Flyspeck Inequalities and Semidefinite Programming

Victor Magron, RA Imperial College

Memory Optimization and Co-Design Meeting
29 June 2015
Mathematicians and Computer Scientists want to eliminate all the uncertainties on their results. Why?
Mathematicians and Computer Scientists 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, ... )
Mathematicians and Computer Scientists 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, …)

Ariane 5 launch failure, Pentium FDIV bug
Errors and Proofs

- Possible workaround: proof assistants
  - COQ (Coquand, Huet 1984)
  - HOL-LIGHT (Harrison, Gordon 1980)
  - Built in top of OCAML
Complex Proofs

- Complex mathematical proofs / mandatory computation

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

From Oranges Stack...

Kepler Conjecture (1611):
The maximal density of sphere packings in 3D-space is \(\frac{\pi}{\sqrt{18}}\)

- Face-centered cubic Packing
- Hexagonal Compact Packing
...to 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]: Formal Proof of Kepler Conjecture

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]: Formal Proof of Kepler Conjecture.

Project Completion on 10 August by the Flyspeck team!!
...to Flyspeck Nonlinear Inequalities

- Nonlinear inequalities: quantified reasoning with “∀”

\[ \forall x \in K, f(x) \geq 0 \]

- NP-hard optimization problem
A “Simple” Example

In the computational part:

- Multivariate Polynomials:

\[ \Delta x := x_1x_4(-x_1 + x_2 + x_3 - x_4 + x_5 + x_6) + x_2x_5(x_1 - x_2 + x_3 + x_4 - x_5 + x_6) + x_3x_6(x_1 + x_2 - x_3 + x_4 + x_5 - x_6) - x_2(x_3x_4 + x_1x_6) - x_5(x_1x_3 + x_4x_6) \]
A “Simple” Example

In the computational part:

- **Semialgebraic** functions: composition of polynomials with $|\cdot|, \sqrt{}, +, -, \times, /, \sup, \inf, \ldots$

$$\begin{align*}
p(x) &:= \partial_4 \Delta x & q(x) &:= 4x_1 \Delta x \\
r(x) &:= p(x) / \sqrt{q(x)}
\end{align*}$$

$$\begin{align*}
l(x) &:= -\frac{\pi}{2} + 1.6294 - 0.2213 (\sqrt{x_2} + \sqrt{x_3} + \sqrt{x_5} + \sqrt{x_6} - 8.0) + 0.913 (\sqrt{x_4} - 2.52) + 0.728 (\sqrt{x_1} - 2.0)
\end{align*}$$
A “Simple” Example

In the computational part:

- **Transcendental** functions $\mathcal{T}$: composition of semialgebraic functions with $\arctan$, $\exp$, $\sin$, $+,-,\times,\ldots$
A “Simple” Example

In the computational part:

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

Lemma$_{9922699028}$ from Flyspeck:

$$\forall x \in K, \arctan\left(\frac{p(x)}{\sqrt{q(x)}}\right) + l(x) \geq 0$$
Existing Formal Frameworks

Formal proofs for Global Optimization:

- Bernstein polynomial methods [Zumkeller’s PhD 08]
- SMT methods [Gao et al. 12]
- Interval analysis and Sums of squares
Existing Formal Frameworks

- Interval analysis
  - Certified interval arithmetic in COQ [Melquiond 12]
  - Taylor methods in HOL Light [Solovyev thesis 13]
    - Formal verification of floating-point operations
  - robust but subject to the **Curse of Dimensionality**
Existing Formal Frameworks

Lemma 9922699028 from Flyspeck:

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

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

Flyspeck Inequalities and Semidefinite Programming
Semidefinite Programming

- Linear Programming (LP):

\[
\min_{\mathbf{z}} \quad \mathbf{c}^\top \mathbf{z} \\
\text{s.t.} \quad \mathbf{A} \mathbf{z} \geq \mathbf{d}.
\]

- Linear cost \( \mathbf{c} \)

- Linear inequalities \( \sum_i A_{ij} z_j \geq d_i \)
Semidefinite Programming (SDP):

\[
\begin{align*}
\text{min} & \quad c^\top z \\
\text{s.t.} & \quad \sum_i F_i z_i \succeq F_0.
\end{align*}
\]

- Linear cost \( c \)
- Symmetric matrices \( F_0, F_i \)
- Linear matrix inequalities "\( F \succeq 0 \)" (\( F \) has nonnegative eigenvalues)
Semidefinite Programming (SDP):

$$\min_{z} \quad c^\top z$$

s.t. \quad \sum_{i} F_i z_i \succeq F_0 , \quad A z = d .$$

- Linear cost $c$
- Symmetric matrices $F_0, F_i$
- Linear matrix inequalities "$F \succeq 0$" (F has nonnegative eigenvalues)

Spectrahedron
Prove polynomial inequalities with SDP:

\[ p(a, b) := a^2 - 2ab + b^2 \geq 0. \]

Find \( z \) s.t.

\[
\begin{pmatrix}
  a \\
  b
\end{pmatrix}
\begin{pmatrix}
  z_1 & z_2 \\
  z_2 & z_3
\end{pmatrix}
\begin{pmatrix}
  a \\
  b
\end{pmatrix} \succeq 0.
\]

