Short simplex paths in lattice polytopes

Alberto Del Pia   Carla Michini
University of Wisconsin-Madison

Lehigh University – Industrial and Systems Engineering
October 20, 2020

Polyhedra and Polytopes

$P \subseteq \mathbb{R}^n$ is a polyhedron if, for some $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

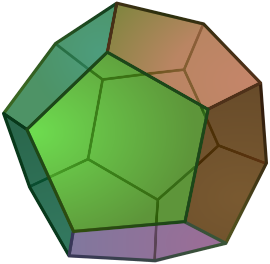
$$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$$

A bounded polyhedron is called polytope

A vertex is a 0-dimensional face

An edge is a 1-dimensional face

Two vertices are adjacent if they lie on a common edge



Polyhedra and Polytopes

$P \subseteq \mathbb{R}^n$ is a polyhedron if, for some $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

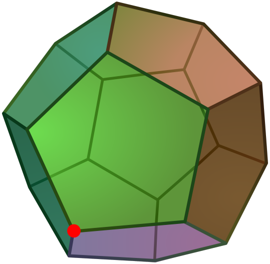
$$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$$

A bounded polyhedron is called polytope

A vertex is a 0-dimensional face

An edge is a 1-dimensional face

Two vertices are adjacent if they lie on a common edge



Polyhedra and Polytopes

$P \subseteq \mathbb{R}^n$ is a polyhedron if, for some $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

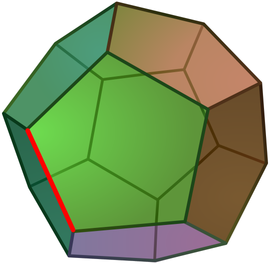
$$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$$

A bounded polyhedron is called polytope

A vertex is a 0-dimensional face

An edge is a 1-dimensional face

Two vertices are adjacent if they lie on a common edge



Polyhedra and Polytopes

$P \subseteq \mathbb{R}^n$ is a polyhedron if, for some $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

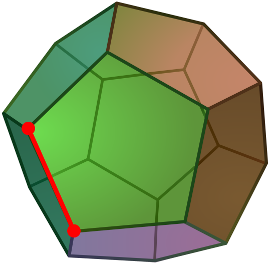
$$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$$

A bounded polyhedron is called polytope

A vertex is a 0-dimensional face

An edge is a 1-dimensional face

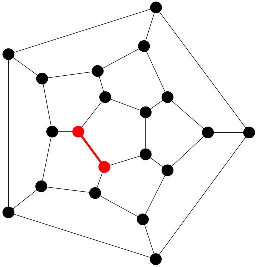
Two vertices are adjacent if they lie on a common edge

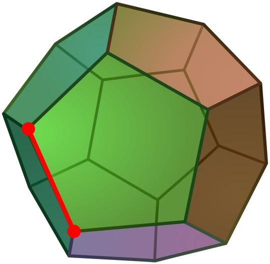


1-skeleton

The 1-skeleton of P is a graph having:

- a node for each vertex of P
- an edge joining two nodes $\Leftrightarrow$ the corresponding vertices of P are adjacent





Linear Programming (LP)

$$\begin{array}{ll} \max & c^\top x \\ \text{s.t.} & x \in P \end{array}$$

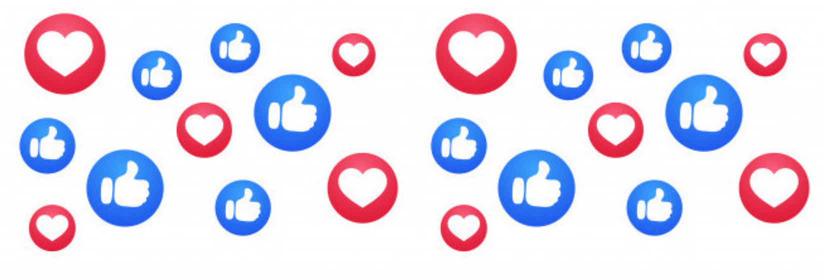
- $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$
- P is a polyhedron in $\mathbb{R}^n$, where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$
- $c \in \mathbb{R}^n$

A powerful tool for tackling wide classes of optimization problems

The simplex method

The most popular algorithm for LP is

the simplex method

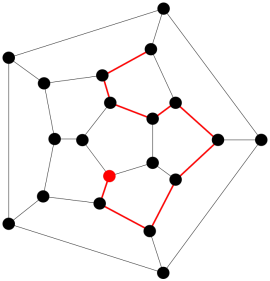


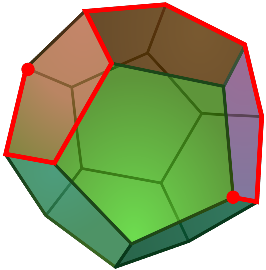
The most popular algorithm for LP is

the simplex method

The simplex method

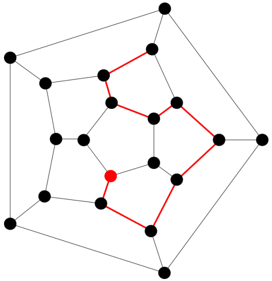
- Start at an initial vertex of P
- Move along an (improving) edge to an adjacent vertex
- Repeat until an optimal vertex is found or unboundedness is detected

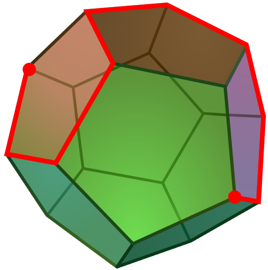




The simplex method

- Start at an initial vertex of P
- Move along an (improving) edge to an adjacent vertex [pivot]
- Repeat until an optimal vertex is found or unboundedness is detected





The choice of the next vertex depends on a pivoting rule

The simplex method

MAJOR OPEN QUESTION $\#1$:

pivoting rule that performs a polynomial nb of steps?

