CEUR-WS.org/Vol-3072/paper13.pdf

A Decidable Theory Treating Addition of Differentiable Real Functions* **

Gabriele Buriola[0000−0002−1612−0985] 1,
Domenico Cantone[0000−0002−1306−1166] 2,
Gianluca Cincotti[0000−0001−8460−1708] 2,
Eugenio G. Omodeo[0000−0003−3917−1942] 3, and
Gaetano T. Spartà[0000−0002−8993−5851] 4

[1] Dept. of Computer Science, University of Verona, Italy
gabriele.buriola@univr.it
[2] Dept. of Mathematics and Computer Science, University of Catania, Italy
domenico.cantone@unict.it
[3] Dept. of Mathematics and Earth Sciences, University of Trieste, Italy
eomodeo@units.it
[4] Pontificia Università Gregoriana, Rome, Italy
g.sparta@unigre.it

Abstract. This paper enriches a pre-existing decision algorithm, which in its turn augmented a fragment of Tarski's elementary algebra with one-argument real functions endowed with continuous first derivative. In its present (still quantifier-free) version, our decidable language embodies addition of functions; the issue we address is the one of satisfiability.
As regards real numbers, individual variables and constructs designating the basic arithmetic operations are available, along with comparison relators. As regards functions, we have another sort of variables, out of which compound terms are formed by means of constructs designating addition and—outermostly—differentiation. An array of predicates designate various relationships between functions, as well as function properties, that may hold over intervals of the real line; those are: function comparisons, strict and non-strict monotonicity / convexity / concavity, comparisons between the derivative of a function and a real term.
With respect to results announced in earlier papers of the same stream, a significant effort went into designing the family of interpolating functions so that it could meet the new constraints stemming from the presence of function addition (along with differentiation) among the constructs of our fragment of mathematical analysis.

Key words: Decidable theories, Tarski's elementary algebra, Functions of a real variable.
MS Classification 2010: 03B25, 26A06.

---

* Copyright ©2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

** We gratefully acknowledge partial support from project "STORAGE—Università degli Studi di Catania, Piano della Ricerca 2020/2022, Linea di intervento 2", and from INdAM-GNCS 2019 and 2020 research funds.

Introduction

This paper addresses the decision problem for a fragment of real analysis which, besides the four operators '+', '−', '·', '/' of elementary real algebra, also provides predicates expressing strict and non-strict monotonicity, concavity, and convexity of C^1 functions of one real variable over bounded or unbounded intervals, as well as strict and non-strict comparisons '$>$' and '$\geqslant$' between real numbers and between functions. Further primitive constructs available in the language are: an operator designating pointwise addition of functions, and a differentiation operator whose usage must be reasonably restrained.[5] The language under study, named RDF^*, is devoid of quantifiers; we reduce the satisfiability problem regarding its formulas to the provability problem for purely existential sentences in Tarski's elementary algebra of real numbers, whose decidability is known since long (cf. [17]). We can thus count upon improved versions of Tarski's original method.

Our decision method consists in preprocessing the given formula into an equi-satisfiable quantifier-free formula of the elementary algebra of real numbers, whose satisfiability can then be checked by means of Tarski's decision method. No direct reference to functions will appear in the target formula, each function variable having been superseded by a collection of stub real variables; hence, in order to prove that the proposed translation is satisfiability-preserving, we must figure out a flexible-enough family of interpolating C^1 functions that can accommodate a model for the source formula whenever the target formula turns out to be satisfiable.

This paper is a sequel of [2] and [1]—hence, indirectly, of their antecedents [4,6]. As for semantics, the language $RMCF^+$ studied in [2] differs from the one treated here in that $RMCF^+$ refers to continuous functions whereas in RDF^* functions are also required to be endowed with continuous derivative: in consequence of this, as will be shown, a satisfiable $RMCF^+$ formula may cease to be satisfiable in RDF^*.[6] As for syntax, the sole significant difference between $RMCF^+$ and RDF^* is that the former did not provide—as we now do—the differentiation operator. A brief account of the decidability result proposed in [2], with examples of usage of its language, can be found in [15, pp.165–177].

Our present language RDF^* differs from the language RDF^+ studied in [1] in that its syntax is richer. We now have a construct for function addition, whose treatment calls for an enhancement of the decision algorithm, to wit, an enhanced reduction to Tarskian algebra. We thus get closer to a language where one can prove the linearity of differentiation; in fact, our next extension will permit multiplication of functions by numbers.

