Reasoning with Nonlinear Formulas in Isabelle/HOL

Wenda Li

University of Cambridge

wl302@cam.ac.uk

January 7, 2020

Joint work with Grant Passmore and Larry Paulson

Alfred Tarski (1930s): the first-order theory of real closed fields is complete and decidable.

That is, we have a decision procedure for closed sentences like the following:

$$\exists x \in \mathbb{R}.\, \forall y \in \mathbb{R}.\, \exists z \in \mathbb{R}.\, xz - y^2 < 0 \land x > 0.$$

The Sturm-Tarski theorem (also known as Tarski's theorem)

Given $P, Q \in \mathbb{R}[X]$,$P \neq 0$, $a, b \in \overline{\mathbb{R}}$, $a < b$ and are not roots of P,

$$\mathrm{TaQ}(Q, P, a, b) = \mathrm{Var}(\mathrm{SRemS}(P, P'Q); a, b),$$

where

- $$\mathrm{TaQ}(Q, P, a, b) = \sum_{x \in (a,b), P(x)=0} \mathrm{sgn}(Q(x)),$$
- P' is the first derivative of P,
- Var computes the sign variations,
- SRemS computes the signed remainder sequence.

Also, $\mathrm{TaQ}(1, P, a, b)$ computes the number of real roots of P within the interval (a, b) (i.e., Sturm's theorem).

To decide $\exists x \in \mathbb{R}.\, P(x) = 0 \wedge Q_1(x) > 0$

Let

$$
\begin{aligned}
& c(Q_1 \bowtie_1 0, \cdots, Q_n \bowtie_n 0) \\
& \qquad = \mathrm{card}(\{x \mid P(x) = 0 \wedge Q_1(x) \bowtie_1 0 \wedge \cdots \wedge Q_n(x) \bowtie_n 0\})
\end{aligned}
$$

, where $\bowtie_i \in \{<, >, =\}$, and $\mathrm{TaQ}_P(Q_i) = \mathrm{TaQ}(Q_i, P, -\infty, +\infty)$.
We have

$$\exists x \in \mathbb{R}.\, P(x) = 0 \wedge Q_1(x) > 0 \Longleftrightarrow c(Q_1 > 0) > 0,$$

while $c(Q_1 > 0)$ can be found by solving the following linear equation:

$$
\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & -1 \\ 0 & 1 & 1 \end{bmatrix}
\begin{bmatrix} c(Q_1 = 0) \\ c(Q_1 > 0) \\ c(Q_1 < 0) \end{bmatrix}
=
\begin{bmatrix} \mathrm{TaQ}_P(1) \\ \mathrm{TaQ}_P(Q_1) \\ \mathrm{TaQ}_P(Q_1^2) \end{bmatrix}.
$$

The number of linear equations grows very quickly.

$$\exists x \in \mathbb{R}.\, P(x) = 0 \wedge Q_1(x) > 0 \wedge Q_2(x) < 0 \iff c(Q_1 > 0, Q_2 < 0) > 0,$$

requires us to solve a system with 9 equations:

$$\left(\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & -1 \\ 0 & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & -1 \\ 0 & 1 & 1 \end{bmatrix}\right) \begin{bmatrix} c(Q_1 = 0, Q_2 = 0) \\ c(Q_1 = 0, Q_2 > 0) \\ \cdots \\ c(Q_1 > 0, Q_2 < 0) \\ \cdots \\ c(Q_1 < 0, Q_2 < 0) \end{bmatrix} = \begin{bmatrix} \mathrm{TaQ}_P(1) \\ \mathrm{TaQ}_P(Q_2) \\ \cdots \\ \mathrm{TaQ}_P(Q_1 Q_2^2) \\ \cdots \\ \mathrm{TaQ}_P(Q_1^2 Q_2^2) \end{bmatrix}$$

where $\otimes$ is tensor product.

Tarski's elimination procedure is mostly of theoretical interest

Univariate case: exponential in the number of polynomials

General case: non-elementary in the number of variables

Due to its elegance, Tarski's elimination procedure has been implemented in Coq[1], HOL Light[2] and PVS[3].

