Conversion Methods for Large Scale SDPs and Their Applications to Polynomial Optimization Problems

*Workshop: Advances in Mathematical Modeling and Computational Algorithms in Information Processing*

The Institute of Statistical Mathematics, Tokyo

November 1, 2008

Masakazu Kojima
Tokyo Institute of Technology
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —
2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion
   2-2. Conversion to a Primal form SDP with small mat. variables
   2-3. Conversion to an LMI form SDP with small mat. variables
3. Applications to SDP relaxation
   3-1. Sensor network localization problems
   3-2. Polynomial optimization problems
   3-3. Polynomial SDPs
4. Concluding remarks
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —

2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion
   2-2. Conversion to a Primal form SDP with small mat. variables
   2-3. Conversion to an LMI form SDP with small mat. variables

3. Correlative sparsity and sparsity of the Schur complement matrix in SDP with small mat. variables

3. Applications to SDP relaxation
   3-1. Sensor network localization problems
   3-2. Polynomial optimization problems
   3-3. Polynomial SDPs
Equality standard form SDP:
\[
\min \ A_0 \cdot X \ \text{sub.to} \ A_p \cdot X = b_p \ (p = 1, \ldots, m), \ S^n \ni X \succeq O
\]

\(A_p \in S^n\) the linear space of \(n \times n\) symmetric matrices

with the inner product \(A_p \cdot X = \sum_{i,j} [A_p]_{ij} X_{ij}\).

\(b_p \in \mathbb{R}, \ X \succeq O \iff X \in S^n\) is positive semidefinite.

Lots of Applications to Various Problems
- Systems and control theory — Linear Matrix Inequality
- SDP relaxations of combinatorial and nonconvex problems
  - Max cut and max clique problems
  - Quadratic assignment problems
  - Polynomial optimization problems — later
  - Polynomial semidefinite programs — later
- Robust optimization
- Quantum chemistry
- Moment problems (applied probability)
- Sensor network localization problem — later
- . . .
Equality standard form SDP:
\[
\min \ A_0 \cdot X \ \text{sub.to} \ A_p \cdot X = b_p \ (p = 1, \ldots, m), \ S^n \ni X \succeq 0
\]

SDP can be large-scale easily
- \( n \times n \) mat. variable \( X \) involves \( n(n+1)/2 \) real variables;
  \( n = 2000 \Rightarrow n(n+1)/2 \approx 2 \) million
- \( m \) linear equality constraints or \( m \ A_p \)'s \( \in S^n \)

◊ How can we solve a larger scale SDP?

(a) Use more powerful computer system such as clusters and grids of computers — parallel computation.
(b) Develop new numerical methods for SDPs.
(c) Improve primal-dual interior-point methods.
(d) Convert a large sparse SDP to an SDP which existing pdipms can solve efficiently:
  - multiple but small size mat. variables.
  - a sparse Schur complement mat. (a coef. mat. of a sys. of equations solved at \( \forall \) iteration of the pdipm).
### Outline of conversion methods

<table>
<thead>
<tr>
<th>Structured sparsity used</th>
<th>A large scale and structured sparse SDP</th>
<th>Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregated sparsity</td>
<td>↓</td>
<td>Positive semidefinite mat. completion</td>
</tr>
<tr>
<td>An SDP with small SDP cones and shared variables among SDP cones</td>
<td>↓</td>
<td>Conversion to Equality form SDP or conversion to LMI form SDP</td>
</tr>
<tr>
<td>An SDP with small mat. variables (i.e., small SDP cones)</td>
<td>↓</td>
<td></td>
</tr>
</tbody>
</table>
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —

2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion (Fukuda et al. ’01, Nakata et al. ’03)
   2-2. Conversion to a Primal form SDP with small mat. variables (Fukuda et al. ’01, Nakata et al. ’03)
   2-3. Conversion to an LMI form SDP with small mat. variables

3. Applications to SDP relaxation
   3-1. Sensor network localization problems
   3-2. Polynomial optimization problems
   3-3. Polynomial SDPs

