Formal Proofs for Global Optimization

Templates and Sums of Squares

Victor MAGRON

INRIA-LIX/CMAP, École Polytechnique

2013 December 9th

Mathematics and Computer Science

- Mathematicians want to eliminate all the uncertainties on their results. Why?

  M. Lecat, Erreurs des Mathématiciens des origines à nos jours, 1935.

  130 pages of errors! (Euler, Fermat, Sylvester, . . .)

- Possible workaround: proof assistants

Mathematics and Computer Science

- Mathematicians want to eliminate all the uncertainties on their results. Why?

  M. Lecat, Erreurs des Mathématiciens des origines à nos jours, 1935.

  130 pages of errors! (Euler, Fermat, Sylvester, . . .)
- Possible workaround: proof assistants

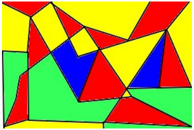
Why using Proof Assistants?

- Proof assistant: piece of software
- Implements a logical formalism
- Precise and formal definitions of propositions and proofs
- Correctness of proofs defined by logical rules
- Proofs are formally checked by the kernel (small trusting base)

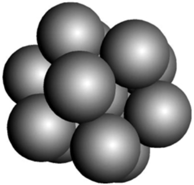
Complex Proofs

- Complex mathematical proofs / mandatory computation

K. Appel and W. Haken , Every Planar Map is Four-Colorable, 1989.



T. Hales, A Proof of the Kepler Conjecture, 1994.



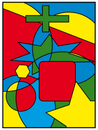
Computational Proofs: in a nutshell

Formalized in the COQ proof assistant [Gonthier 08]

Objects: configurations (`conf`)

Property: `red` (reducible)

`red c` | proved by a program `P` |



In COQ `P : conf -> bool.`

```
Lemma red_ok: ∀ c, P c = true -> red c.
```

- Graspable proof size
- Also automated

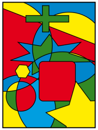
Computational Proofs: in a nutshell

Formalized in the Coq proof assistant [Gonthier 08]

Objects: configurations (`conf`)

Property: `red` (reducible)

`red c` | proved by a program P |



In Coq `P : conf -> bool.`

```
Lemma red_ok: ∀ c, P c = true -> red c.
```

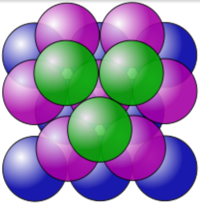
- Graspable proof size
- Also automated

Verifying the proof of Kepler Conjecture by Hales leads to difficulties of a different nature!

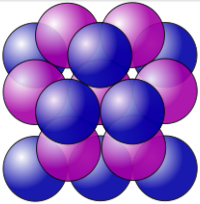
The Kepler Conjecture

Kepler Conjecture (1611):

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



Face-centered cubic Packing



Hexagonal Compact Packing

Motivation: Flyspeck Nonlinear Inequalities

- The proof of T. Hales (1998) contains mathematical and computational parts
- Computation: check thousands of nonlinear inequalities
- Robert MacPherson, editor of The Annals of Mathematics: "[...] the mathematical community will have to get used to this state of affairs."
- **Flyspeck** [Hales 06]: **F**ormal **P**roof of **K**epler Conjecture

Motivation: Flyspeck Nonlinear Inequalities

In the computational part:

- Feasible set $K := [4, 6.3504]^3 \times [6.3504, 8] \times [4, 6.3504]^2$
- $\Delta\mathbf{x} := x_1 x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2 x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3 x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3 x_4 + x_1 x_6) - x_5(x_1 x_3 + x_4 x_6)$
- $l(\mathbf{x}) := -\frac{\pi}{2} + 1.6294 - 0.2213\,(\sqrt{x_2} + \sqrt{x_3} + \sqrt{x_5} + \sqrt{x_6} - 8.0) + 0.913\,(\sqrt{x_4} - 2.52) + 0.728\,(\sqrt{x_1} - 2.0)$

Lemma$_{9922699028}$ from Flyspeck:

$$\forall \mathbf{x} \in K, \arctan\left(\frac{\partial_4 \Delta\mathbf{x}}{\sqrt{4 x_1 \Delta\mathbf{x}}}\right) + l(\mathbf{x}) \geqslant 0$$

Motivation: Flyspeck Nonlinear Inequalities

Inequalities issued from Flyspeck nonlinear 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. Semialgebraic functions algebra $\mathcal{A}$: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \dots$
3. Transcendental functions $\mathcal{T}$: composition of semialgebraic functions with arctan, exp, sin, $+, -, \times, \dots$

Motivation: Flyspeck Nonlinear Inequalities

Inequalities issued from Flyspeck nonlinear part involve:

1. Multivariate Polynomials:
   $x_1 x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2 x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3 x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3 x_4 + x_1 x_6) - x_5(x_1 x_3 + x_4 x_6)$
2. Semialgebraic functions algebra $\mathcal{A}$: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \dots$
3. Transcendental functions $\mathcal{T}$: composition of semialgebraic functions with arctan, exp, sin, $+, -, \times, \dots$

Motivation: Flyspeck Nonlinear Inequalities

Inequalities issued from Flyspeck nonlinear part involve:

1. Multivariate Polynomials:
   $x_1 x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2 x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3 x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3 x_4 + x_1 x_6) - x_5(x_1 x_3 + x_4 x_6)$
2. Semialgebraic functions algebra $\mathcal{A}$: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \dots$
3. Transcendental functions $\mathcal{T}$: composition of semialgebraic functions with arctan, exp, sin, $+, -, \times, \dots$

Motivation: Global Optimization Problems

From the Literature [Appendix B, Ali et al. 05]

Issued from transistor modelling, aircraft design, medicine, ...

- *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*: $\min_{\substack{-1.5 \leqslant x_1 \leqslant 4 \\ -3 \leqslant x_2 \leqslant 3}} \sin(x_1 + x_2) + (x_1 - x_2)^2 - 0.5x_2 + 2.5x_1 + 1$
- *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*: $\min_{\mathbf{x} \in [1,500]^n} -\sum_{i=1}^{n-1} (x_i + \epsilon x_{i+1}) \sin(\sqrt{x_i}) \quad (\epsilon \in \{0, 1\})$

Context

Formal proofs for Global Optimization:

- Bernstein polynomial methods [Zumkeller thesis 08]
  - Restricted to polynomials
- Certified interval arithmetic in COQ [Melquiond 12]
- Taylor methods in HOL Light [Solovyev thesis 13]
  - Formal verification of floating-point operations
- SMT methods [Gao et al. 12]
- Sums of squares techniques
  - Formalized in HOL-LIGHT [Harrison 07] COQ [Besson 07]
  - Precise methods but scalibility and robustness issues (numerical)
  - Restricted to polynomials

Context

Formal proofs for Global Optimization:

- Bernstein polynomial methods [Zumkeller thesis 08]
  - Restricted to polynomials
- Certified interval arithmetic in COQ [Melquiond 12]
- Taylor methods in HOL Light [Solovyev thesis 13]
  - Formal verification of floating-point operations
- SMT methods [Gao et al. 12]
- Sums of squares techniques
  - Formalized in HOL-LIGHT [Harrison 07] COQ [Besson 07]
  - Precise methods but scalibility and robustness issues (numerical)
  - Restricted to polynomials

Context

Formal proofs for Global Optimization:

- Bernstein polynomial methods [Zumkeller thesis 08]
  - Restricted to polynomials
- Certified interval arithmetic in COQ [Melquiond 12]
- Taylor methods in HOL Light [Solovyev thesis 13]
  - Formal verification of floating-point operations
- SMT methods [Gao et al. 12]
- Sums of squares techniques
  - Formalized in HOL-LIGHT [Harrison 07] COQ [Besson 07]
  - Precise methods but scalibility and robustness issues (numerical)
  - Restricted to polynomials

Context

Formal proofs for Global Optimization:

- Bernstein polynomial methods [Zumkeller thesis 08]
  - Restricted to polynomials
- Certified interval arithmetic in COQ [Melquiond 12]
- Taylor methods in HOL Light [Solovyev thesis 13]
  - Formal verification of floating-point operations
- SMT methods [Gao et al. 12]
- Sums of squares techniques
  - Formalized in HOL-LIGHT [Harrison 07] COQ [Besson 07]
  - Precise methods but scalibility and robustness issues (numerical)
  - Restricted to polynomials

Context

Formal proofs for Global Optimization:

- Bernstein polynomial methods [Zumkeller thesis 08]
  - Restricted to polynomials
- Certified interval arithmetic in COQ [Melquiond 12]
- Taylor methods in HOL Light [Solovyev thesis 13]
  - Formal verification of floating-point operations
- SMT methods [Gao et al. 12]
- Sums of squares techniques
  - Formalized in HOL-LIGHT [Harrison 07] COQ [Besson 07]
  - Precise methods but scalibility and robustness issues (numerical)
  - Restricted to polynomials

Existing Frameworks

Interval analysis

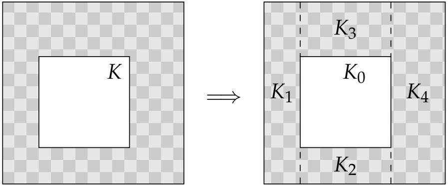
- robust but subject to the curse of dimensionality

Existing Frameworks

Lemma$_{9922699028}$ from Flyspeck:

$$\forall \mathbf{x} \in K, \arctan\left(\frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4x_1 \Delta \mathbf{x}}}\right) + l(\mathbf{x}) \geqslant 0$$

- Dependency issue using Interval Calculus:
  - One can bound $\partial_4 \Delta \mathbf{x} / \sqrt{4x_1 \Delta \mathbf{x}}$ and $l(\mathbf{x})$ separately
  - Too coarse lower bound: -0.87
  - Subdivide K to prove the inequality



  - **Curse of Dimensionality**

Existing Frameworks

Sums of squares techniques

- powerful: global optimality certificates without branching

but

- not so robust: handles moderate size problems

Existing Frameworks

Approximation theory: Chebyshev/Taylor models

- mandatory for non-polynomial problems
- hard to combine with SOS techniques (degree of approximation)

Question

Can we develop a new approach with both keeping the respective strength of interval and precision of SOS?

Proving Flyspeck Inequalities is challenging: medium-size and tight

Answer

- Certificates for lower bounds of Global Optimization Problems using SOS and new ingredients in Global Optimization:
  - Maxplus approximation (Optimal Control)
  - Nonlinear templates (Static Analysis)
- Verification of these certificates inside COQ
- Implementation of all these techniques in `NLCertify`

The General Framework

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

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 (POP)
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 SOS relaxations

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

Outline

Polynomial Optimization Problems (POP)

- Input data: multivariate polynomials $p, g_1, \dots, g_m \in \mathbb{R}[\mathbf{x}]$
- $K := \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geqslant 0, \dots, g_m(\mathbf{x}) \geqslant 0\}$ is a semialgebraic set
- How to certify a lower bound of $p^* := \inf_{\mathbf{x} \in K} p(\mathbf{x})$?

Example with the box $[4, 6.3504]^6$

- $g_1 := x_1 - 4, g_2 := 6.3504 - x_1, \dots, g_{11} := x_6 - 4, g_{12} := 6.3504 - x_6$
- $K := \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geqslant 0, \dots, g_{12}(\mathbf{x}) \geqslant 0\}$
- $\Delta(\mathbf{x}) := x_1x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3x_4 + x_1x_6) - x_5(x_1x_3 + x_4x_6)$

Polynomial Optimization Problems (POP)

- Input data: multivariate polynomials $p, g_1, \ldots, g_m \in \mathbb{R}[\mathbf{x}]$
- $K := \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geqslant 0, \ldots, g_m(\mathbf{x}) \geqslant 0\}$ is a semialgebraic set
- How to certify a lower bound of $p^* := \inf_{\mathbf{x} \in K} p(\mathbf{x})$?

Example with the box $[4, 6.3504]^6$

- $g_1 := x_1 - 4, g_2 := 6.3504 - x_1, \ldots, g_{11} := x_6 - 4, g_{12} := 6.3504 - x_6$
- $K := \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geqslant 0, \ldots, g_{12}(\mathbf{x}) \geqslant 0\}$
- $\Delta(\mathbf{x}) := x_1 x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2 x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3 x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3 x_4 + x_1 x_6) - x_5(x_1 x_3 + x_4 x_6)$

The Cone of Sums of Squares

- Let $\Sigma[\mathbf{x}]$ be the cone of sums of squares (SOS)
- Let $g_0 := 1$ and $M(\mathbf{g})$ be the quadratic module generated by $g_0, \ldots, g_m$:

$$M(\mathbf{g}) = \left\{ \sum_{j=0}^{m} \sigma_j(\mathbf{x}) g_j(\mathbf{x}), \text{ with } \sigma_j \in \Sigma[\mathbf{x}] \right\}$$

- When $q \in M(\mathbf{g})$, $\sigma_0, \ldots, \sigma_m$ is a positivity certificate for q
  $q = q'$ can be checked in COQ
- Much simpler to verify certificates using *sceptical approach*

The Cone of Sums of Squares

- Let $\Sigma[\mathbf{x}]$ be the cone of sums of squares (SOS)
- Let $g_0 := 1$ and $M(\mathbf{g})$ be the quadratic module generated by $g_0, \dots, g_m$:

$$M(\mathbf{g}) = \left\{ \sum_{j=0}^{m} \sigma_j(\mathbf{x}) g_j(\mathbf{x}), \ \text{with } \sigma_j \in \Sigma[\mathbf{x}] \right\}$$

- When $q \in M(\mathbf{g})$, $\sigma_0, \dots, \sigma_m$ is a positivity certificate for q

  q = q′ can be checked in COQ
- Much simpler to verify certificates using *sceptical approach*

The Lasserre Hierarchy of SOS Relaxations

- $K := \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geqslant 0, \ldots, g_m(\mathbf{x}) \geqslant 0\}$, $p^* := \inf_{\mathbf{x} \in K} p(\mathbf{x})$?

Definition

$M(\mathbf{g})$ is Archimedean if there exists a positive constant ρ such that the polynomial $\mathbf{x} \mapsto \rho - \|\mathbf{x}\|_2^2$ belongs to $M(\mathbf{g})$.

Proposition [Putinar 93]

Suppose that $M(g)$ is Archimedean. Then, every polynomial strictly positive on K belongs to $M(\mathbf{g})$.

The Lasserre Hierarchy of SOS Relaxations

- The search space for $\sigma_0, \ldots, \sigma_m \in \Sigma[\mathbf{x}]$ is infinite
- Consider the truncated quadratic module:

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

$$M_0(\mathbf{g}) \subset M_1(\mathbf{g}) \subset M_2(\mathbf{g}) \subset \cdots \subset M(\mathbf{g})$$

- Hierarchy of SOS programs: $\mu_k := \sup_{\mu, \sigma_0, \ldots, \sigma_m} \left\{ \mu : p(\mathbf{x}) - \mu \in M_k(\mathbf{g}) \right\}$

Convergence of Lasserre Hierarchy

Proposition [Lasserre 01]

- Let $k \geqslant k_0 := \max\{\lceil \deg p/2 \rceil, \lceil \deg g_1/2 \rceil, \dots, \lceil \deg g_m/2 \rceil\}$.
- The sequence $\inf(\mu_k)_{k \geqslant k_0}$ is non-decreasing. When $M(\mathbf{g})$ is Archimedean, it converges to p^*.

- Compute μ_k by solving a semidefinite program (SDP)
- External tools: SDP solvers freely available (SDPA, CSDP, . . .)

How to Deal with Semialgebraic Expressions?

- Let $\mathcal{A}$ be the semialgebraic functions algebra obtained by composition of polynomials with $|\cdot|$, $(\cdot)^{\frac{1}{p}} (p \in \mathbb{N}_0)$, $+, -, \times, /, \sup, \inf$
- Example: $f_{\text{sa}}(\mathbf{x}) := \dfrac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}}$
- $K := \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geqslant 0, \dots, g_m(\mathbf{x}) \geqslant 0\}$ is a semialgebraic set
- $f_{\text{sa}}^* := \inf_{\mathbf{x} \in K} f_{\text{sa}}(\mathbf{x})$?