The paper is organized as follows. In Sec. 1 we introduce syntax and semantics of the language of interest, and illustrate its expressive power—partly stemming

[5] The differentiation operator $D[\cdot]$ can only appear as the lead operator in a function term $\mathfrak{g}$ (which will then coincide with $D[\mathfrak{f}]$). The rationale is that $D[\mathfrak{f}]$ might not designate a C^1 function when $\mathfrak{f}$ does; then, e.g., $D\big[\,D[\mathfrak{f}]+\mathfrak{f}\,\big]$ would be meaningless.

[6] On the positive side, since the universe of functions for $RMCF^+$ is richer, any formula judged valid by the decision algorithm for $RMCF^+$ is also valid in RDF^*.

from the ease with which useful derived constructs can be introduced—through a gallery of small examples. In Sec. 2, we describe our decision algorithm: since we cannot afford to specify it in detail, we exemplify its use by manually working out an emulation of how it would process a specific, valid formula. Then Sec. 3 provides clues on the correctness of the proposed decision algorithm. To end, we outline a comparison with related works, and draw conclusions.

1 The RDF^* theory

The augmented version RDF^* of the theory RDF of Reals with Differentiable Functions, is an unquantified first-order theory dealing with reals and with real functions of class C^1 of one real variable, namely functions with continuous first derivative. The function symbols of RDF^* designate the basic operations of real arithmetic, and pointwise addition and differentiation of functions. Its predicate symbols designate: comparisons between reals, pointwise comparisons of functions; strict and non-strict monotonicity, convexity, and concavity; comparisons between first derivatives and real terms. This section introduces the language underlying RDF^*, explains its intended meaning, and briefly illustrates its use.

Syntax and semantics

The language RDF^* has two infinite supplies of individual variables, belonging to the respective sorts: *numerical variables* $x, y, z, \ldots$ and *function variables* $f, g, h, \ldots$. Numerical and function variables are supposed to range, respectively, over the set $\mathbb{R}$ of real numbers and over the class $C^1(\mathbb{R})$ of continuous functions with continuous derivative [the collection of functions which interests us]. Four constants are also available:

- the symbols 0 and 1, designating the numbers 0 and 1;[7]
- the distinguished symbols $+\infty$ and $-\infty$, occurring only as ends of *interval specifications* (see below).

We next specify the syntax of *terms*, *atoms*, and *formulas* for RDF^*:

Definition 1. FUNCTION TERMS*,* NUMERICAL TERMS*, and* INTERVAL SPECS *are so defined:*

a.1) every function variable f is a function term*;*
a.2) if $\mathfrak{f}$ and $\mathfrak{g}$ are function terms, then $\mathfrak{f} + \mathfrak{g}$ is a function term*.*

b.1) Numerical variables and the constants $0, 1$ *are* numerical terms*;*
b.2) if s and t are numerical terms, the following also are numerical terms*:*

$$s + t\,,\; s - t\,,\; \text{and}\; s \cdot t\,;$$

[7] As will turn out, the constants $0, 1$ would be eliminable from our language without loss of expressive power, since $z = 0 \wedge u = 1$ is the sole solution to $u = u \cdot u > z \cdot z = z$.

b.3) if t is a numerical term and $\mathfrak{f}$ is a function term, then

$$\mathfrak{f}(t) \text{ and } D[\mathfrak{f}](t)$$

are numerical terms.[8]

c.1) An interval spec *A is an expression of any of the forms*

$$[e_1, e_2]\,,\ [e_1, e_2[\,,\ ]e_1, e_2]\,, \text{ and }]e_1, e_2[\,,$$

where e_1 stands for either a numerical term or $-\infty$, and e_2 for either a numerical term or $+\infty$;

c.2) we dub the "extended" numerical terms e_1, e_2 of such an A the ENDS *of A.* ⊣

Definition 2. *An* ATOM *of RDF^* is an expression of one of the forms*

$$\begin{array}{rrr}
s = t\,, & s > t\,, & \\
(\mathfrak{f} = \mathfrak{g})_A\,, & (\mathfrak{f} > \mathfrak{g})_A\,, & \\
\mathrm{Up}(\mathfrak{f})_A\,, & \mathrm{Strict_Up}(\mathfrak{f})_A\,, & \\
\mathrm{Down}(\mathfrak{f})_A\,, & \mathrm{Strict_Down}(\mathfrak{f})_A\,, & \\
\mathrm{Convex}(\mathfrak{f})_A\,, & \mathrm{Strict_Convex}(\mathfrak{f})_A\,, & \\
\mathrm{Concave}(\mathfrak{f})_A\,, & \mathrm{Strict_Concave}(\mathfrak{f})_A\,, & (D[\mathfrak{f}] \bowtie t)_A\,,
\end{array}$$

where $\bowtie \in \{=, <, >, \leqslant, \geqslant\}$ and A stands for an interval spec.
A FORMULA *of RDF^* is any truth-functional combination of RDF^* atoms.* ⊣

For definiteness, we will construct the RDF^* formulas from RDF^* atoms by means of the usual propositional connectives $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.

