Non-linear Invariants
for Control-Command Systems

Pierre Roux

ONERA, Toulouse, France

September 6th 2019
Control-Command Systems

plant (physical system to control)
Control-Command Systems

plant (physical system to control)

actuators

controller

sensors

$y_c$

$u_c$

command

Image: public domain

Image: Theauthors/CC BY
Control-Command Systems

plant (physical system to control)

controller

double x[3] = {0, 0, 0};
double nx[3];
double in;
while (1) {
in = acquire_input(); // u_c
nx[0] = 0.9979*x[0] - 0.0381*x[1] - 0.0414*x[2] + 0.0237*in;
x[0] = -0.0404*x[0] + 0.9688*x[1] - 0.0179*x[2] + 0.0043*in;
x[2] = 0.0142*x[0] - 0.0197*x[1] + 0.9823*x[2] + 0.0077*in;
x[0] = nx[0]; x[1] = nx[1]; x[2] = nx[2];
send_output(x); // y_c
wait_next_clock_tick(); // a tick every 10 ms
}
Control-Command Systems

Plant (physical system to control)

Controller

double x[3] = {0, 0, 0};
double nx[3];
double in;
while (1) {
    in = acquire_input(); // u_c
    nx[0] = 0.9379*x[0]-0.0381*x[1]-0.0414*x[2]+0.0237*in;
    nx[1] = -0.0404*x[0]+0.968*x[1]-0.0179*x[2]+0.0143*in;
    nx[2] = 0.0142*x[0]-0.0197*x[1]+0.9823*x[2]+0.0077*in;
    x[0] = nx[0]; x[1] = nx[1]; x[2] = nx[2];
    send_output(x); // y_c
    wait_next_clock_tick(); // a tick every 10 ms
}
Control-Command Systems

plant (physical system to control)

controller

double x[3] = {0, 0, 0};
double nx[3];
double in;
while (1) {
in = acquire_input(); // u_c
nx[0] = 0.9379*x[0]-0.0381*x[1]-0.0414*x[2]+0.0237*in;
x[1] = -0.0404*x[0]+0.968*x[1]-0.0179*x[2]+0.0143*in;
x[2] = 0.0142*x[0]-0.0197*x[1]+0.9823*x[2]+0.0077*in;
x[0] = nx[0]; x[1] = nx[1]; x[2] = nx[2];
send_output(x); // y_c
wait_next_clock_tick(); // a tick every 10 ms
}
Quadratic invariants

- *Linear invariants* commonly used in static analysis are not well suited:
  - at best costly;
  - at worst no result.
Quadratic invariants

- *Linear invariants* commonly used in static analysis are not well suited:
  - at best costly;
  - at worst no result.
Quadratic invariants

- *Linear invariants* commonly used in static analysis are not well suited:
  - at best costly;
  - at worst no result.
Quadratic invariants

- *Linear invariants* commonly used in static analysis are not well suited:
  - at best costly;
  - at worst no result.
- Control theorists know for long that *quadratic invariants* are a good fit for linear systems.
Quadratic invariants

- *Linear invariants* commonly used in static analysis are not well suited:
  - at best costly;
  - at worst no result.
Quadratic invariants

- *Linear invariants* commonly used in static analysis are not well suited:
  - at best costly;
  - at worst no result.
Quadratic invariants

- *Linear invariants* commonly used in static analysis are not well suited:
  - at best costly;
  - at worst no result.
- Control theorists know for long that *quadratic invariants* are a good fit for linear systems.
Example

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

Demo

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

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

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

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

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

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

/*@ requires \valid(s) && inv(s) && -1 <= in0 <= 1;
@ ensures inv(s); */
void step(state *s, double in0) {
    double pre_x0 = s->x0, pre_x1 = s->x1, pre_x2 = s->x2;
    s->x0 = 0.9379 * pre_x0 - 0.0381 * pre_x1 - 0.0414 * pre_x2 + 0.023
    s->x1 = -0.0404 * pre_x0 + 0.968 * pre_x1 - 0.0179 * pre_x2 + 0.014
    s->x2 = 0.0142 * pre_x0 - 0.0197 * pre_x1 + 0.9823 * pre_x2 + 0.007
}
```

```bash
(pierre@machine ~/slides)
% frama-c -wp -wp-model real -wp-prover why3ide intro.c
```
Example (Demo)
Synthesizing Polynomial Invariants

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Example (Demo)
Synthesizing Polynomial Invariants

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Synthesizing Polynomial Invariants

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Polynomial Encoding

Consider the program