How to Deal with Semialgebraic Expressions?

Definition: Basic semialgebraic lifting (b.s.a.l)

A semialgebraic function f_{sa} is said to have a b.s.a.l if there exist $p, s \in \mathbb{N}$, polynomials $h_1, \dots, h_s \in \mathbb{R}[\mathbf{x}, z_1, \dots, z_p]$ and a basic semialgebraic set K_{pop} defined by:

$$K_{\text{pop}} := \{(\mathbf{x}, z_1, \dots, z_p) \in \mathbb{R}^{n+p} : \mathbf{x} \in K, h_1(\mathbf{x}, \mathbf{z}) \geqslant 0, \dots, h_s(\mathbf{x}, \mathbf{z}) \geqslant 0\} \ ,$$

with $\{(\mathbf{x}, f_{\text{sa}}(\mathbf{x})) : \mathbf{x} \in K\} = \{(\mathbf{x}, z_p) : (\mathbf{x}, \mathbf{z}) \in K_{\text{pop}}\}$.

b.s.a.l. lemma [Lasserre-Putinar 10] :

Every well-defined $f_{\text{sa}} \in \mathcal{A}$ has a basic semialgebraic lifting.

The “No Free Lunch” Rule

- Dependency in the relaxation order k (SOS degree) and the number of variables n
- Computing μ_k leads to an SOS with $\binom{n+2k}{n}$ variables
- At k fixed, $O(n^{2k})$ variables

Examples

Previous Example

- $g_1 := x_1 - 4, g_2 := 6.3504 - x_1, \ldots, g_{11} := x_6 - 4, g_{12} := 6.3504 - x_6$
- $K := \{\mathbf{x} \in \mathbb{R}^n : g_1(\mathbf{x}) \geqslant 0, \ldots, g_{12}(\mathbf{x}) \geqslant 0\}$
- $\Delta\mathbf{x} := x_1 x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2 x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3 x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3 x_4 + x_1 x_6) - x_5(x_1 x_3 + x_4 x_6)$

With SOS of degree at most 4: $\mu_2 = 128$

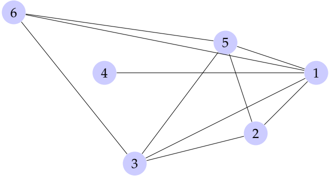
Lemma from Flyspeck (inequality ID 4717061266)

$$\forall \mathbf{x} \in [4, 6.3504]^6, \Delta\mathbf{x} \geqslant 0$$

Sparse Variant of SOS Relaxations

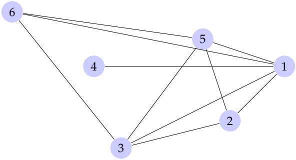
- Partial Remedy: Sparse variant of SOS Relaxations [Waki et al. 04]
- Correlative sparsity pattern (csp) graph for the POP variables

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



Sparse Variant of SOS Relaxations

- csp graph G for the POP variables
- Compute $C_1, \dots, C_l$ the maximal cliques of G
- Let κ be the average size of the cliques
- Hierarchy of SOS Relaxations involving $\binom{\kappa + 2k}{\kappa}$ variables



$$C_1 := \{1,4\}, C_2 := \{1,2,3,5\}, C_3 := \{1,3,5,6\}$$

Dense SOS: 210 variables

Sparse SOS: 115 variables

But only a partial remedy!

Examples

Example from Flyspeck

$$K := [4, 6.3504]^3 \times [6.3504, 8] \times [4, 6.3504]^2$$

$$f_{\text{sa}}(\mathbf{x}) := \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}}$$

Two lifting variables z_1, z_2 to represent the square root and the division

Examples

Example from Flyspeck

$z_1 := \sqrt{4 x_1 \Delta \mathbf{x}}$, $m_1 = \inf_{\mathbf{x} \in K} z_1(\mathbf{x})$, $M_1 = \sup_{\mathbf{x} \in K} z_1(\mathbf{x})$.

$K_{\text{pop}} := \{(\mathbf{x}, \mathbf{z}) \in \mathbb{R}^8 : \mathbf{x} \in K, h_1(\mathbf{x}, \mathbf{z}) \geqslant 0, \dots, h_6(\mathbf{x}, \mathbf{z}) \geqslant 0\}$, with

$$
\begin{aligned}
h_1(\mathbf{x}, \mathbf{z}) &:= z_1 - m_1 \ , & h_4(\mathbf{x}, \mathbf{z}) &:= -z_1^2 + 4 x_1 \Delta \mathbf{x} \ , \\
h_2(\mathbf{x}, \mathbf{z}) &:= M_1 - z_1 \ , & h_5(\mathbf{x}, \mathbf{z}) &:= z_2 z_1 - \partial_4 \Delta \mathbf{x} \ , \\
h_3(\mathbf{x}, \mathbf{z}) &:= z_1^2 - 4 x_1 \Delta \mathbf{x} \ , & h_6(\mathbf{x}, \mathbf{z}) &:= -z_2 z_1 + \partial_4 \Delta \mathbf{x} \ .
\end{aligned}
$$

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

More complex certificates

Examples

Example from Flyspeck

$z_1 := \sqrt{4x_1 \Delta \mathbf{x}}$, $m_1 = \inf_{\mathbf{x} \in K} z_1(\mathbf{x})$, $M_1 = \sup_{\mathbf{x} \in K} z_1(\mathbf{x})$.

$K_{\text{pop}} := \{(\mathbf{x}, \mathbf{z}) \in \mathbb{R}^8 : \mathbf{x} \in K, h_1(\mathbf{x}, \mathbf{z}) \geqslant 0, \ldots, h_6(\mathbf{x}, \mathbf{z}) \geqslant 0\}$, with

$$h_1(\mathbf{x}, \mathbf{z}) := z_1 - m_1 \ , \qquad h_4(\mathbf{x}, \mathbf{z}) := -z_1^2 + 4x_1 \Delta \mathbf{x} \ ,$$
$$h_2(\mathbf{x}, \mathbf{z}) := M_1 - z_1 \ , \qquad h_5(\mathbf{x}, \mathbf{z}) := z_2 z_1 - \partial_4 \Delta \mathbf{x} \ ,$$
$$h_3(\mathbf{x}, \mathbf{z}) := z_1^2 - 4x_1 \Delta \mathbf{x} \ , \qquad h_6(\mathbf{x}, \mathbf{z}) := -z_2 z_1 + \partial_4 \Delta \mathbf{x} \ .$$

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

More complex certificates

High-degree Polynomial Approximation + SOS

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

Classical idea:

- replace $\sin(\sqrt{\cdot})$ by a degree-d Chebyshev polynomial
- Hard to combine with SOS

Indeed:

- Small d: lack of accuracy $\Longrightarrow$ expensive Branch and Bound
- Large d: "No free lunch" rule with $\binom{n+d}{n}$ SOS variables

SWF with $n = 10, d = 4$:

- 38 *min* to compute a lower bound of $-430n$

High-degree Polynomial Approximation + SOS

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

Classical idea:

- replace $\sin(\sqrt{\cdot})$ by a degree-d Chebyshev polynomial
- Hard to combine with SOS

Indeed:

- Small d: lack of accuracy $\Longrightarrow$ expensive Branch and Bound
- Large d: "No free lunch" rule with $\binom{n+d}{n}$ SOS variables

SWF with $n = 10, d = 4$:

- 38 *min* to compute a lower bound of $-430n$

High-degree Polynomial Approximation + SOS

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

Classical idea:

- replace $\sin(\sqrt{\cdot})$ by a degree-d Chebyshev polynomial
- Hard to combine with SOS

Indeed:

- Small d: lack of accuracy $\implies$ expensive Branch and Bound
- Large d: "No free lunch" rule with $\binom{n+d}{n}$ SOS variables

SWF with $n = 10, d = 4$:

- 38 *min* to compute a lower bound of $-430n$

High-degree Polynomial Approximation + SOS

Minimax approximations + Sparse SOS not enough to check the hardest inequalities (multiple variables, multiple semialgebraic lifting)

Outline

Maxplus Approximation

- Initially introduced to solve Optimal Control Problems [Fleming-McEneaney 00]
- Further work by [McEneaney 07, Akian-Gaubert-Lakhoua 08, Dower]
- Value function approximated with “maxplus linear combination” of simple (e.g. quadratic) functions
- Curse of dimensionality reduction [McEaneney Kluberg, Gaubert-McEneaney-Qu 11, Qu 13]. Allowed to solve instances of dim up to 15 (inaccessible by grid methods)
- In our context: approximate transcendental functions

Maxplus Approximation for Semiconvex Functions

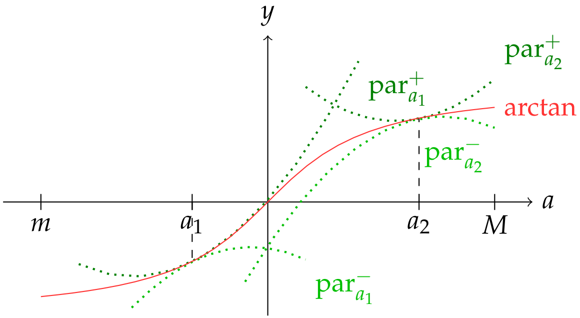
Definition: Semiconvex function

Let $\gamma \geqslant 0$. A function $\phi : \mathbb{R}^n \to \mathbb{R}$ is said to be γ-*semiconvex* if the function $\mathbf{x} \mapsto \phi(\mathbf{x}) + \frac{\gamma}{2} \|\mathbf{x}\|_2^2$ is convex.

Proposition

The set of functions f which can be written as the previous maxplus linear combination for some function $a : \mathcal{B} \to \mathbb{R} \cup \{-\infty\}$ is precisely the set of lower semicontinuous γ-semiconvex functions.

Maxplus Approximation for Semiconvex Functions



Maxplus Approximation Error

Theorem [Akian-Gaubert-Lakhoua 08]

Let $\gamma \in \mathbb{R}$, $\eta > 0$. Let ϕ be $(\gamma - \eta)$-semiconvex and Lipschitz-continuous on a full dimensional compact convex subset $K \subset \mathbb{R}^n$. Let ϕ_N denote the best maxplus approximation by N quadratic forms of Hessian $-\gamma I$. Then $\|\phi - \phi_N\|_\infty = O(1/N^{2/n})$.

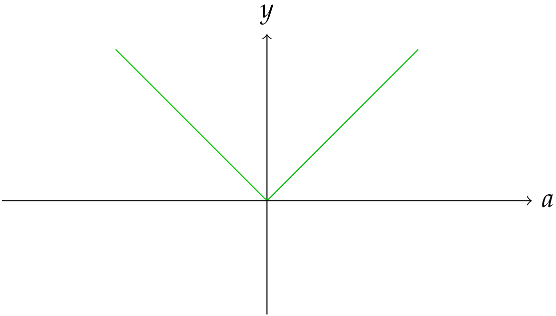
- **Differentiability not mandatory** by contrast with Taylor
- When in addition, ϕ is of class $\mathcal{C}^2$, then the upper bound is tight [Gaubert-McEneaney-Qu 11]

$$\|\phi - \phi_N\|_\infty \sim \frac{\alpha}{N^{2/n}} \Big(\int_K [\det(\mathcal{D}^2(\phi)(\mathbf{x}) + \gamma I_n)]^{\frac{1}{2}} d\mathbf{x} \Big)^{\frac{2}{n}} \text{ as } N \to \infty \ .$$

- In our case $n = 1$, one needs $O(1/\sqrt{\epsilon})$ basis functions

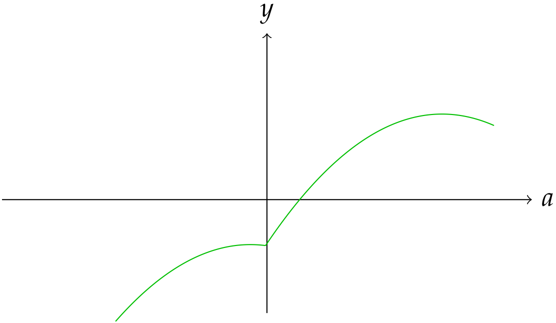
Maxplus Approximation Error

Exact parsimonious maxplus representations



Maxplus Approximation Error

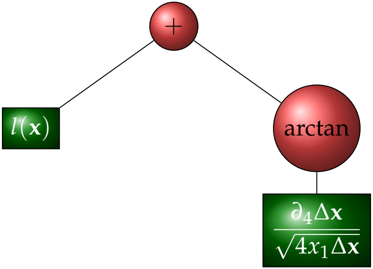
Exact parsimonious maxplus representations



Nonlinear Function Representation

Abstract syntax tree representations of multivariate transcendental functions:

- leaves are semialgebraic functions of $\mathcal{A}$
- nodes are univariate functions of $\mathcal{D}$ or binary operations
- For the "Simple" Example from Flyspeck:



Contents

Nonlinear Maxplus Approximation Algorithm

Input: tree t, box K, SOS relaxation order k, precision p
Output: lower bound m, upper bound M, lower semialgebraic estimator t_2^-, upper semialgebraic estimator t_2^+

if $t \in \mathcal{A}$ **then** $t^- := t, t^+ := t$
else if $u := \texttt{root}(t) \in \mathcal{D}$ with child c **then**
    $m_c, M_c, c^-, c^+ := \texttt{samp\_approx}(c, K, k, p)$
    $I := [m_c, M_c]$
    $u^-, u^+ := \texttt{unary\_approx}(u, I, c, p)$
    $t^-, t^+ := \texttt{compose\_approx}(u, u^-, u^+, I, c^-, c^+)$
else if $\texttt{bop} := \texttt{root}(t)$ is a binary operation with children c_1 and c_2 **then**
    $m_i, M_i, c_i^-, c_i^+ := \texttt{samp\_approx}(c_i, K, k, p)$ for $i \in \{1, 2\}$
    $t^-, t^+ := \texttt{compose\_bop}(c_1^-, c_1^+, c_2^-, c_2^+, \texttt{bop}, [m_2, M_2])$
end
return $\texttt{min\_sa}(t^-, K, k), \texttt{max\_sa}(t^+, K, k), t^-, t^+$

Nonlinear Maxplus Approximation Algorithm

Input: tree t, box K, SOS relaxation order k, precision p

Output: lower bound m, upper bound M, lower semialgebraic estimator t_2^-, upper semialgebraic estimator t_2^+

if $t \in \mathcal{A}$ **then** $t^- := t$, $t^+ := t$

else if $u := \texttt{root}(t) \in \mathcal{D}$ with child c **then**

  $m_c, M_c, c^-, c^+ := \texttt{samp\_approx}(c, K, k, p)$

  $I := [m_c, M_c]$

  $u^-, u^+ := \texttt{unary\_approx}(u, I, c, p)$

  $t^-, t^+ := \texttt{compose\_approx}(u, u^-, u^+, I, c^-, c^+)$

else if $\texttt{bop} := \texttt{root}(t)$ is a binary operation with children c_1 and c_2 **then**

  $m_i, M_i, c_i^-, c_i^+ := \texttt{samp\_approx}(c_i, K, k, p)$ for $i \in \{1, 2\}$

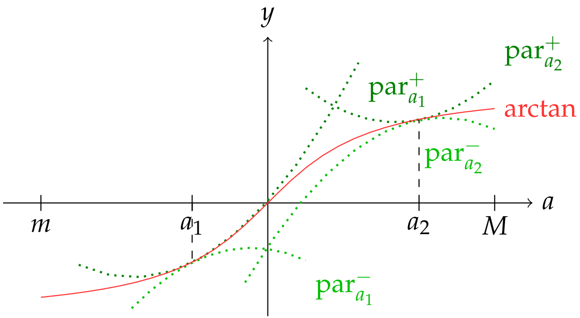
  $t^-, t^+ := \texttt{compose\_bop}(c_1^-, c_1^+, c_2^-, c_2^+, \texttt{bop}, [m_2, M_2])$

end

return $\texttt{min\_sa}(t^-, K, k), \texttt{max\_sa}(t^+, K, k), t^-, t^+$

Maxplus Approximation Example

- Consider the function arctan on $I := [m, M]$.
- $\arctan(x) \geqslant \mathrm{par}_a^-(x) := -\frac{\gamma}{2}(x-a)^2 + f'(a)(x-a) + f(a)$
- Choosing $\gamma = \sup_{x \in I} -f''(x)$ always work
- The precision p is the number of control points



