

Sum-of-squares optimization in Julia

Benoît Legat (UCL)

Joint Work with:

Chris Coey, Robin Deits, Joey Huchette and Amelia Perry (MIT)

October 29, 2026

Université catholique de Louvain (UCL)

Massachusetts Institute of Technology (MIT)

Nonnegative quadratic forms into sum of squares

$$(x_1, x_2, x_3) \quad p(x) = x^T Q x \quad \text{unique}$$

$$x_1^2 + 2x_1x_2 + 5x_2^2 + 4x_2x_3 + x_3^2 = x^T \begin{pmatrix} 1 & 1 & 0 \\ 1 & 5 & 2 \\ 0 & 2 & 1 \end{pmatrix} x$$

$$p(x) \geq 0 \quad \forall x \iff Q \geq 0 \quad \downarrow \text{cholesky}$$

$$(x_1 + x_2)^2 + (2x_2 + x_3)^2 \longleftarrow x^T \begin{pmatrix} 1 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix}^T \begin{pmatrix} 1 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix} x$$

Nonnegative polynomial into sum of squares

$$(x_1, x_2, x_3) \quad (x_1, x_1 x_2, x_2) \quad \text{not unique}$$
$$p(x) = X^T Q X$$

$$x_1^2 + 2x_1^2 x_2 + 5x_1^2 x_2^2 + 4x_1 x_2^2 + x_2^2 = X^T \begin{pmatrix} 1 & 1 & 0 \\ 1 & 5 & 2 \\ 0 & 2 & 1 \end{pmatrix} X$$

$$p(x) \geq 0 \quad \forall x \iff Q \geq 0 \quad \downarrow \text{cholesky}$$

$$(x_1 + x_1 x_2)^2 + (2x_1 x_2 + x_2)^2 \longleftarrow X^T \begin{pmatrix} 1 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix}^T \begin{pmatrix} 1 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix} X$$

When is nonnegativity equivalent to sum of squares ?

Determining whether a polynomial is nonnegative is **NP-hard**.

Hilbert 1888

Nonnegativity of $p(x)$ of n variables and degree $2d$ is equivalent to sum of squares in the following three cases:

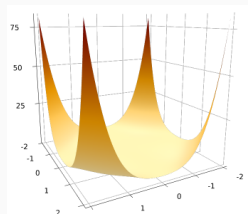
- $n = 1$: Univariate polynomials
- $2d = 2$: Quadratic polynomials
- $n = 2, 2d = 4$: Bivariate quartics

Motzkin 1967

First explicit example:

$$x_1^4 x_2^2 + x_1^2 x_2^4 + 1 - 3x_1^2 x_2^2 \geq 0 \quad \forall x$$

but is **not** a sum of squares.



Sum-of-Squares cone

Nonnegative orthant $\mathbb{R}_+^n \subset \mathbb{R}^n$

Proper and self-dual with scalar product

$$\langle a, b \rangle = b^\top a.$$

Semidefinite cone $\mathcal{S}_+^n \subset \mathcal{S}^n$

Proper and self-dual with scalar product

$$\langle A, B \rangle = \text{Tr}(BA).$$

Sum-of-Squares cone $\Sigma_{n,2d} \subset \mathbb{R}[x]_{n,2d}$

Proper and dual with scalar product

$$\langle \mu, p \rangle = \int_{\mathbb{R}^n} p(x) \mu(dx).$$

is the cone of *pseudo measures*.

What is Sum-of-squares programming ?

Linear Programming

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \langle c, x \rangle$$

$$\begin{aligned} \text{subject to} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

$$\underset{y \in \mathbb{R}^n}{\text{maximize}} \quad \langle b, y \rangle$$

$$\text{subject to} \quad A^\top y \leq c$$

Semidefinite Programming

$$\underset{Q \in \mathcal{S}^n}{\text{minimize}} \quad \langle C, Q \rangle$$

$$\text{subject to} \quad \langle A_i, Q \rangle = b_i$$

$$Q \geq 0$$

$$\underset{y \in \mathbb{R}^n}{\text{maximize}} \quad \langle b, y \rangle$$

$$\text{subject to} \quad \sum_i A_i y_i \preceq C$$

$$A_i = \text{Diag}(a_i), C = \text{Diag}(c), Q = \text{Diag}(x)$$

What is Sum-of-squares programming ?