```plaintext
x = x0;
while (1) {
    in = input();  /* ∈ [-1, 1] */
    x = f(x, in);
}
```

When a polynomial \( p \) satisfies

\[
\begin{align*}
p(x_0) &\geq 0 & \text{initial condition} \\
p \circ f - p - \sigma (1 - in^2) &\geq 0 & \text{inductiveness} \\
\sigma &\geq 0 & (p(x) \geq 0 \implies p(f(x)) \geq 0)
\end{align*}
\]

Then \( p \geq 0 \) is an invariant.

Need to solve polynomial positivity problems.
Sum of Squares (SOS) Polynomials

Definition (SOS Polynomial)

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

$$p = \sum_{i} q_i^2.$$

- If $p$ is SOS then $p \geq 0$
Sum of Squares (SOS) Polynomials

Definition (SOS Polynomial)

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

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

- If $p$ SOS then $p \geq 0$
- $p$ SOS iff there exist $z := \begin{bmatrix} 1, x_1, x_2, x_1 x_2, \ldots, x_n^d \end{bmatrix}$ and $Q \succeq 0$

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

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

---

$^1 Q \succeq 0$ means $Q$ positive semidefinite: $\forall x, x^T Q x \geq 0$
Example

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

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

that is

\[
p(x, y) = q_{11}x^4 + 2q_{13}x^3 y + 2q_{23}xy^3 + (2q_{12} + q_{33})x^2 y^2 + q_{22}y^4
\]
SOS: Example

Example

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

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

that is

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

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

For instance

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

hence \( p(x, y) = \frac{1}{2} \left( 2x^2 - 3y^2 + xy \right)^2 + \frac{1}{2} \left( y^2 + 3xy \right)^2. \)
Synthesizing Polynomial Invariants

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Polynomial Invariants

In a very nice SAS'15 paper, authors offer for

\[(x_1, x_2) \in [0.9, 1.1] \times [0, 0.2]\]

while (1) {
    pre_x1 = x1; pre_x2 = x2;
    if (x1^2 + x2^2 <= 1) {
        x1 = pre_x1^2 + pre_x2^2;
        x2 = pre_x1^3 + pre_x2^2;
    } else {
        x1 = 0.5 * pre_x1^3 + 0.4 * pre_x2^2;
        x2 = -0.6 * pre_x1^2 + 0.3 * pre_x2^2;
    }
}

the inductive invariant

\[2.510902467 + 0.0050x_1 + 0.0148x_2 - 3.0998x_1^2 + 0.8037x_2^2 + 3.0297x_1^3 - 2.5924x_2^2 - 1.5266x_1x_2 + 1.9133x_1^2x_2 + 1.8122x_1x_2^2 - 1.6042x_1^4 - 0.0512x_1^3x_2 + 4.4430x_1^2x_2^2 + 1.8926x_1^3x_2^2 - 0.5464x_2^4 + 0.2084x_1^5 - 0.5866x_1^4x_2 - 2.2410x_1^3x_2^2 - 1.5714x_1^2x_2^3 + 0.0890x_1x_2^4 + 0.9656x_2^5 - 0.0098x_1^6 + 0.0320x_1^5x_2 + 0.0232x_1^4x_2^2 - 0.2660x_1^3x_2^3 - 0.7746x_1^2x_2^4 - 0.9200x_1x_2^5 - 0.6411x_2^6 \geq 0.\]
Should we trust such results?

- Some are correct (we’ll prove it formally).
- Others aren’t (previous degree 6 polynomial)
Polynomial Invariants

In a very nice SAS'15 paper, authors offer for

\[(x_1, x_2) \in [0.9, 1.1] \times [0, 0.2]\]

while (1) {
    pre_x1 = x1; pre_x2 = x2;
    if (x1^2 + x2^2 <= 1) {
        x1 = pre_x1^2 + pre_x2^3;
        x2 = pre_x1^3 + pre_x2^2;
    } else {
        x1 = 0.5 * pre_x1^3 + 0.4 * pre_x2^2;
        x2 = -0.6 * pre_x1^2 + 0.3 * pre_x2^2;
    }
}

the inductive invariant $2.510902467 + 0.0050x_1 + 0.0148x_2 - 3.0998x_1^2 + 0.8037x_2^3 + 3.0297x_1^3 - 2.5924x_2^2 - 1.5266x_1x_2 + 1.9133x_1^2x_2 + 1.8122x_1x_2^2 - 1.6042x_1^4 - 0.0512x_1^3x_2 + 4.4430x_1^2x_2^2 + 1.8926x_1x_2^3 - 0.5464x_2^4 + 0.2084x_1^5 - 0.5866x_2^4x_2 - 2.2410x_1^3x_2^2 - 1.5714x_1^2x_2^3 + 0.0890x_1x_2^4 + 0.9656x_2^5 - 0.0098x_1^6 + 0.0320x_1^5x_2 + 0.0232x_1^4x_2^2 - 0.2660x_1^3x_2^3 - 0.7746x_1^2x_2^4 - 0.9200x_1x_2^5 - 0.6411x_2^6 \geq 0$. 

12 / 35
Should we trust such results?

- Some are correct (we’ll prove it formally).
- Others aren’t (previous degree 6 polynomial)
SDP solvers yield approximate solutions

- Linear programming
  - simplex: exact solution
  - interior-point: approximate solution
SDP solvers yield approximate solutions

- Linear programming
  
  simplex: exact solution  
  interior-point: approximate solution

- Semidefinite programming
  
  no simplex equivalent  
  interior-point: approximate solution
SDP solvers yield approximate solutions

- Linear programming
  - simplex: exact solution
  - interior-point: approximate solution

- Semidefinite programming
  - no simplex equivalent
  - interior-point: approximate solution

⇒ incompleteness, soundness requires care
Results from SDP solvers will only satisfy equality constraints up to some $\epsilon$

$$p = z^T Q z + z^T E z, \quad |E_{i,j}| \leq \epsilon.$$