Minimax Approximation / For Comparison

- More classical approximation method
- The precision is an integer d
- The best-uniform degree-d polynomial approximation of u is the solution of the following optimization problem:

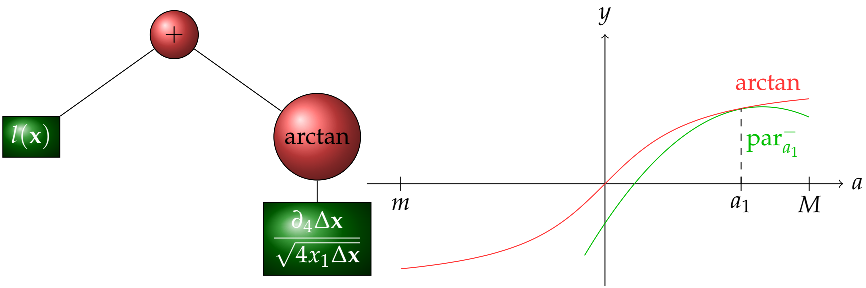
$$\min_{h \in \mathbb{R}_d[x]} \|u - h\|_\infty = \min_{h \in \mathbb{R}_d[x]} \left(\sup_{x \in I} |u(x) - h(x)| \right)$$

- Implementation in `Sollya` [Chevillard-Joldes-Lauter 10]

Contents

Nonlinear Maxplus Optimization Algorithm

First iteration:



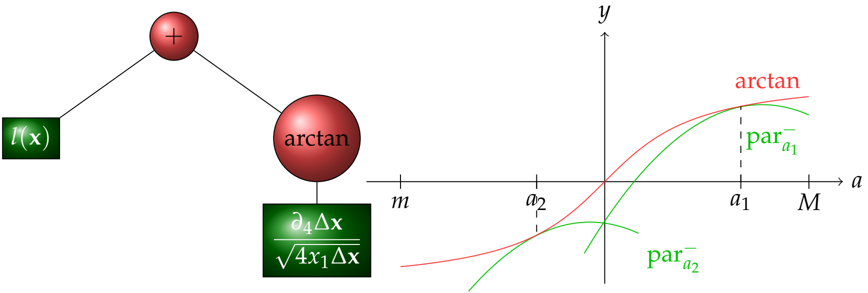
1. Evaluate t with `randeval` and obtain a minimizer guess $\mathbf{x}^1_{opt}$.

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

2. Compute $m_1 \leqslant \min_{\mathbf{x} \in K} (l(\mathbf{x}) + \mathrm{par}^-_{a_1}(f_{\mathrm{sa}}(\mathbf{x})))$

Nonlinear Maxplus Optimization Algorithm

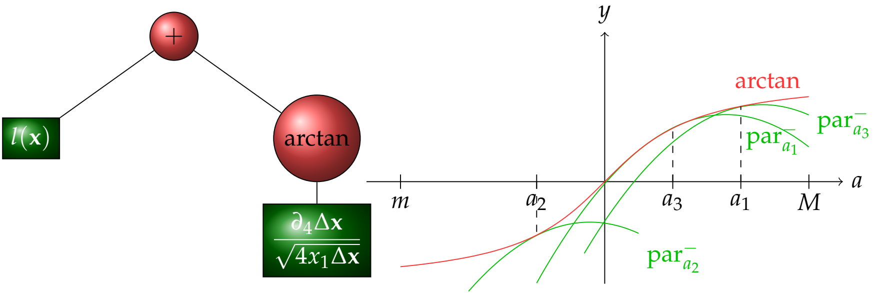
Second iteration:



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

Nonlinear Maxplus Optimization Algorithm

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 $m_3 \leqslant \min_{\mathbf{x}\in K}(l(\mathbf{x}) + \max_{i\in\{1,2,3\}} \{\mathrm{par}^-_{a_i}(f_{\mathrm{sa}}(\mathbf{x}))\})$

Nonlinear Maxplus Optimization Algorithm

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

Nonlinear Maxplus Optimization Algorithm

Input: abstract syntax tree t, semialgebraic set K, $iter_{\max}$ (optional argument), precision p

Output: lower bound m

$s := [\texttt{argmin}(\texttt{randeval}(t))]$ ▷ $s \in K$

$m := -\infty$, $iter := 0$

while $iter \leqslant iter_{\max}$ **do**

  Choose an SOS relaxation order $k \geqslant k_0$

  $m, M, t^-, t^+ := \texttt{samp\_approx}(t, K, k, p)$

  $\mathbf{x}_{opt} := \texttt{guess\_argmin}(t^-)$ ▷ $t^-(\mathbf{x}_{opt}) \simeq m$

  $s := s \cup \{\mathbf{x}_{opt}\}$

  $p := \texttt{update\_precision}(p)$, $iter := iter + 1$

done

return m, $\mathbf{x}_{opt}$

Convergence of the Optimization Algorithm

- Let f be a multivariate transcendental function
- Let t_p^- be the underestimator of f, obtained at precision p
- Let $\mathbf{x}_{opt}^p$ be a minimizer of t_p^- over K

Theorem

Every accumulation point of the sequence $(\mathbf{x}_{opt}^p)_p$ is a global minimizer of f on K.

Ingredients of the proof:

- Convergence of Lasserre SOS hierarchy
- Uniform approximation schemes (Maxplus/Minimax)

Numerical Results for Flyspeck

- Branch and bound subdivisions to reduce the relaxation gap: #boxes sub-problems
- $n = 6$ variables, SOS of degree $2k = 4$
- $n_{\mathcal{D}}$ univariate transcendental functions
- Maxplus arctan + Lifting $\sqrt{x_i}$

Inequality id	$n_{\mathcal{D}}$	n_{lifting}	#boxes	time
9922699028	1	9	47	241 s
3318775219	1	9	338	26 min
7726998381	3	15	70	43 min
7394240696	3	15	351	1.8 h
4652969746_1	6	15	81	1.3 h
OXLZLEZ 6346351218_2_0	6	24	200	5.7 h

Contents

Reducing the Number of Lifting Variables

- Lifting strategy: n_{lifting} increases with the number of control points and components of the semialgebraic functions
- At fixed relaxation order k, the number of SOS variables is in $O((n + n_{\text{lifting}})^{2k})$
- Improvements for more scalibility:
  1. Limit the blow-up at the price of coarsening the semialgebraic estimators
  2. Still produce certificates

Nonlinear Template Abstraction

- Linear templates in static analysis [Sankaranarayana-Sipma-Manna 05]
- Nonlinear extension [Adje-Gaubert-Goubault 12]

Nonlinear Template Approximation

- Invariants of programs with parametric families of subsets of $\mathbb{R}^n$ of the form $S(\alpha) = \{\mathbf{x} \mid w_i(\mathbf{x}) \leqslant \alpha_i,\ 1 \leqslant i \leqslant p\}$, where:
  - $\alpha \in \mathbb{R}^p$ is the parameter
  - $w_1, \ldots, w_p$ is the template
- Level sets of maxplus approximation $\Leftrightarrow$ templates description

Special cases of templates (w_i):

- bounds constraints $(\pm x_i)$: interval calculus
- degree-d minimax polynomials: Chebyshev approximation

