# New SOCP relaxation and branching rule for bipartite bilinear programs

Santanu S. Dey[1] Asteroide Santana[1] and Yang Wang[2]

[1]ISyE, Georgia Institute of Technology.

[2]School of Civil and Environmental Engineering, Georgia Institute of Technology.

Feb, 2018

# 1
# Introduction

# Bilinear Bipartite Problems (BBP)

$$\begin{array}{ll} \min_{x,y} & x^\top Q_0 y + d_1^\top x + d_2^\top y \\ \text{s.t.} & x^\top Q_k y + a_k^\top x + b_k^\top y + c_k = 0,\ k \in \{1,\dots,m\} \\ & l \le (x,y) \le u \\ & (x,y) \in \mathbb{R}^{n_1+n_2}, \end{array} \quad (1)$$

where

1. $n_1, n_2 \in \mathbb{Z}_+$,
2. $Q_0, Q_k \in \mathbb{R}^{n_1 \times n_2}$,
3. $d_1, a_k \in \mathbb{R}^{n_1}$,
4. $d_2, b_k \in \mathbb{R}^{n_2}$,
5. $c_k \in \mathbb{R}$,

for $k \in \{1, \dots, m\}$.

Without loss of generality, we assume that $l_i = 0,\ u_i = 1,\ i \in \{1, \dots, n_1 + n_2\}$.

# Graph corresponding to BBP

For a given instance of BBP, we associate a graph $G(V, E)$:

1. The set of vertices corresponds to the variables in the instance, and
2. There is an edge between two vertices if there is a degree two term involving the corresponding variables in the instance formulation.

# Outline

2

## A Second order cone programming relaxation of BBP

## 2.1
## Convex hull of one-constraint relaxation

# Convex hull of one-constraint relaxation

A very successful approach in integer linear programing is to generate cutting-planes implied by single constraint relaxations.

# Convex hull of one-constraint relaxation

## Theorem

*Let $n_1, n_2 \in \mathbb{Z}_+$, $V_1 \in \{1, \ldots, n_1\}$, $V_2 \in \{1, \ldots, n_2\}$, and $E \subseteq V_1 \times V_2$. Consider the one-constraint BBP set*

$$S := \left\{ (x, y, w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i \in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j, \; (i, j) \in E \\ (x, y, w) \in [0, 1]^{n_1 + n_2 + |E|} \end{array} \right\}.$$

*Then:*

$$\text{conv}(S) \text{ is SOCP-representable.}$$

We give a constructive proof.

# Convex hull of one-constraint relaxation

## Theorem

*Let $n_1, n_2 \in \mathbb{Z}_+$, $V_1 \in \{1, \ldots, n_1\}$, $V_2 \in \{1, \ldots, n_2\}$, and $E \subseteq V_1 \times V_2$. Consider the one-constraint BBP set*

$$S := \left\{ (x, y, w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i \in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j, \ (i,j) \in E \\ (x, y, w) \in [0,1]^{n_1 + n_2 + |E|} \end{array} \right\}.$$

*Then:*

conv($S$) is SOCP-representable.

We give a constructive proof.
Related results:

- M. Tawarmalani, N.V. Sahinidis [2001]
- M. Tawarmalani, J.-P. P. Richard, K. Chung [2010]
- K. M. Anstreicher, S. Burer [2010]
- T. T. Nguyen, J.-P. P. Richard, M. Tawarmalani [2012]
- A. Gupte Thesis [2012]
- B. Kocuk, SSD, and X. A. Sun [2018]
- And many others...(especially on convex envelopes of bilinear functions)

2.2

## Proof of convex hull result

# High-level Sketch: Disjunctive argument

1. If $S$ is compact, then $\text{conv}(S) = \text{conv}(\text{extr}(S))$.

# High-level Sketch: Disjunctive argument

1. If $S$ is compact, then $\text{conv}(S) = \text{conv}(\text{extr}(S))$.
2. $\Rightarrow$ If $S \supseteq \bigcup_{i=1}^{k} T_i \supseteq \text{extr}(S)$, then

$$\text{conv}(S) = \text{conv}\left(\bigcup_{i=1}^{k} \underbrace{\text{conv}(T_i)}_{\text{SOCP Representable}}\right)$$

$$\Downarrow$$

$$\underbrace{\text{conv}(S)}_{\text{SOCP representable}} = \text{conv}\left(\bigcup_{i=1}^{k} \underbrace{\text{conv}(T_i)}_{\text{SOCP Representable}}\right)$$

