An Introduction to Intersection Cuts and Their Applications

Akang Wang

Dept. of Chemical Engineering
Carnegie Mellon University

PSE Seminar
November 30, 2018

Outline

- Intersection Cuts
  - Problem Definition
  - Derivation
  - Geometric Interpretation
- Applications
  - Mixed Integer Linear Programming
  - Reverse Convex Programming
  - Polynomial Programming
- Comments

Problem Definition

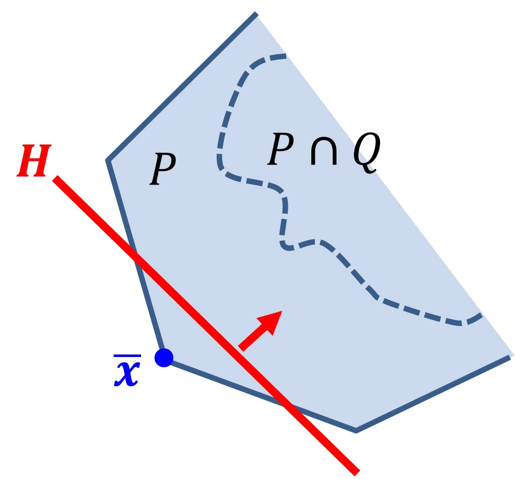
- Optimization problem:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & x \in P \cap Q \end{aligned}$$

$P := \{x \in \mathbb{R}^n, Ax \leq b, x \geq 0\}$ is a polyhedral set, where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^n$
$Q \subseteq \mathbb{R}^n$ represents a non-convex, "complicated" set, such as integrality, reverse convex, etc.

- A polyhedral relaxation:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$



Assume the LP optimality is achieved at $\bar{x}$

Q: How to generate a valid cut ***H*** such that $P \cap Q \subseteq$ ***H*** and $\bar{x} \notin$ ***H*** ?

Standard Form of an LP

- Introduce **slack** variables ***s*** and let $t \coloneqq (x, s)$ represent the variables in LP for convenience

$$\begin{aligned} \underset{x,s}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & Ax + s = b \\ & x, s \geq 0 \end{aligned} \longrightarrow \begin{aligned} \underset{t}{\text{minimize}} \quad & \tilde{c}^T t \\ \text{subject to} \quad & \tilde{A} t = b \\ & t \geq 0 \end{aligned}$$

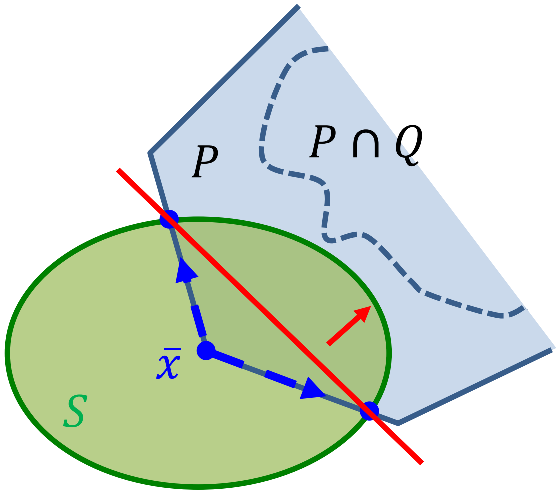
- Notation
  - N index set of **structural** variables $\boldsymbol{x}$, $|N| = n$
  - I index set of **basic** variables, $|I| = m$
  - J index set of **non-basic** variables, $|J| = n$

Carnegie Mellon University

Intersection Cuts

✓ A **convex** set S contains $\bar{x}$ but **no any feasible point** within its **interior**

- S is a convex set
- $\bar{x} \in int(S)$
- $int(S) \cap (P \cap Q) = \emptyset$



✓ Follow the **extreme rays** at $\bar{x}$ and find the **intersection points**

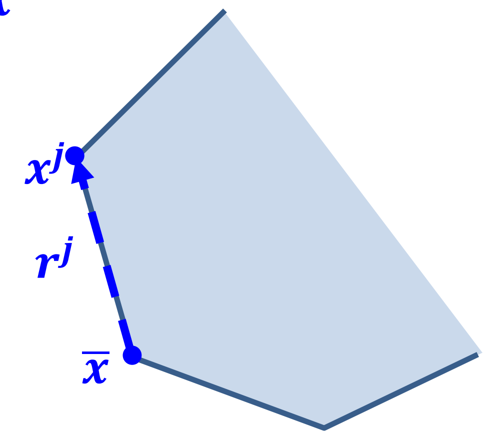
✓ Obtain the **intersection cut** that goes through all intersection points

Balas, E., 1971. Intersection cuts—a new type of cutting planes for integer programming. *Operations Research*, *19*(1), pp.19-39.