Nonlinear Template Approximation

Input: tree t, box K, SOS relaxation order k, precision p

Output: lower bound m, upper bound M, lower semialgebraic estimator t_2^-, upper semialgebraic estimator t_2^+

if $t \in \mathcal{A}$ **then** $t^- := t, t^+ := t$
else if $u := \texttt{root}(t) \in \mathcal{D}$ with child c **then**
    $m_c, M_c, c^-, c^+ := \texttt{template\_approx}(c, K, k, p)$
    $I := [m_c, M_c]$
    $u^-, u^+ := \texttt{unary\_approx}(u, I, c, p)$
    $t^-, t^+ := \texttt{compose\_approx}(u, u^-, u^+, I, c^-, c^+)$
else if $\texttt{bop} := \texttt{root}(t)$ is a binary operation with children c_1 and c_2 **then**
    $m_i, M_i, c_i^-, c_i^+ := \texttt{template\_approx}(c_i, K, k, p)$ for $i \in \{1, 2\}$
    $t^-, t^+ := \texttt{compose\_bop}(c_1^-, c_1^+, c_2^-, c_2^+, \texttt{bop}, [m_2, M_2])$
end
$t_2^- := \texttt{reduce\_lift}(t, K, k, p, t^-), t_2^+ := -\texttt{reduce\_lift}(t, K, k, p, -t^+)$
return $\texttt{min\_sa}(t_2^-, K, k), \texttt{max\_sa}(t_2^+, K, k), t_2^-, t_2^+$

How to Construct Templates?

4 Nonlinear Templates
- Nonlinear Template Abstraction
- Nonlinear Template Approximation
- Nonlinear Quadratic Templates
- Polynomial Estimators for Semialgebraic Functions
- Comparison Results on Global Optimization Problems

Nonlinear Quadratic Templates

- Let $\mathbf{x}_1, \ldots, \mathbf{x}_p \in K$
- Quadratic underestimators of f over K:

$$
\begin{aligned}
f_{\mathbf{x}_c,\lambda'} : K &\longrightarrow \mathbb{R} \\
x &\longmapsto f(\mathbf{x}_c) + \mathcal{D}(f)(\mathbf{x}_c)\,(\mathbf{x} - \mathbf{x}_c) \\
&\quad + \frac{1}{2}(\mathbf{x} - \mathbf{x}_c)^T \mathcal{D}^2(f)(\mathbf{x}_c)(\mathbf{x} - \mathbf{x}_c) \\
&\quad + \frac{1}{2}\lambda' \|\mathbf{x} - \mathbf{x}_c\|_2^2 \ ,
\end{aligned}
$$

with $\lambda' \leqslant \lambda := \min_{\mathbf{x}\in K}\{\lambda_{\min}(\mathcal{D}^2(f)(\mathbf{x}) - \mathcal{D}^2(f)(\mathbf{x}_c))\}$.

- Computation of λ' can be certified (Robust SDP)

Nonlinear Quadratic Templates

- Let $\mathbf{x}_1, \ldots, \mathbf{x}_p \in K$
- Quadratic underestimators of f over K:

$$
\begin{aligned}
f_{\mathbf{x}_c, \lambda'} : K &\longrightarrow \mathbb{R} \\
x &\longmapsto f(\mathbf{x}_c) + \mathcal{D}(f)(\mathbf{x}_c)\,(\mathbf{x} - \mathbf{x}_c) \\
&\qquad + \frac{1}{2}(\mathbf{x} - \mathbf{x}_c)^T \mathcal{D}^2(f)(\mathbf{x}_c)(\mathbf{x} - \mathbf{x}_c) \\
&\qquad + \frac{1}{2}\lambda' \|\mathbf{x} - \mathbf{x}_c\|_2^2 \ ,
\end{aligned}
$$

with $\lambda' \leqslant \lambda := \min_{\mathbf{x} \in K} \{\lambda_{\min}(\mathcal{D}^2(f)(\mathbf{x}) - \mathcal{D}^2(f)(\mathbf{x}_c))\}$.

- Computation of λ' can be certified (Robust SDP)

Nonlinear Quadratic Templates

- Let $\mathbf{x}_1, \ldots, \mathbf{x}_p \in K$
- Quadratic underestimators of f over K:

$$
\begin{aligned}
f_{\mathbf{x}_c, \lambda'} : K &\longrightarrow \mathbb{R} \\
x &\longmapsto f(\mathbf{x}_c) + \mathcal{D}(f)(\mathbf{x}_c)\,(\mathbf{x} - \mathbf{x}_c) \\
&\quad + \frac{1}{2}(\mathbf{x} - \mathbf{x}_c)^T \mathcal{D}^2(f)(\mathbf{x}_c)(\mathbf{x} - \mathbf{x}_c) \\
&\quad + \frac{1}{2}\lambda' \|\mathbf{x} - \mathbf{x}_c\|_2^2 \ ,
\end{aligned}
$$

with $\lambda' \leqslant \lambda := \min_{\mathbf{x} \in K} \{\lambda_{\min}(\mathcal{D}^2(f)(\mathbf{x}) - \mathcal{D}^2(f)(\mathbf{x}_c))\}$.

- Computation of λ' can be certified (Robust SDP)

Contents

Polynomial Estimators for Semialgebraic Functions

- Inspired from [Lasserre - Thanh 13]
- Let $f_{\text{sa}} \in \mathcal{A}$ defined on a box $K \subset \mathbb{R}^n$
- Let λ_n be the standard Lebesgue measure on $\mathbb{R}^n$ (normalized)
- Best polynomial underestimator $h \in \mathbb{R}_d[\mathbf{x}]$ of f_{sa} for the L_1 norm:

$$(P^{\text{sa}}) \left\{ \begin{array}{ll} \min\limits_{h \in \mathbb{R}_d[\mathbf{x}]} & \int_K (f_{\text{sa}} - h) d\lambda_n \\ \text{s.t.} & f_{\text{sa}} - h \geqslant 0 \text{ on } K \ . \end{array} \right.$$

Lemma

Problem (P^{sa}) has a degree-d polynomial minimizer h_d.

Polynomial Estimators for Semialgebraic Functions

- b.s.a.l. $K_{\text{pop}} := \{(\mathbf{x}, \mathbf{z}) \in \mathbb{R}^{n+p} : g_1(\mathbf{x}, \mathbf{z}) \geqslant 0, \dots, g_m(\mathbf{x}, \mathbf{z}) \geqslant 0\}$
- The quadratic module $M(\mathbf{g})$ is Archimedean
- The optimal solution h_d of (P^{sa}) is a maximizer of:

$$(P_d) \left\{ \begin{array}{ll} \max\limits_{h \in \mathbb{R}_d[\mathbf{x}]} & \displaystyle\int_{[0,1]^n} h \, d\lambda_n \\ \text{s.t.} & (z_p - h) \in M(\mathbf{g}) \end{array} \right. .$$

Polynomial Estimators for Semialgebraic Functions

- Let m_d be the optimal value of Problem (P^{sa})
- Let h_{dk} be a maximizer of the SOS relaxation of (P_d)

Convergence of the SOS Hierarchy

The sequence $(\|f_{\text{sa}} - h_{dk}\|_1)_{k \geqslant k_0}$ is non-increasing and converges to m_d. Each accumulation point of the sequence $(h_{dk})_{k \geqslant k_0}$ is an optimal solution of Problem (P^{sa}).

$$f_{\text{sa}}(\mathbf{x}) := \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}}$$

d	k	Upper bound of $\|f_{\text{sa}} - h_{dk}\|_1$	Bound
2	2	0.8024	-1.171
	3	0.3709	-0.4479
4	2	1.617	-1.056
	3	0.1766	-0.4493

Polynomial Estimators for Semialgebraic Functions

- Let m_d be the optimal value of Problem (P^{sa})
- Let h_{dk} be a maximizer of the SOS relaxation of (P_d)

Convergence of the SOS Hierarchy

The sequence $(\|f_{\text{sa}} - h_{dk}\|_1)_{k \geqslant k_0}$ is non-increasing and converges to m_d. Each accumulation point of the sequence $(h_{dk})_{k \geqslant k_0}$ is an optimal solution of Problem (P^{sa}).

$$f_{\text{sa}}(\mathbf{x}) := \frac{\partial_4 \Delta \mathbf{x}}{\sqrt{4 x_1 \Delta \mathbf{x}}}$$

