Sum-of-square optimization for verification

Pierre Roux

ONERA

Joint work with Érik Martin-Dorel, Mohamed Iguernlala and Sylvain Conchon.

May 23rd 2018

Example

SMT solvers have a hard time with non-linear numerical problems.

Demo

```
typedef struct { double x0, x1, x2; } state;

/*@ predicate inv(state *s) =
  @   6.04 * s->x0 * s->x0 - 9.65 * s->x0 * s->x1
  @   - 2.26 * s->x0 * s->x2 + 11.36 * s->x1 * s->x1
  @   + 2.67 * s->x1 * s->x2 + 3.76 * s->x2 * s->x2 <= 1; */

/*@ requires \valid(s) && inv(s) && -1 <= in0 <= 1;
  @ ensures inv(s); */
void step(state *s, double in0) {
  double pre_x0 = s->x0, pre_x1 = s->x1, pre_x2 = s->x2;

  s->x0 = 0.9379*pre_x0 - 0.0381*pre_x1 - 0.0414*pre_x2 + 0.0237*in0;
  s->x1 = -0.0404*pre_x0 + 0.968*pre_x1 - 0.0179*pre_x2 + 0.0143*in0;
  s->x2 = 0.0142*pre_x0 - 0.0197*pre_x1 + 0.9823*pre_x2 + 0.0077*in0;
}
```

Using Numerical Solvers

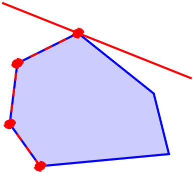
- First order theory of real numbers is decidable (Tarski).
- But complexity remains high.

$\Rightarrow$ We offer to use numerical optimization solvers: semidefinite programming (SDP) solvers.

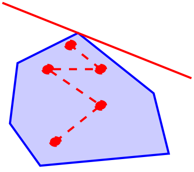
SDP solvers yield approximate solutions

- Linear programming

simplex: exact solution



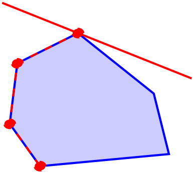
interior-point: approximate solution



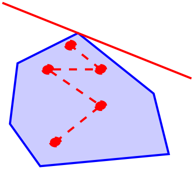
SDP solvers yield approximate solutions

- Linear programming

simplex: exact solution

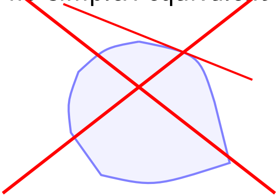


interior-point: approximate solution

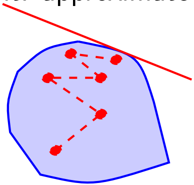


- Semidefinite programming

no simplex equivalent



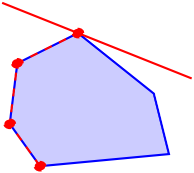
interior-point: approximate solution



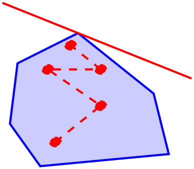
SDP solvers yield approximate solutions

- Linear programming

simplex: exact solution

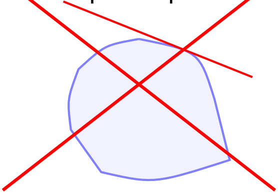


interior-point: approximate solution

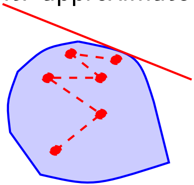


- Semidefinite programming

~~no simplex equivalent~~



interior-point: approximate solution



$\Rightarrow$ incompleteness, soundness requires care

Preliminaries

Ensuring Soundness

Integration into a SMT Solver

Experimental Results

Positivstellensatz

We want to prove that

$$p_1(x_1, \ldots, x_n) \geqslant 0 \wedge \ldots \wedge p_m(x_1, \ldots, x_n) \geqslant 0$$

is not satisfiable.

Positivstellensatz

We want to prove that

$$p_1(x_1, \ldots, x_n) \geqslant 0 \wedge \ldots \wedge p_m(x_1, \ldots, x_n) \geqslant 0$$

is not satisfiable.

Sufficient condition: there exist $r_i \in \mathbb{R}[x]$ s.t.

$$-\sum_i r_i\, p_i > 0 \quad \text{and} \quad \forall i, r_i \geqslant 0$$

Positivstellensatz

We want to prove that

$$p_1(x_1, \ldots, x_n) \geqslant 0 \wedge \ldots \wedge p_m(x_1, \ldots, x_n) \geqslant 0$$

is not satisfiable.

Sufficient condition: there exist $r_i \in \mathbb{R}[x]$ s.t.

$$-\sum_i r_i\, p_i > 0 \quad \text{and} \quad \forall i, r_i \geqslant 0$$

- equivalence under hypotheses (Putinar's Positivstellensatz)
- no practical bound on degrees of $r_i \Rightarrow$ will be arbitrarily fixed

Sum of Squares (SOS) Polynomials

Definition (SOS Polynomial)

A polynomial p is SOS if there are polynomials $q_1, \ldots, q_m$ s.t.

$$p = \sum_i q_i^2 .$$

- If p SOS then $p \geqslant 0$

Sum of Squares (SOS) Polynomials

Definition (SOS Polynomial)

A polynomial p is SOS if there are polynomials $q_1, \ldots, q_m$ s.t.

$$p = \sum_i q_i^2 .$$

- If p SOS then $p \geqslant 0$
- p SOS iff there exist $z := \left[1, x_1, x_2, x_1 x_2, \ldots, x_n^d\right]$ and $Q \succeq 0$

$$p = z^T Q \, z.$$

$\Rightarrow$ SOS can be encoded as semidefinite programming (SDP).