4. Concluding remarks
Equality standard form SDP:
\[
\begin{align*}
\min & \quad A_0 \cdot X \quad \text{sub.to} \quad A_p \cdot X = b_p \quad (p = 1, \ldots, m), \\
& S^n \ni X \succeq O
\end{align*}
\]

\[
E_* = \{(i, j) : i = j \quad \text{or} \quad [A_p]_{ij} \neq 0 \quad \text{for} \quad \exists p = 0, \ldots, m\}
\]

\[
A_* : n \times n \quad \text{aggregated sparsity pattern mat.}
\]

\[
[A_*]_{ij} = \star \quad \text{if} \quad (i, j) \in E_* \quad \text{and} \quad 0 \quad \text{otherwise}
\]

SDP : a-sparse if \(A_*\) allows a sparse Cholesky factorization

Two typical cases: 1. bandwidth along diagonal

\[
A_* = \begin{pmatrix}
\star & \star & 0 & 0 & 0 \\
\star & \star & \star & 0 & 0 \\
0 & \star & \star & \star & 0 \\
0 & 0 & \star & \star & \star \\
0 & 0 & 0 & \star & \star
\end{pmatrix}
\]

\[
\begin{align*}
\min & \quad \sum_{(i, j) \in E_*} [A_0]_{ij} X_{ij} \\
\text{sub.to} & \quad \sum_{(i, j) \in E_*} [A_p]_{ij} X_{ij} = b_p \quad (\forall p)
\end{align*}
\]

\[
\begin{pmatrix}
X_{qq} & X_{q, q+1} \\
X_{q+1, q} & X_{q+1, q+1}
\end{pmatrix} \succeq O
\]

\(q = 1, \ldots, n - 1\).

SDP = SDP with shared variables among small SDP cones
Each \(\star\) can be a block matrix.
Equality standard form SDP:
\[
\min A_0 \cdot X \text{ sub.to } A_p \cdot X = b_p \ (p = 1, \ldots, m), \ S^n \ni X \succeq O
\]

\[
E_* = \{(i, j) : i = j \text{ or } [A_p]_{ij} \neq 0 \text{ for } \exists p = 0, \ldots, m\}
\]

\(A_* : n \times n\) aggregated sparsity pattern mat.

\[
[A_*]_{ij} = * \text{ if } (i, j) \in E_* \text{ and } 0 \text{ otherwise}
\]

SDP : a-sparse if \(A_*\) allows a sparse Cholesky factorization

Two typical cases: 2. arrow \(\downarrow\)

\[
A_* = \begin{pmatrix}
* & 0 & 0 & 0 & * \\
0 & * & 0 & 0 & * \\
0 & 0 & * & 0 & * \\
0 & 0 & 0 & * & * \\
* & * & * & * & *
\end{pmatrix}
\]

\[
\min \sum_{(i,j) \in E_*} [A_0]_{ij} X_{ij} \quad \text{sub.to} \quad \sum_{(i,j) \in E_*} [A_p]_{ij} X_{ij} = b_p \ (\forall p)
\]

\[
\begin{pmatrix}
X_{qq} & X_{qn} \\
X_{nq} & X_{nn}
\end{pmatrix} \succeq O
\]

\((q = 1, \ldots, n - 1)\).

SDP = SDP with shared variables among small SDP cones

Each * can be a block matrix.
Equality standard form SDP:
\[
\min \ A_0 \cdot X \ \text{sub.to} \ A_p \cdot X = b_p \ (p = 1, \ldots, m), \ S^n \ni X \succeq O
\]

\[
E_* = \{(i, j) : i = j \ \text{or} \ [A_p]_{ij} \neq 0 \ \text{for} \ \exists p = 0, \ldots, m\}
\]

\(A_* : n \times n \) aggregated sparsity pattern mat.

\[
[A_*]_{ij} = * \ \text{if} \ (i, j) \in E_* \ \text{and} \ 0 \ \text{otherwise}
\]

SDP : a-sparse if \(A_*\) allows a sparse Cholesky factorization

\[
\downarrow \ \text{positive semidefinite matrix completion}
\]

\[
\exists C_1, \ldots, C_\ell \subset N = \{1, 2, \ldots, n\}, \ \ell \leq n;
\]

SDP \equiv \text{an SDP with shared variables among small SDP cones:}

\[
\min \ \sum_{(i, j) \in E_*} [A_0]_{ij} X_{ij}
\]

\[
\text{s.t.} \ \sum_{(i, j) \in E_*} [A_p]_{ij} X_{ij} = b_p \ (\forall p), \ X(C_r) \succeq O \ (r = 1, \ldots, \ell),
\]

where \(X(C_r) : \text{the submatrix of } X \text{ consisting of } X_{ij} \ (i, j \in C_r)\).