---

[1] Mahboubi and Cohen, "Formal proofs in real algebraic geometry: from ordered fields to quantifier elimination".

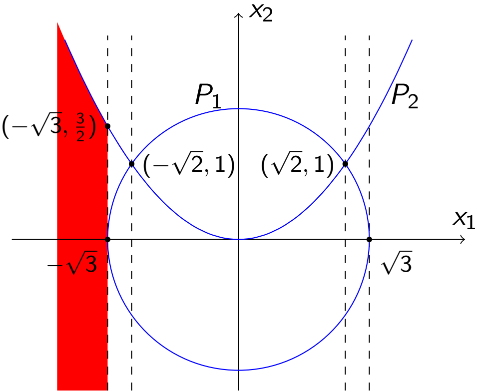
[2] Nieuwenhuis, *CADE-20: 20th International Conference on Automated Deduction, proceedings*.

[3] Narkawicz, Muñoz, and Dutle, "Formally-Verified Decision Procedures for Univariate Polynomial Computation Based on Sturm's and Tarski's Theorems."

Can we have a more practical procedure?

George E. Collins (1976): Yes, here is cylindrical algebraic decomposition (CAD).

What is cylindrical algebraic decomposition (CAD)



$D_{1,1} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 < x_1^2/2\}$

$D_{1,2} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 = x_1^2/2\}$

$D_{1,3} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 > x_1^2/2\}$

$D_{2,1} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 < 0\}$

$D_{2,2} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 = 0\}$

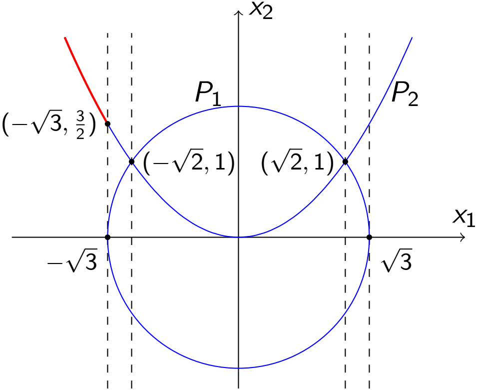
$\vdots$

$D_{9,2} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 = x_1^2/2\}$

$D_{9,3} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 > x_1^2/2\}$

Here, $P_1(x_1, x_2) = x_2^2 + x_1^2 - 3$ and $P_2(x_1, x_2) = x_2 - x_1^2/2$.

What is cylindrical algebraic decomposition (CAD)



$D_{1,1} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 < x_1^2/2\}$

$D_{1,2} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 = x_1^2/2\}$

$D_{1,3} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 > x_1^2/2\}$

$D_{2,1} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 < 0\}$

$D_{2,2} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 = 0\}$

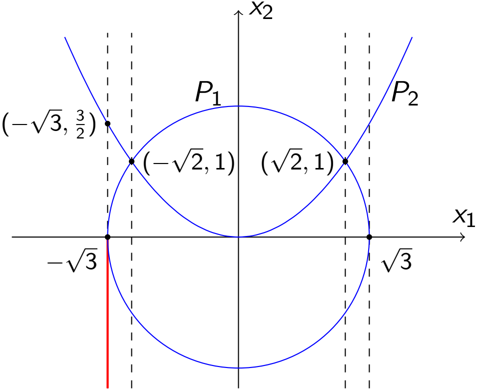
$\vdots$

$D_{9,2} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 = x_1^2/2\}$

$D_{9,3} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 > x_1^2/2\}$

Here, $P_1(x_1, x_2) = x_2^2 + x_1^2 - 3$ and $P_2(x_1, x_2) = x_2 - x_1^2/2$.

