A Coq formalization of a sign determination algorithm

TYPES – Tallinn, May 20 2015

Cyril Cohen and Mathieu Kohli

Cyril Cohen and Mathieu Kohli – A Coq formalization of a sign determination algorithm – TYPES 2015
Context

Fundamental step in some algorithms in real algebraic geometry is the sign determination.

A naive sign determination algorithm has already been formalized (cf Cohen, Mahboubi, LMCS 2012.)

Our goal: formalize more efficient versions, in order to perform computations.

Example of application: Formally-Verified Decision Procedures for Univariate Polynomial Computation Based on Sturms and Tarskis Theorems, Narkawicz, Muoz, Dutle, JAR 2015
Statement of the problem

Knowing how to compute

$$\text{TaQ}(P, Q) = \sum_{x \in \text{roots}(P)} \text{sign}(Q(x)),$$

Given a polynomial $P$ and a list of $n$ polynomials $\vec{Q}$ and a list of sign conditions $\vec{\sigma} \in \{0, 1, -1\}^n$ we want to compute:

$$\text{cnt}(P, \vec{Q}, \vec{\sigma}) = |\{x \in \text{roots}(P) | \forall i, \text{sign}(Q_i(x)) = \sigma_i\}|,$$

using multiple calls of $\text{TaQ}(P, Q^{\vec{\alpha}})$, with $\vec{Q}^{\vec{\alpha}} = \prod_i Q_i^{\alpha_i}$. 
Naive solution
((Algorithms in real algebraic geometry, Basu, Pollack, Roy)

Trivially

\[
(T(1) \ T(Q) \ T(Q^2)) = (C(Q, 0) \ C(Q, +1) \ C(Q, -1)) \cdot \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & -1 & 1 \end{pmatrix}.
\]

More generally,

\[
\left( T_{\alpha}Q(P, \vec{Q}^{\vec{\alpha}}) \right)_{\vec{\alpha} \in \{0,1,2\}^n} = \left( \text{cnt}(P, \vec{Q}, \vec{\sigma}) \right)_{\vec{\sigma} \in \{0,1,-1\}^n} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & -1 & 1 \end{pmatrix} \otimes^n
\]

by induction on \( n \), with appropriate generalization, cf Cohen, Mahboubi, LMCS 2012.
Efficiency issues

Given a polynomial $P$ and a list of $n$ polynomials $\vec{Q}$ and a list of sign conditions $\vec{\sigma} \in \{0, 1, -1\}^n$ we want to compute:

$$\text{cnt}(P, \vec{Q}, \vec{\sigma}) = |\{x \in \text{roots}(P) | \forall i, \text{sign}(Q_i(x)) = \sigma_i\}|,$$

using multiple calls of $\text{TaQ}(P, Q^{\vec{\alpha}})$, with $Q^{\vec{\alpha}} = \prod_i Q_i^{\alpha_i}$, but:

- not too many calls, i.e. using only a small subset $A$ of $\{0, 1, 2\}^n$,
- with small products (i.e. $|\{i | \alpha_i \neq 0\}|$ as small as possible for each $\alpha \in A$.)
Non empty sign conditions

(Algorithms in real algebraic geometry, Basu, Pollack, Roy)

Since

$$\text{cnt}(P, \vec{Q}, \vec{\sigma}) = |\{x \in \text{roots}(P) | \forall i, \text{sign}(Q_i(x)) = \sigma_i\}|,$$

We have

$$\sum_{\vec{\sigma} \in \{0,1,-1\}^n} \text{cnt}(P, \vec{Q}, \vec{\sigma}) \leq \deg P$$

Hence, at most \(\deg P\) sign conditions \(\vec{\sigma}\) are non empty. Let’s call them \(\Sigma\).
Reduction of the system

(Algorithms in real algebraic geometry, Basu, Pollack, Roy)

We have

\[
\left( \text{TaQ}(P, Q^\alpha) \right)_{\alpha \in \text{Ada}(\Sigma)} = \left( \text{cnt}(P, Q, \sigma) \right)_{\sigma \in \Sigma} \cdot M(\Sigma, \text{Ada}(\Sigma))
\]

where

- \text{Ada}(\Sigma) is a subset of \{0, 1, 2\}^n which depends only on \Sigma,
- \text{Ada}(\Sigma) has small products, i.e. for all \alpha \in \text{Ada}(\Sigma),

\[
|\{i|\alpha_i \neq 0\}| \leq \log |\Sigma|
\]

