On Solving SOCPs
with an
Interior-Point Method for NLP

Robert Vanderbei
and
Hande Y. Benson

INFORMS Nov 5, 2001
Miami Beach, FL

Operations Research and Financial Engineering, Princeton University
http://www.princeton.edu/~rvdb
1 Outline

- Introduction and Review of Problem Classes: NLP, SOCP
- Formulating SOCPs as Smooth Convex NLPs
- Applications and Computational Results

Theme. Modeling Matters.
1 Outline

- Introduction and Review of Problem Classes: NLP, SOCP
- Formulating SOCPs as Smooth Convex NLPs
- Applications and Computational Results

Theme. Modeling Matters.
1 Outline

- Introduction and Review of Problem Classes: NLP, SOCP
- Formulating SOCPs as Smooth Convex NLPs
- Applications and Computational Results

Theme. Modeling Matters.
1 Outline

- Introduction and Review of Problem Classes: NLP, SOCP
- Formulating SOCPs as Smooth Convex NLPs
- Applications and Computational Results

Theme. Modeling Matters.
1 Outline

- Introduction and Review of Problem Classes: NLP, SOCP
- Formulating SOCPs as Smooth Convex NLPs
- Applications and Computational Results

Theme: Modeling Matters.
2 Traditional Classes of Optimization Problems

Smooth Convex Nonlinear Programming (NLP)

minimize \( f(x) \)
subject to \( h_i(x) = 0, \quad i \in \mathcal{E}, \)
\( h_i(x) \geq 0, \quad i \in \mathcal{I}. \)

We assume that

- \( h_i \)'s in equality constraints are affine;
- \( h_i \)'s in inequality constraints are concave;
- \( f \) is convex;
- All are twice continuously differentiable.
2 Traditional Classes of Optimization Problems

Smooth Convex Nonlinear Programming (NLP)

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) = 0, \quad i \in \mathcal{E}, \\
& \quad h_i(x) \geq 0, \quad i \in \mathcal{I}.
\end{align*}
\]

We assume that

- $h_i$’s in equality constraints are affine;
- $h_i$’s in inequality constraints are concave;
- $f$ is convex;
- All are twice continuously differentiable.
2 Traditional Classes of Optimization Problems

Smooth Convex Nonlinear Programming (NLP)

\[
\begin{aligned}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) = 0, \quad i \in \mathcal{E}, \\
& \quad h_i(x) \geq 0, \quad i \in \mathcal{I}.
\end{aligned}
\]

We assume that

- \( h_i \)'s in equality constraints are affine;
- \( h_i \)'s in inequality constraints are concave;
- \( f \) is convex;
- All are twice continuously differentiable.
2 Traditional Classes of Optimization Problems

Smooth Convex Nonlinear Programming (NLP)

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) = 0, \quad i \in \mathcal{E}, \\
& \quad h_i(x) \geq 0, \quad i \in \mathcal{I}.
\end{align*}
\]

We assume that

- \( h_i \)'s in equality constraints are affine;
- \( h_i \)'s in inequality constraints are concave;
- \( f \) is convex;
- All are twice continuously differentiable.
2 Traditional Classes of Optimization Problems

Smooth Convex Nonlinear Programming (NLP)

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) = 0, \quad i \in \mathcal{E}, \\
& \quad h_i(x) \geq 0, \quad i \in \mathcal{I}.
\end{align*}
\]

We assume that

- \( h_i \)'s in equality constraints are affine;
- \( h_i \)'s in inequality constraints are concave;
- \( f \) is convex;
- All are twice continuously differentiable.
3 Second-Order Cone Programming (SOCP)

\[
\begin{align*}
\text{minimize} & \quad f^T x \\
\text{subject to} & \quad \|A_i x + b_i\| \leq c_i^T x + d_i, \quad i = 1, \ldots, m,
\end{align*}
\]

Here,

- \( f \) is an \( n \)-vector,
- \( A_i \) is a \( k_i \times n \) matrix,
- \( b_i \) is a \( k_i \)-vector,
- \( c_i \) is an \( n \)-vector, and
- \( d_i \) is a scalar.
3 Second-Order Cone Programming (SOCP)

\[
\begin{align*}
\text{minimize} & \quad f^T x \\
\text{subject to} & \quad \|A_i x + b_i\| \leq c_i^T x + d_i, \quad i = 1, \ldots, m,
\end{align*}
\]

Here,

- \( f \) is an \( n \)-vector,
- \( A_i \) is a \( k_i \times n \) matrix,
- \( b_i \) is a \( k_i \)-vector,
- \( c_i \) is an \( n \)-vector, and
- \( d_i \) is a scalar.
4 Preview of Applications

- FIR Filter Design
- Antenna Array Optimization
- Structural Optimization
- Grasping Problems
- Steiner Tree Problem
- Euclidean Multiple Facility Location
- Plastic Deformation
- Springs in Equilibrium
- Markowitz models in Finance

More later on some of these.
4 Preview of Applications

- FIR Filter Design
- Antenna Array Optimization
- Structural Optimization
- Grasping Problems
- Steiner Tree Problem
- Euclidean Multiple Facility Location
- Plastic Deformation
- Springs in Equilibrium
- Markowitz models in Finance

More later on some of these.
5 Interior-Point Algorithms

- Interior-point methods were first developed in the mid 80's for LP.