What is cylindrical algebraic decomposition (CAD)



$D_{1,1} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 < x_1^2/2\}$

$D_{1,2} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 = x_1^2/2\}$

$D_{1,3} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 > x_1^2/2\}$

$D_{2,1} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 < 0\}$

$D_{2,2} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 = 0\}$

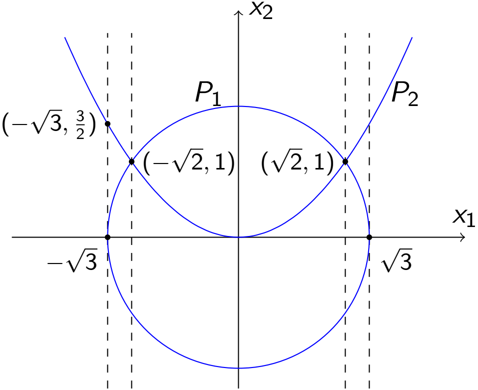
$\vdots$

$D_{9,2} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 = x_1^2/2\}$

$D_{9,3} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 > x_1^2/2\}$

Here, $P_1(x_1, x_2) = x_2^2 + x_1^2 - 3$ and $P_2(x_1, x_2) = x_2 - x_1^2/2$.

What is cylindrical algebraic decomposition (CAD)



$D_{1,1} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 < x_1^2/2\}$

$D_{1,2} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 = x_1^2/2\}$

$D_{1,3} = \{(x_1, x_2) \mid x_1 < -\sqrt{3} \wedge x_2 > x_1^2/2\}$

$D_{2,1} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 < 0\}$

$D_{2,2} = \{(x_1, x_2) \mid x_1 = -\sqrt{3} \wedge x_2 = 0\}$

$\vdots$

$D_{9,2} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 = x_1^2/2\}$

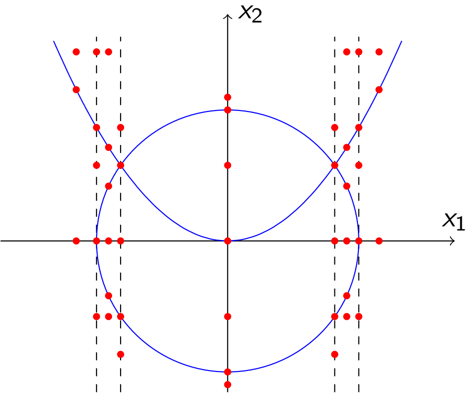
$D_{9,3} = \{(x_1, x_2) \mid x_1 > \sqrt{3} \wedge x_2 > x_1^2/2\}$

such that

$$\bigcup \mathfrak{D} = \mathbb{R}^2$$

$$\forall X \in \mathfrak{D}. \forall Y \in \mathfrak{D}. X \neq Y \to X \cap Y = \emptyset$$

and both $P_1(x_1, x_2) = x_2^2 + x_1^2 - 3$ and $P_2(x_1, x_2) = x_2 - x_1^2/2$ have constant sign over every $X \in \mathfrak{D}$ (or $\{P_1, P_2\}$ is adapted to $\mathfrak{D}$)



$(-2, 0) \in D_{1,1}$

$(-2, 2) \in D_{1,2}$

$(-2, 2.5) \in D_{1,3}$

$(-\sqrt{3}, -1) \in D_{2,1}$

$(-\sqrt{3}, 0) \in D_{2,2}$

$\vdots$

$(2, 2) \in D_{9,2}$

$(2, 2.5) \in D_{9,3}$

Let

$$\mathcal{S} = \{(-2, 0), (-2, 2), (-2, 2.5), \cdots, (2, 2), (2, 2.5)\}.$$

Sentences like the following can be decided:

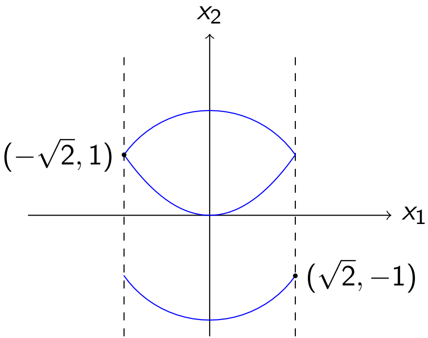
$$\forall x_1 x_2.\, P_1(x_1, x_2) = 0 \wedge P_2(x_1, x_2) > 0$$
$$\iff \forall (x_1, x_2) \in \mathcal{S}.\, P_1(x_1, x_2) = 0 \wedge P_2(x_1, x_2) > 0$$

