Exploiting Sparsity in Large-Scale Polynomial Optimization

Jie Wang (CNRS)
Joint work with Victor Magron and Jean B. Lasserre

2020 Peking University Youth Forum

13/12/2020

Polynomial optimization problem

The polynomial optimization problem (POP):

$$(\mathrm{Q}): \quad \begin{array}{rll} f^* := & \inf & f \\ & \text{s.t.} & g_j \geq 0, \quad j = 1, \ldots, m, \\ & & (h_i = 0, \quad i = 1, \ldots, m') \end{array}$$

where $f, g_j\,(h_i) \in \mathbb{R}[\mathbf{x}] := \mathbb{R}[x_1, \ldots, x_n]$.

In general, the problem (Q) is non-convex, NP-hard.

Diverse applications

- Discrete optimization (e.g. the Max-Cut problem)
- Truncated **K**-moment problem
- Tensor decomposition
- Big-data applications
- Computer vision
- Neural networks
- Quantum information
- ......

What do we care about?

- Compute the global optimal value
- Extract global optimal solutions
- Certify global optimality
- Approximate the global optimal value if the exact computation is impossible/unnecessary/expensive

What do we care about?

- Compute the global optimal value
- Extract global optimal solutions
- Certify global optimality
- Approximate the global optimal value if the exact computation is impossible/unnecessary/expensive

The moment-SOS hierarchy (also known as Lasserre's hierarchy) is a powerful tool to handle POPs and to answer all these questions.

What does "moment" mean?

Assume $\mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0, j = 1, \ldots, m\}$. The POP (Q) is equivalent to

$$\inf_{\mu \in \mathcal{M}(\mathbf{K})_+} \{\int_{\mathbf{K}} f(\mathbf{x})\,\mathrm{d}\mu : \mu(\mathbf{K}) = 1\}. \tag{1}$$

Let $y_{\boldsymbol{\alpha}} = \int_{\mathbf{K}} \mathbf{x}^{\boldsymbol{\alpha}}\,\mathrm{d}\mu$ (moment) for $\boldsymbol{\alpha} \in \mathbb{N}^n$. Then (1) can be rewritten as

$$\inf_{\mathbf{y}}\{L_{\mathbf{y}}(f) = \sum_{\boldsymbol{\alpha} \in \mathrm{supp}(f)} f_{\boldsymbol{\alpha}} y_{\boldsymbol{\alpha}} : \exists \mu \in \mathcal{M}(\mathbf{K})_+ \text{ s.t. } \mathbf{y} \sim \mu \text{ and } y_{\mathbf{0}} = 1\}. \tag{2}$$

What does "moment" mean?

Assume $\mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0, j = 1, \ldots, m\}$. The POP (Q) is equivalent to

$$\inf_{\mu \in \mathcal{M}(\mathbf{K})_+} \{\int_{\mathbf{K}} f(\mathbf{x})\,\mathrm{d}\mu : \mu(\mathbf{K}) = 1\}. \tag{1}$$

Let $y_{\boldsymbol{\alpha}} = \int_{\mathbf{K}} \mathbf{x}^{\boldsymbol{\alpha}}\,\mathrm{d}\mu$ (moment) for $\boldsymbol{\alpha} \in \mathbb{N}^n$. Then (1) can be rewritten as

$$\inf_{\mathbf{y}} \{L_{\mathbf{y}}(f) = \sum_{\boldsymbol{\alpha} \in \mathrm{supp}(f)} f_{\boldsymbol{\alpha}} y_{\boldsymbol{\alpha}} : \exists \mu \in \mathcal{M}(\mathbf{K})_+ \text{ s.t. } \mathbf{y} \sim \mu \text{ and } y_{\mathbf{0}} = 1\}. \tag{2}$$

Question: Which sequence $\mathbf{y} = (y_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha} \in \mathbb{N}^n}$ admits a finite Borel measure representation with support contained in $\mathbf{K}$?

What does "moment" mean?