\bullet \ \text{To solve SDP, we need to convert it into a standard form}

SDP \Rightarrow \text{next subject.}
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —

2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion (Fukuda et al. ’01, Nakata et al. ’03)
   2-2. Conversion to a Primal form SDP with small mat. variables (Fukuda et al. ’01, Nakata et al. ’03)
   2-3. Conversion to an LMI form SDP with small mat. variables

3. Applications to SDP relaxation
   3-1. Sensor network localization problems
   3-2. Polynomial optimization problems
   3-3. Polynomial SDPs

4. Concluding remarks
Equality standard form SDP:
\[ \min \ A_0 \cdot X \text{ sub.to } A_p \cdot X = b_p \ (p = 1, \ldots, m), \ S^n \ni X \succeq O \]

As an example: \[\downarrow\] aggregated sparsity
\[
\begin{align*}
\min & \sum_{(i,j) \in E_*} [A_0]_{ij} X_{ij} \\
\text{sub.to} \sum_{(i,j) \in E_*} [A_p]_{ij} X_{ij} = b_p \text{ and } \ \ X_{11} & \quad X_{12} \\
& \quad X_{22} \\
X_{21} & \quad X_{22}
\end{align*}
\]

(an SDP with smaller SDP cones and shared variables) \[\implies\]
Conversion into a standard form SDP to apply IPM — 2 ways

Primal form SDP with small mat. variables:
\[
\begin{align*}
\min & \ "\text{linear obj. in } Y_{ij}^r \text{s}" \text{ sub.to } "\text{linear eq. in } Y_{ij}^r \text{s}" \text{ and } \\
& \quad \begin{pmatrix} Y_{11}^1 & Y_{12}^1 \\
Y_{21}^1 & Y_{22}^1 \end{pmatrix}, \quad \begin{pmatrix} Y_{22}^2 & Y_{23}^2 & Y_{24}^2 \\
Y_{32}^2 & Y_{33}^2 & Y_{34}^2 \\
Y_{42}^2 & Y_{43}^2 & Y_{44}^2 \end{pmatrix}, \quad \begin{pmatrix} Y_{33}^3 & Y_{34}^3 & Y_{35}^3 \\
Y_{43}^3 & Y_{44}^3 & Y_{45}^3 \\
Y_{53}^3 & Y_{54}^3 & Y_{55}^3 \end{pmatrix} \succeq O,
\end{align*}
\]
\[
Y_{22}^1 = Y_{11}^2, \ Y_{22}^2 = Y_{11}^3, \ Y_{23}^2 = Y_{12}^3, \ Y_{33}^2 = Y_{22}^3.
\]
Equality standard form SDP:
\[ \min A_0 \cdot X \text{ sub.to } A_p \cdot X = b_p \ (p = 1, \ldots, m), \ S^n \ni X \succeq O \]

As an example: \( \downarrow \) aggregated sparsity

\[ \min \sum_{(i,j) \in E^*} [A_0]_{ij} X_{ij} \text{ sub.to } \sum_{(i,j) \in E^*} [A_p]_{ij} X_{ij} = b_p \text{ and } \]
\[ \begin{pmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{pmatrix}, \begin{pmatrix} X_{22} & X_{23} & X_{24} \\ X_{32} & X_{33} & X_{34} \\ X_{42} & X_{43} & X_{44} \end{pmatrix}, \begin{pmatrix} X_{33} & X_{34} & X_{35} \\ X_{43} & X_{44} & X_{45} \\ X_{53} & X_{54} & X_{55} \end{pmatrix} \preceq O \]

(an SDP with smaller SDP cones and shared variables) \( \implies \) Conversion into a standard form SDP to apply IPM — 2 ways

\( \downarrow \) LMI form \ SDP with small mat. variables \ — next Section

SDP with small (independent) matrix variables:
\[ \min \sum_{r=1}^{\ell} A_{0r} \cdot X_r \text{ sub.to } \sum_{r=1}^{\ell} A_{pr} \cdot X_r = b_p \ (p = 1, \ldots, m), \ X_r \succeq O \ (\forall r) \]