Definition (Stack)

A stack $\mathfrak{D} = \{D_1, D_2, \ldots, D_{2k+1}\}$ over a connected $S \subseteq \mathbb{R}^n$ is a decomposition of the cylinder $S \times \mathbb{R}$ such that

- there is a sequence of continuous functions $f_0, f_1, \ldots, f_{k+1} : S \longrightarrow \mathbb{R}$, such that $f_0(x) < f_1(x) < \cdots < f_{k+1}(x)$ for all $x \in S$, $f_0(x) = -\infty$, $f_{k+1}(x) = +\infty$,
- $D_{2i+1} = \{(x, x') \in S \times \mathbb{R} \mid f_i(x) < x' < f_{i+1}(x)\}$, for $i = 0, 1, \ldots, k$,
- $D_{2i} = \{(x, x') \in S \times \mathbb{R} \mid x' = f_i(x)\}$, for $i = 1, 2, \ldots, k$.

Example of a stack



Let

$$\begin{aligned} S &=]-\sqrt{2}, \sqrt{2}[, \\ f_1(x) &= -\sqrt{3 - x^2}, \\ f_2(x) &= x^2/2, \\ f_3(x) &= \sqrt{3 - x^2}. \end{aligned}$$

A stack decomposes $S \times \mathbb{R}$:

$$\begin{aligned} D_1 &= \{(x_1, x_2) \mid x_1 \in S \wedge x_2 < f_1(x_1)\} \\ D_2 &= \{(x_1, x_2) \mid x_1 \in S \wedge x_2 = f_1(x_1)\} \\ D_3 &= \{(x_1, x_2) \mid x_1 \in S \\ &\qquad\qquad \wedge f_1(x_1) < x_2 < f_2(x_2)\} \\ &\vdots \\ D_7 &= \{(x_1, x_2) \mid x_1 \in S \wedge x_2 > f_3(x)\} \end{aligned}$$

Definition (Cylindrical)

A decomposition $\mathfrak{D}$ of $\mathbb{R}^n$ is cylindrical if

- $n = 1$, $\mathfrak{D}$ decomposes $\mathbb{R}$: there exist a finite number of points $a_i \in \mathbb{R}$ for $1 \leq i \leq k$, such that $a_i < a_{i+1}$ $(1 \leq i \leq k-1)$ and

  $$\mathfrak{D} = \{(-\infty, a_1), \{a_1\}, (a_1, a_2), \{a_2\}, \ldots, (a_{k-1}, a_k), \{a_k\}, (a_k, \infty)\}.$$

- $n > 1$, there exists a cylindrical decomposition $\mathfrak{D}'$ of $\mathbb{R}^{n-1}$ such that over each $X \in \mathfrak{D}'$ there is a stack $t(X)$ and

  $$\mathfrak{D} = \bigcup_{X \in \mathfrak{D}'} t(X).$$

Theorem (Delineability)

Let $\mathcal{P} \subseteq \mathbb{R}[x_1, \ldots, x_{n-1}][x_n]$ be a set of polynomials and C be a connected subset of $\mathbb{R}^{n-1}$. If

1. *for every $P \in \mathcal{P}$, the total number of complex roots (counting multiplicities) of $P(\beta, x)$ is constant as β varies over C, where $P(\beta, x)$ is a univariate polynomial in which the variables $x_1, \ldots, x_{n-1}$ are instantiated by $\beta \in \mathbb{R}^{n-1}$,*
2. *for every $P \in \mathcal{P}$, the number of distinct complex roots of $P(\beta, x)$ is constant as β varies over C,*
3. *for every $P, Q \in \mathcal{P}$, the total number of common complex roots (counting multiplicities) of $P(\beta, x)$ and $Q(\beta, x)$ is constant as β varies over C,*

then the total number of distinct real roots of $(\prod \mathcal{P})(\beta, x)$ is constant as β varies over C.