Extreme Rays

- Find its neighboring extreme point $\boldsymbol{x^j}$, then $\boldsymbol{r^j} := \boldsymbol{x^j} - \boldsymbol{\bar{x}}$

- Move from one extreme point $\boldsymbol{\bar{x}}$ to its neighboring extreme point $\boldsymbol{x^j}$ when a **non-basic** variable **enters** the basis and a **basic** variable **leaves** the basis



- Simple tableau

$$\begin{array}{cc} \text{Basic} & \text{Non-basic} \\ \multicolumn{2}{c}{\begin{bmatrix} 1 & & & & & * & & * \\ & 1 & & & & & & \\ & & 1 & & & * & \bar{a}_{ij} & * \\ & & & \vdots & & & & \\ & & & & 1 & * & & * \end{bmatrix} \begin{bmatrix} t_I \\ t_J \end{bmatrix} = \begin{bmatrix} \bar{t}_I \\ \bar{t}_J \end{bmatrix}} \end{array}$$

Focus on structural variables x rows

$$x_i = \boxed{\bar{x}_i} - \sum_{j \in J} \bar{a}_{ij} t_j \quad \forall i \in I \cap N \quad \textbf{Basic}$$

$$x_i = \boxed{0} \quad \forall i \in J \cap N \quad \textbf{Non-basic}$$

$\boldsymbol{\bar{x}}$

Extreme Rays

- Choose a **non-basic** variable (structural or slack) t_j for some $j \in J$ and let t_j **enter the basis** (assume non-degeneracy)

Other non-basic variables will stay unchanged (still at 0)

$$\begin{aligned} x_i &= \boxed{\bar{x}_i} - \sum_{j \in J} \bar{a}_{ij} t_j && \forall i \in I \cap N \\ x_i &= \boxed{0} && \forall i \in J \cap N \end{aligned} \qquad \bar{x}$$

$$\xrightarrow{\text{pivot}}$$

$$\begin{aligned} x_i &= \bar{x}_i - \bar{a}_{ij}\xi && \forall i \in I \cap N \\ x_i &= 0 && \forall i \in J \cap N \setminus \{j\} \\ x_j &= \xi && \text{if } j \in N \end{aligned} \qquad x^j$$

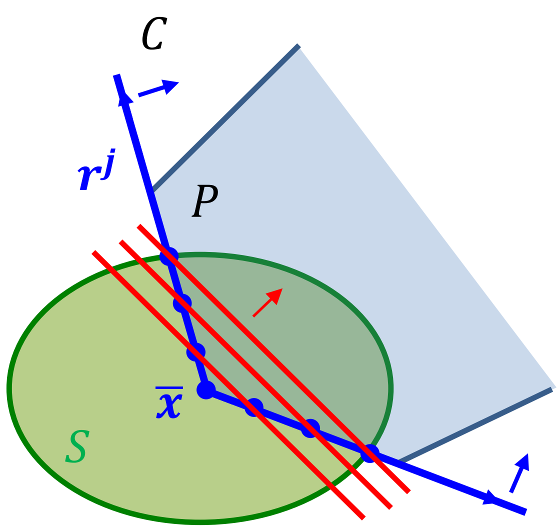
No need to track slack variables s rows

- An extreme ray $r^j = x^j - \bar{x}$

$$\begin{aligned} r_i^j &= -\bar{a}_{ij}\xi && \forall i \in I \cap N \\ r_i^j &= 0 && \forall i \in J \cap N \setminus \{j\} \\ r_j^j &= \xi && \text{if } j \in N \end{aligned} \quad \xrightarrow{\xi > 0} \quad \begin{aligned} r_i^j &= -\bar{a}_{ij} \\ r_i^j &= 0 \\ r_j^j &= 1 \end{aligned}$$

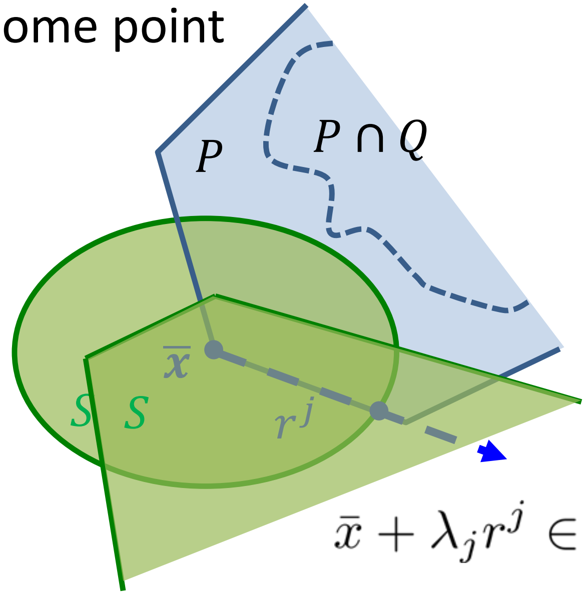
Simplicial Conic Relaxation

- # of extreme rays = # of non-basic variables = $|J|$ = $\boldsymbol{n}$
- These extreme rays are **linearly independent**
- Define a set $C \coloneqq \{x | x = \boldsymbol{\overline{x}} + \sum_{j\in J} \lambda_j \boldsymbol{r^j}, \lambda_j \geq 0 \ \forall j \in J\}$, then $\boldsymbol{P \subseteq C}$