The semantics of RDF^* revolves around the designation rules listed in our next definition, with which any truth-value *assignment* for the formulas of RDF^* must comply.

Definition 3. *An* ASSIGNMENT *for RDF^* is a mapping M whose domain consists of all terms and formulas of RDF^*, satisfying the following conditions:*

0. $M0$ and $M1$ are the real numbers 0 and 1.
1. For each numerical variable x, Mx is a real number.
2. For each function variable f, (Mf) is an everywhere defined differentiable real function of one real variable, endowed with continuous derivative.
3. For each function term of the form $\mathfrak{f} + \mathfrak{g}$, the image $\big(M(\mathfrak{f} + \mathfrak{g})\big)(r)$ of any real number r is $(M\mathfrak{f})(r) + (M\mathfrak{g})(r)$.
4. For each numerical term of the form $t_1 \otimes t_2$ with $\otimes \in \{+, -, \cdot\}$, $M(t_1 \otimes t_2)$ is the real number $Mt_1 \otimes Mt_2$.

[8] Throughout, s, t and $\mathfrak{f}, \mathfrak{g}$ stand, respectively, for numerical terms and function terms while x, y, z and f, g, h stand, more specifically, for numerical *variables* and function *variables*.

5. *For each numerical term of the form* $\mathfrak{f}(t)$, $M(\mathfrak{f}(t))$ *is the real number* $(M\mathfrak{f})(Mt)$*; for each numerical term* $D[\mathfrak{f}](t)$, $M(D[\mathfrak{f}](t))$ *is the real number* $D[(M\mathfrak{f})](Mt)$.
6. *For each interval specification* A, MA *is an interval of* $\mathbb{R}$ *of the appropriate kind, whose endpoints are the evaluations via* M *of the ends of* A.[9]
For example, when $A =]t_1, t_2]$, *then* $MA =]Mt_1, Mt_2]$.
7. *Truth values are assigned to formulas of RDF* according to the following rules, where* s *and* t *stand for numerical terms and* $\mathfrak{f}, \mathfrak{g}$ *for function terms:*
   a) $s = t$ *(respectively* $s > t$*) is true iff* $Ms = Mt$ *(resp.* $Ms > Mt$*) holds;*
   b) $(\mathfrak{f} = \mathfrak{g})_A$ *is true iff* $(M\mathfrak{f})(x) = (M\mathfrak{g})(x)$ *holds for all* x *in* MA*;*
   c) $(\mathfrak{f} > \mathfrak{g})_A$ *is true iff* $(M\mathfrak{f})(x) > (M\mathfrak{g})(x)$ *holds for all* x *in* MA*;*
   d) $(D[\mathfrak{f}] \bowtie t)_A$, *with* $\bowtie \in \{=, <, >, \leqslant, \geqslant\}$, *is true iff* $D[(M\mathfrak{f})](x) \bowtie Mt$ *holds for all* x *in* MA*;*
   e) *Up*$(\mathfrak{f})_A$ *(respectively Strict_Up*$(\mathfrak{f})_A$*) is true iff* $(M\mathfrak{f})$ *is a monotone non-decreasing (resp. strictly increasing) function in* MA*;*
   f) *Convex*$(\mathfrak{f})_A$ *(respectively Strict_Convex*$(\mathfrak{f})_A$*) is true iff* $(M\mathfrak{f})$ *is a convex (resp. strictly convex) function in* MA*;*
   g) *the truth values of Down*$(\mathfrak{f})_A$, *Concave*$(\mathfrak{f})_A$, *Strict_Down*$(\mathfrak{f})_A$, *and Strict_Concave*$(\mathfrak{f})_A$ *are defined in close analogy with items* **e)** *and* **f)***;*
   h) *the truth value which* M *assigns to a formula whose lead symbol is any of* $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ *complies with the usual semantics of the propositional connectives.*

An assignment M *is said to* MODEL *a set* Φ *of formulas when* $M\varphi$ *is true for every* φ *in* Φ. ⊣

Definition 4 (Derived symbols). *In light of the above semantics, we tacitly enrich our language, much as in [1], with derived dyadic and triadic comparators involving numerical terms* t_1, t_2, *and* t_3*; namely* $t_1 \rhd t_2$ *and* $t_1 \bowtie t_2/t_3$, *where* $\rhd \in \{\neq, <, \leqslant, \geqslant\}$ *and* $\bowtie \in \{=, <, >, \leqslant, \geqslant\}$.

