Certification of Inequalities involving Transcendental Functions: combining SDP and Max-plus Approximation

Joint Work withX. Allamigeon, S. Gaubert and B. Werner

Third year PhD Victor MAGRON

LIX/CMAP INRIA, École Polytechnique

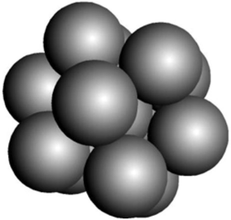
ECC 2013 Thursday July 18 th

The Kepler Conjecture

Kepler Conjecture (1611):

The maximal density of sphere packings in 3D-space is $\frac{\pi}{\sqrt{18}}$

- It corresponds to the way people would intuitively stack oranges, as a pyramid shape
- The proof of T. Hales (1998) consists of thousands of non-linear inequalities
- Many recent efforts have been done to give a formal proof of these inequalities: Flyspeck Project
- Motivation: get positivity certificates and check them with Proof assistants like Coq



Contents

The Kepler Conjecture

Inequalities issued from Flyspeck non-linear part involve:

1. Multivariate Polynomials:
$$x_1x_4(-x_1+x_2+x_3-x_4+x_5+x_6)+x_2x_5(x_1-x_2+x_3+x_4-x_5+x_6)+ x_3x_6(x_1+x_2-x_3+x_4+x_5-x_6)-x_2(x_3x_4+x_1x_6)-x_5(x_1x_3+x_4x_6)$$
2. Semi-Algebraic functions algebra $\mathcal{A}$: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \cdots$
3. Transcendental functions $\mathcal{T}$: composition of semi-algebraic functions with $\arctan, \exp, \sin, +, -, \times, \cdots$

Lemma from Flyspeck (inequality ID 6096597438)

$$\forall x \in [3, 64], 2\pi - 2x \arcsin(\cos(0.797)\sin(\pi/x)) + 0.0331x - 2.097 \geq 0$$

The Kepler Conjecture

Inequalities issued from Flyspeck non-linear part involve:

1. Multivariate Polynomials:
   $x_1x_4(-x_1+x_2+x_3-x_4+x_5+x_6)+x_2x_5(x_1-x_2+x_3+x_4-x_5+x_6)+$
   $x_3x_6(x_1+x_2-x_3+x_4+x_5-x_6)-x_2(x_3x_4+x_1x_6)-x_5(x_1x_3+x_4x_6)$
2. Semi-Algebraic functions algebra $\mathcal{A}$: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \cdots$
3. Transcendental functions $\mathcal{T}$: composition of semi-algebraic functions with $\arctan, \exp, \sin, +, -, \times, \cdots$

Lemma from Flyspeck (inequality ID 6096597438)

$\forall x \in [3, 64], 2\pi - 2x \arcsin(\cos(0.797)\sin(\pi/x)) + 0.0331x - 2.097 \geq 0$

The Kepler Conjecture

Inequalities issued from Flyspeck non-linear part involve:

1. Multivariate Polynomials:
$x_1x_4(-x_1+x_2+x_3-x_4+x_5+x_6)+x_2x_5(x_1-x_2+x_3+x_4-x_5+x_6)+x_3x_6(x_1+x_2-x_3+x_4+x_5-x_6)-x_2(x_3x_4+x_1x_6)-x_5(x_1x_3+x_4x_6)$
2. Semi-Algebraic functions algebra $\mathcal{A}$: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \cdots$
3. Transcendental functions $\mathcal{T}$: composition of semi-algebraic functions with $\arctan, \exp, \sin, +, -, \times, \cdots$

Lemma from Flyspeck (inequality ID 6096597438)

$\forall x \in [3, 64], 2\pi - 2x \arcsin(\cos(0.797)\sin(\pi/x)) + 0.0331x - 2.097 \geq 0$

The Kepler Conjecture

Inequalities issued from Flyspeck non-linear part involve:

1. Multivariate Polynomials:
   $x_1x_4(-x_1+x_2+x_3-x_4+x_5+x_6)+x_2x_5(x_1-x_2+x_3+x_4-x_5+x_6)+ x_3x_6(x_1+x_2-x_3+x_4+x_5-x_6)-x_2(x_3x_4+x_1x_6)-x_5(x_1x_3+x_4x_6)$
2. Semi-Algebraic functions algebra $\mathcal{A}$: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \cdots$
3. Transcendental functions $\mathcal{T}$: composition of semi-algebraic functions with $\arctan, \exp, \sin, +, -, \times, \cdots$

Lemma from Flyspeck (inequality ID 6096597438)

$\forall x \in [3, 64], 2\pi - 2x \arcsin(\cos(0.797) \sin(\pi/x)) + 0.0331x - 2.097 \geq 0$

Global Optimization Problems: Examples from the Literature

- *H3*: $\min_{\mathbf{x}\in[0,1]^3} -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right]$
- *MC*: $\min_{\substack{x_1\in[-3,3]\\ x_2\in[-1.5,4]}} \sin(x_1 + x_2) + (x_1 - x_2)^2 - 0.5x_2 + 2.5x_1 + 1$
- *SBT*: $\min_{\mathbf{x}\in[-10,10]^n} \prod_{i=1}^{n} \left(\sum_{j=1}^{5} j\cos((j+1)x_i + j)\right)$
- *SWF*: $\min_{\mathbf{x}\in[1,500]^n} -\sum_{i=1}^{n} (x_i + \epsilon x_{i+1}) \sin(\sqrt{x_i}) \quad (\epsilon \in \{0,1\})$

Global Optimization Problems: a Framework

Given K a compact set, and f a transcendental function, minor $f^* = \inf_{\mathbf{x} \in K} f(\mathbf{x})$ and prove $f^* \geq 0$

1. f is underestimated by a semialgebraic function f_{sa}
2. We reduce the problem $f^*_{\text{sa}} := \inf_{\mathbf{x} \in K} f_{\text{sa}}(\mathbf{x})$ to a polynomial optimization problem in a lifted space K_{pop} (with lifting variables $\mathbf{z}$)
3. We solve the POP problem $f^*_{\text{pop}} := \inf_{(\mathbf{x},\mathbf{z}) \in K_{\text{pop}}} f_{\text{pop}}(\mathbf{x}, \mathbf{z})$ using a hierarchy of SDP relaxations by Lasserre

If the relaxations are accurate enough, $f^* \geq f^*_{\text{sa}} \geq f^*_{\text{pop}} \geq 0$.

