvement

Pivoting

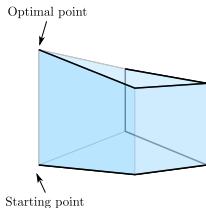
- Clearly, the path followed by the algorithm depends on the **pivoting rule**:
How do we choose the next (improving) extreme point?
- **Dantzig's pivoting rule**: move along the edge that “seems” more promising in term of cost-function improvement
- ▶ [Klee & Minty'72] showed that pivoting according to that rule requires an **exponential** in d number of steps.



Pivoting

- Clearly, the path followed by the algorithm depends on the **pivoting rule**:
How do we choose the next (improving) extreme point?
- **Dantzig's pivoting rule**: move along the edge that “seems” more promising in term of cost-function improvement

- ▶ [Klee & Minty'72] showed that pivoting according to that rule requires an **exponential** in d number of steps.



- Other pivoting rules?

Pivoting

- **Many** pivoting rules have been proposed in the literature in the past decades
 - ▶ Dantzig's rule
 - ▶ Greatest improvement
 - ▶ Bland's rule
 - ▶ Steepest-edge
 - ▶ Random pivot rules
 - ▶ Cunningham's pivot rule
 - ▶ Zadeh's pivot rule
 - ▶ ...

Pivoting

- **Many** pivoting rules have been proposed in the literature in the past decades
 - ▶ Dantzig's rule
 - ▶ Greatest improvement
 - ▶ Bland's rule
 - ▶ Steepest-edge
 - ▶ Random pivot rules
 - ▶ Cunningham's pivot rule
 - ▶ Zadeh's pivot rule
 - ▶ ...

...exhibiting a worst-case (sub)exponential behaviour for the Simplex algorithm
[Klee&Minty'72, Jeroslow'73, Avis&Chvátal'78, Goldfarb&Sit'79,
Friedmann&Hansen&Zwick'11, Friedmann'11, Avis&Friedmann'17,
Disser&Hopp'19]

Pivoting

- **Many** pivoting rules have been proposed in the literature in the past decades
 - ▶ Dantzig's rule
 - ▶ Greatest improvement
 - ▶ Bland's rule
 - ▶ Steepest-edge
 - ▶ Random pivot rules
 - ▶ Cunningham's pivot rule
 - ▶ Zadeh's pivot rule
 - ▶ ...

...exhibiting a worst-case (sub)exponential behaviour for the Simplex algorithm [Klee&Minty'72, Jeroslow'73, Avis&Chvátal'78, Goldfarb&Sit'79, Friedmann&Hansen&Zwick'11, Friedmann'11, Avis&Friedmann'17, Disser&Hopp'19]

- The Simplex algorithm (with e.g. Dantzig's rule) can 'implicitly' solve hard problems [Adler,Papadimitriou&Rubinstein'14, Skutella&Disser'15, Fearnley&Savani'15]

Is there a polynomial pivoting rule?

Is there a polynomial pivoting rule?

- After more than 70 years of use/studies, we still do **not** know the answer!

Is there a polynomial pivoting rule?

- After more than 70 years of use/studies, we still do **not** know the answer!
- LPs can be solved in polynomial-time using other algorithms (Ellipsoid algorithm [Khachiyan'79], Interior-point methods [Karmarkar'84])

Is there a polynomial pivoting rule?

- After more than 70 years of use/studies, we still do **not** know the answer!
- LPs can be solved in polynomial-time using other algorithms (Ellipsoid algorithm [Khachiyan'79], Interior-point methods [Karmarkar'84])
 - ▶ However, such algorithms run in **weakly** polynomial-time ($\text{poly}(d, n, \log L)$ where $L :=$ largest coefficient)

Is there a polynomial pivoting rule?

- After more than 70 years of use/studies, we still do **not** know the answer!
- LPs can be solved in polynomial-time using other algorithms (Ellipsoid algorithm [Khachiyan'79], Interior-point methods [Karmarkar'84])
 - ▶ However, such algorithms run in **weakly** polynomial-time ($\text{poly}(d, n, \log L)$ where $L :=$ largest coefficient)
- The existence of a polynomial pivoting rule could imply a **strongly** polynomial-time algorithm for solving LPs
(# elementary operations polynomial in the number of different input values)

Is there a polynomial pivoting rule?

- After more than 70 years of use/studies, we still do **not** know the answer!
- LPs can be solved in polynomial-time using other algorithms (Ellipsoid algorithm [Khachiyan'79], Interior-point methods [Karmarkar'84])
 - ▶ However, such algorithms run in **weakly** polynomial-time ($\text{poly}(d, n, \log L)$ where $L :=$ largest coefficient)
- The existence of a polynomial pivoting rule could imply a **strongly** polynomial-time algorithm for solving LPs (# elementary operations polynomial in the number of different input values)
 - ▶ Mentioned as one of the mathematical problems for next century by Fields Medalist **S. Smale** in 2000



Is there a polynomial pivoting rule?

- After more than 70 years of use/studies, we still do **not** know the answer!
- LPs can be solved in polynomial-time using other algorithms (Ellipsoid algorithm [Khachiyan'79], Interior-point methods [Karmarkar'84])
 - ▶ However, such algorithms run in **weakly** polynomial-time ($\text{poly}(d, n, \log L)$ where $L :=$ largest coefficient)
- The existence of a polynomial pivoting rule could imply a **strongly** polynomial-time algorithm for solving LPs (# elementary operations polynomial in the number of different input values)
 - ▶ Mentioned as one of the mathematical problems for next century by Fields Medalist **S. Smale** in 2000

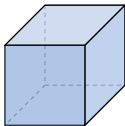


Related Question: *What is the maximum length of a 'shortest path' between two extreme points of a polytope?*

Diameter of polytopes

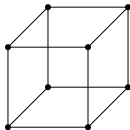
Diameter of polytopes

- We can naturally associate an undirected graph to a given polytope $P \subseteq \mathbb{R}^d$:
 - ▶ the **vertices** correspond to the extreme points of P
 - ▶ the **edges** are given by the 1-dimensional faces of P



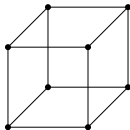
Diameter of polytopes

- We can naturally associate an undirected graph to a given polytope $P \subseteq \mathbb{R}^d$:
 - ▶ the **vertices** correspond to the extreme points of P
 - ▶ the **edges** are given by the 1-dimensional faces of P



Diameter of polytopes

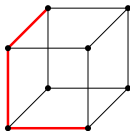
- We can naturally associate an undirected graph to a given polytope $P \subseteq \mathbb{R}^d$:
 - ▶ the **vertices** correspond to the extreme points of P
 - ▶ the **edges** are given by the 1-dimensional faces of P



- The **diameter** of P is the maximum value of a shortest path between a pair of vertices on this graph

Diameter of polytopes

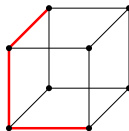
- We can naturally associate an undirected graph to a given polytope $P \subseteq \mathbb{R}^d$:
 - ▶ the **vertices** correspond to the extreme points of P
 - ▶ the **edges** are given by the 1-dimensional faces of P



- The **diameter** of P is the maximum value of a shortest path between a pair of vertices on this graph

Diameter of polytopes

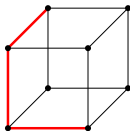
- We can naturally associate an undirected graph to a given polytope $P \subseteq \mathbb{R}^d$:
 - ▶ the **vertices** correspond to the extreme points of P
 - ▶ the **edges** are given by the 1-dimensional faces of P



- The **diameter** of P is the maximum value of a shortest path between a pair of vertices on this graph (**1-skeleton** of P).