- \(M(\Sigma, A)\) is a submatrix of the tensor product, which depends only on \Sigma and \(A\). More precisely \(M(\Sigma, A)_{\tilde{\sigma}, \tilde{\alpha}} = \tilde{\sigma}^{\tilde{\alpha}}\)
- \(M(\Sigma, \text{Ada}(\Sigma))\) is invertible (in particular \(|\Sigma| = |\text{Ada}(\Sigma)|\))
Definition of $M(\Sigma, A)$

We have:

$$M(\Sigma, A)_{\vec{\sigma}, \vec{\alpha}} = \vec{\sigma}^\vec{\alpha}$$

We represent it using encodings between a set $s$ and the finite type $\mathcal{I}_{|S|}$ of the same cardinality as $s$.

Definition `sign (i : 'I_3) : int :=
    match val i with
    0 => 0%R | 1 => 1%R | _ => -1%R end.`

Definition `expo (i : 'I_3) : nat :=
    match val i with
    0 => 0%N | 1 => 1%N | _ => 2%N end.`

Definition `mat_coef n (i : 'I_3 ^ n) (j : 'I_3 ^ n) :=
    (\prod_k (sign (i k)) \ + (expo (j k)))%:Q%R.`

Definition `mat n (s : {set 'I_3 ^ n}) (a : {set 'I_3 ^ n}) :
'M[rat]_(_(#|s|, #|a|) := \matrix_(i,j) mat_coef (enum_val i) (enum_val j).`
Extension and restriction

Given \( \vec{\sigma} \in \{0, 1, -1\}^{n+1} \) one can take the restriction \( \vec{\sigma}' \) by taking out the last component:

\[
\text{Definition restrict n X (b : X ^ n.+1) : X ^ n :=}
[\text{ffun i => b (lift ord_max i)].}
\]

Given \( \vec{\sigma} \in \{0, 1, -1\}^n \) and \( x \in \{0, 1, -1\} \), one can form the extension \( (\sigma, x) \in \{0, 1, -1\}^{n+1} \):

\[
\text{Definition extelt n X (x : X) (s : X ^ n) : X ^ n.+1 :=}
[\text{ffun i => if unlift ord_max i is Some j then s j else x}.}
\]

Given \( \Sigma \subset \{0, 1, -1\}^n \) and \( x \in \{0, 1, -1\} \), one can form the extension \( (\Sigma, x) \subset \{0, 1, -1\}^{n+1} \):

\[
\text{Definition extset n X (x : X) (S : \{set X ^ n\}) : \{set X ^ n.+1} :=}
[\text{set extelt x s | s in S}].
\]
Extensions

Given $\Sigma \subset \{0, 1, -1\}^{n+1}$ and a number $m$, one can form the set $\Xi_m$ of restrictions of $\Sigma$ which have at least $m$ different extensions in $\Sigma$

**Definition** $\Xi n X (S : \{\text{set } X \sim n.+1\}) (m : \text{nat}) :=$

- $[\text{set } s : X \sim n | [\exists E : \{\text{set } X\}, (#|E| == m) \&\& [\forall x \text{ in } E, \text{extelt } x \text{ s } \in S]]]$.  

Given $\Sigma \subset \{0, 1, -1\}^n$ and an elements $\vec{\sigma}$, one can form the set of all possible extensions in $\Sigma$.

**Definition** exts $X n (S : \{\text{set } X \sim n.+1\}) (s : X \sim n) :=$

- $[\text{set } (x : X \sim n.+1) \text{ ord_max } | x \text{ in } S \& \text{ restrict } x == s]$.  

**Lemma** card_extsP $(X : \text{finType}) n (S : \{\text{set } X \sim n.+1\}) (s : X \sim n) m :$

- $(s \setminus \text{in } Xi S m) = (m <= #|\text{exts } S s|)$. 