## 2.2.1
## Step 1: "Support" of extreme points

# Step 1: "Support" of extreme points

$$S := \left\{ (x,y,w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i\in V_1} a_i x_i + \sum_{j\in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j,\ (i,j) \in E \\ (x,y,w) \in [0,1]^{n_1+n_2+|E|} \end{array} \right\}.$$

## Proposition

*Let $(\bar{x}, \bar{y}, \bar{w})$ be an extreme point of the set $S$. Then, there exists $U \subseteq V_1 \cup V_2$, where*

1. *$U = \{i_0, j_0\}$ where $(i_0, j_0) \in E$, or,*
2. *$U = \{i_0\}$ where $i_0 \in V_1$ is an isolated node, or,*
3. *$U = \{j_0\}$ where $j_0 \in V_2$ is an isolated node,*

*such that $\bar{x}_i \in \{0,1\}$, $\forall i \in V_1 \setminus U$, and $\bar{y}_j \in \{0,1\}$, $\forall j \in V_2 \setminus U$.*

# Step 1: "Support" of extreme points

$$S := \left\{ (x, y, w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i\in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j, \; (i,j) \in E \\ (x, y, w) \in [0,1]^{n_1 + n_2 + |E|} \end{array} \right\}.$$

## Proposition

*Let $(\bar{x}, \bar{y}, \bar{w})$ be an extreme point of the set $S$. Then, there exists $U \subseteq V_1 \cup V_2$, where*

1. *$U = \{i_0, j_0\}$ where $(i_0, j_0) \in E$, or,*
2. *$U = \{i_0\}$ where $i_0 \in V_1$ is an isolated node, or,*
3. *$U = \{j_0\}$ where $j_0 \in V_2$ is an isolated node,*

*such that $\bar{x}_i \in \{0,1\}$, $\forall i \in V_1 \setminus U$, and $\bar{y}_j \in \{0,1\}$, $\forall j \in V_2 \setminus U$.*

$\approx$ No two $x$ (or $y$s) can be strictly between bounds in an extreme point.

# Step 1: "Support" of extreme points

Consider a feasible point $(\bar{x}, \bar{y}, \bar{w})$ such that $0 < \bar{x}_1, \bar{x}_2 < 1$:

$$q_{11}x_1y_1 + q_{12}x_1y_2 + q_{21}x_2y_1 + q_{22}x_2y_2 \quad + \cdots + \sum_{i \in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j = -c$$

$$w_{ij} = x_i y_j$$

$$x, y, w \in [0, 1].$$

# Step 1: "Support" of extreme points

Consider a feasible point $(\bar{x}, \bar{y}, \bar{w})$ such that $0 < \bar{x}_1, \bar{x}_2 < 1$:

$$q_{11}\bar{x}_1\bar{y}_1 + q_{12}\bar{x}_1\bar{y}_2 + q_{21}\bar{x}_2\bar{y}_1 + q_{22}\bar{x}_2\bar{y}_2 \quad + \cdots + a_1\bar{x}_1 + a_2\bar{x}_2 + \text{constant} = -c$$

$$\bar{w}_{1j} = \bar{x}_1\bar{y}_j \ \forall j$$

$$\bar{w}_{2j} = \bar{x}_2\bar{y}_j \ \forall j$$

$$\bar{w}_{ij} = \bar{x}_i\bar{y}_j \qquad i \neq 2, \forall j$$

$$\bar{x}, \bar{y}, \bar{w} \in [0, 1].$$

# Step 1: "Support" of extreme points

Consider a feasible point $(\bar{x}, \bar{y}, \bar{w})$ such that $0 < \bar{x}_1, \bar{x}_2 < 1$:

$$\tilde{q}_1 \bar{x}_1 + \tilde{q}_2 \bar{x}_2 = \text{constant}$$

$$\bar{w}_{1j} = \bar{x}_1 \bar{y}_j \quad \forall j, \qquad \bar{w}_{2j} = \bar{x}_2 \bar{y}_j \quad \forall j$$

