Some decidability issues concerning C^n real functions

Gabriele Buriola[1], Domenico Cantone[2], Gianluca Cincotti[2], Eugenio Omodeo[3], Gaetano Spartà[4]

[1]University of Verona (I), [2]University of Catania (I), [3]University of Trieste (I) and [4]Pontificia Università Gregoriana (I)

CILC 2024

Tarski's theory of reals - Description

Tarski's theory of reals

(A. Tarski, 1939/1951).

A first-order (fully quantified) theory of real numbers with operations $+$, $\cdot$, $-$ and relations $>$, $<$, $=$.

An example:
$\forall a \, \forall b \, \forall c \, \forall d \; [\, a \neq 0 \rightarrow \exists x \, (ax^3 + bx^2 + cx + d = 0\,)\,]$

Theorem (Tarski, 1951)

Tarski's theory of reals is decidable.

Extensions:
Complex numbers, n-dimensional vectors, plane geometry, space geometry, n-dimensional geometry, non-Euclidean geometries, projective geometry.

Limitations:
Tarski's theory of reals can not express the predicate isInteger(x).

Tarski's theory of reals - Description

Tarski's theory of reals

(A. Tarski, 1939/1951).

A first-order (fully quantified) theory of real numbers with operations $+$, $\cdot$, $-$ and relations $>$, $<$, $=$.

An example:

$\forall a \, \forall b \, \forall c \, \forall d \, [\, a \neq 0 \rightarrow \exists x \, (ax^3 + bx^2 + cx + d = 0\,)\,]$

Theorem (Tarski, 1951)

Tarski's theory of reals is decidable.

Extensions:

Complex numbers, n-dimensional vectors, plane geometry, space geometry, n-dimensional geometry, non-Euclidean geometries, projective geometry.

Limitations:

Tarski's theory of reals can not express the predicate isInteger(x).

Tarski's theory of reals - Description

Tarski's theory of reals

(A. Tarski, 1939/1951).

A first-order (fully quantified) theory of real numbers with operations $+$, $\cdot$, $-$ and relations $>$, $<$, $=$.

An example:
$\forall a\ \forall b\ \forall c\ \forall d\ [\, a \neq 0 \rightarrow \exists x\, (ax^3 + bx^2 + cx + d = 0\,)\,]$

Theorem (Tarski, 1951)

Tarski's theory of reals is decidable.

Extensions:
Complex numbers, n-dimensional vectors, plane geometry, space geometry, n-dimensional geometry, non-Euclidean geometries, projective geometry.

Limitations:
Tarski's theory of reals can not express the predicate isInteger(x).

Tarski's theory of reals - Description

Tarski's theory of reals

(A. Tarski, 1939/1951).

A first-order (fully quantified) theory of real numbers with operations $+$, $\cdot$, $-$ and relations $>$, $<$, $=$.

An example:
$\forall a \; \forall b \; \forall c \; \forall d \; [\, a \neq 0 \rightarrow \exists x \, (ax^3 + bx^2 + cx + d = 0\,)\,]$

Theorem (Tarski, 1951)

Tarski's theory of reals is decidable.

Extensions:
Complex numbers, *n*-dimensional vectors, plane geometry, space geometry, *n*-dimensional geometry, non-Euclidean geometries, projective geometry.

Limitations:
Tarski's theory of reals can not express the predicate isInteger(x).

Tarski's theory of reals - Description

Tarski's theory of reals

(A. Tarski, 1939/1951).

A first-order (fully quantified) theory of real numbers with operations $+$, $\cdot$, $-$ and relations $>$, $<$, $=$.

An example:
$\forall a\ \forall b\ \forall c\ \forall d\ [\, a \neq 0 \rightarrow \exists x\, (ax^3 + bx^2 + cx + d = 0\,)\,]$

Theorem (Tarski, 1951)

Tarski's theory of reals is decidable.

Extensions:
Complex numbers, *n*-dimensional vectors, plane geometry, space geometry, *n*-dimensional geometry, non-Euclidean geometries, projective geometry.