Global Optimization Problems: a Framework

Given K a compact set, and f a transcendental function, minor $f^* = \inf_{\mathbf{x} \in K} f(\mathbf{x})$ and prove $f^* \geq 0$

1. f is underestimated by a semialgebraic function f_{sa}
2. We reduce the problem $f_{\text{sa}}^* := \inf_{\mathbf{x} \in K} f_{\text{sa}}(\mathbf{x})$ to a polynomial optimization problem in a lifted space K_{pop} (with lifting variables $\mathbf{z}$)
3. We solve the POP problem $f_{\text{pop}}^* := \inf_{(\mathbf{x},\mathbf{z}) \in K_{\text{pop}}} f_{\text{pop}}(\mathbf{x}, \mathbf{z})$ using a hierarchy of SDP relaxations by Lasserre

If the relaxations are accurate enough, $f^* \geq f_{\text{sa}}^* \geq f_{\text{pop}}^* \geq 0$.

Global Optimization Problems: a Framework

Given K a compact set, and f a transcendental function, minor $f^* = \inf_{\mathbf{x} \in K} f(\mathbf{x})$ and prove $f^* \geq 0$

1. f is underestimated by a semialgebraic function f_{sa}
2. We reduce the problem $f_{\text{sa}}^* := \inf_{\mathbf{x} \in K} f_{\text{sa}}(\mathbf{x})$ to a polynomial optimization problem in a lifted space K_{pop} (with lifting variables $\mathbf{z}$)
3. We solve the POP problem $f_{\text{pop}}^* := \inf_{(\mathbf{x},\mathbf{z}) \in K_{\text{pop}}} f_{\text{pop}}(\mathbf{x}, \mathbf{z})$ using a hierarchy of SDP relaxations by Lasserre

If the relaxations are accurate enough, $f^* \geq f_{\text{sa}}^* \geq f_{\text{pop}}^* \geq 0$.

Global Optimization Problems: a Framework

Given K a compact set, and f a transcendental function, minor $f^* = \inf_{\mathbf{x} \in K} f(\mathbf{x})$ and prove $f^* \geq 0$

1. f is underestimated by a semialgebraic function f_{sa}
2. We reduce the problem $f_{\text{sa}}^* := \inf_{\mathbf{x} \in K} f_{\text{sa}}(\mathbf{x})$ to a polynomial optimization problem in a lifted space K_{pop} (with lifting variables $\mathbf{z}$)
3. We solve the POP problem $f_{\text{pop}}^* := \inf_{(\mathbf{x},\mathbf{z}) \in K_{\text{pop}}} f_{\text{pop}}(\mathbf{x}, \mathbf{z})$ using a hierarchy of SDP relaxations by Lasserre

If the relaxations are accurate enough, $f^* \geq f_{\text{sa}}^* \geq f_{\text{pop}}^* \geq 0$.

Global Optimization Problems: a Framework

Given K a compact set, and f a transcendental function, minor $f^* = \inf_{\mathbf{x} \in K} f(\mathbf{x})$ and prove $f^* \geq 0$

1. f is underestimated by a semialgebraic function f_{sa}
2. We reduce the problem $f_{\text{sa}}^* := \inf_{\mathbf{x} \in K} f_{\text{sa}}(\mathbf{x})$ to a polynomial optimization problem in a lifted space K_{pop} (with lifting variables $\mathbf{z}$)
3. We solve the POP problem $f_{\text{pop}}^* := \inf_{(\mathbf{x},\mathbf{z}) \in K_{\text{pop}}} f_{\text{pop}}(\mathbf{x}, \mathbf{z})$ using a hierarchy of SDP relaxations by Lasserre

If the relaxations are accurate enough, $f^* \geq f_{\text{sa}}^* \geq f_{\text{pop}}^* \geq 0$.

Contents

Semialgebraic Optimization Problems

- Polynomial Optimization Problem (POP):
  $p^* := \min_{\mathbf{x} \in K} p(\mathbf{x})$ with K the compact set of constraints:

$$K = \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geq 0, \cdots, g_m(\mathbf{x}) \geq 0\}$$

- Let $\Sigma_d[\mathbf{x}]$ be the cone of Sum-of-Squares (SOS) of degree at most $2d$:

$$\Sigma_d[\mathbf{x}] = \left\{ \sum_i q_i(\mathbf{x})^2, \text{ with } q_i \in \mathbb{R}_d[\mathbf{x}] \right\}$$

- Let $g_0 := 1$ and $M_d(\mathbf{g})$ be the quadratic module:

$$M_d(\mathbf{g}) = \left\{ \sum_{j=0}^{m} \sigma_j(\mathbf{x}) g_j(\mathbf{x}), \text{ with } \sigma_j \in \Sigma[\mathbf{x}], (\sigma_j g_j) \in \mathbb{R}_{2d}[\mathbf{x}] \right\}$$

Semialgebraic Optimization Problems

- Polynomial Optimization Problem (POP):
  $p^* := \min_{\mathbf{x} \in K} p(\mathbf{x})$ with K the compact set of constraints:

$$K = \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geq 0, \cdots, g_m(\mathbf{x}) \geq 0\}$$

- Let $\Sigma_d[\mathbf{x}]$ be the cone of Sum-of-Squares (SOS) of degree at most $2d$:

$$\Sigma_d[\mathbf{x}] = \left\{ \sum_i q_i(\mathbf{x})^2, \text{ with } q_i \in \mathbb{R}_d[\mathbf{x}] \right\}$$