Intersection Points

- The convex set S is intersected by a halfline $\boldsymbol{\eta}^{\boldsymbol{j}} = \bar{\boldsymbol{x}} + \lambda_j \boldsymbol{r}^{\boldsymbol{j}}$, where $\lambda_j \geq 0$ at some point



$$\max_{\lambda_j \geq 0} \quad \lambda_j \qquad (*)$$
$$\text{subject to} \quad \bar{x} + \lambda_j r^j \in S$$

$$\bar{x} + \lambda_j r^j \in S \qquad \forall \lambda_j \geq 0$$

- This problem $(*)$ can be solved in **polynomial time** (e.g. line search) and two cases will arise:
  - $(*)$ has a **unique solution** $\bar{\lambda}_j > 0$ $\qquad J_1$
  - The obj. is **unbounded** ($r^j \in \text{Rec}(S)$, set $\bar{\lambda}_j = +\infty$) $\qquad J_2$

Intersection Cuts

- The intersection cut $\boldsymbol{\beta}^{\boldsymbol{T}}\boldsymbol{x} \leq \boldsymbol{\beta}_{\mathbf{0}}$ is the halfspace whose boundary contains each **intersection point** ($j \in J_1$) and that is parallel to each **extreme ray** ($j \in J_2$) in Rec(S)

$$\begin{aligned} \beta^T(\bar{x} + \bar{\lambda}_j r^j) &= \beta_0 \quad \forall j \in J_1 \\ \beta^T r^j &= 0 \quad \forall j \in J_2 \end{aligned}$$

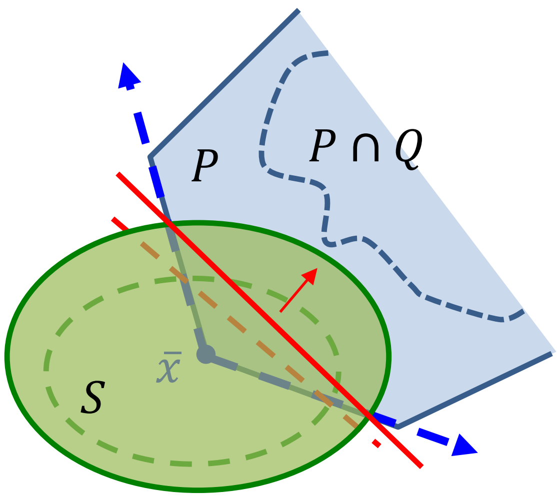
- A system of linear equalities
  - $|J_1| + |J_2| = n$ equations and $n + 1$ variables ($\beta \in \mathbb{R}^n, \beta_0 \in \mathbb{R}$)
  - a **unique solution** (except for a constant factor) since $\{r^j | j \in J\}$ are linearly independent
  - analytical solution: $\boldsymbol{\beta_0 = \sum_{i \in J} \frac{1}{\bar{\lambda}_i} b_i - 1}$ $\quad\quad$ $\boldsymbol{\beta_j = \sum_{i \in J} \frac{1}{\bar{\lambda}_i} a_{ij} \;\; \forall j \in N}$
- An equivalent but more popular version in the literature

$$\sum_{j \in J} \frac{1}{\bar{\lambda}_j} t_j \geq 1$$

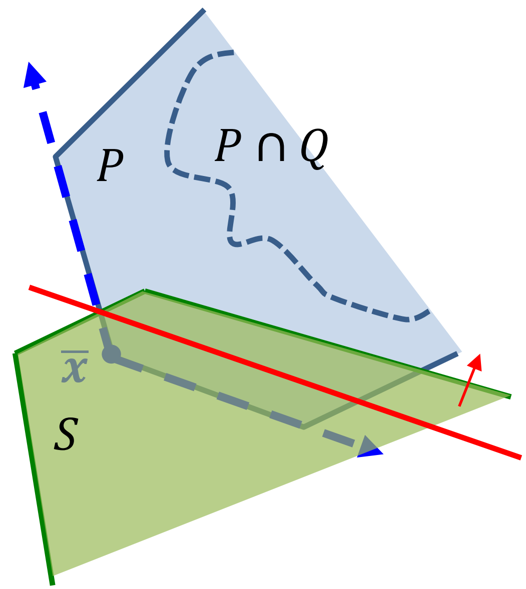
Balas, E., 1971. Intersection cuts—a new type of cutting planes for integer programming. *Operations Research*, *19*(1), pp.19-39.