The d-order moment matrix $M_d(\mathbf{y})$ is defined by $[M_d(\mathbf{y})]_{\boldsymbol{\beta}\boldsymbol{\gamma}} = y_{\boldsymbol{\beta}+\boldsymbol{\gamma}}$ for $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{N}^n_d$.
Given $g \in \mathbb{R}[\mathbf{x}]$, the d-order localizing matrix $M_{d-d_g}(g\mathbf{y})$ is defined by $[M_{d-d_g}(g\mathbf{y})]_{\boldsymbol{\beta}\boldsymbol{\gamma}} = \sum_{\boldsymbol{\alpha}\in\mathrm{supp}(g)} g_{\boldsymbol{\alpha}} y_{\boldsymbol{\alpha}+\boldsymbol{\beta}+\boldsymbol{\gamma}}$ for $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{N}^n_{d-d_g}$ ($d_g = \lceil \deg(g)/2 \rceil$).

Theorem

Assume Archimedean's condition holds. The sequence $\mathbf{y} = (y_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha}\in\mathbb{N}^n}$ has a finite Borel representing measure with support contained in K if and only if $M_d(\mathbf{y}) \succeq 0$, $M_{d-d_j}(g_j\mathbf{y}) \succeq 0$ for all j and d.

What does "moment" mean?

By truncating the order of moments, we then obtain a series of moment relaxations (indexed by d) of (Q) to approximate f^* from below:

$$(\mathrm{Q}_d): \quad \begin{array}{rll} \theta_d := & \inf & L_{\mathbf{y}}(f) \\ & \text{s.t.} & M_d(\mathbf{y}) \succeq 0, \\ & & M_{d-d_j}(g_j\mathbf{y}) \succeq 0, \quad j = 1, \ldots, m, \\ & & y_{\mathbf{0}} = 1. \end{array}$$

Here, $d_j = \lceil \deg(g_j)/2 \rceil$.

This is actually a semidefinite programming (SDP) problem, effectively solved by interior-point solvers (e.g. MOSEK) or first-order solvers (e.g. SDPNAL).

What does "SOS" mean?

Assume $\mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0, j = 1, \ldots, m\}$. The dual of (Q) reads as

$$f^* = \sup_{\lambda}\{\lambda : f(\mathbf{x}) - \lambda \geq 0 \text{ over } \mathbf{K}\}. \tag{3}$$

The convex cone $P_{\mathbf{K}}(\mathbf{x}) := \{g(\mathbf{x}) \mid g(\mathbf{x}) \geq 0 \text{ over } \mathbf{K}\}$ is intractable!

What does "SOS" mean?

Assume $\mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0, j = 1, \ldots, m\}$. The dual of (Q) reads as

$$f^* = \sup_{\lambda}\{\lambda : f(\mathbf{x}) - \lambda \geq 0 \text{ over } \mathbf{K}\}. \tag{3}$$

The convex cone $P_{\mathbf{K}}(\mathbf{x}) := \{g(\mathbf{x}) \mid g(\mathbf{x}) \geq 0 \text{ over } \mathbf{K}\}$ is intractable!

Question: How to effectively approximate $P_{\mathbf{K}}(\mathbf{x})$ by tractable subsets (or supsets)?

What does "SOS" mean?

$\Sigma(\mathbf{x}) := \{f \in \mathbb{R}[\mathbf{x}] \mid f = \sum_i f_i^2, f_i \in \mathbb{R}[\mathbf{x}]\}$ (SOS polynomials)
Given $\mathbf{g} = \{g_j\}_{j=1}^m \subseteq \mathbb{R}[\mathbf{x}]$, the quadratic module generated by $\mathbf{g}$ is
$\mathcal{Q}_{\mathbf{g}} := \{\sigma_0 + \sum_{j=1}^m \sigma_j g_j \mid \sigma_j \in \Sigma(\mathbf{x}), j = 0, 1, \ldots, m\} \subseteq P_{\mathbf{K}}(\mathbf{x})$,
and the truncated quadratic module of degree $2d$ is (with $g_0 := 1$)
$\mathcal{Q}_{\mathbf{g},2d} := \{\sigma_0 + \sum_{j=1}^m \sigma_j g_j \mid \sigma_j \in \Sigma(\mathbf{x}), \deg(\sigma_j g_j) \leq 2d, j = 0, 1, \ldots, m\}$.

What does “SOS” mean?