- Let $g_0 := 1$ and $M_d(\mathbf{g})$ be the quadratic module:

$$M_d(\mathbf{g}) = \left\{ \sum_{j=0}^{m} \sigma_j(\mathbf{x}) g_j(\mathbf{x}), \text{ with } \sigma_j \in \Sigma[\mathbf{x}], (\sigma_j g_j) \in \mathbb{R}_{2d}[\mathbf{x}] \right\}$$

Semialgebraic Optimization Problems

- Polynomial Optimization Problem (POP):
  $p^* := \min_{\mathbf{x} \in K} p(\mathbf{x})$ with K the compact set of constraints:

$$K = \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geq 0, \cdots, g_m(\mathbf{x}) \geq 0\}$$

- Let $\Sigma_d[\mathbf{x}]$ be the cone of Sum-of-Squares (SOS) of degree at most $2d$:

$$\Sigma_d[\mathbf{x}] = \Big\{ \sum_i q_i(\mathbf{x})^2, \text{ with } q_i \in \mathbb{R}_d[\mathbf{x}] \Big\}$$

- Let $g_0 := 1$ and $M_d(\mathbf{g})$ be the quadratic module:

$$M_d(\mathbf{g}) = \Big\{ \sum_{j=0}^{m} \sigma_j(\mathbf{x}) g_j(\mathbf{x}), \text{ with } \sigma_j \in \Sigma[\mathbf{x}], (\sigma_j g_j) \in \mathbb{R}_{2d}[\mathbf{x}] \Big\}$$

Semialgebraic Optimization Problems

$$M(\mathbf{g}) := \bigcup_{d \in \mathbb{N}} M_d(\mathbf{g})$$

Proposition (Putinar)

Suppose $\mathbf{x} \in [\mathbf{a}, \mathbf{b}]$. $p(\mathbf{x}) - p^* > 0$ on $K \implies (p(\mathbf{x}) - p^*) \in M(\mathbf{g})$

- $M_0(\mathbf{g}) \subset M_1(\mathbf{g}) \subset M_2(\mathbf{g}) \subset \cdots \subset M(\mathbf{g})$
- Hence, we consider the hierarchy of SOS *relaxation* programs: $\mu_k := \sup_{\mu, \sigma_0, \cdots, \sigma_m} \left\{ \mu : (p(\mathbf{x}) - \mu) \in M_k(\mathbf{g}) \right\}$
- $\mu_k \uparrow p^*$ (Lasserre Hierarchy Convergence)

Semialgebraic Optimization Problems

Example from Flyspeck:

Also works for Semialgebraic functions via *lifting* variables:

$$\Delta(\mathbf{x}) = x_1 x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2 x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3 x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3 x_4 + x_1 x_6) - x_5(x_1 x_3 + x_4 x_6)$$

$$\partial_4 \Delta \mathbf{x} = x_1(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2 x_5 + x_3 x_6 - x_2 x_3 - x_5 x_6$$

$$f^*_{\text{sa}} := \min_{\mathbf{x} \in [4, 6.3504]^6} \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}}$$

Semialgebraic Optimization Problems

Example from Flyspeck:

$$z_1 := \sqrt{4 x_1 \Delta \mathbf{x}}, \; m_1 = \inf_{\mathbf{x} \in [4, 6.3504]^6} z_1(\mathbf{x}), \; M_1 = \sup_{\mathbf{x} \in [4, 6.3504]^6} z_1(\mathbf{x}).$$

$$K := \{(\mathbf{x}, \mathbf{z}) \in \mathbb{R}^8 : \mathbf{x} \in [4, 6.3504]^6, h_1(\mathbf{x}, \mathbf{z}) \geq 0, \cdots, h_7(\mathbf{x}, \mathbf{z}) \geq 0\}$$

$$h_1 := z_1 - m_1 \qquad h_4 := -z_1^2 + 4 x_1 \Delta \mathbf{x}$$

$$h_2 := M_1 - z_1 \qquad h_5 := z_2 z_1 - \partial_4 \Delta \mathbf{x}$$

$$h_3 := z_1^2 - 4 x_1 \Delta \mathbf{x} \qquad h_6 := -z_2 z_1 + \partial_4 \Delta \mathbf{x}$$

$p^* := \inf_{(\mathbf{x}, \mathbf{z}) \in K} z_2 = f^*_{\text{sa}}$. We obtain $\mu_2 = -0.618$ and $\mu_3 = -0.445$.

Taylor Approximation of Transcendental Functions

SWF: $\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n} (x_i + x_{i+1}) \sin(\sqrt{x_i})$

Classical idea: approximate $\sin(\sqrt{\cdot})$ by a degree-$2k$ Taylor

Polynomial f_{2k}, solve $\min_{\mathbf{x} \in [1,500]^n} -\sum_{i=1}^{n} (x_i + x_{i+1}) f_{2k}(x_i)$ (POP)

Issues:

- If $2k$ is too small $\Rightarrow$ expensive Branch and Bound
- No free lunch: solving POP with $O(n^{2k})$ variables

SWF with $n = 10, d = 4$: takes already $38\,min$ to certify a lower bound of $-430n$

Taylor Approximation of Transcendental Functions

SWF: $\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n} (x_i + x_{i+1}) \sin(\sqrt{x_i})$

Classical idea: approximate $\sin(\sqrt{\cdot})$ by a degree-$2k$ Taylor

Polynomial f_{2k}, solve $\min_{\mathbf{x} \in [1,500]^n} -\sum_{i=1}^{n} (x_i + x_{i+1}) f_{2k}(x_i)$ (POP)

Issues:

- If $2k$ is too small $\Rightarrow$ expensive Branch and Bound
- No free lunch: solving POP with $O(n^{2k})$ variables

SWF with $n = 10, d = 4$: takes already $38\,min$ to certify a lower bound of $-430n$

Taylor Approximation of Transcendental Functions

SWF: $\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n} (x_i + x_{i+1}) \sin(\sqrt{x_i})$