\( \bullet \) Further sparsity "\( A_{pr} \equiv O \) for many pairs of \( p \) and \( r \)" is often satisfied \( \Rightarrow \) correlative sparsity
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —
2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion
   2-2. Conversion to a Primal form SDP with small mat. variables
   2-3. Conversion to an LMI form SDP with small mat. variables
3. Applications to SDP relaxation
   3-1. Sensor network localization problems
   3-2. Polynomial optimization problems
   3-3. Polynomial SDPs
4. Concluding remarks
min \ \sum_{(i,j) \in E_*} [A_0]_{ij} X_{ij}

s.t. \ \sum_{(i,j) \in E_*} [A_p]_{ij} X_{ij} = b_p \ (\forall p), \ \mathbf{X}(C_r) \succeq \mathbf{O} \ (r = 1, \ldots, \ell),

where \mathbf{X}(C_r) : the\ submatrix\ of \mathbf{X}\ consisting\ of \ X_{ij} \ (i, j \in C_r).

Represent each \mathbf{X}(C_r) as

\mathbf{X}(C_r) = \sum_{i, j \in C_r, i \leq j} E_{ij}(C_r) X_{ij},

where \ E_{ij}(C_r) : a sym. mat. with 1 at some one or two elements and 0 elsewhere. For example,

\[
\begin{pmatrix}
X_{11} & X_{13} \\
X_{31} & X_{33}
\end{pmatrix} = \begin{pmatrix}
1 & 0 \\
0 & 1
\end{pmatrix} X_{11} + \begin{pmatrix}
0 & 1 \\
1 & 0
\end{pmatrix} X_{12} + \begin{pmatrix}
0 & 0 \\
0 & 1
\end{pmatrix} X_{33}
\]

Then, an LMI form SDP having eq. const.

\[
\begin{align*}
\min \ & \sum_{(i,j) \in E_*} [A_0]_{ij} X_{ij} \\
\text{sub.to} \ & \sum_{(i,j) \in E_*} [A_p]_{ij} X_{ij} = b_p \ (\forall p), \\
\ & \sum_{i, j \in C_r, i \leq j} E_{ij}(C_r) X_{ij} \succeq \mathbf{O} \ (\forall r).
\end{align*}
\]
## Review of conversion methods

<table>
<thead>
<tr>
<th>Structured sparsity used</th>
<th>A large scale and structured sparse SDP</th>
<th>Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregated sparsity</td>
<td>↓</td>
<td>Positive semidefinite mat. completion</td>
</tr>
<tr>
<td></td>
<td>An SDP with small SDP cones and shared variables among SDP cones</td>
<td>Conversion to Equality form SDP or conversion to LMI form SDP</td>
</tr>
<tr>
<td></td>
<td>↓</td>
<td>↓</td>
</tr>
<tr>
<td></td>
<td>An SDP with small mat. variables (i.e., small SDP cones)</td>
<td></td>
</tr>
</tbody>
</table>
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —

2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion
   2-2. Conversion to a Primal form SDP with small mat. variables
   2-3. Conversion to an LMI form SDP with small mat. variables

3. Applications to SDP relaxation
   3-1. Sensor network localization problems (Kim,Kojima,Waki)
   3-2. Polynomial optimization problems
   3-3. Polynomial SDPs

4. Concluding remarks
Sensor network localization problem: Let $s = 2$ or $3$.

$x^p \in \mathbb{R}^s : \text{unknown location of sensors} \ (p = 1, 2, \ldots, m),$

$x^r = a^r \in \mathbb{R}^s : \text{known location of anchors} \ (r = m + 1, \ldots, n),$

$d_{pq} = \|x^p - x^q\| + \epsilon_{pq} \quad \text{— given for} \ (p, q) \in \mathcal{N},$

$\mathcal{N} = \{(p, q) : \|x^p - x^q\| \leq \rho = \text{a given radio range}\}$

Here $\epsilon_{pq}$ denotes a noise.