Semidefinite Programming

$$\underset{Q \in \mathcal{S}^n}{\text{minimize}} \quad \langle C, Q \rangle$$

$$\text{subject to} \quad \langle A_i, Q \rangle = b_i$$

$$Q \succeq 0$$

$$\underset{y \in \mathbb{R}^n}{\text{maximize}} \quad \langle b, y \rangle$$

$$\text{subject to} \quad \sum_i A_i y_i \preceq C$$

Sum-of-squares Programming

$$\underset{p \in \mathbb{R}[x]_{n,2d}}{\text{minimize}} \quad \langle \nu, p \rangle$$

$$\text{subject to} \quad \langle \mu_i, p \rangle = b_i$$

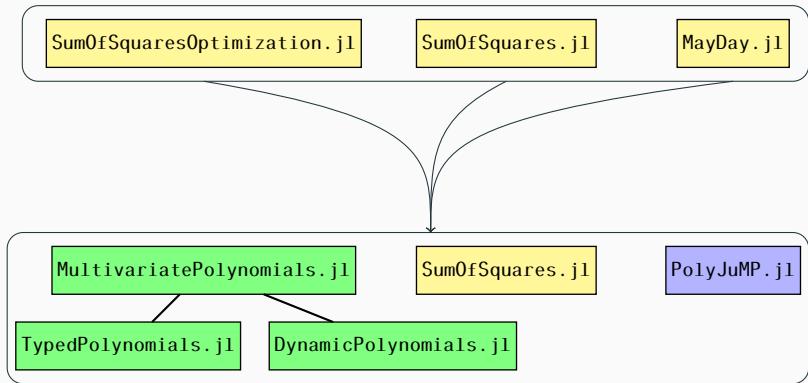
$$p \succeq 0$$

$$\underset{y \in \mathbb{R}^n}{\text{maximize}} \quad \langle b, y \rangle$$

$$\text{subject to} \quad \sum_i \mu_i y_i \preceq \nu$$

$$(A_k)_{ij} = \langle \mu_k, x_i x_j \rangle, C_{ij} = \langle \nu, x_i x_j \rangle, p(x) = x^\top Q x$$

Sum of Squares in Julia : A joint effort



Multivariate Polynomial

Choose TypedPolynomials or DynamicPolynomials:

using TypedPolynomials

@polyvar *y* # *variable with name y*

@polyvar *x[1:2]* # *tuple of variables with names x1, x2*

Build a polynomial from scratch:

$\text{motzkin} = x^4 y^2 + x^2 y^4 + 1 - 3x^2 y^2$

Build a vector of monomials:

$\text{monomials}(x, 2)$ # $\rightarrow [x^2, x_1 x_2, x_2^2]$

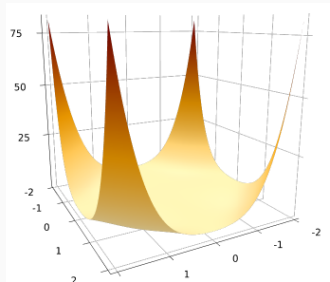
$\text{monomials}(x, 0:2)$ # $\rightarrow [x^2, x_1 x_2, x_2^2, x_1, x_2, 1]$

Constraint

```
m = Model()  
@variable m a  
@constraint a * x^2 - 2x*y + a * y^2 >= 0
```

Variable

```
m = Model()  
X = monomials([x, y], 0:2)  
@variable m p Poly(X)  
# p should be strictly positive  
@constraint m p >= 1  
@constraint m p * motzkin >= 0  
solve(m)  
finds  $p(x) = 0.9x^2 + 0.9y^2 + 2$ .
```



A module and a solver

Module

PolyJuMP needs a polymodule:

```
m = Model()  
setpolymodule!(m, SumOfSquares)
```

equivalent shortcut:

```
m = SOSModel()
```

2 lines version useful if multiple JuMP extensions used !

Solver

SOS variables/constraints need SDP solver, e.g. Mosek, SDPA, CSDP, SCS, ...

DSOS only need LP solver and SDSOS only need SOCP solver !

Domain constraint

Algebraic Set

Finite intersection of algebraic equalities, e.g.

```
@set x^2 == y^3 + z^3 && 2x^2 + 3y*z == x^3z^2
```

Basic semialgebraic set

Finite intersection of algebraic equalities and inequalities, e.g.

```
@set x*z >= y^2 && x + z == 1
```

```
S = @set x^2 + y^2 == 1
```

```
@constraint(m, x^2 + y <= 10,  
            domain = S)
```

finds $(3 - y/6)^2 + 35/36y^2$.

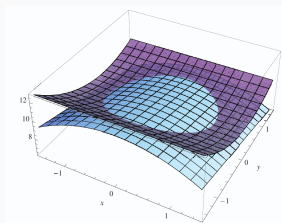


Figure 3.9 of Blekherman, Parrilo and Thomas, 2013, Semidefinite Optimization and Convex Algebraic Geometry.

