

Conic programming for certified polynomial optimization

Victor Magron, LAAS CNRS

Joint work with

Mohab Safey El Din (Sorbonne University)

Markus Schweighofer (Konstanz University)

Timo de Wolff (TU Braunschweig)

Henning Seidler (TU Berlin)

Vu Trung Hieu (Sorbonne University)

Jie Wang (Chinese Academy of Sciences)



Oberwolfach,
April 2022



Motivation: verification of nonlinear systems

SAFETY of critical parts for **computing** \oplus **physical** devices

Cars

Control Software/Hardware



**Smart
Grids**



**Space
Systems**



... **CAST AS CERTIFIED OPTIMIZATION** \rightsquigarrow **SOLVE OFFLINE**

Input: linear  semidefinite  polynomial 

Output: value + numerical/symbolic/formal **certificate**

Motivation: match with symbolic/formal tools

Positivity certificates

- Stability proofs of critical control systems (Lyapunov)

Motivation: match with symbolic/formal tools

Positivity certificates

- Stability proofs of critical control systems (Lyapunov)
- Certified function evaluation [Chevillard et. al 11]

Motivation: match with symbolic/formal tools

Positivity certificates

- Stability proofs of critical control systems (Lyapunov)
- Certified function evaluation [Chevallard et. al 11]



COQ



HOL-LIGHT

Kepler's conjecture (1611): the max density of sphere packings is $\pi/\sqrt{18}$



Flyspeck : Formalizing the **proof** of **Kepler** [Hales et al. 94]
Certification of thousands of “tight” nonlinear inequalities [Hales et al. 17]

Conic programs & polynomial optimization

NP-hard NON CONVEX Problem $f^* = \inf f(x)$

Theory

(Primal)		(Dual)
$\inf \int f d\mu$		$\sup \lambda$
with μ proba \Rightarrow	INFINITE LP	\Leftarrow with $f - \lambda \geq 0$

Conic programs & polynomial optimization

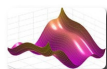
NP-hard NON CONVEX Problem $f^* = \inf f(x)$

Practice

(Primal **Relaxation**)

$$\text{moments } \int x^\alpha d\mu$$

finite number \Rightarrow



SDP

(Dual **Strengthening**)

$$f - \lambda = \text{sum of squares}$$

\Leftarrow fixed degree

LASSERRE'S HIERARCHY of **CONVEX PROBLEMS** $\uparrow f^*$

[Lasserre/Parrilo 01]

degree d & n vars $\Rightarrow \binom{n+2d}{n}$ **SDP** VARIABLES

Numeric solvers \Rightarrow **Approx Certificate**

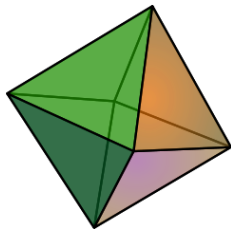


Conic Programming: LP

- Linear Programming (LP):

$$\begin{aligned} \min_{\mathbf{z}} \quad & \mathbf{c}^\top \mathbf{z} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{z} \geq \mathbf{d} \end{aligned}$$

- Linear cost \mathbf{c}
- Linear inequalities “ $\sum_i A_{ij} z_j \geq d_i$ ”

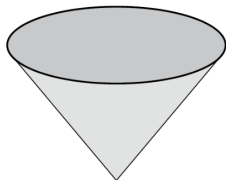


Polyhedron

Conic Programming: SOCP

- Second-order Cone Programming (SOCP):

$$\begin{aligned} \min_{\mathbf{z}} \quad & \mathbf{c}^\top \mathbf{z} \\ \text{s.t.} \quad & z_1 \geq \sqrt{z_2^2 + \cdots + z_n^2} \end{aligned}$$



- Linear cost \mathbf{c}
- convex set defined by the linear/quadratic inequalities

**Second-order
Cone**

Conic Programming: SDP

- Semidefinite Programming (SDP):

$$\begin{aligned} \min_{\mathbf{z}} \quad & \mathbf{c}^\top \mathbf{z} \\ \text{s.t.} \quad & \sum_i \mathbf{F}_i z_i \succcurlyeq \mathbf{F}_0 \end{aligned}$$

- Linear cost \mathbf{c}
- Symmetric matrices $\mathbf{F}_0, \mathbf{F}_i$
- Linear matrix inequalities “ $\mathbf{F} \succcurlyeq 0$ ”
(\mathbf{F} has nonnegative eigenvalues)



Spectrahedron

Deciding nonnegativity

$$\mathbf{x} = (x_1, \dots, x_n)$$

$$f \in \mathbb{Q}[\mathbf{x}]$$

co-NP hard problem: check $f \geq 0$ on \mathbf{K}

NP hard problem: $\min\{f(\mathbf{x}) : \mathbf{x} \in \mathbf{K}\}$

Deciding nonnegativity

$$\mathbf{x} = (x_1, \dots, x_n)$$

co-NP hard problem: check $f \geq 0$ on \mathbf{K}

$$f \in \mathbb{Q}[\mathbf{x}]$$

NP hard problem: $\min\{f(\mathbf{x}) : \mathbf{x} \in \mathbf{K}\}$

1 Unconstrained $\rightsquigarrow \mathbf{K} = \mathbb{R}^n$

2 Constrained $\rightsquigarrow \mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0\} \quad g_j \in \mathbb{Q}[\mathbf{x}]$

$$\deg f, \deg g_j \leq d$$

Deciding nonnegativity

$$\mathbf{x} = (x_1, \dots, x_n)$$

co-NP hard problem: check $f \geq 0$ on \mathbf{K}

$$f \in \mathbb{Q}[\mathbf{x}]$$

NP hard problem: $\min\{f(\mathbf{x}) : \mathbf{x} \in \mathbf{K}\}$

1 Unconstrained $\rightsquigarrow \mathbf{K} = \mathbb{R}^n$

2 Constrained $\rightsquigarrow \mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0\} \quad g_j \in \mathbb{Q}[\mathbf{x}]$

$$\deg f, \deg g_j \leq d$$



[Collins 75] 💡 CAD **doubly exp. in n poly. in d**



[Grigoriev-Vorobjov 88, Basu-Pollack-Roy 98]



Critical points **singly exponential time** $(m+1) \tau d^{O(n)}$

Deciding nonnegativity

💡 Sums of squares (SOS)

$$\sigma = h_1^2 + \dots + h_p^2$$

Deciding nonnegativity

💡 Sums of squares (SOS)

$$\sigma = h_1^2 + \dots + h_p^2$$

HILBERT 17TH PROBLEM: f SOS of rational functions?



[Artin 27] **YES!**

Deciding nonnegativity

💡 Sums of squares (SOS)

$$\sigma = h_1^2 + \dots + h_p^2$$

HILBERT 17TH PROBLEM: f SOS of rational functions?



[Artin 27] **YES!**

💡 [Lasserre/Parrilo 01] **Numerical** solvers compute σ

Semidefinite programming (SDP) \rightsquigarrow **approximate** certificates

Deciding nonnegativity

💡 Sums of squares (SOS)

$$\sigma = h_1^2 + \dots + h_p^2$$

HILBERT 17TH PROBLEM: f SOS of rational functions?



[Artin 27] **YES!**

💡 [Lasserre/Parrilo 01] **Numerical** solvers compute σ

Semidefinite programming (SDP) \rightsquigarrow **approximate** certificates

$$\boxed{\approx \rightarrow =}$$

The Question of Exact Certification

How to go from **approximate** to **exact** certification?

Decomposing nonnegative polynomials

1 Reznick's representation

positive definite form f

[Reznick 95]

$$f = \frac{\sigma}{(x_1^2 + \dots + x_n^2)^D}$$

2 Hilbert-Artin's representation

$f \geq 0$

[Artin 27]

$$f = \frac{\sigma}{h^2}$$

3 Putinar's representation

$f > 0$ on compact K

[Putinar 93]

$$f = \sigma_0 + \sigma_1 g_1 + \dots + \sigma_m g_m$$
$$\deg \sigma_i \leq 2D$$

Decomposing nonnegative polynomials

■ Deciding **polynomial nonnegativity**

$$f(a, b) = a^2 - 2ab + b^2 \geq 0$$

$$\blacksquare f(a, b) = \underbrace{\begin{pmatrix} a & b \end{pmatrix} \begin{pmatrix} z_1 & z_2 \\ z_2 & z_3 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix}}_{\geq 0}$$

$$\blacksquare a^2 - 2ab + b^2 = z_1 a^2 + 2z_2 ab + z_3 b^2 \quad (\mathbf{A} \mathbf{z} = \mathbf{d})$$

$$\blacksquare \begin{pmatrix} z_1 & z_2 \\ z_2 & z_3 \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}}_{\mathbf{F}_1} z_1 + \underbrace{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}_{\mathbf{F}_2} z_2 + \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}}_{\mathbf{F}_3} z_3 \succcurlyeq \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}}_{\mathbf{F}_0}$$

Decomposing nonnegative polynomials

- Choose a cost \mathbf{c} e.g. $(1, 0, 1)$ and solve **SDP**

$$\begin{aligned} \min_{\mathbf{z}} \quad & \mathbf{c}^\top \mathbf{z} \\ \text{s.t.} \quad & \sum_i \mathbf{F}_i z_i \succcurlyeq \mathbf{F}_0, \quad \mathbf{A} \mathbf{z} = \mathbf{d} \end{aligned}$$

- Solution $\begin{pmatrix} z_1 & z_2 \\ z_2 & z_3 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \succcurlyeq 0$ (eigenvalues 0 and 2)

- $a^2 - 2ab + b^2 = (a \ b) \underbrace{\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}}_{\succcurlyeq 0} \begin{pmatrix} a \\ b \end{pmatrix} = (a - b)^2$

- **SUMS OF SQUARES** certificates via **SDP**

Decomposing nonnegative polynomials

4 Circuit polynomial

$$f = b_1 \mathbf{x}^{\alpha(1)} + \cdots + b_t \mathbf{x}^{\alpha(t)} + b_\beta \mathbf{x}^\beta$$

$$b_j > 0 \quad \alpha(j) \in (2\mathbb{N})^n$$

$$\beta = \lambda_1 \alpha(1) + \cdots + \lambda_t \alpha(t) \quad \lambda_j > 0 \text{ and } \lambda_1 + \cdots + \lambda_t = 1$$

Decomposing nonnegative polynomials

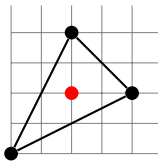
4 Circuit polynomial

$$f = b_1 \mathbf{x}^{\alpha(1)} + \cdots + b_t \mathbf{x}^{\alpha(t)} + b_\beta \mathbf{x}^\beta$$

$$b_j > 0 \quad \alpha(j) \in (2\mathbb{N})^n$$

$$\beta = \lambda_1 \alpha(1) + \cdots + \lambda_t \alpha(t) \quad \lambda_j > 0 \text{ and } \lambda_1 + \cdots + \lambda_t = 1$$

$$f = 1 + x_1^2 x_2^4 + x_1^4 x_2^2 - 3x_1^2 x_2^2$$



Decomposing nonnegative polynomials

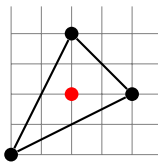
4 Circuit polynomial

$$f = b_1 \mathbf{x}^{\alpha(1)} + \cdots + b_t \mathbf{x}^{\alpha(t)} + b_\beta \mathbf{x}^\beta$$

$$b_j > 0 \quad \alpha(j) \in (2\mathbb{N})^n$$

$$\beta = \lambda_1 \alpha(1) + \cdots + \lambda_t \alpha(t) \quad \lambda_j > 0 \text{ and } \lambda_1 + \cdots + \lambda_t = 1$$

$$f = 1 + x_1^2 x_2^4 + x_1^4 x_2^2 - 3x_1^2 x_2^2$$



$$\text{Circuit number } \Theta_f = \prod_{j=1}^t \left(\frac{b_j}{\lambda_j} \right)^{\lambda_j}$$

