Four Lectures on Polynomial Optimization for Control

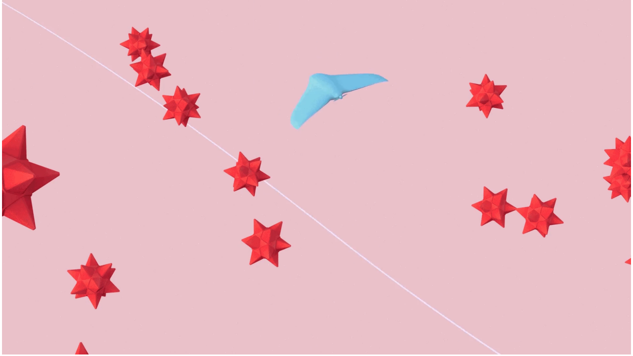
Lecture 3: Polymomial methods for robotics

Joe Warrington

Motivation

KEVIN HARTNETT SCIENCE 05.27.18 07:00 AM

A CLASSICAL MATH PROBLEM GETS PULLED INTO SELF-DRIVING CARS



A collision-free path can be guaranteed by a sum-of-squares algorithm.

OLENA SHMAHALO/QUANTA MAGAZINE

https://www.wired.com/story/a-classical-math-problem-gets-pulled-into-self-driving-cars/

Recap: Sum-of-squares and non-negativity

- A polynomial p is called *sum-of-squares (SOS)* if it can be written

$$p(x) = \sum_i f_i(x)^2$$

  for a finite number of polynomials f_i.
- A polynomial is SOS if and only if it can be written

$$p(x) = \mathbf{x}^\top \mathbf{X} \mathbf{x}$$

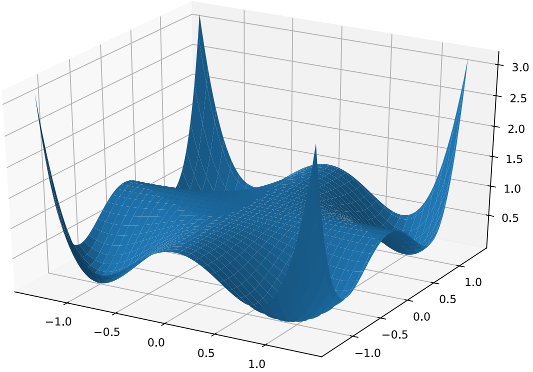
  where $\mathbf{X}$ is positive semidefinite and $\mathbf{x}$ is a vector of monomials in x. *Note:* Without loss of generality $\mathbf{X}$ can be assumed symmetric.
- Optimizing over SOS polynomials is therefore equivalent to optimizing over matrices $\mathbf{X} \succeq 0$.
- If p defined on n dimensions is SOS with degree at most $2d$, we say it belongs to the cone $SOS_{n,2d}$.[1] We use the notation

$$p \in SOS_{n,2d}.$$

[1] Note that odd degree polynomials are never SOS.

Recap: $SOS_{n,2d} \subset PSD_{n,2d}$

- Many positive (semi-)definite polynomials are not in $SOS_{n,2d}$ for any degree d
- E.g. Motzkin polynomial



- Usually we have $SOS_{n,2d} \subset PSD_{n,2d}$.
- But in the case of a very small number of (n, d) pairs featuring small n or d we actually have $SOS_{n,2d} = PSD_{n,2d}$; see Lecture 10.

Downsides of SOS programs

Desired properties of a polynomial decision problem:

1. Cover as much of $PSD_{n,2d}$ in its feasible set as possible;
2. Cheap to determine if a candidate solution is feasible.

The cone $SOS_{n,2d}$ addresses point 1 fairly well, but point 2 is less clear:

- SOS problems must be solved using a semidefinite programming (SDP) solver, e.g. SeDuMi.
- Polynomial, but in practice unattractive, scaling of problem size with d
- Relative immaturity of SDP solvers compared to e.g. LP, SOCP solvers.