Limitations:
Tarski's theory of reals can not express the predicate isInteger(x).

Applications to analysis

Some decidable fragments of real analysis:

- $RMCF \hookrightarrow RMCF^+$, (continuous functions)
- $RDF \hookrightarrow RDF^+ \hookrightarrow RDF^* \hookrightarrow RDF^n$. (continuous functions with derivatives)

Theory RDF^n

(Theory of Reals with n-Differentiable Functions - G. Buriola, D. Cantone, G. Cincotti, E. Omodeo, G. Spartà).

An unquantified first-order theory of real functions of a real variable each endowed with continuous derivatives up to n-th order, which includes predicates expressing function **comparisons**, **concavity**, **convexity**, **monotonicity strict monotonicity** and **comparisons** between a function (or one of its derivatives) and a real term on closed, open or semi-open intervals, bounded or unbounded.

Applications to analysis

Some decidable fragments of real analysis:

- $RMCF \hookrightarrow RMCF^+$, (continuous functions)
- $RDF \hookrightarrow RDF^+ \hookrightarrow RDF^* \hookrightarrow RDF^n$. (continuous functions with derivatives)

Theory RDF^n

(Theory of Reals with n-Differentiable Functions -
G. Buriola, D. Cantone, G. Cincotti, E. Omodeo, G. Spartà).

An unquantified first-order theory of real functions of a real variable each endowed with continuous derivatives up to n-th order, which includes predicates expressing function **comparisons**, **concavity**, **convexity**, **monotonicity strict monotonicity** and **comparisons** between a function (or one of its derivatives) and a real term on closed, open or semi-open intervals, bounded or unbounded.

Applications to analysis

Some decidable fragments of real analysis:

- $RMCF \hookrightarrow RMCF^+$, (continuous functions)
- $RDF \hookrightarrow RDF^+ \hookrightarrow RDF^* \hookrightarrow RDF^n$. (continuous functions with derivatives)

Theory RDF^n

(Theory of Reals with n-Differentiable Functions -
G. Buriola, D. Cantone, G. Cincotti, E. Omodeo, G. Spartà).

An unquantified first-order theory of real functions of a real variable each endowed with continuous derivatives up to n-th order, which includes predicates expressing function **comparisons**, **concavity**, **convexity**, **monotonicity strict monotonicity** and **comparisons** between a function (or one of its derivatives) and a real term on closed, open or semi-open intervals, bounded or unbounded.

Syntax

Idea:
Enrich Tarski's arithmetic by adding variables and relations concerning function terms.

Syntax:
We have two types of variables:

- numerical variables $x, y, z, \ldots$, representing real numbers,
- function variables $f, g, h, \ldots$ representing C^n real functions,

and some constant symbols:

- 0 and 1, designating the numbers 0 and 1,
- the symbols $+\infty$ and $-\infty$, occurring only as endpoints of interval domains;

out of these we build up two types of terms:

- function terms, obtained from function variables:

$$f + g \quad \text{and} \quad t \cdot f,$$

- numerical terms, obtained by combining numerical variables:

$$t_1 + t_2, \; t_1 - t_2, \; t_1 * t_2,$$

- or, by intermixing numerical terms and function terms:

$$\mathfrak{f}(t), \quad D^k[\mathfrak{f}](t),$$

Syntax

Idea:
Enrich Tarski's arithmetic by adding variables and relations concerning function terms.

Syntax:
We have two types of variables:

- numerical variables $x, y, z, \ldots,$ representing real numbers,
- function variables $f, g, h, \ldots$ representing C^n real functions,

and some constant symbols:

- 0 and 1, designating the numbers 0 and 1,
- the symbols $+\infty$ and $-\infty$, occurring only as endpoints of interval domains;

out of these we build up two types of terms:

- function terms, obtained from function variables:

$$f + g \quad \text{and} \quad t \cdot f,$$

- numerical terms, obtained by combining numerical variables:

$$t_1 + t_2, \; t_1 - t_2, \; t_1 * t_2,$$

- or, by intermixing numerical terms and function terms:

$$\mathfrak{f}(t), \quad D^k[\mathfrak{f}](t),$$

Syntax

Idea:
Enrich Tarski's arithmetic by adding variables and relations concerning function terms.