$\Sigma(\mathbf{x}) := \{f \in \mathbb{R}[\mathbf{x}] \mid f = \sum_i f_i^2, f_i \in \mathbb{R}[\mathbf{x}]\}$ (SOS polynomials)
Given $\mathbf{g} = \{g_j\}_{j=1}^m \subseteq \mathbb{R}[\mathbf{x}]$, the quadratic module generated by $\mathbf{g}$ is
$\mathcal{Q}_{\mathbf{g}} := \{\sigma_0 + \sum_{j=1}^m \sigma_j g_j \mid \sigma_j \in \Sigma(\mathbf{x}), j = 0, 1, \ldots, m\} \subseteq P_{\mathbf{K}}(\mathbf{x})$,
and the truncated quadratic module of degree $2d$ is (with $g_0 := 1$)
$\mathcal{Q}_{\mathbf{g},2d} := \{\sigma_0 + \sum_{j=1}^m \sigma_j g_j \mid \sigma_j \in \Sigma(\mathbf{x}), \deg(\sigma_j g_j) \leq 2d, j = 0, 1, \ldots, m\}$.

Theorem (Putinar's Positivstellensatz)

Assume Archimedean's condition holds. If $f > 0$ over $\mathbf{K}$, then $f \in \mathcal{Q}_{\mathbf{g}}$.

Replacing $P_{\mathbf{K}}(\mathbf{x})$ by $\mathcal{Q}_{\mathbf{g},2d}$, we then obtain a series of SOS relaxations (indexed by d) of (Q) to approximate f^* from below:

$$(\mathrm{Q}_d)^*: \quad \begin{aligned} \theta_d^* := \ & \sup \ \lambda \\ & \text{s.t.} \ \ f - \lambda \in \mathcal{Q}_{\mathbf{g},2d}. \end{aligned}$$

This is actually the dual SDP problem of the moment relaxation.

The moment-SOS hierarchy

$$
\begin{array}{ccccc}
 & & f^* & & \\
 & \rotatebox{-45}{$\leq$} & & \rotatebox{45}{$\geq$} & \\
 & \vdots & & \vdots & \\
 & \geq & & \geq & \\
\text{(the moment relaxation)} & \theta_d & \text{“}=\text{”} & \theta_d^* & \text{(the SOS relaxation)} \\
 & \geq & & \geq & \\
 & \vdots & & \vdots & \\
 & \geq & & \geq & \\
 & \theta_{\underline{d}} & \text{“}=\text{”} & \theta_{\underline{d}}^* &
\end{array}
$$

$$\underline{d} := \max\{\deg(f)/2, d_1, \ldots, d_m\}$$

Asymptotical convergence and finite convergence

Under Archimedean's condition ($\approx$ compactness): there exists $N > 0$ s.t. $N - ||\mathbf{x}||^2 \in \mathcal{Q}_{\mathbf{g}}$, we have

- $\theta_d \uparrow f^*$ and $\theta_d^* \uparrow f^*$ as $d \to \infty$ (Lassere, 2001);
- Finite convergence happens generically (Nie, 2014);
- We can verify global optimality by the so-called rank condition (flat extension/truncation);
- We can easily extract minimizers when the rank condition is satisfied.

In practice for most POPs, the moment-SOS hierarchy retrieves f^* in a few steps.

Asymptotical convergence and finite convergence

Under Archimedean's condition ($\approx$ compactness): there exists $N > 0$ s.t. $N - ||\mathbf{x}||^2 \in \mathcal{Q}_{\mathbf{g}}$, we have

- $\theta_d \uparrow f^*$ and $\theta_d^* \uparrow f^*$ as $d \to \infty$ (Lassere, 2001);
- Finite convergence happens generically (Nie, 2014);
- We can verify global optimality by the so-called rank condition (flat extension/truncation);
- We can easily extract minimizers when the rank condition is satisfied.

In practice for most POPs, the moment-SOS hierarchy retrieves f^* in a few steps.

Important Message: The moment-SOS hierarchy enables us to approximate/retrieve the global optimum/optimizers via solving a sequence of SDPs with increasing sizes.

Scalability

The size of SDP (considering $(\mathrm{Q}_d)^*$) at relaxation order d:

- SDP matrix: $\binom{n+d}{d}$
- #equality constraint: $\binom{n+2d}{2d}$

In view of the current state of SDP solvers (e.g. MOSEK), problems are limited to $n \leq 30$ when $d = 2$ on a standard laptop.