- Later they were extended to NLP, SOCP, and SDP.

- Extension to NLP follows closely the LP case. That is, $\geq$ is treated the same in both cases. The nonnegative-orthant cone, $x \geq 0$, plays a fundamental role.

- For SOCP, a different cone is introduced, the Lorentz cone, and algorithms are derived using this cone in place of the nonnegative orthant cone.
5 Interior-Point Algorithms

- Interior-point methods were first developed in the mid 80's for LP.

- Later they were extended to NLP, SOCP, and SDP.

- Extension to NLP follows closely the LP case. That is, $\geq$ is treated the same in both cases. The nonnegative-orthant cone, $x \geq 0$, plays a fundamental role.

- For SOCP, a different cone is introduced, the Lorentz cone, and algorithms are derived using this cone in place of the nonnegative orthant cone.
5 Interior-Point Algorithms

- Interior-point methods were first developed in the mid 80’s for LP.

- Later they were extended to NLP, SOCP, and SDP.

- Extension to NLP follows closely the LP case. That is, $\geq$ is treated the same in both cases. The nonnegative-orthant cone, $x \geq 0$, plays a fundamental role.

- For SOCP, a different cone is introduced, the Lorentz cone, and algorithms are derived using this cone in place of the nonnegative orthant cone.
5 Interior-Point Algorithms

- Interior-point methods were first developed in the mid 80’s for LP.

- Later they were extended to NLP, SOCP, and SDP.

- Extension to NLP follows closely the LP case. That is, $\geq$ is treated the same in both cases. The nonnegative-orthant cone, $x \geq 0$, plays a fundamental role.

- For SOCP, a different cone is introduced, the Lorentz cone, and algorithms are derived using this cone in place of the nonnegative orthant cone.
5 Interior-Point Algorithms

- Interior-point methods were first developed in the mid 80’s for LP.

- Later they were extended to NLP, SOCP, and SDP.

- Extension to NLP follows closely the LP case. That is, $\geq$ is treated the same in both cases. The nonnegative-orthant cone, $x \geq 0$, plays a fundamental role.

- For SOCP, a different cone is introduced, the Lorentz cone, and algorithms are derived using this cone in place of the nonnegative orthant cone.
Recently, SOCP and SDP have been unified under the banner of Conic Programming and software has appeared to solve problems from the union of the SOCP and SDP problem classes.

The aim of our work is to show that SOCP and SDP can be included under the banner of NLP and solved with “generic” NLP software. This is a much more general setting.
Recently, SOCP and SDP have been unified under the banner of Conic Programming and software has appeared to solve problems from the union of the SOCP and SDP problem classes.

The aim of our work is to show that SOCP and SDP can be included under the banner of NLP and solved with “generic” NLP software. This is a much more general setting.
Recently, SOCP and SDP have been unified under the banner of Conic Programming and software has appeared to solve problems from the union of the SOCP and SDP problem classes.

The aim of our work is to show that SOCP and SDP can be included under the banner of NLP and solved with “generic” NLP software. This is a much more general setting.
Formulating SOCPs as Smooth Convex NLPs
For SOCP, 

\[ h_i(x) = c_i^T x + d_i - \|A_i x + b_i\| \]

is concave but not differentiable on 

\[ \{ x : A_i x + b_i = 0 \} . \]

Nondifferentiability should not be a problem unless it happens at optimality...
For SOCP,

\[ h_i(x) = c_i^T x + d_i - \|A_i x + b_i\| \]

is concave but not differentiable on

\[ \{x : A_i x + b_i = 0\} . \]

Nondifferentiability should not be a problem unless it happens at optimality...
8 SOCP as NLP

For SOCP,

\[ h_i(x) = c_i^T x + d_i - \| A_i x + b_i \| \]

is concave but not differentiable on

\[ \{ x : A_i x + b_i = 0 \} . \]

Nondifferentiability should not be a problem unless it happens at optimality...
For SOCP,

\[ h_i(x) = c_i^T x + d_i - \| A_i x + b_i \| \]

is concave but not differentiable on

\[ \{ x : A_i x + b_i = 0 \} . \]

Nondifferentiability should not be a problem unless it happens at optimality...
An Example

\[
\begin{align*}
\text{minimize} & \quad ax_1 + x_2 & \quad (-1 < a < 1) \\
\text{subject to} & \quad |x_1| \leq x_2,
\end{align*}
\]

Clearly, \((x_1^*, x_2^*) = (0, 0)\).

Dual feasibility:

\[
\begin{bmatrix}
 a \\
 1
\end{bmatrix} + \begin{bmatrix}
 \frac{d|x_1|}{dx_1} \\
 -1
\end{bmatrix} y = 0.
\]

An interior-point method must pick the correct value for \(\frac{d|x_1|}{dx_1}\) when \(x_1 = 0\):

\[
\left.\frac{d|x_1|}{dx_1}\right|_{x_1=0} = -a.
\]

Not possible \textit{a priori}.
9 An Example

minimize \( ax_1 + x_2 \quad (-1 < a < 1) \)
subject to \( |x_1| \leq x_2, \)

Clearly, \((x_1^*, x_2^*) = (0, 0)\).