Syntax:
We have two types of variables:

- numerical variables $x, y, z, \ldots,$ representing real numbers,
- function variables $f, g, h, \ldots$ representing C^n real functions,

and some constant symbols:

- 0 and 1, designating the numbers 0 and 1,
- the symbols $+\infty$ and $-\infty$, occurring only as endpoints of interval domains;

out of these we build up two types of terms:

- function terms, obtained from function variables:

$$f + g \quad \text{and} \quad t \cdot f,$$

- numerical terms, obtained by combining numerical variables:

$$t_1 + t_2, \; t_1 - t_2, \; t_1 * t_2,$$

- or, by intermixing numerical terms and function terms:

$$\mathfrak{f}(t), \quad D^k[\mathfrak{f}](t),$$

Syntax

Idea:
Enrich Tarski's arithmetic by adding variables and relations concerning function terms.

Syntax:
We have two types of variables:

- numerical variables $x, y, z, \dots,$ representing real numbers,
- function variables $f, g, h, \dots$ representing C^n real functions,

and some constant symbols:

- 0 and 1, designating the numbers 0 and 1,
- the symbols $+\infty$ and $-\infty$, occurring only as endpoints of interval domains;

out of these we build up two types of terms:

- function terms, obtained from function variables:

$$f + g \quad \text{and} \quad t \cdot f,$$

- numerical terms, obtained by combining numerical variables:

$$t_1 + t_2, \; t_1 - t_2, \; t_1 * t_2,$$

- or, by intermixing numerical terms and function terms:

$$\mathfrak{f}(t), \quad D^k[\mathfrak{f}](t),$$

Atomic formulas

Atomic formulas of RDF^n:

$$
\begin{array}{rl}
t_1 = t_2 \ , & t_1 > t_2 \ , \\
\mathfrak{f}(s) = t \ , & D^k[\mathfrak{f}](s) = t \ , \\
(\mathfrak{f} = \mathfrak{g})_A \ , & (\mathfrak{f} > \mathfrak{g})_A \ , \\
\mathrm{Up}(\mathfrak{f})_A \ , & \mathrm{Strict\_Up}(\mathfrak{f})_A \ , \\
\mathrm{Down}(\mathfrak{f})_A \ , & \mathrm{Strict\_Down}(\mathfrak{f})_A \ , \\
\mathrm{Convex}(\mathfrak{f})_A \ , & \mathrm{Strict\_Convex}(\mathfrak{f})_A \ , \\
\mathrm{Concave}(\mathfrak{f})_A \ , & \mathrm{Strict\_Concave}(\mathfrak{f})_A \ , \\
(D^k[\mathfrak{f}] \bowtie t)_A \ , & \text{with } \bowtie \in \{<, >, =, \leq, \geq\} \ ,
\end{array}
$$

where A is a closed, open or semi-open interval, bounded or unbounded.

Derived relators:

$$
\begin{array}{rcl}
\mathrm{Linear}(\mathfrak{f})_A & \leftrightarrow_{\mathrm{Def}} & \mathrm{Convex}(\mathfrak{f})_A \wedge \mathrm{Concave}(\mathfrak{f})_A \\
(D[\mathfrak{f}] \neq t)_A & \leftrightarrow_{\mathrm{Def}} & (D[\mathfrak{f}] < t)_A \vee (D[\mathfrak{f}] > t)_A \\
\left(g = \frac{m}{n} \cdot f\right)_{]-\infty,+\infty[} & \leftrightarrow_{\mathrm{Def}} & \left(\underbrace{g + \cdots + g}_{n \ \text{times}} = \underbrace{f + \cdots + f}_{m \ \text{times}}\right)_{]-\infty,+\infty[}
\end{array}
$$

