Wolfe's Combinatorial Method is Exponential

Jamie Haddock

UCLA Combinatorics Seminar, February 7, 2019

Computational and Applied Mathematics, UCLA


joint with Jesús De Loera and Luis Rademacher (UC Davis)
https://arxiv.org/abs/1710.02608

Minimum Norm Point (MNP(P))

Minimum Norm Point in Polytope

We are interested in solving the problem (MNP(P)):

$$\min_{\mathbf{x} \in P} \|\mathbf{x}\|_2$$

where P is a polytope, and determining the minimum dimension face, F, which achieves distance $\|\mathbf{x}\|_2$.

Minimum Norm Point in Polytope

We are interested in solving the problem (MNP(P)):

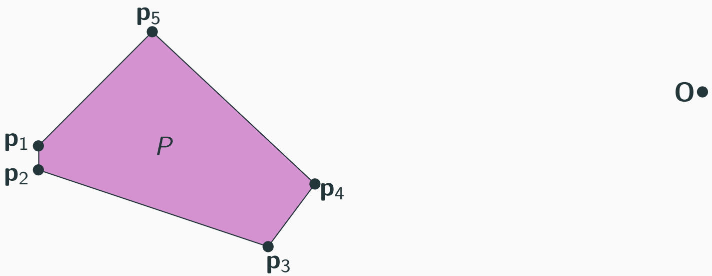
$$\min_{\mathbf{x} \in P} \|\mathbf{x}\|_2$$

where P is a polytope, and determining the minimum dimension face, F, which achieves distance $\|\mathbf{x}\|_2$.

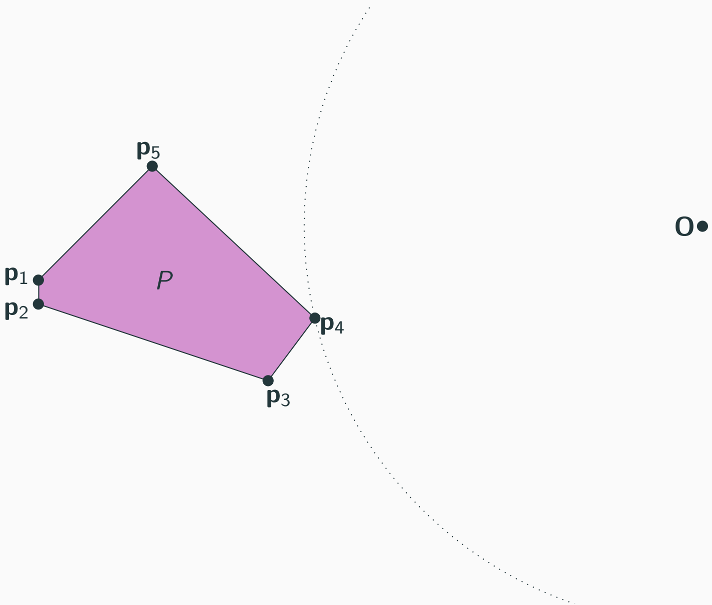
Note: We consider polytopes, P, given in V-representation as the convex hull of points $\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m$,

$$P = \left\{ \sum_{i=1}^{m} \lambda_i \mathbf{p}_i : \sum_{i=1}^{m} \lambda_i = 1, \lambda_i \geq 0 \text{ for all } i = 1, 2, ..., m \right\}.$$

Minimum Norm Point in Polytope



Minimum Norm Point in Polytope



Some Simple Facts

- MNP is a convex quadratic program

Some Simple Facts

- ▷ MNP is a convex quadratic program
- ▷ can be solved via interior-point methods

Some Simple Facts

▷ MNP is a convex quadratic program

▷ can be solved via interior-point methods

▷ MNP of a polytope given by rational points is rational

Some Simple Facts

- ▷ MNP is a convex quadratic program
- ▷ can be solved via interior-point methods
- ▷ MNP of a polytope given by rational points is rational

`permits combinatorial algorithms`

Applications

- arbitrary polytope projection

Applications

- arbitrary polytope projection
- nearest point problem for transportation polytopes

Applications

- arbitrary polytope projection
- nearest point problem for transportation polytopes
- subroutine in colorful linear programming

Applications

- arbitrary polytope projection
- nearest point problem for transportation polytopes
- subroutine in colorful linear programming
- subroutine in submodular function minimization

Applications

- arbitrary polytope projection
- nearest point problem for transportation polytopes
- subroutine in colorful linear programming
- subroutine in submodular function minimization
- machine learning - vision, large-scale learning

Applications

- arbitrary polytope projection
- nearest point problem for transportation polytopes
- subroutine in colorful linear programming
- subroutine in submodular function minimization
- machine learning - vision, large-scale learning
- compute distance to polytope

Applications

Theorem (De Loera, H., Rademacher '17)
Linear programming reduces to distance to a simplex in vertex-representation in strongly-polynomial time.

Applications

Theorem (De Loera, H., Rademacher '17)
Linear programming reduces to distance to a simplex in vertex-representation in strongly-polynomial time.

If a strongly-polynomial method for projection onto a polytope exists then this gives a strongly-polynomial method for LP.

Applications

Theorem (De Loera, H., Rademacher '17)
Linear programming reduces to distance to a simplex in vertex-representation in strongly-polynomial time.

If a strongly-polynomial method for projection onto a polytope exists then this gives a strongly-polynomial method for LP.

It was previously known that linear programming reduces to MNP on a polytope in weakly-polynomial time [Fujishige, Hayashi, Isotani '06].

Spoiler

Theorem (De Loera, H., Rademacher ’17)

There exists a family of polytopes on which Wolfe’s method requires exponential time to compute the MNP.

Wolfe's Optimality Condition

Theorem (Wolfe '74)

Let $P = conv(\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m)$*. Then* $\mathbf{x} \in P$ *is* $MNP(P)$ *if and only if*

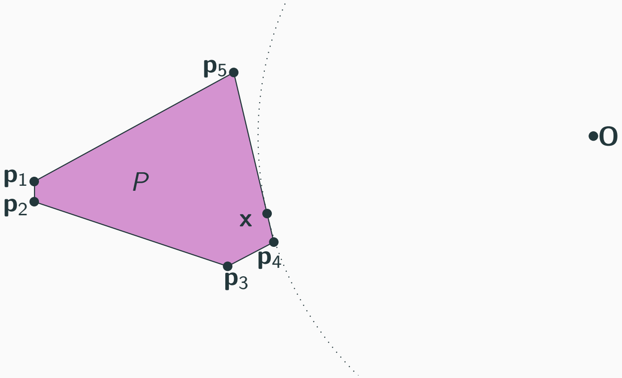
$$\mathbf{x}^T \mathbf{p}_j \geq \|\mathbf{x}\|_2^2 \text{ for all } j = 1, 2, ..., m.$$

Wolfe's Optimality Condition

Theorem (Wolfe '74)

Let $P = conv(\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m)$. *Then* $\mathbf{x} \in P$ *is MNP*(P) *if and only if*

$$\mathbf{x}^T \mathbf{p}_j \geq \|\mathbf{x}\|_2^2 \text{ for all } j = 1, 2, ..., m.$$

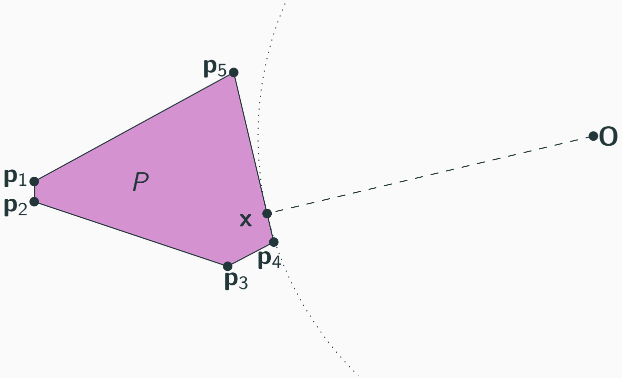


Wolfe's Optimality Condition

Theorem (Wolfe '74)

Let $P = conv(\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m)$. Then $\mathbf{x} \in P$ is $MNP(P)$ if and only if

$$\mathbf{x}^T \mathbf{p}_j \geq \|\mathbf{x}\|_2^2 \text{ for all } j = 1, 2, ..., m.$$

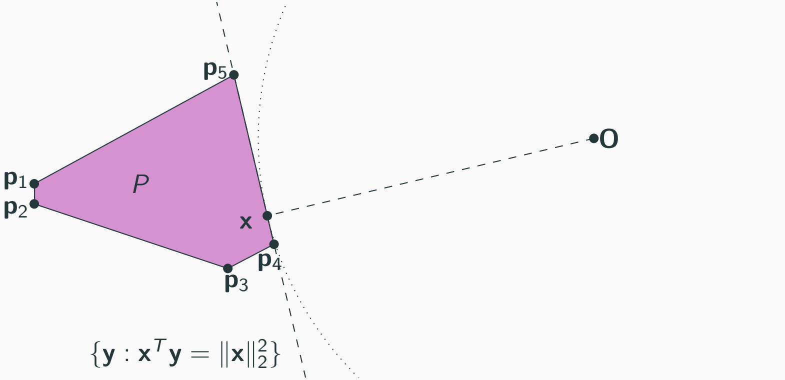


Wolfe's Optimality Condition

Theorem (Wolfe '74)

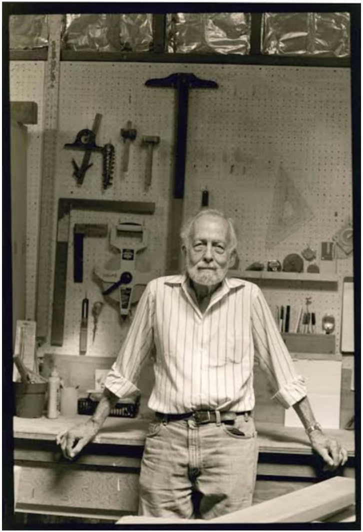
Let $P = conv(\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m)$. *Then* $\mathbf{x} \in P$ *is* $MNP(P)$ *if and only if*

$$\mathbf{x}^T \mathbf{p}_j \geq \|\mathbf{x}\|_2^2 \text{ for all } j = 1, 2, ..., m.$$



Wolfe's Method

Philip Wolfe



- Frank-Wolfe method
- Dantzig-Wolfe decomposition
- simplex method for quadratic programming

Intuition and Definitions

Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

Intuition and Definitions

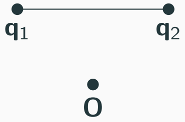
Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

Def: An affinely independent set of points $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ is a *corral* if $\text{MNP}(\text{aff}(Q)) \in \text{relint}(\text{conv}(Q))$.