Decomposing nonnegative polynomials

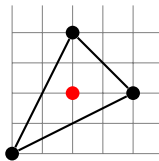
4 Circuit polynomial

$$f = b_1 \mathbf{x}^{\alpha(1)} + \dots + b_t \mathbf{x}^{\alpha(t)} + b_\beta \mathbf{x}^\beta$$

$$b_j > 0 \quad \alpha(j) \in (2\mathbb{N})^n$$

$$\beta = \lambda_1 \alpha(1) + \dots + \lambda_t \alpha(t) \quad \lambda_j > 0 \text{ and } \lambda_1 + \dots + \lambda_t = 1$$

$$f = 1 + x_1^2 x_2^4 + x_1^4 x_2^2 - 3x_1^2 x_2^2$$



$$\text{Circuit number } \Theta_f = \prod_{j=1}^t \left(\frac{b_j}{\lambda_j} \right)^{\lambda_j}$$

Theorem (Illiman-de Wolff 16)

$$f \geq 0 \Leftrightarrow |b_\beta| \leq \Theta_f \text{ or } (b_\beta \geq -\Theta_f, \beta \text{ even})$$

💡 sums of nonnegative circuits computed via geometric programs

Decomposing nonnegative polynomials

5 arithmetic-geometric-mean-exponential (AGE)

$$f = c_1 \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(1)] + \cdots + c_t \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(t)] + \beta \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(0)]$$

$$c_j \in \mathbb{Q}_{>0} \quad \beta \in \mathbb{Q} \quad \boldsymbol{\alpha}(j) \in \mathbb{N}^n$$

Decomposing nonnegative polynomials

5 arithmetic-geometric-mean-exponential (AGE)

$$f = c_1 \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(1)] + \cdots + c_t \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(t)] + \beta \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(0)]$$
$$c_j \in \mathbb{Q}_{>0} \quad \beta \in \mathbb{Q} \quad \boldsymbol{\alpha}(j) \in \mathbb{N}^n$$

relative entropy $D(\mathbf{v}, \mathbf{c}) = \sum_j v_j \log \frac{v_j}{c_j}$

Decomposing nonnegative polynomials

5 arithmetic-geometric-mean-exponential (AGE)

$$f = c_1 \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(1)] + \cdots + c_t \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(t)] + \beta \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(0)]$$
$$c_j \in \mathbb{Q}_{>0} \quad \beta \in \mathbb{Q} \quad \boldsymbol{\alpha}(j) \in \mathbb{N}^n$$

relative entropy $D(\mathbf{v}, \mathbf{c}) = \sum_j v_j \log \frac{v_j}{c_j}$

Theorem (Chandrasekaran-Shah 16)

$$f \geq 0 \Leftrightarrow \exists \mathbf{v} \mid D(\mathbf{v}, \mathbf{c}) \leq \beta \text{ and } \sum_j \boldsymbol{\alpha}(j) v_j = (\mathbf{1} \cdot \mathbf{v}) \boldsymbol{\alpha}(0)$$

💡 **SAGE (sums of AGE)** computed via relative entropy programs

From approximate to exact certificates

APPROXIMATE SOLUTIONS

sum of squares of $a^2 - 2ab + b^2$?



$(1.00001a - 0.99998b)^2!$



$$a^2 - 2ab + b^2 \simeq (1.00001a - 0.99998b)^2$$

$$a^2 - 2ab + b^2 \neq 1.0000200001a^2 - 1.9999799996ab + 0.9999600004b^2$$

$$\boxed{\simeq \rightarrow = ?}$$

Rational SOS decompositions

- Let $f \in \mathbb{R}[x]$ and $f \geq 0$ on \mathbb{R} ($n = 1$)

Theorem

There exist $f_1, f_2 \in \mathbb{R}[x]$ s.t. $f = f_1^2 + f_2^2$.

Rational SOS decompositions

- Let $f \in \mathbb{R}[x]$ and $f \geq 0$ on \mathbb{R} ($n = 1$)

Theorem

There exist $f_1, f_2 \in \mathbb{R}[x]$ s.t. $f = f_1^2 + f_2^2$.

Proof.

$$f = h^2(q + ir)(q - ir)$$



Rational SOS decompositions

- Let $f \in \mathbb{R}[x]$ and $f \geq 0$ on \mathbb{R} ($n = 1$)

Theorem

There exist $f_1, f_2 \in \mathbb{R}[x]$ s.t. $f = f_1^2 + f_2^2$.

Proof.

$$f = h^2(q + ir)(q - ir)$$

□

Examples

$$1 + x + x^2 = \left(x + \frac{1}{2}\right)^2 + \left(\frac{\sqrt{3}}{2}\right)^2$$

$$1 + x + x^2 + x^3 + x^4 = \left(x^2 + \frac{1}{2}x + \frac{1 + \sqrt{5}}{4}\right)^2 + \left(\frac{\sqrt{10 + 2\sqrt{5}} + \sqrt{10 - 2\sqrt{5}}}{4}x + \frac{\sqrt{10 - 2\sqrt{5}}}{4}\right)^2$$

Rational SOS decompositions

- $f \in \mathbb{Q}[x] \cap \overset{\circ}{\Sigma}[x]$ (interior of the SOS cone)

Existence Question

Does there exist $f_i \in \mathbb{Q}[x], c_i \in \mathbb{Q}^{>0}$ s.t. $f = \sum_i c_i f_i^2$?

Rational SOS decompositions

- $f \in \mathbb{Q}[x] \cap \overset{\circ}{\Sigma}[x]$ (interior of the SOS cone)

Existence Question

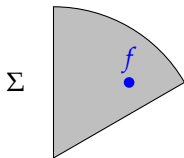
Does there exist $f_i \in \mathbb{Q}[x], c_i \in \mathbb{Q}^{>0}$ s.t. $f = \sum_i c_i f_i^2$?

Examples

$$1 + x + x^2 = \left(x + \frac{1}{2}\right)^2 + \left(\frac{\sqrt{3}}{2}\right)^2 = 1 \left(x + \frac{1}{2}\right)^2 + \frac{3}{4}(1)^2$$

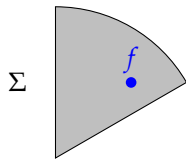
$$1 + x + x^2 + x^3 + x^4 = \left(x^2 + \frac{1}{2}x + \frac{1 + \sqrt{5}}{4}\right)^2 + \left(\frac{\sqrt{10 + 2\sqrt{5}} + \sqrt{10 - 2\sqrt{5}}}{4}x + \frac{\sqrt{10 - 2\sqrt{5}}}{4}\right)^2 = ???$$

Round & Project algorithm [Peyrl-Parrilo 08]




$$f \in \mathring{\Sigma}[\mathbf{x}] \text{ with } \deg f = 2D$$

Round & Project algorithm [Peyrl-Parrilo 08]



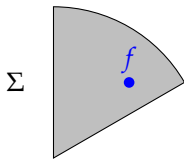
$$f \in \mathring{\Sigma}[\mathbf{x}] \text{ with } \deg f = 2D$$

 Find $\tilde{\mathbf{G}}$ with SDP at tolerance $\tilde{\delta}$ satisfying

$$f(\mathbf{x}) \simeq \mathbf{v}_D^T(\mathbf{x}) \tilde{\mathbf{G}} \mathbf{v}_D(\mathbf{x}) \quad \tilde{\mathbf{G}} \succ 0$$

$\mathbf{v}_D(\mathbf{x})$: vector of monomials of $\deg \leq D$

Round & Project algorithm [Peyrl-Parrilo 08]



$$f \in \mathring{\Sigma}[\mathbf{x}] \text{ with } \deg f = 2D$$

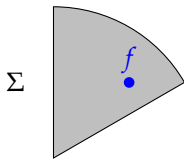
🔄 Find $\tilde{\mathbf{G}}$ with SDP at tolerance $\tilde{\delta}$ satisfying

$$f(\mathbf{x}) \simeq \mathbf{v}_D^T(\mathbf{x}) \tilde{\mathbf{G}} \mathbf{v}_D(\mathbf{x}) \quad \tilde{\mathbf{G}} \succ 0$$

$\mathbf{v}_D(\mathbf{x})$: vector of monomials of $\deg \leq D$

💡 Exact $\mathbf{G} \implies f_\gamma = \sum_{\alpha'+\beta'=\gamma} G_{\alpha',\beta'}$

Round & Project algorithm [Peyrl-Parrilo 08]



$$f \in \mathring{\Sigma}[\mathbf{x}] \text{ with } \deg f = 2D$$

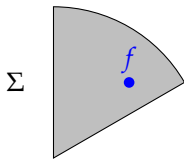
Find $\tilde{\mathbf{G}}$ with SDP at tolerance $\tilde{\delta}$ satisfying

$$f(\mathbf{x}) \simeq \mathbf{v}_D^T(\mathbf{x}) \tilde{\mathbf{G}} \mathbf{v}_D(\mathbf{x}) \quad \tilde{\mathbf{G}} \succ 0$$

$\mathbf{v}_D(\mathbf{x})$: vector of monomials of $\deg \leq D$

$$\text{Exact } \mathbf{G} \implies f_\gamma = \sum_{\alpha'+\beta'=\gamma} G_{\alpha',\beta'} \quad f_{\alpha+\beta} = \sum_{\alpha'+\beta'=\alpha+\beta} G_{\alpha',\beta'}$$

Round & Project algorithm [Peyrl-Parrilo 08]



$$f \in \mathring{\Sigma}[\mathbf{x}] \text{ with } \deg f = 2D$$

Find $\tilde{\mathbf{G}}$ with SDP at tolerance $\tilde{\delta}$ satisfying

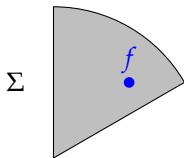
$$f(\mathbf{x}) \simeq \mathbf{v}_D^T(\mathbf{x}) \tilde{\mathbf{G}} \mathbf{v}_D(\mathbf{x}) \quad \tilde{\mathbf{G}} \succ 0$$

$\mathbf{v}_D(\mathbf{x})$: vector of monomials of $\deg \leq D$

Exact $\mathbf{G} \implies f_\gamma = \sum_{\alpha'+\beta'=\gamma} G_{\alpha',\beta'} \quad f_{\alpha+\beta} = \sum_{\alpha'+\beta'=\alpha+\beta} G_{\alpha',\beta'}$