SOS: Example

Example

Is $p(x,y) := 2x^4 + 2x^3y - x^2y^2 + 5y^4$ SOS ?

$$p(x,y) = \begin{bmatrix} x^2 \\ y^2 \\ xy \end{bmatrix}^T \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \begin{bmatrix} x^2 \\ y^2 \\ xy \end{bmatrix}$$

that is

$$p(x,y) = q_{11}x^4 + 2q_{13}x^3y + 2q_{23}xy^3 + (2q_{12} + q_{33})x^2y^2 + q_{22}y^4$$

SOS: Example

Example

Is $p(x,y) := 2x^4 + 2x^3y - x^2y^2 + 5y^4$ SOS ?

$$p(x,y) = \begin{bmatrix} x^2 \\ y^2 \\ xy \end{bmatrix}^T \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \begin{bmatrix} x^2 \\ y^2 \\ xy \end{bmatrix}$$

that is

$p(x,y) = q_{11}x^4 + 2q_{13}x^3y + 2q_{23}xy^3 + (2q_{12} + q_{33})x^2y^2 + q_{22}y^4$

hence $q_{11} = 2$, $2q_{13} = 2$, $2q_{23} = 0$, $2q_{12} + q_{33} = -1$, $q_{22} = 5$.

SOS: Example

Example

Is $p(x,y) := 2x^4 + 2x^3y - x^2y^2 + 5y^4$ SOS ?

$$p(x,y) = \begin{bmatrix} x^2 \\ y^2 \\ xy \end{bmatrix}^T \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \begin{bmatrix} x^2 \\ y^2 \\ xy \end{bmatrix}$$

that is

$$p(x,y) = q_{11}x^4 + 2q_{13}x^3y + 2q_{23}xy^3 + (2q_{12} + q_{33})x^2y^2 + q_{22}y^4$$

hence $q_{11} = 2$, $2q_{13} = 2$, $2q_{23} = 0$, $2q_{12} + q_{33} = -1$, $q_{22} = 5$.

For instance

$$Q = \begin{bmatrix} 2 & -3 & 1 \\ -3 & 5 & 0 \\ 1 & 0 & 5 \end{bmatrix} = R^T R \qquad R = \frac{1}{\sqrt{2}} \begin{bmatrix} 2 & -3 & 1 \\ 0 & 1 & 3 \end{bmatrix}$$

hence $p(x,y) = \dfrac{1}{2}\left(2x^2 - 3y^2 + xy\right)^2 + \dfrac{1}{2}\left(y^2 + 3xy\right)^2$.

SOS: Example, Dual Formulation

Example

The constraints

$$\begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \succeq 0$$

and $q_{11} = 2$, $2q_{13} = 2$, $2q_{23} = 0$, $2q_{12} + q_{33} = -1$, $q_{22} = 5$ can also be expressed as

$$\begin{bmatrix} 2 & -\lambda & 1 \\ -\lambda & 5 & 0 \\ 1 & 0 & 2\lambda - 1 \end{bmatrix} \succeq 0$$

SOS: Example, Dual Formulation

Example

The constraints
$$\begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \succeq 0$$
and $q_{11} = 2$, $2q_{13} = 2$, $2q_{23} = 0$, $2q_{12} + q_{33} = -1$, $q_{22} = 5$
can also be expressed as
$$\begin{bmatrix} 2 & -\lambda & 1 \\ -\lambda & 5 & 0 \\ 1 & 0 & 2\lambda - 1 \end{bmatrix} \succeq 0 \text{ or } \lambda \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix} + \begin{bmatrix} 2 & 0 & 1 \\ 0 & 5 & 0 \\ 1 & 0 & -1 \end{bmatrix} \succeq 0$$
which is the dual form of (another) SDP.

SOS: Example, Dual Formulation

Example

The constraints
$$\begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \succeq 0$$
and $q_{11} = 2$, $2q_{13} = 2$, $2q_{23} = 0$, $2q_{12} + q_{33} = -1$, $q_{22} = 5$ can also be expressed as
$$\begin{bmatrix} 2 & -\lambda & 1 \\ -\lambda & 5 & 0 \\ 1 & 0 & 2\lambda - 1 \end{bmatrix} \succeq 0 \text{ or } \lambda \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix} + \begin{bmatrix} 2 & 0 & 1 \\ 0 & 5 & 0 \\ 1 & 0 & -1 \end{bmatrix} \succeq 0$$
which is the dual form of (another) SDP.

- first solution sometime yields smaller problems
- second solution can sometimes be more robust

Cholesky Decomposition

- To prove that $q \in \mathbb{R}$ is non negative,
  we can exhibit r such that $q = r^2$ (typically $r = \sqrt{q}$).

Cholesky Decomposition

- To prove that $q \in \mathbb{R}$ is non negative,
  we can exhibit r such that $q = r^2$ (typically $r = \sqrt{q}$).
- To prove that a matrix $Q \in \mathbb{R}^{s \times s}$ is positive semidefinite
  we can similarly expose R such that $Q = R^T R$
  (since $x^T \left(R^T R\right) x = (Rx)^T (Rx) = \|Rx\|_2^2 \geqslant 0$).