Intuition and Definitions

Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

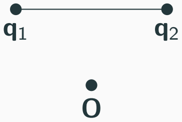
Def: An affinely independent set of points $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ is a *corral* if $\text{MNP}(\text{aff}(Q)) \in \text{relint}(\text{conv}(Q))$.

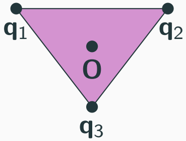


Intuition and Definitions

Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

Def: An affinely independent set of points $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ is a *corral* if $\text{MNP}(\text{aff}(Q)) \in \text{relint}(\text{conv}(Q))$.

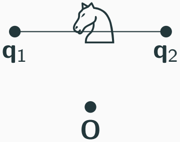


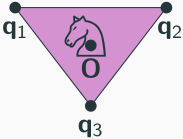


Intuition and Definitions

Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

Def: An affinely independent set of points $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ is a *corral* if $\text{MNP}(\text{aff}(Q)) \in \text{relint}(\text{conv}(Q))$.

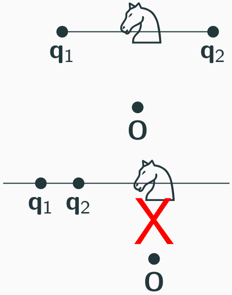


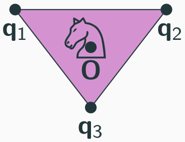


Intuition and Definitions

Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

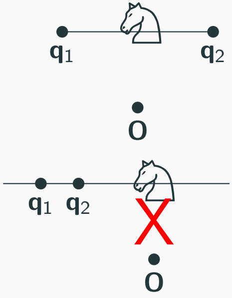
Def: An affinely independent set of points $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ is a *corral* if $\mathrm{MNP}(\mathrm{aff}(Q)) \in \mathrm{relint}(\mathrm{conv}(Q))$.

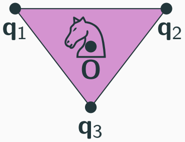




Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

Def: An affinely independent set of points $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ is a *corral* if $\mathrm{MNP}(\mathrm{aff}(Q)) \in \mathrm{relint}(\mathrm{conv}(Q))$.



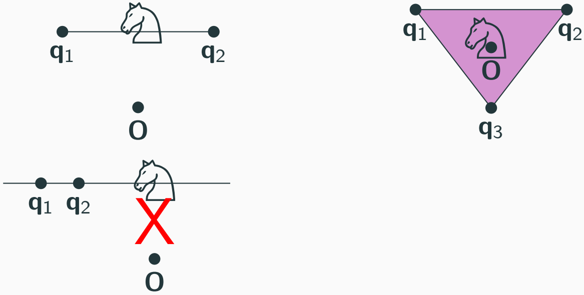


Note: Singletons are corrals.

Intuition and Definitions

Idea: Exploit linear information about the problem in order to progress towards the quadratic solution.

Def: An affinely independent set of points $Q = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k\}$ is a *corral* if $\text{MNP}(\text{aff}(Q)) \in \text{relint}(\text{conv}(Q))$.



Note: Singletons are corrals.

Note: There is a corral in P whose convex hull contains $\text{MNP}(P)$.

Intuition

Wolfe's method : combinatorial method for computing projection onto a vertex-representation polytope

Wolfe's method : combinatorial method for computing projection onto a vertex-representation polytope

- pivots between corrals which may contain $\mathrm{MNP}(P)$

Intuition

Wolfe's method : combinatorial method for computing projection onto a vertex-representation polytope

- pivots between corrals which may contain $\text{MNP}(P)$
- projects onto affine hull of sets to check whether a corral

Intuition

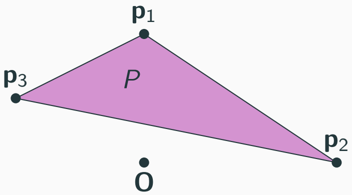
Wolfe's method : combinatorial method for computing projection onto a vertex-representation polytope

- pivots between corrals which may contain MNP(P)
- projects onto affine hull of sets to check whether a corral
- optimality criterion checks if correct corral

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
    $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
    $C = C \cup \{\mathbf{p}_j\}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
        $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
        $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
        $\mathbf{x} = \mathbf{z}$
        $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$



Sketch of Method

$\mathbf{x} \in \mathbf{P} = \{\mathbf{p_1}, \mathbf{p_2}, ..., \mathbf{p_m}\}$

$C = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

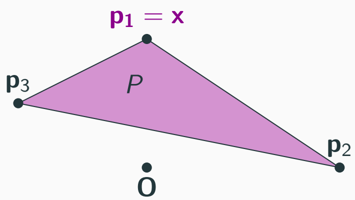
- $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
- $C = C \cup \{\mathbf{p}_j\}$
- $\mathbf{y} = \text{MNP}(\text{aff}(C))$
- while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
  - $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
  - $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
  - $\mathbf{x} = \mathbf{z}$
  - $\mathbf{y} = \text{MNP}(\text{aff}(C))$
- $\mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$



Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$\mathbf{C} = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

$\quad \mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$

$\quad C = C \cup \{\mathbf{p}_j\}$

$\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

$\quad$ while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

$\quad\quad \mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$

$\quad\quad C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

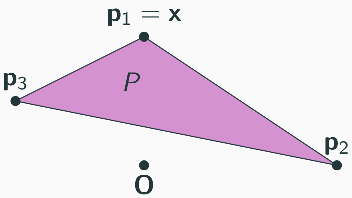
$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$



$\mathbf{C} = \{\mathbf{p}_1\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$C = \{\mathbf{x}\}$

while x is not MNP(P)

$\quad \mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$

$\quad C = C \cup \{\mathbf{p}_j\}$

$\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

$\quad$ while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

$\quad\quad \mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$

$\quad\quad C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

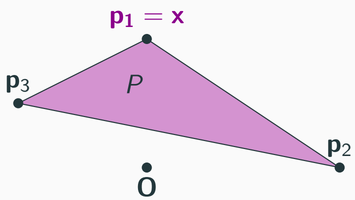
$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$



$C = \{\mathbf{p}_1\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$C = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

$\quad \mathbf{p_j} \in \{\mathbf{p} \in \mathbf{P} : \mathbf{x}^\mathbf{T}\mathbf{p} < ||\mathbf{x}||_2^2\}$

$\quad C = C \cup \{\mathbf{p}_j\}$

$\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

$\quad$ while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

$\quad\quad \mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$

$\quad\quad C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

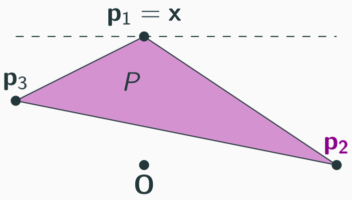
$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$

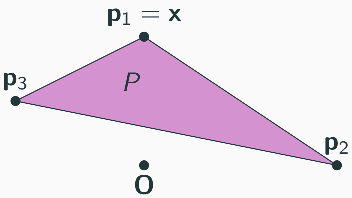


$C = \{\mathbf{p}_1\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $\mathbf{C} = \mathbf{C} \cup \{\mathbf{p_j}\}$
  $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

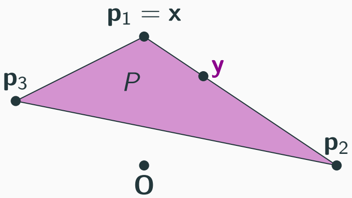


$\mathbf{C} = \{\mathbf{p_1}, \mathbf{p_2}\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $C = C \cup \{\mathbf{p}_j\}$
  $\mathbf{y} = \mathbf{MNP(aff(C))}$
  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

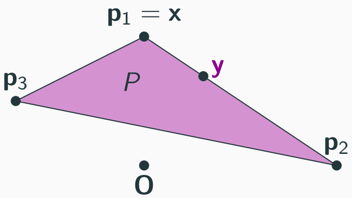


$C = \{\mathbf{p}_1, \mathbf{p}_2\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
    $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
    $C = C \cup \{\mathbf{p}_j\}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    while $\mathbf{y} \notin$ relint(conv(C))
        $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
        $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
        $\mathbf{x} = \mathbf{z}$
        $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