1 Rounding step $\hat{\mathbf{G}} \leftarrow \text{round}(\tilde{\mathbf{G}}, \hat{\delta})$

Round & Project algorithm [Peyrl-Parrilo 08]



$$f \in \mathring{\Sigma}[\mathbf{x}] \text{ with } \deg f = 2D$$

Find $\tilde{\mathbf{G}}$ with SDP at tolerance $\tilde{\delta}$ satisfying

$$f(\mathbf{x}) \simeq \mathbf{v}_D^T(\mathbf{x}) \tilde{\mathbf{G}} \mathbf{v}_D(\mathbf{x}) \quad \tilde{\mathbf{G}} \succ 0$$

$\mathbf{v}_D(\mathbf{x})$: vector of monomials of $\deg \leq D$

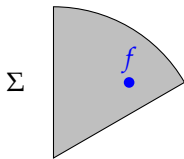
Exact $\mathbf{G} \implies f_\gamma = \sum_{\alpha'+\beta'=\gamma} G_{\alpha',\beta'} \quad f_{\alpha+\beta} = \sum_{\alpha'+\beta'=\alpha+\beta} G_{\alpha',\beta'}$

1 Rounding step $\hat{\mathbf{G}} \leftarrow \text{round}(\tilde{\mathbf{G}}, \hat{\delta})$

2 Projection step

$$G_{\alpha,\beta} \leftarrow \hat{G}_{\alpha,\beta} - \frac{1}{\eta(\alpha+\beta)} \left(\sum_{\alpha'+\beta'=\alpha+\beta} \hat{G}_{\alpha',\beta'} - f_{\alpha+\beta} \right)$$

Round & Project algorithm [Peyrl-Parrilo 08]



$$f \in \dot{\Sigma}[\mathbf{x}] \text{ with } \deg f = 2D$$

Find $\tilde{\mathbf{G}}$ with SDP at tolerance $\tilde{\delta}$ satisfying

$$f(\mathbf{x}) \simeq \mathbf{v}_D^T(\mathbf{x}) \tilde{\mathbf{G}} \mathbf{v}_D(\mathbf{x}) \quad \tilde{\mathbf{G}} \succ 0$$

$\mathbf{v}_D(\mathbf{x})$: vector of monomials of $\deg \leq D$

Exact $\mathbf{G} \implies f_\gamma = \sum_{\alpha'+\beta'=\gamma} G_{\alpha',\beta'} \quad f_{\alpha+\beta} = \sum_{\alpha'+\beta'=\alpha+\beta} G_{\alpha',\beta'}$

1 Rounding step $\hat{\mathbf{G}} \leftarrow \text{round}(\tilde{\mathbf{G}}, \tilde{\delta})$

2 Projection step

$$G_{\alpha,\beta} \leftarrow \hat{G}_{\alpha,\beta} - \frac{1}{\eta(\alpha+\beta)} (\sum_{\alpha'+\beta'=\alpha+\beta} \hat{G}_{\alpha',\beta'} - f_{\alpha+\beta})$$

Small enough $\tilde{\delta}, \hat{\delta} \implies f(\mathbf{x}) = \mathbf{v}_D^T(\mathbf{x}) \mathbf{G} \mathbf{v}_D(\mathbf{x})$ and $\mathbf{G} \succcurlyeq 0$

One Answer when $\mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0\}$

💡 Hybrid **SYMBOLIC/NUMERIC** methods

📄 M.-Allamigeon-Gaubert-Werner 14

$$f \simeq \tilde{\sigma}_0 + \tilde{\sigma}_1 g_1 + \cdots + \tilde{\sigma}_m g_m$$

$$u = f - \tilde{\sigma}_0 + \tilde{\sigma}_1 g_1 + \cdots + \tilde{\sigma}_m g_m$$

One Answer when $\mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0\}$

💡 Hybrid **SYMBOLIC/NUMERIC** methods

📄 M.-Allamigeon-Gaubert-Werner 14

$$f \simeq \tilde{\sigma}_0 + \tilde{\sigma}_1 g_1 + \cdots + \tilde{\sigma}_m g_m$$

$$u = f - \tilde{\sigma}_0 + \tilde{\sigma}_1 g_1 + \cdots + \tilde{\sigma}_m g_m$$

$$\boxed{\simeq \rightarrow =}$$

💡 $\forall \mathbf{x} \in [0, 1]^n, u(\mathbf{x}) \leq -\varepsilon$

$\min_{\mathbf{K}} f \geq \varepsilon$ when $\varepsilon \rightarrow 0$

Compact $\mathbf{K} \subseteq [0, 1]^n$



From Approximate to Exact Solutions

Win TWO-PLAYER GAME



sum of squares of f ?



\approx Output!



From Approximate to Exact Solutions

Win TWO-PLAYER GAME



💡 **Hybrid** Symbolic/Numeric Algorithms

sum of squares of $f + \varepsilon$?

\approx Output!



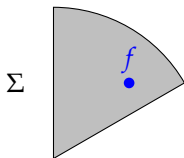
Error Compensation



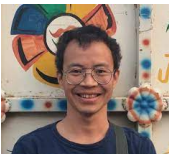
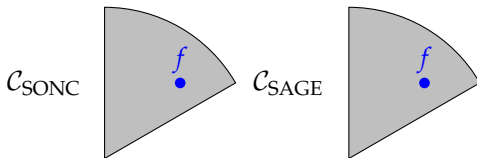
$\approx \rightarrow =$

From Approximate to Exact Solutions

Exact SOS



Exact SONC/SAGE



Exact optimization via SOS: [RealCertify](#)

Maple & arbitrary precision SDP solver SDPA-GMP

[Nakata 10]

univsos $n = 1$

multivsos $n > 1$

Exact optimization via SONC/SAGE: [POEM](#) [SONCSOCP](#)

Python (SymPy) & geometric programming/relative entropy ECOS

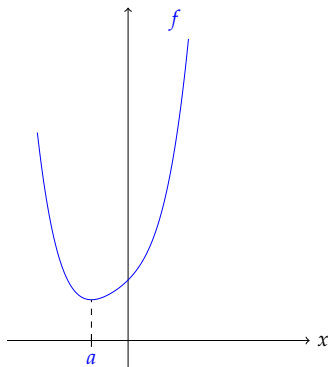
[Domahidi-Chu-Boyd 13]

Julia & second-order programming Mosek

[Andersen-Andersen 00]

univsos1: outline [Schweighofer 99]

$f \in \mathbb{Q}[x]$ and $f > 0$
Minimizer a may not be in $\mathbb{Q} \dots$



$$f = 1 + x + x^2 + x^3 + x^4$$

$$a = \frac{5}{4(135+60\sqrt{6})^{1/3}} - \frac{4(135+60\sqrt{6})^{1/3}}{12} - \frac{1}{4}$$

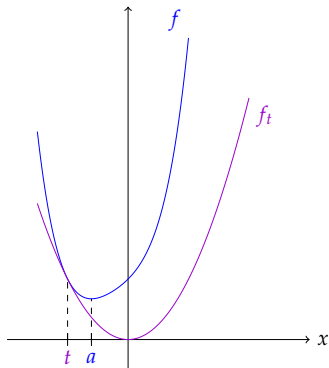
univsos1: outline [Schweighofer 99]

$f \in \mathbb{Q}[x]$ and $f > 0$

Minimizer a may not be in $\mathbb{Q} \dots$

💡 Find $f_t \in \mathbb{Q}[x]$ s.t. :

- $\deg f_t \leq 2$
- $f_t \geq 0$
- $f \geq f_t$
- $f - f_t$ has a root $t \in \mathbb{Q}$



$$f = 1 + x + x^2 + x^3 + x^4$$

$$a = \frac{5}{4(135+60\sqrt{6})^{1/3}} - \frac{4(135+60\sqrt{6})^{1/3}}{12} - \frac{1}{4}$$

$$f_t = x^2$$

$$t = -1$$

univos1: outline [Schweighofer 99]

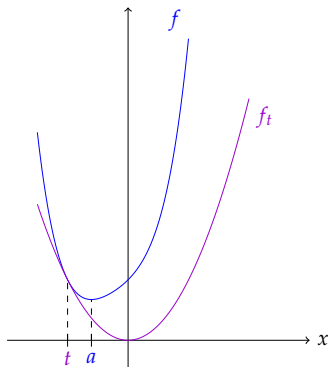
$f \in \mathbb{Q}[x]$ and $f > 0$

Minimizer a may not be in $\mathbb{Q} \dots$

💡 Square-free decomposition:

$$f - f_t = gh^2$$

- $\deg g \leq \deg f - 2$
- $g > 0$
- Do it again on g



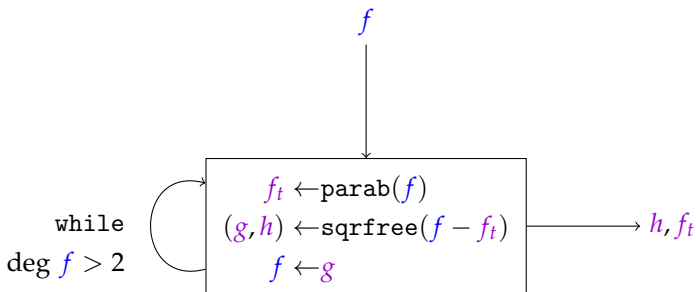
$$f = 1 + x + x^2 + x^3 + x^4$$

$$f_t = x^2$$

$$f - f_t = (x^2 + 2x + 1)(x + 1)^2$$

univos1: algorithm [Schweighofer 99]

- **Input:** $f \geq 0 \in \mathbb{Q}[x]$ of degree $d \geq 2$
- **Output:** SOS decomposition with coefficients in \mathbb{Q}



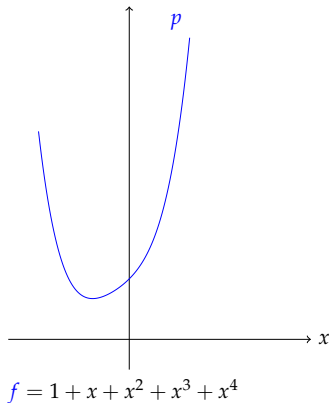
univos1: output bitsize

Theorem [M.-Safey El Din-Schweighofer 17]

Let $0 < f \in \mathbb{Q}[x]$ with bitsize τ , $\deg f = d$.

The output bitsize of univos1 on f is $\mathcal{O}\left(\left(\frac{d}{2}\right)^{\frac{3d}{2}} \tau\right)$.

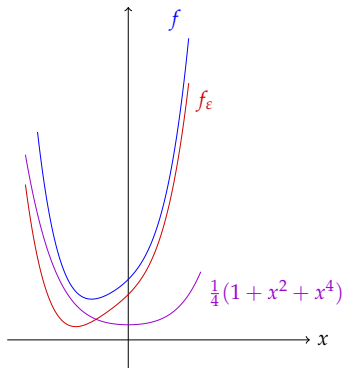
$$f \in \mathbb{Q}[x], \deg f = d = 2k, f > 0$$



$$f \in \mathbb{Q}[x], \deg f = d = 2k, f > 0$$

💡 **PERTURB:** find $\varepsilon \in \mathbb{Q}$ s.t.

$$f_\varepsilon := f - \varepsilon \sum_{i=0}^k x^{2i} > 0$$



$$f = 1 + x + x^2 + x^3 + x^4$$

$$\varepsilon = \frac{1}{4}$$

$$f > \frac{1}{4}(1 + x^2 + x^4)$$

$f \in \mathbb{Q}[x]$, $\deg f = d = 2k$, $f > 0$

💡 **PERTURB**: find $\varepsilon \in \mathbb{Q}$ s.t.