Diameter of polytopes

- We can naturally associate an undirected graph to a given polytope $P \subseteq \mathbb{R}^d$:
 - ▶ the **vertices** correspond to the extreme points of P
 - ▶ the **edges** are given by the 1-dimensional faces of P



- The **diameter** of P is the maximum value of a shortest path between a pair of vertices on this graph (**1-skeleton** of P).

Remark: *In order for a polynomial pivoting rule to exist, a **necessary** condition is a polynomial bound on the value of the diameter!*

Diameter of polytopes

- A famous conjecture, proposed by [Hirsch'57], was the **Hirsch conjecture**, stating that the diameter of a d -dimensional polytope with n facets is $\leq n - d$.

Diameter of polytopes

- A famous conjecture, proposed by [Hirsch'57], was the **Hirsch conjecture**, stating that the diameter of a d -dimensional polytope with n facets is $\leq n - d$.
 - ▶ Disproved first for unbounded polyhedra [Klee&Walkup'67]
 - ▶ ..and later for bounded ones [Santos'12] (awarded **Fulkerson Prize** in 2015)

Diameter of polytopes

- A famous conjecture, proposed by [Hirsch'57], was the **Hirsch conjecture**, stating that the diameter of a d -dimensional polytope with n facets is $\leq n - d$.
 - ▶ Disproved first for unbounded polyhedra [Klee&Walkup'67]
 - ▶ ..and later for bounded ones [Santos'12] (awarded **Fulkerson Prize** in 2015)
 - ▶ holds e.g. for 0/1-polytopes [Naddef'89]

Diameter of polytopes

- A famous conjecture, proposed by [Hirsch'57], was the **Hirsch conjecture**, stating that the diameter of a d -dimensional polytope with n facets is $\leq n - d$.
 - ▶ Disproved first for unbounded polyhedra [Klee&Walkup'67]
 - ▶ ..and later for bounded ones [Santos'12] (awarded **Fulkerson Prize** in 2015)
 - ▶ holds e.g. for 0/1-polytopes [Naddef'89]
- Importantly, its **polynomial-version** is still open:

*Is the diameter of a d -dimensional polytope with n facets bounded by $f(d, n)$, for some **polynomial** function $f(d, n)$?*

Diameter of polytopes

- A famous conjecture, proposed by [Hirsch'57], was the **Hirsch conjecture**, stating that the diameter of a d -dimensional polytope with n facets is $\leq n - d$.

- ▶ Disproved first for unbounded polyhedra [Klee&Walkup'67]
- ▶ ..and later for bounded ones [Santos'12] (awarded **Fulkerson Prize** in 2015)
- ▶ holds e.g. for 0/1-polytopes [Naddef'89]

- Importantly, its **polynomial-version** is still open:

*Is the diameter of a d -dimensional polytope with n facets bounded by $f(d, n)$, for some **polynomial** function $f(d, n)$?*

- Best bound: $\sim (n - d)^{\log O(d / \log d)}$ [Sukegawa'18]
(strengthening [Kalai&Kleitman'92, Todd'14, Sukegawa&Kitahara'15])

Diameter of polytopes

- A famous conjecture, proposed by [Hirsch'57], was the **Hirsch conjecture**, stating that the diameter of a d -dimensional polytope with n facets is $\leq n - d$.

- ▶ Disproved first for unbounded polyhedra [Klee&Walkup'67]
- ▶ ..and later for bounded ones [Santos'12] (awarded **Fulkerson Prize** in 2015)
- ▶ holds e.g. for 0/1-polytopes [Naddef'89]

- Importantly, its **polynomial-version** is still open:

*Is the diameter of a d -dimensional polytope with n facets bounded by $f(d, n)$, for some **polynomial** function $f(d, n)$?*

- Best bound: $\sim (n - d)^{\log O(d / \log d)}$ [Sukegawa'18]
(strengthening [Kalai&Kleitman'92, Todd'14, Sukegawa&Kitahara'15])
- The diameter of a polytope has been studied from **many** different perspectives...

Diameter of polytopes

Diameter of polytopes

- Many researchers studied the diameter of polytopes describing feasible solutions of **combinatorial optimization** problems (and their relaxations).

Diameter of polytopes

- Many researchers studied the diameter of polytopes describing feasible solutions of **combinatorial optimization** problems (and their relaxations).

→ Just to mention a few: Matching, TSP, Flow and Transportation, Edge-cover, Stable marriage, Stable set, Partition, *and many more...*

Diameter of polytopes

- Many researchers studied the diameter of polytopes describing feasible solutions of **combinatorial optimization** problems (and their relaxations).

→ Just to mention a few: Matching, TSP, Flow and Transportation, Edge-cover, Stable marriage, Stable set, Partition, *and many more...*

- The diameter of a polytope has been investigated also from a **computational complexity** point of view.

Diameter of polytopes

- Many researchers studied the diameter of polytopes describing feasible solutions of **combinatorial optimization** problems (and their relaxations).

→ Just to mention a few: Matching, TSP, Flow and Transportation, Edge-cover, Stable marriage, Stable set, Partition, *and many more*. . .

- The diameter of a polytope has been investigated also from a **computational complexity** point of view.
 - ▶ [Frieze&Teng'94]: Computing the diameter of a polytope is **weakly NP-hard**.
 - ▶ [S.'18]: Computing the diameter of a polytope is **strongly NP-hard**. Computing a pair of vertices at maximum distance is **APX-hard**.

Diameter of polytopes

- Many researchers studied the diameter of polytopes describing feasible solutions of **combinatorial optimization** problems (and their relaxations).

→ Just to mention a few: Matching, TSP, Flow and Transportation, Edge-cover, Stable marriage, Stable set, Partition, *and many more*...

- The diameter of a polytope has been investigated also from a **computational complexity** point of view.

- ▶ [Frieze&Teng'94]: Computing the diameter of a polytope is **weakly NP-hard**.
- ▶ [S.'18]: Computing the diameter of a polytope is **strongly NP-hard**. Computing a pair of vertices at maximum distance is **APX-hard**.