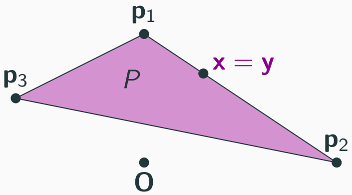


$C = \{\mathbf{p}_1, \mathbf{p}_2\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $C = C \cup \{\mathbf{p}_j\}$
  $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

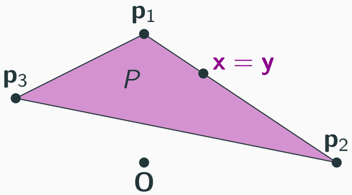


$C = \{\mathbf{p}_1, \mathbf{p}_2\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while x is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $C = C \cup \{\mathbf{p}_j\}$
  $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of $\text{conv}(C)$
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$



$C = \{\mathbf{p}_1, \mathbf{p}_2\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$C = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

  $\mathbf{p_j} \in \{\mathbf{p} \in \mathbf{P} : \mathbf{x}^\mathbf{T}\mathbf{p} < ||\mathbf{x}||_2^2\}$

  $C = C \cup \{\mathbf{p}_j\}$

  $\mathbf{y} = \text{MNP}(\text{aff}(C))$

  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$

    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)

    $\mathbf{x} = \mathbf{z}$

    $\mathbf{y} = \text{MNP}(\text{aff}(C))$

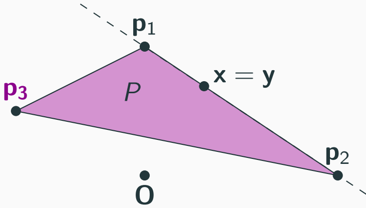
  $\mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$



$C = \{\mathbf{p}_1, \mathbf{p}_2\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$C = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

$\quad \mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$

$\quad \mathbf{C} = \mathbf{C} \cup \{\mathbf{p_j}\}$

$\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

$\quad$ while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

$\quad\quad \mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$

$\quad\quad C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

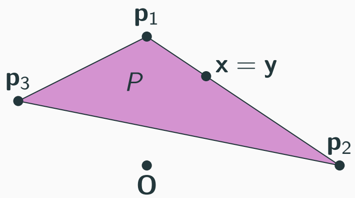
$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$

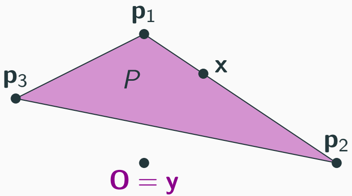


$\mathbf{C} = \{\mathbf{p_1}, \mathbf{p_2}, \mathbf{p_3}\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
    $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
    $C = C \cup \{\mathbf{p}_j\}$
    $\mathbf{y} = \mathbf{MNP(aff(C))}$
    while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
        $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
        $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
        $\mathbf{x} = \mathbf{z}$
        $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

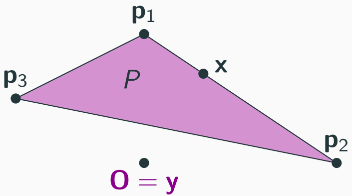


$C = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
    $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
    $C = C \cup \{\mathbf{p}_j\}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    while y $\notin$ relint(conv(C))
        $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
        $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
        $\mathbf{x} = \mathbf{z}$
        $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$



$C = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$C = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

$\quad \mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$

$\quad C = C \cup \{\mathbf{p}_j\}$

$\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

$\quad$ while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

$\qquad \mathbf{z} = \underset{\mathbf{z} \in \mathbf{conv}(\mathbf{C}) \cap \overline{\mathbf{xy}}}{\textbf{argmin}} ||\mathbf{z} - \mathbf{y}||_2$

$\qquad C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)

$\qquad \mathbf{x} = \mathbf{z}$

$\qquad \mathbf{y} = \text{MNP}(\text{aff}(C))$

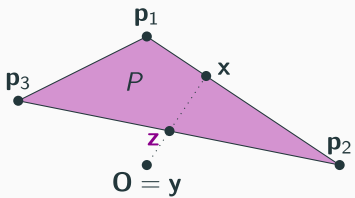
$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$



$C = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$C = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

$\quad \mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$

$\quad C = C \cup \{\mathbf{p}_j\}$

$\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

$\quad$ while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

$\quad\quad \mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$

$\quad\quad$ **$\mathbf{C} = \mathbf{C} - \{\mathbf{p_i}\}$ where $\mathbf{p_i}, \mathbf{z}$ are on different faces of conv($\mathbf{C}$)**

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

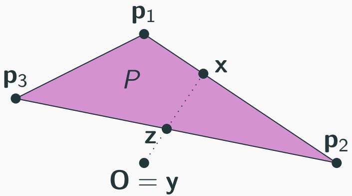
$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$

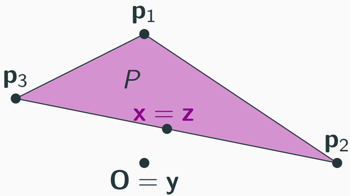


$\mathbf{C} = \{\mathbf{p_2}, \mathbf{p_3}\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $C = C \cup \{\mathbf{p}_j\}$
  $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of $\text{conv}(C)$
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

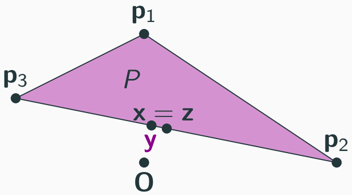


$C = \{\mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $C = C \cup \{\mathbf{p}_j\}$
  $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \textbf{MNP}(\textbf{aff}(\mathbf{C}))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

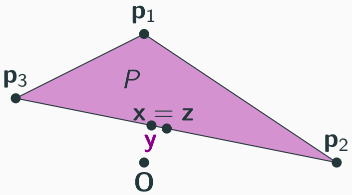


$C = \{\mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $C = C \cup \{\mathbf{p}_j\}$
  $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  while $\mathbf{y} \notin$ relint(conv(C))
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

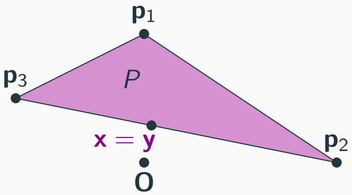


$C = \{\mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while $\mathbf{x}$ is not MNP(P)
  $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
  $C = C \cup \{\mathbf{p}_j\}$
  $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
    $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
    $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)
    $\mathbf{x} = \mathbf{z}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
  $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$

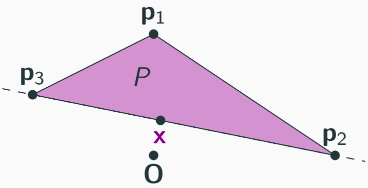


$C = \{\mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$
$C = \{\mathbf{x}\}$
while x is not MNP(P)
    $\mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$
    $C = C \cup \{\mathbf{p}_j\}$
    $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$
        $\mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$
        $C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of $\text{conv}(C)$
        $\mathbf{x} = \mathbf{z}$
        $\mathbf{y} = \text{MNP}(\text{aff}(C))$
    $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

$\mathbf{p}_1 = (0, 2)$
$\mathbf{p}_2 = (3, 0)$
$\mathbf{p}_3 = (-2, 1)$



$C = \{\mathbf{p}_2, \mathbf{p}_3\}$

Sketch of Method

$\mathbf{x} \in P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$

$C = \{\mathbf{x}\}$

while $\mathbf{x}$ is not MNP(P)

$\quad \mathbf{p}_j \in \{\mathbf{p} \in P : \mathbf{x}^T\mathbf{p} < \|\mathbf{x}\|_2^2\}$

$\quad C = C \cup \{\mathbf{p}_j\}$

$\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

$\quad$ while $\mathbf{y} \notin \text{relint}(\text{conv}(C))$

$\quad\quad \mathbf{z} = \underset{\mathbf{z} \in \text{conv}(C) \cap \overline{\mathbf{xy}}}{\text{argmin}} \|\mathbf{z} - \mathbf{y}\|_2$

$\quad\quad C = C - \{\mathbf{p}_i\}$ where $\mathbf{p}_i$, $\mathbf{z}$ are on different faces of conv(C)

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad \mathbf{y} = \text{MNP}(\text{aff}(C))$

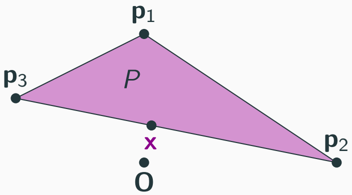
$\quad \mathbf{x} = \mathbf{y}$

return x

$\mathbf{p}_1 = (0, 2)$

$\mathbf{p}_2 = (3, 0)$

$\mathbf{p}_3 = (-2, 1)$



$C = \{\mathbf{p}_2, \mathbf{p}_3\}$

Wolfe's Method

$\mathbf{x} = \mathbf{p}_i$ for some $i = 1, 2, ..., m$, $\lambda = \mathbf{e}_i$

$C = \{i\}$

while $\mathbf{x} \neq \mathbf{0}$ and there exists $\mathbf{p}_j$ with $\mathbf{x}^T \mathbf{p}_j < \|\mathbf{x}\|_2^2$

$\quad C = C \cup \{j\}$

$\quad \alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\operatorname{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$

$\quad$ while $\alpha_i \leq 0$ for some $i = 1, 2, ..., m$

$\quad\quad \theta = \min_{i : \alpha_i \leq 0} \frac{\lambda_i}{\lambda_i - \alpha_i}$

$\quad\quad \mathbf{z} = \theta \mathbf{y} + (1 - \theta) \mathbf{x}$

$\quad\quad i \in \{j : \theta \alpha_j + (1 - \theta) \lambda_j = 0\}$

$\quad\quad C = C - \{i\}$

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad$ solve $\mathbf{x} = P\lambda$ for λ

$\quad\quad \alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\operatorname{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$

$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

Wolfe's Method

$\mathbf{x} = \mathbf{p}_i$ for some $i = 1, 2, ..., m$, $\lambda = \mathbf{e}_i$

$C = \{i\}$

while $\mathbf{x} \neq \mathbf{0}$ and there exists $\mathbf{p}_j$ with $\mathbf{x}^T \mathbf{p}_j < \|\mathbf{x}\|_2^2$

$\quad C = C \cup \{j\}$

$\quad \alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\text{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$

$\quad$ while $\alpha_i \leq 0$ for some $i = 1, 2, ..., m$

$\quad\quad \theta = \min_{i : \alpha_i \leq 0} \frac{\lambda_i}{\lambda_i - \alpha_i}$

$\quad\quad \mathbf{z} = \theta \mathbf{y} + (1 - \theta)\mathbf{x}$

$\quad\quad i \in \{j : \theta \alpha_j + (1 - \theta)\lambda_j = 0\}$

$\quad\quad C = C - \{i\}$

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad$ solve $\mathbf{x} = P\lambda$ for λ

$\quad\quad \alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\text{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$

$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

Choice 1: Initial vertex.