$m = 5, \ n = 9.$

1, \ldots, 5: \text{sensors}\n
6, 7, 8, 9: \text{anchors}\n
\begin{itemize}
  \item Anchors’ positions are known.
  \item A distance is given for $\forall$ edge.
  \item Compute locations of sensors.
\end{itemize}

$\Rightarrow$ Some nonconvex QOPs

- SDP relaxation — FSDP by Biswas-Ye ’06, ESDP by Wang et al ’07, ... for $s = 2$.
- SOCP relaxation — Tseng ’07 for $s = 2$.

...
Numerical results on 4 methods (a), (b), (c) and (d) applied to a sensor network localization problem with

\[ m = \text{the number of sensors dist. randomly in } [0, 1]^2, \]
\[ 4 \text{ anchors located at the corner of } [0, 1]^2, \]
\[ \rho = \text{radio distance } = 0.1, \text{ no noise.} \]

(a) FSDP
(b) FSDP + Conv. to LMI form SDP, as strong as (a)
(c) FSDP + Conv. to Equality form SDP as strong as (a)

<table>
<thead>
<tr>
<th>m</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>389.1</td>
<td>35.0</td>
<td>69.5</td>
</tr>
<tr>
<td>1000</td>
<td>3345.2</td>
<td>60.4</td>
<td>178.8</td>
</tr>
<tr>
<td>2000</td>
<td>111.1</td>
<td>326.0</td>
<td></td>
</tr>
<tr>
<td>4000</td>
<td>182.1</td>
<td>761.0</td>
<td></td>
</tr>
</tbody>
</table>

Cholesky factor of aggregated sparsity pattern ⇒ next slide
This aggregated sparsity pattern is exploited in

(b) **FSDP + Conv. to LMI form SDP** — cpu time 60.4 sec

(c) **FSDP + Conv. to Equality form SDP** — cpu time 178.8 sec
(b) FSDP + Conv. to LMI form SDP — cpu time 60.4 sec
(c) FSDP + Conv. to Equality form SDP — cpu time 178.8 sec
3 dim, 500 sensors, 27 anchors, r.range = 0.3, noise ← N(0,0.1);
(estimated dist.) \( \hat{d}_{pq} = (1 + \epsilon_{pq})d_{pq} \) (true unknown dist.),
\( \epsilon_{pq} ← N(0, 0.1) \)

(b) FSDP + Conv. to LMI form SDP

![Diagram showing anchor, true, computed, and deviation points with 24.2 sec. annotation.]
3 dim, 500 sensors, 27 anchors, r.range = 0.3, noise ← N(0,0.1);

(estimated dist.) \( \hat{d}_{pq} = (1 + \epsilon_{pq})d_{pq} \) (true unknown dist.),

\( \epsilon_{pq} ← N(0, 0.1) \)

(b) FSDP + Conv. to LMI form SDP + Gradient method

anchor : ♦
true : ○
computed : ∗
deviation : —
24.2 sec. + 8.4 sec.
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —

2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion
   2-2. Conversion to a Primal form SDP with small mat. variables
   2-3. Conversion to an LMI form SDP with small mat. variables

3. Applications to SDP relaxation
   3-1. Sensor network localization problems
   3-2. Polynomial optimization problems (Kim, Kojima, Muramatsu, Waki)
   3-3. Polynomial SDPs

4. Concluding remarks
POP (Polynomial Optimization Problem)

\[
\begin{align*}
\min & \quad f_0(x) \\
\text{sub.to} & \quad f_i(x) \geq 0 \quad (i = 1, 2, \ldots, m).
\end{align*}
\]

Here \( f_p(x) \) denotes a polynomial in \( x = (x_1, \ldots, x_n) \).

(a) Apply SDP relaxation to POP \( \Rightarrow \) SDP

\[ \text{— SparsePOP(MATLAB)} \]

(b) Convert SDP into LMI form SDP with small mat. variables

\[ \text{— SparsePOP(MATLAB)} \]

(c) Solve LMI form SDP by the primal-dual interior-point method

\[ \text{— SeDuMi(MATLAB)} \]