Scalability

The size of SDP (considering $(\mathrm{Q}_d)^*$) at relaxation order d:

- SDP matrix: $\binom{n+d}{d}$
- #equality constraint: $\binom{n+2d}{2d}$

In view of the current state of SDP solvers (e.g. MOSEK), problems are limited to $n \le 30$ when $d = 2$ on a standard laptop.

Exploiting structure:

- quotient ring
- symmetry
- constant trace property
- sparsity (correlative sparsity and term sparsity)

Correlative sparsity (Waki et al., 2006)

The basic idea is to partition the variables into cliques according to the correlations between variables.

Correlative sparsity pattern (csp) graph $G^{\mathrm{csp}}(V, E)$:
$V := \{x_1, \ldots, x_n\}$
$\{x_i, x_j\} \in E \iff x_i, x_j$ appear in the same term of f or appear in the same constraint g_j

Correlative sparsity (Waki et al., 2006)

The basic idea is to partition the variables into cliques according to the correlations between variables.

Correlative sparsity pattern (csp) graph $G^{\mathrm{csp}}(V, E)$:
$V := \{x_1, \ldots, x_n\}$
$\{x_i, x_j\} \in E \Longleftrightarrow x_i, x_j$ appear in the same term of f or appear in the same constraint g_j

We then construct moment/localizing matrices with respect to the variables involved in each maximal clique of the csp graph:

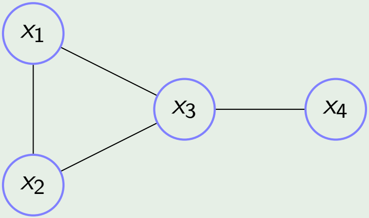
$$I_k \longmapsto M_d(\mathbf{y}, I_k), M_{d-d_j}(g_j\mathbf{y}, I_k)$$

Correlative sparsity

Example

Consider $f = x_1^4 + x_1x_2^2 + x_2x_3 + x_3^2x_4^2$ and $g_1 = 1 - x_1^2 - x_2^2 - x_3^2$, $g_2 = 1 - x_3x_4$.

Figure: The csp graph for f and $\{g_1, g_2\}$



Two maximal cliques: $\{x_1, x_2, x_3\}$ and $\{x_3, x_4\}$

The correlative sparsity adapted moment-SOS hierarchy

- If the csp graph is chordal (otherwise we need a chordal extension), then the correlative sparsity adapted moment-SOS hierarchy shares the same convergence as the standard one;
- We can still verify global optimality by the (adapted) rank condition;
- We can still extract global minimizers if certain rank conditions are satisfied;
- Significantly improve scalability if the sizes of maximal cliques of the csp graph are small (e.g. ≤ 10).

Term sparsity

In contrast with correlative sparsity concerning variables, term sparsity treats sparsity at the term/monomial level.

Term sparsity

In contrast with correlative sparsity concerning variables, term sparsity treats sparsity at the term/monomial level.

$V_d(\mathbf{x}) := \{1, x_1, \ldots, x_n, x_1^d, \ldots, x_n^d\}$ the monomial basis of degree $\leq d$.

Term sparsity pattern (tsp) graph $G^{\mathrm{tsp}}(V, E)$ (with relaxation order d):
$V := V_d(\mathbf{x})$
$\{\mathbf{x}^{\boldsymbol{\alpha}}, \mathbf{x}^{\boldsymbol{\beta}}\} \in E \iff \mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\beta}} = \mathbf{x}^{\boldsymbol{\alpha}}\mathbf{x}^{\boldsymbol{\beta}} \in \mathrm{supp}(f) \cup \bigcup_{j=1}^m \mathrm{supp}(g_j) \cup V_d(\mathbf{x})^2$

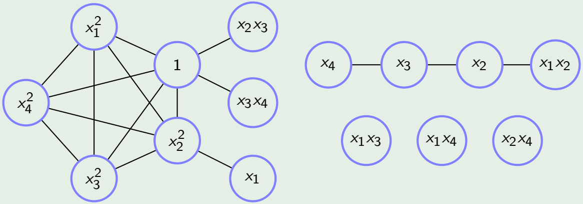
(For $f = \sum_{\boldsymbol{\alpha}} f_{\boldsymbol{\alpha}} \mathbf{x}^{\boldsymbol{\alpha}} \in \mathbb{R}[\mathbf{x}]$, $\mathrm{supp}(f) := \{\mathbf{x}^{\boldsymbol{\alpha}} \mid f_{\boldsymbol{\alpha}} \neq 0\}$)

Term sparsity

Example

Consider $f = x_1^4 + x_1x_2^2 + x_2x_3 + x_3^2x_4^2$ and $g_1 = 1 - x_1^2 - x_2^2 - x_3^2$, $g_2 = 1 - x_3x_4$.