Atomic formulas

Atomic formulas of RDF^n:

$$
\begin{array}{rl}
t_1 = t_2 \ , & t_1 > t_2 \ , \\
\mathfrak{f}(s) = t \ , & D^k[\mathfrak{f}](s) = t \ , \\
(\mathfrak{f} = \mathfrak{g})_A \ , & (\mathfrak{f} > \mathfrak{g})_A \ , \\
\mathrm{Up}(\mathfrak{f})_A \ , & \mathrm{Strict\_Up}(\mathfrak{f})_A \ , \\
\mathrm{Down}(\mathfrak{f})_A \ , & \mathrm{Strict\_Down}(\mathfrak{f})_A \ , \\
\mathrm{Convex}(\mathfrak{f})_A \ , & \mathrm{Strict\_Convex}(\mathfrak{f})_A \ , \\
\mathrm{Concave}(\mathfrak{f})_A \ , & \mathrm{Strict\_Concave}(\mathfrak{f})_A \ , \\
(D^k[\mathfrak{f}] \bowtie t)_A \ , & \text{with } \bowtie \in \{<, >, =, \leq, \geq\} \ ,
\end{array}
$$

where A is a closed, open or semi-open interval, bounded or unbounded.

Derived relators:

$$
\begin{array}{ccc}
\mathrm{Linear}(\mathfrak{f})_A & \leftrightarrow_{\mathrm{Def}} & \mathrm{Convex}(\mathfrak{f})_A \wedge \mathrm{Concave}(\mathfrak{f})_A \\
(D[\mathfrak{f}] \neq t)_A & \leftrightarrow_{\mathrm{Def}} & (D[\mathfrak{f}] < t)_A \vee (D[\mathfrak{f}] > t)_A \\
\left(g = \frac{m}{n} \cdot f\right)_{]-\infty,+\infty[} & \leftrightarrow_{\mathrm{Def}} & \left(\underbrace{g + \cdots + g}_{n \ times} = \underbrace{f + \cdots + f}_{m \ times}\right)_{]-\infty,+\infty[}
\end{array}
$$

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, f+g, . . . , are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, f+g, ..., are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, f+g, ..., are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, $\mathrm{f}{+}\mathrm{g}$, ..., are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, f+g, . . . , are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, f+g, . . . , are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, f+g, . . . , are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Semantics of RDF^n

In the standard semantics for RDF^n:

- number variables are real numbers;
- function variables are C^n functions from $\mathbb{R}$ to $\mathbb{R}$;
- terms: $s \cdot t$, f+g, . . . , are interpreted accordingly;
- atomic formulas are true according their analytic "meaning":
- e.g., $(f > g)_A$ is true if: $\forall x \in \tilde{A}\ \tilde{f}(x) > \tilde{g}(x)$;
- other formulas are evaluated according the connectives.

Satisfiability and Validity

The decision problem:
Since RDF^n is an unquantified theory, the related decision problem shifts from *truth*- to *validity*- checking.
We want to establish whether or not a formula of RDF^n is *valid*, i.e., *true* under any assignment.

Validity and Satisfiability:
A decision algorithm for validity exists **if and only if** a decision algorithm for satisfiability exists, because "θ is valid if and only if $\neg\theta$ is unsatisfiable."

The decision algorithm: Through the algorithm we transform a formula θ of RDF^n into an equisatisfiable formula ψ which belongs to Tarski's elementary algebra. We then submit ψ to Tarski's decision method.

Satisfiability and Validity

The decision problem:
Since RDF^n is an unquantified theory, the related decision problem shifts from *truth*- to *validity*- checking.
We want to establish whether or not a formula of RDF^n is *valid*, i.e., *true* under any assignment.

Validity and Satisfiability:
A decision algorithm for validity exists **if and only if** a decision algorithm for satisfiability exists, because "θ is valid if and only if $\neg\theta$ is unsatisfiable."

The decision algorithm: Through the algorithm we transform a formula θ of RDF^n into an equisatisfiable formula ψ which belongs to Tarski's elementary algebra.
We then submit ψ to Tarski's decision method.

Satisfiability and Validity

The decision problem:
Since RDF^n is an unquantified theory, the related decision problem shifts from *truth*- to *validity*- checking.
We want to establish whether or not a formula of RDF^n is *valid*, i.e., *true* under any assignment.