Cholesky Decomposition

- To prove that $q \in \mathbb{R}$ is non negative,
  we can exhibit r such that $q = r^2$ (typically $r = \sqrt{q}$).
- To prove that a matrix $Q \in \mathbb{R}^{s \times s}$ is positive semidefinite
  we can similarly expose R such that $Q = R^T R$
  (since $x^T \left(R^T R\right) x = (Rx)^T (Rx) = \|Rx\|_2^2 \geqslant 0$).
- The Cholesky decomposition computes such a matrix R
  in $\Theta(s^3)$ arithmetic operations.

Preliminaries

Ensuring Soundness

Integration into a SMT Solver

Experimental Results

SOS: Using approximate SDP solvers

Results from SDP solvers will only satisfy equality constraints up to some ϵ

$$p = z^T Q\, z + z^T E\, z, \qquad\qquad |E_{i,j}| \leqslant \epsilon.$$

SOS: Using approximate SDP solvers

Results from SDP solvers will only satisfy equality constraints up to some ϵ

$$p = z^T Q z + z^T E z, \qquad\qquad |E_{i,j}| \leqslant \epsilon.$$

Two validation methods in the litterature

- Check that for any $|E_{i,j}| \leqslant \epsilon$, $Q + E \succeq 0$
- Round Q to an exact solution $\widetilde{Q}$ s.t. $p = z^T \widetilde{Q} z$ and check $\widetilde{Q} \succeq 0$

Proving Existence of a Nearby Solution

Results from SDP solvers will only satisfy equality constraints up to some ϵ

$$p = z^T Q\, z + z^T E\, z, \qquad\qquad |E_{i,j}| \leqslant \epsilon.$$

Proving Existence of a Nearby Solution

Results from SDP solvers will only satisfy equality constraints up to some ϵ

$$p = z^T Q\, z + z^T E\, z, \qquad |E_{i,j}| \leqslant \epsilon.$$

If $Q - s\,\epsilon\, I \succeq 0$ then $Q + E \succeq 0$ and $p = z^T (Q + E)\, z$ is SOS.

Proving Existence of a Nearby Solution

Results from SDP solvers will only satisfy equality constraints up to some ϵ

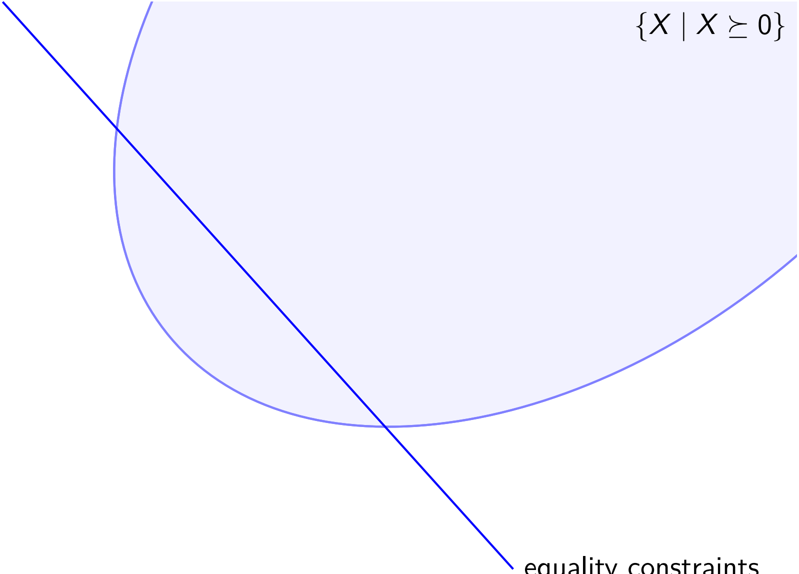
$$p = z^T Q\, z + z^T E\, z, \qquad\qquad |E_{i,j}| \leqslant \epsilon.$$

If $Q - s\,\epsilon\, I \succeq 0$ then $Q + E \succeq 0$ and $p = z^T (Q + E)\, z$ is SOS.

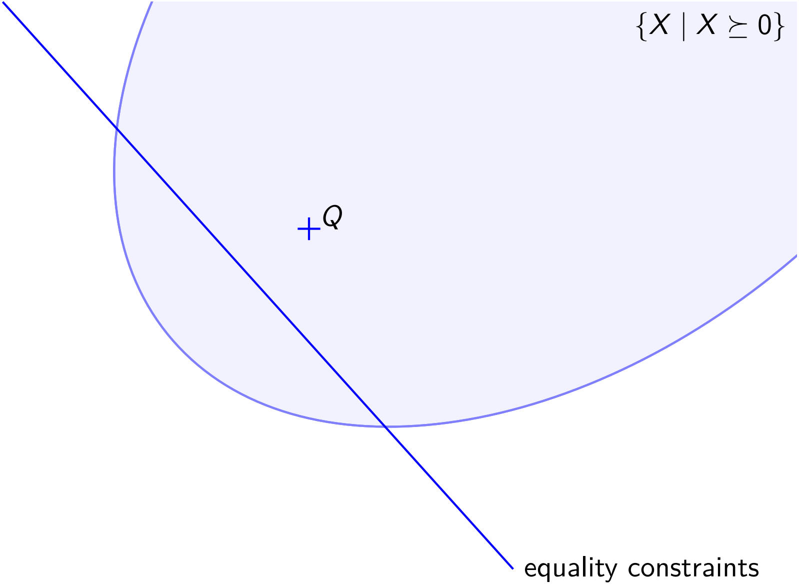
- Hence the validation method: given $Q \in \mathbb{R}^{s \times s}$, $p \simeq z^T Q\, z$
  1. Bound difference ϵ between coefficients of p and $z^T Q\, z$.
  2. If $Q - s\,\epsilon\, I \succeq 0$, then p is proved SOS.
- 1 can be done with interval arithmetic (in $\Theta(s^2)$ flops) (although rational arithmetic is more precise and fast enough) and 2 with a Cholesky decomposition ($\Theta(s^3)$ flops).