SOS: Using approximate SDP solvers

Results from SDP solvers will only satisfy equality constraints up to some $\epsilon$

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

Two validation methods in the literature:

- Round $Q$ to an exact solution $\tilde{Q}$ s.t. $p = z^T \tilde{Q} z$
  - rounding is heuristic
  - check done with rational arithmetic (expensive)
- Check that for any $|E_{i,j}| \leq \epsilon$, $Q + E \succeq 0$
  - entirely with floating-point arithmetic (more tricky but fast)
Intuitively, Proving Existence of a Nearby Solution

\[ \{ X \mid X \geq 0 \} \]

equality constraints
SOS: Using approximate SDP solvers

Results from SDP solvers will only satisfy equality constraints up to some $\epsilon$

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

Two validation methods in the literature

- Round $Q$ to an exact solution $\tilde{Q}$ s.t. $p = z^T \tilde{Q} z$ and check $\tilde{Q} \succeq 0$
  - rounding is heuristic
  - check done with rational arithmetic (expensive)
- Check that for any $|E_{i,j}| \leq \epsilon$, $Q + E \succeq 0$
  - entirely with floating-point arithmetic (more tricky but fast)
Intuitively, Proving Existence of a Nearby Solution

\{ X \mid X \geq 0 \}

equality constraints
Intuitively, Proving Existence of a Nearby Solution

\[ \{ X \mid X \geq 0 \} \]

\[ \{ Q + E \} \]

\[ p \text{ SOS} \]

equality constraints
Intuitively, Proving Existence of a Nearby Solution

$\{ X \mid X \geq 0 \}$

cannot conclude

equality constraints

$+Q$

$\{ Q + E \}$
Intuitively, Proving Existence of a Nearby Solution

\[ \{ X \mid X \geq 0 \} \]

\[ \{ Q + E \} \]

cannot conclude

equality constraints
Intuitively, Proving Existence of a Nearby Solution

\[ \{ X \mid X \geq 0 \} \]

\[ +Q \]

\[ \{ Q + E \} \]

$\rho$ SOS

equality constraints
Intuitively, Proving Existence of a Nearby Solution

\[ \{ X \mid X \geq 0 \} \]

\[ +Q \]

\[ \{ Q + E \} \]

cannot conclude

equality constraints
Padding

\[ \{ X \mid X \geq 0 \} \]

\[ \{ X \mid X - s \epsilon I \geq 0 \} \]

\{ Q + E \}

equality constraints
Intuitively, Proving Existence of a Nearby Solution

\[ \{ X \mid X \geq 0 \} \]

+Q

\{ Q + E \}

cannot conclude
equality constraints
Intuitively, Proving Existence of a Nearby Solution

\[ \{ X \mid X \geq 0 \} \]

\[ \{ Q + E \} \]

\[ +Q \]

cannot conclude

equality constraints
Intuitively, Proving Existence of a Nearby Solution

\{X \mid X \geq 0\}

\{Q + E\}

cannot conclude

equality constraints
Padding

\{X \mid X \geq 0\} \quad \{X \mid X - s \epsilon l \geq 0\}

\{Q + E\}

equality constraints
Synthesizing Polynomial Invariants

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Floating-Point Values

Definition

A floating-point format $\mathbb{F}$ is a subset of $\mathbb{R}$. $x \in \mathbb{F}$ when

$$x = m/\beta^e$$

for some $m, e \in \mathbb{Z}$, $|m| < \beta^p$ and $e \geq e_{\text{min}}$. 
# Floating-Point Values

## Definition

A floating-point format $F$ is a subset of $\mathbb{R}$. $x \in F$ when

$$x = m/\beta^e$$

for some $m, e \in \mathbb{Z}$, $|m| < \beta^p$ and $e \geq e_{\text{min}}$.

- $m$: mantissa of $x$
- $e$: exponent of $x$
- $\beta$: radix of $F$
- $p$: precision of $F$
- $e_{\text{min}}$: minimal exponent of $F$
## Floating-Point Values

### Definition

A floating-point format $F$ is a subset of $\mathbb{R}$. $x \in F$ when

$$x = m \beta^e$$

for some $m, e \in \mathbb{Z}$, $|m| < \beta^p$ and $e \geq e_{\text{min}}$.

- $m$: mantissa of $x$
- $e$: exponent of $x$
- $\beta$: radix of $F$
- $p$: precision of $F$
- $e_{\text{min}}$: minimal exponent of $F$

### Two kind of numbers

- **Normalized**: encoded with $p$ figures ($|m| \geq \beta^{p-1}$)
- **Denormalized**: tiny values ($e = e_{\text{min}}$, $|m| < \beta^{p-1}$)
Standard Model of Floating-Point Arithmetic

Definition

$\text{fl}(e)$: floating-point evaluation of expression $e$ (from left to right).

For $\diamond \in \{+, -, \sqrt{\cdot}\}$:

$$\exists \delta, |\delta| \leq \frac{\varepsilon}{1 + \varepsilon} \land \text{fl}(x \diamond y) = (1 + \delta)(x \diamond y).$$
Standard Model of Floating-Point Arithmetic

Definition

$\text{fl}(e)$: floating-point evaluation of expression $e$ (from left to right).

For $\odot \in \{+, -, \sqrt{\cdot}\}$:

$$\exists \delta, |\delta| \leq \frac{\varepsilon}{1 + \varepsilon} \land \text{fl}(x \odot y) = (1 + \delta)(x \odot y).$$

For $\odot \in \{\times, \div\}$:

$$\exists \delta, \omega, |\delta| \leq \frac{\varepsilon}{1 + \varepsilon} \land |\omega| \leq \eta \land \text{fl}(x \odot y) = (1 + \delta)(x \odot y) + \omega.$$