The simplex method

MAJOR OPEN QUESTION #1:

pivoting rule that performs a polynomial nb of steps?

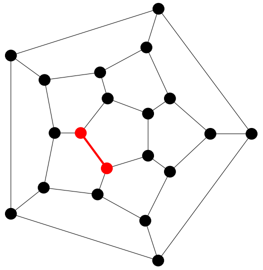
MAJOR OPEN QUESTION #2:

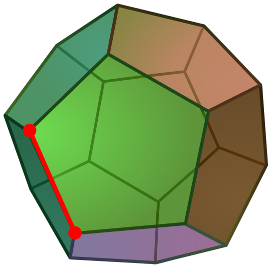
'best possible' pivoting rule performs a polynomial nb of steps?

Diameter

Distance between two vertices:

length of shortest path between two vertices on 1-skeleton of P

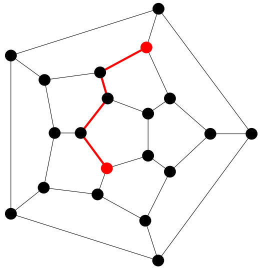


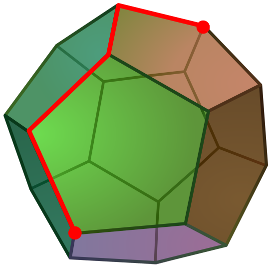


Diameter

Distance between two vertices:

length of shortest path between two vertices on 1-skeleton of P

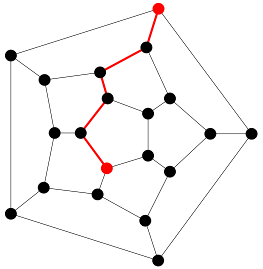


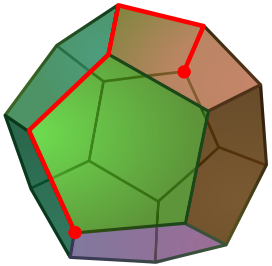


Diameter

Diameter:

largest value among the distances between any pair of vertices of P





The diameter of P is a LOWER BOUND on the worst-case simplex path length for a LP problem on P

The simplex method

MAJOR OPEN QUESTION #1:

pivoting rule that performs a polynomial nb of steps?

MAJOR OPEN QUESTION #2: (Polynomial Hirsch conjecture)

'best possible' pivoting rule performs polynomial nb of steps?
$\Leftrightarrow$ diameter bounded by a polynomial?

The simplex method

MAJOR OPEN QUESTION #1:

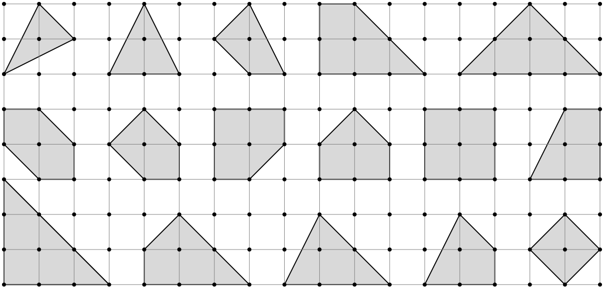
pivoting rule that performs a polynomial nb of steps?

MAJOR OPEN QUESTION #2: (Polynomial Hirsch conjecture)

'best possible' pivoting rule performs polynomial nb of steps?
$\Leftrightarrow$ diameter bounded by a polynomial?

Lattice polytopes

Lattice polytopes



Lattice polytope: polytope whose vertices are integral

LP on lattice polytopes

We study the LP problem:

$$\begin{array}{ll} \max & c^\top x \\ \text{s.t.} & x \in P \end{array}$$

- P is a lattice polytope in $[0, k]^n$
- $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$, where $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$
- $c \in \mathbb{Z}^n$

Applications in: polyhedral combinatorics, integer programming, combinatorial optimization

LP on lattice polytopes

We study the LP problem:

$$\begin{aligned} \max \quad & c^\top x \\ \text{s.t.} \quad & x \in P \end{aligned}$$

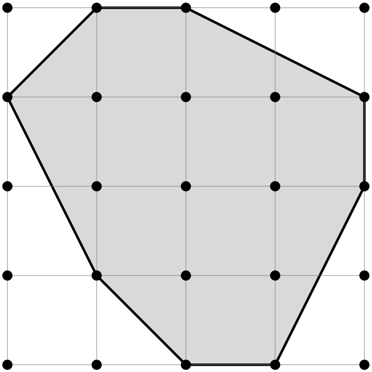
- P is a lattice polytope in $[0, k]^n$
- $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$, where $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$
- $c \in \mathbb{Z}^n$

GOAL: Simplex algorithm that traces "short" simplex paths on P from given vertex x^0 to optimal vertex x^*

...possibly, polynomially far from the worst-case diameter

How "short" can a simplex path be on a lattice polytope?