# Step 1: "Support" of extreme points

Consider a feasible point $(\bar{x}, \bar{y}, \bar{w})$ such that $0 < \bar{x}_1, \bar{x}_2 < 1$:

$$\underset{\substack{\uparrow \\ +\epsilon}}{\tilde{q}_1 \bar{x}_1} + \tilde{q}_2 \underset{\substack{\downarrow \\ -\delta}}{\bar{x}_2} = \text{constant}$$

$$\overset{\uparrow}{\bar{w}_{1j}} = \overset{\substack{+\epsilon \\ \uparrow}}{\bar{x}_1} \bar{y}_j \quad \forall j \qquad \underset{\downarrow}{\bar{w}_{2j}} = \underset{\substack{\downarrow \\ -\delta}}{\bar{x}_2} \bar{y}_j \quad \forall j$$

# Step 1: "Support" of extreme points

Consider a feasible point $(\bar{x}, \bar{y}, \bar{w})$ such that $0 < \bar{x}_1, \bar{x}_2 < 1$:

$$\underset{\substack{\downarrow \\ -\epsilon}}{\tilde{q}_1 \bar{x}_1} + \overset{\substack{+\delta \\ \uparrow}}{\tilde{q}_2 \bar{x}_2} = \text{constant}$$

$$\underset{\downarrow}{\bar{w}_{1j}} = \underset{\substack{\downarrow \\ -\epsilon}}{\bar{x}_1} \bar{y}_j \quad \forall j \qquad \overset{\uparrow}{\bar{w}_{2j}} = \overset{\substack{+\delta \\ \uparrow}}{\bar{x}_2} \bar{y}_j \quad \forall j$$

## 2.2.2
## Step 2: Analyzing each fixing

# Step 2: Analyzing each fixing

At most one $x_i$ and at most one $y_j$ unfixed in an extreme point:

$$
\begin{aligned}
\tilde{q}x_1y_1 + \tilde{a}x_1 + \tilde{b}y_1 &= \tilde{c} \\
w_{11} &= x_1y_1 \\
w_{1j} &= x_1\bar{y}_j \qquad \forall j \neq 1 \\
w_{i1} &= \bar{x}_iy_1 \qquad \forall i \neq 1 \\
x_1, y_1, w &\in [0,1]^2 \times [0,1]^{n_1+n_2+1}.
\end{aligned}
$$

# Step 2: Analyzing each fixing

## At most one $x_i$ and at most one $y_j$ unfixed in an extreme point:

$$\begin{aligned}
\tilde{q}x_1y_1 + \tilde{a}x_1 + \tilde{b}y_1 &= \tilde{c} \\
w_{11} &= x_1y_1 \\
w_{1j} &= x_1\bar{y}_j && \forall j \neq 1 \\
w_{i1} &= \bar{x}_iy_1 && \forall i \neq 1 \\
x_1, y_1, w &\in [0,1]^2 \times [0,1]^{n_1+n_2+1}.
\end{aligned}$$

Case 1: $\tilde{q} \neq 0$

$$\begin{aligned}
\tilde{q}x_1y_1 + \tilde{a}x_1 + \tilde{b}y_1 &= \tilde{c} \\
w_{11} &= \frac{\tilde{c}-(\tilde{a}x_1+\tilde{b}y_1)}{\tilde{q}} \\
w_{1j} &= x_1\bar{y}_j && \forall j \\
w_{i1} &= \bar{x}_iy_1 && \forall i \\
0 \leq x_1, y_1 &\leq 1
\end{aligned}$$

Case 2: $\tilde{q} = 0,\ \tilde{a} \neq 0$

$$\begin{aligned}
w_{11} &= \tilde{c}/\tilde{a}y_1 - \tilde{b}/\tilde{a}y_1^2 \\
x_1 &= \tilde{c}/\tilde{a} - \tilde{b}/\tilde{a}y_1 \\
w_{1j} &= x_1\bar{y}_j && \forall j \neq 1 \\
w_{i1} &= \bar{x}_iy_1 && \forall i \neq 1 \\
0 \leq w_1 \leq 1, \alpha \leq y_1 &\leq \beta
\end{aligned}$$

Other cases “easy".