Additional relators intermixing function terms and numerical terms, e.g. the construct $(D[\mathfrak{f}] \neq t)_A$, *can also be introduced by means of shortening definitions. Among others, any function of the form* $x \mapsto q \cdot x + q'$, *with* q *and* q' *fixed rational numbers, can be characterized by means of a formula as in [1]; in particular, we may define* $h = \mathbf{0} \leftrightarrow_{Def} (h = h + h)_{]-\infty,+\infty[}$. *Thanks to the availability of function addition, one can now also specify the multiplication of a function* f *by any fixed rational number* $\pm\frac{m}{n}$*: in fact, for* n, m *positive integers, we can define:*

$$\begin{aligned}\left(g = \tfrac{m}{n} \cdot f\right)_{]-\infty,+\infty[} &\leftrightarrow_{Def} \overbrace{g + \cdots + g}^{n \text{ times}} = \overbrace{f + \cdots + f}^{m \text{ times}},\\ \left(g = -\tfrac{m}{n} \cdot f\right)_{]-\infty,+\infty[} &\leftrightarrow_{Def} \underbrace{g + \cdots + g}_{n \text{ times}} + \underbrace{f + \cdots + f}_{m \text{ times}} = \mathbf{0}.\end{aligned}$$ ⊣

[9] It goes without saying what is meant when M is undefined at either end of A (actually, $M(-\infty)$ and $M(+\infty)$ are undefined).

A gallery of examples heading to Cauchy's mean value theorem

In sight of becoming able to express the classical Cauchy's mean value theorem—which calls, alas, for a syntactical extension of RDF^*—, we formulate in RDF^* many basic facts of real analysis:

- ► $(D[f] = t)_{[a,b]} \quad \to \quad \mathrm{Linear}(f)_{[a,b]}$

  If f is a function with constant derivative in the interval $[a, b]$, then it will be linear in the same interval. With a slight abuse of notation we can improve the result as $(D[f] = \frac{f(b)-f(a)}{b-a})_{[a,b]} \leftrightarrow \mathrm{Linear}(f)_{[a,b]}$, where an equivalence holds when the derivative is equal to the difference quotient.

- ► $(D[f] > 0)_{]a,b[} \quad \to \quad \mathrm{Strict\_Up}(f)_{[a,b]}$

  If f is a real function endowed with continuous derivative f', and $f'(x) > 0$ holds for all $x \in\,]a, b[$, then f is monotone increasing in $[a, b]$.

- ► $\big[b > a \wedge (b - a) \cdot t = f(b) - f(a)\big] \quad \to \quad \neg\,(D[f] \neq t)_{]a,b[}$

  If f is a real function endowed with continuous derivative f', then to any interval $]a, b[$ with $a < b$ there belongs a c such that $f'(c) = \frac{f(b)-f(a)}{b-a}$ holds. This is a weak version of *Lagrange's mean value theorem.*

- ► $[\,\mathrm{Up}(f)_A \wedge \mathrm{Strict\_Up}(g)_A\,] \quad \to \quad \mathrm{Strict\_Up}(f + g)_A$

  If f and g are, over the interval A, respectively monotone non-decreasing and monotone increasing, their sum $f + g$ is monotone increasing on A.

- ► $[\,\mathrm{Up}(f)_A \wedge f + g = \mathbf{0}\,] \quad \to \quad \mathrm{Down}(g)_A$

  If f is an increasing function all over the interval A, its additive inverse g decreases over A.

- ► $\big[\big(D[f] = r\big)_A \wedge \big(D[g] = s\big)_A\big] \quad \to \quad \big(D[f + g] = r + s\big)_A$

  If the graphs of f and g, restricted to A, are straight lines with slopes r and s, then the graph of $f + g$ restricted to A is a straight line with slope $r + s$. Moreover, when the interval A reduces to a single point, this formula states the additive property of derivative over all $\mathbb{R}$.

- ► $[\,\mathrm{Convex}(f)_A \wedge \mathrm{Convex}(g)_A \wedge (h + h = f + g)_A\,] \quad \to \quad \mathrm{Convex}(h)_A$

  If f and g are two convex functions on A, also their average $h = \frac{f+g}{2}$ is convex on A.

- ► $[\,(f > k)_A \wedge (g > k)_A \wedge (h + h = f + g)_A\,] \quad \to \quad (h > k)_A$

  If f and g both exceed the function k on A, so does in its turn their average.

Every instance of the formula schemes shown before can be validated by means of the decision algorithm that will be presented. Actually, the first three could also be handled by the simpler decision algorithm tailored for RDF^+ (cf. [1]), since they do not involve sums of functions.