$\Rightarrow$ Efficient validation method using just floats.

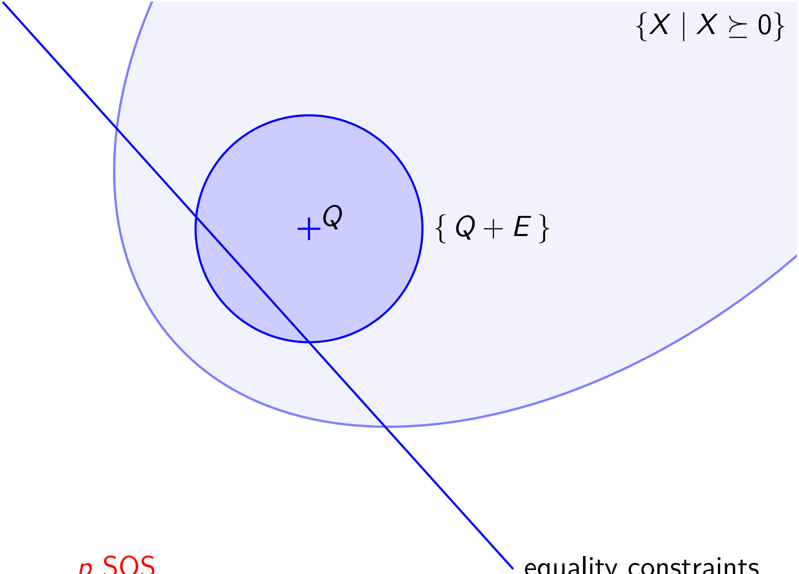
Intuitively



Intuitively



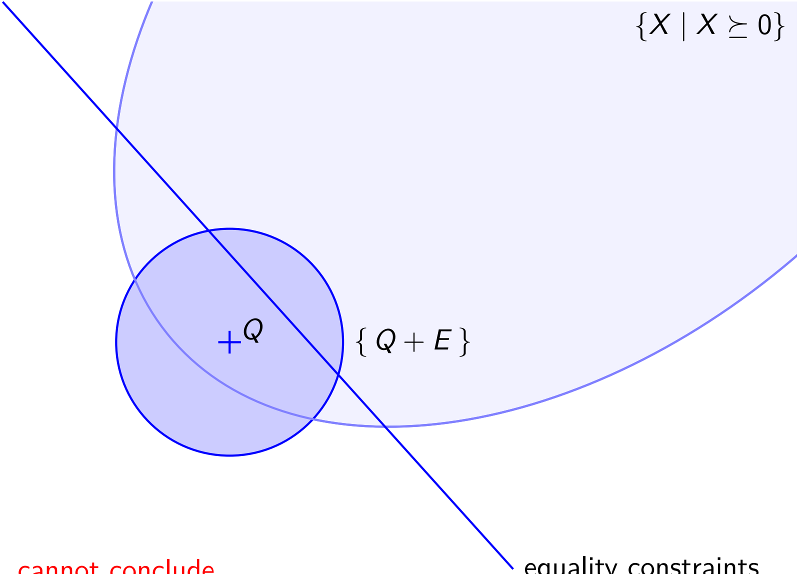
Intuitively



p SOS

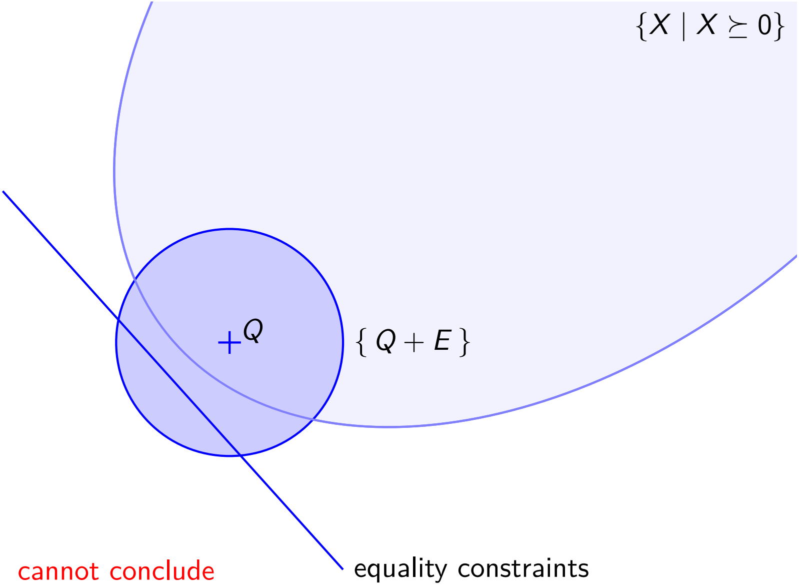
equality constraints

Intuitively

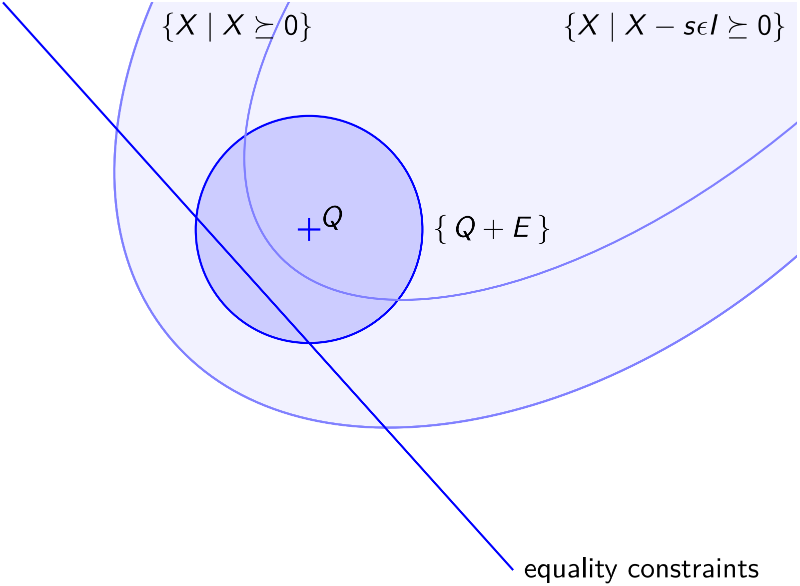


cannot conclude

Intuitively



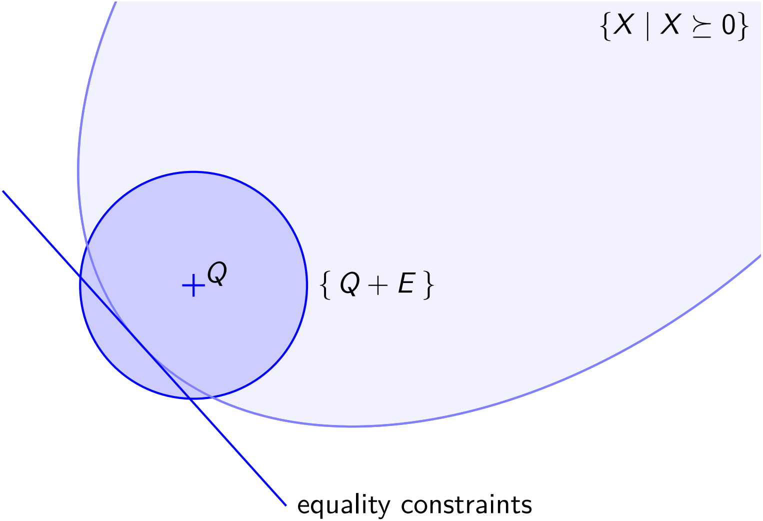
Padding



$\{X \mid X \succeq 0\}$
$\{X \mid X - s\epsilon I \succeq 0\}$
Q
$\{\, Q + E \,\}$
equality constraints

Incompleteness: Empty Interior SDP Problems

If the interior of the feasibility set of the problem is empty (i.e., no feasible Q s.t. every Q' in a small neighborhood is feasible) previous method almost never works.