# Step 2: Analyzing each fixing

At most one $x_i$ and at most one $\bar{y}_j$ in an extreme point:

$$\begin{aligned} \tilde{q}x_1y_1 + \tilde{a}x_1 + \tilde{b}y_1 &= \tilde{c} \\ w_{11} &= x_1y_1 \\ w_{1j} &= x_1\bar{y}_j && \forall j \neq 1 \\ w_{i1} &= \bar{x}_iy_1 && \forall i \neq 1 \\ x_1, y_1, w &\in [0,1]^2 \times [0,1]^{n_1+n_2+1}. \end{aligned}$$

Case 1: $\underline{\tilde{q} \neq 0}$

$$\begin{aligned} \tilde{q}x_1y_1 + \tilde{a}x_1 + \tilde{b}y_1 &= \tilde{c} \quad \leftarrow \boxed{\text{Quad}} \\ w_{11} &= \frac{\tilde{c}-(\tilde{a}x_1+\tilde{b}y_1)}{\tilde{q}} \\ w_{1j} &= x_1\bar{y}_j \quad \forall j \\ w_{i1} &= \bar{x}_iy_1 \quad \forall i \\ 0 \leq x_1, y_1 &\leq 1 \end{aligned}$$

Case 2: $\underline{\tilde{q} = 0,\ \tilde{a} \neq 0}$

$$\begin{aligned} w_{11} &= \tilde{c}/\tilde{a}y_1 - \tilde{b}/\tilde{a}y_1^2 \quad \leftarrow \boxed{\text{Quad}} \\ x_1 &= \tilde{c}/\tilde{a} - \tilde{b}/\tilde{a}y_1 \\ w_{1j} &= x_1\bar{y}_j \quad \forall j \neq 1 \\ w_{i1} &= \bar{x}_iy_1 \quad \forall i \neq 1 \\ 0 \leq w_1 \leq 1, &\ \alpha \leq y_1 \leq \beta \end{aligned}$$

Other cases “easy".

# Step 2: Analyzing each fixing

- Quadratic equation in two variables $x \in \mathbb{R}^2$ with box constraints,
- $w_i = l_i^\top x + l_{i,0} \; \forall i$

# Step 2: Analyzing each fixing

- Quadratic equation in two variables $x \in \mathbb{R}^2$ with box constraints,
- $w_i = l_i^\top x + l_{i,0} \forall i$ ✓

## Lemma

*Let* $B = \{(x, w) \in [0,1]^n \times \mathbb{R} \,|\, x \in B_0,\ w = l^\top x + l_0\}$*, where* $B_0 \subseteq \mathbb{R}^n$*, and* $l^\top x + l_0$ *is an affine function of* $x$*. Then,*

$$\text{conv}(B) = \{(x, w) \in [0,1]^n \times \mathbb{R} \,|\, x \in \text{conv}(B_0),\ w = l^\top x + l_0\}.$$

# Step 2: Analyzing each fixing

We are left with:

- Quadratic equation in two variables $x \in \mathbb{R}^2$ with box constraints.

## Lemma (Richard, Tawarmalani (2013))

*Let $f : [0,1]^n \to \mathbb{R}$ be a continuous function. Then*

$$\text{conv}\left(\{x \in [0,1]^n \mid f(x) = 0\}\right) = \text{conv}\left(\{x \in [0,1]^n \mid f(x) \leq 0\}\right) \bigcap \text{conv}(\{x \in [0,1]^n \mid f(x) \geq 0\}).$$

# Step 2: Analyzing each fixing

We are left with:

- Quadratic inequality in two variables $x \in \mathbb{R}^2$ with box constraints.

# Step 2: Analyzing each fixing

We are left with:

- Quadratic inequality in two variables $x \in \mathbb{R}^2$ with box constraints.

In all case the convex hull of resulting set can be shown to SOCP representable.

## 2.2.3
## Wrap-up of proof

# Wrap-up

$$S := \left\{ (x,y,w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i\in V_1} a_i x_i + \sum_{j\in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j,\ (i,j) \in E \\ (x,y,w) \in [0,1]^{n_1+n_2+|E|} \end{array} \right\}.$$

# Wrap-up

$$S := \left\{ (x, y, w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i \in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j, \ (i,j) \in E \\ (x, y, w) \in [0,1]^{n_1 + n_2 + |E|} \end{array} \right\}.$$

1. In order to obtain extreme points fix variables, so that no more than one $x$ and one $y$ are left free.
2. For each fixing, obtain a SOCP relaxation of the extreme points.