As announced in the Introduction, a satisfiable $RMCF^+$ formula may cease to be satisfiable in RDF^*. Such a formula is (see [2]):

$$\text{Convex}(f)_{[0,1]} \;\wedge\; \text{Convex}(f)_{[1,1+1]} \;\wedge\; f(0) = f(1+1) = 0 \;\wedge\; f(1) = 1 \;\wedge\; \text{Concave}(f)_{[0,1+1]}\,;$$

as a matter of fact, no real function satisfying its subformula

$$\text{Convex}(f)_{[0,1]} \wedge \text{Convex}(f)_{[1,1+1]} \wedge f(0) = f(1+1) = 0 \wedge f(1) = 1$$

belongs to the C^1 class.

2 Clues on the decision algorithm

Establishing that an RDF^* formula ψ is valid amounts to establishing that its negation $\neg\psi$ is not satisfiable; moreover, satisfying $\neg\psi$ amounts to satisfying one of the disjuncts of its disjunctive normal form, hence the key issue concerning the decidability of RDF^* is: how can we determine whether or not a given conjunction of RDF^* *literals* (that is, RDF^* atoms and negations thereof) is satisfiable? Via routinary flattening techniques and in view of some basic properties of C^1, we can restate each instance of this problem as the one of determining the satisfiability of an arbitrary conjunction φ_0 of atoms of the forms

$$\begin{array}{rrrr} z = x + y\,, & (h = f + g)_A\,, & \text{Strict\_Up}(f)_A\,, & \text{Convex}(f)_A\,, \\ z = x \cdot y\,, & (f = g)_A\,, & \text{Strict\_Down}(f)_A\,, & \text{Concave}(f)_A\,, \\ x > y\,, & (f > g)_A\,, & (D[f] \bowtie z)_A\,, & \text{Strict\_Convex}(f)_A\,, \\ & z = f(x)\,, & z = D[f](x)\,, & \text{Strict\_Concave}(f)_A \end{array}$$

and of literals which are the complements of atoms of these forms that involve an interval spec. Here A stands for an interval spec whose ends can be numerical variables, $-\infty$, or $+\infty$; as ever, x, y, z stand for numerical variables and f, g, h stand for function variables. Through a process furcating at various points, φ_0 will undergo a series $\varphi_0 \rightsquigarrow \varphi_1 \rightsquigarrow \varphi_2 \rightsquigarrow \varphi_3 \rightsquigarrow \varphi_4 = \widehat{\varphi}$ of transformations, ending in a formula $\widehat{\varphi}$ where function variables do no longer occur, so that $\widehat{\varphi}$ can be tested for satisfiability by means of Tarski's celebrated decision algorithm [17,7]. The proper functioning of this method relies on certain assumptions about the detailed structure of φ, easy to ensure, which we must fly over.

The transformations $\varphi_{i-1} \rightsquigarrow \varphi_i$ $(i = 1, 2, 3, 4)$ serve the following purposes:

1. Subdivide into cases each literal of the form $(f > g)_A$ whose A is *not* of the form $[v, w]$. E.g., $(f > g)_{]v,w]}$ offers two choices: $f(v) > g(v)$, $f(v) = g(v)$.
2. Substitute every negative literal with an implicit existential assertion. E.g., $\neg\text{Strict\_Up}(f)_{[v,w]}$ will bring into play new variables x, y, x', y' subject to constraints $v \leqslant x < x' \leqslant w \wedge y = f(x) \wedge y' = f(x') \wedge y \geqslant y'$.
3. With certain salient variables v_j in the domains of the functions designated by the function variables in φ_0, associate new variables y_j^f, t_j^f (one for each function variable f) subject to the constraints $y_j^f = f(v_j)$, $t_j^f = D[f](v_j)$.

4. Get rid of all literals involving function variables, whose graphs are already outlined by the variables y_j^f, t_j^f introduced above. This elimination phase calls for the introduction of new variables subject to suitable algebraic constraints.

The decision algorithm at work

Our decision algorithm for RDF^* cannot be specified in full in these few pages; to convey a feel of how it works, we consider a paradigmatic formula ψ, and carry out one by one the key transformations leading from ψ to a formula directly submittable to Tarski's algorithm for elementary real algebra.

Suppose that we want to establish whether the formula ψ,

$$\Big[\big(D[f]=r\big)_{[a,b]} \wedge \big(D[g]=s\big)_{[a,b]}\Big] \;\rightarrow\; \big(D[f+g]=r+s\big)_{[a,b]},$$

is true under every value assignment; equivalently, we can check whether its negation $\neg\psi$ is unsatisfiable. After introduction of convenient stub variables h and p, this negation becomes the following formula φ:

$$\big(D[f]=r\big)_{[a,b]} \wedge \big(D[g]=s\big)_{[a,b]} \wedge \big(h=f+g\big)_{[a,b]} \wedge \neg\big(D[h]=p\big)_{[a,b]} \wedge p=r+s\,.$$

Then φ undergoes the following transformations:

1. Behavior at the ends: Generally speaking, function-comparison literals of the form $(f>g)_A$ must be bestowed special care, possibly leading to a subcase analysis. Since no such literal appears in our φ, this phase produces $\varphi_1 \coloneqq \varphi$.
2. Negative clause removal: This phase removes the negative literal and obtains φ_2 from φ_1 by substituting the conjunction $a \leqslant x \leqslant b \wedge y = D[h](x) \wedge y \neq p$ for $\neg\big(D[h]=p\big)_{[a,b]}$ inside it.
3. Explicit evaluation of function variables: This phase introduces a new variable to designate each function-application term $\ell(v)$, where ℓ stands for a function variable of φ and v for one of its so-called 'domain' variables. More precisely, for each function variable f and each domain variable a, we introduce two new numerical variables y_a^f, t_a^f and two literals $f(a)=y_a^f, D[f](a)=t_a^f$ evaluating, respectively, the function f and its derivative f' in a.
   To describe evaluation more transparently, let us do the renaming: $a \rightsquigarrow v_1,\ x \rightsquigarrow v_2,\ b \rightsquigarrow v_3$. From the previous formula φ_2, we get the following φ_3:

$$\begin{array}{rcccccl}
\big(D[f]=r\big)_{[v_1,v_3]} & \wedge & \big(D[g]=s\big)_{[v_1,v_3]} & \wedge & \big(h=f+g\big)_{[v_1,v_3]} & \wedge & \\
v_1 \leqslant v_2 \leqslant v_3 & \wedge & D[h](v_2)=y & \wedge & p=r+s & \wedge & y\neq p \;\wedge \\
h(v_1)=y_1^h & \wedge & h(v_2)=y_2^h & \wedge & h(v_3)=y_3^h & \wedge & \\
D[h](v_1)=t_1^h & \wedge & D[h](v_2)=t_2^h & \wedge & D[h](v_3)=t_3^h & \wedge & y=t_2^h \;\wedge \\
f(v_1)=y_1^f & \wedge & f(v_2)=y_2^f & \wedge & f(v_3)=y_3^f & \wedge & \\
D[f](v_1)=t_1^f & \wedge & D[f](v_2)=t_2^f & \wedge & D[f](v_3)=t_3^f & \wedge & \\
g(v_1)=y_1^g & \wedge & g(v_2)=y_2^g & \wedge & g(v_3)=y_3^g & \wedge & \\
D[g](v_1)=t_1^g & \wedge & D[g](v_2)=t_2^g & \wedge & D[g](v_3)=t_3^g. & &
\end{array}$$

4. Elimination of function variables: This final phase removes all literals still containing function variables. We get rid of them by suitable replacements involving algebraic conditions, such as the difference quotient for literals regarding derivatives. For example, the literal $\big(D[f]=r\big)_{[v_1,v_3]}$ becomes the conjunction $t_1^f=r \wedge t_2^f=r \wedge t_3^f=r \wedge \frac{y_2^f-y_1^f}{v_2-v_1}=r \wedge \frac{y_3^f-y_2^f}{v_3-v_2}=r$, where the t_i^f's represent salient values of the derivative and $\frac{y_{i+1}^f-y_i^f}{v_{i+1}-v_i}=r$ the corresponding difference quotients. To make the manner we eliminate the "function literals" clearer we mark with the same color the two literals $\big(D[f]=r\big)_{[v_1,v_3]}$, $\big(h=f+g\big)_{[v_1,v_3]}$ and their transformations.
At the end we obtain an equisatisfiable Tarskian formula we can then test for satisfiability by Tarski's algorithm.
From the previous formula φ_3 we get the following final formula φ_4:

$$\begin{array}{lllll}
v_1 \leqslant v_2 \leqslant v_3 \wedge & p=r+s \quad \wedge & y \neq p \quad \wedge & y=t_2^h \quad \wedge & \\
t_1^f=r \wedge & t_2^f=r \quad \wedge & t_3^f=r \quad \wedge & \frac{y_2^f-y_1^f}{v_2-v_1}=r \quad \wedge & \frac{y_3^f-y_2^f}{v_3-v_2}=r \wedge \\
t_1^g=s \wedge & t_2^g=s \quad \wedge & t_3^g=s \quad \wedge & \frac{y_2^g-y_1^g}{v_2-v_1}=s \quad \wedge & \frac{y_3^g-y_2^g}{v_3-v_2}=s \wedge \\
 & y_1^h=y_1^f+y_1^g \quad \wedge & y_2^h=y_2^f+y_2^g \quad \wedge & y_3^h=y_3^f+y_3^g \quad \wedge & \\
 & t_1^h=t_1^f+t_1^g \quad \wedge & t_2^h=t_2^f+t_2^g \quad \wedge & t_3^h=t_3^f+t_3^g. &
\end{array}$$