Cyril Cohen and Mathieu Kohli – A Coq formalization of a sign determination algorithm – TYPES 2015
Adapted family

The adapted family $\text{Ada}(\Sigma)$ is defined recursively as the disjoint union of $(\Xi_1, 0)$, $(\Xi_2, 1)$ and $(\Xi_3, 2)$.

```coq
Fixpoint adapt n (S : {set 'I_3 ^ n}) : {set 'I_3 ^ n} :=
  match n return {set 'I_3 ^ n} -> {set 'I_3 ^ n} with
  | 0 => fun S => S
  | n'.+1 => fun S => \bigcup_(i : 'I_3) extset i (adapt (Xi S i.+1))
  end S.
```

We prove the union is disjoint:

```coq
Lemma partition_adapt n (S : {set 'I_3 ^ n.+1}) :
  partition [set extset i (adapt (Xi S (i : 'I_3).+1))
  | i in 'I_3 & Xi S i.+1 != set0] (adapt S).
```
Intermediate results

Lemma Xi_monotonic n (X : finType) (S S' : {set X ^ n.+1}) m :
  S \subset S' -> Xi S m \subset Xi S' m.

Lemma leq_Xi n (X : finType) (S : {set X ^ n.+1}) :
  {homo Xi S : m p / (p <= m)%N >-> m \subset p}.

Lemma adapt_monotonic n (S S' : {set I_3 ^ n}) :
  S \subset S' -> adapt S \subset adapt S'.

Lemma adapt_down_closed n (S : {set I_3 ^ n}) (a b : Expos n) :
  (forall i, b i <= a i)%N -> a \in adapt S -> b \in adapt S.

Lemma partition_Signs n (S : {set I_3 ^ n.+1}) :
  partition [set reext S (i : I__) | i in I_3 & Xi S i.+1 != set0] S.
Main proofs

Completed:

Lemma prop1084 n (S : {set 'I_3 ^ n}) a :
a \in adapt S -> 2 ^ #\{set i : 'I_n | a i != 0%R\} <= #|S|.

Lemma card_adapt n (S : {set 'I_3 ^ n}) : #|adapt S| = #|S|.

Ongoing:

Lemma adapt_adapted n (S : {set 'I_3 ^ n}) : adapted S (adapt S).
Difficulties

Encountered

- A lot of reindexing (kept implicit in the book)
- Many different partitioning of the same set (kept implicit in the book).

Avoided (so far):

- Using matrices with judgmentally different but propositionally identical indexes.
- Set extensionality problems, thanks to finite sets.
Conclusions

• The new formal proof of \( \text{prop1084} \) and the intermediate lemmas was backported to the future revision of the book.

• The new paper proof of \( \text{adapt\_adapted} \) contains a pseudo-recurrence which was not in the first version.
We want to prove that all $\lambda_r$ are zero.

If $\sigma \in \text{SIGN}(Q, Z)_3$, we denote by $\sigma_1 \preceq_{\text{lex}} \sigma_2 \preceq_{\text{lex}} \sigma_3$ the sign conditions of $\text{SIGN}(P, Z)$ extending $\sigma$.

Similarly, if $\sigma \in \text{SIGN}(Q, Z)_2 \setminus \text{SIGN}(Q, Z)_3$, we denote by

$$\sigma_1 \preceq_{\text{lex}} \sigma_2$$

the sign conditions of $\text{SIGN}(P, Z)$ extending $\sigma$.

Finally if $\sigma \in \text{SIGN}(Q, Z) \setminus \text{SIGN}(Q, Z)_2$, we denote by $\sigma_1$ the sign condition of $\text{SIGN}(P, Z)$ extending $\sigma$.

Since by induction hypothesis, the matrix

$$\text{Mat}(\text{Ada}(Q, Z), \text{SIGN}(Q, Z))$$

is invertible,

$$\lambda_{\sigma_1} = 0, \quad \text{for } \sigma \in \text{SIGN}(Q, Z) \setminus \text{SIGN}(Q, Z)_2,$$

$$\lambda_{\sigma_1} + \lambda_{\sigma_2} = 0, \quad \text{for } \sigma \in \text{SIGN}(Q, Z)_2 \setminus \text{SIGN}(Q, Z)_3,$$

$$\lambda_{\sigma_1} + \lambda_{\sigma_2} + \lambda_{\sigma_3} = 0, \quad \text{for } \sigma \in \text{SIGN}(Q, Z)_3.$$

Thus $\lambda_{\sigma_1} = 0$ for every $\sigma \in \text{SIGN}(Q, Z) \setminus \text{SIGN}(Q, Z)_2$.