# Wrap-up

$$S := \left\{ (x,y,w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i \in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j, \; (i,j) \in E \\ (x,y,w) \in [0,1]^{n_1+n_2+|E|} \end{array} \right\}.$$

1. In order to obtain extreme points fix variables, so that no more than one $x$ and one $y$ are left free.
2. For each fixing, obtain a SOCP relaxation of the extreme points.
3. Finally, convex hull of union of SOCP representable sets is SOCP representable.

# Wrap-up

$$S := \left\{ (x,y,w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i \in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j, \ (i,j) \in E \\ (x,y,w) \in [0,1]^{n_1+n_2+|E|} \end{array} \right\}.$$

1. In order to obtain extreme points fix variables, so that no more than one $x$ and one $y$ are left free.
2. For each fixing, obtain a SOCP relaxation of the extreme points.
3. Finally, convex hull of union of SOCP representable sets is SOCP representable.

Remarks:

1. A total of $\mathcal{O}(n_1 n_2 2^{n_1+n_2})$ sets in the disjunction.

# Wrap-up

$$S := \left\{ (x, y, w) \;\middle|\; \begin{array}{l} \sum_{(i,j)\in E} q_{ij} w_{ij} + \sum_{i \in V_1} a_i x_i + \sum_{j \in V_2} b_j y_j + c = 0, \\ w_{ij} = x_i y_j, \; (i,j) \in E \\ (x, y, w) \in [0,1]^{n_1+n_2+|E|} \end{array} \right\}.$$

1. In order to obtain extreme points fix variables, so that no more than one $x$ and one $y$ are left free.
2. For each fixing, obtain a SOCP relaxation of the extreme points.
3. Finally, convex hull of union of SOCP representable sets is SOCP representable.

Remarks:

1. A total of $\mathcal{O}(n_1 n_2 2^{n_1+n_2})$ sets in the disjunction.
2. Key strength constructing convex hull in the extended $w$ space.

2.3
## SOCP Relaxation of BBP

# SOCP relaxation and its strength

## Definition (SOCP relaxation of BBP)

$$S^{SOCP} = \bigcap_{k=1}^{m} \text{conv}(S_k),$$

where $S_k = \{(x, y, w) \in [0, 1]^{n_1 \times n_2 \times |E|} \mid x^\top Q_k y + a_k^\top x + b_k^\top y + c_k = 0, w_{ij} = x_i y_j \ \forall (i, j) \in E\}$ and $E$ is the edge set of the graph corresponding to the BBP instance (and not just of one row).

# SOCP relaxation and its strength

## Definition (SOCP relaxation of BBP)

$$S^{SOCP} = \bigcap_{k=1}^{m} \text{conv}(S_k),$$

where $S_k = \{(x, y, w) \in [0,1]^{n_1 \times n_2 \times |E|} \mid x^\top Q_k y + a_k^\top x + b_k^\top y + c_k = 0, w_{ij} = x_i y_j \ \forall (i,j) \in E\}$ and $E$ is the edge set of the graph corresponding to the BBP instance (and not just of one row).

## Theorem

*For any BBP, we have that*

$$\text{proj}_{x,y,w}\left(S^{SDP}\right) \bigcap S^{QBP} \supseteq S^{SOCP}.$$

$$\begin{aligned} S^{QBP} &:= \{(x,y,w) \mid \sum_{(ij) \in E} (Q_k)_{ij} w_{ij} + a_k^\top x + b_k^\top y + c_k = 0 \ k \in [m]\} \\ & \bigcap \text{conv}\left(\{(x,y,w) \in [0,1]^{n_1+n_2+|E|} \mid w_{ij} = x_i y_j \ \forall (i,j) \in E\}\right). \\ S^{SDP} &:= \text{standard SDP relaxation.} \end{aligned}$$

# 3
# Branching rules

# Branching rule: key ideas

- Suppose we have decided to branch on the variable $x_1$. We examine all such two variable sets involving $x_1$ obtained from each of the constraints.

# Branching rule: key ideas

- Suppose we have decided to branch on the variable $x_1$. We examine all such two variable sets involving $x_1$ obtained from each of the constraints.
- For each of these sets, there is an *ideal point* to divide the range of $x_1$ so that the sum of the volume of the two convex hulls of the two-dimensional sets corresponding to the two resulting branches is minimized.