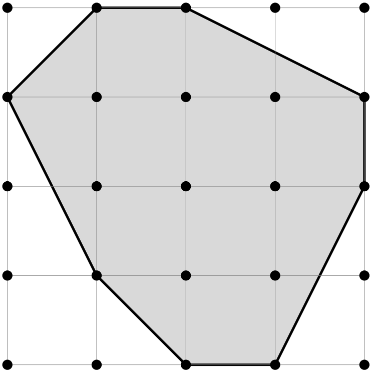
Diameter of lattice polytopes in $[0, k]^n$



Upper bounds:

- n if $k = 1$ [Naddef 89]
- nk [Kleinschmidt Onn 92]
- $\left\lfloor n(k - \frac{1}{2}) \right\rfloor$ if $k \geq 2$ [Del Pia Michini 16]
- $nk - \left\lceil \frac{2}{3} n \right\rceil - (k - 3)$ if $k \geq 3$ [Deza Pournin 18]

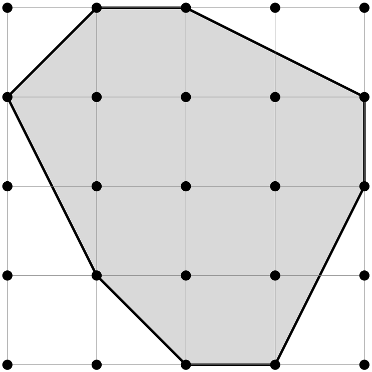
Diameter of lattice polytopes in $[0, k]^n$



Lower bounds:

- n if $k = 1$
- $\left\lfloor \frac{3}{2} n \right\rfloor$ if $k = 2$ [Del Pia Michini 16]
- $\left\lfloor \frac{1}{2} n(k+1) \right\rfloor$ if $k < 2n$ [Deza Manoussakis Onn 18]
- $ck^{\frac{2}{3}}$ if $n = 2$, $k \to \infty$ [Balog Bárány 91]
- $c(n)k^{\frac{n}{n+1}}$ if n fixed, $k \to \infty$ [Deza Pournin Sukegawa 19]

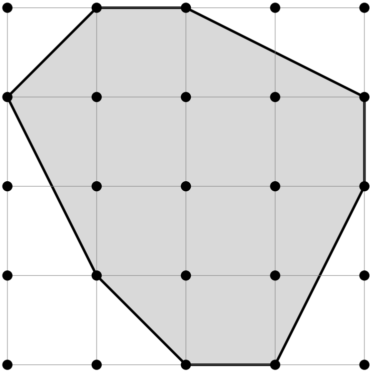
Diameter of lattice polytopes in $[0, k]^n$



Take-home message:

Worst-case diameter $\approx nk$.

Diameter of lattice polytopes in $[0, k]^n$



Take-home message:
Worst-case diameter $\approx nk$.

GOAL: Simplex algorithm that traces simplex paths of length polynomially far from nk

Other simplex algorithms

Dantzig, **best improvement** [Kitahara Mizuno '12], and **steepest edge** [Blanchard, De Loera, Louveaux '20] pivoting rules:

$\Rightarrow$ simplex path length

$$\leq n^2 \max\{k, S\} \log(n \max\{k, S\})$$

where $S = \max\{\|b - Ax\|_\infty \mid x \in P\}$.

- However even for $k = 1$, S can be $\frac{(n-1)^{\frac{n-1}{2}}}{2^{2n+o(n)}}$
- This bound is not polynomially bounded in nk.

Other simplex algorithms

Dantzig, **best improvement** [Kitahara Mizuno '12], and **steepest edge** [Blanchard, De Loera, Louveaux '20] pivoting rules:

$\Rightarrow$ simplex path length

$$\leq n^2 \max\{k, S\} \log(n \max\{k, S\})$$

where $S = \max\{\|b - Ax\|_\infty \mid x \in P\}$.

- However even for $k = 1$, S can be $\frac{(n-1)^{\frac{n-1}{2}}}{2^{2n+o(n)}}$
- This bound is not polynomially bounded in nk.

Can we eliminate dependence on S?

Short simplex paths in lattice polytopes

Theorem 1: Simplex algorithm for lattice polytopes in $[0, k]^n$ s.t. simplex path length in $O(n^4 k \log(nk))$

The simplex path length is polynomially far from optimal $\approx nk$

Short simplex paths in lattice polytopes

Theorem 1: Simplex algorithm for lattice polytopes in $[0, k]^n$ s.t. simplex path length in $O(n^4 k \log(nk))$

Independent on:

- cost vector c
- description $Ax \leq b$ of P
- number of inequalities m

Short simplex paths in lattice polytopes

Theorem 1: Simplex algorithm for lattice polytopes in $[0, k]^n$ s.t. simplex path length in $O(n^4 k \log(nk))$

Most known lattice polytopes are defined via $0, \pm 1$ constraint matrices

Can we exploit the largest absolute value α of the entries in the constraint matrix?

Short simplex paths in lattice polytopes

Theorem 2: Simplex algorithm for lattice polytopes in $[0, k]^n$ s.t. simplex path length in $O(n^2 k \log(nk\alpha))$

Most known lattice polytopes are defined via $0, \pm 1$ constraint matrices

Can we exploit the largest absolute value α of the entries in the constraint matrix?

Short simplex paths in lattice polytopes

Theorem 2: Simplex algorithm for lattice polytopes in $[0, k]^n$ s.t. simplex path length in $O(n^2 k \log(nk\alpha))$

- If $\alpha \leq \text{poly}(n, k)$, then simplex path length in $O(n^2 k \log(nk))$
- If also $k = 1$, then simplex path length in $O(n^2 \log n)$

How does it work?

How does it work?

We move to an adjacent vertex by calling:

Oracle

Input: Polytope P, $c \in \mathbb{Z}^n$, vertex $\bar{x}$ of P

Output:

- Either a statement that $\bar{x}$ maximizes $c^\top x$ over P
- or a vertex adjacent to $\bar{x}$ with strictly larger cost

---

The input to the oracle is key to compute a short simplex path...