Carnegie Mellon University

Geometric Interpretation



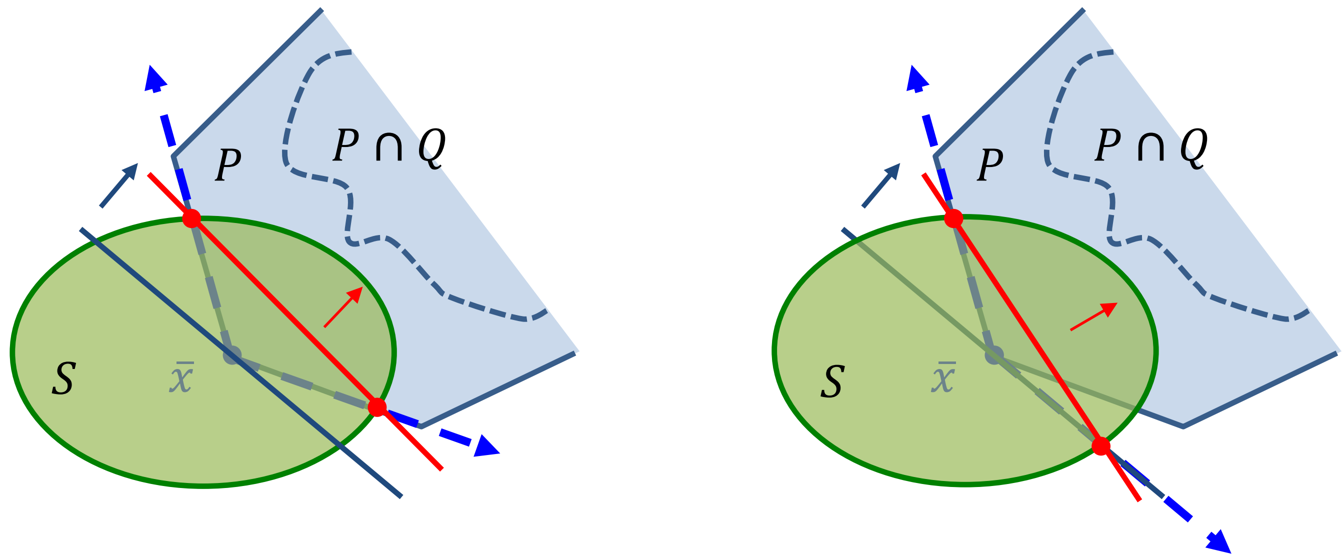
The larger S → the deeper cut



The intersection cut is parallel to an extreme ray in Rec(S)

Carnegie Mellon University

Degeneracy



- The **degeneracy** will not affect the correctness of the intersection cut formula
- The **choice of a basis** will lead to different (and valid) intersection cuts
- In general, **no dominance** relationship among these cuts is guaranteed

Implementation Details

- ✓ $\bar{\lambda}_i$ should be approximated below for **numerical validity**
  - a valid approximation to the intersection cut

- ✓ Scale a cut and perform reduction on small coefficients if necessary for **numerical stability**

- ✓ For a more generic LP as follows, the derivation for intersection cut has to be updated
  - **extreme rays** r^j
  - **intersection cut formula**

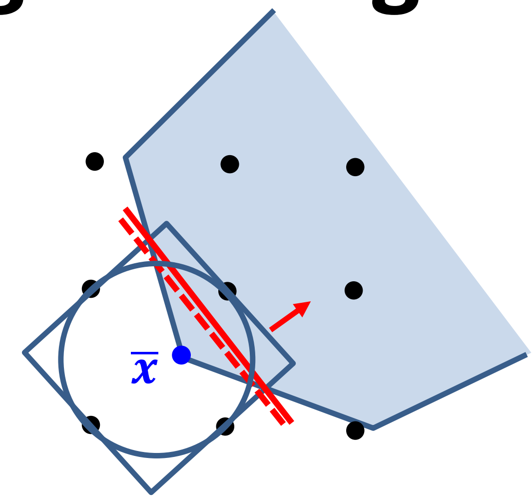
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & Ax \leq b \\ & x^L \leq x \leq x^U \end{aligned}$$

$$\sum_{j \in J^L} \frac{1}{\bar{\lambda}_j} t_j + \sum_{j \in J^U} \frac{-1}{\bar{\lambda}_j} t_j \geq 1$$

J^L: index set of non-basic variables at lower bounds
J^U: index set of non-basic variables at upper bounds

Mixed Integer Linear Programming

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & Ax = b \\ & x \geq 0 \\ & x_i \in Z \qquad i \in N \end{aligned}$$



- The hypersphere can be selected as a valid convex set S
  - S is a convex set ✔
  - $\bar{x} \in int(S)$ ✔
  - $int(S) \cap (P \cap Q) = \emptyset$ ✔
- Hard to find the "optimal" set S
- $\bar{\lambda}_j$ can be identified analytically