# Branching rule: key ideas

- Suppose we have decided to branch on the variable $x_1$. We examine all such two variable sets involving $x_1$ obtained from each of the constraints.
- For each of these sets, there is an *ideal point* to divide the range of $x_1$ so that the sum of the volume of the two convex hulls of the two-dimensional sets corresponding to the two resulting branches is minimized.
- We develop a simple heuristic to find *an "ideal" range*.

# Branching rule: key ideas

- Suppose we have decided to branch on the variable $x_1$. We examine all such two variable sets involving $x_1$ obtained from each of the constraints.
- For each of these sets, there is an *ideal point* to divide the range of $x_1$ so that the sum of the volume of the two convex hulls of the two-dimensional sets corresponding to the two resulting branches is minimized.
- We develop a simple heuristic to find *an "ideal" range*.
- Collect all such ideal rages corresponding to all the two-dimensional sets involving $x_1$.

# Branching rule: key ideas

- Suppose we have decided to branch on the variable $x_1$. We examine all such two variable sets involving $x_1$ obtained from each of the constraints.
- For each of these sets, there is an *ideal point* to divide the range of $x_1$ so that the sum of the volume of the two convex hulls of the two-dimensional sets corresponding to the two resulting branches is minimized.
- We develop a simple heuristic to find *an “ideal" range*.
- Collect all such ideal rages corresponding to all the two-dimensional sets involving $x_1$.
- We develop a heuristic to select one of these points (based on corresponding volume reduction) to finally partition the domain of $x_1$.

# Illustration of Ideal range: hyperbola, two branches

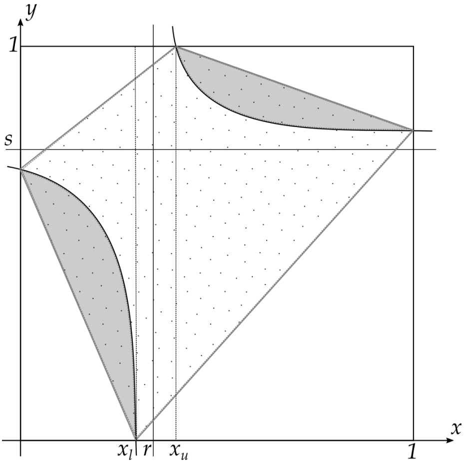



Figure: Convex hull of the set defined by the intersection of two branches of a hyperbola with the $[0, 1]^2$ box.

# Illustration of Ideal range: hyperbola, one branches

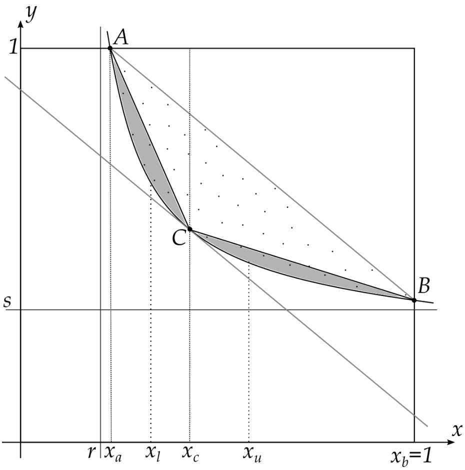



Figure: Convex hull of the set defined by the intersection of a single branch of a hyperbola with the $[0, 1]^2$ box. $A$ and $B$ are the intersection points of the curve with the $[0, 1]^2$ box and $C$ is the point of the curve at which the tangent line is parallel to $AB$.

4

# Computational results

## 4.1
## Instances

# FEM Model Updating problem

Find $x$ and $y^l$ so that:

$$\left[K_0 + \sum_{i=1}^{n_1} x_i K_i - \lambda_l M\right] \begin{bmatrix} \bar{y}^l \\ y^l \end{bmatrix} = 0, \; l \in \{1, \ldots, n_3\},$$

# FEM Model Updating problem

Find $x$ and $y^l$ so that:

$$\left[K_0 + \sum_{i=1}^{n_1} x_i K_i - \lambda_l M\right] \begin{bmatrix} \bar{y}^l \\ y^l \end{bmatrix} = 0, \; l \in \{1, \ldots, n_3\},$$

Minimizing $l_1$ error:

$$\begin{aligned} \min \quad & \sum_{k=1}^{m} z_k \\ \text{s.t.} \quad & |x^\top Q_k y + a_k^\top x + b_k^\top y + c_k| = z_k, \; k \in \{1, \ldots, m\} \\ & x \in [0,1]^{n_1}, y \in [0,1]^{n_2}, \end{aligned} \tag{2}$$