Classical idea: approximate $\sin(\sqrt{\cdot})$ by a degree-$2k$ Taylor

Polynomial f_{2k}, solve $\min_{\mathbf{x} \in [1,500]^n} - \sum_{i=1}^{n} (x_i + x_{i+1}) f_{2k}(x_i)$ (POP)

Issues:

- If $2k$ is too small $\Rightarrow$ expensive Branch and Bound
- No free lunch: solving POP with $O(n^{2k})$ variables

SWF with $n = 10, d = 4$: takes already $38\,min$ to certify a lower bound of $-430n$

Taylor Approximation of Transcendental Functions

SWF: $\min_{\mathbf{x}\in[1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n}(x_i + x_{i+1})\sin(\sqrt{x_i})$

Classical idea: approximate $\sin(\sqrt{\cdot})$ by a degree-$2k$ Taylor

Polynomial f_{2k}, solve $\min_{\mathbf{x}\in[1,500]^n} -\sum_{i=1}^{n}(x_i + x_{i+1}) f_{2k}(x_i)$ (POP)

Issues:

- If $2k$ is too small $\Rightarrow$ expensive Branch and Bound
- No free lunch: solving POP with $O(n^{2k})$ variables

SWF with $n = 10, d = 4$: takes already $38\,min$ to certify a lower bound of $-430n$

Taylor Approximation of Transcendental Functions

SWF: $\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = - \sum_{i=1}^{n} (x_i + x_{i+1}) \sin(\sqrt{x_i})$

Classical idea: approximate $\sin(\sqrt{\cdot})$ by a degree-$2k$ Taylor

Polynomial f_{2k}, solve $\min_{\mathbf{x} \in [1,500]^n} - \sum_{i=1}^{n} (x_i + x_{i+1}) f_{2k}(x_i)$ (POP)

Issues:

- If $2k$ is too small $\Rightarrow$ expensive Branch and Bound
- No free lunch: solving POP with $O(n^{2k})$ variables

SWF with $n = 10, d = 4$: takes already $38\, min$ to certify a lower bound of $-430n$

Taylor Approximation of Transcendental Functions

SWF: $\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n} (x_i + x_{i+1}) \sin(\sqrt{x_i})$

Classical idea: approximate $\sin(\sqrt{\cdot})$ by a degree-$2k$ Taylor Polynomial f_{2k}, solve $\min_{\mathbf{x} \in [1,500]^n} -\sum_{i=1}^{n} (x_i + x_{i+1}) f_{2k}(x_i)$ (POP)

Issues:

- If $2k$ is too small $\Rightarrow$ expensive Branch and Bound
- No free lunch: solving POP with $O(n^{2k})$ variables

SWF with $n = 10, d = 4$: takes already $38\, min$ to certify a lower bound of $-430n$

Contents

Max-Plus Approximations

Goals:

- Reduce the $O(n^{2k})$ exponential dependency: use low degree approximations
- Reduce the Branch and Bound iterations: refine the approximations with an adaptive iterative algorithm

Max-Plus Approximations

- Let $\hat{f} \in \mathcal{T}$ be a transcendental univariate function ($\arctan$, $\exp$) defined on an interval I.
- $\hat{f}$ is semi-convex: there exists a constant $c_j > 0$ s.t. $a \mapsto \hat{f}(a) + c_j/2(a - a_j)^2$ is convex
- By convexity: $\forall a \in I, \hat{f}(a) \geq -c_j/2(a-a_j)^2 + \hat{f}'(a_j)(a-a_j) + \hat{f}(a_j) = \mathrm{par}^-_{a_j}(a)$
- $\forall j, \hat{f} \geq \mathrm{par}^-_{a_j} \implies \hat{f} \geq \max_j \{\mathrm{par}^-_{a_j}\}$ **Max-Plus underestimator**

Example with $\arctan$:

- $\hat{f}'(a_j) = \dfrac{1}{1 + a_j^2}, \quad c_j = \sup_{a \in I} \{-\hat{f}''(a)\}$ (always work)
- c_j depends on a_j and the curvature variations of $\arctan$ on the considered interval I

Max-Plus Approximations

- Let $\hat{f} \in \mathcal{T}$ be a transcendental univariate function ($\arctan$, $\exp$) defined on an interval I.
- $\hat{f}$ is semi-convex: there exists a constant $c_j > 0$ s.t. $a \mapsto \hat{f}(a) + c_j/2(a - a_j)^2$ is convex
- By convexity: $\forall a \in I, \hat{f}(a) \geq -c_j/2(a - a_j)^2 + \hat{f}'(a_j)(a - a_j) + \hat{f}(a_j) = \mathrm{par}^-_{a_j}(a)$
- $\forall j, \hat{f} \geq \mathrm{par}^-_{a_j} \implies \hat{f} \geq \max_j \{\mathrm{par}^-_{a_j}\}$ **Max-Plus underestimator**

Example with $\arctan$:

- $\hat{f}'(a_j) = \dfrac{1}{1 + a_j^2}, \quad c_j = \sup_{a \in I}\{-\hat{f}''(a)\}$ (always work)
- c_j depends on a_j and the curvature variations of $\arctan$ on the considered interval I

Max-Plus Approximations

- Let $\hat{f} \in \mathcal{T}$ be a transcendental univariate function ($\arctan$, $\exp$) defined on an interval I.
- $\hat{f}$ is semi-convex: there exists a constant $c_j > 0$ s.t. $a \mapsto \hat{f}(a) + c_j/2(a - a_j)^2$ is convex
- By convexity: $\forall a \in I, \hat{f}(a) \geq -c_j/2(a - a_j)^2 + \hat{f}'(a_j)(a - a_j) + \hat{f}(a_j) = \text{par}^-_{a_j}(a)$
- $\forall j, \hat{f} \geq \text{par}^-_{a_j} \implies \hat{f} \geq \max_j \{\text{par}^-_{a_j}\}$ **Max-Plus underestimator**

Example with $\arctan$:

- $\hat{f}'(a_j) = \dfrac{1}{1 + a_j^2}, \quad c_j = \sup_{a \in I} \{-\hat{f}''(a)\}$ (always work)
- c_j depends on a_j and the curvature variations of $\arctan$ on the considered interval I