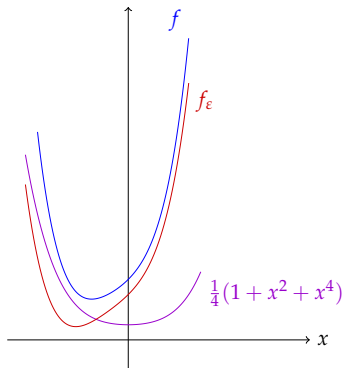
$$f_\varepsilon := f - \varepsilon \sum_{i=0}^k x^{2i} > 0$$

💡 **Root isolation or SDP**:

$$f - \varepsilon \sum_{i=0}^k x^{2i} = \tilde{\sigma} + u$$

💡 **ABSORB**: small enough u_i

$\implies \varepsilon \sum_{i=0}^k x^{2i} + u$ SOS



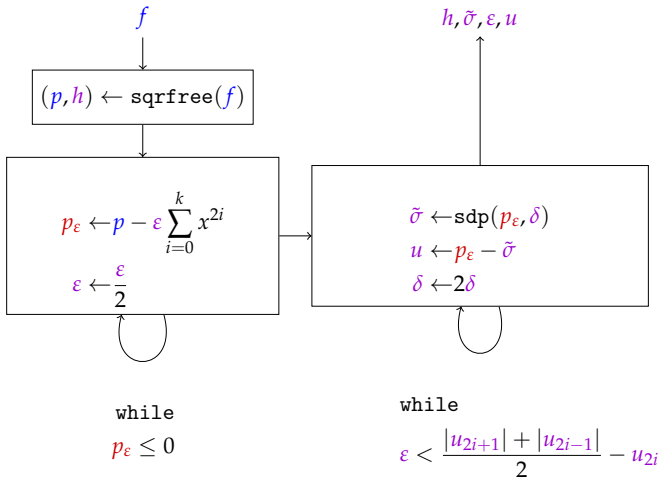
$$f = 1 + x + x^2 + x^3 + x^4$$

$$\varepsilon = \frac{1}{4}$$

$$f > \frac{1}{4}(1 + x^2 + x^4)$$

univsos2: algorithm

- **Input** $f \geq 0 \in \mathbb{Q}[x]$ of degree $d \geq 2$, $\varepsilon \in \mathbb{Q}^{>0}$, $\delta \in \mathbb{N}^{>0}$
- **Output**: SOS decomposition with coefficients in \mathbb{Q}



univsos2: absorb

$$\text{💡 } x = \frac{1}{2} [(x+1)^2 - 1 - x^2]$$

$$\text{💡 } -x = \frac{1}{2} [(x-1)^2 - 1 - x^2]$$

univos2: absorb

$$\text{💡 } x = \frac{1}{2} [(x+1)^2 - 1 - x^2]$$

$$\text{💡 } -x = \frac{1}{2} [(x-1)^2 - 1 - x^2]$$

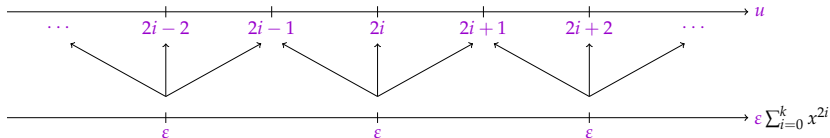
$$u_{2i+1} x^{2i+1} = \frac{|u_{2i+1}|}{2} [(x^{i+1} + \text{sgn}(u_{2i+1})x^i)^2 - x^{2i} - x^{2i+2}]$$

univsos2: absorb

$$\text{💡 } x = \frac{1}{2} [(x+1)^2 - 1 - x^2]$$

$$\text{💡 } -x = \frac{1}{2} [(x-1)^2 - 1 - x^2]$$

$$u_{2i+1} x^{2i+1} = \frac{|u_{2i+1}|}{2} [(x^{i+1} + \text{sgn}(u_{2i+1})x^i)^2 - x^{2i} - x^{2i+2}]$$

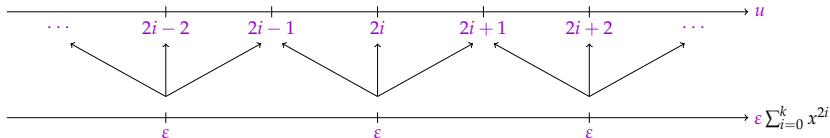


univsos2: absorb

$$\text{💡 } x = \frac{1}{2} [(x+1)^2 - 1 - x^2]$$

$$\text{💡 } -x = \frac{1}{2} [(x-1)^2 - 1 - x^2]$$

$$u_{2i+1} x^{2i+1} = \frac{|u_{2i+1}|}{2} [(x^{i+1} + \text{sgn}(u_{2i+1})x^i)^2 - x^{2i} - x^{2i+2}]$$



$$\epsilon \geq \frac{|u_{2i+1}| + |u_{2i-1}|}{2} - u_{2i} \implies \epsilon \sum_{i=0}^k x^{2i} + u \text{ SOS}$$

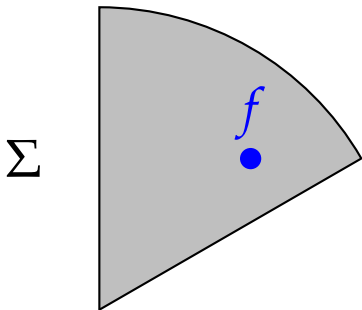
univos2: output bitsize

Theorem [M.-Safey El Din-Schweighofer 17]

Let $0 < f \in \mathbb{Q}[x]$ with bitsize τ , $\deg f = d$.

The output bitsize of `univos2` on f is $\mathcal{O}(d^3 + d^2\tau)$.

multisos: perturbation



PERTURBATION idea

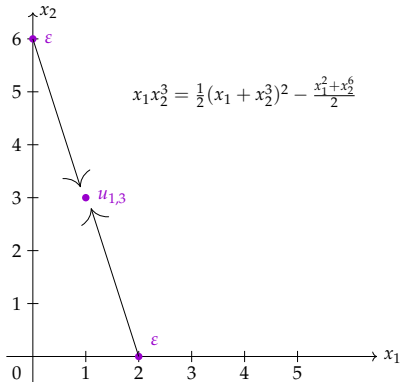
💡 Approximate SOS Decomposition

$$f(\mathbf{x}) - \varepsilon \sum_{\alpha \in \mathcal{P}/2} \mathbf{x}^{2\alpha} = \tilde{\sigma} + u$$

multivsos: absorb

$$f(x) - \varepsilon \sum_{\alpha \in \mathcal{P}/2} x^{2\alpha} = \tilde{\sigma} + u$$

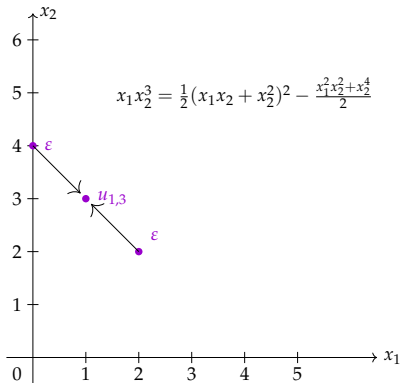
Choice of \mathcal{P} ?



multivsos: absorb

$$f(x) - \varepsilon \sum_{\alpha \in \mathcal{P}/2} x^{2\alpha} = \tilde{\sigma} + u$$

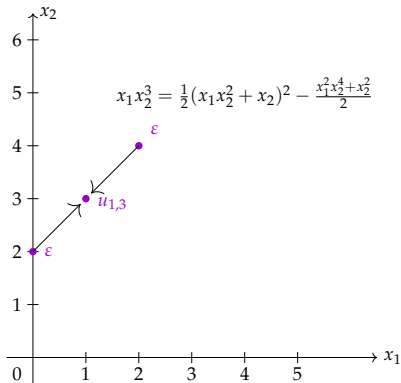
Choice of \mathcal{P} ?



multivsos: absorb

$$f(x) - \varepsilon \sum_{\alpha \in \mathcal{P}/2} x^{2\alpha} = \tilde{\sigma} + u$$

Choice of \mathcal{P} ?

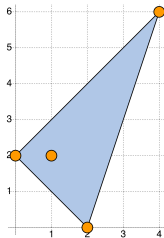


$$f(x) - \varepsilon \sum_{\alpha \in \mathcal{P}/2} x^{2\alpha} = \tilde{\sigma} + u$$

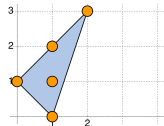
Choice of \mathcal{P} ?

$$f = 4x_1^4x_2^6 + x_1^2 - x_1x_2^2 + x_2^2$$
$$\text{spt}(f) = \{(4,6), (2,0), (1,2), (0,2)\}$$

Newton Polytope $\mathcal{P} = \text{conv}(\text{spt}(f))$

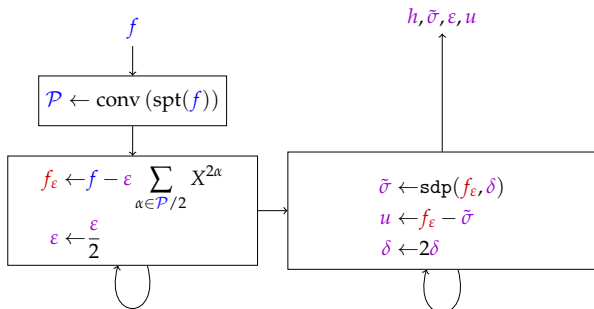


Squares in SOS decomposition $\subseteq \frac{\mathcal{P}}{2} \cap \mathbb{N}^n$
[Reznick 78]



multisos: algorithm

- **Input** $f \in \mathbb{Q}[\mathbf{x}] \cap \overset{\circ}{\Sigma}[\mathbf{x}]$ of degree d , $\varepsilon \in \mathbb{Q}^{>0}$, $\delta \in \mathbb{N}^{>0}$
- **Output**: SOS decomposition with coefficients in \mathbb{Q}



while
 $f_\varepsilon \leq 0$

while
 $u + \varepsilon \sum_{\alpha \in \mathcal{P}/2} X^{2\alpha} \notin \Sigma$

multivsos: output bitsize

Theorem (Exact Certification Cost in $\mathring{\Sigma}$)

$f \in \mathbb{Q}[\mathbf{x}] \cap \mathring{\Sigma}[\mathbf{x}]$ with $\deg f = d = 2k$ and bit size τ

\implies multivsos terminates with SOS output of bit size $\tau d^{d^{\mathcal{O}(n)}}$

multivsos: output bitsize

Theorem (Exact Certification Cost in $\mathring{\Sigma}$)

$f \in \mathbb{Q}[\mathbf{x}] \cap \mathring{\Sigma}[\mathbf{x}]$ with $\deg f = d = 2k$ and bit size τ

