DSOS and SDSOS: More Tractable Alternatives to SOS

Amir Ali Ahmadi
Goldstine Fellow, IBM Watson Research Center

Joint work with:
Anirudha Majumdar
MIT

Geometry and Algebra of Linear Matrix Inequalities, CIRM, Marseille, Nov. 2013
SOS Optimization—The Good

• Arguably, the strongest tool we have to tackle a fundamental problem in computational mathematics:

  **optimization over nonnegative polynomials**

• Essentially *any* semialgebraic problem can be reduced to it

• **Diverse applications:**
  • Polynomial optimization, control and robotics, combinatorial optimization, approximation algorithms, robust optimization, quantum computation, continuous games, filter design, automated theorem proving, ..., ..., ...

• **Beautiful theory:**
  • Classical problems in algebra revisited through the lens of computation
  • Strongest known bounds for several intractable problems
SOS Optimization—The Bad

- **Scalability** is a pain in the (___)

**Thm:** $p(x)$ of degree **2d** is sos if and only if

$$p(x) = z^T Q z \quad Q \succeq 0$$

$$z = [1, x_1, x_2, \ldots, x_n, x_1 x_2, \ldots, x_n^d]^T$$

- The size of the Gram matrix is:

$$\binom{n + d}{d} \times \binom{n + d}{d}$$

- Polynomial in $n$ for fixed $d$, but grows quickly
  - The semidefinite constraint is expensive

- E.g., a quartic in 50 variables has 316251 coefficients and the Gram matrix (that needs to be psd) has 879801 entries...
SOS Optimization—The Ugly The Hope

Make SOS scalable through a number of interesting approaches:

• Techniques for exploiting structure (e.g., symmetry and sparsity)
  • [Gatermann, Parrilo], [Vallentin], [de Klerk, Sotirov], [Riener et al.], ...

• Customized algorithms (e.g., first order methods)
  • [Bertsimas, Freund, Sun], [Nie, Wang], ...

Our approach:

• Let’s not work with SOS to begin with...

• Give other sufficient conditions for nonnegativity that are perhaps stronger than SOS, but hopefully cheaper
Not totally clear a priori how to do this...

Consider, e.g., the following two sets:

1) All polynomials that are sums of 3 squares of polynomials
2) All polynomials that are sums of $4^{th}$ powers of polynomials

Both sets are clearly inside the SOS cone

- But linear optimization over either set is intractable!
- So set inclusion doesn’t mean anything in terms of complexity
- We have to work a bit harder...
dsos and sdsos

**Defn.** A polynomial $p$ is *diagonally-dominant-sum-of-squares (dsos)* if it can be written as:

$$p = \sum_i \alpha_i m_i^2 + \sum_{i,j} \beta_{ij} (m_i \pm m_j)^2,$$

for some monomials $m_i, m_j$ and some nonnegative constants $\alpha_i, \beta_{i,j}$.

**Defn.** A polynomial $p$ is *scaled-diagonally-dominant-sum-of-squares (sdsos)* if it can be written as:

$$p = \sum_i \alpha_i m_i^2 + \sum_{i,j} (\beta_i m_i \pm \gamma_j m_j)^2,$$

for some monomials $m_i, m_j$ and some constants $\alpha_i \geq 0, \beta_i, \gamma_i$.

**Obvious:** $\mathcal{DSOS}_{n,d} \subseteq \mathcal{SDSOS}_{n,d} \subseteq \mathcal{SOS}_{n,d} \subseteq \mathcal{POS}_{n,d}$
r-dsos and r-sdsos

Defn. A polynomial $p$ is $r$-diagonally-dominant-sum-of-squares ($r$-dsos) if

$$p \cdot \left( \sum_i x_i^2 \right)^r$$

is dsos.

Defn. A polynomial $p$ is $r$-scaled-diagonally-dominant-sum-of-squares ($r$-sdsos) if

$$p \cdot \left( \sum_i x_i^2 \right)^r$$

is sdsos.

Easy: $r\, D\, S\, O\, S\, S_{n,d} \subseteq r\, S\, D\, S\, O\, S\, S_{n,d} \subseteq P\, O\, S_{n,d}, \forall r.$
**dd and sdd matrices**

**Defn.** A symmetric matrix $A$ is *diagonally dominant* (*dd*) if

$$ a_{ii} \geq \sum_{j \neq i} |a_{ij}| \text{ for all } i. $$

**Defn*. A symmetric matrix $A$ is *scaled diagonally dominant* (*sdd*) if there exists a diagonal matrix $D > 0$ s.t.

$$ DAD \text{ is dd.} $$

$$dd \Rightarrow sdd \Rightarrow psd$$

Greshgorin’s circle theorem

*Thanks to Pablo Parrilo for telling us about sdd matrices.*
Optimization over these sets is an **SDP, SOCP, LP** !!