Wolfe's Method

$\mathbf{x} = \mathbf{p}_i$ for some $i = 1, 2, ..., m$, $\lambda = \mathbf{e}_i$
$C = \{i\}$
while $\mathbf{x} \neq \mathbf{0}$ and there exists $\mathbf{p}_j$ with $\mathbf{x}^T\mathbf{p}_j < \|\mathbf{x}\|_2^2$
    $C = C \cup \{j\}$
    $\alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\operatorname{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$
    while $\alpha_i \leq 0$ for some $i = 1, 2, ..., m$
        $\theta = \min_{i : \alpha_i \leq 0} \frac{\lambda_i}{\lambda_i - \alpha_i}$
        $\mathbf{z} = \theta\mathbf{y} + (1 - \theta)\mathbf{x}$
        $i \in \{j : \theta\alpha_j + (1 - \theta)\lambda_j = 0\}$
        $C = C - \{i\}$
        $\mathbf{x} = \mathbf{z}$
        solve $\mathbf{x} = P\lambda$ for λ
        $\alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\operatorname{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$
    $\mathbf{x} = \mathbf{y}$
return $\mathbf{x}$

Choice 1: Initial vertex.

Choice 2: Adding to corral.

Wolfe's Method

$\mathbf{x} = \mathbf{p}_i$ for some $i = 1, 2, ..., m$, $\lambda = \mathbf{e}_i$

$C = \{i\}$

while $\mathbf{x} \neq \mathbf{0}$ and there exists $\mathbf{p}_j$ with $\mathbf{x}^T \mathbf{p}_j < \|\mathbf{x}\|_2^2$

$\quad C = C \cup \{j\}$

$\quad \alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\operatorname{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$

$\quad$ while $\alpha_i \leq 0$ for some $i = 1, 2, ..., m$

$\quad\quad \theta = \min_{i : \alpha_i \leq 0} \frac{\lambda_i}{\lambda_i - \alpha_i}$

$\quad\quad \mathbf{z} = \theta \mathbf{y} + (1 - \theta) \mathbf{x}$

$\quad\quad i \in \{j : \theta \alpha_j + (1 - \theta) \lambda_j = 0\}$

$\quad\quad C = C - \{i\}$

$\quad\quad \mathbf{x} = \mathbf{z}$

$\quad\quad$ solve $\mathbf{x} = P\lambda$ for λ

$\quad\quad \alpha = \underset{\sum_{i \in C} \alpha_i = 1}{\operatorname{argmin}} \|\sum_{i \in C} \alpha_i \mathbf{p}_i\|_2$, $\mathbf{y} = \sum_{i \in C} \alpha_i \mathbf{p}_i$

$\quad \mathbf{x} = \mathbf{y}$

return $\mathbf{x}$

Choice 1: Initial vertex.

Choice 2: Adding to corral.

Choice 3: Removing from corral.

Rules

Initial: `minnorm`

Insertion: `linopt` (select $\mathbf{p}_j$ minimizing $\mathbf{x}^T\mathbf{p}_j$), `minnorm`

Rules

Initial: `minnorm`
Insertion: `linopt` (select $\mathbf{p}_j$ minimizing $\mathbf{x}^T\mathbf{p}_j$), `minnorm`

- insertion rules have different benefits

Rules

Initial: `minnorm`
Insertion: `linopt` (select $\mathbf{p}_j$ minimizing $\mathbf{x}^T\mathbf{p}_j$), `minnorm`

- insertion rules have different benefits
- behavior depends on choice of insertion rule

Rules

Initial: `minnorm`
Insertion: `linopt` (select $\mathbf{p}_j$ minimizing $\mathbf{x}^T\mathbf{p}_j$), `minnorm`

- insertion rules have different benefits
- behavior depends on choice of insertion rule
- examples in which each insertion rule is better

Related Methods

- von Neumann's algorithm for linear programming

Related Methods

- von Neumann's algorithm for linear programming
- Frank-Wolfe method for convex programming (and variants)

Related Methods

- von Neumann's algorithm for linear programming
- Frank-Wolfe method for convex programming (and variants)
- Gilbert's procedure for quadratic programming

Related Methods

- ▷ von Neumann's algorithm for linear programming
- ▷ Frank-Wolfe method for convex programming (and variants)
- ▷ Gilbert's procedure for quadratic programming
  - `projection onto simple convex hull`

Related Methods

- Hanson-Lawson procedure for non-negative least-squares

Related Methods

- Hanson-Lawson procedure for non-negative least-squares
- Betke's combinatorial relaxation algorithm for linear feasibility

Related Methods

- ▷ Hanson-Lawson procedure for non-negative least-squares
- ▷ Betke's combinatorial relaxation algorithm for linear feasibility
  - `combinatorial methods`

Related Methods

- Fujishige-Wolfe method for submodular optimization

Related Methods

- ▷ Fujishige-Wolfe method for submodular optimization
- ▷ Bárány-Onn approximation method for colorful linear programming

Related Methods

- ▷ Fujishige-Wolfe method for submodular optimization
- ▷ Bárány-Onn approximation method for colorful linear programming
  - `combinatorial problems`

Previous Results

- $\#$ iterations $\leq \sum_{i=1}^{n+1} i\binom{m}{i}$ with any rules (Wolfe '74)

Previous Results

- $\#$ iterations $\leq \sum_{i=1}^{n+1} i\binom{m}{i}$ with any rules (Wolfe '74)

- ϵ-approximate solution in $\mathcal{O}(nM^2/\epsilon)$ iterations with `linopt` insertion rule (Chakrabarty, Jain, Kothari '14)

Previous Results

- $\#$ iterations $\leq \sum_{i=1}^{n+1} i\binom{m}{i}$ with any rules (Wolfe '74)

- ϵ-approximate solution in $\mathcal{O}(nM^2/\epsilon)$ iterations with `linopt` insertion rule (Chakrabarty, Jain, Kothari '14)
- ϵ-approximate solution in $\mathcal{O}(\rho \log(1/\epsilon))$ iterations with `linopt` insertion rule (Lacoste-Julien, Jaggi '15)

Previous Results

- $\#$ iterations $\leq \sum_{i=1}^{n+1} i\binom{m}{i}$ with any rules (Wolfe '74)

- ϵ-approximate solution in $\mathcal{O}(nM^2/\epsilon)$ iterations with `linopt` insertion rule (Chakrabarty, Jain, Kothari '14)
- ϵ-approximate solution in $\mathcal{O}(\rho \log(1/\epsilon))$ iterations with `linopt` insertion rule (Lacoste-Julien, Jaggi '15)

  ▷ pseudo-polynomial complexity

Exponential Behavior

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

- dimension and number of points grow linearly

Exponential Example

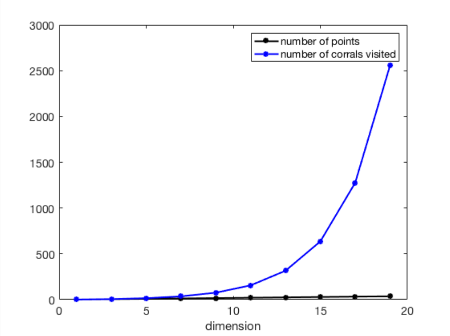
Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

- dimension and number of points grow linearly
- number of corrals visited grows exponentially

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

- dimension and number of points grow linearly
- number of corrals visited grows exponentially



Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

Recursively Defined Instances

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

Recursively Defined Instances

dim: $d-2$
Instance: $P(d-2)$
Points: $2d-5$

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

Recursively Defined Instances

dim: $d-2$
Instance: $P(d-2)$
Points: $2d-5$

$$\xrightarrow[+4 \text{ points}]{+2 \text{ dim}}$$

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

Recursively Defined Instances

dim: $d-2$	$+2$ dim	**dim**: d
Instance: $P(d-2)$	$\longrightarrow$	**Instance**: $P(d)$
Points: $2d-5$	$+4$ points	**Points**: $2d-1$

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

Recursively Defined Instances

dim: $d-2$	$+2$ dim	**dim**: d
Instance: $P(d-2)$	$\longrightarrow$	**Instance**: $P(d)$
Points: $2d-5$	$+4$ points	**Points**: $2d-1$