→ The latter result holds for half-integral polytopes with a very easy description (**fractional matching polytope**).

In this lecture

In this lecture

- Characterization of the diameter of two polytopes (well-known in the combinatorial optimization community):
 - ▶ the matching polytope
 - ▶ the fractional matching polytope

In this lecture

- Characterization of the diameter of two polytopes (well-known in the combinatorial optimization community):
 - ▶ the matching polytope
 - ▶ the fractional matching polytope
- Discuss general algorithmic and hardness implications

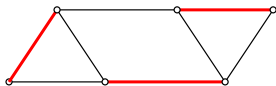
In this lecture

- Characterization of the diameter of two polytopes (well-known in the combinatorial optimization community):
 - ▶ the matching polytope
 - ▶ the fractional matching polytope
- Discuss general algorithmic and hardness implications
- Highlight open questions

The matching polytope

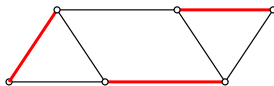
The matching polytope

- For a graph $G = (V, E)$, a **matching** is a subset of edges that have no node in common.

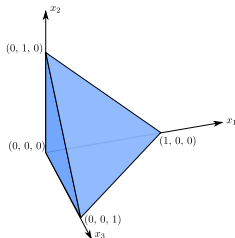
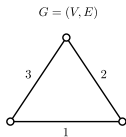


The matching polytope

- For a graph $G = (V, E)$, a **matching** is a subset of edges that have no node in common.



- The **matching polytope** ($\mathcal{P}_M$) is given by the **convex hull** of characteristic vectors of matchings of G .



The matching polytope

- [Edmonds'65] gave an LP-description of $\mathcal{P}_M$:

$$\mathcal{P}_M := \{x \in \mathbb{R}^E : \begin{array}{ll} \sum_{e \in \delta(v)} x_e \leq 1 & \forall v \in V, \\ \sum_{e \in E[S]} x_e \leq \frac{|S|-1}{2} & \forall S \subseteq V : |S| \text{ odd} \\ x \geq 0 \end{array}\}$$

The matching polytope

- [Edmonds'65] gave an LP-description of $\mathcal{P}_M$:

$$\mathcal{P}_M := \{x \in \mathbb{R}^E : \begin{array}{ll} \sum_{e \in \delta(v)} x_e \leq 1 & \forall v \in V, \\ \sum_{e \in E[S]} x_e \leq \frac{|S|-1}{2} & \forall S \subseteq V : |S| \text{ odd} \\ x \geq 0 \end{array}\}$$

How do we characterize adjacency of extreme points?

The matching polytope

- [Edmonds'65] gave an LP-description of $\mathcal{P}_M$:

$$\mathcal{P}_M := \{x \in \mathbb{R}^E : \begin{array}{ll} \sum_{e \in \delta(v)} x_e \leq 1 & \forall v \in V, \\ \sum_{e \in E[S]} x_e \leq \frac{|S|-1}{2} & \forall S \subseteq V : |S| \text{ odd} \\ x \geq 0 \end{array}\}$$

How do we characterize adjacency of extreme points?

Note: For a polyhedron $\mathcal{P} := \{x \in \mathbb{R}^d : Ax \leq b\}$ the following are equivalent:

- ▶ $z, y \in \mathcal{P}$ are adjacent extreme points on $\mathcal{P}$;
- ▶ There exists an **cost vector** $c \in \mathbb{R}^d$ such that z, y are the **only** optimal extreme points of $\max\{c^T x : x \in \mathcal{P}\}$;

The matching polytope

- [Edmonds'65] gave an LP-description of $\mathcal{P}_M$:

$$\mathcal{P}_M := \{x \in \mathbb{R}^E : \begin{array}{ll} \sum_{e \in \delta(v)} x_e \leq 1 & \forall v \in V, \\ \sum_{e \in E[S]} x_e \leq \frac{|S|-1}{2} & \forall S \subseteq V : |S| \text{ odd} \\ x \geq 0 \end{array}\}$$

How do we characterize adjacency of extreme points?

Note: For a polyhedron $\mathcal{P} := \{x \in \mathbb{R}^d : Ax \leq b\}$ the following are equivalent:

- ▶ $z, y \in \mathcal{P}$ are adjacent extreme points on $\mathcal{P}$;
- ▶ There exists an **cost vector** $c \in \mathbb{R}^d$ such that z, y are the **only** optimal extreme points of $\max\{c^T x : x \in \mathcal{P}\}$;
- ▶ The matrix corresp. to the constraints **tight** for both y, z has rank $d - 1$.

The matching polytope

- [Edmonds'65] gave an LP-description of $\mathcal{P}_M$:

$$\mathcal{P}_M := \{x \in \mathbb{R}^E : \begin{array}{ll} \sum_{e \in \delta(v)} x_e \leq 1 & \forall v \in V, \\ \sum_{e \in E[S]} x_e \leq \frac{|S|-1}{2} & \forall S \subseteq V : |S| \text{ odd} \\ x \geq 0 \end{array}\}$$

How do we characterize adjacency of extreme points?

Note: For a polyhedron $\mathcal{P} := \{x \in \mathbb{R}^d : Ax \leq b\}$ the following are equivalent:

- ▶ $z, y \in \mathcal{P}$ are adjacent extreme points on $\mathcal{P}$;
 - ▶ There exists an **cost vector** $c \in \mathbb{R}^d$ such that z, y are the **only** optimal extreme points of $\max\{c^T x : x \in \mathcal{P}\}$;
 - ▶ The matrix corresp. to the constraints **tight** for both y, z has rank $d - 1$.
- Matching is a graph problem. Any **graphical** characterization of adjacency?

The matching polytope

Theorem [Balinski&Russakoff'74, Chvátal'75]

*Two vertices of $\mathcal{P}_M$ are adjacent iff the **symmetric difference** of the corresponding matchings induces **one** component.*

The matching polytope

Theorem [Balinski&Russakoff'74, Chvátal'75]

Two vertices of $\mathcal{P}_M$ are adjacent iff the *symmetric difference* of the corresponding matchings induces **one** component.

Proof: The symmetric difference of two matchings is a union of alternating cycles and alternating paths.



(a)



(b)

The matching polytope

Theorem [Balinski&Russakoff'74,Chvátal'75]

Two vertices of $\mathcal{P}_M$ are adjacent iff the *symmetric difference* of the corresponding matchings induces **one** component.

Proof: The symmetric difference of two matchings is a union of alternating cycles and alternating paths.



(a)



(b)

- *Sufficiency:* There is an objective function for which these matchings are the *only* optimal extreme point solutions.