Carnegie Mellon University

Reverse Convex Programming

A constraint $g(x) \geq 0$ is called **reverse convex** if g is convex

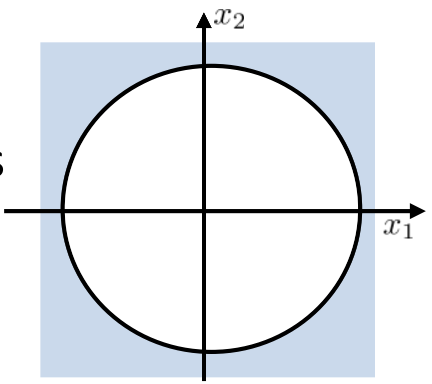
$$
\begin{aligned}
\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & c^T x \\
\text{subject to} \quad & f_k(x) \leq 0 \qquad \forall k = 1, 2, \ldots, p \\
& g_l(x) \geq 0 \qquad \forall l = 1, 2, \ldots, q
\end{aligned}
$$

Convex

Reverse Convex

where $f_k(x)$ and $g_l(x)$ are both convex on $\mathbb{R}^n$

- $f_k(x) \leq 0$ can be outer-approximated by linear inequalities
- $g_l(x) \geq 0$ represent the "complicated" constraints



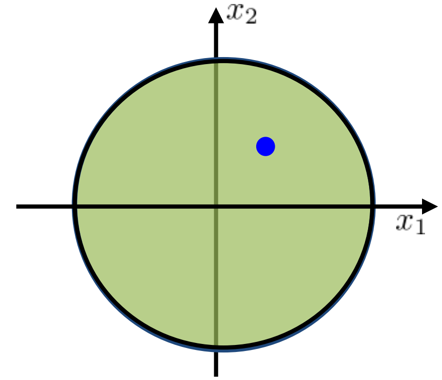
$x_1^2 + x_2^2 \geq 1$

Hillestad, R.J. and Jacobsen, S.E., 1980. Reverse convex programming. *Applied Mathematics and Optimization*, *6*(1), pp.63-78.

Carnegie Mellon University

Reverse Convex Programming

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c^T x \\ \text{subject to} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$



$x_1^2 + x_2^2 \leq 1$

- ❑ Define $S = \{x \in \mathbb{R}^n : g_{\bar{l}}(x) \leq 0\}$ for some $\bar{l}$ such that $g_{\bar{l}}(\bar{x}) < 0$
  - S is a convex set ✔
  - $\bar{x} \in int(S)$ ✔
  - $int(S) \cap (P \cap Q) = \emptyset$ ✔

- ❑ $\bar{\lambda}_j$ can be identified via solving $g_{\bar{l}}(\bar{x} + \lambda_j r^j) = 0$ with $\lambda_j \geq 0$

Hillestad, R.J. and Jacobsen, S.E., 1980. Reverse convex programming. *Applied Mathematics and Optimization*, *6*(1), pp.63-78.

Polynomial Programming

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & p_0(x) \\ \text{subject to} \quad & p_i(x) \leq 0 \qquad \forall i = 1, 2..., m \end{aligned}$$

where $p_i(x)$ is a **polynomial** function with respect to $x \in \mathbb{R}^n$

- e.g. $p_i(x) = 2 + 3x_1 - 3.2x_1x_2^2 + 4x_2^4, d = 4$

Define $m_r(x) := [1, x_1, x_2 \dots x_n, x_1^2, x_1x_2, \dots, x_n^2, \dots, x_n^r]^T$, where $r = \lceil {}^{d_{max}}/_{2} \rceil$

$$p_i(x) = m_r^T(x) A_i m_r(x) \leq 0 \iff \langle A_i, m_r(x) \cdot m_r^T(x) \rangle \leq 0$$

where A_i is an appropriately defined symmetric matrix

$$\langle A, B \rangle = \sum_i \sum_j a_{ij} b_{ij}$$

Moment-based Reformulation (lifted space)

$$p_i(x) = \underbrace{[1 \; x_1 \; x_2 \; x_1^2 \; x_1x_2 \; x_2^2]}_{m_r^T(x)} \underbrace{\begin{bmatrix} 2 & 1.5 & & & & \\ 1.5 & & & & & \\ & & & & & \\ & & & & -1.6 & \\ & & & -1.6 & 4 & \\ & & & & & \end{bmatrix}}_{A_i} \underbrace{\begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1^2 \\ x_1x_2 \\ x_2^2 \end{bmatrix}}_{m_r(x)}$$