Require: a finite set of polynomials $\mathcal{P} \subseteq \mathbb{R}[x_1, \ldots, x_n]$
Ensure: Return a set of sample points $\mathcal{S}_n \subseteq \mathbb{R}^n$ from a CAD that is adapted to $\mathcal{P}$

procedure $\mathrm{CAD}(\mathcal{P})$
    $\mathcal{P}_n \leftarrow \mathcal{P}$
    for $i = n$ to 2 **do** ▷ Projection phase, where $\mathcal{P}_i \subseteq \mathbb{R}[x_1, \ldots, x_i]$
        $\mathcal{P}_{i-1} \leftarrow \mathrm{proj}(\mathcal{P}_i)$
    end for
    $\mathcal{S}_1 \leftarrow \mathrm{base}(\mathcal{P}_1)$ ▷ Base case, where $\mathrm{base}(\mathcal{Q})$ returns a set of sample points adapted to $\mathcal{Q} \subseteq \mathbb{R}[x]$
    for $i = 1$ to $n-1$ **do** ▷ Lifting phase, where $\mathcal{S}_i \subseteq \mathbb{R}^i$
        $\mathcal{S}_{i+1} \leftarrow \bigcup_{\beta \in \mathcal{S}_i} (\{\beta\} \times \mathrm{base}(\mathcal{P}_{i+1}(\beta, x)))$
    end for
    return $\mathcal{S}_n$
end procedure

Given $\mathcal{P} = \{x_2^2 + x_1^2 - 3, x_2 - x_1^2/2\}$,

$$\mathrm{proj}(\mathcal{P}) = \{x_1^4/4 + x_1^2 - 3, 4x_1^2 - 12, 2, 1\}$$

$$\mathcal{S}_1 = \mathrm{base}(\mathrm{proj}(\mathcal{P})) = \{-2, -\sqrt{3}, -\frac{3}{2}, -\sqrt{2}, 0, \sqrt{2}, \frac{3}{2}, \sqrt{3}, 2\}.$$

We start to lift:

$$\begin{array}{lcl}
\mathcal{P}(-2, x_2) & = & \{x_2^2 + 1, x_2 - 2\} \\
\mathrm{base}(\mathcal{P}(-2, x_2)) & = & \{0, 2, 2.5\} \\
\{-2\} \times \mathrm{base}(\mathcal{P}(-2, x_2)) & = & \{(-2, 0), (-2, 2), (-2, 2.5)\} \\
 & \vdots & \\
2 \times \mathrm{base}(\mathcal{P}(2, x_2)) & = & \{(2, 0), (2, 2), (2, 2.5)\}
\end{array}$$

combining which yields

$$\begin{aligned}
\mathcal{S}_2 &= \bigcup_{\beta \in \mathcal{S}_1} (\{\beta\} \times \mathrm{base}(\mathcal{P}(\beta, x))) \\
&= \{(-2, 0), (-2, 2), (-2, 2.5), (-\sqrt{3}, -1), \cdots, (2, 2.5)\},
\end{aligned}$$

Real algebraic numbers

To encode an real algebraic number α, we can use a polynomial $P \in \mathbb{Q}[x]$ and two rational numbers $a, b \in \mathbb{Q}$:

$$\alpha = (P, a, b),$$

so that α is the only root of P within the interval (a, b). For example, $\sqrt{2} = (x^2 - 2, 0, 2)$.

They are closed under normal arithmetic:
$(x^2 - 2, 0, 2) + (x^2 - 3, 0, 2) = (x^4 - 10x^2 + 1, 1, 4)$
$(x^2 - 2, 0, 2) \times (x^2 - 3, 0, 2) = (x^2 - 6, 1, 4)$

Exact algebraic arithmetic is too slow

Exact algebraic arithmetic has been implemented in Isabelle/HOL[4][5] and Coq[6].

The implementation by Joosten, Thiemann and Yamada is arguably the most efficient one (with 70K LOC in Isabelle/HOL), and it takes 20s to compute $\sum_{i=1}^{6} \sqrt[3]{i}$.

Even Mathematica[7] fails to give an awnser to $\sum_{i=1}^{7} \sqrt[3]{i}$ within 30m.

---

[4] Joosten, Thiemann, and Yamada, "A Verified Implementation of Algebraic Numbers in Isabelle/HOL".