Standard Model of Floating-Point Arithmetic

**Definition**

$\text{fl}(e)$: floating-point evaluation of expression $e$ (from left to right).

For $\diamond \in \{+, -, \sqrt{\cdot}\}$:

$$\exists \delta, |\delta| \leq \frac{\varepsilon}{1 + \varepsilon} \land \text{fl}(x \diamond y) = (1 + \delta)(x \diamond y).$$

For $\diamond \in \{\times, \div\}$:

$$\exists \delta, \omega, |\delta| \leq \frac{\varepsilon}{1 + \varepsilon} \land |\omega| \leq \eta \land \text{fl}(x \diamond y) = (1 + \delta)(x \diamond y) + \omega.$$

**Example**

$\varepsilon = 2^{-53} \approx 10^{-16}$ and $\eta = 2^{-1075} \approx 10^{-323}$

for binary64 format (double in C) and rounding to nearest.
Example: Summation

Bounds can be combined:

**Theorem**

For all \( x \in \mathbb{R}^n \)

\[
\left| \text{fl} \left( \sum_{i=1}^{n} x_i \right) - \sum_{i=1}^{n} x_i \right| \leq n \varepsilon \sum_{i=1}^{n} |x_i| + (1 + n \varepsilon)n \eta
\]

Proved in Coq (https://github.com/validsdp/validsdp/).

Floating-Point arithmetic model from the Flocq library (http://flocq.gforge.inria.fr/).
Cholesky Decomposition

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

- To prove that a matrix $A \in \mathbb{R}^{n \times n}$ is positive semidefinite we can similarly expose $R$ such that $A = R^T R$ (since $x^T (R^T R) x = (Rx)^T (Rx) = \|Rx\|_2^2 \geq 0$).

- The Cholesky decomposition computes such a matrix $R$:

  $R := 0$;
  
  for $j$ from 1 to $n$ do
    for $i$ from 1 to $j - 1$ do
      
      $R_{i,j} := \left( A_{i,j} - \sum_{k=1}^{i-1} R_{k,i} R_{k,j} \right) / R_{i,i}$;
    
    od
  
  $R_{j,j} := \sqrt{M_{j,j} - \sum_{k=1}^{j-1} R_{k,j}^2}$;
  
  od

- If it succeeds (no $\sqrt{}$ of negative or div. by 0) then $A \succeq 0$. 


Cholesky Decomposition (end)

With rounding errors $A \neq R^T R$, Cholesky can succeed while $A \succeq 0$.

But error is bounded and for some (tiny) $c \in \mathbb{R}$: if Cholesky succeeds on $A$ then $A + cI \succeq 0$.

Hence:

**Theorem**

If Cholesky succeeds on $A - cI$ then $A \succeq 0$

holds for any $c \geq \frac{(s + 1)\varepsilon}{1 - (s + 1)\varepsilon} \text{tr}(A) + 4s \left(2(s + 1) + \max_i (A_{i,i})\right) \eta$

Proved in Coq (paper proof: 6 pages, Coq: 5.1 kloc)
Synthesizing Polynomial Invariants

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Cholesky Decomposition (end)

With rounding errors $A \neq R^T R$, Cholesky can succeed while $A \succeq 0$.

But error is bounded and for some (tiny) $c \in \mathbb{R}$: if Cholesky succeeds on $A$ then $A + cI \succeq 0$.

Hence:

**Theorem**

If Cholesky succeeds on $A - cI$ then $A \succeq 0$

holds for any $c \geq \frac{(s+1)\varepsilon}{1 - (s+1)\varepsilon} \text{tr}(A) + 4s \left(2(s+1) + \max_{i}(A_{i,i})\right) \eta$

Proved in Coq (paper proof: 6 pages, Coq: 5.1 kloc)
Intuitively, Proving Existence of a Nearby Solution

\{ X \mid X \geq 0 \}

\{ Q + E \}

\pmb{p} \text{ SOS}

equality constraints
Cholesky Decomposition (end)

With rounding errors $A \neq R^T R$, Cholesky can succeed while $A \succeq 0$.

But error is bounded and for some (tiny) $c \in \mathbb{R}$: if Cholesky succeeds on $A$ then $A + c I \succeq 0$.

Hence:

**Theorem**

If Cholesky succeeds on $A - c I$ then $A \succeq 0$

holds for any $c \geq \frac{(s + 1)\varepsilon}{1 - (s + 1)\varepsilon} \operatorname{tr}(A) + 4s \left(2(s + 1) + \max_i (A_{i,i})\right) \eta$

Proved in Coq (paper proof: 6 pages, Coq: 5.1 kloc)
Synthesizing Polynomial Invariants

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Integration in Alt-Ergo

Joint work with Mohamed Iguernlala and Sylvain Conchon

- Integrated into Alt-Ergo 2

(1) AC(LA) framework

- AC-Completion
- linear equalities
- Union-Find Modulo Theories

- SAT-Solver
  - \( s = t \)

- Bounds inference
  - \( s \leq t \)

- Fourier-Motzkin
- Map from terms to Intervals
  - encoding \( \text{unsat/unknown} \)

(2) Interval Calculus

- Unfortunately no tight collaboration:
  - one shot, no incrementality
  - mostly a boolean result

- available at https://github.com/OCamlPro/alt-ergo/pull/124
## Experimental Results (1/3)

**Benchmarks QF_NIA from SMT-LIB.**

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

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

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB.
### Experimental Results (2/3)

Benchmarks QF_NRA from SMT-LIB.

<table>
<thead>
<tr>
<th></th>
<th>AE</th>
<th>AESDP</th>
<th>AESDPap</th>
<th>AESDPex</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>Sturm-MBO (300)</td>
<td>155</td>
<td>12950</td>
<td>155</td>
<td>13075</td>
</tr>
<tr>