How does it work?

1. **Basic algorithm** length $\leq nk\,\|c\|_\infty$

2. **Scaling algorithm** length $O(nk\log\|c\|_\infty)$

3. **Preprocessing & scaling algorithm** length $O(n^4 k\log(nk))$

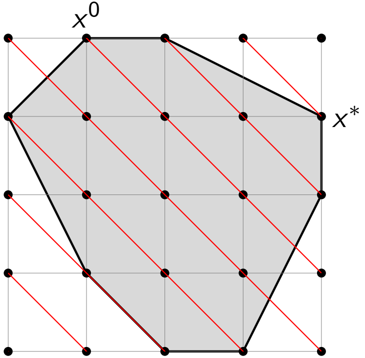
4. **Face-fixing algorithm** length $O(n^2 k\log(nk\alpha))$

Basic algorithm

Basic algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$.

for $t = 0, 1, 2, \ldots$ **do**
Invoke **oracle**(P, c, x^t)
If the **oracle** states that x^t is optimal, return x^t
Otherwise, $x^{t+1} :=$ oracle(P, c, x^t)



Observation: The length of the simplex path generated is at most $c^\top x^* - c^\top x^0 \leq nk \|c\|_\infty$

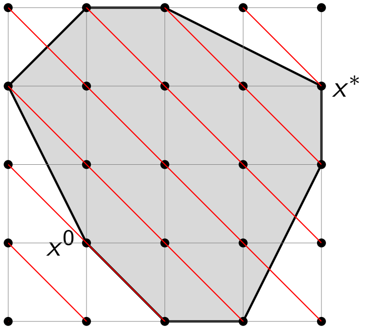
Example: $c = (1, 1)$

Basic algorithm

Basic algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$.

for $t = 0, 1, 2, \ldots$ **do**
  Invoke **oracle**(P, c, x^t)
  If the **oracle** states that x^t is optimal, return x^t
  Otherwise, $x^{t+1} :=$ oracle(P, c, x^t)



Observation: The length of the simplex path generated is at most $c^\top x^* - c^\top x^0 \leq nk \, \|c\|_\infty$

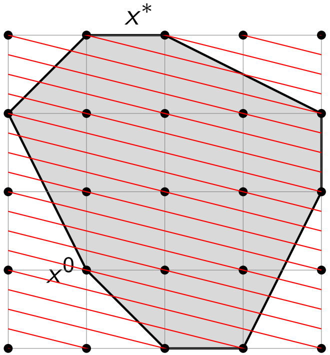
Example: $c = (1, 1)$

Basic algorithm

Basic algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$.

for $t = 0, 1, 2, \ldots$ **do**
 Invoke **oracle**(P, c, x^t)
 If the **oracle** states that x^t is optimal, return x^t
 Otherwise, $x^{t+1} :=$ oracle(P, c, x^t)



Observation: The length of the simplex path generated is at most $c^\top x^* - c^\top x^0 \leq nk \|c\|_\infty$

Example: $c = (1, 4)$

Scaling algorithm

Let $\ell := \lceil \log \|c\|_\infty \rceil$
For $t = 0, \ldots, \ell$, define the integral approximations of c:

$$c^t := \left\lceil \frac{c}{2^{\ell - t}} \right\rceil \qquad \text{(Note: } c^\ell = c\text{)}$$

Example:

$$\begin{aligned} c &= (1,2,3,4,5,6,7) \\ c^0 &= (1,1,1,1,1,1,1) \\ c^1 &= (1,1,1,1,2,2,2) \\ c^2 &= (1,1,2,2,3,3,4) \\ c^3 &= (1,2,3,4,5,6,7) \end{aligned}$$

- $\|c^t\|_\infty \leq 2^t$ for $t = 0, \ldots, \ell$
- $c^\ell = c$
- $2c^{t-1} - c^t \in \{0,1\}^n$ for $t = 1, \ldots, \ell$

Scaling algorithm

For $t = 0, \ldots, \ell$: $\quad c^t := \left\lceil \frac{c}{2^{\ell - t}} \right\rceil$

Scaling algorithm

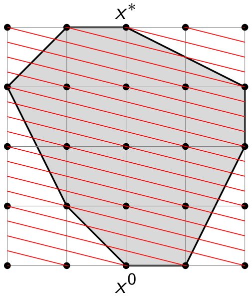
Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**
  $x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)
Return $x^{\ell+1}$

Scaling algorithm

For $t = 0, \dots, \ell$: $\quad c^t := \left\lceil \frac{c}{2^{\ell - t}} \right\rceil$

Example: $c = (1, 4)$



Scaling algorithm

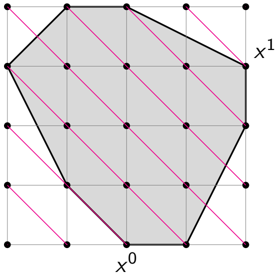
Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \dots, \ell$ **do**
$\quad x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)
Return $x^{\ell+1}$

Scaling algorithm

For $t = 0, \dots, \ell$: $\quad c^t := \left\lceil \frac{c}{2^{\ell - t}} \right\rceil$

Example: $c = (1, 4)$ $c^0 = (1, 1)$



Scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

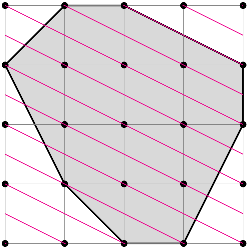
for $t = 0, \dots, \ell$ **do**
$\quad x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)