$P(1) := \{1\}$

Exponential Example

Goal : build family of instances on which the number of iterations of Wolfe's method is at least exponential in the dimension and number of points

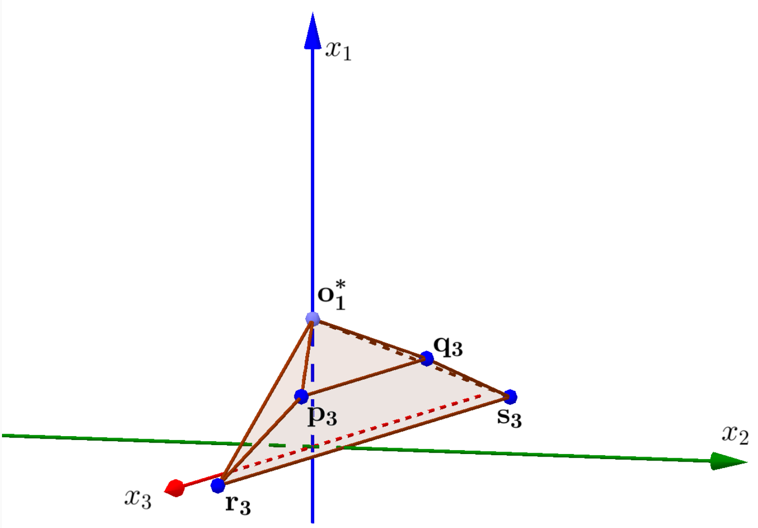
Recursively Defined Instances

dim: $d-2$	$+2$ dim	**dim**: d
Instance: $P(d-2)$	$\longrightarrow$	**Instance**: $P(d)$
Points: $2d-5$	$+4$ points	**Points**: $2d-1$

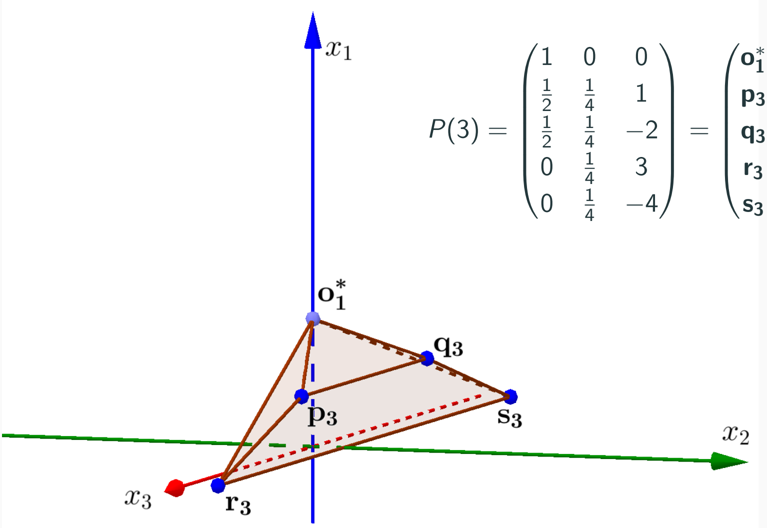
$P(1) := \{1\}$

$P(3) := \{(1,0,0), \mathbf{p}_3, \mathbf{q}_3, \mathbf{r}_3, \mathbf{s}_3\}$

Exponential Example: dim 3

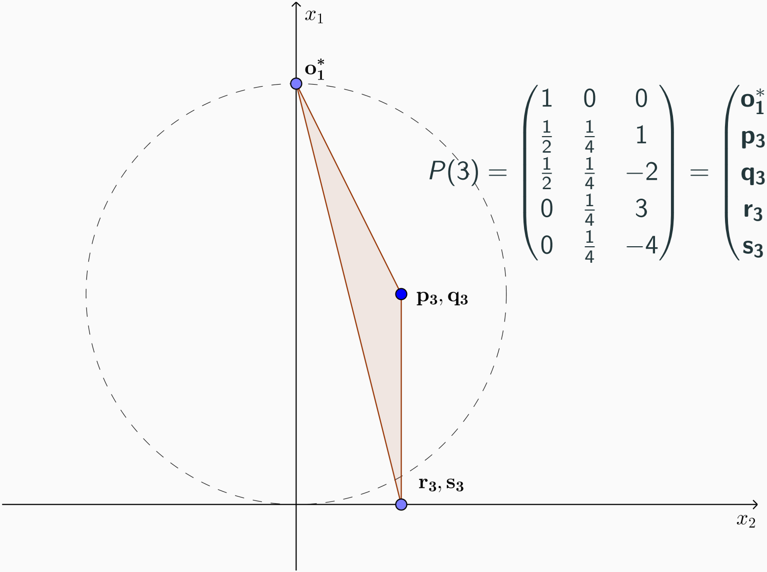


Exponential Example: dim 3



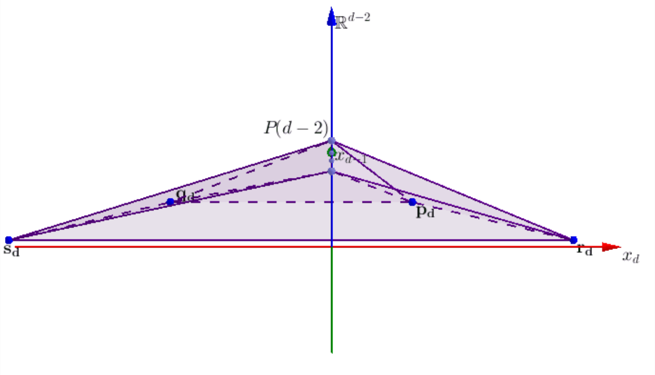
$$P(3) = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & 1 \\ \frac{1}{2} & \frac{1}{4} & -2 \\ 0 & \frac{1}{4} & 3 \\ 0 & \frac{1}{4} & -4 \end{pmatrix} = \begin{pmatrix} \mathbf{o}_1^* \\ \mathbf{p}_3 \\ \mathbf{q}_3 \\ \mathbf{r}_3 \\ \mathbf{s}_3 \end{pmatrix}$$

Exponential Example: dim 3



$$P(3) = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & 1 \\ \frac{1}{2} & \frac{1}{4} & -2 \\ 0 & \frac{1}{4} & 3 \\ 0 & \frac{1}{4} & -4 \end{pmatrix} = \begin{pmatrix} \mathbf{o}_1^* \\ \mathbf{p}_3 \\ \mathbf{q}_3 \\ \mathbf{r}_3 \\ \mathbf{s}_3 \end{pmatrix}$$

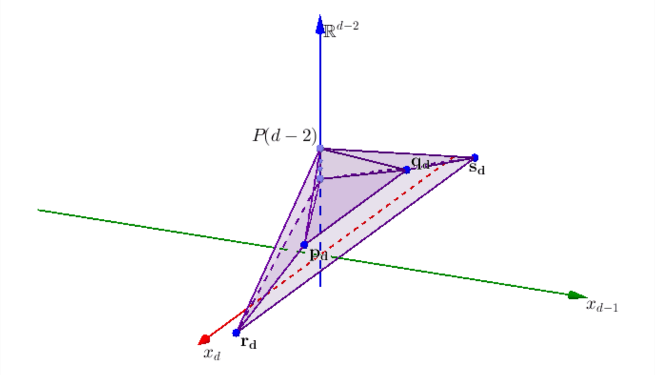
Exponential Example



$$P(d) = \begin{pmatrix} P(d-2) & 0 & 0 \\ \frac{1}{2}\mathbf{o}^*_{\mathbf{d-2}} & \frac{m_{d-2}}{4} & M_{d-2} \\ \frac{1}{2}\mathbf{o}^*_{\mathbf{d-2}} & \frac{m_{d-2}}{4} & -(M_{d-2}+1) \\ 0 & \frac{m_{d-2}}{4} & M_{d-2}+2 \\ 0 & \frac{m_{d-2}}{4} & -(M_{d-2}+3) \end{pmatrix}$$

$\mathbf{o}^*_{\mathbf{d-2}}$: MNP($P(d-2)$)
$m_{d-2} \leq \|\mathbf{o}^*_{\mathbf{d-2}}\|_\infty$
$M_{d-2} \geq \max_{\mathbf{p} \in P(d-2)} \|\mathbf{p}\|_1$

Exponential Example



$$P(d) = \begin{pmatrix} P(d-2) & 0 & 0 \\ \frac{1}{2}\mathbf{o^*_{d-2}} & \frac{m_{d-2}}{4} & M_{d-2} \\ \frac{1}{2}\mathbf{o^*_{d-2}} & \frac{m_{d-2}}{4} & -(M_{d-2}+1) \\ 0 & \frac{m_{d-2}}{4} & M_{d-2}+2 \\ 0 & \frac{m_{d-2}}{4} & -(M_{d-2}+3) \end{pmatrix} \Big\downarrow \|\cdot\|$$

$\mathbf{o^*_{d-2}}$: MNP$(P(d-2))$

$m_{d-2} \leq \|\mathbf{o^*_{d-2}}\|_\infty$

$M_{d-2} \geq \max_{\mathbf{p}\in P(d-2)} \|\mathbf{p}\|_1$

Exponential Example

Theorem (De Loera, H., Rademacher '17)

Consider the execution of Wolfe's method with the $\mathtt{minnorm}$ *insertion rule on input* $P(d)$ *where* $d = 2k - 1$*. Then the sequence of corrals,* $C(d)$ *has length* $5 \cdot 2^{k-1} - 4$*.*

Exponential Example

Theorem (De Loera, H., Rademacher '17)

Consider the execution of Wolfe's method with the $\mathtt{minnorm}$ *insertion rule on input* $P(d)$ *where* $d = 2k - 1$*. Then the sequence of corrals,* $C(d)$ *has length* $5 \cdot 2^{k-1} - 4$.