In particular, the unsatisfiability of this last formula is given by the conjunction:

$$y=t_2^h \wedge t_2^h=t_2^f+t_2^g \wedge t_2^f=r \wedge t_2^g=s \wedge p=r+s \wedge y \neq p.$$

3 Correctness of the algorithm

In order to prove the correctness of the algorithm, it is enough to show that each one of the (*terminating*) transformations $\varphi \rightsquigarrow \varphi_1$, $\varphi_1 \rightsquigarrow \varphi_2$, $\varphi_2 \rightsquigarrow \varphi_3$, $\varphi_3 \rightsquigarrow \varphi_4$ is *satisfiability preserving*. Regarding the first three of them (behavior at the endpoints, negative-clause removal, explicit evaluation of function variables), this emerges as a rather straightforward fact.

We must focus on the equisatisfiability of the formulas φ_3 and φ_4, because the transformation $\varphi_3 \rightsquigarrow \varphi_4$ is less transparent than the previous ones: we are, in fact, comparing a formula whose predicates regard the behavior of functions in real intervals with another one which only involves relations between numerical variables. Let us sketch the idea behind the proof (the full proof is unaffordably long for a conference paper and relies on various propositions of elementary real analysis). As usual, the proof consists of two parts: *soundness* and *completeness*. Recall that φ_4 is obtained from φ_3 by adding some formulas that involve only numerical variables, and by removing all predicates which refer to function variables.

Soundness: If a model exists for φ_3, it can be extended to a model that also verifies the numerical formulas added in φ_4, since these formulas reflect properties of the functions in φ_3 at specific points of real intervals.

Completeness: Conversely, if there exists a model for φ_4, it is possible to extend it to φ_3 by interpreting the function variables with suitable interpolating functions. In the proof, to perform the interpolation we resort to a family of functions, closed under addition. Each function is constructed by combination of pieces, each of which results from a linear component affected by a so-called 'perturbation'. In its turn, each perturbation stems from a parabola. On the one hand, we use the linear components to satisfy pointwise properties, such as $f(v_i) = y_i^f$; on the other hand, we use parabolic perturbations to satisfy constraints on derivatives, such as $D[f](v_i) = t_i^f$. Closure under addition of the family of interpolating functions ensures proper treatment of literals involving the sum of functions: actually, if $(h = f + g)_A$ is one such literal and F, G, H are the functions interpreting the symbols f, g, h, we must require $H = F + G$. *This non trivial condition forced us to shift from the original approach presented in [1] (where the perturbations were based on exponential functions) to the present one,* in which the perturbation of each interpolating segment originates from the envelope of two straight lines in the interval $\left[0, \frac{1}{2}\right]$. These novel interpolating functions, and their derivatives, comply with the properties at points and on real intervals dictated by φ_3.

Sample function resulting from perturbated linear segments

To shed light upon the nature of these interpolating functions, here we give a very simple example. Suppose that we must cope, at the end of the decision algorithm, with the following quantifier-free formula of the elementary algebra of real numbers:

$$v_1 = 0 \wedge v_2 = 1 \wedge y_1^f = 0 \wedge y_2^f = 3 \wedge t_1^f = 1 \wedge t_2^f = 0\,.$$

Driven by this satisfiable formula, we want to construct an appropriate C^1 function f. The interpolation constraints which f must comply with are pointwise conditions, two of which regard its derivative:

$$f(0) = 0, \quad f(1) = 3, \quad f'(0) = 1, \quad f'(1) = 0\,.$$

We can meet the first two with a linear segment, namely $r(x) = 3\,x$, but in order to satisfy the conditions on the derivative we need a suitable perturbation of this linear segment. The basic elements out of which we build our perturbations are envelope functions defined on the interval $\left[0, \frac{1}{2}\right]$, of the form:

$$\begin{aligned} G[k,\theta_1,\theta_2](x) \coloneqq \tfrac{1}{(1-4k)^2} \Big\{ &(1-4k)[\theta_2 - 2k\,(\theta_1+\theta_2)]x + \\ &-2k(\theta_1-\theta_2)(1-2k)\Big(2k - \sqrt{2}\sqrt{2k^2+(1-4k)\,x}\Big)\Big\}\,, \end{aligned}$$

where k is a shrinking parameter involved in the completeness proof and θ_1, θ_2 represent the values of the derivative at endpoints. Starting with these basic elements and through reflections, translations and dilations, we can define all required perturbations.