Dual feasibility:

\[
\begin{bmatrix}
a \\
1
\end{bmatrix} + \begin{bmatrix}
\frac{d|x_1|}{dx_1} \\
-1
\end{bmatrix} y = 0.
\]

An interior-point method must pick the correct value for \( \frac{d|x_1|}{dx_1} \) when \( x_1 = 0 \):

\[
\left. \frac{d|x_1|}{dx_1} \right|_{x_1=0} = -a.
\]

Not possible \text{ a priori}. 

Smooth Alternative Formulations

Constraint formulation:

\[ \phi(A_i x + b_i, c_i^T x + d_i) \geq 0, \quad i = 1, \ldots, m \]

where

\[ \phi(u, t) = t - \|u\|. \]

Not differentiable at \( u = 0 \).

Smooth alternatives:

\[ t - \sqrt{\epsilon^2 + \sum_i u_i^2} \geq 0, \quad \text{concave} \quad \text{not equiv.} \]

\[ t^2 - \|u\|^2 \geq 0, \quad t \geq 0 \quad \text{nonconcave} \quad \text{equiv.} \]

\[ t - \|u\|^2/t \geq 0, \quad t > 0 \quad \text{concave} \quad \text{equiv.} \quad \text{interior} \]
10 Smooth Alternative Formulations

Constraint formulation:

\[
\phi(A_i x + b_i, c_i^T x + d_i) \geq 0, \quad i = 1, \ldots, m
\]

where

\[
\phi(u, t) = t - \|u\|.
\]

Not differentiable at \( u = 0 \).

Smooth alternatives:

\[
t - \sqrt{\epsilon^2 + \sum_i u_i^2} \geq 0, \quad \text{concave} \quad \text{not equiv.}
\]

\[
t^2 - \|u\|^2 \geq 0, \quad t \geq 0 \quad \text{nonconcave} \quad \text{equiv.}
\]

\[
t - \|u\|^2/t \geq 0, \quad t > 0 \quad \text{concave} \quad \text{equiv.} \quad \text{interior}
\]
Smooth Alternative Formulations

Constraint formulation:

\[ \phi(A_i x + b_i, c_i^T x + d_i) \geq 0, \quad i = 1, \ldots, m \]

where

\[ \phi(u, t) = t - \|u\|. \]

Not differentiable at \( u = 0 \).

Smooth alternatives:

\[ t - \sqrt{\epsilon^2 + \sum_i u_i^2} \geq 0, \quad \text{concave} \quad \text{not equiv.} \]

\[ t^2 - \|u\|^2 \geq 0, \quad t \geq 0 \quad \text{nonconcave} \quad \text{equiv.} \]

\[ t - \frac{\|u\|^2}{t} \geq 0, \quad t > 0 \quad \text{concave} \quad \text{equiv.} \quad \text{interior} \]
Smooth Alternative Formulations

Constraint formulation:

\[ \phi(A_i x + b_i, c_i^T x + d_i) \geq 0, \quad i = 1, \ldots, m \]

where \( \phi(u, t) = t - \|u\| \).

Not differentiable at \( u = 0 \).

Smooth alternatives:

\[ t - \sqrt{e^2 + \sum_i u_i^2} \geq 0, \quad \text{concave} \quad \text{not equiv.} \]

\[ t^2 - \|u\|^2 \geq 0, \quad t \geq 0 \quad \text{nonconcave} \quad \text{equiv.} \]

\[ t - \|u\|^2 / t \geq 0, \quad t > 0 \quad \text{concave} \quad \text{equiv.} \quad \text{interior} \]
10 Smooth Alternative Formulations

Constraint formulation:

\[ \phi(A_i x + b_i, c_i^T x + d_i) \geq 0, \quad i = 1, \ldots, m \]

where

\[ \phi(u, t) = t - \|u\|. \]

Not differentiable at \( u = 0 \).

Smooth alternatives:

\[ t - \sqrt{\epsilon^2 + \sum_i u_i^2} \geq 0, \quad \text{concave} \quad \text{not equiv.} \]

\[ t^2 - \|u\|^2 \geq 0, \quad t \geq 0 \quad \text{nonconcave} \quad \text{equiv.} \]

\[ t - \|u\|^2/t \geq 0, \quad t > 0 \quad \text{concave} \quad \text{equiv.} \quad \text{interior} \]
Smooth Alternative Formulations

Constraint formulation:

$$
\phi(A_i x + b_i, c_i^T x + d_i) \geq 0, \quad i = 1, \ldots, m
$$

where

$$
\phi(u, t) = t - \|u\|.
$$

Not differentiable at $u = 0$.

Smooth alternatives:

$$
t - \sqrt{\epsilon^2 + \sum_i u_i^2} \geq 0, \quad \text{concave not equiv.}
$$

$$
t^2 - \|u\|^2 \geq 0, \quad t \geq 0 \quad \text{nonconcave equiv.}
$$

$$
t - \|u\|^2 / t \geq 0, \quad t > 0 \quad \text{concave equiv. interior}
$$
### Which Formulation is Best?