Key Lemma: Sequence of Corrals

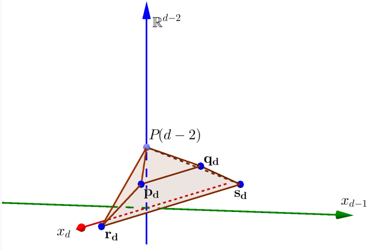
Exponential Example

Theorem (De Loera, H., Rademacher '17)

Consider the execution of Wolfe's method with the `minnorm` *insertion rule on input* $P(d)$ *where* $d = 2k - 1$*. Then the sequence of corrals,* $C(d)$ *has length* $5 \cdot 2^{k-1} - 4$*.*

Key Lemma: Sequence of Corrals

$C(d-2)$



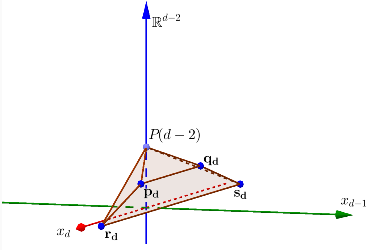
Exponential Example

Theorem (De Loera, H., Rademacher '17)

Consider the execution of Wolfe's method with the `minnorm` *insertion rule on input $P(d)$ where $d = 2k - 1$. Then the sequence of corrals, $C(d)$ has length $5 \cdot 2^{k-1} - 4$.*

Key Lemma: Sequence of Corrals

$C(d-2) \longrightarrow$



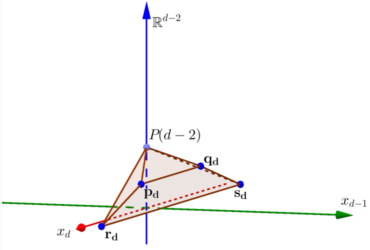
Exponential Example

Theorem (De Loera, H., Rademacher '17)

Consider the execution of Wolfe's method with the `minnorm` *insertion rule on input $P(d)$ where $d = 2k - 1$. Then the sequence of corrals, $C(d)$ has length $5 \cdot 2^{k-1} - 4$.*

Key Lemma: Sequence of Corrals

$$C(d-2) \longrightarrow \begin{array}{l} C(d-2) \\ O(d-2)\mathbf{p_d} \\ \mathbf{p_d q_d} \\ \mathbf{q_d r_d} \\ \mathbf{r_d s_d} \\ C(d-2)\mathbf{r_d s_d} \end{array}$$



Exponential Example

Theorem (De Loera, H., Rademacher '17)

Consider the execution of Wolfe's method with the $\mathtt{minnorm}$ *insertion rule on input* $P(d)$ *where* $d = 2k - 1$*. Then the sequence of corrals,* $C(d)$ *has length* $5 \cdot 2^{k-1} - 4$*.*

Sequence of Corrals: dim 1 $\rightarrow$ dim 3

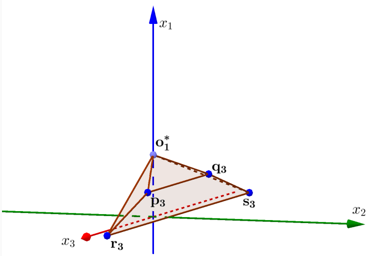
Exponential Example

Theorem (De Loera, H., Rademacher '17)

Consider the execution of Wolfe's method with the `minnorm` *insertion rule on input $P(d)$ where $d = 2k - 1$. Then the sequence of corrals, $C(d)$ has length $5 \cdot 2^{k-1} - 4$.*

Sequence of Corrals: dim 1 → dim 3

1



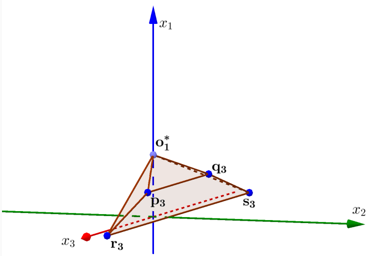
Exponential Example

Theorem (De Loera, H., Rademacher ’17)

Consider the execution of Wolfe’s method with the `minnorm` *insertion rule on input $P(d)$ where $d = 2k - 1$. Then the sequence of corrals, $C(d)$ has length $5 \cdot 2^{k-1} - 4$.*

Sequence of Corrals: dim 1 → dim 3

$1 \longrightarrow$



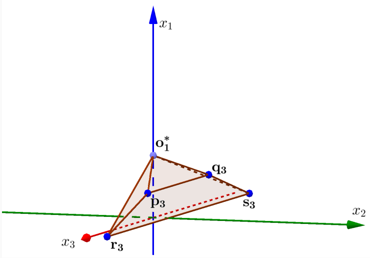
Exponential Example

Theorem (De Loera, H., Rademacher '17)

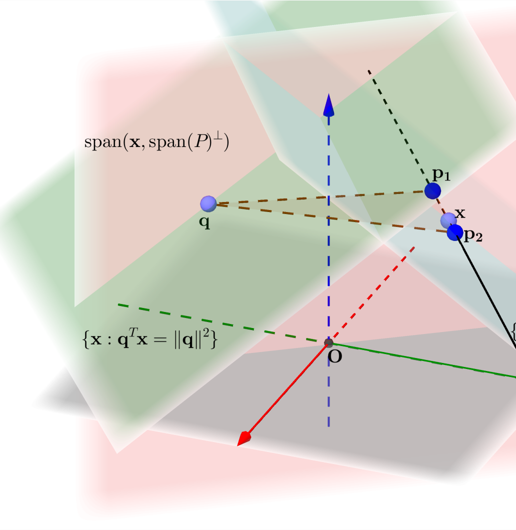
Consider the execution of Wolfe's method with the $\mathtt{minnorm}$ *insertion rule on input $P(d)$ where $d = 2k - 1$. Then the sequence of corrals, $C(d)$ has length $5 \cdot 2^{k-1} - 4$.*

Sequence of Corrals: dim 1 $\rightarrow$ dim 3

$$1 \longrightarrow \begin{array}{l} (1,0,0) \\ (1,0,0)\mathbf{p_3} \\ \mathbf{p_3 q_3} \\ \mathbf{q_3 r_3} \\ \mathbf{r_3 s_3} \\ (1,0,0)\mathbf{r_3 s_3} \end{array}$$

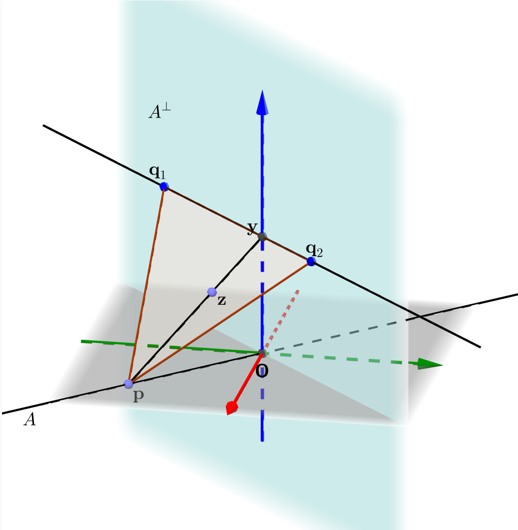


- a corral with a point made from MNP and orthogonal directions is still a corral



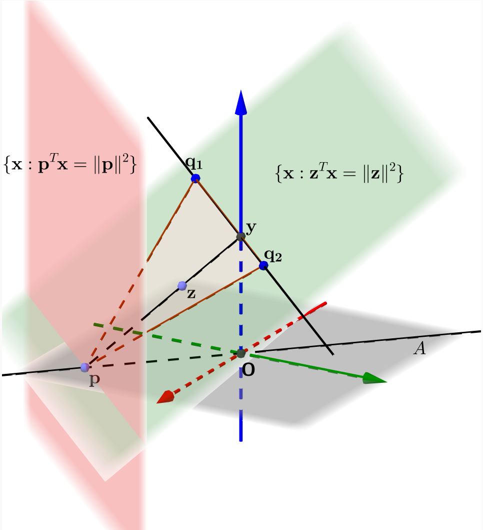
Three Lemmas

- a corral with a point made from MNP and orthogonal directions is still a corral
- the union of orthogonal corrals is still a corral

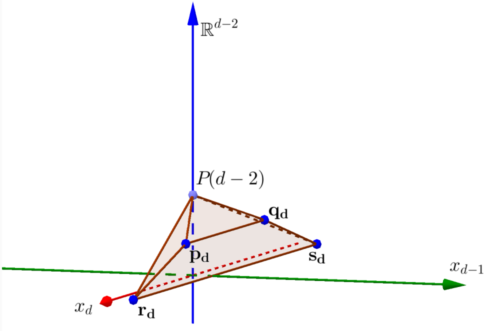


Three Lemmas

- a corral with a point made from MNP and orthogonal directions is still a corral
- the union of orthogonal corrals is still a corral
- adding orthogonal points to the corral doesn't create any available points