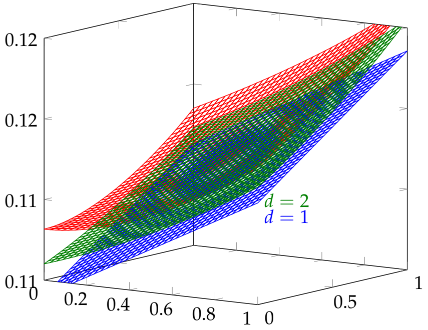
d	k	Upper bound of $\|f_{\text{sa}} - h_{dk}\|_1$	Bound
2	2	0.8024	-1.171
	3	0.3709	-0.4479
4	2	1.617	-1.056
	3	0.1766	-0.4493

Polynomial Estimators for Semialgebraic Functions

- $\text{rad}_2 : (x_1, x_2) \mapsto \dfrac{-64x_1^2 + 128x_1x_2 + 1024x_1 - 64x_2^2 + 1024x_2 - 4096}{-8x_1^2 + 8x_1x_2 + 128x_1 - 8x_2^2 + 128x_2 - 512}$
- Linear and quadratic underestimators for rad_2 ($k = 3$):

Polynomial Estimators for Semialgebraic Functions

- $\text{rad}_2 : (x_1, x_2) \mapsto \dfrac{-64x_1^2 + 128x_1x_2 + 1024x_1 - 64x_2^2 + 1024x_2 - 4096}{-8x_1^2 + 8x_1x_2 + 128x_1 - 8x_2^2 + 128x_2 - 512}$
- Linear and quadratic underestimators for rad_2 ($k = 3$):



Contents

Comparison Results on Global Optimization Problems

$$\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = - \sum_{i=1}^{n} x_i \sin(\sqrt{x_i})$$
$$f^* \lesssim -418.9n$$

Minimax Approximation + SOS

- $d = 4, n = 10$
- 38 *min* to certify a lower bound of $-430n$
- Poor accuracy of Minimax Estimators

Comparison Results on Global Optimization Problems

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

$$f^* \lesssim -418.9n$$

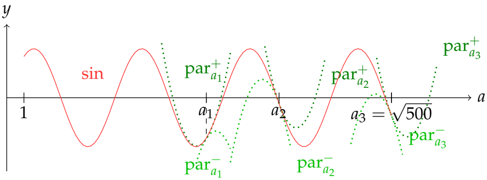
Inteval Arithmetic for sin + SOS

n	lower bound	n_{lifting}	#boxes	time
10	$-430n$	0	3830	$129\,s$
10	$-430n$	$2n$	16	$40\,s$

Comparison Results on Global Optimization Problems

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

$$f^* \lesssim -418.9n$$



n	lower bound	n_{lifting}	#boxes	time
10	$-430n$	0	3830	$129\,s$
10	$-430n$	$2n$	16	$40\,s$

Comparison Results on Global Optimization Problems

$$\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n} x_i \sin(\sqrt{x_i})$$
$$f^* \lesssim -418.9n$$

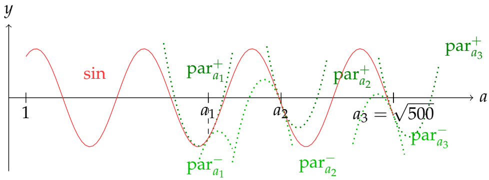
Inteval Arithmetic for sin + SOS

n	lower bound	n_{lifting}	#boxes	time
100	$-440n$	0	> 10000	$> 10\,h$
100	$-440n$	$2n$	274	$1.9\,h$

Comparison Results on Global Optimization Problems

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

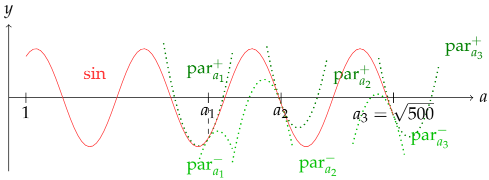
$$f^* \lesssim -418.9n$$



n	lower bound	n_{lifting}	#boxes	time
100	$-440n$	0	> 10000	$> 10\,h$
100	$-440n$	$2n$	274	$1.9\,h$

Comparison Results on Global Optimization Problems

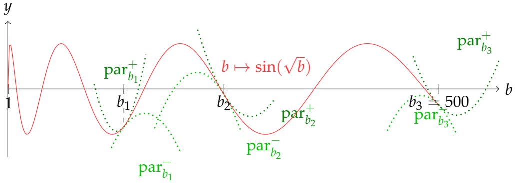
$$\min_{\mathbf{x} \in [1,500]^n} f(\mathbf{x}) = -\sum_{i=1}^{n-1} (x_i + x_{i+1}) \sin(\sqrt{x_i})$$



n	lower bound	n_{lifting}	#boxes	time
1000	$-967n$	$2n$	1	$543\,s$
1000	$-968n$	n	1	$272\,s$

Comparison Results on Global Optimization Problems

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



n	lower bound	n_{lifting}	#boxes	time
1000	$-967n$	$2n$	1	$543\,s$
1000	$-968n$	n	1	$272\,s$

Outline

Hybrid Symbolic-Numeric Certification

- Certified lower bound of $\inf_{\mathbf{x} \in K} p(\mathbf{x})$?
- At relaxation order k, SOS solvers output:
  - floating-point lower bound μ_k
  - floating-point SOS $\sigma_0, \dots, \sigma_m$
- Projection and rounding by [Parrilo-Peyrl 08]:
  - Seek rational SOS $\sigma_0', \dots, \sigma_m'$ so that $p - \mu_k = \sum_{j=0}^{m} \sigma_j'(\mathbf{x}) g_j(\mathbf{x})$
  - Try with a lower bound $\mu_k' \lesssim \mu_k$ when it fails

Hybrid Symbolic-Numeric Certification

- Alternative to the projection and rounding by [Parrilo-Peyrl 08]:
  - Normalized POP ($\mathbf{x} \in [0, 1]^n$)
  - Conversion into rationals: SOS $\tilde{\sigma_0}, \ldots, \tilde{\sigma_m}$, lower bound $\tilde{\mu_k}$
  - $\epsilon_{\text{pop}}(\mathbf{x}) := p(\mathbf{x}) - \tilde{\mu_k} - \sum_{j=0}^{m} \tilde{\sigma_j}(\mathbf{x}) g_j(\mathbf{x})$
  - Bounding: $\forall \mathbf{x} \in [0, 1]^n, \epsilon_{\text{pop}}(\mathbf{x}) \geqslant \epsilon^*_{\text{pop}} := \sum_{\epsilon_{\boldsymbol{\alpha}} \leqslant 0} \epsilon_{\boldsymbol{\alpha}}$
- More concise SOS certificates / Simpler rounding

Customized Polynomial Ring

Check symbolic polynomial equalities $\mathbf{q} = \mathbf{q}'$

- Existing tactic `ring` [Grégoire-Mahboubi 05]
- Polynomials coefficients: arbitrary-size rationals `bigQ` [Grégoire-Théry 06]

Checking Polynomial Equalities

- Sparse Horner normal form

```
Inductive PolC: Type :=
 | Pc   : bigQ → PolC
 | Pinj: positive → PolC → PolC
 | PX   : PolC → positive → PolC → PolC.
```

  - `(Pc c)` for constant polynomials
  - `(Pinj i p)` shifts the index of `i` in the variables of `p`
  - `(PX p j q)` evaluates to $p x_1^j + q(x_2, \ldots, x_n)$
- Encoding SOS certificates with Sparse Horner polynomials