Figure: The tsp graph for f and $\{g_1, g_2\}$ with $d = 2$



Term sparsity

Suppose the tsp graph G^{tsp} has connected components: $\mathscr{B}_1, \ldots, \mathscr{B}_t$. So

$$V_d(\mathbf{x}) = \bigsqcup_{i=1}^{t} \mathscr{B}_i.$$

For each $\mathscr{B}_i$, we construct a block of the moment matrix: $M_{\mathscr{B}_i}(\mathbf{y})$.

Term sparsity

Suppose the tsp graph G^{tsp} has connected components: $\mathscr{B}_1, \ldots, \mathscr{B}_t$. So

$$V_d(\mathbf{x}) = \bigsqcup_{i=1}^{t} \mathscr{B}_i.$$

For each $\mathscr{B}_i$, we construct a block of the moment matrix: $M_{\mathscr{B}_i}(\mathbf{y})$.

In such a way, we replace one big matrix $M_d(\mathbf{y})$ by a series of smaller submatrices $M_{\mathscr{B}_i}(\mathbf{y}), i = 1, \ldots, t$ in the moment relaxation.

Remark: The same thing can be also done for the localizing matrices $M_{d-d_j}(\mathbf{y}), j = 1, \ldots, m$.

Extending to an iterative procedure

For simplicity, we consider the unconstrained case. For a graph $G(V, E)$ with nodes $V_d(\mathbf{x})$ ($d = \deg(f)/2$), define

$$\mathrm{supp}(G) := \{\mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\beta}} \mid \{\mathbf{x}^{\boldsymbol{\alpha}}, \mathbf{x}^{\boldsymbol{\beta}}\} \in E\}.$$

Let $G^{(0)} = G^{\mathrm{tsp}}$. We iteratively define a sequence of graphs $(G^{(k)})_{k\geq 1}$ via two successive operations:

1. Support extension: let $F^{(k)}$ be the graph with nodes $V_d(\mathbf{x})$ and edges

$$E(F^{(k)}) := \{\{\mathbf{x}^{\boldsymbol{\alpha}}, \mathbf{x}^{\boldsymbol{\beta}}\} \mid \mathbf{x}^{\boldsymbol{\alpha}+\boldsymbol{\beta}} \in \mathrm{supp}(G^{(k-1)}) \cup V_d(\mathbf{x})^2\}$$

2. Block closure: $G^{(k)} = \overline{F^{(k)}}$, i.e. $G^{(k)}$ is obtained by completing every connected components of $F^{(k)}$

The term sparsity adapted moment-SOS (TSSOS) hierarchy

Let $\mathscr{B}_1^{(k)},\ldots,\mathscr{B}_{t_k}^{(k)}$ be the connected components of $G^{(k)}$. For each $k \geq 1$, let us consider

$$(\mathrm{Q}^k): \quad \begin{array}{rll} \theta^{(k)} := & \inf & L_{\mathbf{y}}(f) \\ & \text{s.t.} & M_{\mathscr{B}_i^{(k)}}(\mathbf{y}) \succeq 0, \quad i = 1,\ldots,t_k \\ & & y_{\mathbf{0}} = 1. \end{array}$$

One then obtains

$$\theta_{\text{sdsos}} \leq \theta^{(1)} \leq \theta^{(2)} \leq \cdots \leq f^*.$$

We call $(\mathrm{Q}^k), k = 1,2,\ldots$ the TSSOS hierarchy for (Q) and k the sparse order.