- SDP could become large-scale even when POP is small (say \( n = 20, m = 20 \)).
- Sparsity is exploited in (a) too.
- Both lower and upper bounds for the optimal value are obtained.
A POP alkyl from globalib

\[
\begin{align*}
\text{min} & \quad -6.3x_5x_8 + 5.04x_2 + 0.35x_3 + x_4 + 3.36x_6 \\
\text{sub.to} & \quad -0.820x_2 + x_5 - 0.820x_6 = 0, \\
& \quad 0.98x_4 - x_7(0.01x_5x_{10} + x_4) = 0, \quad -x_2x_9 + 10x_3 + x_6 = 0, \\
& \quad x_5x_{12} - x_2(1.12 + 0.132x_9 - 0.0067x_9^2) = 0, \\
& \quad x_8x_{13} - 0.01x_9(1.098 - 0.038x_9) - 0.325x_7 = 0.574, \\
& \quad x_{10}x_{14} + 22.2x_{11} = 35.82, \quad x_1x_{11} - 3x_8 = -1.33, \\
\end{align*}
\]

\[lbd_i \leq x_i \leq ubd_i \quad (i = 1, 2, \ldots, 14).\]

- 14 variables, 7 poly. equality constraints with deg. 3.

<table>
<thead>
<tr>
<th>Sparse+Conversion</th>
<th>Dense (Lasserre)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon_{\text{obj}}$</td>
<td>$\epsilon_{\text{obj}}$</td>
</tr>
<tr>
<td>$\epsilon_{\text{feas}}$</td>
<td>$\epsilon_{\text{feas}}$</td>
</tr>
<tr>
<td>cpu</td>
<td>cpu</td>
</tr>
<tr>
<td>5.6e-10</td>
<td>2.0e-08</td>
</tr>
</tbody>
</table>

$\epsilon_{\text{obj}} = \text{approx. opt. val.} - \text{lower bound for opt. val.}$

$\epsilon_{\text{feas}} = \text{the maximum error in the equality constraints}$

- Global optimality is guaranteed with high accuracy.
A POP ex2_1_8 from globalib

\[ \begin{align*}
\min & \quad \text{nonconvex diag. quad. funct.} + \text{linear funct.} \\
\text{sub.to} & \quad 10 \text{ sparse linear equalities} \\
& \quad \text{lbd}_i \leq x_i \leq \text{ubd}_i \quad (i = 1, 2, \ldots, 24).
\end{align*} \]

<table>
<thead>
<tr>
<th>Sparse+Conversion</th>
<th>Dense (Lasserre)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \epsilon_{\text{obj}} )</td>
<td>( \epsilon_{\text{obj}} )</td>
</tr>
<tr>
<td>( \epsilon_{\text{feas}} )</td>
<td>( \epsilon_{\text{feas}} )</td>
</tr>
<tr>
<td>cpu</td>
<td>cpu</td>
</tr>
<tr>
<td>5.0e-9</td>
<td>5.8e-10</td>
</tr>
<tr>
<td>1.3e-11</td>
<td>3.0e-12</td>
</tr>
<tr>
<td>20.0</td>
<td>288.8</td>
</tr>
</tbody>
</table>

\( \epsilon_{\text{obj}} \) = \text{approx. opt.val.} - \text{lower bound for opt.val.}

\( \epsilon_{\text{feas}} \) = \text{the maximum error in the equality constraints}

Global optimality is guaranteed with high accuracy.
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —

2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion
   2-2. Conversion to a Primal form SDP with small mat. variables
   2-3. Conversion to an LMI form SDP with small mat. variables

3. Applications to SDP relaxation
   3-1 Sensor network localization problems
   3-2. Polynomial optimization problems
   3-3. Polynomial SDPs

4. Concluding remarks
**SDP O (polynomial SDP):** \(\min f_0(x) \text{ sub.to } F(x) \succeq O.\)

\[
f_0(x) : \text{ a polynomial in } x \in \mathbb{R}^m
\]

\[
F : \mathbb{R}^m \to S^n, \quad F_{ij}(x) : \text{ a polynomial in } x \in \mathbb{R}^m
\]

\[
A_* : \text{ the sparsity pattern matrix;}
\]

\[
[A_*]_{ij} = 0 \text{ if } F_{ij}(x) \equiv 0, [A_*]_{ij} = * \text{ otherwise}
\]

\[\Downarrow\]