<table>
<thead>
<tr>
<th>Problem</th>
<th>socp</th>
<th>$\epsilon$-pert</th>
<th>nonconvex</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>antenna [6]</td>
<td>5.728 (40)</td>
<td>9.664 (60)</td>
<td>34.169 (201)</td>
<td>3.365 (28)</td>
</tr>
<tr>
<td>emfl</td>
<td>318.869 (201)</td>
<td>29.332 (24)</td>
<td>277.499 (201)</td>
<td>61.739 (30)</td>
</tr>
<tr>
<td>fir</td>
<td>0.42 (37)</td>
<td>0.42 (37)</td>
<td>0.47 (42)</td>
<td>0.46 (40)</td>
</tr>
<tr>
<td>grasp</td>
<td>0.31 (201)</td>
<td>0.04 (37)</td>
<td>0.34 (201)</td>
<td>0.04 (29)</td>
</tr>
<tr>
<td>minsurf</td>
<td>14.511 (34)</td>
<td>7.811 (20)</td>
<td>109.948 (201)</td>
<td>7.28 (17)</td>
</tr>
<tr>
<td>springs</td>
<td>0.01 (17)</td>
<td>0.02 (17)</td>
<td>0.01 (15)</td>
<td>0.01 (15)</td>
</tr>
<tr>
<td>steiner</td>
<td>0.861 (201)</td>
<td>0.11 (27)</td>
<td>0.811 (201)</td>
<td>0.16 (41)</td>
</tr>
<tr>
<td>structure</td>
<td>68.829 (201)</td>
<td>10.645 (43)</td>
<td>81.186 (201)</td>
<td>12.898 (54)</td>
</tr>
</tbody>
</table>

- These problems are AMPL encodings of problems in Lobo, Vandenberghe, Boyd, and Lebret.
- “Nonconvex” refers to the “square-both-sides” reformulation.
- Numbers in parens are iteration counts.
- Red indicates failure.
### Which Formulation is Best?

<table>
<thead>
<tr>
<th>Problem</th>
<th>socp</th>
<th>$\epsilon$-pert</th>
<th>nonconvex</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>antenna [6]</td>
<td>5.728 (40)</td>
<td>9.664 (60)</td>
<td>34.169 (201)</td>
<td>3.365 (28)</td>
</tr>
<tr>
<td>emfl</td>
<td>318.869 (201)</td>
<td>29.332 (24)</td>
<td>277.499 (201)</td>
<td>61.739 (30)</td>
</tr>
<tr>
<td>fir</td>
<td>0.42 (37)</td>
<td>0.42 (37)</td>
<td>0.47 (42)</td>
<td>0.46 (40)</td>
</tr>
<tr>
<td>grasp</td>
<td>0.31 (201)</td>
<td>0.04 (37)</td>
<td>0.34 (201)</td>
<td>0.04 (29)</td>
</tr>
<tr>
<td>minsurf</td>
<td>14.511 (34)</td>
<td>7.811 (20)</td>
<td>109.948 (201)</td>
<td>7.28 (17)</td>
</tr>
<tr>
<td>springs</td>
<td>0.01 (17)</td>
<td>0.02 (17)</td>
<td>0.01 (15)</td>
<td>0.01 (15)</td>
</tr>
<tr>
<td>steiner</td>
<td>0.861 (201)</td>
<td>0.11 (27)</td>
<td>0.811 (201)</td>
<td>0.16 (41)</td>
</tr>
<tr>
<td>structure</td>
<td>68.829 (201)</td>
<td>10.645 (43)</td>
<td>81.186 (201)</td>
<td>12.898 (54)</td>
</tr>
</tbody>
</table>

- These problems are AMPL encodings of problems in Lobo, Vandenberghe, Boyd, and Lebret.
- “Nonconvex” refers to the “square-both-sides” reformulation.
- Numbers in parens are iteration counts.
- Red indicates failure.
## Which Formulation is Best?

<table>
<thead>
<tr>
<th>Problem</th>
<th>socp</th>
<th>$\epsilon$-pert</th>
<th>nonconvex</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>antenna [6]</td>
<td>5.728 (40)</td>
<td>9.664 (60)</td>
<td>34.169 (201)</td>
<td>3.365 (28)</td>
</tr>
<tr>
<td>emfl</td>
<td>318.869 (201)</td>
<td>29.332 (24)</td>
<td>277.499 (201)</td>
<td>61.739 (30)</td>
</tr>
<tr>
<td>fir</td>
<td>0.42 (37)</td>
<td>0.42 (37)</td>
<td>0.47 (42)</td>
<td>0.46 (40)</td>
</tr>
<tr>
<td>grasp</td>
<td>0.31 (201)</td>
<td>0.04 (37)</td>
<td>0.34 (201)</td>
<td>0.04 (29)</td>
</tr>
<tr>
<td>mnsurf</td>
<td>14.511 (34)</td>
<td>7.811 (20)</td>
<td>109.948 (201)</td>
<td>7.28 (17)</td>
</tr>
<tr>
<td>springs</td>
<td>0.01 (17)</td>
<td>0.02 (17)</td>
<td>0.01 (15)</td>
<td>0.01 (15)</td>
</tr>
<tr>
<td>steiner</td>
<td>0.861 (201)</td>
<td>0.11 (27)</td>
<td>0.811 (201)</td>
<td>0.16 (41)</td>
</tr>
<tr>
<td>structure</td>
<td>68.829 (201)</td>
<td>10.645 (43)</td>
<td>81.186 (201)</td>
<td>12.898 (54)</td>
</tr>
</tbody>
</table>

- These problems are AMPL encodings of problems in Lobo, Vandenberghe, Boyd, and Lebret.
- “Nonconvex” refers to the “square-both-sides” reformulation.
- Numbers in parens are iteration counts.
- Red indicates failure.
### Which Formulation is Best?

