Polynomial Optimization

Benoît Legat

July 28th, 2023

ERC "Back to the Roots" with Prof. Bart De Moor, STADIUS, KU Leuven

JUMP-dev workshop 2023

Polynomial optimization

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & p(x) \\ \text{s.t.} \quad & h_i(x) = 0 && \forall i \in \{1, \dots, m_h\} \\ & g_i(x) \leq 0 && \forall i \in \{1, \dots, m_g\} \end{aligned}$$

where p, h_i, g_i are polynomials.

Easy or hard ?

- If p, g_i are convex and h_i are linear, it is convex...
- ... but in general, it is NP-hard
- Special case with $p = 0$ and $m_g = 0$, system of polynomial equation, already NP-hard...

Two types of Lagrangian multipliers

Constant multipliers – local certificate – KKT system

Find $x \in \mathbb{R}^n, \lambda \in \mathbb{R}^{m_h}, \sigma \in \mathbb{R}^{m_g}$ s.t.

$$p(x) + \sum_{i=1}^{m_h} \lambda_i h_i(x) + \sum_{i=1}^{m_g} {\sigma_i}^2 g_i(x)$$

Polynomial multipliers – global certificate – Putinar

Find $\gamma \in \mathbb{R}$ and polynomials $\lambda_i(x), \sigma_{i,j}(x) \in \mathbb{R}[x]$ s.t.

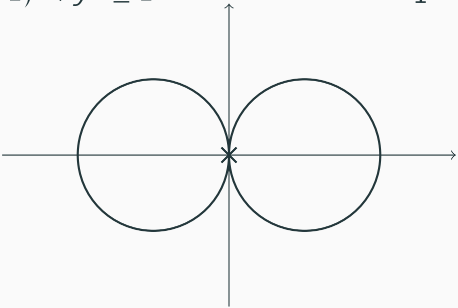
$$\forall x \in \mathbb{R}^n, \quad p(x) + \sum_{i=1}^{m_h} \lambda_i(x) h_i(x) + \sum_{i=1}^{m_g} \left(\sum_j {\sigma_{i,j}}^2(x) \right) g_i(x) = \gamma$$

Corner cases for KKT system

$$\min_{x \in \mathbb{R}^2} \quad x + y$$
$$\text{s.t.} \quad (x+1)^2 + y^2 \leq 1$$
$$(x-1)^2 + y^2 \leq 1$$

KKT system at $x = 0$:

$$1 + \sigma_1^2 - \sigma_2^2 = 0 \qquad \partial/\partial x$$
$$1 = 0 \qquad \partial/\partial y$$



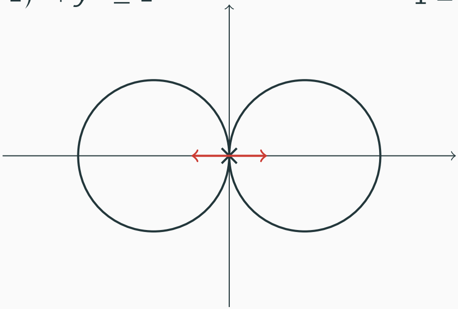
What went wrong ?

Corner cases for KKT system

$$\min_{x \in \mathbb{R}^2} \quad x + y$$
$$\text{s.t.} \quad (x+1)^2 + y^2 \le 1$$
$$(x-1)^2 + y^2 \le 1$$

KKT system at $x = 0$:

$$1 + \sigma_1^2 - \sigma_2^2 = 0 \qquad \partial/\partial x$$
$$1 = 0 \qquad \partial/\partial y$$



What went wrong ?

The rank of the jacobian dropped to 1 at $x = 0$.

Solving the KKT system

Given a system $S = \{x \mid h_i(x) = 0\}$, for any $\lambda \in \mathbb{R}[x]^m$,

$$\sum_{i=1}^{m} \lambda_i(x) h_i(x) = 0, \qquad \forall x \in S.$$

This is the linear span of, for all $\alpha \in \mathbb{N}^n, i \in \{1, \dots, m\}$,

$$x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} h_i$$

Macaulay matrix M_d has those of maxdegree below d as rows

Buchberger : Gaussian elimination → numerically unstable!

- Compute right null space Z_d with SVD → numerically stable!
- Mind the gap : rank of truncation of Z_d equal to rank of Z_d
- Gap condition is sufficient for finding the solution from Z_d.

Obtaining the minimizers for Sum-of-Squares

$$\forall x \in \mathbb{R}^n, \quad p(x) + \sum_{i=1}^{m_h} \lambda_i(x) h_i(x) + \sum_{i=1}^{m_g} \left(\sum_j {\sigma_{i,j}}^2(x) \right) g_i(x) = \gamma$$

We search over dual Lagrangian multipliers $\lambda_i(x)$, $\sigma_{i,j}(x)$. But then what's the primal x ? Isn't it the "dual of the dual" ?

- Conic dual is a symmetric PSD matrix of *moments* M_d.
- Positive semidefiniteness necessary but not sufficient for existence of a *measure* with these *moments*.
- Flatness property : Rank of truncation equal to rank of M_d
- Flatness sufficient condition for existence of an *atomic* measure with these moments.