Sketch of Proof of Sequence $C(d)$: $C(d-2)$



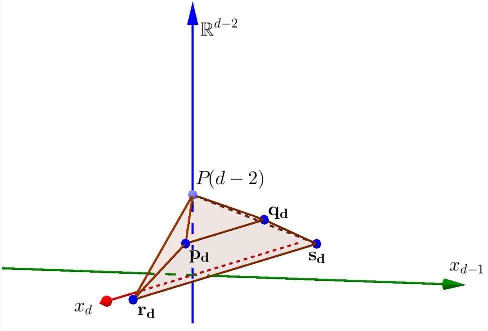
$$P(d) = \begin{pmatrix} P(d-2) & 0 & 0 \\ \frac{1}{2}\mathbf{o}^*_{\mathbf{d-2}} & \frac{m_{d-2}}{4} & M_{d-2} \\ \frac{1}{2}\mathbf{o}^*_{\mathbf{d-2}} & \frac{m_{d-2}}{4} & -(M_{d-2}+1) \\ 0 & \frac{m_{d-2}}{4} & M_{d-2}+2 \\ 0 & \frac{m_{d-2}}{4} & -(M_{d-2}+3) \end{pmatrix}$$

$\mathbf{o}^*_{\mathbf{d-2}}$: MNP($P(d-2)$)

$m_{d-2} \leq \|\mathbf{o}^*_{\mathbf{d-2}}\|_\infty$

$M_{d-2} \geq \max_{\mathbf{p} \in P(d-2)} \|\mathbf{p}\|_1$

Sketch of Proof of Sequence $C(d)$: $C(d-2)$



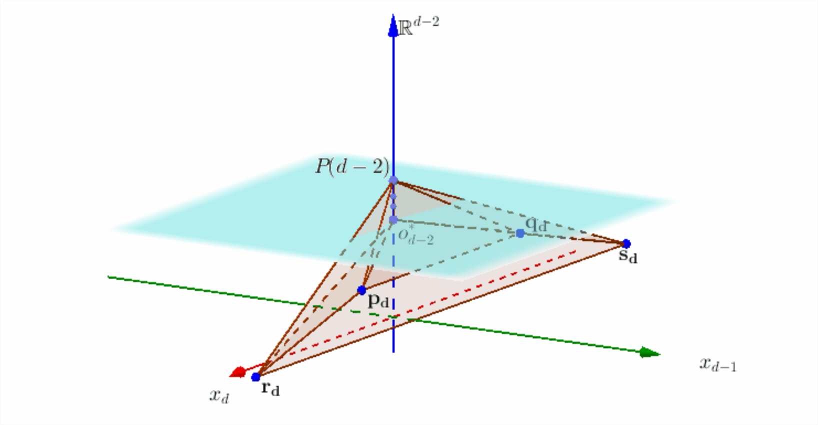
$$P(d) = \begin{pmatrix} P(d-2) & 0 & 0 \\ \frac{1}{2}\mathbf{o^*_{d-2}} & \frac{m_{d-2}}{4} & M_{d-2} \\ \frac{1}{2}\mathbf{o^*_{d-2}} & \frac{m_{d-2}}{4} & -(M_{d-2}+1) \\ 0 & \frac{m_{d-2}}{4} & M_{d-2}+2 \\ 0 & \frac{m_{d-2}}{4} & -(M_{d-2}+3) \end{pmatrix} \Big\downarrow \|\cdot\|$$

$\mathbf{o^*_{d-2}}$: MNP$(P(d-2))$

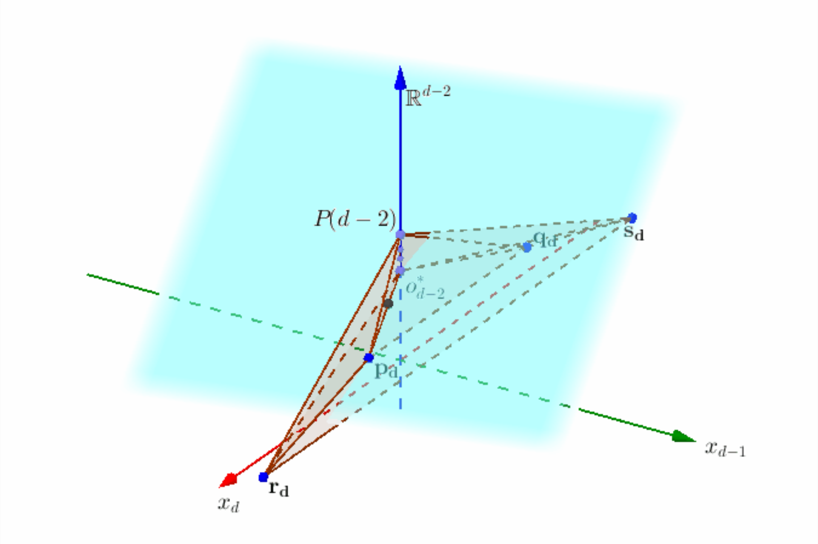
$m_{d-2} \leq \|\mathbf{o^*_{d-2}}\|_\infty$

$M_{d-2} \geq \max_{\mathbf{p}\in P(d-2)} \|\mathbf{p}\|_1$

Sketch of Proof of Sequence $C(d)$: $O(d-2)\mathbf{p_d}$

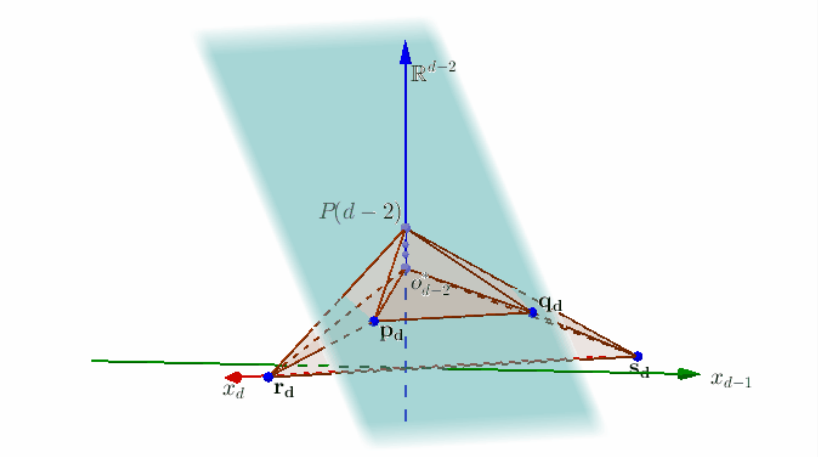


Sketch of Proof of Sequence $C(d)$: $O(d-2)\mathbf{p_d}$

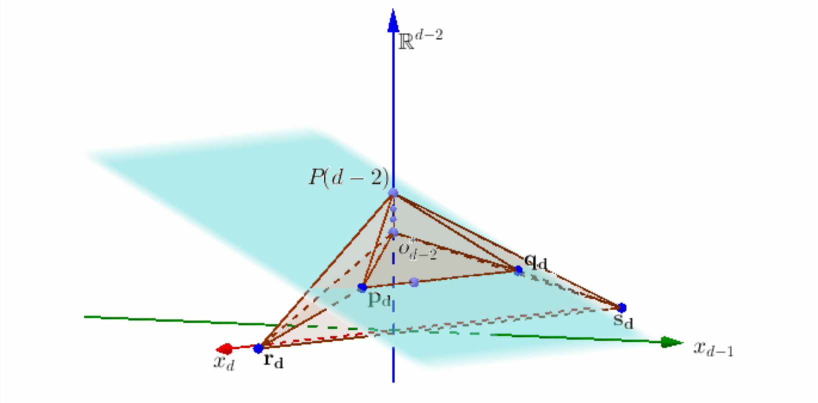


a corral with a point made from MNP and orthogonal directions is still a corral

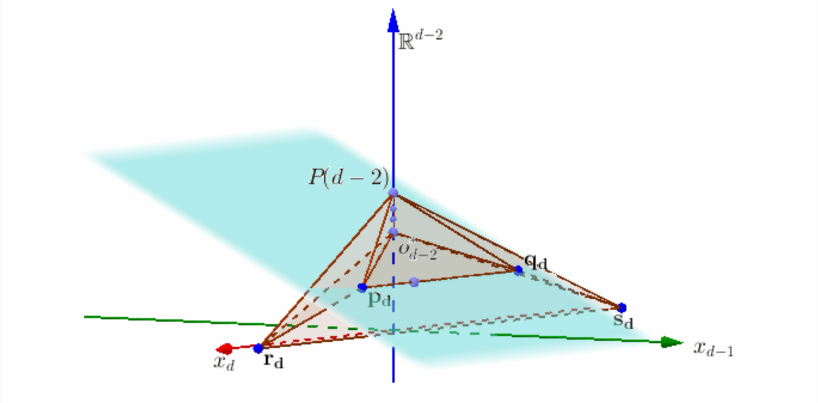
Sketch of Proof of Sequence $C(d)$: $\mathbf{p_d q_d}$



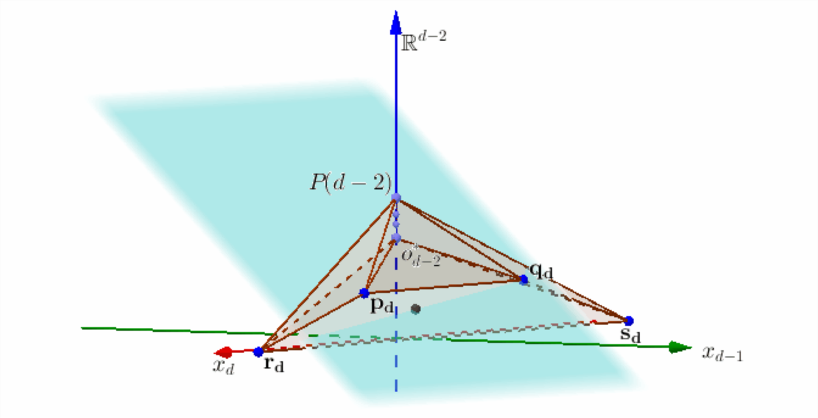
Sketch of Proof of Sequence $C(d)$: $\mathbf{p_d q_d}$