<table>
<thead>
<tr>
<th>Problem</th>
<th>socp</th>
<th>$\epsilon$-pert</th>
<th>nonconvex</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>antenna [6]</td>
<td>5.728 (40)</td>
<td>9.664 (60)</td>
<td>34.169 (201)</td>
<td>3.365 (28)</td>
</tr>
<tr>
<td>emfl</td>
<td>318.869 (201)</td>
<td>29.332 (24)</td>
<td>277.499 (201)</td>
<td>61.739 (30)</td>
</tr>
<tr>
<td>fir</td>
<td>0.42 (37)</td>
<td>0.42 (37)</td>
<td>0.47 (42)</td>
<td>0.46 (40)</td>
</tr>
<tr>
<td>grasp</td>
<td>0.31 (201)</td>
<td>0.04 (37)</td>
<td>0.34 (201)</td>
<td>0.04 (29)</td>
</tr>
<tr>
<td>minsurf</td>
<td>14.511 (34)</td>
<td>7.811 (20)</td>
<td>109.948 (201)</td>
<td>7.28 (17)</td>
</tr>
<tr>
<td>springs</td>
<td>0.01 (17)</td>
<td>0.02 (17)</td>
<td>0.01 (15)</td>
<td>0.01 (15)</td>
</tr>
<tr>
<td>steiner</td>
<td>0.861 (201)</td>
<td>0.11 (27)</td>
<td>0.811 (201)</td>
<td>0.16 (41)</td>
</tr>
<tr>
<td>structure</td>
<td>68.829 (201)</td>
<td>10.645 (43)</td>
<td>81.186 (201)</td>
<td>12.898 (54)</td>
</tr>
</tbody>
</table>

- These problems are AMPL encodings of problems in Lobo, Vandenberghe, Boyd, and Lebret.

- “Nonconvex” refers to the “square-both-sides” reformulation.

- Numbers in parens are iteration counts.

- Red indicates failure.
### Which Formulation is Best?

<table>
<thead>
<tr>
<th>Problem</th>
<th>socp</th>
<th>$\epsilon$-pert</th>
<th>nonconvex</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>antenna</td>
<td>5.728 (40)</td>
<td>9.664 (60)</td>
<td>34.169 (201)</td>
<td>3.365 (28)</td>
</tr>
<tr>
<td>emfl</td>
<td>318.869 (201)</td>
<td>29.332 (24)</td>
<td>277.499 (201)</td>
<td>61.739 (30)</td>
</tr>
<tr>
<td>fir</td>
<td>0.42 (37)</td>
<td>0.42 (37)</td>
<td>0.47 (42)</td>
<td>0.46 (40)</td>
</tr>
<tr>
<td>grasp</td>
<td>0.31 (201)</td>
<td>0.04 (37)</td>
<td>0.34 (201)</td>
<td>0.04 (29)</td>
</tr>
<tr>
<td>minsurf</td>
<td>14.511 (34)</td>
<td>7.811 (20)</td>
<td>109.948 (201)</td>
<td>7.28 (17)</td>
</tr>
<tr>
<td>springs</td>
<td>0.01 (17)</td>
<td>0.02 (17)</td>
<td>0.01 (15)</td>
<td>0.01 (15)</td>
</tr>
<tr>
<td>steiner</td>
<td>0.861 (201)</td>
<td>0.11 (27)</td>
<td>0.811 (201)</td>
<td>0.16 (41)</td>
</tr>
<tr>
<td>structure</td>
<td>68.829 (201)</td>
<td>10.645 (43)</td>
<td>81.186 (201)</td>
<td>12.898 (54)</td>
</tr>
</tbody>
</table>

- These problems are AMPL encodings of problems in Lobo, Vandenberghe, Boyd, and Lebret.
- “Nonconvex” refers to the “square-both-sides” reformulation.
- Numbers in parens are iteration counts.
- Red indicates failure.
Is SOCP necessary?

<table>
<thead>
<tr>
<th>Problem</th>
<th>simplest nlp</th>
<th>socp</th>
</tr>
</thead>
<tbody>
<tr>
<td>emfl</td>
<td>0.59 (17)</td>
<td>29.332 (24)</td>
</tr>
<tr>
<td>minsurf</td>
<td>1.883 (16)</td>
<td>7.28 (17)</td>
</tr>
<tr>
<td>steiner</td>
<td>0.2 (57)</td>
<td>0.11 (27)</td>
</tr>
<tr>
<td>structure</td>
<td>1.211 (17)</td>
<td>10.645 (43)</td>
</tr>
<tr>
<td>random LP (50x100)</td>
<td>0.44 (15)</td>
<td>1.041 (30)</td>
</tr>
<tr>
<td>random LP (200x500)</td>
<td>26.468 (18)</td>
<td>75.679 (42)</td>
</tr>
</tbody>
</table>

- LP’s were converted to SOCP’s using

\[ x_j \geq 0, x_{j-1} \geq 0 \iff |x_j - x_{j-1}| \leq x_j + x_{j-1}. \]
12 Is SOCP necessary?