Can you find the minimizer for the Goldstein-price function ?

```
MomentMatrix with row/column basis:
 MonomialBasis([1, x[2], x[1], x[2]^2, x[1]*x[2], x[1]^2, x[2]^3, x[1]*x[2]^2,
 x[1]^2*x[2], x[1]^3, x[2]^4, x[1]*x[2]^3, x[1]^2*x[2]^2, x[1]^3*x[2], x[1]^4])
And entries in a 15×15 SymMatrix{Float64}:
  0.999999      -0.999999       4.381387e-7  ...    1.004378e-8
 -0.999999       0.999999      -4.374988e-7         5.650864e-7
  4.381387e-7   -4.374988e-7    2.282174e-9         9.580218e-7
  0.999999      -0.999999       4.381366e-7         0.001309
 -4.374973e-7    4.381360e-7    4.364830e-10        0.001473
  2.280602e-9    4.350399e-10   3.053123e-9  ...    0.002703
 -0.999999       0.999999      -4.371625e-7         0.099747
  4.381382e-7   -4.371629e-7    1.629694e-9         0.688158
  4.349555e-10   1.628054e-9    3.198895e-9         0.497798
  3.053181e-9    3.199220e-9    1.004332e-8         1.286501
  0.999999      -0.999999       6.523214e-7  ... 1766.382361
 -4.371617e-7    6.523217e-7    2.263102e-7      -362.739066
  1.628781e-9    2.263096e-7    4.378267e-7      2468.848686
  3.198388e-9    4.378288e-7    5.650866e-7       691.956682
  1.004378e-8    5.650864e-7    9.580218e-7      4051.055429
```

Heuristic from Eq (6.15) of Laurent, M. (2009). *Sums of squares, moment matrices and optimization over polynomials*

JuMP interface has all we need

Rounding moment matrices

No flatness ? Try heuristic rounding (like MIP for fractional)

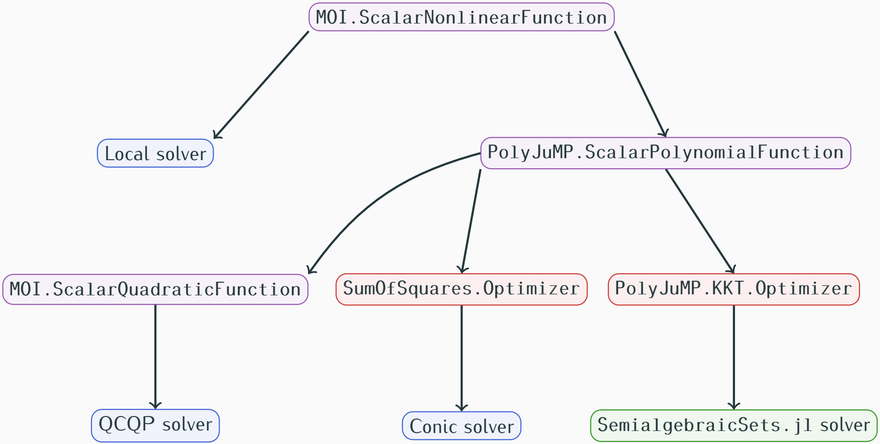
Reporting solution

- `ObjectiveBound`: Sum-of-Squares certified bound
- `ResultStatusCode`: `FEASIBLE_POINT`, `INFEASIBLE_POINT`
- Sort multiple solutions by status then objective value

Stateful optimizer object better than one-shot function

- P-time solver but exponential size increase with degree
- By default, `optimize!` solves one degree
- Next `optimize!` climbs one degree of hierarchy

Polynomial Optimization Interface



A bridge that can fail

Is that an issue ?

`MOI.ScalarNonlinearFunction` $\to$ `MOI.ScalarPolynomialFunction` may fail, e.g., for $exp(x)$.

A bridge that can fail

Is that an issue ?

`MOI.ScalarNonlinearFunction` $\to$
`MOI.ScalarPolynomialFunction` may fail, e.g., for $exp(x)$.

We already have that

`MOI.Bridges.Constraint.QuadtoSOCBridge` fails if quadratic form not PSD.

A bridge that can fail

Is that an issue ?

`MOI.ScalarNonlinearFunction` $\to$ `MOI.ScalarPolynomialFunction` may fail, e.g., for $exp(x)$.

We already have that

`MOI.Bridges.Constraint.QuadtoSOCBridge` fails if quadratic form not PSD.

And we want to continue

`MOI.ScalarNonlinearFunction` to conic with DCP fails if nonlinear expression not DCP.

A bridge that can fail

Is that an issue ?

`MOI.ScalarNonlinearFunction` → `MOI.ScalarPolynomialFunction` may fail, e.g., for $exp(x)$.

We already have that

`MOI.Bridges.Constraint.QuadtoSOCBridge` fails if quadratic form not PSD.

And we want to continue

`MOI.ScalarNonlinearFunction` to conic with DCP fails if nonlinear expression not DCP.

User chooses the solver, no default solver

These bridges are the only hope, so they are allowed to fail.

Complementarity between Macaulay and Moment matrices

	Moment matrix	Macaulay matrix
Relies on	Conic solver	SVD
Fixed d	Polynomial	Polynomial
Growing d	Exponential	Exponential
M_d	Real radical	Spurious complex solutions
M_d	Complementary slackness	Spurious FOCPs
M_d	Low-accuracy system	Numerically robust

Complementarity between Macaulay and Moment matrices

	Moment matrix	Macaulay matrix
Relies on	Conic solver	SVD
Fixed d	Polynomial	Polynomial
Growing d	Exponential	Exponential
M_d	Real radical	Spurious complex solutions
M_d	Complementary slackness	Spurious FOCPs
M_d	Low-accuracy system	Numerically robust

Seems complementary, could they work together ?

Mixing Macaulay and Sum-of-Squares frameworks