In some applications, speed and numerical stability are more important than optimality
$\Rightarrow$ use a cheaper positivity certificate than sum-of-squares

Gershgorin discs[2]

The *Gershgorin disc* for row i of matrix $A \in \mathbb{C}^{n\times n}$ is a closed subset of $\mathbb{C}$, centred at a_{ii} and with radius $\sum_{i\neq j} |a_{ij}|$.

Theorem (Gershgorin 1931)

All eigenvalues of A lie within at least one Gershgorin disc.

Proof.

All of A's eigenvalue-eigenvector pairs (λ, w) satisfy $Aw = \lambda w$. Normalize any eigenvector w such that its largest element i is 1, then we have for row i,

$$\sum_{i\neq j} a_{ij} w_j + a_{ii} = \lambda w_i = \lambda,$$

or equivalently $\lambda - a_{ii} = \sum_{i\neq j} a_{ij} w_j$. The triangle inequality then yields

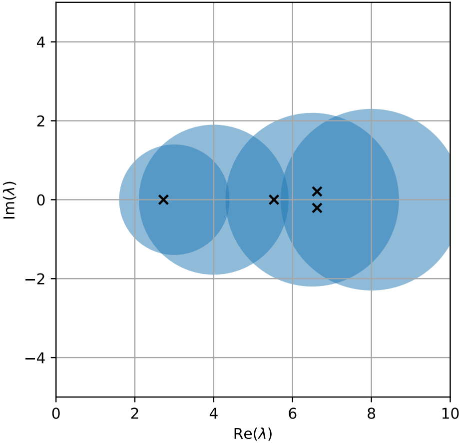
$$|\lambda - a_{ii}| \leq \sum_{i\neq j} |a_{ij}||w_j| \leq \sum_{i\neq j} |a_{ij}|.$$

Thus, λ is never more than a distance of $\sum_{i\neq j} |a_{ij}|$ from a_{ii} in the complex plane. □

[2] S. A. Gershgorin, "Über die Abgrenzung der Eigenwerte einer Matrix," *Bulletin de l'Académie des Sciences de l'URSS*, no. 6, pp. 749-954, 1931

Gershgorin discs example

$$A = \begin{bmatrix} 4 & 0.2 & -2 & -0.1 \\ 0.5 & 6.5 & 0.2 & 1.5 \\ 2 & 0.1 & 8 & 0.2 \\ 0.4 & 0.7 & 0.3 & 3 \end{bmatrix}$$



Diagonal dominance

A matrix A is *diagonally dominant (DD)* if $a_{ii} > \sum_{j \neq i} |a_{ij}|$ for all rows i.[3]

Theorem

If a symmetric matrix A is DD then it is positive semidefinite, i.e. $v^T A v \geq 0$ for all $v \in \mathbb{R}^n$.

Proof.

Follows trivially from Gershgorin disc theorem, noting that real symmetric matrices have eigenvalues lying on the real axis. □

A **diagonally-dominant sum-of-squares (DSOS) polynomial** can be written

$$p(x) = \mathbf{x}^\top Q \mathbf{x}$$

where Q is DD. Clearly from the theorem above, all DSOS polynomials of even degree $2d$ are non-negative on $\mathbb{R}^n$. Denoting the set of such polynomials $DSOS_{n,2d}$, we have

$$DSOS_{n,2d} \subseteq PSD_{n,2d}, \quad \forall n \geq 1, d \geq 0.$$

[3] *Note:* The definition automatically implies $a_{ii} \geq 0$ for all i.

DSOS representation

Theorem (AA2019, Theorem 3.4)

Polynomial p of degree $2d$ can be written $p(x) = \mathbf{x}^\top Q \mathbf{x}$, with Q a DD matrix

$$\Updownarrow$$

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

for some monomials $m_i(x), m_j(x)$ and some nonnegative scalars $\alpha_i, \beta_{ij}^+, \beta_{ij}^-$.

- Thus, DSOS polynomials really do have a sum-of-squares representation, but this is clearly at least as restrictive as the generic SOS representation. Thus

$$DSOS_{n,2d} \subseteq SOS_{n,2d}$$

- The SOS representation need not be unique.
- However, the condition $p(x) \in DSOS_{n,2d}$ can be checked (or enforced) using **linear inequalities**, whereas membership of $SOS_{n,2d}$ requires a more expensive LMI.