<table>
<thead>
<tr>
<th>Problem</th>
<th>simplest nlp</th>
<th>socp</th>
</tr>
</thead>
<tbody>
<tr>
<td>emfl</td>
<td>0.59 ( 17)</td>
<td>29.332 ( 24)</td>
</tr>
<tr>
<td>minsurf</td>
<td>1.883 ( 16)</td>
<td>7.28 ( 17)</td>
</tr>
<tr>
<td>steiner</td>
<td>0.2 ( 57)</td>
<td>0.11 ( 27)</td>
</tr>
<tr>
<td>structure</td>
<td>1.211 (17)</td>
<td>10.645 ( 43)</td>
</tr>
<tr>
<td>random LP (50x100)</td>
<td>0.44 ( 15)</td>
<td>1.041 ( 30)</td>
</tr>
<tr>
<td>random LP (200x500)</td>
<td>26.468 (18)</td>
<td>75.679 ( 42)</td>
</tr>
</tbody>
</table>

- LP’s were converted to SOCP’s using

\[ x_j \geq 0, x_{j-1} \geq 0 \iff |x_j - x_{j-1}| \leq x_j + x_{j-1}. \]
Applications and Computational Results

Mostly inspired by Lobo, Vandenberghe, Boyd, and Lebret, Applications of Second-Order Cone Programming
14 Finite Impulse Response (FIR) Filter Design

- Audio is stored digitally in a computer as a stream of short integers: $u_k, k \in \mathbb{Z}$.

- When the music is played, these integers are used to drive the displacement of the speaker from its resting position.

- For CD quality sound, 44100 short integers get played per second per channel.

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-32768</td>
<td>8</td>
<td>-23681</td>
</tr>
<tr>
<td>1</td>
<td>-32768</td>
<td>9</td>
<td>-18449</td>
</tr>
<tr>
<td>2</td>
<td>-32768</td>
<td>10</td>
<td>-11025</td>
</tr>
<tr>
<td>3</td>
<td>-30753</td>
<td>11</td>
<td>-6913</td>
</tr>
<tr>
<td>4</td>
<td>-28865</td>
<td>12</td>
<td>-4337</td>
</tr>
<tr>
<td>5</td>
<td>-29105</td>
<td>13</td>
<td>-1329</td>
</tr>
<tr>
<td>6</td>
<td>-29201</td>
<td>14</td>
<td>1743</td>
</tr>
<tr>
<td>7</td>
<td>-26513</td>
<td>15</td>
<td>6223</td>
</tr>
</tbody>
</table>
• A finite impulse response (FIR) filter takes as input a digital signal and convolves this signal with a finite set of fixed numbers $h_{-n}, \ldots, h_n$ to produce a filtered output signal:

$$y_k = \sum_{i=-n}^{n} h_i u_{k-i}.$$

• Sparing the details, the output power at frequency $\nu$ is given by

$$|H(\nu)|$$

where

$$H(\nu) = \sum_{k=-n}^{n} h(k) e^{2\pi ik\nu},$$

• Similarly, the mean squared deviation from a flat frequency response over a frequency range, say $\mathcal{L} \subset [0,1]$, is given by

$$\frac{1}{|\mathcal{L}|} \int_{\mathcal{L}} |H(\nu) - 1|^2 d\nu$$
minimize \( \rho \)

subject to \[
\left( \frac{1}{|\mathcal{L}|} \int_{\mathcal{L}} |H(\nu) - 1|^2 d\nu \right)^{1/2} \leq \rho
\]

\[|H(\nu)| \leq \rho \quad \nu \in \mathcal{H}\]

where

\[
H(\nu) = \sum_{k=-5}^{19} h(k) e^{2\pi ik\nu},
\]

\[
h(k) = \text{Complex filter coefficients, i.e., decision variables}
\]

\[
\mathcal{L} = [0.1, 0.5]
\]

\[
\mathcal{H} = [0.6, 0.9]
\]

Discretizing the integral, this is an SOCP.
17 Specific Example

<table>
<thead>
<tr>
<th>Constraints</th>
<th>1880</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variables</td>
<td>1648</td>
</tr>
</tbody>
</table>

**Time (Iterations)**

- **LOQO**: 17.7 (33)
- **SEDUMI (Sturm)**: 46.1 (18)

Ref: J.O. Coleman and D.P. Scholnik, U.S. Naval Research Laboratory,

MWSCAS99 paper available:

[engr.umbc.edu/~jeffc/pubs/abstracts/mwscas99socp.html](engr.umbc.edu/~jeffc/pubs/abstracts/mwscas99socp.html)

Click [here](#) for an animation.
17 Specific Example

<table>
<thead>
<tr>
<th>Constraints</th>
<th>Variables</th>
<th>Time (Iterations)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1880</td>
<td>1648</td>
<td>LOQO 17.7 (33)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SEDUMI (Sturm) 46.1 (18)</td>
</tr>
</tbody>
</table>

Ref: J.O. Coleman and D.P. Scholnik, U.S. Naval Research Laboratory,

MWSCAS99 paper available:
engr.umbc.edu/~jeffc/pubs/abstracts/mwscas99socp.html

Click here for an animation.
Low Pass Filter—Reformulation as QCLP

- Replace $\rho$ with $\sqrt{\sigma}$ everywhere except the objective function (since the square root function is monotone):

$$\begin{align*}
\text{minimize} & \quad \sigma \\
\text{subject to} & \quad \frac{1}{|\mathcal{L}|} \int_{\mathcal{L}} |H(\nu) - 1|^2 d\nu \leq \sigma \\
& \quad |H|^2(\nu) \leq \sigma \quad \nu \in \mathcal{H}
\end{align*}$$

- This variant involves smooth convex quadratic constraints.