**Assumption.** \(A_*\) allows a sparse Cholesky factorization.  

**Positive semidefinite matrix completion technique**

**SDP C (poly. SDP with multiple but smaller SDP cones):**

\[
\min f_0(x) \text{ sub.to } F_p(x) + \sum_{k=1}^{\ell} B_{pk} z_k \succeq O \quad (p = 1, \ldots, \ell).
\]

\[
F_p : \mathbb{R}^m \to S^{n_p}, \quad B_{pk} \in S^{n_p}.
\]

\(n_p < < n\) under Assumption.
SDP O (tridiag. quad. SDP): \( \min \sum_{i=1}^{n} c_i x_i \) sub.to \( F(x) \succeq O \).

\[ F : \mathbb{R}^n \to S^n, \text{ each element } F_{ij} \text{ is quadratic or linear; } \]

\[ F_{ij}(x) = \begin{cases} 
  d_i - x_i^2 & \text{if } i = j, \\
  (a_i - 0.5)x_i + (b_i - 0.5)x_{i+1} & \text{if } i \leq n - 1, j = i + 1, \\
  (a_j - 0.5)x_j + (b_j - 0.5)x_{j+1} & \text{if } j \leq n - 1, i = j + 1, \\
  0 & \text{otherwise.} 
\end{cases} \]

All \( a_i, b_i, c_i, d_i \) are chosen randomly from \([0, 1]\).

\( \Downarrow \)

the sparsity p. mat. \( A_\ast \) — tridiagonal \( \Rightarrow \) sparse Cholesky fact.

SDP C (quad. SDP with multiple but smaller SDP cones):

\[ \min \sum_{i=1}^{n} c_i x_i \text{ sub.to } F_p(x) + \sum_{k=1}^{n-1} B_{pk} z_k \succeq O \ (p = 1, \ldots, n - 1). \]

\[ F_p : \mathbb{R}^m \to S^2, \ B_{pk} \in S^2 \]

We will apply a (linear) SDP relaxation for poly. SDP to SDP O and SDP C, and compare their numerical results.
SDP O (tridiag. quad. SDP): min \( \sum_{i=1}^{n} c_i x_i \) sub.to \( F(x) \succeq O \).

\[
F : \mathbb{R}^n \rightarrow S^n, \text{ each element } F_{ij} \text{ is quadratic or linear;}
\]

\[
F_{ij}(x) = \begin{cases} 
    d_i - x_i^2 & \text{if } i = j, \\
    (a_i - 0.5)x_i + (b_i - 0.5)x_{i+1} & \text{if } i \leq n - 1, j = i + 1, \\
    (a_j - 0.5)x_j + (b_j - 0.5)x_{j+1} & \text{if } j \leq n - 1, i = j + 1, \\
    0 & \text{otherwise.}
\end{cases}
\]

All \( a_i, b_i, c_i, d_i \) are chosen randomly from \([0, 1]\).

<table>
<thead>
<tr>
<th></th>
<th>SDP O, no conversion</th>
<th>SDP C, conversion</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>sizeA</td>
<td>cpu</td>
</tr>
<tr>
<td>50</td>
<td>1325 × 5101</td>
<td>28.74</td>
</tr>
<tr>
<td>100</td>
<td>5150 × 20201</td>
<td>2874.45</td>
</tr>
<tr>
<td>200</td>
<td>797 × 2587</td>
<td>1.38</td>
</tr>
<tr>
<td>400</td>
<td>3197 × 10387</td>
<td>6.29</td>
</tr>
<tr>
<td>800</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Contents

1. Introduction
   — Semidefinite Programs (SDPs) and their conversion —

2. Conversion methods
   2-1. Aggregated sparsity and positive semidefinite matrix completion
   2-2. Conversion to a Primal form SDP with small mat. variables
   2-3. Conversion to an LMI form SDP with small mat. variables

3. Applications to SDP relaxation
   3-1. Polynomial optimization problems
   3-2. Sensor network localization problems
   3-3. Polynomial SDPs

4. Concluding remarks
1. Conversion of a large scale SDP into an SDP having small matrix variables

2. Two different methods:
   - Conversion to Equality form SDP
   - Conversion to LMI form SDP

3. Some applications to SDP relaxation and successful numerical results

4. In general, it is often difficult to solve SDPs arising from SDP relaxation of POPs and polynomial SDPs; too large to solve, numerical difficulty.

Thank you!