The matching polytope

Theorem [Balinski&Russakoff'74, Chvátal'75]

Two vertices of $\mathcal{P}_M$ are adjacent iff the *symmetric difference* of the corresponding matchings induces **one** component.

Proof: The symmetric difference of two matchings is a union of alternating cycles and alternating paths.



(a)



(b)

- *Sufficiency:* There is an objective function for which these matchings are the *only* optimal extreme point solutions.
- *Necessity:* If not, such an objective function can't exist!



The matching polytope

Corollary

*The diameter of $\mathcal{P}_M$ is equal to the size of a **maximum matching** of G .*

The matching polytope

Corollary

*The diameter of $\mathcal{P}_M$ is equal to the size of a **maximum matching** of G .*

Proof:

- ▶ Let M be a maximum matching. The distance between **any** two matchings is at most $|M|$.

The matching polytope

Corollary

*The diameter of $\mathcal{P}_M$ is equal to the size of a **maximum matching** of G .*

Proof:

- ▶ Let M be a maximum matching. The distance between **any** two matchings is at most $|M|$.
- ▶ The distance between the **empty** matching (extreme point **0**) and the matching M is $|M|$.



The matching polytope

Corollary

*The diameter of $\mathcal{P}_M$ is equal to the size of a **maximum matching** of G .*

Proof:

- ▶ Let M be a maximum matching. The distance between **any** two matchings is at most $|M|$.
- ▶ The distance between the **empty** matching (extreme point **0**) and the matching M is $|M|$.



- **Obs 1:** From [Edmonds'65] it follows that the diameter of the matching polytope can be computed in **polynomial time**.

The matching polytope

Corollary

*The diameter of $\mathcal{P}_M$ is equal to the size of a **maximum matching** of G .*

Proof:

- ▶ Let M be a maximum matching. The distance between **any** two matchings is at most $|M|$.
- ▶ The distance between the **empty** matching (extreme point $\mathbf{0}$) and the matching M is $|M|$.



- **Obs 1:** From [Edmonds'65] it follows that the diameter of the matching polytope can be computed in **polynomial time**.
- **Obs 2:** We can restate as:

$$\text{diameter}(\mathcal{P}_M) = \max_{x \in \text{vertices}(\mathcal{P}_M)} \{\mathbf{1}^T x\}$$

The fractional matching polytope

The fractional matching polytope

- The **fractional** matching polytope is given by a standard *LP-relaxation*:

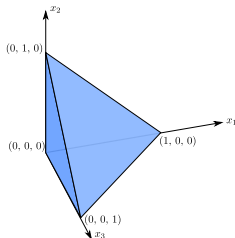
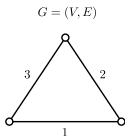
$$\mathcal{P}_{FM} := \{x \in \mathbb{R}^E : \sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in V, \quad x \geq 0\}$$

The fractional matching polytope

- The **fractional** matching polytope is given by a standard *LP-relaxation*:

$$\mathcal{P}_{FM} := \{x \in \mathbb{R}^E : \sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in V, \quad x \geq 0\}$$

- This polytope can have **fractional** extreme points

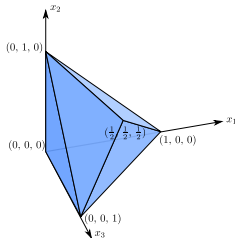
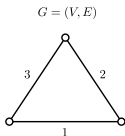


The fractional matching polytope

- The **fractional** matching polytope is given by a standard *LP-relaxation*:

$$\mathcal{P}_{FM} := \{x \in \mathbb{R}^E : \sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in V, \quad x \geq 0\}$$

- This polytope can have **fractional** extreme points



The fractional matching polytope

- [Balinski'65]: $\mathcal{P}_{FM}$ is a half-integral polytope.

The fractional matching polytope

- [Balinski'65]: $\mathcal{P}_{FM}$ is a **half-integral** polytope. For a vertex x of $\mathcal{P}_{FM}$
 - ▶ the edges $\{e \in E : x_e = 1\} \rightarrow$ induce a **matching** ($\mathcal{M}_x$)
 - ▶ the edges $\{e \in E : x_e = \frac{1}{2}\} \rightarrow$ induce a collection of **odd cycles** ($\mathcal{C}_x$)



The fractional matching polytope

- [Balinski'65]: $\mathcal{P}_{FM}$ is a **half-integral** polytope. For a vertex x of $\mathcal{P}_{FM}$
 - ▶ the edges $\{e \in E : x_e = 1\} \rightarrow$ induce a **matching** ($\mathcal{M}_x$)
 - ▶ the edges $\{e \in E : x_e = \frac{1}{2}\} \rightarrow$ induce a collection of **odd cycles** ($\mathcal{C}_x$)



- Adjacency relations have also been studied (see e.g. Behrend'13)

The fractional matching polytope

- [Balinski'65]: $\mathcal{P}_{FM}$ is a **half-integral** polytope. For a vertex x of $\mathcal{P}_{FM}$
 - ▶ the edges $\{e \in E : x_e = 1\} \rightarrow$ induce a **matching** ($\mathcal{M}_x$)
 - ▶ the edges $\{e \in E : x_e = \frac{1}{2}\} \rightarrow$ induce a collection of **odd cycles** ($\mathcal{C}_x$)



- Adjacency relations have also been studied (see e.g. Behrend'13)
 - $\rightarrow$ Let's derive some **graphical** properties of adjacent extreme points!

Adjacency

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in V$$

$$x_e \geq 0 \quad \forall e \in E$$

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\begin{aligned}\sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E\end{aligned}$$

$\bar{\mathcal{P}}_{FM}$:

$$\begin{aligned}\sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V\end{aligned}$$

- Add **slack** variables $\rightarrow$

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in V$$

$$x_e \geq 0 \quad \forall e \in E$$

$\bar{\mathcal{P}}_{FM}$:

$$\sum_{e \in \delta(v)} x_e + x_v = 1 \quad \forall v \in V$$

$$x_f \geq 0 \quad \forall f \in E \cup V$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E \end{aligned}$$

$\bar{\mathcal{P}}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V \end{aligned}$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!
- Let $\bar{y}, \bar{z}$ be adjacent vertices of $\bar{\mathcal{P}}_{FM}$, and $\bar{\mathcal{C}}_{\bar{x}}, \bar{\mathcal{C}}_{\bar{y}}$ the set of their **odd cycles**.

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E \end{aligned}$$

$\bar{\mathcal{P}}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V \end{aligned}$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!
- Let $\bar{y}, \bar{z}$ be adjacent vertices of $\bar{\mathcal{P}}_{FM}$, and $\bar{\mathcal{C}}_{\bar{x}}, \bar{\mathcal{C}}_{\bar{y}}$ the set of their **odd cycles**.
- Claim:** $\bar{\mathcal{C}}_{\bar{x}} \Delta \bar{\mathcal{C}}_{\bar{y}} \leq 2$.