\implies multivsos terminates with SOS output of bit size $\tau d^{d^{\mathcal{O}(n)}}$

Algorithm Reznicksos

f positive definite form has **Reznick's** representation:

$$f = \frac{\sigma}{(x_1 + \cdots + x_n)^{2D}} \quad \text{with } \sigma \in \Sigma[\mathbf{x}]$$

Algorithm Reznicksos

f positive definite form has **Reznick's** representation:

$$f = \frac{\sigma}{(x_1 + \cdots + x_n)^{2D}} \quad \text{with } \sigma \in \Sigma[\mathbf{x}]$$

Theorem

$$f(x_1 + \cdots + x_n)^{2D} \in \Sigma[\mathbf{x}] \implies f(x_1 + \cdots + x_n)^{2D+2} \in \overset{\circ}{\Sigma}[\mathbf{x}]$$

Algorithm Reznicksos

f positive definite form has **Reznick's** representation:

$$f = \frac{\sigma}{(x_1 + \cdots + x_n)^{2D}} \quad \text{with } \sigma \in \Sigma[\mathbf{x}]$$

Theorem

$$f(x_1 + \cdots + x_n)^{2D} \in \Sigma[\mathbf{x}] \implies f(x_1 + \cdots + x_n)^{2D+2} \in \overset{\circ}{\Sigma}[\mathbf{x}]$$

💡 Apply Algorithm multivsos on $f(x_1 + \cdots + x_n)^{2D+2}$

Algorithm Reznicksos

f positive definite form has **Reznick's** representation:

$$f = \frac{\sigma}{(x_1 + \cdots + x_n)^{2D}} \quad \text{with } \sigma \in \Sigma[\mathbf{x}]$$

Theorem

$$f(x_1 + \cdots + x_n)^{2D} \in \Sigma[\mathbf{x}] \implies f(x_1 + \cdots + x_n)^{2D+2} \in \overset{\circ}{\Sigma}[\mathbf{x}]$$

💡 Apply Algorithm multivsos on $f(x_1 + \cdots + x_n)^{2D+2}$

Theorem (Exact Certification Cost of Reznick's representations)

$f \in \mathbb{Q}[\mathbf{x}]$ positive definite form with $\deg f = d$ and bit size τ

$$\implies D \leq 2^{2^{\mathcal{O}(\tau \cdot (4d+6)3n+3)}}$$

Algorithm Putinar's

Assumption: $\exists i$ s.t. $g_i = 1 - \|\mathbf{x}\|_2^2$

$f > 0$ on $\mathbf{K} := \{\mathbf{x} : g_j(\mathbf{x}) \geq 0\}$ has **Putinar's** representation:

$$f = \sigma_0 + \sum_j \sigma_j g_j \quad \text{with } \sigma_j \in \Sigma[\mathbf{x}], \deg \sigma_j \leq 2D$$

Algorithm Putinarsos

Assumption: $\exists i$ s.t. $g_i = 1 - \|\mathbf{x}\|_2^2$
 $f > 0$ on $\mathbf{K} := \{\mathbf{x} : g_j(\mathbf{x}) \geq 0\}$ has **Putinar's** representation:

$$f = \sigma_0 + \sum_j \sigma_j g_j \quad \text{with } \sigma_j \in \Sigma[\mathbf{x}], \deg \sigma_j \leq 2D$$

Theorem [M.-Safey El Din 18]

$$f = \check{\sigma}_0 + \sum_j \check{\sigma}_j g_j$$

with $\check{\sigma}_j \in \check{\Sigma}[\mathbf{x}], \deg \check{\sigma}_j \leq 2D$

Algorithm Putinarsos

Assumption: $\exists i$ s.t. $g_i = 1 - \|\mathbf{x}\|_2^2$
 $f > 0$ on $\mathbf{K} := \{\mathbf{x} : g_j(\mathbf{x}) \geq 0\}$ has **Putinar's** representation:

$$f = \sigma_0 + \sum_j \sigma_j g_j \quad \text{with } \sigma_j \in \Sigma[\mathbf{x}], \deg \sigma_j \leq 2D$$

Theorem [M.-Safey El Din 18]

$$f = \check{\sigma}_0 + \sum_j \check{\sigma}_j g_j$$

with $\check{\sigma}_j \in \check{\Sigma}[\mathbf{x}], \deg \check{\sigma}_j \leq 2D$

💡 **ABSORB** as in Algorithm multivsos: $u = f_\varepsilon - \check{\sigma}_0 - \sum_j \check{\sigma}_j g_j$

Algorithm Putinarsos

Assumption: $\exists i$ s.t. $g_i = 1 - \|\mathbf{x}\|_2^2$
 $f > 0$ on $\mathbf{K} := \{\mathbf{x} : g_j(\mathbf{x}) \geq 0\}$ has **Putinar's** representation:

$$f = \sigma_0 + \sum_j \sigma_j g_j \quad \text{with } \sigma_j \in \Sigma[\mathbf{x}], \deg \sigma_j \leq 2D$$

Theorem [M.-Safey El Din 18]

$$f = \tilde{\sigma}_0 + \sum_j \tilde{\sigma}_j g_j$$

$$\text{with } \tilde{\sigma}_j \in \tilde{\Sigma}[\mathbf{x}], \deg \tilde{\sigma}_j \leq 2D$$

💡 **ABSORB** as in Algorithm multivsos: $u = f_\varepsilon - \tilde{\sigma}_0 - \sum_j \tilde{\sigma}_j g_j$

$$\text{OUTPUT BIT SIZE} = \boxed{D^{D^{\mathcal{O}(n)}} \text{ with } \log D = \mathcal{O}(2^\tau d^{n c_K})}$$

gradsos: SOS modulo the gradient ideal

Theorem [M.-Safey El Din-Vu 21]

Let $f \in \mathbb{Q}[\mathbf{x}]$. Assume that

- 1 The infimum f^* is attained
- 2 The gradient ideal $\mathcal{I}_{\text{grad}}(f)$ is zero-dimensional radical

$\implies f$ is nonnegative over $\mathbb{R}^n \iff f$ is SOS modulo $\mathcal{I}_{\text{grad}}(f)$,

i.e. there exists $q_j, \phi_i \in \mathbb{Q}[\mathbf{x}]$, $g_i \in \mathcal{I}_{\text{grad}}(f)$ s.t. $f = \sum_{j=1}^s q_j^2 + \sum_{i=1}^m \phi_i g_i$.

gradsos: SOS modulo the gradient ideal

Theorem [M.-Safey El Din-Vu 21]

Let $f \in \mathbb{Q}[\mathbf{x}]$. Assume that

- 1 The infimum f^* is attained
- 2 The gradient ideal $\mathcal{I}_{\text{grad}}(f)$ is zero-dimensional radical

$\implies f$ is nonnegative over $\mathbb{R}^n \iff f$ is SOS modulo $\mathcal{I}_{\text{grad}}(f)$,

i.e. there exists $q_j, \phi_i \in \mathbb{Q}[\mathbf{x}]$, $g_i \in \mathcal{I}_{\text{grad}}(f)$ s.t. $f = \sum_{j=1}^s q_j^2 + \sum_{i=1}^m \phi_i g_i$.

Idea of the proof: reduce the number of variables from n to 1.

gradsos: SOS modulo the gradient ideal

Theorem [M.-Safey El Din-Vu 21]

Let $f \in \mathbb{Q}[\mathbf{x}]$. Assume that

- 1 The infimum f^* is attained
- 2 The gradient ideal $\mathcal{I}_{\text{grad}}(f)$ is zero-dimensional radical

$\implies f$ is nonnegative over $\mathbb{R}^n \iff f$ is SOS modulo $\mathcal{I}_{\text{grad}}(f)$,

i.e. there exists $q_j, \phi_i \in \mathbb{Q}[\mathbf{x}]$, $g_i \in \mathcal{I}_{\text{grad}}(f)$ s.t. $f = \sum_{j=1}^s q_j^2 + \sum_{i=1}^m \phi_i g_i$.

Idea of the proof: reduce the number of variables from n to 1.

Robinson and Scheiderer's polynomials satisfy both conditions

$$f_R = x_1^6 + x_2^6 - x_1^4 x_2^2 + 3x_1^2 x_2^2 - x_1^2 x_2^4 - x_1^4 - x_2^4 - x_1^2 - x_2^2 + 1$$

$$f_S = x_1^4 + x_1 x_2^3 + x_2^4 + 3x_1^2 x_2 + 4x_1 x_2^2 + 2x_1^2 - x_1 - x_2 + 1$$

\implies They are SOS modulo their gradient ideals over $\mathbb{Q}[\mathbf{x}]$

gradsos: algorithm and bitsize

1: Find a 0-dim rational parametrization of $V_{\text{grad}}(f)$:

$$\left\{ w, \frac{\kappa_1}{w'}, \dots, \frac{\kappa_n}{w'} \right\}$$

2: $h := (w')^d f(x_1, \frac{\kappa_2}{w'}, \dots, \frac{\kappa_n}{w'})$

3: Find ϕ_1, \dots, ϕ_n satisfying

$$(w')^d f - h = \sum_{i=1}^n \phi_i \left(x_i - \frac{\kappa_i}{w'} \right)$$

by using divisions of univariate polynomials

4: Find an SOS decomposition of h by using `univsos1` or `univsos2`

gradsos: algorithm and bitsize

1: Find a 0-dim rational parametrization of $V_{\text{grad}}(f)$:

$$\left\{ w, \frac{\kappa_1}{w'}, \dots, \frac{\kappa_n}{w'} \right\}$$

2: $h := (w')^d f(x_1, \frac{\kappa_2}{w'}, \dots, \frac{\kappa_n}{w'})$

3: Find ϕ_1, \dots, ϕ_n satisfying

$$(w')^d f - h = \sum_{i=1}^n \phi_i(x_i - \frac{\kappa_i}{w'})$$

by using divisions of univariate polynomials

4: Find an SOS decomposition of h by using `univsos1` or `univsos2`

Theorem

With $f \in \mathbb{Q}[x]$ with $\deg f = d$ and bitsize τ

gradsos terminates with output bitsize $\tilde{\mathcal{O}}((\tau + n + d)d^{3n+2})$

Algorithm optsonc: numerical steps

SONC (SUMS OF NONNEGATIVE CIRCUITS)

- **Input** $f = \sum_{\alpha} b_{\alpha} x^{\alpha}$ of degree d , $\hat{\delta} \in \mathbb{Q}^{>0}$, $\tilde{\delta} \in \mathbb{Q}^{>0}$
Monomial squares = $\Lambda(f)$ Complement = $\Gamma(f)$

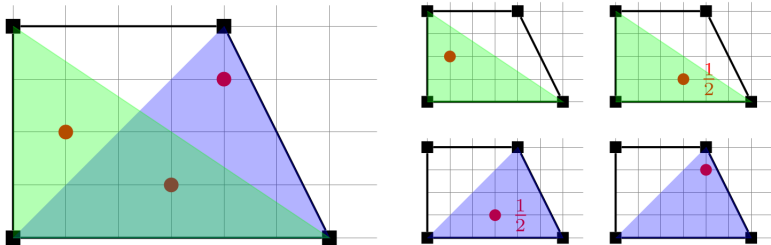
Algorithm optsonc: numerical steps

SONC (SUMS OF NONNEGATIVE CIRCUITS)