<td>hong (20)</td>
<td>1</td>
<td>0</td>
<td>20</td>
<td>28</td>
</tr>
<tr>
<td>hycomp (2494)</td>
<td>1285</td>
<td>15351</td>
<td>1266</td>
<td>15857</td>
</tr>
<tr>
<td>keymaera (320)</td>
<td>261</td>
<td>36</td>
<td>291</td>
<td>356</td>
</tr>
<tr>
<td>LassoRanker (627)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>meti-tarski (2615)</td>
<td>1882</td>
<td>10</td>
<td>2273</td>
<td>91</td>
</tr>
<tr>
<td>UltimateAutom (13)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>zankl (85)</td>
<td>14</td>
<td>1.00</td>
<td>24</td>
<td>15.46</td>
</tr>
<tr>
<td>total (6549)</td>
<td>3571</td>
<td>28348</td>
<td>4029</td>
<td>29423</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CVC4</th>
<th>Smtrat</th>
<th>Yices2</th>
<th>Z3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>Sturm-MBO (300)</td>
<td>285</td>
<td>1403</td>
<td>285</td>
<td>620</td>
</tr>
<tr>
<td>hong (20)</td>
<td>20</td>
<td>1</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>hycomp (2494)</td>
<td>2184</td>
<td>208</td>
<td>1588</td>
<td>13784</td>
</tr>
<tr>
<td>keymaera (320)</td>
<td>249</td>
<td>4</td>
<td>307</td>
<td>13</td>
</tr>
<tr>
<td>LassoRanker (627)</td>
<td>441</td>
<td>32786</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>meti-tarski (2615)</td>
<td>1643</td>
<td>804</td>
<td>2520</td>
<td>3345</td>
</tr>
<tr>
<td>UltimateAutom (13)</td>
<td>5</td>
<td>0.52</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>zankl (85)</td>
<td>24</td>
<td>9.40</td>
<td>19</td>
<td>13.47</td>
</tr>
<tr>
<td>total (6549)</td>
<td>4853</td>
<td>35239</td>
<td>4740</td>
<td>17775</td>
</tr>
</tbody>
</table>

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB.
## Experimental Results (1/3)

Benchmarks QF_NIA from SMT-LIB.

<table>
<thead>
<tr>
<th></th>
<th>AE</th>
<th>AESDP</th>
<th>AESDPap</th>
<th>AESDPex</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>AProVE (746)</td>
<td>103</td>
<td>7387</td>
<td>319</td>
<td>23968</td>
</tr>
<tr>
<td>calypto (97)</td>
<td>92</td>
<td>357</td>
<td>88</td>
<td>679</td>
</tr>
<tr>
<td>LassoRanker (102)</td>
<td>57</td>
<td>9</td>
<td>62</td>
<td>959</td>
</tr>
<tr>
<td>LCTES (2)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>leipzig (5)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mcm (161)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UltimateAutom (7)</td>
<td>1</td>
<td>0.35</td>
<td>7</td>
<td>0.73</td>
</tr>
<tr>
<td>UltimateLasso (26)</td>
<td>26</td>
<td>118</td>
<td>26</td>
<td>212</td>
</tr>
<tr>
<td>total (1146)</td>
<td>279</td>
<td>7872</td>
<td>502</td>
<td>25818</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CVC4</th>
<th>Smtrat</th>
<th>Yices2</th>
<th>Z3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>AProVE (746)</td>
<td>586</td>
<td>10821</td>
<td>185</td>
<td>3879</td>
</tr>
<tr>
<td>calypto (97)</td>
<td>87</td>
<td>7</td>
<td>89</td>
<td>754</td>
</tr>
<tr>
<td>LassoRanker (102)</td>
<td>72</td>
<td>27</td>
<td>20</td>
<td>12</td>
</tr>
<tr>
<td>LCTES (2)</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>leipzig (5)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mcm (161)</td>
<td>4</td>
<td>2489</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UltimateAutom (7)</td>
<td>6</td>
<td>0.03</td>
<td>1</td>
<td>7.22</td>
</tr>
<tr>
<td>UltimateLasso (26)</td>
<td>4</td>
<td>66</td>
<td>26</td>
<td>177</td>
</tr>
<tr>
<td>total (1146)</td>
<td>780</td>
<td>13411</td>
<td>321</td>
<td>4829</td>
</tr>
</tbody>
</table>

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB.
Experimental Results (3/3)

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

<table>
<thead>
<tr>
<th></th>
<th>AE</th>
<th>AESDP</th>
<th>AESDPap</th>
<th>AESDPex</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>C (67)</td>
<td>11</td>
<td>0.05</td>
<td>63</td>
<td>39.78</td>
</tr>
<tr>
<td>quadratic (67)</td>
<td>13</td>
<td>0.06</td>
<td>67</td>
<td>14.68</td>
</tr>
<tr>
<td>flyspeck (20)</td>
<td>1</td>
<td>0.00</td>
<td>19</td>
<td>26.35</td>
</tr>
<tr>
<td>global-opt (14)</td>
<td>2</td>
<td>0.01</td>
<td>14</td>
<td>8.72</td>
</tr>
<tr>
<td>total (168)</td>
<td>27</td>
<td>0.12</td>
<td>163</td>
<td>89.53</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CVC4</th>
<th>Smtrat</th>
<th>Yices2</th>
<th>Z3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>C (67)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>quadratic (67)</td>
<td>14</td>
<td>2.46</td>
<td>18</td>
<td>1.26</td>
</tr>
<tr>
<td>flyspeck (20)</td>
<td>6</td>
<td>695.59</td>
<td>9</td>
<td>36.54</td>
</tr>
<tr>
<td>global-opt (14)</td>