[5] Li and Paulson, "A modular, efficient formalisation of real algebraic numbers".

[6] Cohen, "Construction of Real Algebraic Numbers in Coq."

[7] RootReduce[Sum[Surd[i, 3], i, 1, 7]] on Mathematica 12

Sign determination using only rational arithmetic

The Sturm-Tarski theorem provides a way to effectively compute the sign of a univariate polynomial at a real algebraic point:

$$\begin{aligned}\operatorname{sgn}(Q(\alpha)) = \sum_{x \in (a,b), P(x)=0} \operatorname{sgn}(Q(x)) \\ = \operatorname{TaQ}(Q, P, a, b) \\ = \operatorname{Var}(\operatorname{SRemS}(P, P'Q); a, b).\end{aligned}$$

where $P, Q \in \mathbb{Q}[x]$ and $\alpha = (P, a, b)$. For example,

value *"sgn_at [:-1,1:] (Alg [:-2,0,1:] 1 2)"*

which stands for the sign of $(x-1)[x \to \sqrt{2}]$ and returns *1*.

To prove $\forall x.\ x^2 - 2 > 0 \vee x < 2$,

Let $\mathcal{Q} = \{x^2 - 2, x - 2\}$, with a root isolation procedure we can find all real roots of $\mathcal{Q}$: $\{-\sqrt{2}, \sqrt{2}, 2\}$, and construct sample points:

$$\{-2, -\sqrt{2}, 0, \sqrt{2}, \frac{3}{2}, 2, 3\}.$$

Do we want to isolate (find) roots within Isabelle?

Nah, it is easier to check a root than finding it.

To prove $\forall x.\ x^2 - 2 > 0 \vee x < 2$,

$$\forall x. Q_1(x) > 0 \vee Q_2(x) < 0$$

$$\Longleftarrow \{\text{Pick } \{-\sqrt{2}, \sqrt{2}, 2\} \text{ from an unstructed computer algebra system}\}$$

$$\{-\sqrt{2}, \sqrt{2}, 2\} \text{ are all the roots of } Q_1 \text{ and } Q_2$$

$$\wedge\, \forall x \in \{-2, -\sqrt{2}, 0, \sqrt{2}, \frac{3}{2}, 2, 3\}. Q_1(x) > 0 \vee Q_2(x) < 0$$

$$\Longleftrightarrow \sum_{\alpha \in \{-\sqrt{2}, \sqrt{2}, 2\}} \sum_{Q \in \{Q_1, Q_2\}} \mathrm{sgn}(Q(\alpha)) = \mathrm{TaQ}(1, Q_1, -\infty, \infty) + \mathrm{TaQ}(1, Q_2, -\infty, \infty)$$

$$\wedge\, \forall x \in \{-2, -\sqrt{2}, 0, \sqrt{2}, \frac{3}{2}, 2, 3\}. Q_1(x) > 0 \vee Q_2(x) < 0$$

To prove $\forall x.\ x^2 - 2 > 0 \vee x < 2$,

```
lemma "(∀x::real. x*x - 2>0 ∨ x < 2)"
  by (all_tac "[Arep [:-2,0,1:] -2 0,
    Arep [:-2,0,1:] 1 1.5,Rat 2]")
```

Here, $[-\sqrt{2}, \sqrt{2}, 2]$ (encoded as *[Arep [:-2,0,1:] -2 0, Arep [:-2,0,1:] 1 1.5, Rat 2]*) has been found automatically by external solvers.

To prove $\exists x.\ x^2 = 2 \wedge x^3 > 2.5$

Proving the existential case is even easier as we only need one witness:

```
lemma "∃x::real. x*x = 2 ∧ x*x*x>2.5"
  by (ex_tac "[Arep [:-2,0,1:] 1 2]")
```

Promising results[8] compared to Tarski's elimination procedure

	Time (s)		
Formula	univ_rcf (Isabelle)	univ_rcf_cert (Isabelle)	tarski (PVS)
ex1	0.9	0.3	2.0
ex2	1.4	0.6	6.8
ex3	1.6	0.7	13.0
ex4	1.3	0.5	20.1
ex5	1.6	0.6	315.7
ex6	5.6	3.9	timeout
ex7	38.4	34.9	timeout

Note: timeout indicates failure to terminate within 24 hours

[8] Li, Passmore, and Paulson, "Deciding Univariate Polynomial Problems Using Untrusted Certificates in Isabelle/HOL".