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM} :$

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E \end{aligned}$$

$\bar{\mathcal{P}}_{FM} :$

$$\begin{aligned} \sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V \end{aligned}$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!
- Let $\bar{y}, \bar{z}$ be adjacent vertices of $\bar{\mathcal{P}}_{FM}$, and $\bar{\mathcal{C}}_{\bar{x}}, \bar{\mathcal{C}}_{\bar{y}}$ the set of their **odd cycles**.
- Claim:** $\bar{\mathcal{C}}_{\bar{x}} \Delta \bar{\mathcal{C}}_{\bar{y}} \leq 2$. Why?

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E \end{aligned}$$

$\bar{\mathcal{P}}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V \end{aligned}$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!
- Let $\bar{y}, \bar{z}$ be adjacent vertices of $\bar{\mathcal{P}}_{FM}$, and $\bar{\mathcal{C}}_{\bar{x}}, \bar{\mathcal{C}}_{\bar{y}}$ the set of their **odd cycles**.
- Claim:** $\bar{\mathcal{C}}_{\bar{x}} \Delta \bar{\mathcal{C}}_{\bar{y}} \leq 2$. Why?
 - $\text{support}(\bar{y}) \cup \text{support}(\bar{z})$ contains **one** component $\bar{K}$ with some f : $\bar{y}_f \neq \bar{z}_f$.

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E \end{aligned}$$

$\bar{\mathcal{P}}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V \end{aligned}$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!
- Let $\bar{y}, \bar{z}$ be adjacent vertices of $\bar{\mathcal{P}}_{FM}$, and $\bar{\mathcal{C}}_{\bar{x}}, \bar{\mathcal{C}}_{\bar{y}}$ the set of their **odd cycles**.
- Claim:** $\bar{\mathcal{C}}_{\bar{x}} \Delta \bar{\mathcal{C}}_{\bar{y}} \leq 2$. Why?
 - support($\bar{y}$) $\cup$ support($\bar{z}$) contains **one** component $\bar{K}$ with some f : $\bar{y}_f \neq \bar{z}_f$.
 - $\bar{y}, \bar{z}$ adjacent $\Leftrightarrow$ there are $|E \cup V| - 1$ linearly independent constraints of $\bar{\mathcal{P}}_{FM}$ that are **tight** for both $\bar{y}$ and $\bar{z}$

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E \end{aligned}$$

$\bar{\mathcal{P}}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V \end{aligned}$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!
- Let $\bar{y}, \bar{z}$ be adjacent vertices of $\bar{\mathcal{P}}_{FM}$, and $\bar{\mathcal{C}}_{\bar{x}}, \bar{\mathcal{C}}_{\bar{y}}$ the set of their **odd cycles**.
- Claim:** $\bar{\mathcal{C}}_{\bar{x}} \Delta \bar{\mathcal{C}}_{\bar{y}} \leq 2$. Why?
 - support($\bar{y}$) $\cup$ support($\bar{z}$) contains **one** component $\bar{K}$ with some f : $\bar{y}_f \neq \bar{z}_f$.
 - $\bar{y}, \bar{z}$ adjacent $\Leftrightarrow$ there are $|E \cup V| - 1$ linearly independent constraints of $\bar{\mathcal{P}}_{FM}$ that are **tight** for both $\bar{y}$ and $\bar{z} \Rightarrow \bar{K}$ has at most $\leq |\bar{K}| + 1$ edges.

Adjacency

- Consider again the LP-description.

$\mathcal{P}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &\leq 1 & \forall v \in V \\ x_e &\geq 0 & \forall e \in E \end{aligned}$$

$\bar{\mathcal{P}}_{FM}$:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e + x_v &= 1 & \forall v \in V \\ x_f &\geq 0 & \forall f \in E \cup V \end{aligned}$$

- Add **slack** variables $\rightarrow$ corresponds to adding **one loop edge** on each node!
- Let $\bar{y}, \bar{z}$ be adjacent vertices of $\bar{\mathcal{P}}_{FM}$, and $\bar{\mathcal{C}}_{\bar{x}}, \bar{\mathcal{C}}_{\bar{y}}$ the set of their **odd cycles**.
- Claim:** $\bar{\mathcal{C}}_{\bar{x}} \Delta \bar{\mathcal{C}}_{\bar{y}} \leq 2$. Why?
 - support($\bar{y}$) $\cup$ support($\bar{z}$) contains **one** component $\bar{K}$ with some f : $\bar{y}_f \neq \bar{z}_f$.
 - $\bar{y}, \bar{z}$ adjacent $\Leftrightarrow$ there are $|E \cup V| - 1$ linearly independent constraints of $\bar{\mathcal{P}}_{FM}$ that are **tight** for both $\bar{y}$ and $\bar{z} \Rightarrow \bar{K}$ has at most $\leq |\bar{K}| + 1$ edges.
 - Obs:** An n -connected graph with $n + 1$ edges has ≤ 2 odd cycles!



Adjacency

- We explicitly highlight the following **adjacencies**:



(a)



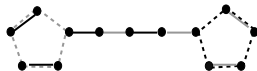
(b)



(c)



(d)



(e)



(f)

Adjacency

- We explicitly highlight the following **adjacencies**:



(a)



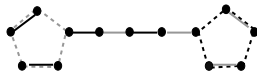
(b)



(c)



(d)



(e)



(f)

Exercise: *Prove that these fractional matchings are adjacent extreme points!*

Diameter

Diameter

Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \left\{ \mathbf{1}^T x + \frac{|C_x|}{2} \right\}$$

Diameter

Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \left\{ \mathbf{1}^T x + \frac{|\mathcal{C}_x|}{2} \right\}$$

- **Obs:** For a bipartite graph $\mathcal{C}_x = \emptyset$

Diameter

Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \{ \mathbf{1}^T x + \frac{|C_x|}{2} \}$$

- **Obs:** For a bipartite graph $C_x = \emptyset \rightarrow \text{diameter}(\mathcal{P}_{FM}) = \text{diameter}(\mathcal{P}_M)$.

Diameter

Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \left\{ \mathbf{1}^T x + \frac{|\mathcal{C}_x|}{2} \right\}$$

- **Obs:** For a bipartite graph $\mathcal{C}_x = \emptyset \rightarrow \text{diameter}(\mathcal{P}_{FM}) = \text{diameter}(\mathcal{P}_M)$.

Lower bound: Let w be any vertex.



Diameter

Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \{ \mathbf{1}^T x + \frac{|\mathcal{C}_x|}{2} \}$$

- **Obs:** For a bipartite graph $\mathcal{C}_x = \emptyset \rightarrow \text{diameter}(\mathcal{P}_{FM}) = \text{diameter}(\mathcal{P}_M)$.