Validity and Satisfiability:
A decision algorithm for validity exists **if and only if** a decision algorithm for satisfiability exists, because "θ is valid if and only if $\neg\theta$ is unsatisfiable."

The decision algorithm: Through the algorithm we transform a formula θ of RDF^n into an equisatisfiable formula ψ which belongs to Tarski's elementary algebra.
We then submit ψ to Tarski's decision method.

A few examples

1. $\text{Linear}(f)_{]-\infty,+\infty[} \quad \longleftrightarrow \quad (D^2[f]=0)_{]-\infty,+\infty[}$.
2. $\{(a<x<b) \ \wedge \ [(\text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}) \ \vee$ $(\text{S\_Concave}(f)_{[a,x]} \wedge \ \text{S\_Convex}(f)_{[x,b]})]\} \quad \rightarrow \quad D^2[f](x)=0$.
3. $\left[(D^{k-1}[f]=y)_{]-\infty,\infty[} \quad \rightarrow \quad (D^k[f]=0)_{]-\infty,\infty[}\right] \ \wedge$ $\left\{(D^k[f]=0)_{]-\infty,\infty[} \quad \rightarrow \quad \left[D^{k-1}[f](x)=y \ \rightarrow \ (D^{k-1}[f]=y)_{]-\infty,\infty[}\right]\right\}$.
4. $$\left\{(a<x<b) \ \wedge \ D[f](x)=0 \ \wedge \ (D^2[f]\geq 0)_{[a,b]} \ \wedge \ f(x)=y\right\} \quad \rightarrow \quad (f\geq y)_{[a,b]};$$ $$\left\{(a<x<b) \ \wedge \ D[f](x)=0 \ \wedge \ (D^2[f]\leq 0)_{[a,b]} \ \wedge \ f(x)=y\right\} \quad \rightarrow \quad (f\leq y)_{[a,b]}.$$
5. $$\left\{\begin{array}{c} (a<x<b) \ \wedge \ D[f](x)=0 \ \wedge \ D^2[f](x)=0 \ \wedge \\ \left[(D^3[f]<0)_{[a,b]} \ \vee \ (D^3[f]>0)_{[a,b]}\right] \end{array}\right\} \quad \rightarrow$$ $$\left\{\begin{array}{c} (\text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}) \ \vee \\ (\text{S\_Concave}(f)_{[a,x]} \ \wedge \ \text{S\_Convex}(f)_{[x,b]}) \end{array}\right\}.$$

The algorithm at work

We briefly illustrate the algorithm in the case of one of the previous examples:

$$\{(a < x < b) \ \wedge \ \text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

Step 0: consider the negation of our formula,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

Step 1: do some preliminaries in case of not closed intervals.

Step 2: negative literals with intervals are substituted with suitable existential conditions.

Step 3: evaluate all function variables over the so-called "domain variables". Let us do the renaming: $a \rightsquigarrow v_1, \quad x \rightsquigarrow v_2, \quad b \rightsquigarrow v_3$. From the previous formula we get the following:

$$\begin{array}{ccccccccc}
(v_1 < v_2 < v_3) & \wedge & \text{S_Convex}(f)_{[v_1,v_2]} & \wedge & \text{S_Concave}(f)_{[v_2,v_3]} & \wedge & D^2[f](v_2) \neq 0 & \wedge \\
f(v_1) = y_1^f & \wedge & f(v_2) = y_2^f & \wedge & f(v_3) = y_3^f & \wedge & & \\
D^1[f](v_1) = t_1^f & \wedge & D^1[f](v_2) = t_2^f & \wedge & D^1[f](v_3) = t_3^f & \wedge & & \\
D^2[f](v_1) = s_1^f & \wedge & D^2[f](v_2) = s_2^f & \wedge & D^2[f](v_3) = s_3^f & \wedge & s_2^f \neq 0. &
\end{array}$$

The algorithm at work

We briefly illustrate the algorithm in the case of one of the previous examples:

$$\{(a < x < b) \ \wedge \ \text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

Step 0: consider the negation of our formula,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

Step 1: do some preliminaries in case of not closed intervals.