<td>5</td>
<td>0.12</td>
<td>12</td>
<td>41.18</td>
</tr>
<tr>
<td>total (168)</td>
<td>25</td>
<td>698.17</td>
<td>39</td>
<td>78.98</td>
</tr>
</tbody>
</table>

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB. All times are in seconds.
Experimental Results (2/3)

Benchmarks QF_NRA from SMT-LIB.

<table>
<thead>
<tr>
<th></th>
<th>AE</th>
<th>AESDP</th>
<th>AESDPap</th>
<th>AESDPex</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>Sturm-MBO (300)</td>
<td>155</td>
<td>12950</td>
<td>155</td>
<td>13075</td>
</tr>
<tr>
<td>hong (20)</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>28</td>
</tr>
<tr>
<td>hycomp (2494)</td>
<td>1285</td>
<td>15351</td>
<td>1266</td>
<td>15857</td>
</tr>
<tr>
<td>keymaera (320)</td>
<td>261</td>
<td>36</td>
<td>291</td>
<td>356</td>
</tr>
<tr>
<td>LassoRanker (627)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>meti-tarski (2615)</td>
<td>1882</td>
<td>10</td>
<td>2273</td>
<td>91</td>
</tr>
<tr>
<td>UltimateAutom (13)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>zankl (85)</td>
<td>14</td>
<td>1.00</td>
<td>24</td>
<td>15.46</td>
</tr>
<tr>
<td>total (6549)</td>
<td>3571</td>
<td>28348</td>
<td>4029</td>
<td>29423</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CVC4</th>
<th>Smtrat</th>
<th>Yices2</th>
<th>Z3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>Sturm-MBO (300)</td>
<td>285</td>
<td>1403</td>
<td>285</td>
<td>620</td>
</tr>
<tr>
<td>hong (20)</td>
<td>20</td>
<td>1</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>hycomp (2494)</td>
<td>2184</td>
<td>208</td>
<td>1588</td>
<td>13784</td>
</tr>
<tr>
<td>keymaera (320)</td>
<td>249</td>
<td>4</td>
<td>307</td>
<td>13</td>
</tr>
<tr>
<td>LassoRanker (627)</td>
<td>441</td>
<td>32786</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>meti-tarski (2615)</td>
<td>1643</td>
<td>804</td>
<td>2520</td>
<td>3345</td>
</tr>
<tr>
<td>UltimateAutom (13)</td>
<td>5</td>
<td>0.52</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>zankl (85)</td>
<td>24</td>
<td>9.40</td>
<td>19</td>
<td>13.47</td>
</tr>
<tr>
<td>total (6549)</td>
<td>4853</td>
<td>35239</td>
<td>4740</td>
<td>17775</td>
</tr>
</tbody>
</table>

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

\[ \{ X \mid X \geq 0 \} \quad \{ X \mid X - s \epsilon I \geq 0 \} \]

\[ +Q \quad \{ Q + E \} \]

equality constraints
Intuitively, Proving Existence of a Nearby Solution

\{ X \mid X \geq 0 \}

\{ Q + E \}

+Q
cannot conclude
equality constraints
## Experimental Results (1/3)

Benchmarks QF_NIA from SMT-LIB.

<table>
<thead>
<tr>
<th></th>
<th>AE</th>
<th>AESDP</th>
<th>AESDPap</th>
<th>AESDPex</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>AProVE (746)</td>
<td>103</td>
<td>7387</td>
<td>319</td>
<td>23968</td>
</tr>
<tr>
<td>calypto (97)</td>
<td>92</td>
<td>357</td>
<td>88</td>
<td>679</td>
</tr>
<tr>
<td>LassoRanker (102)</td>
<td>57</td>
<td>9</td>
<td>62</td>
<td>959</td>
</tr>
<tr>
<td>LCTES (2)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>leipzig (5)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mcm (161)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UltimateAutom (7)</td>
<td>1</td>
<td>0.35</td>
<td>7</td>
<td>0.73</td>
</tr>
<tr>
<td>UltimateLasso (26)</td>
<td>26</td>
<td>118</td>
<td>26</td>
<td>212</td>
</tr>
<tr>
<td>total (1146)</td>
<td>279</td>
<td>7872</td>
<td>502</td>
<td>25818</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CVC4</th>
<th>Smtstat</th>
<th>Yices2</th>
<th>Z3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>AProVE (746)</td>
<td>586</td>
<td>10821</td>
<td>185</td>
<td>3879</td>
</tr>
<tr>
<td>calypto (97)</td>
<td>87</td>
<td>7</td>
<td>89</td>
<td>754</td>
</tr>
<tr>
<td>LassoRanker (102)</td>
<td>72</td>
<td>27</td>
<td>20</td>
<td>12</td>
</tr>
<tr>
<td>LCTES (2)</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>leipzig (5)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mcm (161)</td>
<td>4</td>
<td>2489</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UltimateAutom (7)</td>
<td>6</td>
<td>0.03</td>
<td>1</td>
<td>7.22</td>
</tr>
<tr>
<td>UltimateLasso (26)</td>
<td>4</td>
<td>66</td>
<td>26</td>
<td>177</td>
</tr>
<tr>
<td>total (1146)</td>
<td>780</td>
<td>13411</td>
<td>321</td>
<td>4829</td>
</tr>
</tbody>
</table>

On Intel Xeon 2.3 GHz, time limits 900 s and memory limits 2 GB.
Experimental Results (3/3)

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

<table>
<thead>
<tr>
<th></th>
<th>AE</th>
<th>AESDP</th>
<th>AESDPap</th>
<th>AESDPex</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>C (67)</td>