Towards multivariate CAD: multivariate sign determination

Previous univariate sign determination procedure require arithmetic in $\mathbb{Q}(\alpha)$ (e.g. $\alpha = \sqrt{2}$).

We convert arithmetic in $\mathbb{Q}(\alpha)$ to polynomial arithmetic in $\mathbb{Q}[\alpha]$ where α is a symbolic indeterminate with some constraint (e.g. $\alpha^2 = 2$).

To eliminate arithmetic in $\mathbb{Q}(\alpha)$ when calculating TaQ:

$$\frac{\deg_x(Q) = \deg(Q[y \to \alpha]) \qquad \deg_x(P) = \deg(P[y \to \alpha])}{P[y \to \alpha] \underbrace{\mathrm{pmod}}_{\text{arithmetic in } \mathbb{Q}(\alpha)} Q[y \to \alpha] = (P \underbrace{\mathrm{pmod}}_{\text{arithmetic in } \mathbb{Q}} Q)[y \to \alpha]}$$

where pmod is pseudo-division of polynomials.

Bivariate sign determination procedure using only rational arithmetic

We finish the bivariate sign determination procedure:

```
value "bsgn_at [:[:0,-1:],[:1:]:] (Alg [:-2,0,1:] 1 2)
                                   (Alg [:-3,0,1:] 1 2)"
```

which stands for the sign of $(x - y)[x \to \sqrt{2}, y \to \sqrt{3}]$ and returns *-1*.

Root isolation with real algebraic coefficient?

$$
\begin{array}{c}
\mathcal{P} \subseteq \mathbb{Q}[x_1, \cdots, x_n] \\
\Big\downarrow \text{Projection} \\
\mathcal{P}_1, \cdots, \mathcal{P}_n \text{ such that } \mathcal{P}_k \subseteq \mathbb{Q}[x_1, \cdots, x_k] \\
\Big\downarrow \text{Base \& Lifting} \\
\mathcal{S}_1, \cdots, \mathcal{S}_n \text{ such that } \mathcal{S}_k \subseteq \tilde{\mathbb{Q}}^k
\end{array}
$$

Here, $\tilde{\mathbb{Q}}$ is the real closure of $\mathbb{Q}$.
In the base and lifting phase, we may need to root-isolate polynomials like $\sqrt{2}x^2 - 3x + 1$.

Again, we want to use certificates

There are efficient algorithms to isolate roots with algebraic coefficients[9][10][11], but none of them is easy to implement (and certify) in a proof assistant.

With a multivariate sign determination procedure, we can efficiently check that $\mathcal{S}_1, \cdots, \mathcal{S}_n$ are indeed sample points drawn from cells described by $\mathcal{P}_1, \cdots, \mathcal{P}_n$.

---

[9] Moura and Passmore, "Computation in Real Closed Infinitesimal and Transcendental Extensions of the Rationals."

[10] Strzeboński, "Cylindrical Algebraic Decomposition using validated numerics".

[11] Boulier et al., "Real root isolation of regular chains".

Towards certifying the projection

```
theorem bivariate_CAD_delineability:
  fixes p q :: "real bpoly" and S::"real set"
  defines "pc≡λy. map_poly complex_of_real (poly_y p y)"
  defines "qc≡λy. map_poly complex_of_real (poly_y q y)"
  assumes
    "connected S" and
    deg_p_inv:"(λy. degree (poly_y p y)) constant_on S" and
    pzero_inv:"(λy. poly_y p y = 0) constant_on S" and
    distinct_p_inv:
      "(λy. (card (proots (pc y)))) constant_on S" and
    deg_q_inv:"(λy. degree (poly_y q y)) constant_on S" and
    qzero_inv:"(λy. poly_y q y = 0) constant_on S" and
    distinct_q_inv:
      "(λy. (card ( proots (qc y)))) constant_on S" and
    common_pq_inv:"(λy. degree (gcd (pc y) (qc y))) constant_on S"
  shows "(λy. card (proots (poly_y (p*q) y))) constant_on S"
```

The proof relies on that polynomial roots continuously depend on the coefficients, which was further derived by Rouché's theorem[12].