A two-level hierarchy of lower bounds

The above procedure can be extended to the constrained case. As a consequence, we obtain a two-level hierarchy of lower bounds for f^*: $(\underline{d} := \max\{\deg(f)/2, d_1, \ldots, d_m\})$

$$
\begin{array}{ccccccc}
\theta^{(1)}_{\underline{d}} & \leq & \theta^{(2)}_{\underline{d}} & \leq & \cdots & \leq & \theta_{\underline{d}} \\
\wedge| & & \wedge| & & & & \wedge| \\
\theta^{(1)}_{\underline{d}+1} & \leq & \theta^{(2)}_{\underline{d}+1} & \leq & \cdots & \leq & \theta_{\underline{d}+1} \\
\wedge| & & \wedge| & & & & \wedge| \\
\vdots & & \vdots & & \vdots & & \vdots \\
\wedge| & & \wedge| & & & & \wedge| \\
\theta^{(1)}_{d} & \leq & \theta^{(2)}_{d} & \leq & \cdots & \leq & \theta_{d} \\
\wedge| & & \wedge| & & & & \wedge| \\
\vdots & & \vdots & & \vdots & & \vdots
\end{array}
$$

The TSSOS hierarchy

Regarding the TSSOS hierarchy, we have

- For QCQP, $\theta_1^{(1)} = \theta_{\rm shor}$;
- Fixing a sparse order k, the sequence $(\theta_d^{(k)})_{d\geq\underline{d}}$ is monotone nondecreasing;
- Fixing a relaxation order d, the sequence $(\theta_d^{(k)})_{k\geq 1}$ converges to θ_d in finitely many steps.

Combining correlative sparsity with term sparsity

The combination of correlative sparsity with term sparsity splits into two steps:

1. Partitioning the variables with respect to the maximal cliques of the csp graph;
2. For each subsystem involving variables from one maximal clique, applying the above iterative procedure to exploit term sparsity.

In doing so, we again obtain a two-level hierarchy of lower bounds for f^*, which is called the CS-TSSOS hierarchy.

Polynomials in noncommutating variables

$\underline{X} = (X_1, \ldots, X_n)$: a tuple of noncommutating variables
$\mathbb{R}\langle \underline{X} \rangle$: the ring of real polynomials in $\underline{X}$
Given $S = \{g_1, \ldots, g_m\} \subseteq \operatorname{Sym} \mathbb{R}\langle \underline{X} \rangle$, the semialgebraic set $\mathcal{D}_S$ is

$$\mathcal{D}_S := \bigcup_{r \in \mathbb{N} \setminus \{0\}} \{\underline{A} = (A_1, \ldots, A_n) \in (\mathbb{S}^r)^n \mid g_j(\underline{A}) \succeq 0, j \in [m]\}.$$

The operator semialgebraic set $\mathcal{D}_S^\infty$ is the set of all bounded self-adjoint operators $\underline{A}$ on a Hilbert space endowed with a scalar product $\langle \cdot, \cdot \rangle$ making $g(\underline{A})$ a PSD operator, for all $g \in S$.

Eigenvalue minimization and trace minimization

Given $f = \sum_w a_w w \in \mathrm{Sym}\,\mathbb{R}\langle \underline{X} \rangle$ and $S = \{g_1, \ldots, g_m\} \subseteq \mathrm{Sym}\,\mathbb{R}\langle \underline{X} \rangle$, the **eigenvalue minimization problem** for f over the operator semialgebraic set $\mathcal{D}_S^\infty$ is defined by

$$(\mathrm{EQ}_0): \quad \lambda_{\min}(f, S) := \inf\{\langle f(\underline{A})\boldsymbol{v}, \boldsymbol{v}\rangle : \underline{A} \in \mathcal{D}_S^\infty, ||\boldsymbol{v}|| = 1\},$$

and the **trace minimization problem** for f over the semialgebraic set $\mathcal{D}_S$ is defined by

$$(\mathrm{TQ}_0): \quad \mathrm{tr}_{\min}(f, S) := \inf\{\mathrm{tr}\, f(\underline{A}) : \underline{A} \in \mathcal{D}_S\}.$$

For $A = [a_{ij}] \in \mathbb{S}^r$, $\mathrm{tr}\, A := \frac{1}{r} \sum_{i=1}^r a_{ii}$.