Rounding to an Exact Solution

- Round Q to an exact solution $\widetilde{Q}$ s.t. $p = z^T \widetilde{Q} z$
  round every coefficients of Q up to $1, \frac{1}{2}, \frac{1}{3}, \ldots$
- and check each time whether $\widetilde{Q} \succeq 0$

− Requires the dual representation (primal just doesn't work).

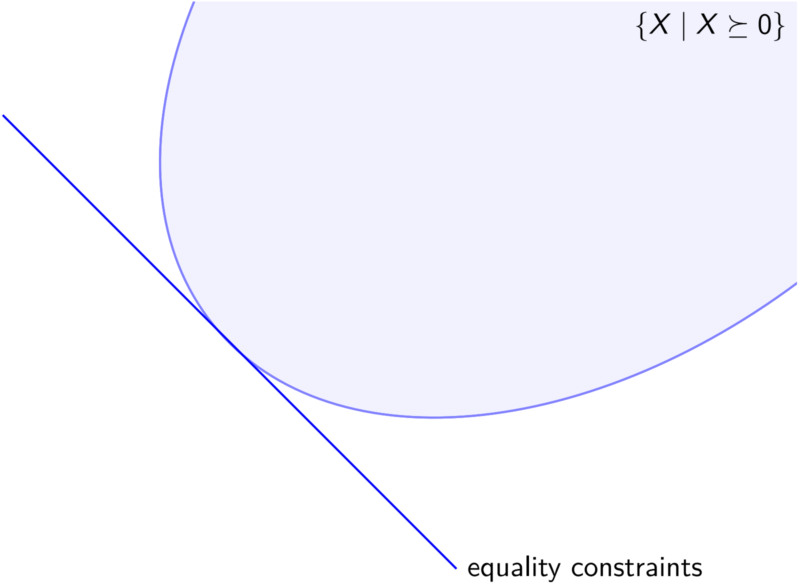
+ Can prove some empty interior problems, but still incomplete

− and requires exact checking of $Q \succeq 0$ (not just $Q \succ 0$)
  prevents using floating-point Cholesky
  but exact rational LDLT can be expensive.