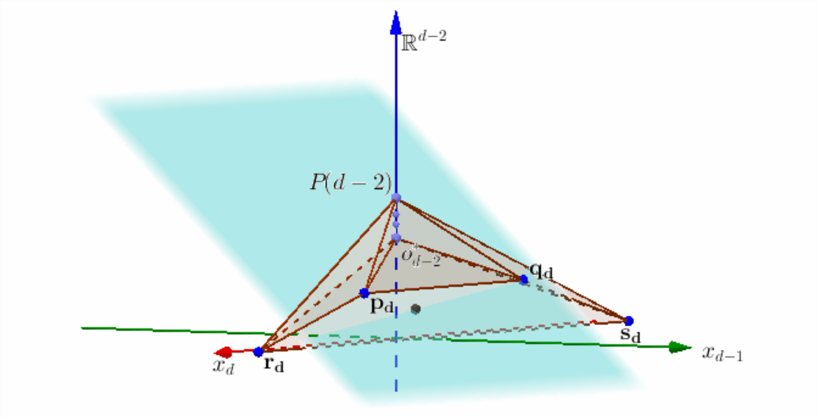
Sketch of Proof of Sequence $C(d)$: $\mathbf{q_d r_d}$



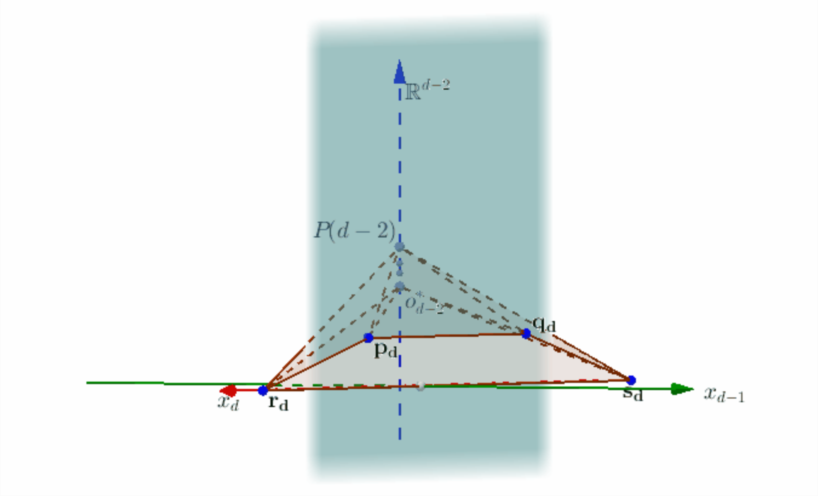
Sketch of Proof of Sequence $C(d)$: $\mathbf{q_d r_d}$



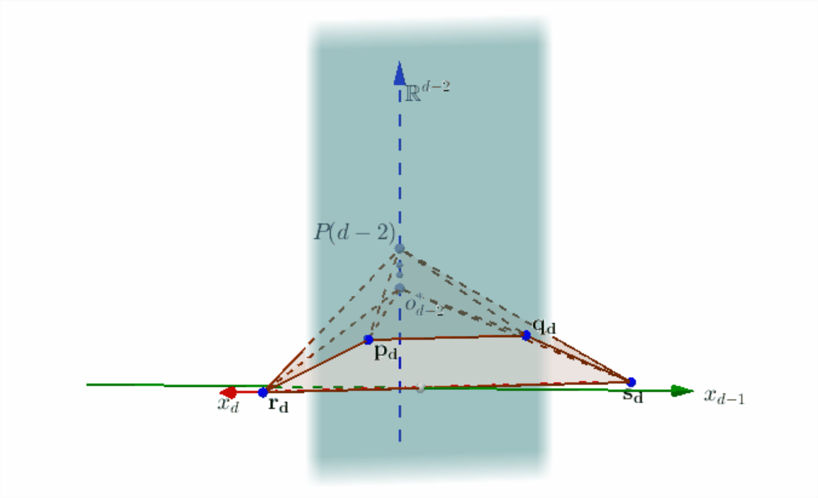
Sketch of Proof of Sequence $C(d)$: $\mathbf{r_d s_d}$



Sketch of Proof of Sequence $C(d)$: $\mathbf{r_d s_d}$



Sketch of Proof of Sequence $C(d)$: $C(d-2)\mathbf{r_d}\mathbf{s_d}$



- the union of orthogonal corrals is still a corral
- adding orthogonal points to the corral doesn't create any available points

Conclusions

Future Directions

Future Directions

1. Find an exponential example for Wolfe's method with `linopt` insertion rule.

Future Directions

1. Find an exponential example for Wolfe's method with `linopt` insertion rule.
2. Search for types of polytopes where Wolfe's method is polynomial (e.g. base polytopes).

Future Directions

1. Find an exponential example for Wolfe's method with `linopt` insertion rule.
2. Search for types of polytopes where Wolfe's method is polynomial (e.g. base polytopes).
3. Understand the structure of polytopes formed by reduction of linear programs.

Future Directions

1. Find an exponential example for Wolfe's method with `linopt` insertion rule.
2. Search for types of polytopes where Wolfe's method is polynomial (e.g. base polytopes).
3. Understand the structure of polytopes formed by reduction of linear programs.
4. Give an average (or smoothed) analysis of Wolfe's method.

Questions?

[1] I. Bárány and S. Onn.
Colourful linear programming and its relatives.
Mathematics of Operations Research, 22(3):550–567, 1997.

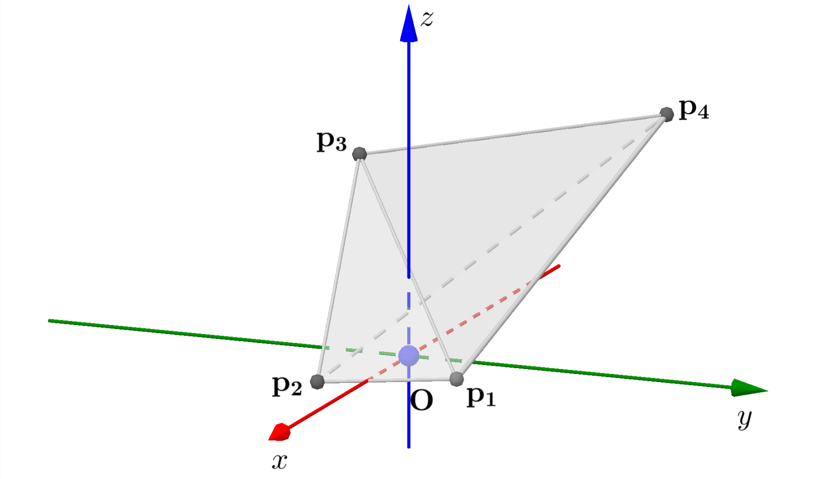
[2] D. Chakrabarty, P. Jain, and P. Kothari.
Provable submodular minimization using wolfe's algorithm.
CoRR, abs/1411.0095, 2014.

[3] J. A. De Loera, J. Haddock, and L. Rademacher.
The minimum Euclidean-norm point on a convex polytope: Wolfes combinatorial algorithm is exponential.
2017.

[4] S. Fujishige, T. Hayashi, and S. Isotani.
The minimum-norm-point algorithm applied to submodular function minimization and linear programming.
Citeseer, 2006.

Example: `minnorm` < `linopt`

$P = \mathrm{conv}\{(0.8, 0.9, 0), (1.5, -0.5, 0), (-1, -1, 2), (-4, 1.5, 2)\} \subset \mathbb{R}^3$



Example: `minnorm` $<$ `linopt`

Major Cycle	Minor Cycle	C
0	0	$\{p_1\}$
1	0	$\{p_1, p_2\}$
2	0	$\{p_1, p_2, p_3\}$
3	0	$\{p_1, p_2, p_3, p_4\}$
3	1	$\{p_1, p_2, p_4\}$

Major Cycle	Minor Cycle	C
0	0	$\{p_1\}$
1	0	$\{p_1, p_4\}$
2	0	$\{p_1, p_4, p_3\}$
2	1	$\{p_1, p_3\}$
3	0	$\{p_1, p_3, p_2\}$
4	0	$\{p_1, p_2, p_3, p_4\}$
4	1	$\{p_1, p_2, p_4\}$

Example: minnorm < linopt

Major Cycle	Minor Cycle	C
0	0	$\{p_1\}$
1	0	$\{p_1, p_2\}$
2	0	$\{p_1, p_2, p_3\}$
3	0	$\{p_1, p_2, p_3, p_4\}$
3	1	$\{p_1, p_2, p_4\}$

minnorm < linopt

Major Cycle	Minor Cycle	C
0	0	$\{p_1\}$
1	0	$\{p_1, p_4\}$
2	0	$\{p_1, p_4, p_3\}$
2	1	$\{p_1, p_3\}$
3	0	$\{p_1, p_3, p_2\}$
4	0	$\{p_1, p_2, p_3, p_4\}$
4	1	$\{p_1, p_2, p_4\}$