Step 2: negative literals with intervals are substituted with suitable existential conditions.

Step 3: evaluate all function variables over the so-called "domain variables". Let us do the renaming: $a \rightsquigarrow v_1, \quad x \rightsquigarrow v_2, \quad b \rightsquigarrow v_3$. From the previous formula we get the following:

$$\begin{array}{ccccccc} (v_1 < v_2 < v_3) & \wedge & \text{S\_Convex}(f)_{[v_1,v_2]} & \wedge & \text{S\_Concave}(f)_{[v_2,v_3]} & \wedge & D^2[f](v_2) \neq 0 \quad \wedge \\ f(v_1) = y_1^f & \wedge & f(v_2) = y_2^f & \wedge & f(v_3) = y_3^f & \wedge & \\ D^1[f](v_1) = t_1^f & \wedge & D^1[f](v_2) = t_2^f & \wedge & D^1[f](v_3) = t_3^f & \wedge & \\ D^2[f](v_1) = s_1^f & \wedge & D^2[f](v_2) = s_2^f & \wedge & D^2[f](v_3) = s_3^f & \wedge & s_2^f \neq 0. \end{array}$$

The algorithm at work

We briefly illustrate the algorithm in the case of one of the previous examples:

$$\{(a < x < b) \ \wedge \ \text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

Step 0: consider the negation of our formula,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

Step 1: do some preliminaries in case of not closed intervals.

Step 2: negative literals with intervals are substituted with suitable existential conditions.

Step 3: evaluate all function variables over the so-called "domain variables". Let us do the renaming: $a \rightsquigarrow v_1, \quad x \rightsquigarrow v_2, \quad b \rightsquigarrow v_3$. From the previous formula we get the following:

$$\begin{array}{ccccccccc}
(v_1 < v_2 < v_3) & \wedge & \text{S_Convex}(f)_{[v_1,v_2]} & \wedge & \text{S_Concave}(f)_{[v_2,v_3]} & \wedge & D^2[f](v_2) \neq 0 & \wedge \\
f(v_1) = y_1^f & \wedge & f(v_2) = y_2^f & \wedge & f(v_3) = y_3^f & \wedge & & \\
D^1[f](v_1) = t_1^f & \wedge & D^1[f](v_2) = t_2^f & \wedge & D^1[f](v_3) = t_3^f & \wedge & & \\
D^2[f](v_1) = s_1^f & \wedge & D^2[f](v_2) = s_2^f & \wedge & D^2[f](v_3) = s_3^f & \wedge & s_2^f \neq 0. &
\end{array}$$

The algorithm at work

We briefly illustrate the algorithm in the case of one of the previous examples:

$$\{(a < x < b) \ \wedge \ \text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

Step 0: consider the negation of our formula,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

Step 1: do some preliminaries in case of not closed intervals.

Step 2: negative literals with intervals are substituted with suitable existential conditions.

Step 3: evaluate all function variables over the so-called "domain variables". Let us do the renaming: $a \rightsquigarrow v_1, \quad x \rightsquigarrow v_2, \quad b \rightsquigarrow v_3$. From the previous formula we get the following:

$$\begin{array}{ccccccccc}
(v_1 < v_2 < v_3) & \wedge & \text{S_Convex}(f)_{[v_1,v_2]} & \wedge & \text{S_Concave}(f)_{[v_2,v_3]} & \wedge & D^2[f](v_2) \neq 0 & \wedge \\
f(v_1) = y_1^f & \wedge & f(v_2) = y_2^f & \wedge & f(v_3) = y_3^f & \wedge & & \\
D^1[f](v_1) = t_1^f & \wedge & D^1[f](v_2) = t_2^f & \wedge & D^1[f](v_3) = t_3^f & \wedge & & \\
D^2[f](v_1) = s_1^f & \wedge & D^2[f](v_2) = s_2^f & \wedge & D^2[f](v_3) = s_3^f & \wedge & s_2^f \neq 0. &
\end{array}$$

The algorithm at work

We briefly illustrate the algorithm in the case of one of the previous examples:

$$\{(a < x < b) \ \wedge \ \text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

Step 0: consider the negation of our formula,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

Step 1: do some preliminaries in case of not closed intervals.