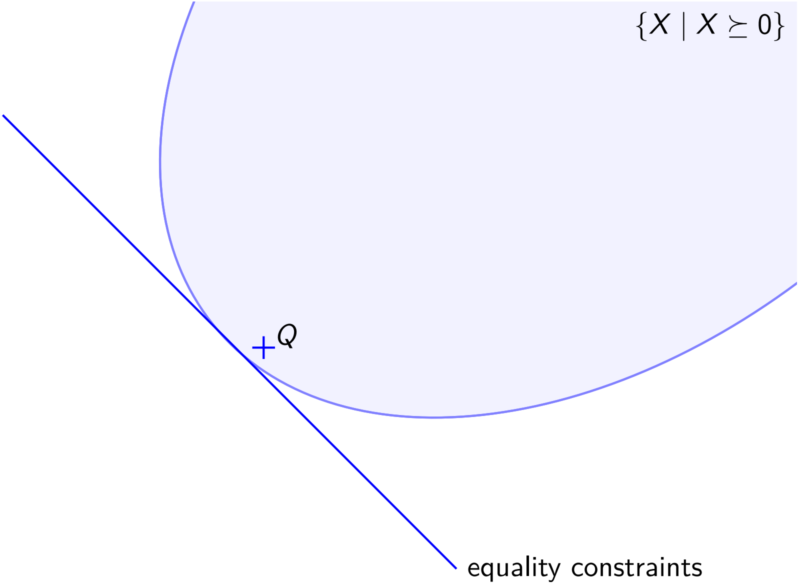
+ Can handle strict/non strict inequalities and (dis)equalities

− but requires expensive alternative relaxation scheme.

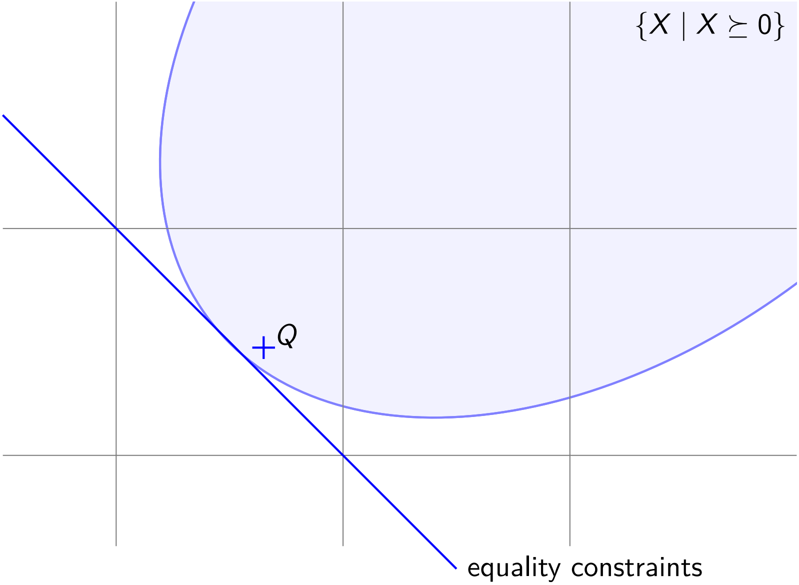
Intuitively, Rounding to an Exact Solution



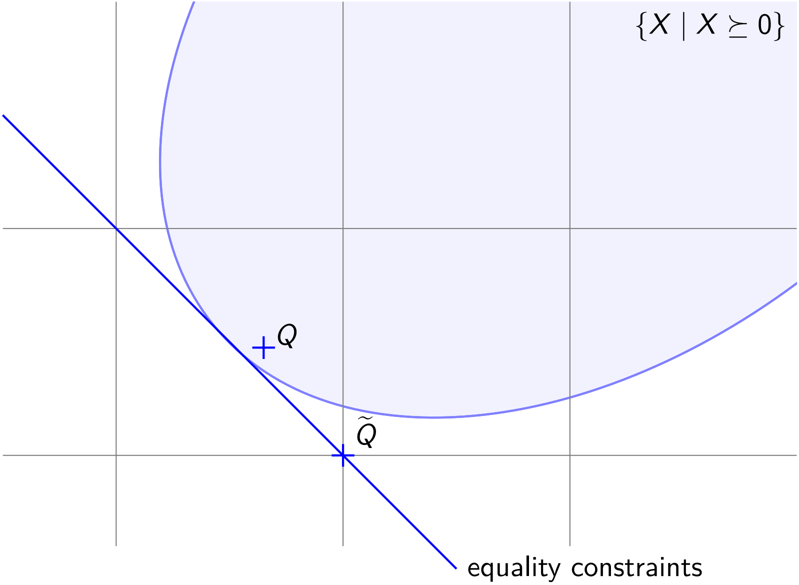
Intuitively, Rounding to an Exact Solution