<td>11</td>
<td>0.05</td>
<td>63</td>
<td>39.78</td>
</tr>
<tr>
<td>quadratic (67)</td>
<td>13</td>
<td>0.06</td>
<td>67</td>
<td>14.68</td>
</tr>
<tr>
<td>flyspeck (20)</td>
<td>1</td>
<td>0.00</td>
<td>19</td>
<td>26.35</td>
</tr>
<tr>
<td>global-opt (14)</td>
<td>2</td>
<td>0.01</td>
<td>14</td>
<td>8.72</td>
</tr>
<tr>
<td>total (168)</td>
<td>27</td>
<td>0.12</td>
<td>163</td>
<td>89.53</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CVC4</th>
<th>Smtrat</th>
<th>Yices2</th>
<th>Z3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>unsat</td>
<td>time</td>
<td>unsat</td>
<td>time</td>
</tr>
<tr>
<td>C (67)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>quadratic (67)</td>
<td>14</td>
<td>2.46</td>
<td>18</td>
<td>1.26</td>
</tr>
<tr>
<td>flyspeck (20)</td>
<td>6</td>
<td>695.59</td>
<td>9</td>
<td>36.54</td>
</tr>
<tr>
<td>global-opt (14)</td>
<td>5</td>
<td>0.12</td>
<td>12</td>
<td>41.18</td>
</tr>
<tr>
<td>total (168)</td>
<td>25</td>
<td>698.17</td>
<td>39</td>
<td>78.98</td>
</tr>
</tbody>
</table>

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

Ensuring Soundness

Bound Floating-Point Rounding Errors

Integration into a SMT Solver

Formalized Proofs with Coq
Coq Implementation

Joint work Érik Martin-Dorel

- Available at https://sourcesup.renater.fr/validsdp/
- LGPL license
- uses libraries
  - CoqEAL [Cano, Cohen, Dénès, Mörtberg, Rouhling, Siles] for refinement proofs
    (based on SSReflect and MathComp [Gonthier et al.])
  - SSrMultinomials [Strub] for multivariate polynomials
  - CoqInterval [Melquiond] and Flocq [Boldo, Melquiond] for floating-point numbers
- 15 kloc of Coq + 0.3 kloc of OCaml code
The validsdp tactic – the big picture

Joint work Érik Martin-Dorel

**Goal**

\[ x_i : \mathbb{R} \vdash 0 \leq r \]

Ltac

reification (Ltac)

\[ (x, p) : \text{list}(\mathbb{R}) \times \text{AST} \]

transform to effective datatypes

**convertibility rule**

\[ 0 \leq \text{interp}(x, p) \]

correctness theorem

\[ \text{check}(x, p, (z, Q)) = \text{true} \]

**soswitness (OCaml)**

\[ P : \text{list}(\text{list}(\mathbb{N}) \times \mathbb{Q}) \]

computation

\[ (z, Q) : \text{list}(\text{list}(\mathbb{N})) \times \text{list}(\text{list}(\mathbb{R})) \]
## Benchmarks (1/2)

<table>
<thead>
<tr>
<th>Problem</th>
<th>n</th>
<th>d</th>
<th>OSDP</th>
<th>Monniaux C11</th>
<th>QEPCAD</th>
<th>ValidSOP</th>
<th>PVS/Bernstein</th>
<th>NLCertify</th>
<th>HOL Light/Taylor</th>
</tr>
</thead>
<tbody>
<tr>
<td>adaptativeLV</td>
<td>4</td>
<td>4</td>
<td>0.75</td>
<td>2.67</td>
<td>1.12</td>
<td>3.97</td>
<td>5.16</td>
<td>14.93</td>
<td>2.61</td>
</tr>
<tr>
<td>butcher</td>
<td>6</td>
<td>4</td>
<td>1.58</td>
<td>—</td>
<td>1.05</td>
<td>—</td>
<td>9.40</td>
<td>48.44</td>
<td>8.36</td>
</tr>
<tr>
<td>caprasse</td>
<td>4</td>
<td>4</td>
<td>0.41</td>
<td>1.82</td>
<td>0.88</td>
<td>5.74</td>
<td>5.19</td>
<td>25.89</td>
<td>2.63</td>
</tr>
<tr>
<td>heart</td>
<td>8</td>
<td>4</td>
<td>3.18</td>
<td>268.75</td>
<td>—</td>
<td>—</td>
<td>16.67</td>
<td>131.13</td>
<td>—</td>
</tr>
<tr>
<td>magnetism</td>
<td>7</td>
<td>2</td>
<td>1.11</td>
<td>2.04</td>
<td>1.64</td>
<td>4.61</td>
<td>5.18</td>
<td>245.52</td>
<td>14.50</td>
</tr>
<tr>
<td>reaction</td>
<td>3</td>
<td>2</td>
<td>0.81</td>
<td>1.56</td>
<td>0.24</td>
<td>4.38</td>
<td>4.33</td>
<td>11.48</td>
<td>1.96</td>
</tr>
<tr>
<td>schuwefel</td>
<td>3</td>
<td>4</td>
<td>0.95</td>
<td>2.45</td>
<td>2.76</td>
<td>4.17</td>
<td>3.70</td>
<td>14.72</td>
<td>56.13</td>
</tr>
<tr>
<td>fs260</td>
<td>6</td>
<td>4</td>
<td>1.25</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>5.99</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>fs461</td>
<td>6</td>
<td>4</td>
<td>0.70</td>
<td>11.18</td>
<td>0.87</td>
<td>—</td>
<td>5.18</td>
<td>621.06</td>
<td>7.46</td>
</tr>
<tr>
<td>fs491</td>
<td>6</td>
<td>4</td>
<td>0.54</td>
<td>21.81</td>
<td>—</td>
<td>—</td>
<td>5.38</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>fs745</td>
<td>6</td>
<td>4</td>
<td>0.98</td>
<td>11.74</td>
<td>0.94</td>
<td>—</td>
<td>5.55</td>
<td>623.17</td>
<td>6.90</td>
</tr>
<tr>
<td>fs752</td>
<td>6</td>
<td>2</td>
<td>0.35</td>
<td>1.81</td>
<td>0.90</td>
<td>—</td>
<td>3.80</td>
<td>54.52</td>
<td>7.88</td>
</tr>
<tr>
<td>fs8</td>
<td>6</td>
<td>2</td>
<td>0.43</td>
<td>1.53</td>
<td>1.48</td>
<td>—</td>
<td>3.93</td>
<td>52.63</td>
<td>6.62</td>
</tr>
<tr>
<td>fs859</td>
<td>6</td>
<td>8</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>fs860</td>
<td>6</td>
<td>4</td>
<td>1.21</td>
<td>10.53</td>
<td>1.11</td>
<td>—</td>