Max-Plus Approximations

- Let $\hat{f} \in \mathcal{T}$ be a transcendental univariate function ($\arctan$, $\exp$) defined on an interval I.
- $\hat{f}$ is semi-convex: there exists a constant $c_j > 0$ s.t. $a \mapsto \hat{f}(a) + c_j/2(a - a_j)^2$ is convex
- By convexity:
  $\forall a \in I, \hat{f}(a) \geq -c_j/2(a - a_j)^2 + \hat{f}'(a_j)(a - a_j) + \hat{f}(a_j) = \text{par}^-_{a_j}(a)$
- $\forall j, \hat{f} \geq \text{par}^-_{a_j} \implies \hat{f} \geq \max_j\{\text{par}^-_{a_j}\}$ **Max-Plus underestimator**

Example with $\arctan$:

- $\hat{f}'(a_j) = \dfrac{1}{1 + a_j^2}, \quad c_j = \sup_{a \in I}\{-\hat{f}''(a)\}$ (always work)
- c_j depends on a_j and the curvature variations of $\arctan$ on the considered interval I

Max-Plus Approximations

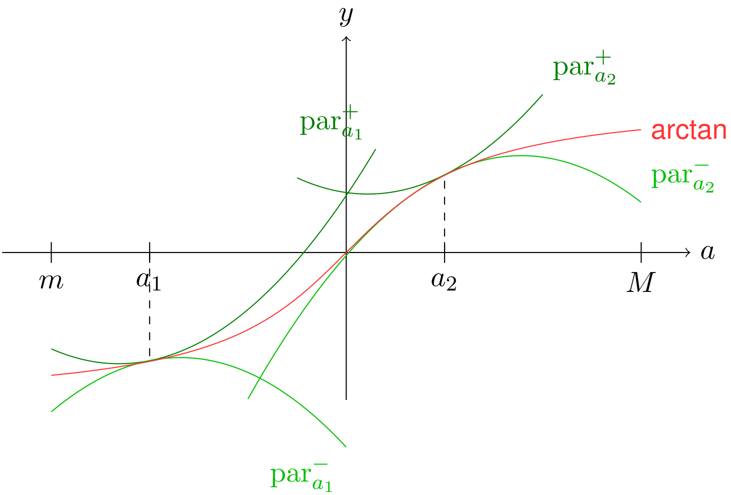
- Let $\hat{f} \in \mathcal{T}$ be a transcendental univariate function ($\arctan$, $\exp$) defined on an interval I.
- $\hat{f}$ is semi-convex: there exists a constant $c_j > 0$ s.t. $a \mapsto \hat{f}(a) + c_j/2(a - a_j)^2$ is convex
- By convexity: $\forall a \in I, \hat{f}(a) \geq -c_j/2(a - a_j)^2 + \hat{f}'(a_j)(a - a_j) + \hat{f}(a_j) = \mathrm{par}^-_{a_j}(a)$
- $\forall j, \hat{f} \geq \mathrm{par}^-_{a_j} \implies \hat{f} \geq \max_j \{\mathrm{par}^-_{a_j}\}$ **Max-Plus underestimator**

Example with $\arctan$:

- $\hat{f}'(a_j) = \dfrac{1}{1 + a_j^2}, \quad c_j = \sup_{a \in I} \{-\hat{f}''(a)\}$ (always work)
- c_j depends on a_j and the curvature variations of $\arctan$ on the considered interval I

Max-Plus Approximations

Example with $\arctan$:



Max-Plus Approximations

- $l := -\frac{\pi}{2} + 1.6294 - 0.2213\,(\sqrt{x_2} + \sqrt{x_3} + \sqrt{x_5} + \sqrt{x_6} - 8.0) + 0.913\,(\sqrt{x_4} - 2.52) + 0.728\,(\sqrt{x_1} - 2.0)$

Lemma$_{9922699028}$ from Flyspeck:

$$\forall \mathbf{x} \in [4, 6.3504]^6, \arctan\left(\frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}}\right) + l(\mathbf{x}) \geq 0$$

- Using semialgebraic optimization methods:

$$\forall x \in [4, 6.3504]^6,\ m \leq \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}} \leq M$$

Max-Plus Approximations

- $l := -\frac{\pi}{2} + 1.6294 - 0.2213\,(\sqrt{x_2} + \sqrt{x_3} + \sqrt{x_5} + \sqrt{x_6} - 8.0) + 0.913\,(\sqrt{x_4} - 2.52) + 0.728\,(\sqrt{x_1} - 2.0)$

Lemma$_{9922699028}$ from Flyspeck:

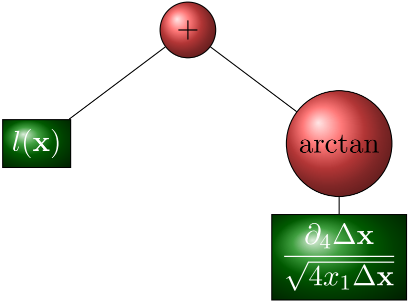
$$\forall \mathbf{x} \in [4, 6.3504]^6, \arctan\left(\frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4x_1 \Delta \mathbf{x}}}\right) + l(\mathbf{x}) \geq 0$$

- Using semialgebraic optimization methods:
  $\forall x \in [4, 6.3504]^6,\ m \leq \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4x_1 \Delta \mathbf{x}}} \leq M$

Semialgebraic Max-Plus Algorithm

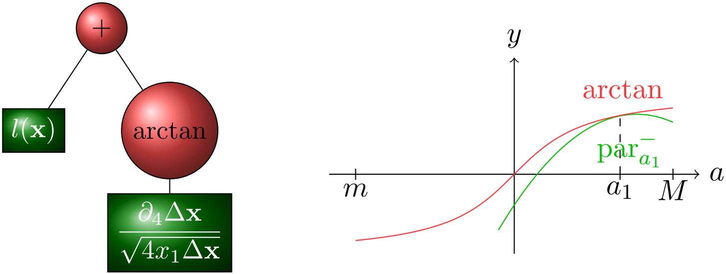
Abstract syntax tree representations of multivariate transcendental function:

- leaves are semialgebraic functions of $\mathcal{A}$
- nodes are univariate transcendental functions of $\mathcal{T}$ or binary operations



Semialgebraic Max-Plus Algorithm

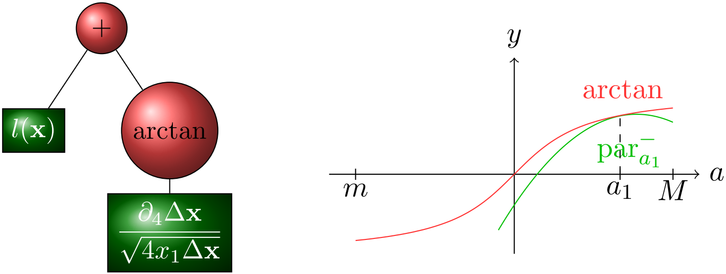
`samp_optim` First iteration:



1. Evaluate f with `randeval` and obtain a minimizer guess $\mathbf{x}^1_{opt}$.
   Compute $a_1 := \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4x_1 \Delta \mathbf{x}}}(\mathbf{x}^1_{opt}) = f_{\mathsf{sa}}(\mathbf{x}^1_{opt}) = 0.84460$
2. Get the equation of $\mathrm{par}^-_{a_1}$ with $\mathtt{build}_{\mathrm{par}}$
3. Compute $m_1 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \mathrm{par}^-_{a_1}(f_{\mathsf{sa}}(\mathbf{x})))$

Semialgebraic Max-Plus Algorithm

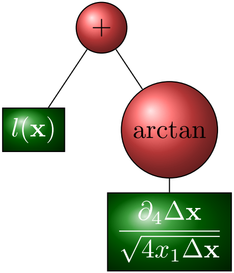
`samp_optim` First iteration:

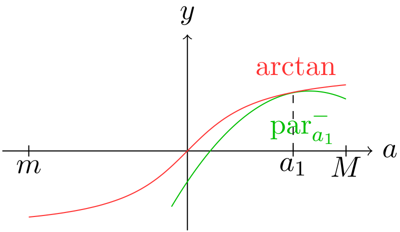


1. Evaluate f with `randeval` and obtain a minimizer guess $\mathbf{x}^1_{opt}$.
   Compute $a_1 := \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}} (\mathbf{x}^1_{opt}) = f_{\text{sa}}(\mathbf{x}^1_{opt}) = 0.84460$
2. Get the equation of $\mathrm{par}^-_{a_1}$ with $\mathtt{build}_{\mathrm{par}}$
3. Compute $m_1 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \mathrm{par}^-_{a_1}(f_{\text{sa}}(\mathbf{x})))$

Semialgebraic Max-Plus Algorithm

`samp_optim` First iteration:





1. Evaluate f with `randeval` and obtain a minimizer guess $\mathbf{x}^1_{opt}$.

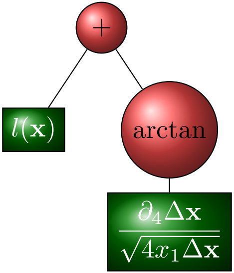
   Compute $a_1 := \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}} (\mathbf{x}^1_{opt}) = f_{\text{sa}}(\mathbf{x}^1_{opt}) = 0.84460$

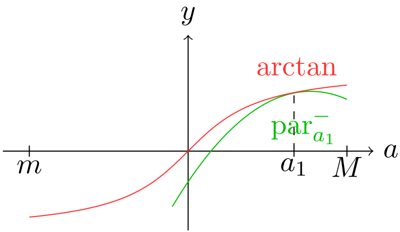
2. Get the equation of $\text{par}^{-}_{a_1}$ with $\texttt{build}_{\texttt{par}}$

3. Compute $m_1 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \text{par}^{-}_{a_1}(f_{\text{sa}}(\mathbf{x})))$

Semialgebraic Max-Plus Algorithm

`samp_optim` First iteration:

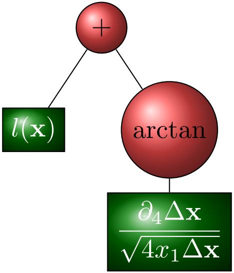


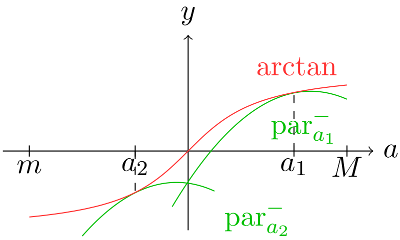


1. Evaluate f with `randeval` and obtain a minimizer guess $\mathbf{x}^1_{opt}$.
   Compute $a_1 := \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4x_1 \Delta \mathbf{x}}} (\mathbf{x}^1_{opt}) = f_{\text{sa}}(\mathbf{x}^1_{opt}) = 0.84460$
2. Get the equation of $\mathrm{par}^-_{a_1}$ with $\texttt{build}_{\texttt{par}}$
3. Compute $m_1 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \mathrm{par}^-_{a_1}(f_{\text{sa}}(\mathbf{x})))$

Semialgebraic Max-Plus Algorithm

samp_optim Second iteration:

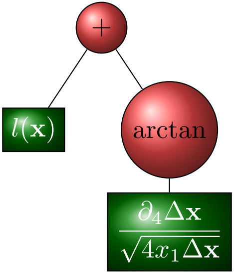


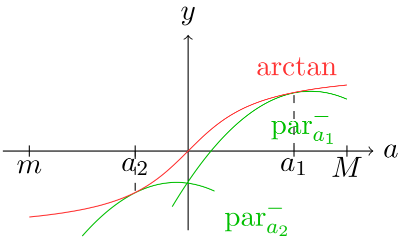


1. For $k = 2, m_1 = -0.746 < 0$, obtain a new minimizer $\mathbf{x}^2_{opt}$.
2. Compute $a_2 := f_{\text{sa}}(\mathbf{x}^2_{opt}) = -0.374$ and $\text{par}^-_{a_2}$
3. Compute $m_2 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \max_{i \in \{1,2\}} \{\text{par}^-_{a_i}(f_{\text{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