Find \( z \) s.t.

\[ a^2 - 2ab + b^2 = z_1 a^2 + 2z_2 ab + z_3 b^2 \quad (A z = d) \]

\[
\begin{pmatrix}
  z_1 & z_2 \\
  z_2 & z_3
\end{pmatrix} = \begin{pmatrix}
  1 & 0 \\
  0 & 0
\end{pmatrix} z_1 + \begin{pmatrix}
  0 & 1 \\
  1 & 0
\end{pmatrix} z_2 + \begin{pmatrix}
  0 & 0 \\
  0 & 1
\end{pmatrix} z_3 \succeq \begin{pmatrix}
  0 & 0 \\
  0 & 0
\end{pmatrix}
\]

\[ F_1 + F_2 + F_3 \succeq F_0 \]
Choose a cost $c$ e.g. $(1, 0, 1)$ and solve:

$$\min_z \quad c^\top z$$

subject to

$$\sum_i F_i z_i \succeq F_0, \quad A z = d.$$ 

Solution

$$\begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix} \succeq 0 \quad \text{(eigenvalues 0 and 1)}$$

$$a^2 - 2ab + b^2 = (a \ b) \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = (a - b)^2.$$ 

Solving SDP $\implies$ Finding SUMS OF SQUARES certificates
Polynomial Optimization

Semidefinite Programming

\[
\begin{pmatrix}
1 & a & b \\
a & 1 & c \\
b & c & 1
\end{pmatrix} \succeq 0
\]

\[\Rightarrow\] control, polynomial optim (Henrion, Lasserre, Parrilo)

\[\Rightarrow\] combinatorial optim. electrical engineering (Laurent, Steurers)
Polynomial Optimization

Semidefinite Programming

\[
\begin{pmatrix}
1 & a & b \\
a & 1 & c \\
b & c & 1
\end{pmatrix} \succeq 0
\]

\(\leadsto\) control, polynomial optim (Henrion, Lasserre, Parrilo)

\(\leadsto\) combinatorial optim. electrical engineering (Laurent, Steurers)

**Theoretical Approach**

\[ p^* := \inf_{\mathbb{R}^n} p(x) \]

\[
\sup \lambda \\
\Leftarrow \text{with} \quad p - \lambda \geq 0
\]

**INFINITE LP**
Polynomial Optimization

Semidefinite Programming

\[
\begin{pmatrix}
1 & a & b \\
a & 1 & c \\
b & c & 1
\end{pmatrix} \succeq 0
\]

\[\Rightarrow\text{control, polynomial optim} \ (\text{Henrion, Lasserre, Parrilo})\]

\[\Rightarrow\text{combinatorial optim. electrical engineering} \ (\text{Laurent, Steurers})\]

Practical Approach

\[p^* := \inf_{\mathbb{R}^n} p(x) \ ?\]

\[\sup \lambda \]

\[\Leftarrow \text{with } p - \lambda = \text{sums of squares of fixed degree}\]
Polynomial Optimization

Semidefinite Programming
\[
\begin{pmatrix}
1 & a & b \\
a & 1 & c \\
b & c & 1
\end{pmatrix} \succeq 0
\]

\(\leadsto\) control, polynomial optim (Henrion, Lasserre, Parrilo)

\(\leadsto\) combinatorial optim. electrical engineering (Laurent, Steurers)

Practical Approach

\[ p^* := \inf_{\mathbb{R}^n} p(x) ? \]

\[ \sup \lambda \]

\[ \Leftarrow \text{ with } p - \lambda = \text{sums of squares of fixed degree} \]

FINITE SDP

SDP bounds Hierarchy \(\uparrow p^*\)

degree \(d\) \(\Rightarrow (n+2d) \binom{n}{n} \) variables SDP

n variables
Polynomial Optimization

Semidefinite Programming

\[
\begin{pmatrix} 1 & a & b \\ a & 1 & c \\ b & c & 1 \end{pmatrix} \succeq 0
\]

\[\mapsto\] control, polynomial optim (Henrion, Lasserre, Parrilo)

\[\mapsto\] combinatorial optim. electrical engineering (Laurent, Steurers)

**Practical Approach**

\[ p^* := \inf_{\mathbb{R}^n} p(x) ? \]

\[
\sup \lambda
\]

\[ \Leftrightarrow \text{with } p - \lambda = \text{sums of squares of fixed degree} \]

**FINITE SDP**

SDP bounds Hierarchy \( \uparrow p^* \)

degree \( d \) \Rightarrow \binom{n+2d}{n} \text{ variables SDP}

\( n \) variables

\[ \text{\textbullet \ Strengthening } p - \lambda = \text{sums of squares} \implies p \geq \lambda \]

\[ 1 + x_1^4 - 2x_1^2x_2 + x_2^4 = 1 + (x_1^2 - x_2^2)^2 \]
Non-polynomial Optimization

**Taylor + Intervals:**

\[ K_0 \Rightarrow K_1 \Rightarrow K_2 \Rightarrow K_3 \Rightarrow K_4 \]

\[ K \mapsto K_0 \rightarrow K_1 \rightarrow K_2 \rightarrow K_3 \rightarrow K_4 \]