Return $x^{\ell+1}$

Scaling algorithm

For $t = 0, \ldots, \ell$: $\quad c^t := \left\lceil \frac{c}{2^{\ell - t}} \right\rceil$

Example: $c = (1, 4)$ $c^1 = (1, 2)$



Scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P

Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**

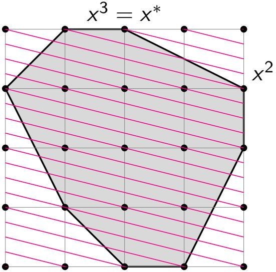
$\quad x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)

Return $x^{\ell+1}$

Scaling algorithm

For $t = 0, \ldots, \ell$: $\quad c^t := \left\lceil \frac{c}{2^{\ell - t}} \right\rceil$

Example: $c = (1, 4)$ $c^2 = (1, 4)$



Scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P

Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**

$\quad x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)

Return $x^{\ell+1}$

Scaling algorithm

Proposition: Simplex path length is in $O(nk \log \|c\|_\infty)$

Scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**
  $x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)
Return $x^{\ell+1}$

Scaling algorithm

Proposition: Simplex path length is in $O(nk \log \|c\|_\infty)$
Proof:

- **Basic algorithm** is called $\ell + 1 := \lceil \log \|c\|_\infty \rceil + 1$ times.
- Each time **basic algorithm** generates path of length $\leq nk$:
  - $2c^{t-1} - c^t \in \{0,1\}^n$
  - ${c^t}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^t$
  - ${c^t}^\top x^{t+1} - {c^t}^\top x^t \leq 2{c^{t-1}}^\top x^t - {c^t}^\top x^t = (2c^{t-1} - c^t)^\top x^t \leq nk$

Scaling algorithm

Input: Lattice polytope P in $[0,k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**
    $x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)
Return $x^{\ell+1}$

Scaling algorithm

Proposition: Simplex path length is in $O(nk \log \|c\|_\infty)$

Proof:

- **Basic algorithm** is called $\ell + 1 := \lceil \log \|c\|_\infty \rceil + 1$ times.
- Each time **basic algorithm** generates path of length $\leq nk$:
  - $2c^{t-1} - c^t \in \{0, 1\}^n$
  - ${c^t}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^t$
  - ${c^t}^\top x^{t+1} - {c^t}^\top x^t \leq 2{c^{t-1}}^\top x^t - {c^t}^\top x^t = (2c^{t-1} - c^t)^\top x^t \leq nk$

Scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P

Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**

  $x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)

Return $x^{\ell+1}$

Scaling algorithm

Proposition: Simplex path length is in $O(nk \log \|c\|_\infty)$

Proof:

- **Basic algorithm** is called $\ell + 1 := \lceil \log \|c\|_\infty \rceil + 1$ times.
- Each time **basic algorithm** generates path of length $\leq nk$:
  - $2c^{t-1} - c^t \in \{0,1\}^n$
  - ${c^t}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^t$
  - ${c^t}^\top x^{t+1} - {c^t}^\top x^t \leq 2{c^{t-1}}^\top x^t - {c^t}^\top x^t = (2c^{t-1} - c^t)^\top x^t \leq nk$

Scaling algorithm

Input: Lattice polytope P in $[0,k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P

Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**

$\quad x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)

Return $x^{\ell+1}$

Scaling algorithm

Proposition: Simplex path length is in $O(nk \log \|c\|_\infty)$
Proof:

- **Basic algorithm** is called $\ell + 1 := \lceil \log \|c\|_\infty \rceil + 1$ times.
- Each time **basic algorithm** generates path of length $\leq nk$:
  - $2c^{t-1} - c^t \in \{0, 1\}^n$
  - ${c^t}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^{t+1} \leq 2{c^{t-1}}^\top x^t$
  - ${c^t}^\top x^{t+1} - {c^t}^\top x^t \leq 2{c^{t-1}}^\top x^t - {c^t}^\top x^t = (2c^{t-1} - c^t)^\top x^t \leq nk$

Scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

for $t = 0, \ldots, \ell$ **do**
  $x^{t+1} :=$ **basic algorithm**(P, c^t, x^t)
Return $x^{\ell+1}$

Preprocessing algorithm

Preprocessing algorithm
Input: $c \in \mathbb{Q}^n$, positive integer N
Output: $\tilde{c} \in \mathbb{Z}^n$ such that
▶ $\|\tilde{c}\|_\infty \leq 2^{4n^3} N^{n(n+2)}$
▶ $\text{sign}(c^\top z) = \text{sign}(\tilde{c}^\top z)\ \forall z \in \mathbb{Z}^n$ with $\|z\|_1 \leq N-1$

- Due to [Frank Tardos 87]
- Relies on the simultaneous approximation algorithm of [Lenstra Lenstra Lovász 82]

Setting $N := nk + 1$, x^* optimal for $\tilde{c} \Rightarrow$ optimal for c:

- $\forall x \in P \cap \mathbb{Z}^n$:
- $x - x^* \in \mathbb{Z}^n$ and $\|x - x^*\|_1 \leq nk = N - 1$
- $\tilde{c}^\top (x - x^*) \leq 0 \quad \Rightarrow \quad c^\top (x - x^*) \leq 0$