Lower bound: Let w be any vertex.

- Show: the **distance** between w and the **0**-vertex is $\geq \mathbf{1}^T w + \frac{|\mathcal{C}_w|}{2}$.



Diameter

Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \left\{ \mathbf{1}^T x + \frac{|\mathcal{C}_x|}{2} \right\}$$

- **Obs:** For a bipartite graph $\mathcal{C}_x = \emptyset \rightarrow \text{diameter}(\mathcal{P}_{FM}) = \text{diameter}(\mathcal{P}_M)$.

Lower bound: Let w be any vertex.

- Show: the **distance** between w and the **0**-vertex is $\geq \mathbf{1}^T w + \frac{|\mathcal{C}_w|}{2}$.
 - ▶ Add a **loop edge** on each node v in $\text{support}(w)$.



Diameter

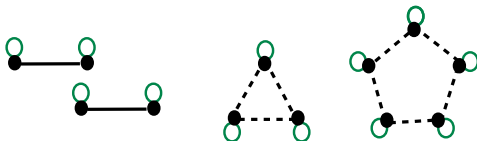
Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \{ \mathbf{1}^T x + \frac{|\mathcal{C}_x|}{2} \}$$

- **Obs:** For a bipartite graph $\mathcal{C}_x = \emptyset \rightarrow \text{diameter}(\mathcal{P}_{FM}) = \text{diameter}(\mathcal{P}_M)$.

Lower bound: Let w be any vertex.

- Show: the **distance** between w and the **0**-vertex is $\geq \mathbf{1}^T w + \frac{|\mathcal{C}_w|}{2}$.
 - ▶ Add a **loop edge** on each node v in $\text{support}(w)$.



Diameter

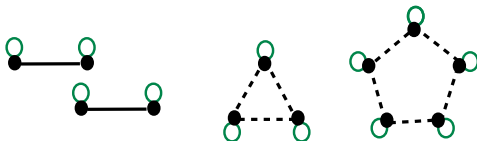
Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \{ \mathbf{1}^T x + \frac{|\mathcal{C}_x|}{2} \}$$

- **Obs:** For a bipartite graph $\mathcal{C}_x = \emptyset \rightarrow \text{diameter}(\mathcal{P}_{FM}) = \text{diameter}(\mathcal{P}_M)$.

Lower bound: Let w be any vertex.

- Show: the **distance** between w and the **0**-vertex is $\geq \mathbf{1}^T w + \frac{|\mathcal{C}_w|}{2}$.
 - ▶ Add a **loop edge** on each node v in $\text{support}(w)$.
 - ▶ **Note:** $(\# \text{ of odd cycles}) = 2 \left(\mathbf{1}^T w + \frac{|\mathcal{C}_w|}{2} \right)$



Diameter

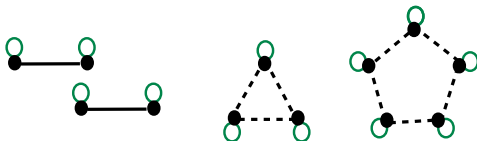
Theorem [S.'18]

$$\text{diameter}(\mathcal{P}_{FM}) = \max_{x \in \text{vertices}(\mathcal{P}_{FM})} \{ \mathbf{1}^T x + \frac{|\mathcal{C}_x|}{2} \}$$

- **Obs:** For a bipartite graph $\mathcal{C}_x = \emptyset \rightarrow \text{diameter}(\mathcal{P}_{FM}) = \text{diameter}(\mathcal{P}_M)$.

Lower bound: Let w be any vertex.

- Show: the **distance** between w and the **0**-vertex is $\geq \mathbf{1}^T w + \frac{|\mathcal{C}_w|}{2}$.
 - ▶ Add a **loop edge** on each node v in $\text{support}(w)$.
 - ▶ **Note:** $(\# \text{ of odd cycles}) = 2 \left(\mathbf{1}^T w + \frac{|\mathcal{C}_w|}{2} \right)$
 - ▶ At each move, the above quantity can decrease by at most 2



Upper bound

- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, the selection of the moves to take is not straightforward.

Upper bound

- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, the selection of the moves to take is not straightforward. Recall the adjacencies we mentioned:



(a)



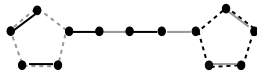
(b)



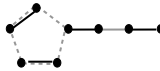
(c)



(d)



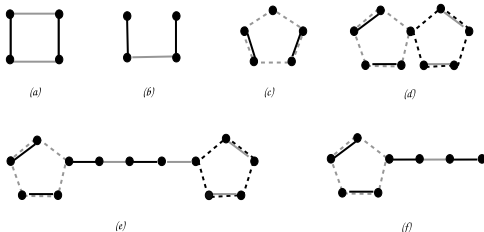
(e)



(f)

Upper bound

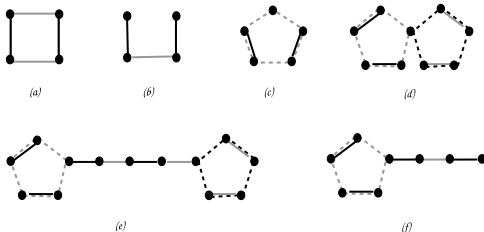
- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, the selection of the moves to take is not straightforward. Recall the adjacencies we mentioned:



- An easy “attempt” to go from z to y would be to define:

Upper bound

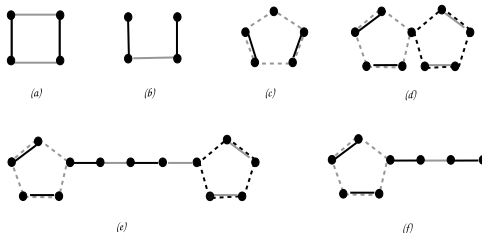
- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, the selection of the moves to take is not straightforward. Recall the adjacencies we mentioned:



- An easy “attempt” to go from z to y would be to define:
 - (i) a path from z to a 0/1-vertex $\bar{z}$ by removing one $C \in \mathcal{C}_z$ at each step

Upper bound

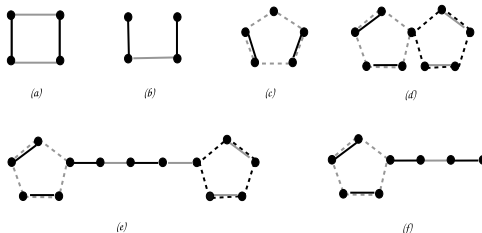
- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, the selection of the moves to take is not straightforward. Recall the adjacencies we mentioned:



- An easy “attempt” to go from z to y would be to define:
 - ▶ (i) a path from z to a 0/1-vertex $\bar{z}$ by removing one $C \in \mathcal{C}_z$ at each step
 - ▶ (ii) a path from y to a 0/1-vertex $\bar{y}$ by removing one $C \in \mathcal{C}_y$ at each step