- **Input** $f = \sum_{\alpha} b_{\alpha} x^{\alpha}$ of degree d , $\hat{\delta} \in \mathbb{Q}^{>0}$, $\tilde{\delta} \in \mathbb{Q}^{>0}$
Monomial squares = $\Lambda(f)$ Complement = $\Gamma(f)$

1 **Cover** each $\beta \in \Gamma(f)$ to get nonnegative circuit f_{β}

$$\implies \lambda^{\beta} \geq 0 \text{ with } \sum_{\alpha \in \Lambda(f)} \lambda_{\alpha}^{\beta} \cdot \alpha = \beta$$



Algorithm optsonc: numerical steps

■ **Input** $f = \sum_{\alpha} b_{\alpha} \mathbf{x}^{\alpha}$, $\hat{\delta}$, $\tilde{\delta}$

2 **Numerical** resolution of GEOMETRIC PROGRAM

$$\begin{aligned} f_{\text{SONC}} &= \min_{G > 0} \sum_{\beta \in \Gamma(f)} G_{\beta,0} \\ \text{s.t.} \quad &\sum_{\beta \in \Gamma(f)} G_{\beta,\alpha} \leq b_{\alpha}, \quad \alpha \in \Lambda(f), \alpha \neq \mathbf{0} \\ &\prod_{\alpha \in \text{Cov}^{\beta}} \left(\frac{G_{\beta,\alpha}}{\lambda_{\alpha}^{\beta}} \right)^{\lambda_{\alpha}^{\beta}} = -b_{\beta}, \quad \beta \in \Gamma(f) \end{aligned}$$

Algorithm optsonc: symbolic steps

■ **Input** $f = \sum_{\alpha} b_{\alpha} \mathbf{x}^{\alpha}$, $\hat{\delta}$, $\tilde{\delta}$

GEOMETRIC PROGRAM provides “IN THEORY”

$$f_{\beta} = \sum_{\alpha} G_{\beta,\alpha} \cdot \mathbf{x}^{\alpha} + b_{\beta} \mathbf{x}^{\beta}, f + \sum_{\beta} G_{\beta,0} - b_0 = \sum_{\beta} f_{\beta} \geq 0$$

Algorithm options: symbolic steps

■ **Input** $f = \sum_{\alpha} b_{\alpha} \mathbf{x}^{\alpha}$, $\hat{\delta}$, $\tilde{\delta}$

GEOMETRIC PROGRAM provides “IN THEORY”

$$f_{\beta} = \sum_{\alpha} G_{\beta,\alpha} \cdot \mathbf{x}^{\alpha} + b_{\beta} \mathbf{x}^{\beta}, f + \sum_{\beta} G_{\beta,0} - b_0 = \sum_{\beta} f_{\beta} \geq 0$$

Tolerance $\tilde{\delta} \implies$ “IN PRACTICE” \tilde{G} violates the constraints

Algorithm optsonc: symbolic steps

■ **Input** $f = \sum_{\alpha} b_{\alpha} \mathbf{x}^{\alpha}$, $\hat{\delta}$, $\tilde{\delta}$

GEOMETRIC PROGRAM provides “IN THEORY”

$$f_{\beta} = \sum_{\alpha} G_{\beta,\alpha} \cdot \mathbf{x}^{\alpha} + b_{\beta} \mathbf{x}^{\beta}, f + \sum_{\beta} G_{\beta,0} - b_0 = \sum_{\beta} f_{\beta} \geq 0$$

Tolerance $\tilde{\delta} \implies$ “IN PRACTICE” \tilde{G} violates the constraints

3 **Rounding step** $\hat{G} \leftarrow \text{round}(\tilde{G}, \hat{\delta})$

Algorithm options: symbolic steps

■ **Input** $f = \sum_{\alpha} b_{\alpha} \mathbf{x}^{\alpha}$, $\hat{\delta}$, $\tilde{\delta}$

GEOMETRIC PROGRAM provides “IN THEORY”

$$f_{\beta} = \sum_{\alpha} G_{\beta,\alpha} \cdot \mathbf{x}^{\alpha} + b_{\beta} \mathbf{x}^{\beta}, f + \sum_{\beta} G_{\beta,0} - b_0 = \sum_{\beta} f_{\beta} \geq 0$$

Tolerance $\tilde{\delta} \implies$ “IN PRACTICE” \tilde{G} violates the constraints

3 **Rounding step** $\hat{G} \leftarrow \text{round}(\tilde{G}, \hat{\delta})$

4 **Projection step**

$$G_{\beta,\alpha} \leftarrow b_{\alpha} \cdot \hat{G}_{\beta,\alpha} / \sum_{\beta'} \hat{G}_{\beta',\alpha}$$

$$\tilde{G}_{\beta,0} \leftarrow \lambda_0^{\beta} \left(-b_{\beta} \cdot \prod_{\alpha} \left(\frac{\lambda_{\alpha}^{\beta}}{G_{\beta,\alpha}} \right)^{\lambda_{\alpha}^{\beta}} \right)^{\frac{1}{\lambda_0^{\beta}}}$$

$$\hat{G}_{\beta,0} \leftarrow \text{round} \uparrow (\tilde{G}_{\beta,0}, \hat{\delta})$$

Algorithm options: symbolic steps

■ **Input** $f = \sum_{\alpha} b_{\alpha} x^{\alpha}$, $\hat{\delta}$, $\tilde{\delta}$

GEOMETRIC PROGRAM provides “IN THEORY”

$$f_{\beta} = \sum_{\alpha} G_{\beta,\alpha} \cdot x^{\alpha} + b_{\beta} x^{\beta}, f + \sum_{\beta} G_{\beta,0} - b_0 = \sum_{\beta} f_{\beta} \geq 0$$

Tolerance $\tilde{\delta} \implies$ “IN PRACTICE” \tilde{G} violates the constraints


3 **Rounding step** $\hat{G} \leftarrow \text{round}(\tilde{G}, \hat{\delta})$

4 **Projection step**

$$G_{\beta,\alpha} \leftarrow b_{\alpha} \cdot \hat{G}_{\beta,\alpha} / \sum_{\beta'} \hat{G}_{\beta',\alpha}$$

$$\tilde{G}_{\beta,0} \leftarrow \lambda_0^{\beta} \left(-b_{\beta} \cdot \prod_{\alpha} \left(\frac{\lambda_{\alpha}^{\beta}}{G_{\beta,\alpha}} \right)^{\lambda_{\alpha}^{\beta}} \right)^{\frac{1}{\lambda_0^{\beta}}}$$

$$\hat{G}_{\beta,0} \leftarrow \text{round} \uparrow (\tilde{G}_{\beta,0}, \hat{\delta})$$

 $f \geq b_0 - \sum_{\beta} \hat{G}_{\beta,0}$

Algorithm optsonc with SOCP

Averages of distinct rational points in M

$$A(M) := \left\{ \frac{1}{2}(\mathbf{v} + \mathbf{w}) \mid \mathbf{v} \neq \mathbf{w}, \mathbf{v}, \mathbf{w} \in M \right\}$$

M is an \mathcal{A} -rational mediated set if $\mathcal{A} \subseteq M \subseteq A(M) \cup \mathcal{A}$

Algorithm optsonc with SOCP

Averages of distinct rational points in M

$$A(M) := \left\{ \frac{1}{2}(\mathbf{v} + \mathbf{w}) \mid \mathbf{v} \neq \mathbf{w}, \mathbf{v}, \mathbf{w} \in M \right\}$$

M is an \mathcal{A} -rational mediated set if $\mathcal{A} \subseteq M \subseteq A(M) \cup \mathcal{A}$

Theorem [Wang-Magron 20] for a circuit f

There exists a $\Lambda(f)$ -rational mediated set M containing β

Algorithm optsonc with SOCP

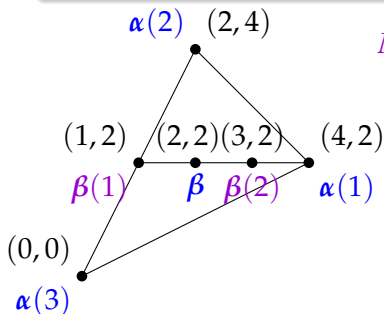
Averages of distinct rational points in M

$$A(M) := \left\{ \frac{1}{2}(\mathbf{v} + \mathbf{w}) \mid \mathbf{v} \neq \mathbf{w}, \mathbf{v}, \mathbf{w} \in M \right\}$$

M is an \mathcal{A} -rational mediated set if $\mathcal{A} \subseteq M \subseteq A(M) \cup \mathcal{A}$

Theorem [Wang-Magron 20] for a circuit f

There exists a $\Lambda(f)$ -rational mediated set M containing β



$$M = \{\alpha(1), \alpha(2), \alpha(3), \beta, \beta(1), \beta(2)\}$$

$$\mathbf{u} \in M \setminus \Lambda(f) \implies \mathbf{u} = (\mathbf{v} + \mathbf{w})/2$$

Algorithm optsonc with SOCP

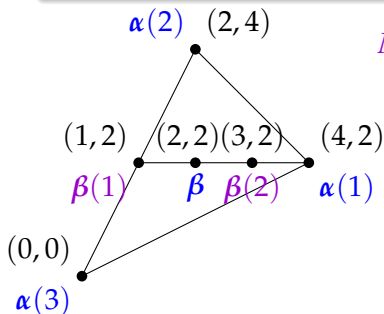
Averages of distinct rational points in M

$$A(M) := \left\{ \frac{1}{2}(\mathbf{v} + \mathbf{w}) \mid \mathbf{v} \neq \mathbf{w}, \mathbf{v}, \mathbf{w} \in M \right\}$$

M is an \mathcal{A} -rational mediated set if $\mathcal{A} \subseteq M \subseteq A(M) \cup \mathcal{A}$

Theorem [Wang-Magron 20] for a circuit f

There exists a $\Lambda(f)$ -rational mediated set M containing β



$$M = \{\alpha(1), \alpha(2), \alpha(3), \beta, \beta(1), \beta(2)\}$$

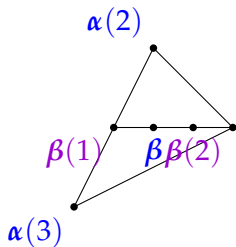
$$\mathbf{u} \in M \setminus \Lambda(f) \implies \mathbf{u} = (\mathbf{v} + \mathbf{w})/2$$

💡 There exist a_i, b_i, c_i such that

$$f(\mathbf{x}) = \sum_i a_i \mathbf{x}^{\mathbf{v}_i} + b_i \mathbf{x}^{\mathbf{w}_i} - 2c_i \mathbf{x}^{\mathbf{u}_i}$$

$$a_i b_i \geq c_i^2, \quad a_i, b_i \geq 0 \quad (\text{SOCP})$$

Circuits and sums of binomial squares



$$f = \sum_i (\sqrt{a_i} x^{\frac{v_i}{2}} - \sqrt{b_i} x^{\frac{w_i}{2}})^2$$

$$f = (1 - x_1 x_2^2)^2 + 2(x_1^{\frac{1}{2}} x_2 - x_1^{\frac{3}{2}} x_2)^2 + (x_1 x_2 - x_1^2 x_2)^2$$