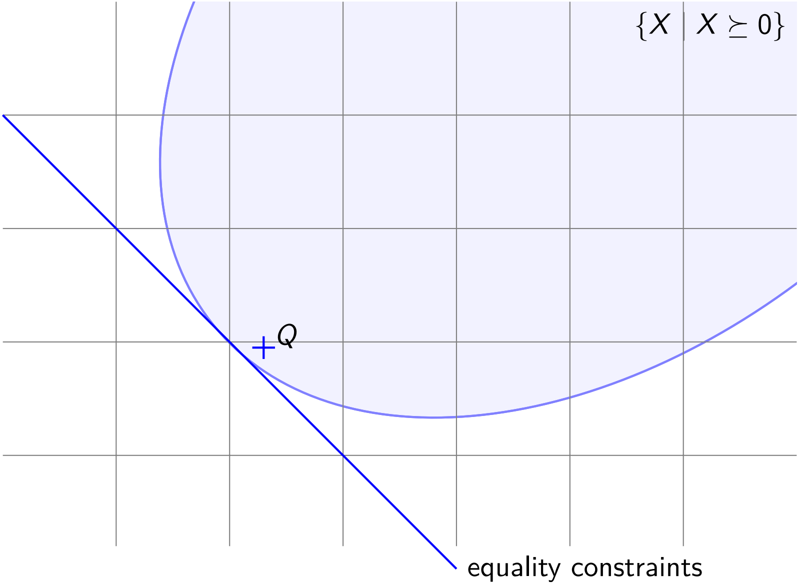
Intuitively, Rounding to an Exact Solution



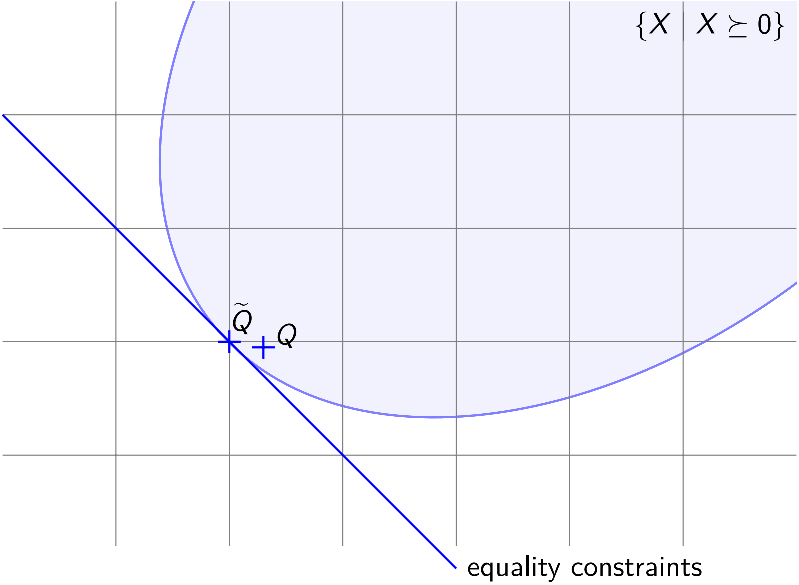
Intuitively, Rounding to an Exact Solution



Intuitively, Rounding to an Exact Solution



Intuitively, Rounding to an Exact Solution



Handling Equalities and Strict inequaities

Example

To prove

$$x_1 \geq 0 \wedge x_2 \geq 0 \wedge q_1 = 0 \wedge q_2 = 0 \wedge p > 0$$

unsatisfiable, with $q_1 := x_1^2 + x_2^2 - x_3^2 - x_4^2 - 2$, $q_2 := x_1 x_3 + x_2 x_4$ and $p := x_3 x_4 - x_1 x_2$
one can exhibit $l_1 := -\frac{1}{2}(x_1 x_2 - x_3 x_4)$, $l_2 := -\frac{1}{2}(x_2 x_3 + x_1 x_4)$, $s_2 := \frac{1}{2}\left(x_3^2 + x_4^2\right)$ and $s_7 := \frac{1}{2}\left(x_1^2 + x_2^2 + x_3^2 + x_4^2\right)$ s.t.

$$l_1 q_1 + l_2 q_2 + s_2 p + s_7 x_1 x_2 + p = 0, \quad s_2 \geq 0, \quad s_7 \geq 0.$$

Remark

Replacing $p > 0$ by $p \geq 0$, $(x_1, x_2, x_3, x_4) = (0, \sqrt{2}, 0, 0)$ is solution.

Soundness Verification for SOS: Conclusion

	exact solution	nearby solution
empty interior problems	some	no
$>, =, \neq$	some	only $\geq$
relaxation scheme	exponential	linear
proof of $Q \succeq 0$	expensive	fast
	(rational LDLT)	(fp Cholesky)
possible representation	dual	any
completenes	no	no
use off the shelf SDP	yes	yes
formal proof	easy	non trivial
	(HOL Light, Coq)	(Coq)

$\Rightarrow$ first try (cheap) nearby solution method
then if it fails and problem is small, look for exact solution

Integration into a SMT Solver

Incrementality

- common practice with simplex algorithm
- some SDP do offer to provide an initial solution
- but due to the nature of interior point algorithms doesn't give significant speed ups (can even slow down)

Small Conflict Sets

- exact method: relaxation coeffs rounded to zero indicate useless constraint
- nearby solution: heuristic solving $\log(n)$ SDPs for n constraints

The OSDP Library

OCaml library OSDP:

- simple interface to SOS programming
- interfaces SDP solvers
  - Csdp
  - Mosek
  - SDPA
- under LGPL license
- available at `https://cavale.enseeiht.fr/osdp/`

Integration in Alt-Ergo

- Alt-Ergo maintains a map: polynomial $p_i \to$ interval $[a_i, b_i]$.
- The constraints

$$-\sum_i r_i\,(p_i - a_i)(b_i - p_i) > 0 \quad \text{and} \quad \forall i, r_i \geqslant 0$$

  are provided to OSDP.
- If OSDP returns a valid solution, $\bigwedge_i p_i \in [a_i, b_i]$ is `unsat` and set of conflicting constraints can be minimized
- otherwise: `unknown`
- Integrated into Alt-Ergo 1.30 under CeCILL-C license
- available at `https://cavale.enseeiht.fr/osdp/aesdp/`

Experimental Results (1/3)

Benchmarks QF_NIA from SMT-LIB.