Step 2: negative literals with intervals are substituted with suitable existential conditions.

Step 3: evaluate all function variables over the so-called "domain variables". Let us do the renaming: $a \rightsquigarrow v_1, \quad x \rightsquigarrow v_2, \quad b \rightsquigarrow v_3$. From the previous formula we get the following:

$$\begin{array}{ccccccccc}
(v_1 < v_2 < v_3) & \wedge & \text{S_Convex}(f)_{[v_1,v_2]} & \wedge & \text{S_Concave}(f)_{[v_2,v_3]} & \wedge & D^2[f](v_2) \neq 0 & \wedge \\
f(v_1) = y_1^f & \wedge & f(v_2) = y_2^f & \wedge & f(v_3) = y_3^f & \wedge & & \\
D^1[f](v_1) = t_1^f & \wedge & D^1[f](v_2) = t_2^f & \wedge & D^1[f](v_3) = t_3^f & \wedge & & \\
D^2[f](v_1) = s_1^f & \wedge & D^2[f](v_2) = s_2^f & \wedge & D^2[f](v_3) = s_3^f & \wedge & s_2^f \neq 0. &
\end{array}$$

The algorithm at work (2)

Step 4: replace all literals involving functional terms by algebraic conditions,

$$
\begin{array}{ccccccc}
(v_1 < v_2 < v_3) & \wedge & s_2^f \neq 0 & \wedge & & & \\
t_1^f < \frac{y_2^f - y_1^f}{v_2 - v_1} < t_2^f & \wedge & s_1^f \geq 0 & \wedge & s_2^f \geq 0 & \wedge & \\
t_2^f > \frac{y_3^f - y_2^f}{v_3 - v_2} > t_3^f & \wedge & s_2^f \leq 0 & \wedge & s_3^f \leq 0. & &
\end{array}
$$

The result

The output of the algorithm is a conjunction formula ψ of the Elementary Algebra of Real numbers (*EAR*), decidable by Tarski's well-know result.
It contains the following unsatisfiable conjunction:

$$s_2^f \neq 0 \wedge s_2^f \geq 0 \wedge s_2^f \leq 0.$$

Thus,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

is unsatisfiable

$\Downarrow$

$$\{(a < x < b) \ \wedge \ \text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

is valid.

The result

The output of the algorithm is a conjunction formula ψ of the Elementary Algebra of Real numbers (*EAR*), decidable by Tarski's well-know result.
It contains the following unsatisfiable conjunction:

$$s_2^f \neq 0 \wedge s_2^f \geq 0 \wedge s_2^f \leq 0.$$

Thus,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

is unsatisfiable

$$\Downarrow$$

$$\{(a < x < b) \ \wedge \ \text{S\_Convex}(f)_{[a,x]} \ \wedge \ \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

is valid.

The result

The output of the algorithm is a conjunction formula ψ of the Elementary Algebra of Real numbers (*EAR*), decidable by Tarski's well-know result.
It contains the following unsatisfiable conjunction:

$$s_2^f \neq 0 \wedge s_2^f \geq 0 \wedge s_2^f \leq 0.$$

Thus,

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

is unsatisfiable

$\Downarrow$

$$\{(a < x < b) \wedge \text{S\_Convex}(f)_{[a,x]} \wedge \text{S\_Concave}(f)_{[x,b]}\} \quad \rightarrow \quad D^2[f](x) = 0,$$

is valid.

Correctness

The correctness of the algorithm amounts to the equisatisfiability of input formula $\neg\theta$:

$$(a < x < b) \quad \wedge \quad \text{S\_Convex}(f)_{[a,x]} \quad \wedge \quad \text{S\_Concave}(f)_{[x,b]} \quad \wedge \quad D^2[f](x) \neq 0.$$

and the output formula ψ, in particular with respect the conjunction

$$s_2^f \neq 0 \wedge s_2^f \geq 0 \wedge s_2^f \leq 0.$$

$\neg\theta \Rightarrow \psi$: given a model of $\neg\theta$, viz. three interpreting functions f, g, h and real values for the numerical variables, we must find a set of real numbers satisfying ψ.