$$X := \begin{bmatrix} X_{11} & X_{12} & X_{13} & X_{14} & X_{15} & X_{16} \\ X_{21} & X_{22} & X_{23} & X_{24} & X_{25} & X_{26} \\ X_{31} & X_{32} & X_{33} & X_{34} & X_{35} & X_{36} \\ X_{41} & X_{42} & X_{43} & X_{44} & X_{45} & X_{46} \\ X_{51} & X_{52} & X_{53} & X_{54} & X_{55} & X_{56} \\ X_{61} & X_{62} & X_{63} & X_{64} & X_{65} & X_{66} \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_2 & x_1^2 & x_1x_2 & x_2^2 \\ x_1 & x_1^2 & x_1x_2 & x_1^3 & x_1^2x_2 & x_1x_2^2 \\ x_2 & x_1x_2 & x_2^2 & x_1^2x_2 & x_1x_2^2 & x_2^3 \\ x_1^2 & x_1^3 & x_1^2x_2 & x_1^4 & x_1^3x_2 & x_1^2x_2^2 \\ x_1x_2 & x_1^2x_2 & x_1x_2^2 & x_1^3x_2 & x_1^2x_2^2 & x_1x_2^3 \\ x_2^2 & x_1x_2^2 & x_2^3 & x_1^2x_2^2 & x_1x_2^3 & x_2^4 \end{bmatrix} = m_r(x)m_r^T(x)$$

$$\begin{aligned} \text{minimize} \quad & \langle A_0, X \rangle \\ \text{subject to} \quad & \langle A_i, X \rangle \leq 0 \qquad \forall i = 1, 2, ..., m \\ & X = m_r(x) \cdot m_r^T(x) \end{aligned}$$

Bienstock, D., Chen, C. and Munoz, G., 2016. Outer-product-free sets for polynomial optimization and oracle-based cuts. *arXiv preprint arXiv:1610.04604*.

Polynomial Programming

Linear | Convex | **Non-convex**

$$X = m_r(x) \cdot m_r^T(x) \iff \text{consistency}, X \succeq 0, rank(X) \le 1$$

Diagonal entries $X_{ii} \geq 0$

$$X = \begin{bmatrix} 1 & x_1 & x_2 & x_1^2 & x_1x_2 & x_2^2 \\ x_1 & x_1^2 & x_1x_2 & x_1^3 & x_1^2x_2 & x_1x_2^2 \\ x_2 & x_1x_2 & x_2^2 & x_1^2x_2 & x_1x_2^2 & x_2^3 \\ x_1^2 & x_1^3 & x_1^2x_2 & x_1^2x_2^2 & x_1^3x_2 & x_1^2x_2^2 \\ x_1x_2 & x_1^2x_2 & x_1x_2^2 & x_1^3x_2 & x_1^2x_2^2 & x_1x_2^3 \\ x_2^2 & x_1x_2^2 & x_2^3 & x_1^2x_2^2 & x_1x_2^3 & x_2^4 \end{bmatrix}$$

$$\boxed{X_{62} = X_{53} = X_{35} = X_{26}}$$

$$Q := \{X \in \mathbb{S}^{n\times n} | X \succeq 0, rank(X) \le 1\}$$

Polyhedral Relaxation

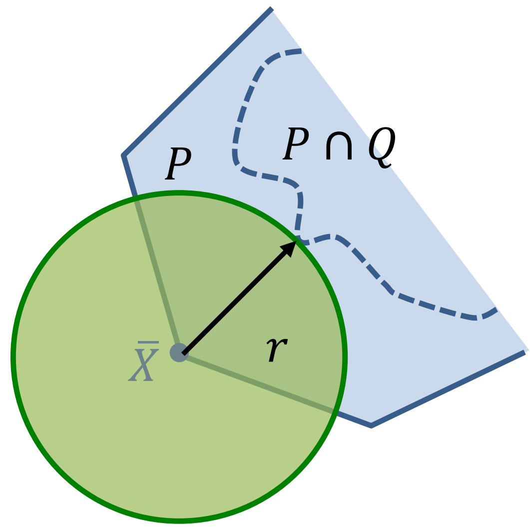
$$\begin{aligned} \underset{X}{\text{minimize}} \quad & \langle A_0, X\rangle \\ \text{subject to} \quad & \langle A_i, X\rangle \le 0 \qquad \forall i = 1, 2..., m \\ & X_{11} = 1 \\ & X_{ii} \ge 0 \qquad \forall i = 2..., n \\ & \text{consistency} \end{aligned}$$

Bienstock, D., Chen, C. and Munoz, G., 2016. Outer-product-free sets for polynomial optimization and oracle-based cuts. *arXiv preprint arXiv:1610.04604*.