Preprocessing algorithm

Preprocessing algorithm

Input: $c \in \mathbb{Q}^n$, positive integer N

Output: $\tilde{c} \in \mathbb{Z}^n$ such that

- $\|\tilde{c}\|_\infty \leq 2^{4n^3} N^{n(n+2)}$
- $\text{sign}(c^\top z) = \text{sign}(\tilde{c}^\top z)$ $\forall z \in \mathbb{Z}^n$ with $\|z\|_1 \leq N - 1$

- Due to [Frank Tardos 87]
- Relies on the simultaneous approximation algorithm of [Lenstra Lenstra Lovász 82]

Setting $N := nk + 1$, x^* optimal for $\tilde{c}$ $\Rightarrow$ optimal for c:

- $\forall x \in P \cap \mathbb{Z}^n$:
- $x - x^* \in \mathbb{Z}^n$ and $\|x - x^*\|_1 \leq nk = N - 1$
- $\tilde{c}^\top x^* \geq \tilde{c}^\top x \quad \Rightarrow \quad c^\top x^* \geq c^\top x$

Preprocessing & scaling algorithm

Preprocessing & scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

$\tilde{c} :=$ preprocessing algorithm$(c, N := nk + 1)$
$x^* :=$ **scaling algorithm**$(P, \tilde{c}, x^0)$
Return x^*

Theorem 1: Simplex path length in $O(nk \log \|\tilde{c}\|_\infty)$

$\log \|\tilde{c}\|_\infty$ in $O(n^3 \log(nk))$

Preprocessing & scaling algorithm

Preprocessing & scaling algorithm

Input: Lattice polytope P in $[0, k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

$\tilde{c} :=$ preprocessing algorithm$(c, N := nk + 1)$
$x^* :=$ **scaling algorithm**$(P, \tilde{c}, x^0)$
Return x^*

Theorem 1: Simplex path length in $O(n^4 k \log(nk))$

Face-fixing algorithm

GOAL: shorter simplex path length, dependent on α

$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$, where $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$
$\alpha :=$ largest absolute value of the entries of A

IDEA: Identify at each iteration one constraint of $Ax \leq b$ that is active at each optimal solution of $\max\{c^\top x \mid x \in P\}$

Inspired by [Tardos '86]

Face-fixing algorithm

Face-fixing algorithm

Input: Lattice polytope P in $[0,k]^n$, $c \in \mathbb{Z}^n$, vertex x^0 of P
Output: A vertex x^* of P maximizing $c^\top x$

0: Let $\mathcal{E} := \emptyset$ and $x^* := x^0$

1: Let $\bar{c}$ be the projection of c onto $\{x \in \mathbb{R}^n \mid a_i^\top x = 0 \ \forall i \in \mathcal{E}\}$.
If $\bar{c} = 0$ return x^*

2: Let $\tilde{c} \in \mathbb{Z}^n$ be defined by $\tilde{c}_i := \left\lfloor \frac{n^3 k \alpha}{\|\bar{c}\|_\infty} \bar{c}_i \right\rfloor$ for $i = 1, \dots, n$

3: Consider the following pair of primal and dual LP problems:

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \quad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \quad (\tilde{D})$$

Compute optimal vertex $\tilde{x}$ of $(\tilde{P})$ with **scaling alg** from x^*
Compute an optimal solution $\tilde{y}$ to $(\tilde{D})$ s.t. $[\dots]$
Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$
$\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$, $x^* \leftarrow \tilde{x}$ and go back to step 1

Main Results

(correctness)
Proposition: Vector x^* returned maximizes $c^\top x$ over P.

(short simplex paths)
Theorem 2: Simplex path length in $O(n^2 k \log(nk\alpha))$

(polynomial runtime)
Proposition: The number of operations to construct the next vertex in the simplex path is bounded by $\mathrm{poly}(n, m, \log \alpha, \log k)$. If P is 'well-described' by $Ax \leq b$, then it is bounded by $\mathrm{poly}(n, m, \log k)$.

Correctness – key lemma

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$

Lemma: Each optimal solution of $\max\{c^\top x \mid x \in P\}$ lies in F

Correctness – key lemma

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$

Lemma: Each optimal solution of $\max\{c^\top x \mid x \in P\}$ lies in F

- Beginning of the algorithm: $\mathcal{E} = \emptyset \Rightarrow F = P$.
- Prove claim is true when we update $\mathcal{E}$ in step 3:
  Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$. Set $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$

Correctness – key lemma

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: Each optimal solution of $\max\{c^\top x \mid x \in F\}$ satisfies

$$a_i^\top x = b_i \text{ for } i \in \mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

- Beginning of the algorithm: $\mathcal{E} = \emptyset \Rightarrow F = P$.
- Prove claim is true when we update $\mathcal{E}$ in step 3:
  Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$. Set $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$

Correctness – key lemma

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \quad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \quad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: Each optimal solution of $\max\{c^\top x \mid x \in F\}$ satisfies

$$a_i^\top x = b_i \text{ for } i \in \mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

Correctness – key lemma

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: Each optimal solution of $\max\{\bar{c}^\top x \mid x \in F\}$ satisfies

$$a_i^\top x = b_i \text{ for } i \in \mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

Correctness – key lemma

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: Each optimal solution of $\max\{\bar{c}^\top x \mid x \in F\}$ satisfies

$$a_i^\top x = b_i \text{ for } i \in \mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

Let $\hat{c}$ be defined by $\hat{c}_i := \frac{n^3 k \alpha}{\|\bar{c}\|_\infty} \bar{c}_i$ for $i = 1, \dots, n \quad \Rightarrow \quad \tilde{c} = \lfloor \hat{c} \rfloor$