	AE		AESDP		AESDPap		AESDPex	
	unsat	time	unsat	time	unsat	time	unsat	time
AProVE (746)	103	7387	319	23968	359	7664	318	22701
calypto (97)	92	357	88	679	88	489	89	816
LassoRanker (102)	57	9	62	959	64	274	63	878
LCTES (2)	0	0	0	0	0	0	0	0
leipzig (5)	0	0	0	0	0	0	0	0
mcm (161)	0	0	0	0	0	0	0	0
UltimateAutom (7)	1	0.35	7	0.73	7	0.62	7	0.69
UltimateLasso (26)	26	118	26	212	26	126	26	215
total (1146)	279	7872	502	25818	544	8553	503	24611

	CVC4		Smtrat		Yices2		Z3	
	unsat	time	unsat	time	unsat	time	unsat	time
AProVE (746)	586	10821	185	3879	**709**	**1982**	252	5156
calypto (97)	87	7	89	754	**97**	**409**	95	613
LassoRanker (102)	72	27	20	12	**84**	**595**	84	2538
LCTES (2)	**1**	**0**	0	0	0	0	0	0
leipzig (5)	0	0	0	0	**1**	**0**	0	0
mcm (161)	**4**	**2489**	0	0	0	0	4	2527
UltimateAutom (7)	6	0.03	1	7.22	**7**	**0.04**	7	0.31
UltimateLasso (26)	4	66	26	177	**26**	**6**	26	21
total (1146)	780	13411	321	4829	**924**	**2993**	468	10855

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB.

Experimental Results (2/3)

Benchmarks QF_NRA from SMT-LIB.

	AE		AESDP		AESDPap		AESDPex	
	unsat	time	unsat	time	unsat	time	unsat	time
Sturm-MBO (300)	155	12950	155	13075	155	13053	155	12973
hong (20)	1	0	20	28	20	24	20	27
hycomp (2494)	1285	15351	1266	15857	1271	16080	1265	14909
keymaera (320)	261	36	291	356	278	97	291	360
LassoRanker (627)	0	0	0	0	0	0	0	0
meti-tarski (2615)	1882	10	2273	91	2267	65	2241	73
UltimateAutom (13)	0	0	0	0	0	0	0	0
zankl (85)	14	1.00	24	15.46	24	16.09	24	15.67
total (6549)	3571	28348	4029	29423	4015	29334	3996	28357

	CVC4		Smtrat		Yices2		Z3	
	unsat	time	unsat	time	unsat	time	unsat	time
Sturm-MBO (300)	285	1403	**285**	**620**	2	0	47	21
hong (20)	20	1	**20**	**0**	8	240	9	6
hycomp (2494)	2184	208	1588	13784	2182	1241	2201	4498
keymaera (320)	249	4	307	13	270	359	**318**	**2**
LassoRanker (627)	**441**	**32786**	0	0	236	30835	119	1733
meti-tarski (2615)	1643	804	2520	3345	2578	2027	**2611**	**337**
UltimateAutom (13)	5	0.52	0	0	12	57.19	**13**	**19.23**
zankl (85)	24	9.40	19	13.47	**32**	**7.22**	27	0.43
total (6549)	4853	35239	4740	17775	5331	36849	**5355**	**6658**

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB.

Experimental Results (3/3)

More numerical benchmarks (incl. control-command programs).

	AE		AESDP		AESDPap		AESDPex	
	unsat	time	unsat	time	unsat	time	unsat	time
C (67)	11	0.05	**63**	**39.78**	63	40.01	13	1.18
quadratic (67)	13	0.06	**67**	**14.68**	67	15.44	15	0.08
flyspeck (20)	1	0.00	**19**	26.35	19	26.62	3	0.01
global-opt (14)	2	0.01	**14**	**8.72**	14	8.83	5	0.20
total (168)	27	0.12	**163**	**89.53**	163	90.90	36	1.47

	CVC4		Smtrat		Yices2		Z3	
	unsat	time	unsat	time	unsat	time	unsat	time
C (67)	0	0	0	0	0	0	0	0
quadratic (67)	14	2.46	18	1.26	0	0	25	257.39
flyspeck (20)	6	695.59	9	36.54	10	0.05	9	0.05
global-opt (14)	5	0.12	12	41.18	12	0.16	13	683.45
total (168)	25	698.17	39	78.98	22	0.21	47	940.89

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB.
All times are in seconds.

Conclusion

- Does not outperform state-of-the-art symbolic methods.
- But enables to solve problems out of reach for such methods.
- In particular, numerical problems arising in verification of functional properties of control-command programs.

Conclusion

- Does not outperform state-of-the-art symbolic methods.
- But enables to solve problems out of reach for such methods.
- In particular, numerical problems arising in verification of functional properties of control-command programs.

Future work

- Combination with symbolic (or other numerical) methods.
- Address properties *about* floating-point programs.

Questions

Thanks for your attention!

?