Remarks on noncommutative polynomial optimization

- Noncommutative polynomial optimization is typically harder than commutative polynomial optimization ($|W_d(\underline{X})| \gg |V_d(\mathbf{x})|$);
- The whole sparsity-exploiting framework can be adapted to the noncommutative case;
- Many computation problems emerging from quantum information can be modeled as a NCPOP;
- For NCPOPs from physics, symmetry is frequently present. It is mandatory to exploit several structures (sparsity, symmetry, quotient ring) simultaneously in order to reduce the computational cost.

Software

All sparsity-exploiting techniques (reduced monomial basis, quotient structure, correlative sparsity, term sparsity, combined correlative-term sparsity) have been implemented in the following two software (freely available on GitHub):

- `TSSOS`: solving commutative polynomial optimization
- `NCTSSOS`: solving noncommutative polynomial optimization

Randomly generated polynomials of the SOS form

TSSOS, NCTSSOS, GloptiPoly, Yalmip: MOSEK   SparsePOP: SDPT3

Table: Running time (in seconds) comparison with GloptiPoly, Yalmip and SparsePOP for minimizing randomly generated sparse polynomials of the SOS form with the same optimum; the symbol "-" indicates out of memory

n	$2d$	TSSOS	GloptiPoly	Yalmip	SparsePOP
8	8	0.24	306	10	24
8	8	0.34	348	13	130
8	8	0.36	326	19	175
8	10	0.58	-	92	323
8	10	0.53	-	72	1526
8	10	0.38	-	22	134
9	10	0.50	-	44	324
9	10	0.72	-	143	-
9	10	0.79	-	109	284
10	12	2.2	-	474	-
10	12	1.6	-	147	318
10	12	1.8	-	350	404
10	16	15	-	-	-
10	16	14	-	-	-
10	16	12	-	-	-
12	12	8.4	-	-	-
12	12	5.7	-	-	-
12	12	7.4	-	-	-

Randomly generated polynomials with simplex Newton polytopes

Table: Running time (in seconds) comparison with GloptiPoly, Yalmip and SparsePOP for minimizing randomly generated sparse polynomials with simplex Newton polytopes with the same optimum; the symbol "-" indicates out of memory

n	$2d$	TSSOS	GloptiPoly	Yalmip	SparsePOP
8	8	0.36	346	31	271
8	8	0.51	447	24	496
8	8	0.31	257	6.0	178
9	8	1.0	-	-	-
9	8	0.63	-	363	611
9	8	0.76	-	141	578
9	10	6.6	-	322	-
9	10	5.0	-	233	-
9	10	4.9	-	249	-
10	8	1.2	-	-	-
10	8	8.0	-	536	-
10	8	1.0	-	-	-
11	8	1.7	-	655	398
11	8	1.8	-	-	221
11	8	1.9	-	340	293
12	8	10	-	-	-
12	8	7.4	-	-	-
12	8	2.9	-	-	-

AC-OPF problems

Table: The results for AC-OPF problems; the symbol "-" indicates out of memory

n	m	CS ($d = 2$)				CS+TS ($d = 2$)			
		mb	opt	time (s)	rel. gap	mb	opt	time (s)	rel. gap
12	28	28	1.1242e4	0.21	0.00%	22	1.1242e4	0.09	0.00%
20	55	28	1.7543e4	0.56	0.05%	22	1.7543e4	0.30	0.05%
114	315	66	1.3442e5	5.59	0.39%	31	1.3396e5	2.01	0.73%
114	315	120	7.6943e4	94.9	0.00%	39	7.6942e4	14.8	0.00%
72	297	45	4.9927e3	4.43	0.07%	22	4.9920e3	2.69	0.08%
344	971	153	4.2246e5	758	0.06%	44	4.2072e5	96.0	0.48%
344	971	153	2.2775e5	504	0.00%	44	2.2766e5	71.5	0.04%
344	1325	253	–	–	–	31	2.4180e5	82.7	0.11%
344	1325	253	–	–	–	73	1.0470e5	169	0.50%
348	1809	253	–	–	–	34	1.0802e5	278	0.05%
348	1809	253	–	–	–	34	1.2096e5	201	0.03%
766	3322	153	3.3072e6	585	0.68%	44	3.3042e6	33.9	0.77%
1112	4613	231	4.2413e4	3114	0.85%	39	4.2408e4	46.6	0.86%
1112	4613	496	–	–	–	31	7.2396e4	410	0.25%
4356	18257	378	–	–	–	27	1.3953e6	934	0.51%