\( \Rightarrow \text{scalable} \quad \text{\( \Leftarrow \) coarse} \)

\( \sim \Rightarrow \text{Curse of dimensionality} \)
Non-polynomial Optimization

**TAYLOR + INTERVALS:**

- Scalable
- Coarse

$$\Rightarrow \text{Curse of dimensionality}$$

**TAYLOR + SUMS OF SQUARES:**

- Not scalable
- Precise

$$\Rightarrow \text{No free lunch}$$

$$(n + 2d) \choose n$$
Non-polynomial Optimization

**Taylor + Intervals:**

\[ K = \Rightarrow K_0 \rightarrow K_1 \rightarrow K_2 \rightarrow K_3 \rightarrow K_4 \]

\[ \sim \sim \text{Curse of dimensionality} \]

**Taylor + Sums of Squares:**

\[ \text{high degree } d \Rightarrow \binom{n+2d}{n} \]

\[ \sim \sim \text{No free lunch} \]

**Maxplus + Sums of Squares:**

\[ \oplus \text{ scalable} \quad \ominus \text{precise} \]

Maxplus in control (Akian Gaubert)

\[ \uparrow \]

Templates in static analysis (Manna)

\[ \sim \sim \text{Curse reduction} \]

Maxplus Approximations

Approximate \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) with supremum of quadratic forms.
Non-polynomial Optimization

**Maxplus + Sums of squares:**  

\[ l(x) \arctan r(x) \]

Function from “simple” inequality:

Theorem

The algorithm converges to a global optimum and certifies inequalities.

| Hales: \( \frac{\text{time ratio between formal and numerical certification}}{V. Voevodsky} \) |
| \( n \) Hales \( \lesssim 10 \) (Maxplus + Sums of squares) \( \ll 2000 \) (Taylor + Intervals) |
Non-polynomial Optimization

**Maxplus + Sums of Squares:** ⊕ scalable ⊕ precise

**Verification software NLCertify, 1st iteration:**

\[y = \left\{ \begin{array}{l}
\arctan(m) & m < a_1 \\
\text{par}_{a_1} & m = a_1 \\
\arctan(M) & m > a_1
\end{array} \right.\]

1 control point \(\{a_1\}\)

\[m_1 = -4.7 \times 10^{-3} < 0\]
Non-polynomial Optimization

**Maxplus + Sums of squares:** 

\[ \oplus \text{ scalable} \quad \oplus \text{ precise} \]

**Verification software NLCertify, 2nd iteration:**

\[ l(x) \]
\[ r(x) \]
\[ \arctan \]

\[ \frac{y}{m} \]

2 control points \( \{a_1, a_2\} \)

\[ m_2 = -6.1 \times 10^{-5} < 0 \]
Non-polynomial Optimization

**Maxplus + Sums of Squares:**

[scalable][precise]

Verification software NLCertify, 3rd iteration:

\[ l(x) + \arctan(r(x)) \]

- 3 control points \( \{a_1, a_2, a_3\} \)
- \( m_3 = 4.1 \times 10^{-6} \)
- \( > 0 \)

Theorem

The algorithm converges to a global optimum and certifies inequalities.

**n Hales**: time ratio between formal and numerical certification (V. Voevodsky);

**n Hales** \( \ll 10^{(\text{Maxplus} + \text{Sums of squares})} \ll 2000 \) (Taylor + Intervals)

Victor Magron

Flyspeck Inequalities and Semidefinite Programming
Non-polynomial Optimization

**Maxplus + Sums of Squares:**  

- scalable  
- precise

Verification software NLCertify, 3rd iteration:

![Graphical representation](image)

3 control points \{a_1, a_2, a_3\}

\[ m_3 = 4.1 \times 10^{-6} \]

> 0

**Theorem**

The algorithm **converges** to a global optimum and **certifies** inequalities.

\[ n_{Hales} : \text{time ratio between formal and numerical certification (V. Voevodsky)} \]

\[ \sim n_{Hales} \lesssim 10 \text{ (Maxplus + Sums of squares)} \ll 2000 \text{ (Taylor + Intervals)} \]
Contributions

**CERTIFICATION MAXPLUS–SUMS OF SQUARES: NUMERIC 🐪 OR FORMAL 🐘**

**Journals**


**Conferences**

Allamigeon, Gaubert, Magron & Werner, *Calculemus Conference* 2013

Allamigeon, Gaubert, Magron & Werner, *European Control Conference* 2013

Magron, *ICMS Conference* 2014 + software NLCertify

**A FORMAL PROOF OF KEPLER CONJECTURE**

Hales, Adams, Bauer, Dang, Harrison, Hoang, Kaliszyk, Magron, Mclaughlin, Nguyen, Nguyen, Nipkow, Obua, Pleso, Rute, Solovyev, Ta, Tran, Trieu, Urban, Vu & Zumkeller, *Prepublication, submitted Sigma/Pi Journal* 2015
Thank you for your attention!

cas.ee.ic.ac.uk/people/vmagron