$\psi \Rightarrow \neg\theta$: given a model of ψ, viz. a set of real numbers satisfying some algebraic conditions, we must define real functions satisfying the analytics properties of $\neg\theta$. For the function variable f we take a suitable interpolation function between points (a, v_a^f).

We have produced explicitly an *ad hoc* interpolation method for the case $n = 1$; when $n = 2$, we could borrow an interpolation method due to C. Manni; when $n > 2$, we hope for, and remain in debt with the listener of, a proof of existence of the suitable interpolating function.

The threshold of undecidability

Tarski himself showed that decidability of his full elementary algebra of real numbers would be disrupted if its language were enriched with a periodic real function, e.g., $\sin x$.
D. Richardson proved the undecidability of the existential theory of reals extended with the numbers $\log 2$ and π, and with the functions e^x, $\sin x$; these results have been subsequently improved by B. F. Caviness, P. S. Wang and M. Laczkovich.

In consequence of Laczkovich's result and of our reduction of RDF^n to Tarskian algebra, any extension of RDF^n enabling us to express $\sin x$ turns out to be undecidable. For example, an atomic formula $\left(D^2[f] = g\right)_A$ for equality between a second derivative and a function would allow one to specify $f = \sin x$ through the differential characterization:

$$f(0) = 0 \quad D^1[f](0) = 1 \quad (D^2[f] = -f)_{]-\infty,+\infty[} \,.$$

Establishing whether or not an analogous extension of RDF^1 is decidable is harder.

Extensions and applications

Other **extensions** of *EAR* could be:

1. the theory RDF^∞, whose set of formulas is the union of RDF^n formulas for all n;
2. decision methods regarding differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^m$.

Applications in automated theorem proving:

- proof verification of **mathematical theories**;
- **program verification** and **hardware validation**.

$RMCF^+$, RDF^* and RDF^n should be integrated in the **proof-checker ÆtnaNova**, which is under development.

Thank you!

Extensions and applications

Other **extensions** of *EAR* could be:

1. the theory RDF^∞, whose set of formulas is the union of RDF^n formulas for all n;
2. decision methods regarding differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^m$.

Applications in automated theorem proving:

- proof verification of **mathematical theories**;
- **program verification** and **hardware validation**.

$RMCF^+$, RDF^* and RDF^n should be integrated in the **proof-checker ÆtnaNova**, which is under development.

Thank you!

Extensions and applications

Other **extensions** of *EAR* could be:

1. the theory RDF^∞, whose set of formulas is the union of RDF^n formulas for all n;
2. decision methods regarding differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^m$.

Applications in automated theorem proving:

- proof verification of **mathematical theories**;
- **program verification** and **hardware validation**.

$RMCF^+$, RDF^* and RDF^n should be integrated in the **proof-checker ÆtnaNova**, which is under development.

Thank you!

Extensions and applications

Other **extensions** of EAR could be:

1. the theory RDF^∞, whose set of formulas is the union of RDF^n formulas for all n;
2. decision methods regarding differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^m$.

Applications in automated theorem proving:

- proof verification of **mathematical theories**;
- **program verification** and **hardware validation**.

$RMCF^+$, RDF^* and RDF^n should be integrated in the **proof-checker ÆtnaNova**, which is under development.

Thank you!

Extensions and applications

Other **extensions** of *EAR* could be:

1. the theory RDF^∞, whose set of formulas is the union of RDF^n formulas for all n;
2. decision methods regarding differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^m$.

Applications in automated theorem proving:

- proof verification of **mathematical theories**;
- **program verification** and **hardware validation**.

$RMCF^+$, RDF^* and RDF^n should be integrated in the **proof-checker ÆtnaNova**, which is under development.

Thank you!