\implies sum of 3 binomial squares

Circuits and sums of binomial squares

$f = \sum_i (\sqrt{a_i} x^{\frac{v_i}{2}} - \sqrt{b_i} x^{\frac{w_i}{2}})^2$
 $f = (1 - x_1 x_2^2)^2 + 2(x_1^{\frac{1}{2}} x_2 - x_1^{\frac{3}{2}} x_2)^2 + (x_1 x_2 - x_1^2 x_2)^2$
 \implies sum of 3 binomial squares

"Arbitrary support" $f = \sum_{\alpha \in \Lambda(f)} c_{\alpha} x^{\alpha} - \sum_{\beta \in \Gamma(f)} d_{\beta} x^{\beta}$
 Monomial squares = $\Lambda(f)$ Complement = $\Gamma(f)$

Circuits and sums of binomial squares

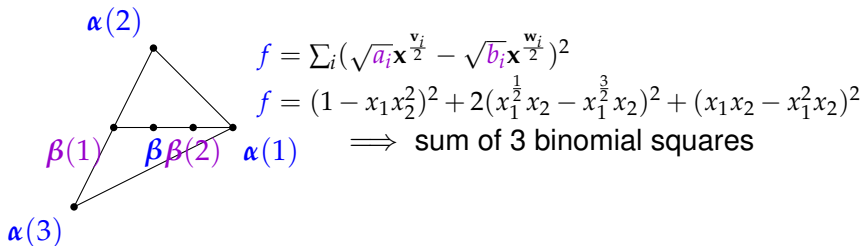
$f = \sum_i (\sqrt{a_i} x^{\frac{v_i}{2}} - \sqrt{b_i} x^{\frac{w_i}{2}})^2$
 $f = (1 - x_1 x_2^2)^2 + 2(x_1^{\frac{1}{2}} x_2 - x_1^{\frac{3}{2}} x_2)^2 + (x_1 x_2 - x_1^2 x_2)^2$
 \implies sum of 3 binomial squares

“Arbitrary support” $f = \sum_{\alpha \in \Lambda(f)} c_{\alpha} x^{\alpha} - \sum_{\beta \in \Gamma(f)} d_{\beta} x^{\beta}$

Monomial squares = $\Lambda(f)$ Complement = $\Gamma(f)$

Simplex **Cover** of each $\beta \implies \mathcal{A}$ and mediated set $M_{\mathcal{A}\beta}$

Circuits and sums of binomial squares

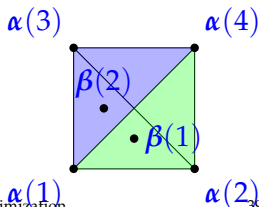


“Arbitrary support” $f = \sum_{\alpha \in \Lambda(f)} c_{\alpha} x^{\alpha} - \sum_{\beta \in \Gamma(f)} d_{\beta} x^{\beta}$

Monomial squares = $\Lambda(f)$ Complement = $\Gamma(f)$

Simplex **Cover** of each $\beta \implies \mathcal{A}$ and mediated set $M_{\mathcal{A}\beta}$

$$f = 50x_1^4 x_2^4 + x_1^4 + 3x_2^4 + 800 - 100x_1 x_2^2 - 100x_1^2 x_2$$



Algorithm opt sage: numerical steps

SAGE (SUMS OF ARITHMETIC-GEOMETRIC-MEAN-EXPONENTIALS)

- **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)]$, $\hat{\delta}$, $\tilde{\delta}$

Algorithm optsage: numerical steps

SAGE (SUMS OF ARITHMETIC-GEOMETRIC-MEAN-EXPONENTIALS)

■ **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)]$, $\hat{\delta}$, $\tilde{\delta}$

$f \in \mathcal{C}_{\text{SAGE}} \Leftrightarrow \exists \mathbf{v}^{(j)}, \mathbf{c}^{(j)}$ such that

$$\sum_j \mathbf{c}^{(j)} = \mathbf{b}, \quad \sum_i \boldsymbol{\alpha}(i) \mathbf{v}_i^{(j)} = \mathbf{0}, \quad -\mathbf{1} \cdot \mathbf{v}_{\setminus j}^{(j)} = \mathbf{v}_j^{(j)}$$

$$\mathbf{c}_{\setminus j}^{(j)}, \mathbf{v}_{\setminus j}^{(j)} \geq \mathbf{0}, \quad D\left(\mathbf{v}_{\setminus j}^{(j)}, e\mathbf{c}_{\setminus j}^{(j)}\right) \leq \mathbf{c}_j^{(j)}$$

Algorithm opt sage: numerical steps

SAGE (SUMS OF ARITHMETIC-GEOMETRIC-MEAN-EXPONENTIALS)

■ **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)]$, $\hat{\delta}$, $\tilde{\delta}$

$f \in \mathcal{C}_{\text{SAGE}} \Leftrightarrow \exists \mathbf{v}^{(j)}, \mathbf{c}^{(j)}$ such that

$$\sum_j \mathbf{c}^{(j)} = \mathbf{b}, \quad \sum_i \boldsymbol{\alpha}(i) \mathbf{v}_i^{(j)} = \mathbf{0}, \quad -\mathbf{1} \cdot \mathbf{v}_{\setminus j}^{(j)} = \mathbf{v}_j^{(j)}$$

$$\mathbf{c}_{\setminus j}^{(j)}, \mathbf{v}_{\setminus j}^{(j)} \geq \mathbf{0}, \quad D\left(\mathbf{v}_{\setminus j}^{(j)}, e^{\mathbf{c}_{\setminus j}^{(j)}}\right) \leq \mathbf{c}_j^{(j)}$$

1 Numerical resolution of RELATIVE ENTROPY PROGRAM

Precision $\tilde{\delta} \implies$ “IN PRACTICE” $\tilde{\mathbf{v}}, \tilde{\mathbf{c}}$ violate the constraints

Algorithm optsage: symbolic steps

- **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)], \hat{\delta}, \tilde{\delta}$

Algorithm opt sage: symbolic steps

■ **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)], \hat{\delta}, \tilde{\delta}$

Build the matrix Q with columns $(\boldsymbol{\alpha}(i), 1)$

Algorithm opt sage: symbolic steps

■ **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)], \hat{\delta}, \tilde{\delta}$

Build the matrix Q with columns $(\boldsymbol{\alpha}(i), 1)$

2 **Rounding step** $\hat{\mathbf{v}} \leftarrow \text{round}(\tilde{\mathbf{v}}, \hat{\delta}), \hat{\mathbf{c}} \leftarrow \text{round}(\tilde{\mathbf{c}}, \hat{\delta})$

Algorithm opt sage: symbolic steps

- **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)]$, $\hat{\delta}$, $\tilde{\delta}$

Build the matrix Q with columns $(\boldsymbol{\alpha}(i), 1)$

- **2 Rounding step** $\hat{\mathbf{v}} \leftarrow \text{round}(\tilde{\mathbf{v}}, \hat{\delta})$, $\hat{\mathbf{c}} \leftarrow \text{round}(\tilde{\mathbf{c}}, \hat{\delta})$

- **3 Projection step**

$$\mathbf{v}^{(j)} \leftarrow (\mathbf{I} - \mathbf{Q}^+ \mathbf{Q}) \hat{\mathbf{v}}^{(j)}, \quad \mathbf{c}_{\setminus j}^{(j)} \leftarrow \hat{\mathbf{c}}_{\setminus j}^{(j)}, \quad c_j^{(j)} \leftarrow b_j - \mathbf{1} \cdot \mathbf{c}_{\setminus j}^{(j)}$$

Algorithm opt sage: symbolic steps

- **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)]$, $\hat{\delta}$, $\tilde{\delta}$

Build the matrix \mathbf{Q} with columns $(\boldsymbol{\alpha}(i), 1)$

- **2 Rounding step** $\hat{\mathbf{v}} \leftarrow \text{round}(\tilde{\mathbf{v}}, \hat{\delta})$, $\hat{\mathbf{c}} \leftarrow \text{round}(\tilde{\mathbf{c}}, \hat{\delta})$

- **3 Projection step**

$$\mathbf{v}^{(j)} \leftarrow (\mathbf{I} - \mathbf{Q}^+ \mathbf{Q}) \hat{\mathbf{v}}^{(j)}, \quad \mathbf{c}_{\setminus j}^{(j)} \leftarrow \hat{\mathbf{c}}_{\setminus j}^{(j)}, \quad c_j^{(j)} \leftarrow b_j - \mathbf{1} \cdot \mathbf{c}_{\setminus j}^{(j)}$$

Compute $c_1^{(j)}$ such that

$$c_j^{(j)} \geq D\left(\mathbf{v}_{\setminus j}^{(j)}, e\mathbf{c}_{\setminus j}^{(j)}\right) = \sum_{j \neq i > 1} v_i^{(j)} \log \frac{v_i^{(j)}}{ec_i^{(j)}} + v_1^{(j)} \log \frac{v_1^{(j)}}{ec_1^{(j)}}$$

Algorithm opt sage: symbolic steps

- **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)]$, $\hat{\delta}$, $\tilde{\delta}$

Build the matrix \mathbf{Q} with columns $(\boldsymbol{\alpha}(i), 1)$

- **2 Rounding step** $\hat{\mathbf{v}} \leftarrow \text{round}(\tilde{\mathbf{v}}, \hat{\delta})$, $\hat{\mathbf{c}} \leftarrow \text{round}(\tilde{\mathbf{c}}, \hat{\delta})$

- **3 Projection step**

$$\mathbf{v}^{(j)} \leftarrow (\mathbf{I} - \mathbf{Q}^+ \mathbf{Q}) \hat{\mathbf{v}}^{(j)}, \quad \mathbf{c}_{\setminus j}^{(j)} \leftarrow \hat{\mathbf{c}}_{\setminus j}^{(j)}, \quad c_j^{(j)} \leftarrow b_j - \mathbf{1} \cdot \mathbf{c}_{\setminus j}^{(j)}$$

Compute $c_1^{(j)}$ such that

$$c_j^{(j)} \geq D\left(\mathbf{v}_{\setminus j}^{(j)}, \mathbf{e} \mathbf{c}_{\setminus j}^{(j)}\right) = \sum_{j \neq i > 1} v_i^{(j)} \log \frac{v_i^{(j)}}{e c_i^{(j)}} + v_1^{(j)} \log \frac{v_1^{(j)}}{e c_1^{(j)}}$$

$$c_1^{(j)} \leftarrow \text{round} \uparrow (\exp(\dots), \hat{\delta})$$

Algorithm opt sage: symbolic steps

- **Input** $f = \sum_i b_i \exp[\mathbf{x} \cdot \boldsymbol{\alpha}(i)]$, $\hat{\delta}$, $\tilde{\delta}$

Build the matrix Q with columns $(\boldsymbol{\alpha}(i), 1)$

- **2 Rounding step** $\hat{\mathbf{v}} \leftarrow \text{round}(\tilde{\mathbf{v}}, \hat{\delta})$, $\hat{\mathbf{c}} \leftarrow \text{round}(\tilde{\mathbf{c}}, \hat{\delta})$


- **3 Projection step**

$$\mathbf{v}^{(j)} \leftarrow (\mathbf{I} - \mathbf{Q}^+ \mathbf{Q}) \hat{\mathbf{v}}^{(j)}, \quad \mathbf{c}_{\setminus j}^{(j)} \leftarrow \hat{\mathbf{c}}_{\setminus j}^{(j)}, \quad c_j^{(j)} \leftarrow b_j - \mathbf{1} \cdot \mathbf{c}_{\setminus j}^{(j)}$$

Compute $c_1^{(j)}$ such that

$$c_j^{(j)} \geq D(\mathbf{v}_{\setminus j}^{(j)}, \mathbf{e} \mathbf{c}_{\setminus j}^{(j)}) = \sum_{i \neq j} v_i^{(j)} \log \frac{v_i^{(j)}}{e c_i^{(j)}} + v_1^{(j)} \log \frac{v_1^{(j)}}{e c_1^{(j)}}$$

$$c_1^{(j)} \leftarrow \text{round} \uparrow (\exp(\dots), \hat{\delta})$$

 $f \geq b_1 - \sum_j c_1^{(j)}$

SOS

- rounding-projection (SOS) [Peyrl-Parrilo]
- RAGLib (critical points) [Safey El Din]
- SamplePoints (CAD) [Moreno Maza-Alvandi et al.]