DSOS program

$$\begin{aligned} \min_{X \in \mathbb{S}^n} \quad & \langle C, X \rangle \\ \text{s. t.} \quad & \langle A_i, X \rangle = b_i, \quad i = 1, \ldots, m, \\ & X \quad \text{diagonally dominant.} \end{aligned}$$

Observations:

- This is just a linear program
  - *Exercise:* Write the problem in standard LP form.
- Encompasses optimization over the coefficients of polynomials $p(x) \in DSOS_{n,2d}$.
- Linear programs can be solved at far larger scale, to far higher accuracy, than semidefinite programs
- However, X is restricted to a subset of the positive definite cone $\Rightarrow$ suboptimal solutions in general.

SDSOS concept

A matrix A is *scaled diagonally dominant (SDD)* if there exists a diagonal matrix D with $d_{ii} > 0$ for all rows i, such that DAD is diagonally dominant.[4]

Theorem

If a symmetric matrix A is SDD matrices then it is positive semidefinite. **Proof:** *follows immediately from the DD case, after noting that pre- and post-multiplication by D does not change the signs of the eigenvalues.*

One can then define **scaled diagonally dominant SOS (SDSOS)** polynomials:

Theorem (AA2019, Theorem 3.6)

Polynomial p of degree $2d$ can be written $p(x) = \mathbf{x}^\top Q\mathbf{x}$, with Q an SDD matrix

$$\Updownarrow$$

$$p(x) = \sum_i \alpha_i m_i^2(x) + \sum_{i,j} \left(\hat{\beta}_{ij}^+ m_i(x) + \tilde{\beta}_{ij}^+ m_j(x)\right)^2 + \sum_{i,j} \left(\hat{\beta}_{ij}^- m_i(x) - \tilde{\beta}_{ij}^- m_j(x)\right)^2$$

for some monomials $m_i(x), m_j(x)$ and some scalars $\alpha_i, \hat{\beta}_{ij}^+, \tilde{\beta}_{ij}^+, \hat{\beta}_{ij}^-, \tilde{\beta}_{ij}^-$ with $\alpha_i \geq 0$.

[4] *Note:* Clearly all DD matrices are SDD, via the choice $D = I$.

SDSOS program

$$\begin{aligned} \min_{X \in \mathbb{S}^n} \quad & \langle C, X \rangle \\ \text{s. t.} \quad & \langle A_i, X \rangle = b_i, \quad i = 1, \ldots, m, \\ & X \quad \text{scaled diagonally dominant.} \end{aligned}$$

Observations:

- It can be shown [AA2019, Lemma 3.8] that X is SDD if and only if it can be written

$$X = \sum_{i<j} M^{ij}, \text{ where } M^{ij} \text{ takes the form } \begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & x_{ii} & 0 & x_{ij} & \vdots \\ 0 & 0 & 0 & 0 & 0 \\ \vdots & x_{ji} & 0 & x_{jj} & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \succeq 0.$$

- Each constraint $M_{ij} \succeq 0$ is in fact equivalent to the simple expressions, $x_{ii} + x_{jj} \geq 0$ and $\left\| \begin{matrix} 2x_{ij} \\ x_{ii} - x_{jj} \end{matrix} \right\|_2 \leq x_{ii} + x_{jj}$.
- Thus, we have a so-called **second order cone program (SOCP)** implemented with a combination of linear and "rotated quadratic cone" constraints coupling certain elements of X. Solve with an SOCP solver such as *ECOS*.

Nested cones

- The convex cones of DSOS, SDSOS, SOS, and PSD polynomials are related by

$$DSOS_{n,2d} \subseteq SDSOS_{n,2d} \subseteq SOS_{n,2d} \subseteq PSD_{n,2d}$$

- The boundary of $DSOS_{n,2d}$ (resp. $SDSOS_{n,2d}$) is defined by a finite number of affine (resp. rotated quadratic) constraints.