Oracle Ball Cut

- Define S as a **ball** $B(\bar{X}, r)$ centering at $\bar{X}$ with a radius r

  - S is a convex set
  - $\bar{X} \in int(S)$
  - $int(S) \cap (P \cap Q) = \emptyset$



$$\begin{array}{lll} \underset{Y}{\text{minimize}} & \|\bar{X} - Y\|_F & (\#) \\ \text{subject to} & Y \succeq 0 & \\ & rank(Y) \leq 1 & Q \end{array}$$

- Problem (#) : calculate the shortest distance between $\bar{X}$ and a point from Q
  – it can be **analytically solved** ($\bar{\lambda}_j = r =$ (#) opt. val.)
- This convex set S can be enlarged (strengthened cut)

Bienstock, D., Chen, C. and Munoz, G., 2016. Outer-product-free sets for polynomial optimization and oracle-based cuts. *arXiv preprint arXiv:1610.04604*.

2×2 Cut

Theorem: $X \succcurlyeq 0$ and $rank(X) = 1$ iff all the 2×2 **principle minors** of X are zero

$$\bar{X} = \begin{array}{c c} \begin{array}{cccccc} & & 3 & & 5 & \end{array} & \\ \begin{bmatrix} \bar{X}_{11} & \bar{X}_{12} & \bar{X}_{13} & \bar{X}_{14} & \bar{X}_{15} & \bar{X}_{16} \\ \bar{X}_{21} & \bar{X}_{22} & \bar{X}_{23} & \bar{X}_{24} & \bar{X}_{25} & \bar{X}_{26} \\ \bar{X}_{31} & \bar{X}_{32} & \bar{X}_{33} & \bar{X}_{34} & \bar{X}_{35} & \bar{X}_{36} \\ \bar{X}_{41} & \bar{X}_{42} & \bar{X}_{43} & \bar{X}_{44} & \bar{X}_{45} & \bar{X}_{46} \\ \bar{X}_{51} & \bar{X}_{52} & \bar{X}_{53} & \bar{X}_{54} & \bar{X}_{55} & \bar{X}_{56} \\ \bar{X}_{61} & \bar{X}_{62} & \bar{X}_{63} & \bar{X}_{64} & \bar{X}_{65} & \bar{X}_{66} \end{bmatrix} & \begin{array}{c} \\ \\ 3 \\ \\ 5 \\ \\ \end{array} \end{array}$$

$X_{[i,j]}$: submatrix induced by i, j

$$det(X_{[i,j]}) = 0$$

If $\bar{X}_{[i,j]} \succ 0$ for some i, j ($1 \leq i < j \leq n$), define $S := \{X \in \mathbb{S}^{n \times n} | X_{[i,j]} \succeq 0\}$

- S is a convex set ✓
- $\bar{x} \in int(S)$ ✓
- $int(S) \cap (P \cap Q) = \emptyset$ ✓

$$int(S) : X_{[i,j]} \succ 0 \Rightarrow det(X_{[i,j]}) > 0$$

$$Q := \{X \in \mathbb{S}^{n \times n} | X \succeq 0, rank(X) \leq 1\}$$

$$det(X_{[i,j]}) = 0 \qquad \forall X \in Q$$

Bienstock, D., Chen, C. and Munoz, G., 2016. Outer-product-free sets for polynomial optimization and oracle-based cuts. *arXiv preprint arXiv:1610.04604*.

2×2 Cut

How to find the **intersection points**?

$$\bar{X} + \lambda R \in S = \{X \in \mathbb{S}^{n \times n} | X_{[i,j]} \succeq 0\}$$

extreme ray

$$\Updownarrow$$

$$\bar{X}_{[i,j]} + \lambda R_{[i,j]} \succeq 0$$

$$\Updownarrow$$

$$\begin{bmatrix} \bar{X}_{ii} + \lambda R_{ii} & \bar{X}_{ij} + \lambda R_{ij} \\ \bar{X}_{ji} + \lambda R_{ji} & \bar{X}_{jj} + \lambda R_{jj} \end{bmatrix} \succeq 0$$

- If $R_{[i,j]} \succcurlyeq 0$, no intersection point (set $\bar{\lambda} = +\infty$)
- Else, $\bar{\lambda}$ can be analytically computed

Bienstock, D., Chen, C. and Munoz, G., 2016. Outer-product-free sets for polynomial optimization and oracle-based cuts. *arXiv preprint arXiv:1610.04604*.

Computational Results

- Implementation: Python 2.7.13 / Gurobi 7.0.1
- Instances:
  - 26 Quadratically Constrained Quadratic Programs (**QCQP**) from GLOBALLib, $n = 6\sim63$
  - 99 **BoxQP** (non-convex quadratic objective, bound constraints), $n = 12\sim126$
- Compare the **root node bound**
  - McCormick estimator and RLT (Reformulation Linearization Technique) relaxation
- Stopping conditions:
  - Time limit 600 sec
  - No improvement in obj. val. (10 iter)
  - No violated cut
  - LP becomes numerically unstable

$OPT = 100$
$RLT = 80$
$GLB = 90$

$$\text{Initial Gap} = \frac{OPT - RLT}{|OPT| + \epsilon} \qquad {}^{20}/_{100}$$

$$\text{End Gap} = \frac{OPT - GLB}{|OPT| + \epsilon} \qquad {}^{10}/_{100}$$

$$\text{Gap Closed} = \frac{GLB - RLT}{OPT - RLT} \qquad {}^{10}/_{20}$$

Bienstock, D., Chen, C. and Munoz, G., 2016. Outer-product-free sets for polynomial optimization and oracle-based cuts. *arXiv preprint arXiv:1610.04604*.