Eigenvalue minimization for the nc generalized Rosenbrock function

Table: The eigenvalue minimization for the nc generalized Rosenbrock function over $\mathcal{D}$, where $\mathcal{D}$ is defined by $\{1 - X_1^2, \ldots, 1 - X_n^2, X_1 - 1/3, \ldots, X_n - 1/3\}$; the symbol "-" indicates out of memory

n	CS+TS ($d = 2$)			Dense ($d = 2$)		
	mb	opt	time (s)	mb	opt	time (s)
20	3	1.0000	0.14	-	-	-
40	3	1.0000	0.22	-	-	-
60	3	0.9999	0.28	-	-	-
80	3	0.9999	0.35	-	-	-
100	3	0.9999	0.46	-	-	-
200	3	0.9999	0.89	-	-	-
400	3	1.0000	2.40	-	-	-
600	3	1.0000	4.47	-	-	-
800	3	1.0000	6.95	-	-	-
1000	3	0.9999	10.2	-	-	-
2000	3	0.9999	37.2	-	-	-
3000	3	0.9999	87.2	-	-	-
4000	3	0.9998	145	-	-	-

Trace minimization for the nc Broyden tridiagonal function

Table: The trace minimization for the nc Broyden tridiagonal function over $\mathcal{D}$, where $\mathcal{D}$ is defined by $\{1 - X_1^2, \ldots, 1 - X_n^2, X_1 - 1/3, \ldots, X_n - 1/3\}$; the symbol "-" indicates out of memory

n	CS+TS ($d = 2$)			Dense ($d = 2$)		
	mb	opt	time (s)	mb	opt	time (s)
20	6	1.1805	0.27	-	-	-
40	6	1.1828	0.53	-	-	-
60	6	1.1828	0.68	-	-	-
80	6	1.1828	0.82	-	-	-
100	6	1.1828	1.07	-	-	-
200	6	1.1828	2.45	-	-	-
400	6	1.1828	6.18	-	-	-
600	6	1.1828	12.2	-	-	-
800	6	1.1828	20.1	-	-	-
1000	6	1.1828	28.6	-	-	-
2000	6	1.1828	104	-	-	-
3000	6	1.1828	204	-	-	-
4000	6	1.1828	363	-	-	-

Conclusion and outlook

- The concept of term sparsity patterns opens a new window to exploit sparsity at the term level for polynomial optimization;
- The CS-TSSOS hierarchy is a powerful tool to handle large-scale polynomial optimization problems;
- One can exploit term sparsity for generalized moment problems (more general than polynomial optimization), SOS programming, SDP problems;
- Fruitful potential applications: optimal power flow, computer vision, big data, deep learning, quantum information, tensor decomposition, ...

Main references

- Jie Wang, Victor Magron and Jean B. Lasserre, *TSSOS: A Moment-SOS hierarchy that exploits term sparsity*, SIAM Optimization, 2020.
- Jie Wang, Victor Magron and Jean B. Lasserre, *Chordal-TSSOS: a moment-SOS hierarchy that exploits term sparsity with chordal extension*, SIAM Optimization, 2020.
- Jie Wang, Victor Magron, Jean B. Lasserre and Ngoc H. A. Mai, *CS-TSSOS: Correlative and term sparsity for large-scale polynomial optimization*, arXiv:2005.02828, 2020.
- Jie Wang and Victor Magron, Exploiting Term Sparsity in Noncommutative Polynomial Optimization, arXiv:2010.06956, 2020.
- Jie Wang, Martina Maggio and Victor Magron, *SparseJSR: A Fast Algorithm to Compute Joint Spectral Radius via Sparse SOS Decompositions*, arXiv:2008.11441, 2020.
- Jared Miller, Jie Wang, Mario Sznaier and Octavia Camps, *Model Fitting by Semialgebraic Clustering*, 2020.
- `TSSOS`: https://github.com/wangjie212/TSSOS
- `NCTSSOS`: https://github.com/wangjie212/NCTSSOS
- `SparseJSR`: https://github.com/wangjie212/SparseJSR

Thanks for your attention!