Upper bound

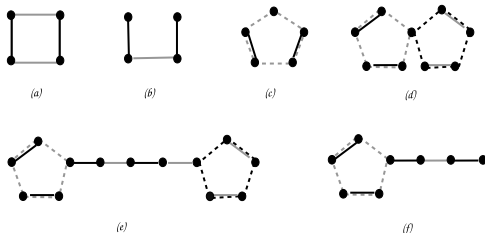
- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, the selection of the moves to take is not straightforward. Recall the adjacencies we mentioned:



- An easy “attempt” to go from z to y would be to define:
 - ▶ (i) a path from z to a 0/1-vertex $\bar{z}$ by removing one $C \in \mathcal{C}_z$ at each step
 - ▶ (ii) a path from y to a 0/1-vertex $\bar{y}$ by removing one $C \in \mathcal{C}_y$ at each step
 - ▶ (iii) a path from $\bar{z}$ to $\bar{y}$ (e.g. using the 1-skeleton of $\mathcal{P}_M$)

Upper bound

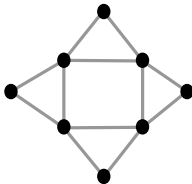
- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, the selection of the moves to take is not straightforward. Recall the adjacencies we mentioned:



- An easy “attempt” to go from z to y would be to define:
 - ▶ (i) a path from z to a 0/1-vertex $\bar{z}$ by removing one $C \in \mathcal{C}_z$ at each step
 - ▶ (ii) a path from y to a 0/1-vertex $\bar{y}$ by removing one $C \in \mathcal{C}_y$ at each step
 - ▶ (iii) a path from $\bar{z}$ to $\bar{y}$ (e.g. using the 1-skeleton of $\mathcal{P}_M$)
- ...but unfortunately this may lead to paths **longer** than the claimed bound!

Upper bound

- Bad example:



Upper bound

- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, we
 - ▶ Define a **path** of the form: $z \rightarrow w \rightarrow y$ for some **maximal** vertex w of $\mathcal{P}_{FM}$ satisfying: $\text{support}(w) \subseteq \text{support}(z) \cup \text{support}(y)$

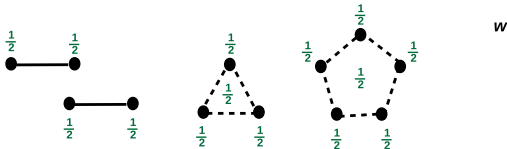
Upper bound

- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, we
 - ▶ Define a **path** of the form: $z \rightarrow w \rightarrow y$ for some **maximal** vertex w of $\mathcal{P}_{FM}$ satisfying: $\text{support}(w) \subseteq \text{support}(z) \cup \text{support}(y)$
 - ▶ Rely on a **token** argument: assign a token of **value** $\frac{1}{2}$ to each node v and each cycle C in $\text{support}(w)$



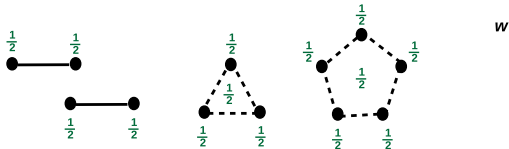
Upper bound

- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, we
 - Define a **path** of the form: $z \rightarrow w \rightarrow y$ for some **maximal** vertex w of $\mathcal{P}_{FM}$ satisfying: $\text{support}(w) \subseteq \text{support}(z) \cup \text{support}(y)$
 - Rely on a **token** argument: assign a token of **value** $\frac{1}{2}$ to each node v and each cycle C in $\text{support}(w)$



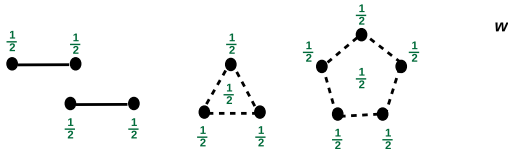
Upper bound

- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, we
 - Define a **path** of the form: $z \rightarrow w \rightarrow y$ for some **maximal** vertex w of $\mathcal{P}_{FM}$ satisfying: $\text{support}(w) \subseteq \text{support}(z) \cup \text{support}(y)$
 - Rely on a **token** argument: assign a token of **value** $\frac{1}{2}$ to each node v and each cycle C in $\text{support}(w)$ (**Note**: total token value = $\mathbf{1}^T w + \frac{|C_w|}{2}$)



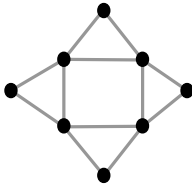
Upper bound

- Given two distinct vertices z and y of $\mathcal{P}_{FM}$, we
 - Define a **path** of the form: $z \rightarrow w \rightarrow y$ for some **maximal** vertex w of $\mathcal{P}_{FM}$ satisfying: $\text{support}(w) \subseteq \text{support}(z) \cup \text{support}(y)$
 - Rely on a **token** argument: assign a token of **value** $\frac{1}{2}$ to each node v and each cycle C in $\text{support}(w)$ (**Note**: total token value = $\mathbf{1}^T w + \frac{|C_w|}{2}$)
 - Show: each **move** on the path can be **payed** using **two tokens** of nodes/cycles



Upper bound

- Example:



Algorithmic and hardness implications

Hardness

Theorem [S.'18]

Computing the diameter of a polytope is a strongly NP-hard problem.

Hardness

Theorem [S.'18]

Computing the diameter of a polytope is a strongly NP-hard problem.

- Reduction from the (strongly) NP-hard problem **Partition Into Triangles**.
 - ▶ **Given:** A graph $G = (V, E)$
 - ▶ **Decide:** V can be partitioned into $\{V_1, \dots, V_q\}$: $\forall i, V_i$ induces a triangle

Hardness

Theorem [S.'18]

Computing the diameter of a polytope is a strongly NP-hard problem.

- Reduction from the (strongly) NP-hard problem **Partition Into Triangles**.
 - ▶ **Given:** A graph $G = (V, E)$
 - ▶ **Decide:** V can be partitioned into $\{V_1, \dots, V_q\}$: $\forall i, V_i$ induces a triangle
- Given G , consider the fractional matching polytope $\mathcal{P}_{FM}$ associated to G .
- Let x be a vertex of $\mathcal{P}_{FM}$. Then: (i) $\mathbf{1}^T x \leq \frac{|V|}{2}$ (ii) $\frac{|C_x|}{2} \leq \frac{|V|}{6}$

Hardness

Theorem [S.'18]

Computing the diameter of a polytope is a strongly NP-hard problem.