# FEM Model Updating problem

Find $x$ and $y^l$ so that:

$$\left[K_0 + \sum_{i=1}^{n_1} x_i K_i - \lambda_l M\right] \begin{bmatrix} \bar{y}^l \\ y^l \end{bmatrix} = 0, \ l \in \{1, \dots, n_3\},$$

Minimizing $l_1$ error:

$$\begin{aligned} \min \quad & \sum_{k=1}^{m} z_k \\ \text{s.t.} \quad & |x^\top Q_k y + a_k^\top x + b_k^\top y + c_k| = z_k, \ k \in \{1, \dots, m\} \\ & x \in [0,1]^{n_1}, y \in [0,1]^{n_2}, \end{aligned} \tag{2}$$

BBP:

$$\begin{aligned} \min \quad & \sum_{k=1}^{m} z'_k + z''_k \\ \text{s.t.} \quad & x^\top Q_k y + a_k^\top x + b_k^\top y + c_k = z'_k - z''_k, \ k \in \{1, \dots, m\} \\ & x \in [0,1]^{n_1}, y \in [0,1]^{n_2}. \\ & 0 \le z'_k, z''_k \le u, \ k \in \{1, \dots, m\}. \end{aligned} \tag{3}$$

# Size of instances

Table: Instances description

| Inst | # var.s | # eqn.s | # bilinear terms | # max var.s per row |
|---|---|---|---|---|
| inst1 | 186 | 312 | 990 | 9 |
| inst2 | 186 | 312 | 954 | 9 |
| inst3 | 174 | 312 | 966 | 9 |
| inst4 | 174 | 312 | 972 | 9 |
| inst5 | 162 | 312 | 900 | 9 |
| inst6 | 150 | 312 | 780 | 10 |

## 4.2
## A more tractable relaxation

# A more tractable relaxation

## SOCP-light

1. The number of disjunction needed to model the convex hull of a single bilinear equation can be computationally prohibitive.
2. In our computational experiments, we write the convex hull of each row only in the space of the variable (and corresponding products) appearing in the row.

# A more tractable relaxation

## SOCP-light

1. The number of disjunction needed to model the convex hull of a single bilinear equation can be computationally prohibitive.
2. In our computational experiments, we write the convex hull of each row only in the space of the variable (and corresponding products) appearing in the row.
3. Fortunately, our instances are sparse, so not too many disjunctions.

# A more tractable relaxation

## SOCP-light

1. The number of disjunction needed to model the convex hull of a single bilinear equation can be computationally prohibitive.
2. In our computational experiments, we write the convex hull of each row only in the space of the variable (and corresponding products) appearing in the row.
3. Fortunately, our instances are sparse, so not too many disjunctions.

## Polyhedral outer-approximation

1. Basically piece-wise linear out approximation of the convex quadratic inequalities.
2. We use only the tangents where the curve intersects the $[0, 1]^2$ box.

# Illustration of outer-approximation

Figure: Hyperbola outer-approx

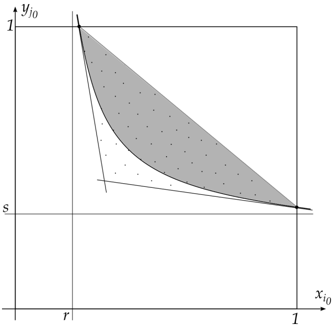
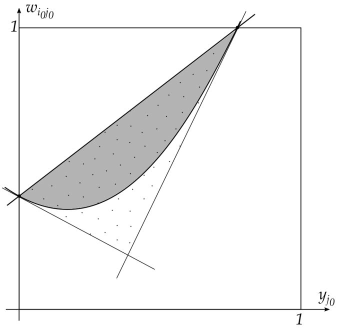



Figure: Parabola outer-approx



4.3

## Computational results

# Software and Hardware

Machine and OS: Windows 10 machine with 64-bit operating system, x64 based processor with 2.19GHz, and 32GB RAM.