(code courtesy of Pablo Parrilo)
Two natural matrix programs: DDP and SDPP

**LP:**
\[
\min \langle C, X \rangle \\
A(X) = b \\
X \text{ diagonal and nonnegative}
\]

**DDP:**
\[
\min \langle C, X \rangle \\
A(X) = b \\
X \text{ dd}
\]

**SDDP:**
\[
\min \langle C, X \rangle \\
A(X) = b \\
X \text{ sdd}
\]

**SDP:**
\[
\min \langle C, X \rangle \\
A(X) = b \\
X \succeq 0
\]
From matrices to polynomials

**Thm.** A polynomial $p$ is **dsos**

\[
p = \sum_i \alpha_i m_i^2 + \sum_{i,j} \beta_{ij} (m_i \pm m_j)^2,
\]

if and only if

\[
p(x) = z^T(x)Qz(x)
\]

\[Q \quad dd\]

**Thm.** A polynomial $p$ is **sdsos**

\[
p = \sum_i \alpha_i m_i^2 + \sum_{i,j} (\beta_i m_i \pm \gamma_j m_j)^2,
\]

if and only if

\[
p(x) = z^T(x)Qz(x)
\]

\[Q \quad sdd\]
Optimization over r-dsos and r-dsos polynomials

**Thm.** For any integer $r$, the set $r\text{DSOS}_{n,d}$ is polyhedral and the set $r\text{SDSOS}_{n,d}$ has a second order cone representation.

- For any fixed $r$ and $d$, optimization over $r\text{DSOS}_{n,d}$ (resp. $r\text{SDSOS}_{n,d}$) can be done with an LP (resp. SOCP) of size polynomial in $n$.

- Commercial solvers such as CPLEX and GUROBI are very mature (very fast, deal with numerical issues)
- We have automated the generation of these LPs and SOCPs in SPOTless (package by Megretski, Tobenkin –MIT)
How well does it do?! 

- We’ll show preliminary experiments
  - From combinatorial optimization, control, copositive programming, polynomial optimization, etc.
- And we’ll give Positivstellensatz results (converse results)
First observation: $r$-dsos can outperform sos

The Motzkin polynomial:

\[ M(x_1, x_2, x_3) = x_1^4 x_2^2 + x_2^4 x_3^4 - 3x_1^2 x_2^2 x_3^2 + x_3^6 \]

psd but not sos!

...but it’s 2-dsos.

(certificate of nonnegativity using LP)

Another ternary sextic:

\[ p(x_1, x_2, x_3) = x_1^4 x_2^2 + x_2^4 x_3^2 + x_3^4 x_1^2 - 3x_1^2 x_2^2 x_3^2 \]

not sos but 1-dsos (hence psd)
A parametric family of polynomials

\[ p(x) = 2x_1^4 + cx_2^4 + ax_1^2x_2^2 + bx_1^3x_2 \]

Compactify:

\[ p(x) = 2x_1^4 + (8 - a - b)x_2^4 + ax_1^2x_2^2 + bx_1^3x_2 \]
Minimizing a form on the sphere

\[ \min_{x \in S^{n-1}} p(x) \]