[12] Li and Paulson, "A formal proof of Cauchy's residue theorem".

What's left for multivariate CAD

In general, we decided to fully certify the projection phase and deal with base & lifting in a certificate-based way.

The undergoing formalisation efforts are:

- multivariate sign determination
- multivariate subresultants (univariate ones are already available[13])

Still, costly algebraic arithmetic has been avoided!

[13] Joosten, Thiemann, and Yamada, "Subresultants".

Remarks

Formalisation is time consuming – we may want to use certificates if possible.

Many objects and sub-procedures in computer algebra are already in the Archive of Formal Proofs:

- executable multivariate polynomials
- procedures to count real or complex roots of a polynomial
- subresultants
- polynomial factorisation
- Gröbner bases
- ODE
- ...

We can expect more verified computation in proof assistants.

- Boulier, François et al. “Real root isolation of regular chains”. In: *Computer Mathematics*. Springer, 2014, pp. 33–48.
- Cohen, Cyril. “Construction of Real Algebraic Numbers in Coq.” In: *Proceedings of the 3rd International Conference on Interactive Theorem Proving, ITP 2012*. Ed. by Lennart Beringer and Amy Felty. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 67–82.
- Joosten, Sebastiaan, Ren Thiemann, and Akihisa Yamada. “Subresultants”. In: *Archive of Formal Proofs* (Apr. 2017). `http://isa-afp.org/entries/Subresultants.html`, Formal proof development. ISSN: 2150-914x.
- Joosten, Sebastiaan JC, René Thiemann, and Akihisa Yamada. “A Verified Implementation of Algebraic Numbers in Isabelle/HOL”. In: *Journal of automated reasoning* (2018), pp. 1–27.

Li, Wenda, Grant Olney Passmore, and Lawrence C Paulson. "Deciding Univariate Polynomial Problems Using Untrusted Certificates in Isabelle/HOL". In: *Journal of Automated Reasoning* 44.3 (Aug. 2017), pp. 175–23.

Li, Wenda and Lawrence C Paulson. "A formal proof of Cauchy's residue theorem". In: *Proceedings of the 4th International Conference on Interactive Theorem Proving, ITP 2013*. Ed. by Jasmin Christian Blanchette and Stephan Merz. Nancy, France: Springer, Aug. 2016, pp. 235–251.

– ."A modular, efficient formalisation of real algebraic numbers". In: *Proceedings of the 5th ACM SIGPLAN Conference on Certified Programs and Proofs, CPP 2016*. Ed. by Jeremy Avigad and Adam Chlipala. St. Petersburg, FL, USA: ACM, Jan. 2016, pp. 66–75.

Mahboubi, Assia and Cyril Cohen. "Formal proofs in real algebraic geometry: from ordered fields to quantifier elimination". In: *Logical Methods in Computer Science* 8.1 (2012).

- Moura, Leonardo de and Grant Olney Passmore. "Computation in Real Closed Infinitesimal and Transcendental Extensions of the Rationals." In: *Proceedings of the 24th International Conference on Automated Deduction, CADE '13*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 178–192.
- Narkawicz, Anthony, César A Muñoz, and Aaron Dutle. "Formally-Verified Decision Procedures for Univariate Polynomial Computation Based on Sturm's and Tarski's Theorems." In: *Journal of Automated Reasoning* 54.4 (2015), pp. 285–326.
- Nieuwenhuis, Robert, ed. *CADE-20: 20th International Conference on Automated Deduction, proceedings*. Tallinn, Estonia: Springer-Verlag, 2005.
- Strzeboński, Adam W. "Cylindrical Algebraic Decomposition using validated numerics". In: *Journal of Symbolic Computation* 41.9 (Sept. 2006), pp. 1021–1038.