Using again the induction hypothesis, the matrix

$$\text{Mat}(\text{Ada}(Q, Z)_2, \text{SIGN}(Q, Z)_2)$$

is invertible, so

$$\sigma_1(P) \lambda_{\sigma_1} - \sigma_2(P) \lambda_{\sigma_2} = 0, \quad \text{for } \sigma \in \text{SIGN}(Q, Z)_2 \setminus \text{SIGN}(Q, Z)_3,$$

$$\lambda_{\sigma_2} - \lambda_{\sigma_3} = 0, \quad \text{for } \sigma \in \text{SIGN}(Q, Z)_3.$$

Thus $\lambda_{\sigma_1} = \lambda_{\sigma_2} = 0$, for every $\sigma \in \text{SIGN}(Q, Z)_2 \setminus \text{SIGN}(Q, Z)_3$.

Finally, using once more the induction hypothesis, the matrix

$$\text{Mat}(\text{Ada}(Q, Z)_3, \text{SIGN}(Q, Z)_3)$$

$$\ldots$$
i) We prove the statement in the special case when $\Xi^{(0)} = \Xi^{(2)}$. The statement is clear since $\text{Mat}(\text{Ada}(\Sigma), \Sigma)$ is the tensor product of $\text{Mat}(\text{Ada}(\Xi^{(2)}), \Xi^{(2)})$ by

$$
\begin{bmatrix}
1 & 1 & 1 \\
0 & 1 & -1 \\
0 & 1 & 1
\end{bmatrix},
$$

and is invertible, because $\text{Mat}(\text{Ada}(\Xi^{(2)}), \Xi^{(2)})$ is.

ii) We now prove the statement in the special case when $\Xi^{(0)} = \Xi^{(1)}$. We first prove that all $\lambda_\tau$ for $\tau$ having a restriction to $\Xi^{(1)} \setminus \Xi^{(2)}$ are zero. Indeed, using that $\text{Mat}(\text{Ada}(\Xi^{(1)}), \Xi^{(1)})$ is invertible

(a) $\lambda_{\sigma_1} + \lambda_{\sigma_2} = 0$ and $\sigma_1(P_i)\lambda_{\sigma_1} + \sigma_2(P_i)\lambda_{\sigma_2} = 0$, for every $\sigma \in \Xi^{(1)} \setminus \Xi^{(2)}$,
(b) $\lambda_{\sigma_1} + \lambda_{\sigma_2} + \lambda_{\sigma_3} = 0$ and $\lambda_{\sigma_2} - \lambda_{\sigma_3} = 0$, for every $\sigma \in \Xi^{(2)}$.

We conclude by i), removing from $\Sigma$ the elements having a restriction in $\Xi^{(1)} \setminus \Xi^{(2)}$.

iii) We finally prove the statement for a general $\Xi^{(0)}$. We want to prove that all $\lambda_\tau$ are zero. We first prove that all $\lambda_\tau$ for $\tau$ having a restriction in $\Xi^{(0)} \setminus \Xi^{(1)}$ are zero. Indeed, using that $\text{Mat}(\text{Ada}(\Xi^{(0)}), \Xi^{(0)})$ is invertible,

(a) $\lambda_{\sigma_1} = 0$, for every $\sigma \in \Xi^{(0)} \setminus \Xi^{(1)}$,
(b) $\lambda_{\sigma_1} + \lambda_{\sigma_2} = 0$, for every $\sigma \in \Xi^{(2)} \setminus \Xi^{(1)}$,
(c) $\lambda_{\sigma_1} + \lambda_{\sigma_2} + \lambda_{\sigma_3} = 0$, for every $\sigma \in \Xi^{(2)}$.

We conclude by ii), removing from $\Sigma$ the elements having a restriction in $\Xi^{(0)} \setminus \Xi^{(1)}$. $\square$
Future work

• Finish adapt_adapted
• Reintegration into the previous development.
• Efficient computation using refinements.
Thanks for your attention