- More examples: See Figs. 1, 2, 5 in [AA2019].

r-DSOS and r-SDSOS

- Another family of cones that can approximate some polynomials in $PSD_{n,2d}$ that cannot be approximated by $SOS_{n,2d}$.
- For integer $r \geq 0$, we say polynomial $p \in rDSOS_{n,2d}$ (resp. $rSDSOS_{n,2d}$) if

$$p(x) \cdot \left(\sum_{i=1}^{n} x_i^2 \right)^r \quad \text{is DSOS (resp. SDSOS).}$$

- As $(\sum_{i=1}^{n} x_i^2)^r \geq 0$, being in $rDSOS_{n,2d}$ or $rSDSOS_{n,2d}$ implies $p(x)$ is non-negative. Thus for any r,

$$rDSOS_{n,2d} \subseteq rSDSOS_{n,2d} \subseteq PSD_{n,2d}.$$

- Optimizing over rDSOS (rSDSOS) polynomials is still an LP (SOCP).

Example: Homogeneous Motzkin polynomial

It can be shown that $p(x) = x_1^4 x_2^2 + x_1^2 x_2^4 - 3x_1^2 x_2^2 x_3^2 + x_3^6 \in 2DSOS_{3,6}$ but is not in $SOS_{3,6}$.

Control application I: Analysis of multi-joint pendulum

- Inverted N-link pendulum:
  - $2N$ states, $(\theta_i, \dot{\theta}_i)$ for each link i
  - $N - 1$ inputs (main pivot is not actuated)
- Control to the vertical upright position

Task

With given controller $u = K(x)$, approximate the region of attraction of the vertical position $\theta_i = \dot{\theta}_i = 0$ for each link.

- Do this by choosing a Lyapunov function *a priori* and **maximize** β, the threshold for which

$$V(x) \leq \beta \Rightarrow \dot{V}(x) < 0$$

or equivalently

$$[\nabla_x V(x)]^\top f(x) < 0 \quad \forall x \in \{x : V(x) \leq \beta, x \neq 0\}$$

- Compare positivity certificates

Control application II: Control synthesis for humanoid robot

- Humanoid robot designed by Boston Dynamics, with 30 states and 14 inputs.

Task

Balance the robot on its right toe (assumed to have hinge behaviour at contact point).

- Synthesis problem:

$$\begin{aligned} \max_{\rho, L(x), V(x), u(x)} \quad & \rho \\ \text{s.t.} \quad & V(x) \in DSOS_{4,8} \\ & -\dot{V}(x) + L(x)(V(x) - \rho) \in DSOS_{4,10} \\ & L(x) \in DSOS_{4,4} \\ & \sum_j V(e_j) = 1 \end{aligned}$$

  Note that $\dot{V}(x)$ depends on the choice of controller $u(x)$.
  $L(x)$ is a multiplier polynomial for the region of attraction.
  Last constraint is a normalization constraint: e_j is the j^{th} unit vector.
- Solved by sequential minimization iterating over subsets of the variables.
- Video of resulting controller recovering from different initial conditions: `http://youtu.be/lmAT556Ar5c`

Conclusions

- DSOS and SDSOS optimization offer alternative certificates of positivity for polynomials, which are
  - Cheaper to compute, but
  - Conservative in general with respect to SOS.
- Reduction from SDP to either LP (DSOS) or SOCP (SDSOS), which can be solved at far larger scale.
- rDSOS and rSDSOS offer additional degrees of freedom which can in some cases outperform SOS!
- Possibility of "optimization in the loop" applications of polynomial optimization.

Notes and references

Toolbox

DSOS and SDSOS:
`https://github.com/anirudhamajumdar/spotless/tree/spotless_isos`

This lecture was based on the following publications:

Literature

A. A. Ahmadi and A. Majumdar, "DSOS and SDSOS Optimization: More Tractable Alternatives to Sum of Squares and Semidefinite Optimization," *SIAM Journal on Applied Algebra and Geometry*, vol. 3, no. 2, pp. 193–230, 2019.
`https://arxiv.org/pdf/1706.02586.pdf`

A. Majumdar, A. A. Ahmadi, and R. Tedrake, "Control and Verification of High-Dimensional Systems with DSOS and SDSOS Programming," *IEEE Conference on Decision and Control, Los Angeles, CA, USA*, pp. 394-401, 2014