<td>6.08</td>
<td>73.65</td>
<td>7.34</td>
</tr>
<tr>
<td>fs861</td>
<td>6</td>
<td>4</td>
<td>1.09</td>
<td>10.48</td>
<td>1.20</td>
<td>—</td>
<td>5.15</td>
<td>69.74</td>
<td>7.87</td>
</tr>
<tr>
<td>fs862</td>
<td>6</td>
<td>4</td>
<td>1.27</td>
<td>79.25</td>
<td>1.25</td>
<td>—</td>
<td>5.37</td>
<td>73.54</td>
<td>7.58</td>
</tr>
<tr>
<td>fs863</td>
<td>6</td>
<td>2</td>
<td>0.94</td>
<td>1.50</td>
<td>—</td>
<td>—</td>
<td>3.85</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>fs864</td>
<td>6</td>
<td>2</td>
<td>0.56</td>
<td>2.05</td>
<td>—</td>
<td>—</td>
<td>4.05</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>fs865</td>
<td>6</td>
<td>2</td>
<td>0.76</td>
<td>2.11</td>
<td>—</td>
<td>—</td>
<td>3.68</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>fs867</td>
<td>6</td>
<td>2</td>
<td>0.21</td>
<td>2.09</td>
<td>1.74</td>
<td>—</td>
<td>4.22</td>
<td>—</td>
<td>8.04</td>
</tr>
</tbody>
</table>

On Intel Core i5 2.9 GHz, time limits 900 s. All times in seconds.
## Benchmarks (2/2)

<table>
<thead>
<tr>
<th>Problem</th>
<th>n</th>
<th>d</th>
<th>OSDP (not verified)</th>
<th>MonniauxC11 (not verified)</th>
<th>NL Certify (not verified)</th>
<th>QEPCAD (not verified)</th>
<th>ValidSDP</th>
<th>PVS/Bernstein</th>
<th>NL Certify</th>
<th>HOL Light/Taylor</th>
</tr>
</thead>
<tbody>
<tr>
<td>fs868</td>
<td>6</td>
<td>4</td>
<td>0.94</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fs884</td>
<td>6</td>
<td>4</td>
<td></td>
<td></td>
<td>7.78</td>
<td></td>
<td>6.05</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fs890</td>
<td>6</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex4_d4</td>
<td>2</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex4_d6</td>
<td>2</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex4_d8</td>
<td>2</td>
<td>24</td>
<td>16.99</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex4_d10</td>
<td>2</td>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex5_d4</td>
<td>3</td>
<td>8</td>
<td>1.67</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex5_d6</td>
<td>3</td>
<td>12</td>
<td>16.10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex5_d8</td>
<td>3</td>
<td>16</td>
<td>203.06</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex5_d10</td>
<td>3</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex6_d4</td>
<td>4</td>
<td>8</td>
<td>16.82</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex6_d6</td>
<td>4</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex7_d4</td>
<td>2</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex7_d6</td>
<td>2</td>
<td>18</td>
<td>1.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex7_d8</td>
<td>2</td>
<td>24</td>
<td>15.38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex7_d10</td>
<td>2</td>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex8_d4</td>
<td>2</td>
<td>8</td>
<td>0.87</td>
<td>15.72</td>
<td></td>
<td></td>
<td>73.75</td>
<td></td>
<td>7.52</td>
<td></td>
</tr>
<tr>
<td>ex8_d6</td>
<td>2</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex8_d8</td>
<td>2</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ex8_d10</td>
<td>2</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

On Intel Core i5 2.9 GHz, time limits 900 s. All times in seconds.
Coq offers efficient machine integers

Enables effective floating-point computation by emulating floats with integers

But slow ($\times 1000$ compared to OCaml)
The validsdp tactic – the big picture

Joint work Érik Martin-Dorel

Goal

\( x_i : \mathbb{R} \vdash 0 \leq r \)

Ltac

\( (x, p) : \text{list}(\mathbb{R}) \times \text{AST} \)

reification (Ltac)

\( P : \text{list}(\text{list}(\mathbb{N}) \times \mathbb{Q}) \)

transform to effective datatypes

\( (z, Q) : \text{list}(\text{list}(\mathbb{N})) \times \text{list}(\text{list}(\mathbb{R})) \)

soswitness (OCaml)

convertibility rule

\( 0 \leq \text{interp}(x, p) \)

correctness theorem

\( \text{check}(x, p, (z, Q)) = \text{true} \)

computation
Primitive Floats in Coq

Joint work Guillaume Bertholon and Érik Martin-Dorel

- Coq offers efficient machine integers
- Enables effective floating-point computation by emulating floats with integers
- But slow (x1000 compared to OCaml)
Primitive Floats in Coq

Joint work Guillaume Bertholon and Érik Martin-Dorel

- Coq offers efficient machine integers
- Enables effective floating-point computation by emulating floats with integers
- But slow \((x1000\) compared to OCaml\)
- Add sound access to machine floating-point in Coq
- https://github.com/coq/coq/pull/9867
- Presentation at ITP next week