- But, squared things vary over a larger dynamic range which might lead to numerical problems.

- Tried one example, $n = 14$, frequency discretized in 2000 parts.

- SOCP variant solves in 66 iterations and 12.4 seconds.

- QCLP variant solves in 65 iterations and 11.1 seconds.

- Not much difference in this case.
Filter Design: Woofer, Midrange, Tweeter

minimize \( \rho \)

subject to \( \int_0^1 (H_w(\nu) + H_m(\nu) + H_t(\nu) - 1)^2 d\nu \leq \epsilon \)

\[
\left( \frac{1}{|W|} \int_W H_w^2(\nu) d\nu \right)^{1/2} \leq \rho \quad W = [.2, .8]
\]

\[
\left( \frac{1}{|M|} \int_M H_m^2(\nu) d\nu \right)^{1/2} \leq \rho \quad M = [.4, .6] \cup [.9, .1]
\]

\[
\left( \frac{1}{|T|} \int_T H_t^2(\nu) d\nu \right)^{1/2} \leq \rho \quad T = [.7, .3]
\]

where

\[
H_i(\nu) = h_i(0) + 2 \sum_{k=1}^{n-1} h_i(k) \cos(2\pi k \nu), \quad i = W, M, T
\]

\[
h_i(k) = \text{filter coefficients, i.e., decision variables}
\]
Specific Example: Pink Floyd’s “Money”

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter length: $n$</td>
<td>14</td>
</tr>
<tr>
<td>Integral discretization: $N$</td>
<td>1000</td>
</tr>
<tr>
<td>Constraints</td>
<td>4</td>
</tr>
<tr>
<td>Variables</td>
<td>43</td>
</tr>
<tr>
<td>Time (secs)</td>
<td></td>
</tr>
<tr>
<td>LOQO</td>
<td>79</td>
</tr>
<tr>
<td>MINOS</td>
<td>164</td>
</tr>
<tr>
<td>LANCELOT</td>
<td>3401</td>
</tr>
<tr>
<td>SNOPT</td>
<td>35</td>
</tr>
</tbody>
</table>

Ref: J.O. Coleman, U.S. Naval Research Laboratory, CISS98 paper available: [engr.umbc.edu/~jeffc/pubs/abstracts/ciss98.html](engr.umbc.edu/~jeffc/pubs/abstracts/ciss98.html)

Click [here](#) for demo
Specific Example: Pink Floyd’s “Money”

filter length: \( n = 14 \)

integral discretization: \( N = 1000 \)

constraints 4

variables 43

time (secs)

LOQO 79

MINOS 164

LANCELOT 3401

SNOPT 35

Ref: J.O. Coleman, U.S. Naval Research Laboratory,
CISS98 paper available: engr.umbc.edu/~jeffc/pubs/abstracts/ciss98.html

Click here for demo
Wide-Band Antenna Array Design

- Given: a linear array (or a 2-D grid) of radar antennae.
- An incoming signal produces a signal at each antenna.
- A linear combination of the signals is made to produce one output signal.
- Coefficients of the linear combination can be chosen to accentuate and/or attenuate the output signal’s strength as a function of the input signal’s source direction.
- Similar to FIR filter design (if freq of incoming signal is fixed).
- The set of antennae is analogous to the set of time delays in FIR filter design.
- The direction of the input signal is analogous to frequency in FIR filter design.
- **Wide band** means that we consider a range of frequencies. This adds an extra dimension to the problem (literally).
minimize $\alpha$

subject to $\int\int_{(\theta, \nu) \in S} |A(\theta, \nu)|^2 d\theta d\nu \leq \alpha,$

$|A(\theta, \nu)| \leq 10^{-25/20}, \quad (\theta, \nu) \in S$

$|A(\theta, \nu)| \leq 10^{-45/20}, \quad (\theta, \nu) \in S_0$

$\int_{\nu \in P} |A(\theta_m, \nu) - \beta_m|^2 d\nu \leq 10^{-50/10}, \quad m = 1, \ldots, M$

where

$A(\theta, \nu) = \sum_k \sum_n c_{kn} e^{-2\pi i (k\theta + n\nu)}$

$c_{kn} =$ complex-valued **design weight** for array element $k$ at freq tap $n$

$P =$ subset of direction/freq pairs representing passband

$S =$ subset of direction/freq pairs representing sidelobe

$S_0 =$ subset of sidelobe spelling NRL

$\{\theta_m\} =$ finite set of directions “covering” pass band
Specific Example

15 antennae in a linear array
21 “taps” on each array
671 Chebychev constraints to spell “NRL”

<table>
<thead>
<tr>
<th>Constraints</th>
<th>Variables</th>
<th>Time (Iterations)</th>
</tr>
</thead>
<tbody>
<tr>
<td>constraints</td>
<td>6230</td>
<td>LOQO 1049 (48)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SEDUMI(Sturm) 573 (27)</td>
</tr>
</tbody>
</table>