Bounding the Polynomial Remainder

- Normalized POP ($\mathbf{x} \in [0,1]^n$)
- $\epsilon_{\text{pop}}(\mathbf{x}) := p(\mathbf{x}) - \tilde{\mu}_k - \sum_{j=0}^{m} \tilde{\sigma}_j(\mathbf{x}) g_j(\mathbf{x})$
- $\forall \mathbf{x} \in [0,1]^n, \epsilon_{\text{pop}}(\mathbf{x}) \geqslant \epsilon_{\text{pop}}^* := \sum_{\epsilon_\alpha \leqslant 0} \epsilon_\alpha$

```
Fixpoint lower_bnd :=
 match eps_pol with
  | Pc c       ⇒ cmin c zero
  | Pinj _ p ⇒ lower_bnd p
  | PX p _ q ⇒ lower_bnd p
              +! lower_bnd q
 end.
```

```
Lemma remainder_lemma l eps_pol :
(forall i, i \in vars eps_pol → 0 <= l i ∧ l i <= 1)
    → [lower_bnd eps_pol] <= PolCeval l eps_pol.
```

Bounding the Polynomial Remainder

- Normalized POP ($\mathbf{x} \in [0,1]^n$)
- $\epsilon_{\text{pop}}(\mathbf{x}) := p(\mathbf{x}) - \tilde{\mu_k} - \sum_{j=0}^{m} \tilde{\sigma}_j(\mathbf{x}) g_j(\mathbf{x})$
- $\forall \mathbf{x} \in [0,1]^n, \epsilon_{\text{pop}}(\mathbf{x}) \geqslant \epsilon^*_{\text{pop}} := \sum_{\epsilon_\alpha \leqslant 0} \epsilon_\alpha$

```
Fixpoint lower_bnd :=
 match eps_pol with
  | Pc c       ⇒ cmin c zero
  | Pinj _ p ⇒ lower_bnd p
  | PX p _ q ⇒ lower_bnd p
             +! lower_bnd q
 end.
```

```
Lemma remainder_lemma l eps_pol :
(forall i, i \in vars eps_pol → 0 <= l i ∧ l i <= 1)
   → [lower_bnd eps_pol] <= PolCeval l eps_pol.
```

Checking SOS Certificates inside COQ

- $\epsilon_{\text{pop}}(\mathbf{x}) := p(\mathbf{x}) - \tilde{\mu}_k - \sum_{j=0}^{m} \tilde{\sigma}_j(\mathbf{x}) g_j(\mathbf{x})$ (represented by `r`)
- $\forall \mathbf{x} \in [0, 1]^n, p(\mathbf{x}) \geqslant \tilde{\mu}_k + \epsilon^*_{\text{pop}}$
- `obj` represents $p(\mathbf{x}) - \tilde{\mu}_k - \epsilon^*_{\text{pop}}$
- `ineq` indexes $g_1, \ldots, g_m$

```
Lemma Putinar_Psatz_correct l obj r ineq lambda sos :
  (forall i, i \in vars r → 0 <= l i ∧ l i <= 1 ) →
   forall lambda_idx, 0 [<=] lambda lambda_idx →
   forall ineq_idx, 0 <= PolCeval l (ineq ineq_idx) →
   pol_checker obj r ineq lambda sos = true →
     0 <= PEeval l obj.
```

Checking SOS Certificates inside COQ

- $\epsilon_{\text{pop}}(\mathbf{x}) := p(\mathbf{x}) - \tilde{\mu}_k - \sum_{j=0}^{m} \tilde{\sigma}_j(\mathbf{x}) g_j(\mathbf{x})$ (represented by `r`)
- $\forall \mathbf{x} \in [0,1]^n, p(\mathbf{x}) \geqslant \tilde{\mu}_k + \epsilon^*_{\text{pop}}$
- `obj` represents $p(\mathbf{x}) - \tilde{\mu}_k - \epsilon^*_{\text{pop}}$
- `ineq` indexes $g_1, \ldots, g_m$

```
Lemma Putinar_Psatz_correct l obj r ineq lambda sos :
  (forall i, i \in vars r → 0 <= l i ∧ l i <= 1 ) →
   forall lambda_idx, 0 [<=] lambda lambda_idx →
   forall ineq_idx, 0 <= PolCeval l (ineq ineq_idx) →
   pol_checker obj r ineq lambda sos = true →
     0 <= PEeval l obj.
```

Software Package `NLCertify`

- Performs sparse semialgebraic optimization, interface with SDPA
- Nonlinear maxplus dynamic approximation
- Interface with `Sollya` for comparison
- Nonlinear templates approximation
- Informal Certification: more than 12000 lines of code
- Formal Verification of certificates for semialgebraic optimization
- Formal Verification: more than 2500 lines of code

Formal Verification Results

Formal SOS Checker

- *POP1*: $\forall \mathbf{x} \in K, \partial_4 \Delta \mathbf{x} \geqslant -41$.
- *POP2*: $\forall \mathbf{x} \in K, \Delta \mathbf{x} \geqslant 0$.

Problem	n	`NLCertify`	`micromega` [Besson 07]
POP1	6	$0.08\,s$	$9\,s$
POP2	2	$0.09\,s$	$0.36\,s$
	3	$0.39\,s$	—
	6	$13.2\,s$	—

Sparse SOS relaxations + concise rational SOS $\Longrightarrow$ Speedup

Formal Verification Results

Formal Bounds for POP relaxations of Flyspeck Inequalities

Inequality	#boxes	Informal Nonlinear Optimization Time	Formal SOS Checker Time
9922699028	39	190 s	2218 s
3318775219	338	1560 s	19136 s

- Comparable with Taylor interval methods in HOL-LIGHT [Solovyev 13]

Bottleneck of informal optimizer is SOS solver

22 times slower! $\Longrightarrow$ Current bottleneck is to check polynomial equalities

Contributions

- Combining Minimax/Maxplus Templates with Sparse SOS
- Framework for a large class of functions
- Combines precision of SOS with scalibility of Interval calculus
- Templates: limit the blow-up by coarsening estimators
- Software package `NLCertify` to solve hard Global Optimization Problems
- More concise SOS certificates / Simpler rounding: POP checker in COQ has better performance than `micromega`

Contributions

- Combining Minimax/Maxplus Templates with Sparse SOS
- Framework for a large class of functions
- Combines precision of SOS with scalibility of Interval calculus
- Templates: limit the blow-up by coarsening estimators

Software package `NLCertify` to solve hard Global Optimization Problems

More concise SOS certificates / Simpler rounding: POP checker in COQ has better performance than `micromega`

Perspectives

- Problems with transcendental constraints
- Optimal control problems (Continuous / Discrete)
- Backward propagation of templates (analogy with co-state equations)
- Formal bounds for transcendental univariate functions
- Extension to Non-commutative POP
- More efficient POP checker (alternative to sparse Horner)

Perspectives

- Problems with transcendental constraints
- Optimal control problems (Continuous / Discrete)
- Backward propagation of templates (analogy with co-state equations)
- Formal bounds for transcendental univariate functions
- Extension to Non-commutative POP
- More efficient POP checker (alternative to sparse Horner)

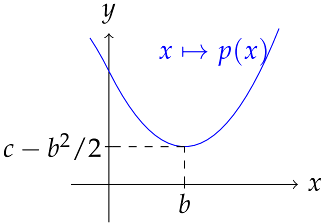
End

Thank you for your attention!

A Simple Polynomial Optimization Problem

POP: $\min_{x \in \mathbb{R}} p(x) = 1/2x^2 - bx + c$

- A program written in OCaml/C provides the sums of squares (SOS) decomposition: $1/2(x - b)^2$
- A program written in Coq checks: $\forall x \in \mathbb{R}, p(x) = 1/2(x - b)^2 + c - b^2/2$



- Sceptical approach: obtain *certificates* of positivity with efficient oracles and check them formally certificates

Rationals in COQ

Fast integers/rationals computation available inside COQ:

- Functional modular arithmetic [Grégoire-Théry 06] (`bigN`)
- Generic implementation of rational numbers [Grégoire-Théry 07] $\Longrightarrow$ build arbitrary-size rationals `bigQ`
- Native arithmetic operations on $\mathbb{Z}/2^{31}\mathbb{Z}$ [Spiwack 06]

Checking Polynomial Equalities

Example with $p(x_1, x_2) := x_1^2 - 2x_1x_2 + x_2^2 = (x_1 - x_2)^2$

- The sparse Horner `x_12` represents $x_1 - x_2$
- SOS certificate $(x_1 - x_2)^2$ encoded by `[([(one, x_12)], xH)]`