Correctness – key lemma

$$\begin{array}{lll} \max & \tilde{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: Each optimal solution of $\max\{\hat{c}^\top x \mid x \in F\}$ satisfies

$$a_i^\top x = b_i \text{ for } i \in \mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

Let $\hat{c}$ be defined by $\hat{c}_i := \frac{n^3 k \alpha}{\|\bar{c}\|_\infty} \bar{c}_i$ for $i = 1, \ldots, n \quad \Rightarrow \quad \tilde{c} = \lfloor \hat{c} \rfloor$

Correctness – key lemma

$$\begin{array}{lll} \max & \hat{c}^\top x & \\ \text{s.t.} & a_i^\top x = b_i & i \in \mathcal{E} \\ & a_i^\top x \le b_i & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\hat{P}) \qquad \begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \ge 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E},\ a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: Each optimal solution of $\max\{\hat{c}^\top x \mid x \in F\}$ satisfies

$$a_i^\top x = b_i \text{ for } i \in \mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

Complementary slackness conditions for $(\hat{P})/(\tilde{D})$:
If $\tilde{y}$ optimal for $(\tilde{D})$ then $\forall\ \hat{x}$ optimal for $(\hat{P})$:

$$\tilde{y}_i > nk \quad \Rightarrow \quad a_i^\top \hat{x} = b_i \qquad\qquad i \in [m] \setminus \mathcal{E} \quad (*)$$

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \leq b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \leq n - j$
$\Rightarrow$ at most n iterations

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \leq b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \leq n - j$
$\Rightarrow$ at most n iterations

(correctness)
Proposition: Vector x^* returned maximizes $c^\top x$ over P.

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \leq b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \leq n - j$
$\Rightarrow$ at most n iterations

At each iteration, we run the **scaling algorithm** to solve $(\tilde{P})$
Obs: F is a lattice polytope in $[0, k]^n$ and $\|\tilde{c}\|_\infty \leq n^3 k \alpha$.

At each iteration the **scaling algorithm** constructs a simplex path of length in $O(nk \log \|\tilde{c}\|_\infty) \in O(nk \log(nk\alpha))$

Theorem 2: Simplex path length in $O(n^2 k \log(nk\alpha))$

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \leq b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \leq n - j$
$\Rightarrow$ at most n iterations

In step 3: Compute an optimal solution $\tilde{y}$ to $(\tilde{D})$ s.t. $[\dots]$
Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$, and set $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \le b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \le n - j$
$\Rightarrow$ at most n iterations

$\Uparrow$

We prove that at each iteration the rank of the row submatrix of A indexed by $\mathcal{E}$ increases by at least 1

In step 3: Compute an optimal solution $\tilde{y}$ to $(\tilde{D})$ s.t. $[\dots]$
Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$, and set $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \leq b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \leq n - j$
$\Rightarrow$ at most n iterations

$$\Uparrow$$

We prove that at each iteration the rank of the row submatrix of A indexed by $\mathcal{E}$ increases by at least 1

In step 3: Compute an optimal solution $\tilde{y}$ to $(\tilde{D})$ s.t. $[\ldots]$
Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$, and set $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$

(i) $\tilde{y}_j = 0 \; \forall \, j \in [m] \setminus \mathcal{E}$ s.t. a_j is linear combination of a_i, $i \in \mathcal{E}$

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \leq b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \leq n - j$
$\Rightarrow$ at most n iterations

$$\Uparrow$$

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

In step 3: Compute an optimal solution $\tilde{y}$ to $(\tilde{D})$ s.t. $[\ldots]$
Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$, and set $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$

(i) $\tilde{y}_j = 0 \; \forall \, j \in [m] \setminus \mathcal{E}$ s.t. a_j is linear combination of a_i, $i \in \mathcal{E}$

Short Simplex Paths – key lemma

At each iteration, we restrict to a face F of P defined as

$$F := \{x \in \mathbb{R}^n \mid a_i^\top x \leq b_i \text{ for } i \in [m] \setminus \mathcal{E}, a_i^\top x = b_i \text{ for } i \in \mathcal{E}\}$$

Lemma: At the end of iteration j: $\dim(F) \leq n - j$
$\Rightarrow$ at most n iterations

$$\Uparrow$$

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

In step 3: Compute an optimal solution $\tilde{y}$ to $(\tilde{D})$ s.t. $[\dots]$
Let $\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$, and set $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{H}$

(i) $\tilde{y}_j = 0 \; \forall \, j \in [m] \setminus \mathcal{E}$ s.t. a_j is linear combination of a_i, $i \in \mathcal{E}$
(ii) $\tilde{y}$ has at most n nonzero components

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

$$\begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \geq 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

$\tilde{y}$ optimal solution of $(\tilde{D})$ with at most n nonzero components

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

$$\begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \geq 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

$\tilde{y}$ optimal solution of $(\tilde{D})$ with at most n nonzero components

Let $\mathcal{B} = \{i \in \{1, \ldots, m\} \mid \tilde{y}_i \neq 0\}$

$$\tilde{c} = \sum_{i \in [m]} a_i \tilde{y}_i = \sum_{i \in \mathcal{B}} a_i \tilde{y}_i$$

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

$$\begin{aligned} \min \quad & b^\top y \\ \text{s.t.} \quad & A^\top y = \tilde{c} \\ & y_i \geq 0 \qquad i \in [m] \setminus \mathcal{E} \end{aligned} \tag{$\tilde{D}$}$$

$\tilde{y}$ optimal solution of $(\tilde{D})$ with at most n nonzero components

Let $\mathcal{B} = \{i \in \{1, \dots, m\} \mid \tilde{y}_i \neq 0\}$

$$\tilde{c} = \sum_{i \in [m]} a_i \tilde{y}_i = \sum_{i \in \mathcal{B}} a_i \tilde{y}_i$$