U.S. Naval Research Laboratory,

24 Solution and Animation

- Red represents high intensity (hot)
- Blue represents low intensity (cold)
- Horizontal axis represents angle of input signal
- Vertical axis represents freq of input signal
- This problem can be formulated either as an SOCP or as a QCLP.
- Click on image to run animation.
\[\begin{align*}
\text{minimize} & \quad \rho \\
\text{subject to} & \quad |A(p)|^2 \leq \rho, \quad p \in S \\
A(p_0) & = 1,
\end{align*}\]

where

\[A(p) = \sum_{l \in \{\text{array elements}\}} w_l e^{-2\pi i p \cdot l}, \quad p \in S\]

\[w_l = \text{complex-valued design weight for array element } l\]

\[S = \text{subset of unit hemisphere: sidelobe directions}\]

\[p_0 = \text{“look” direction}\]
Specific Example: Hexagonal Lattice of 61 Elements

\[ \rho = -20 \text{ dB} = 0.01 \]
\[ S = 889 \text{ points outside } 20^\circ \text{ from look direction} \]
\[ p_0 = 40^\circ \text{ from zenith} \]

- Constraints: 839
- Variables: 123
- Time (secs):
  - LOQO: 722
  - MINOS: > 60000
  - LANCELOT: 55462
  - SNOPT: —
Solution
Given:

- A region of space in which to build something.
- Thing is essentially planar but with varying thickness.
- A place (or places) where the thing will be anchored.
- A place (or places) where loads will be applied.
- A certain total amount of material out of which to build the thing.

Objective: Design the thing to be as strong as possible.

Approach:

- Partition 2-D region into finite elements.
- Assign a thickness to each element.
Structural Design

Given:

- A region of space in which to build something.
- Thing is essentially planar but with varying thickness.
- A place (or places) where the thing will be anchored.
- A place (or places) where loads will be applied.
- A certain total amount of material out of which to build the thing.

Objective: Design the thing to be as strong as possible.

Approach:

- Partition 2-D region into finite elements.
- Assign a thickness to each element.
Given:

- A region of space in which to build something.
- Thing is essentially planar but with varying thickness.
- A place (or places) where the thing will be anchored.
- A place (or places) where loads will be applied.
- A certain total amount of material out of which to build the thing.

**Objective:** Design the thing to be as strong as possible.

**Approach:**

- Partition 2-D region into finite elements.
- Assign a thickness to each element.
**28 Structural Design**

Given:
- A region of space in which to build something.
- Thing is essentially planar but with varying thickness.
- A place (or places) where the thing will be anchored.
- A place (or places) where loads will be applied.
- A certain total amount of material out of which to build the thing.

**Objective:** Design the thing to be as strong as possible.

**Approach:**
- Partition 2-D region into finite elements.
- Assign a thickness to each element.
minimize $-p^T w$

subject to \[ \frac{V}{A_e} w^T K_e w \leq 1, \quad e \in \mathcal{E} \]

where

- $p$ = applied load
- $w$ = node displacements; optimization vars
- $V$ = total volume
- $A_e$ = thickness of element $e$
- $K_e$ = element stiffness matrix ($\succeq 0$)
- $\mathcal{E}$ = set of elements

Intrinsically a QCLP. Can be cast as an SOCP.
Specific Example: Michel Bracket

<table>
<thead>
<tr>
<th>Element Grid</th>
<th>40x72</th>
<th>20x36</th>
<th>5x9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraints</td>
<td>2880</td>
<td>720</td>
<td>45</td>
</tr>
<tr>
<td>Variables</td>
<td>5965</td>
<td>1536</td>
<td>112</td>
</tr>
</tbody>
</table>

Time (secs)

<table>
<thead>
<tr>
<th>Solver</th>
<th>LOQO</th>
<th>MINOS</th>
<th>LANCELOT</th>
<th>SNOPT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>412</td>
<td>∞</td>
<td>∞</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>89.7</td>
<td>(IL)</td>
<td>(BS)</td>
<td>(IS)</td>
</tr>
<tr>
<td></td>
<td>2.32</td>
<td>(BS)</td>
<td>15.73</td>
<td>(BS)</td>
</tr>
</tbody>
</table>
Solution
minimize $\sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} \|x_j - a_i\| + \sum_{j=1}^{n} \sum_{j'=1}^{j-1} v_{jj'} \|x_j - x_{j'}\|.$

where

$a_i = \text{location of existing facilities, } i = 1, \ldots, m$

$x_j = \text{location of new facilities, } j = 1, \ldots, n$

Classification: not smooth, convex, not SOCP.
Example: Randomly Generated

\[ m = 200 \]
\[ n = 25 \]

Used \( \epsilon \)-perturbation for smoothing.

<table>
<thead>
<tr>
<th>Constraints</th>
<th>Variables</th>
<th>Time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1849</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Solver</th>
<th>Time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOQO</td>
<td>2.3</td>
</tr>
<tr>
<td>MINOS</td>
<td>9.7</td>
</tr>
<tr>
<td>LANCELOT</td>
<td>11.0</td>
</tr>
<tr>
<td>SNOPT</td>
<td>4.7</td>
</tr>
</tbody>
</table>