Id	n	d	RealCertify		RoundProject		RAGLib	CAD
			τ_1 (bits)	t_1 (s)	τ_2 (bits)	t_2 (s)	t_3 (s)	t_4 (s)
f_{20}	2	20	745 419	110.	78 949 497	141.	0.16	0.03
M	3	8	17 232	0.35	18 831	0.29	0.15	0.03
f_2	2	4	1 866	0.03	1 031	0.04	0.09	0.01
f_6	6	4	56 890	0.34	475 359	0.54	598.	—
f_1	10	4	344 347	2.45	8 374 082	4.59	—	—

SONC vs SAGE

terms	bit size		time	
	optsonc	optsage	optsonc	optsage
6	432	1005	0.06	0.26
9	806	2696	0.19	0.66
12	1261	5568	0.37	1.29
20	2592	19203	0.64	4.00
24	3826	32543	0.97	6.66
30	5029	53160	1.34	10.58
50	10622	167971	3.95	32.78

SONC vs SAGE

terms	bit size		time	
	optsonc	optsage	optsonc	optsage
6	432	1005	0.06	0.26
9	806	2696	0.19	0.66
12	1261	5568	0.37	1.29
20	2592	19203	0.64	4.00
24	3826	32543	0.97	6.66
30	5029	53160	1.34	10.58
50	10622	167971	3.95	32.78

💡 “IN PRACTICE” optsonc faster and more concise than optsage

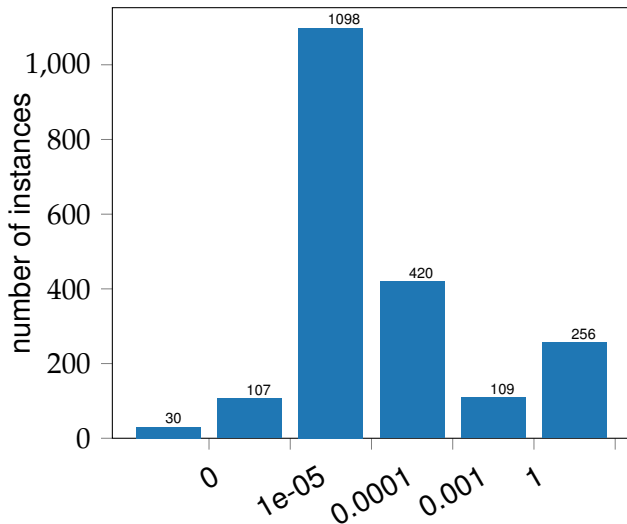
SONC vs SAGE

terms	bit size		time	
	optsonc	optsage	optsonc	optsage
6	432	1005	0.06	0.26
9	806	2696	0.19	0.66
12	1261	5568	0.37	1.29
20	2592	19203	0.64	4.00
24	3826	32543	0.97	6.66
30	5029	53160	1.34	10.58
50	10622	167971	3.95	32.78

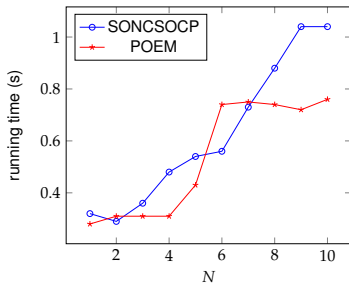
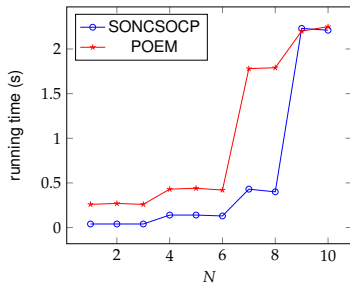
💡 “IN PRACTICE” optsonc faster and more concise than optsage

💡 “IN THEORY” optsonc less accurate than optsage

SONC: gap between numeric & symbolic



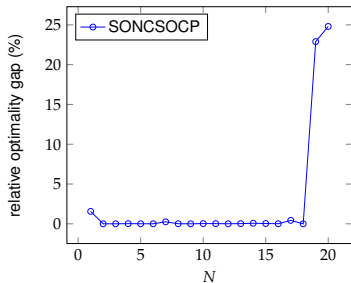
SONC: SOCP vs GP



Arbitrary support

$$n \sim 40$$

$$d \sim 60$$



Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
------	-------	--------------	------------

Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
univosos	Σ	\mathbb{R}	$\mathcal{O}(d^3 + d^2\tau)$

Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
univos	Σ	\mathbb{R}	$\mathcal{O}(d^3 + d^2\tau)$
csos	$\dot{\Sigma}$	\mathbb{S}^1	$\tilde{\mathcal{O}}(d^6(d + \tau))$

Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
univsos	Σ	\mathbb{R}	$\mathcal{O}(d^3 + d^2\tau)$
csos	$\dot{\Sigma}$	\mathbb{S}^1	$\tilde{\mathcal{O}}(d^6(d + \tau))$
multivsos	$\dot{\Sigma}$	\mathbb{R}^n	$\tau d^{d^{\mathcal{O}(n)}}$

Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
univsos	Σ	\mathbb{R}	$\mathcal{O}(d^3 + d^2\tau)$
csos	$\dot{\Sigma}$	\mathbb{S}^1	$\tilde{\mathcal{O}}(d^6(d + \tau))$
multivos	$\dot{\Sigma}$	\mathbb{R}^n	$\tau d^{d^{\mathcal{O}(n)}}$
gradsos	≥ 0	$V_{\text{grad}}(f)$	$\tilde{\mathcal{O}}((\tau + n + d)d^{3n+2})$

Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
univsos	Σ	\mathbb{R}	$\mathcal{O}(d^3 + d^2\tau)$
csos	$\mathring{\Sigma}$	\mathbb{S}^1	$\tilde{\mathcal{O}}(d^6(d + \tau))$
multivos	$\mathring{\Sigma}$	\mathbb{R}^n	$\tau d^{d^{\mathcal{O}(n)}}$
gradsos	≥ 0	$V_{\text{grad}}(f)$	$\tilde{\mathcal{O}}((\tau + n + d)d^{3n+2})$
intsage	$\mathring{\mathcal{C}}_{\text{SAGE}}$	\mathbb{R}^n	$\tilde{\mathcal{O}}(\tau_{\text{SAGE}}(f) + \tau + t)t^7$

Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
univsos	Σ	\mathbb{R}	$\mathcal{O}(d^3 + d^2\tau)$
csos	$\mathring{\Sigma}$	\mathbb{S}^1	$\tilde{\mathcal{O}}(d^6(d + \tau))$
multivos	$\mathring{\Sigma}$	\mathbb{R}^n	$\tau d^{d^{\mathcal{O}(n)}}$
gradsos	≥ 0	$V_{\text{grad}}(f)$	$\tilde{\mathcal{O}}((\tau + n + d)d^{3n+2})$
intsage	$\mathring{\mathcal{C}}_{\text{SAGE}}$	\mathbb{R}^n	$\tilde{\mathcal{O}}(\tau_{\text{SAGE}}(f) + \tau + t)t^7)$
intsonc	$\mathring{\mathcal{C}}_{\text{SONC}}$	\mathbb{R}^n	$\tilde{\mathcal{O}}(\tau_{\text{SONC}}(g)d^{3.5n}n^{10.5}))$

Conclusion and Perspectives

Input f on \mathbf{K} with $\deg f = d$, t terms and bit size τ

Algo	Input	\mathbf{K}	COMPLEXITY
univsos	Σ	\mathbb{R}	$\mathcal{O}(d^3 + d^2\tau)$
csos	$\mathring{\Sigma}$	\mathbb{S}^1	$\tilde{\mathcal{O}}(d^6(d + \tau))$
multivsos	$\mathring{\Sigma}$	\mathbb{R}^n	$\tau d^{d^{\mathcal{O}(n)}}$
gradsos	≥ 0	$V_{\text{grad}}(f)$	$\tilde{\mathcal{O}}((\tau + n + d)d^{3n+2})$
intsage	$\mathring{C}_{\text{SAGE}}$	\mathbb{R}^n	$\tilde{\mathcal{O}}(\tau_{\text{SAGE}}(f) + \tau + t)t^7$
intsonc	$\mathring{C}_{\text{SONC}}$	\mathbb{R}^n	$\tilde{\mathcal{O}}(\tau_{\text{SONC}}(g)d^{3.5n}n^{10.5})$

- 💡 How to bound the distance to the SONC/SAGE cone?
- 💡 Arbitrary precision SDP/GP/REP/SOCP solvers
- 💡 How to handle degenerate situations?

Thank you for your (certified) attention!

RealCertify

POEM

SONCSOCP

<https://homepages.laas.fr/vmagron>



M., Allamigeon, Gaubert & Werner. Formal proofs for Nonlinear Optimization, *Journal of Formalized Reasoning*. arXiv:1404.7282



M. & Safey El Din. On Exact Polya and Putinar's Representations, *ISSAC'18*. arXiv:1802.10339 [RealCertify](#)



M. & Safey El Din. RealCertify: a Maple package for certifying non-negativity, *ISSAC'18*. arXiv:1805.02201 [RealCertify](#)



M., Safey El Din & Schweighofer. Algorithms for weighted sum of squares decomposition of non-negative univariate polynomials, *Journal of Symbolic Computation*. arXiv:1706.03941



M., Seidler & de Wolff. Exact optimization via sums of nonnegative circuits and AM/GM exponentials, *ISSAC'19*. arXiv:1902.02123

[POEM](#)



M. & Wang. SONC Optimization & Exact Nonnegativity Certificates via Second-order Conic Programming, *ISSAC'20*. arXiv:2012.07903

[SONCSOCP](#)



M., Safey El Din & Vu. Sum of Squares Decompositions of Polynomials over their Gradient Ideals with Rational Coefficients. arXiv:2107.11825



M., Safey El Din, Schweighofer, & Vu. Exact SOHS decompositions of trigonometric univariate polynomials with Gaussian coefficients. arXiv:2202.06544