`samp_optim` Second iteration:

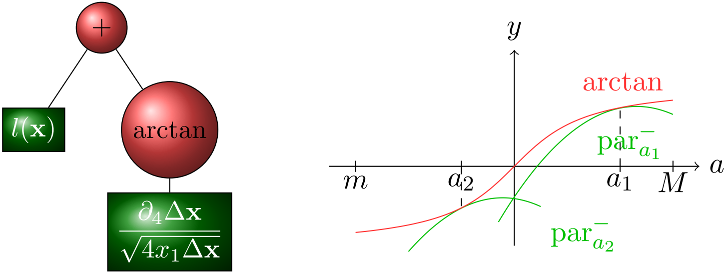




1. For $k = 2, m_1 = -0.746 < 0$, obtain a new minimizer $\mathbf{x}^2_{opt}$.
2. Compute $a_2 := f_{\text{sa}}(\mathbf{x}^2_{opt}) = -0.374$ and $\text{par}^-_{a_2}$
3. Compute $m_2 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \max_{i \in \{1,2\}} \{\text{par}^-_{a_i}(f_{\text{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

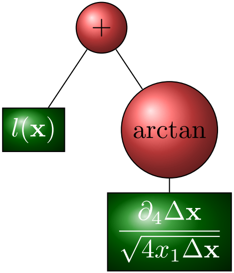
samp_optim Second iteration:

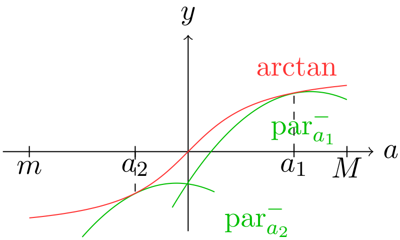


1. For $k = 2, m_1 = -0.746 < 0$, obtain a new minimizer $\mathbf{x}^2_{opt}$.
2. Compute $a_2 := f_{\text{sa}}(\mathbf{x}^2_{opt}) = -0.374$ and $\mathrm{par}^-_{a_2}$
3. Compute $m_2 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \max_{i \in \{1,2\}} \{\mathrm{par}^-_{a_i}(f_{\text{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

`samp_optim` Second iteration:

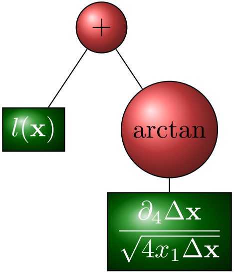


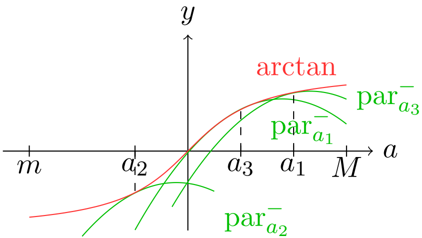


1. For $k = 2, m_1 = -0.746 < 0$, obtain a new minimizer $\mathbf{x}_{opt}^2$.
2. Compute $a_2 := f_{\text{sa}}(\mathbf{x}_{opt}^2) = -0.374$ and $\text{par}_{a_2}^-$
3. Compute $m_2 \leq \min\limits_{\mathbf{x} \in [4,6.3504]} (l(\mathbf{x}) + \max\limits_{i \in \{1,2\}} \{\text{par}_{a_i}^-(f_{\text{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

samp_optim Third iteration:

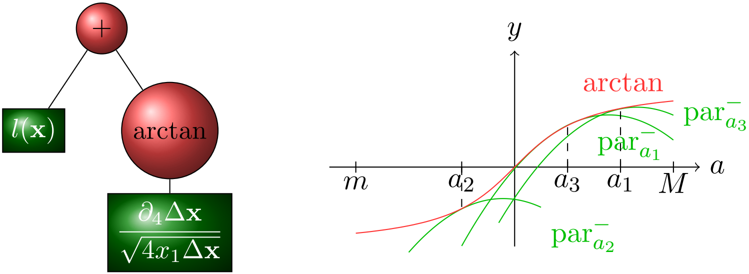




1. For $k = 2, m_2 = -0.112 < 0$, obtain a new minimizer $\mathbf{x}^3_{opt}$.
2. Compute $a_3 := f_{\text{sa}}(\mathbf{x}^3_{opt}) = 0.357$ and $\text{par}^-_{a_3}$
3. Compute $m_3 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \max_{i \in \{1,2,3\}} \{\text{par}^-_{a_i}(f_{\text{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

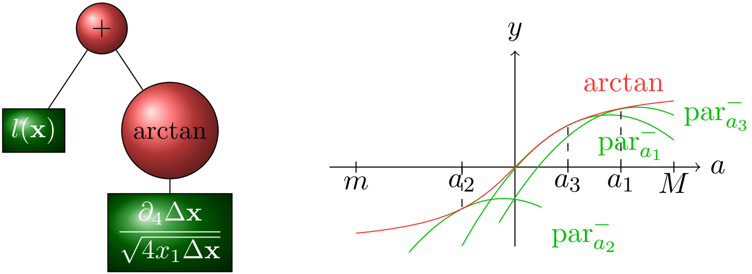
samp_optim Third iteration:



1. For $k = 2, m_2 = -0.112 < 0$, obtain a new minimizer $\mathbf{x}^3_{opt}$.
2. Compute $a_3 := f_{\mathrm{sa}}(\mathbf{x}^3_{opt}) = 0.357$ and $\mathrm{par}^-_{a_3}$
3. Compute $m_3 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \max_{i \in \{1,2,3\}} \{\mathrm{par}^-_{a_i}(f_{\mathrm{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

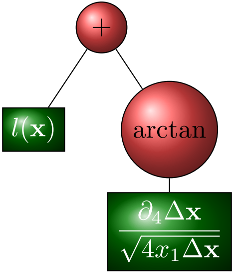
`samp_optim` Third iteration:

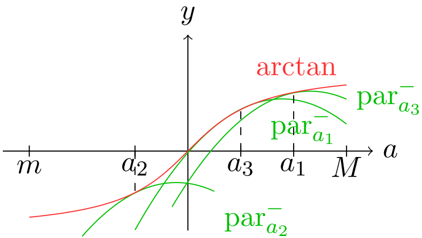


1. For $k = 2, m_2 = -0.112 < 0$, obtain a new minimizer $\mathbf{x}_{opt}^3$.
2. Compute $a_3 := f_{\text{sa}}(\mathbf{x}_{opt}^3) = 0.357$ and $\text{par}_{a_3}^-$
3. Compute $m_3 \leq \min_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \max_{i \in \{1,2,3\}} \{\text{par}_{a_i}^-(f_{\text{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

samp_optim Third iteration:





1. For $k = 2, m_2 = -0.112 < 0$, obtain a new minimizer $\mathbf{x}^3_{opt}$.
2. Compute $a_3 := f_{\mathsf{sa}}(\mathbf{x}^3_{opt}) = 0.357$ and $\mathrm{par}^-_{a_3}$
3. Compute $m_3 \leq \min\limits_{\mathbf{x} \in [4, 6.3504]} (l(\mathbf{x}) + \max\limits_{i \in \{1,2,3\}} \{\mathrm{par}^-_{a_i}(f_{\mathsf{sa}}(\mathbf{x}))\})$

Semialgebraic Max-Plus Algorithm

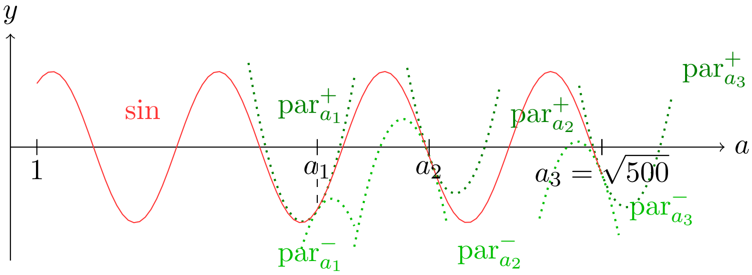
- For $k = 2, m_3 = -0.0333 < 0$, obtain a new minimizer $\mathbf{x}^4_{opt}$ and iterate again...

Theorem: Convergence of Semialgebraic underestimators

Let $f : K \to \mathbb{R}$ be a multivariate transcendental function.
Let $(\mathbf{x}^p_{opt})_{p \in \mathbb{N}}$ be a sequence of control points. Suppose that $(\mathbf{x}^p_{opt})_{p \in \mathbb{N}} \to \mathbf{x}^*$.
Then, $\mathbf{x}^*$ is a global minimizer of f on K.

Max-Plus Based Templates Approach

Example with sin:



Benchmarks

$$\min_{\mathbf{x}\in[1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n} (x_i + \epsilon x_{i+1}) \sin(\sqrt{x_i})$$

n	lower bound	n_{lifting}	#boxes	time
$10 (\epsilon = 0)$	$-430n$	$2n$	16	$40\,s$
$10 (\epsilon = 0)$	$-430n$	0	827	$177\,s$

Benchmarks

- Comparison with Newton interval method (`intsolver`)
- Random sparse polynomials: quartic p, quadratic form q
- $\mathbf{x} \mapsto \arctan(p(\mathbf{x})) + \arctan(q(\mathbf{x}))$, with $\mathbf{x} \in [0, 1]^n$

n	`samp_optim`		`intsolver`	
	m	time	m	time
3	0.4581	3.8 *s*	0.4581	15.5 *s*
4	0.4157	12.9 *s*	0.4157	172.1 *s*
5	0.4746	1 *min*	0.4746	10.2 *min*
6	0.4476	4.6 *min*	0.4476	3.4 *h*

Benchmarks

- $n = 6$ variables, SOS of degree $2k = 4$
- $n_{\mathcal{T}}$ univariate transcendental functions, #boxes sub-problems

Inequality id	$n_{\mathcal{T}}$	n_{lifting}	#boxes	time
9922699028	1	9	47	$241\,s$
7726998381	3	15	70	$43\,min$
4652969746_1	6	15	81	$1.3\,h$

Hybrid Symbolic-Numeric Issues

Formal proofs for lower bounds of POP:

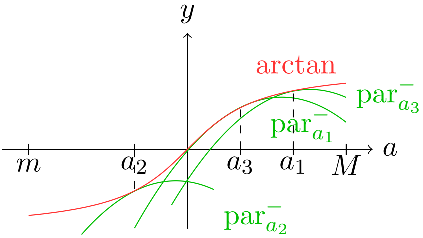
- The SDP solver returns floating point certificate: $\mu, \sigma_0, \cdots, \sigma_m$
- Check equality of polynomials: $f(\mathbf{x}) - \mu = \sum_{i=0}^{m} \sigma_i(\mathbf{x}) g_i(\mathbf{x})$ with a proof assistant tactic.

Hybrid Symbolic-Numeric Issues

- The equality test often fails. Workaround:

Bounds $f(\mathbf{x}) - \mu - \sum_{i=0}^{m} \sigma_i(\mathbf{x}) g_i(\mathbf{x}) = \sum_{\alpha} \epsilon_\alpha \mathbf{x}^\alpha$ since $\mathbf{x} \in [\mathbf{a}, \mathbf{b}]$

- Formal proofs for Max-Plus estimators



Exploiting System Properties

- Max-plus estimators preserve system properties: Sparsity / Symmetries
- Implementation in OCaml of the sparse variant of SOS relaxations (Kojima) for POP and semialgebraic underestimators
- Reducing the size of SOS input data has a positive domino effect:
  1. on the global optimization oracle to decrease the $O(n^{2d})$ complexity
  2. to check SOS formally with proof assistants (Coq)

End

Thank you for your attention!

Questions?