By contradiction, $\mathcal{H} \setminus \mathcal{E} = \emptyset$

$$\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

$\Rightarrow |\tilde{y}_j| \leq nk \; \forall\, j \in \mathcal{B} \setminus \mathcal{E}$

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

$$\begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \geq 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

$\tilde{y}$ optimal solution of $(\tilde{D})$ with at most n nonzero components

Let $\mathcal{B} = \{i \in \{1, \dots, m\} \mid \tilde{y}_i \neq 0\}$

$$\tilde{c} = \sum_{i \in [m]} a_i \tilde{y}_i = \sum_{i \in \mathcal{B}} a_i \tilde{y}_i$$

By contradiction, $\mathcal{H} \setminus \mathcal{E} = \emptyset$

$$\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

$\Rightarrow |\tilde{y}_j| \leq nk \ \forall\, j \in \mathcal{B} \setminus \mathcal{E}$

$$\|\tilde{c}\|_\infty \leq \sum_{j \in \mathcal{B}} \|a_j \tilde{y}_j\|_\infty = \sum_{j \in \mathcal{B}} \left(|\tilde{y}_j| \, \|a_j\|_\infty\right)$$

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

$$\begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \geq 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

$\tilde{y}$ optimal solution of $(\tilde{D})$ with at most n nonzero components

Let $\mathcal{B} = \{i \in \{1, \ldots, m\} \mid \tilde{y}_i \neq 0\}$

$$\tilde{c} = \sum_{i \in [m]} a_i \tilde{y}_i = \sum_{i \in \mathcal{B}} a_i \tilde{y}_i$$

By contradiction, $\mathcal{H} \setminus \mathcal{E} = \emptyset$

$$\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$$

$\Rightarrow |\tilde{y}_j| \leq nk \ \forall\, j \in \mathcal{B} \setminus \mathcal{E}$

Case 1: $\mathcal{B} \setminus \mathcal{E} = \mathcal{B}$

$$\|\tilde{c}\|_\infty \leq \sum_{j \in \mathcal{B}} \|a_j \tilde{y}_j\|_\infty = \sum_{j \in \mathcal{B}} \left(|\tilde{y}_j| \, \|a_j\|_\infty\right) \leq \sum_{j \in \mathcal{B}} (nk \cdot \alpha) \leq n^2 k \alpha$$

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

$$\begin{array}{lll} \min & b^\top y & \\ \text{s.t.} & A^\top y = \tilde{c} & \\ & y_i \geq 0 & i \in [m] \setminus \mathcal{E} \end{array} \qquad (\tilde{D})$$

$\tilde{y}$ optimal solution of $(\tilde{D})$ with at most n nonzero components

Let $\mathcal{B} = \{i \in \{1, \ldots, m\} \mid \tilde{y}_i \neq 0\}$

$$\tilde{c} = \sum_{i \in [m]} a_i \tilde{y}_i = \sum_{i \in \mathcal{B}} a_i \tilde{y}_i$$

By contradiction, $\mathcal{H} \setminus \mathcal{E} = \emptyset$

$\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$

$\Rightarrow |\tilde{y}_j| \leq nk \ \forall\, j \in \mathcal{B} \setminus \mathcal{E}$

Case 1: $\mathcal{B} \setminus \mathcal{E} = \mathcal{B}$

$$\|\tilde{c}\|_\infty \leq \sum_{j \in \mathcal{B}} \|a_j \tilde{y}_j\|_\infty = \sum_{j \in \mathcal{B}} \left(|\tilde{y}_j|\, \|a_j\|_\infty\right) \leq \sum_{j \in \mathcal{B}} (nk \cdot \alpha) \leq n^2 k\alpha$$

But $\tilde{c} = \left\lfloor \frac{n^3 k\alpha}{\|\bar{c}\|_\infty} \bar{c} \right\rfloor \Rightarrow \|\tilde{c}\|_\infty = n^3 k\alpha$

contradiction

We prove that at each iteration $\mathcal{H} \setminus \mathcal{E} \neq \emptyset$

$$\begin{aligned} \min \quad & b^\top y \\ \text{s.t.} \quad & A^\top y = \tilde{c} \\ & y_i \geq 0 \qquad i \in [m] \setminus \mathcal{E} \end{aligned} \tag{$\tilde{D}$}$$

$\tilde{y}$ optimal solution of $(\tilde{D})$ with at most n nonzero components

Let $\mathcal{B} = \{i \in \{1, \ldots, m\} \mid \tilde{y}_i \neq 0\}$

$\tilde{c} = \sum_{i \in [m]} a_i \tilde{y}_i = \sum_{i \in \mathcal{B}} a_i \tilde{y}_i$

By contradiction, $\mathcal{H} \setminus \mathcal{E} = \emptyset$

$\mathcal{H} := \{i \mid \tilde{y}_i > nk\}$

$\Rightarrow |\tilde{y}_j| \leq nk \ \forall \, j \in \mathcal{B} \setminus \mathcal{E}$

Case 2: $\mathcal{B} \setminus \mathcal{E} \neq \mathcal{B}$

$\|\tilde{c}\|_\infty \leq \sum_{j \in \mathcal{B}} \|a_j \tilde{y}_j\|_\infty = \sum_{j \in \mathcal{B}} \left(|\tilde{y}_j| \, \|a_j\|_\infty\right)$

more involved...

Thank you!