SOS (resp. DSOS and SDSOS)

Variable

```
X = monomials([x, y], 0:2)
```

```
@variable m p Poly(X)
```

Variable $p(x) = X^T Q X$ where Q is semidefinite (resp. diagonally dominant, scaled diagonally dominant).

```
@variable m p SOSPoly(X)
```

```
@variable m p DSOSPoly(X)
```

```
@variable m p SDSOSPoly(X)
```

Constraint

```
@constraint m p in SOScone() # equivalent to  $p \geq 0$ 
```

```
@constraint m p in DSOScone()
```

```
@constraint m p in SDSOScone()
```

SOS matrix and SOS convex polynomial

Sum of square matrix

$$P(x) = \begin{bmatrix} x^2 - 2x + 2 & x \\ x & x^2 \end{bmatrix} = \begin{bmatrix} 1 & x \\ x - 1 & 0 \end{bmatrix}^T \begin{bmatrix} 1 & x \\ x - 1 & 0 \end{bmatrix}$$
$$y^T P(x) y = (y_1 + xy_2)^2 + (x - 1)^2 y_1^2$$

`@SDconstraint m [x^2-2x+2 x; x x^2] >= 0`

Convex polynomial

Positive semidefinite hessian:

`@SDconstraint m differentiate(p, x, 2) >= 0`

Newton Polytope

$$p(x) = X^T Q X \quad X = ?$$

Default : cheap outer approx. $\tilde{\mathcal{N}}(p)$.

Exact newton polytope

```
@constraint(m, p >= 0,  
newtonpolytope=CDDLibrary(:float))
```

```
@constraint(m, p >= 0,  
newtonpolytope=CDDLibrary(:exact))
```

Sparse multipartite

```
H = differentiate(p, x, 2)
```

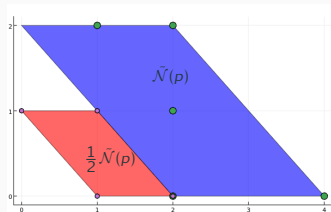
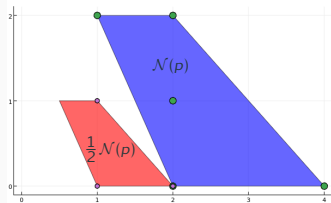
```
@constraint(m, y^T H y >= 0,
```

```
newtonpolytope=
```

```
(x=>CDDLibrary(:float),
```

```
y=>CheapOuterLibrary()))
```

$$p(x) = x^4 + 5x^2y^2 - 2x^2y \\ - xy^2 + x^2$$



Application : Polynomial optimization

Find $\min_{x \in S} p(x)$, e.g.

$$p = x^3 - x^2 + 2xy - y^2 + y^3$$

$$S = \{x \geq 0 \ \&\& \ y \geq 0 \ \&\& \ x + y \geq 1\}$$

SOS program:

```
m = SOSModel()
```

```
@variable m lb
```

```
@objective m Max lb
```

```
constr = @constraint m p >= lb, domain = S
```

How to recover the minimizer ? Get the dual μ and check whether it is atomic, i.e. $\mu = \sum_i \lambda_i \delta_{x_i}$.

```
AtomicMeasure(getdual(constr))
```

Atomic $\Rightarrow x_i$ global minimizers and lb exact minimum.

Application : Stability of Switched Systems

System $x_{k+1} = A_1 x_k$ or $x_{k+1} = A_2 x_k$. Find a common Lyapunov $V(x)$ such that $V(x) > 0$, $V(A_1 x) \leq V(x)$ and $V(A_2 x) \leq V(x)$.

```
m = SOSModel()  
X = monomials(x, 2*d)  
@variable m V Poly(X)  
@constraint m V >= sum(x.^(2d))  
@constraint m constr[i=1:2] V(x=>A[i]*x) <= V
```

How to recover an unstability certificate if it is infeasible ?

```
AtomicMeasure.(getdual(constr))
```

Atomic $\Rightarrow \mu_i$ occupation measure of unstable trajectory¹.

¹See SwitchedSystems.jl

Future work

- Symmetry reduction.
- Different polynomial basis (Lagrange, orthogonal, ...)
- Specialized method for specific algebraic sets (e.g. hypercube) and sampling algebraic varieties.
- Modelisation with measures.
- Inclusion of decision variables in semialgebraic sets using moment relaxation.
- Non-commutative (done), hermitian, orthogonal, idempotent variables.
- Syntax for hierarchies.