- Reduction from the (strongly) NP-hard problem **Partition Into Triangles**.
 - ▶ **Given:** A graph $G = (V, E)$
 - ▶ **Decide:** V can be partitioned into $\{V_1, \dots, V_q\}$: $\forall i, V_i$ induces a triangle
- Given G , consider the fractional matching polytope $\mathcal{P}_{FM}$ associated to G .
- Let x be a vertex of $\mathcal{P}_{FM}$. Then: (i) $\mathbf{1}^T x \leq \frac{|V|}{2}$ (ii) $\frac{|C_x|}{2} \leq \frac{|V|}{6}$

Proposition: $\text{diam}(\mathcal{P}_{FM}) = \frac{2}{3}|V| \Leftrightarrow G$ is a yes-instance to *PIT*.



Hardness

Theorem [S.'18]

Computing the diameter of a polytope is a strongly NP-hard problem.

- Reduction from the (strongly) NP-hard problem **Partition Into Triangles**.
 - ▶ **Given:** A graph $G = (V, E)$
 - ▶ **Decide:** V can be partitioned into $\{V_1, \dots, V_q\}$: $\forall i, V_i$ induces a triangle
- Given G , consider the fractional matching polytope $\mathcal{P}_{FM}$ associated to G .
- Let x be a vertex of $\mathcal{P}_{FM}$. Then: (i) $\mathbf{1}^T x \leq \frac{|V|}{2}$ (ii) $\frac{|C_x|}{2} \leq \frac{|V|}{6}$

Proposition: $\text{diam}(\mathcal{P}_{FM}) = \frac{2}{3}|V| \Leftrightarrow G$ is a yes-instance to *PIT*.



- With some extra effort, we can strengthen the result to show **APX-hardness**.

Hardness

Hardness

- Do the previous results have some **hardness** implication on the performance Simplex algorithm? Not in the current form...

Hardness

- Do the previous results have some **hardness** implication on the performance Simplex algorithm? Not in the current form... but some implications can be derived easily with a little extra work!

Hardness

- Do the previous results have some **hardness** implication on the performance Simplex algorithm? Not in the current form... but some implications can be derived easily with a little extra work!
- In particular, one can observe the following (see [De Loera, Kafer, S.'19]):

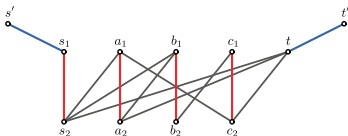
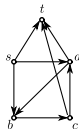
*Given a vertex of a **bipartite** matching polytope and an objective function, deciding if there exists a neighboring **optimal** vertex is NP-hard.*

Hardness

- Do the previous results have some **hardness** implication on the performance Simplex algorithm? Not in the current form... but some implications can be derived easily with a little extra work!
- In particular, one can observe the following (see [De Loera, Kafer, S.'19]):

*Given a vertex of a **bipartite** matching polytope and an objective function, deciding if there exists a neighboring **optimal** vertex is NP-hard.*

- **Proof.** Reduction: Given a directed graph H we:
 - ▶ construct a **bipartite** graph G , extreme point x of $\mathcal{P}_{FM}(G)$, obj function c .
 - ▶ show that $\exists$ a **neighboring optimal** extreme point of x iff H is Hamiltonian.



Hardness

- Do the previous results have some **hardness** implication on the performance Simplex algorithm? Not in the current form... but some implications can be derived easily with a little extra work!
- In particular, one can observe the following (see [De Loera, Kafer, S.'19]):
*Given a vertex of a **bipartite** matching polytope and an objective function, deciding if there exists a neighboring **optimal** vertex is NP-hard.*
- **Note:** Similar observation in [Barahona&Tardos'89] for circulation polytope.

Hardness

- Do the previous results have some **hardness** implication on the performance Simplex algorithm? Not in the current form... but some implications can be derived easily with a little extra work!

- In particular, one can observe the following (see [De Loera, Kafer, S.'19]):

*Given a vertex of a **bipartite** matching polytope and an objective function, deciding if there exists a neighboring **optimal** vertex is NP-hard.*

- **Note:** Similar observation in [Barahona&Tardos'89] for circulation polytope.

Corollary

*Finding the **shortest** monotone path to an optimal solution is NP-hard, and hard-to-approximate within a factor better than 2.*

Hardness

- Do the previous results have some **hardness** implication on the performance Simplex algorithm? Not in the current form... but some implications can be derived easily with a little extra work!

- In particular, one can observe the following (see [De Loera, Kafer, S.'19]):

*Given a vertex of a **bipartite** matching polytope and an objective function, deciding if there exists a neighboring **optimal** vertex is NP-hard.*

- **Note:** Similar observation in [Barahona&Tardos'89] for circulation polytope.

Corollary

*Finding the **shortest** monotone path to an optimal solution is NP-hard, and hard-to-approximate within a factor better than 2.*

- **Consequences** (unless $P=NP$):
 - ▶ For **any** efficient pivoting rule, an edge-augmentation algorithm (like Simplex) can't reach the optimum with a **min number** of augmentations.

Final remarks

- **Main questions:**

- ▶ Is the polynomial-Hirsch conjecture true?
- ▶ Is there a polynomial pivoting rule for the Simplex algorithm?

Final remarks

- **Main questions:**
 - ▶ Is the **polynomial-Hirsch conjecture** true?
 - ▶ Is there a polynomial pivoting rule for the **Simplex** algorithm?
- Diameter of the **perfect** matching polytope?

Final remarks

- Main questions:

- ▶ Is the polynomial-Hirsch conjecture true?
- ▶ Is there a polynomial pivoting rule for the Simplex algorithm?

- Diameter of the perfect matching polytope?

- All the hardness results discussed are for non simple polytopes.

→ A d -dimensional polytope is simple if every vertex is in exactly d facets

Final remarks

- Main questions:

- ▶ Is the polynomial-Hirsch conjecture true?
- ▶ Is there a polynomial pivoting rule for the Simplex algorithm?

- Diameter of the perfect matching polytope?

- All the hardness results discussed are for non simple polytopes.

→ A d -dimensional polytope is simple if every vertex is in exactly d facets

- ▶ Can one extend them to simple polytopes?
- ▶ Note: the complexity of computing the diameter of a simple polytope is mentioned as an open question in the survey of [Kaibel&Pfetsch'03]

Final remarks

- **Main questions:**

- ▶ Is the **polynomial-Hirsch conjecture** true?
- ▶ Is there a polynomial pivoting rule for the **Simplex** algorithm?

- Diameter of the **perfect** matching polytope?

- All the hardness results discussed are for **non simple** polytopes.

→ A d -dimensional polytope is **simple** if every vertex is in exactly d facets

- ▶ Can one extend them to simple polytopes?
- ▶ Note: the complexity of computing the diameter of a simple polytope is mentioned as an open question in the survey of [\[Kaibel&Pfetsch'03\]](#)

Thank you!