- degree=4; all coefficients present – generated randomly

<table>
<thead>
<tr>
<th>n=10</th>
<th>Lower bound</th>
<th>Run time (secs)</th>
<th>n=15</th>
<th>Lower bound</th>
<th>Run time (secs)</th>
<th>n=20</th>
<th>Lower bound</th>
<th>Run time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOS (sedumi)</td>
<td>-1.920</td>
<td>1.01</td>
<td>SOS (sedumi)</td>
<td>-3.263</td>
<td>165.3</td>
<td>SOS (sedumi)</td>
<td>-3.579</td>
<td>5749</td>
</tr>
<tr>
<td>SOS (mosek)</td>
<td>-1.920</td>
<td>0.184</td>
<td>SOS (mosek)</td>
<td>-3.263</td>
<td>5.537</td>
<td>SOS (mosek)</td>
<td>-3.579</td>
<td>79.06</td>
</tr>
<tr>
<td>sdsos</td>
<td>-5.046</td>
<td>0.152</td>
<td>sdsos</td>
<td>-10.433</td>
<td>0.444</td>
<td>sdsos</td>
<td>-17.333</td>
<td>1.935</td>
</tr>
<tr>
<td>dsos</td>
<td>-5.312</td>
<td>0.067</td>
<td>dsos</td>
<td>-10.957</td>
<td>0.370</td>
<td>dsos</td>
<td>-18.015</td>
<td>1.301</td>
</tr>
<tr>
<td>BARON</td>
<td>-175.4</td>
<td>0.35</td>
<td>BARON</td>
<td>-1079.9</td>
<td>0.62</td>
<td>BARON</td>
<td>-5287.9</td>
<td>3.69</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>n=30</th>
<th>Lower bound</th>
<th>Run time (secs)</th>
<th>n=40</th>
<th>Lower bound</th>
<th>Run time (secs)</th>
<th>n=50</th>
<th>Lower bound</th>
<th>Run time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOS (sedumi)</td>
<td>----------</td>
<td>∞</td>
<td>SOS (sedumi)</td>
<td>----------</td>
<td>∞</td>
<td>SOS (sedumi)</td>
<td>----------</td>
<td>∞</td>
</tr>
<tr>
<td>SOS (mosek)</td>
<td>----------</td>
<td>∞</td>
<td>SOS (mosek)</td>
<td>----------</td>
<td>∞</td>
<td>SOS (mosek)</td>
<td>----------</td>
<td>∞</td>
</tr>
<tr>
<td>sdsos</td>
<td>-36.038</td>
<td>9.431</td>
<td>sdsos</td>
<td>-61.248</td>
<td>53.95</td>
<td>sdsos</td>
<td>-93.22</td>
<td>100.5</td>
</tr>
<tr>
<td>dsos</td>
<td>-36.850</td>
<td>8.256</td>
<td>dsos</td>
<td>-62.2954</td>
<td>26.02</td>
<td>dsos</td>
<td>-94.25</td>
<td>72.79</td>
</tr>
<tr>
<td>BARON</td>
<td>-28546.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

PC: 3.4 GHz, 16 Gb RAM
Maximum Clique

Can be reformulated (via Motzkin-Straus) as a copositive program
→ positivity of a quartic

Polynomial optimization problem in 12 variables

Upper bound on max clique:

- \textbf{dsos}: 6.0000
- \textbf{sdsos}: 6.0000
- \textbf{1-dsos}: 4.3333
- \textbf{1-sdsos}: 4.3333
- \textbf{2-dsos}: 3.8049
- \textbf{2-sdsos}: 3.6964
- \textbf{sos}: 3.2362

LP of de Klerk and Pasechnik

- Level 0: \( \infty \)
- Level 1: \( \infty \)
- Level 2: 6.0000

\textbf{r-dsos LP guaranteed to give exact value for } \textbf{r}=(\text{max clique})^2
Stabilizing the inverted N-link pendulum (2N states)

Running time (N=5):

dsos: 37 secs    sdsos: 40 secs    sos: 1.6 hours    (speed-up: x150)
Converse results 1&2

**Thm.** Any **even** positive definite form \( p \) is \( r \)-dsos for some \( r \).

- Hence proof of positivity can always be found with LP
- Even forms (of degree 4) include **copositive programming**!
- Proof follows from a result of **Polya (1928)** on Hilbert’s 17\(^{th}\) problem:
  - \( p.\|x\|^r \) will always become a sum of squares of monomials for sufficiently large \( r \).

**Thm.** Any positive definite **bivariate** form \( p \) is \( r \)-sdsos for some \( r \).

- Proof follows from a result of **Reznick (1995)**
  - \( p.\|x\|^r \) will always become a sum of powers of linear forms for sufficiently large \( r \).
Converse result 3

**Thm.** For any positive definite form $p$, there exists an integer $r$ and a polynomial $q$ of degree $r$ such that

\[ q \text{ is dsos} \]

and

\[ pq \text{ is dsos}. \]

• Search for $q$ is an LP
• Such a $q$ is a certificate of nonnegativity of $p$
• Proof follows from a result of Habicht (1940) on Hilbert’s 17th problem:
  • Every positive definite form is a quotient of two sums of squares of monomials
The general polynomial optimization problem (POP)

\[
\begin{align*}
\min_{x} & \quad \mu(x) \\
\text{s.t.} & \quad f_i(x) \leq 0 \\
& \quad h_i(x) = 0
\end{align*}
\]

• Similar to the Lasserre/Parrilo hierarchies of SDP (based on Putinar, Schmüdgen or Stengle's Positivstellensatz) that solve POP to global optimality, our converse results imply that POP can be solved to global optimality using hierarchies of LP and SOCP coming from dsos and sdsos.
Main messages…

- Inner approximations to SOS cone and move away from SDP towards LP and SOCP
- Many of the guarantees still go through!
- This can be used *anywhere* SOS is used!
- Potential to integrate with CVXGEN for real-time algebraic optimization
- There is a range of problems with 20-100 variables that are within reach

- 60 variables is becoming a piece of (birthday) cake!
  - Happy birthday JB Lasserre!!
Thank you for your attention!
Questions?

Want to know more?
http://aaa.lids.mit.edu/