1. LP and IP: Gurobi 7.5.1.
2. SDP: MOSEK ver 9.
3. Global solver: BARON 15.6.5.
4. Nonlinear solver: IPOPT.

Table: Root relaxations

| | Mc | | SDP+Mc | | SOCP | |
|---|---|---|---|---|---|---|
| Inst | Bound | Time | Bound | Time | Bound | Time |
| 1 | 0.1777 | 0.07 | 0.1777 | 35.89 | 0.1779 | 17.59 |
| 2 | 0.0000 | 0.05 | 0.0000 | 38.98 | 0.0000 | 20.93 |
| 3 | 0.2754 | 0.07 | 0.2754 | 44.02 | 0.2820 | 16.22 |
| 4 | 0.1009 | 0.08 | 0.1009 | 36.13 | 0.1010 | 20.71 |
| 5 | 0.3477 | 0.05 | 0.3477 | 31.58 | 0.3493 | 13.17 |
| 6 | 0.9776 | 0.05 | 0.9776 | 28.47 | 1.0027 | 11.64 |

Table: Branch-and-bound methods: dual bounds

| Inst | BB-SOCP-1 | BB-SOCP-2 | BB-SOCP-3 | BB-Mc | Mc-Descr ($2^{16}$) |
|---|---|---|---|---|---|
| 1 | 2.5074 | 0.1847 | 0.1823 | 0.1834 | 1.8539 |
| 2 | 2.8644 | 0.00000 | 0.0000 | 0.0000 | 2.1471 |
| 3 | 3.1308 | 0.2911 | 0.2898 | 0.2888 | 2.1427 |
| 4 | 3.1115 | 0.1053 | 0.1025 | 0.1041 | 2.4485 |
| 5 | 3.7896 | 0.3526 | 0.3539 | 0.3540 | 3.4027 |
| 6 | 4.6399 | 1.1110 | 1.0954 | 1.1519 | 4.1650 |

Running-time: 10 hours.

- *SOCP-1*: Uses the polyhedral relaxation with new branching rule.
- *SOCP-2*: Uses the same relaxation of BB-SOCP-1 above. The branching variable is selected according to the *gap-error-rule*. Then uses the *incumbent-rule* for branching point selection, whenever possible, otherwise uses the *maximum-deviation-rule*.
- *SOCP-3*: Same as BB-SOCP-2 except that uses *bisection for branching point selection*.
- *BB-Mc*: Uses McCormick relaxation with gap-error-rule as branching variable selection rule and *bisection for branching point selection*.

Table: Comparison with BARON

| | BB-SOCP-1 | | | BARON | | |
|---|---|---|---|---|---|---|
| Inst | Dual | Primal | % Gap | Dual | Primal | % Gap |
| 1 | 2.5074 | 3.4785 | 27.91 | 0.3312 | 3.4789 | 90.47 |
| 2 | 2.8644 | 3.4998 | 18.15 | 0.5245 | 3.4993 | 85.01 |
| 3 | 3.1308 | 3.6810 | 14.94 | 0.4760 | 3.6831 | 87.07 |
| 4 | 3.1115 | 3.7522 | 17.07 | 0.7863 | 3.7530 | 79.04 |
| 5 | 3.7896 | 4.1328 | 8.30 | 0.3840 | 4.1354 | 90.71 |
| 6 | 4.6399 | 5.6609 | 18.03 | 2.2657 | 5.6605 | 59.97 |

Running-time: 10 hours.

$$\%\text{gap} = \frac{\text{Primal} - \text{Dual}}{\textit{Primal}}$$

# Open Questions

1. SOCP-light/SOCP:
   - Strength of SOCP-light?

# Open Questions

1. SOCP-light/SOCP:
   - Strength of SOCP-light?
   - Better methods to obtain SOCP-light: theoretical and computational?

# Open Questions

1. SOCP-light/SOCP:
   - Strength of SOCP-light?
   - Better methods to obtain SOCP-light: theoretical and computational?
   - Taking row combinations?

# Open Questions

1. SOCP-light/SOCP:
   - Strength of SOCP-light?
   - Better methods to obtain SOCP-light: theoretical and computational?
   - Taking row combinations?
   - Using more structure: e.g. pooling problem?

# Open Questions

1. SOCP-light/SOCP:
    - Strength of SOCP-light?
    - Better methods to obtain SOCP-light: theoretical and computational?
    - Taking row combinations?
    - Using more structure: e.g. pooling problem?
2. Multi-linear: As long as we can finally reduce (by fixing) to sets whose convex hull can be described?
3. Multi-row extensions?

Thank You!