Carnegie Mellon University

Computational Results

OB: Oracle Ball Cuts

SO: Strengthened OB

OA: Outer Approximation cuts for $X \succcurlyeq 0$

2x2: 2×2 cuts

Cut Family	Initial Gap	End Gap	Closed Gap	# Cuts	Iters	Time (s)	LPTime (%)
OB	1387.92%	1387.85%	1.00%	16.48	17.20	2.59	2.06%
SO		1387.83%	8.77%	18.56	19.52	4.14	2.29%
OA		1001.81%	8.61%	353.40	83.76	33.25	7.51%
2x2 + OA		1003.33%	32.61%	284.98	118.08	30.40	15.03%
SO+2x2+OA		1069.59%	31.91%	174.79	107.16	29.55	12.56%

Averages for GLOBALLib instances

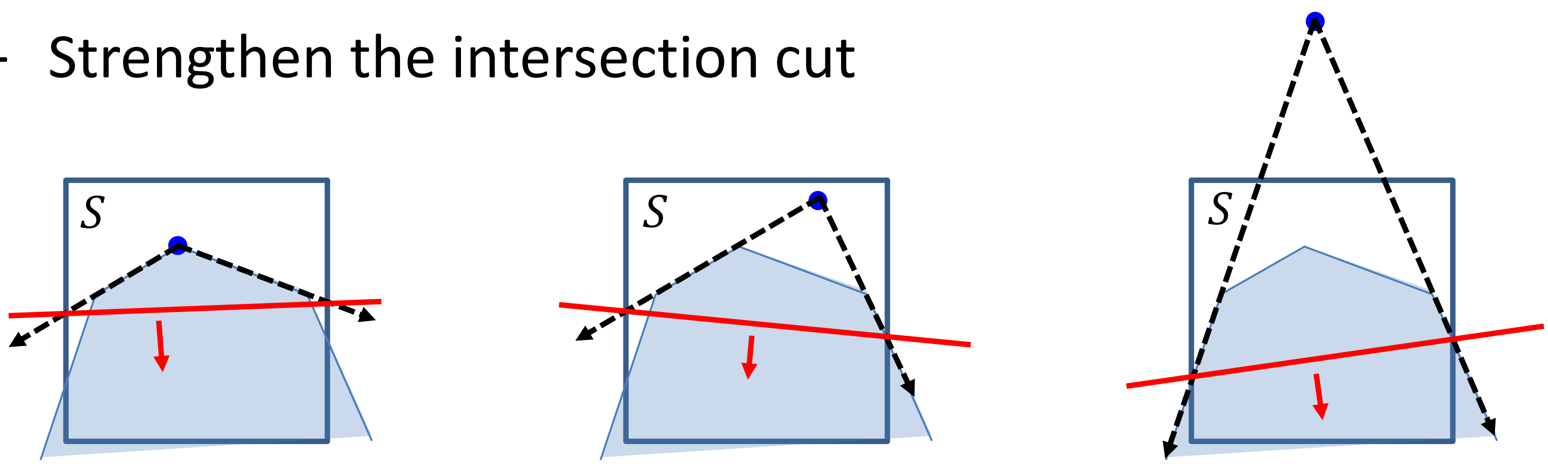
Cut Family	Initial Gap	End Gap	Closed Gap	# Cuts	Iters	Time (s)	LPTime (%)
OB	103.59%	103.56%	0.04%	12.84	13.62	127.15	0.40%
SO		103.33%	0.34%	14.34	15.45	132.07	0.49%
OA		30.88%	75.55%	676.90	137.52	459.28	31.80%
2x2 + OA		32.84%	74.52%	349.21	140.40	473.18	28.76%
SO+2x2+OA		33.43%	74.03%	227.39	136.93	475.38	26.59%

Averages for BoxQP instances

Bienstock, D., Chen, C. and Munoz, G., 2016. Outer-product-free sets for polynomial optimization and oracle-based cuts. *arXiv preprint arXiv:1610.04604*.

Comments

- The intersection cut is quite **generic** and **computationally cheap** to generate if a set S is given
- How to find a valid set S for your problem? **NO GENERIC** ANSWER
- **Research opportunities**
  - Find a valid set S in your application
  - Strengthen the intersection cut



Balas, E. and Margot, F., 2013. Generalized intersection cuts and a new cut generating paradigm. *Mathematical Programming*, *137*(1-2), pp.19-35.

THANK YOU!