In the example we are considering, after entering the right values for the derivative, we obtain the function

$$f(x) := \begin{cases} 3x + \frac{9}{4}x + \frac{17}{32}\left(1 - \sqrt{1+16x}\right) & \text{in } \left[0, \frac{1}{2}\right] \\ 3x + \frac{11}{4}(x-1) + \frac{23}{32}(\sqrt{17-16x} - 1) & \text{in } \left]\frac{1}{2}, 1\right], \end{cases}$$

where $3\,x$ is the linear part and the rest specifies the parabolic perturbation. The graph of this function (whose first half falls directly under $G[k, \theta_1, \theta_2](x)$, while the second half results from that scheme through appropriate manipulations) can be seen below.

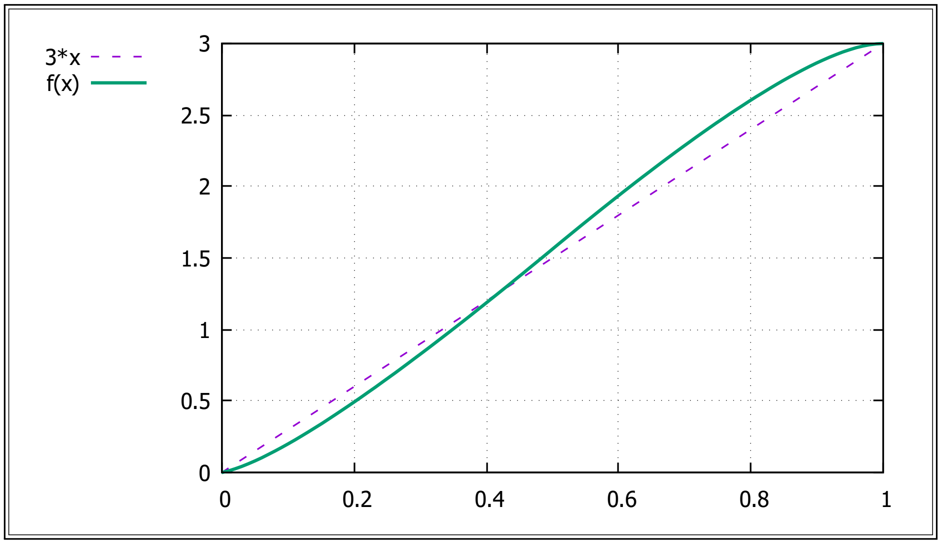

Fig. 1. Graph of the interpolating function with $f(0)=0$, $f(1)=3$, $f'(0)=1$, $f'(1)=0$.

4 Related works and complexity issues

The decidability of the theory RDF^* treated above is a follow-up of a series of previous results, regarding the theories $RMCF$, $RMCF^+$, RDF, and RDF^+ [4,2,6,3,1]. A general survey on those results, save the last, can be found in [5], where other decidability results on real analysis are also treated, in particular the FS theory [10,11].

Since the decidability of RDF^* is obtained via an explicit algorithm, some complexity issues are worth being discussed here. Tarski's decision method enters into ours, hence our algorithm inherits its complexity as a lower bound. The first complexity amelioration w.r.t. Tarski's historical result is due to Collins

[7], whose procedure has doubly exponential complexity relative to the number of variables occurring in the sentence (or just exponential, if the endowment of variables is finite and fixed). A refinement of this result was achieved with Grigoriev's algorithm [12], applicable to sentences in prenex normal form, whose complexity is doubly exponential relative to the number of quantifier alternations. If we merely focus on the *existential* theory of reals, the known decision algorithms have a complexity at best exponential relative to the number n of variables [9]; however, if one fixes beforehand how many variables can be used, then the algorithmic complexity becomes polynomial [13].

Finally, while Tarski himself showed that decidability of his full elementary algebra of real numbers [17] would be disrupted if the language were enriched with certain real functions, in particular $sin\ x$, Richardson proved in [14] the undecidability of the existential theory of reals extended with the numbers $\log 2$ and π, and with the functions e^x, $sin\ x$.

5 Conclusions and Future Works

This article has presented a decision algorithm for a syntactically delimited fragment, RDF^*, of real analysis. RDF^* extends the unquantified part of Tarski's elementary algebra EAR of real numbers with variables designating functions of a real variable endowed with a continuous derivative. After showing how to add derived relators and how to specify common properties of real functions and theorems about them in RDF^*, we have discussed an algorithm that translates a generic formula φ of RDF^* into an equisatisfiable formula ψ of EAR; rather than specifying the algorithm in gory detail, we have illustrated its functioning through a concrete example. Proving the correctness of the decision algorithm amounts to showing the equisatisfiability of formulas φ and ψ, and we have sketched the salient points of this correctness proof. The proof relies upon the construction of a set of C^1 functions rich enough to enable the modeling of any satisfiable RDF^* formula: each function results from smoothly connecting linear segments perturbated by means of parabolic deformations.
While working on the algorithm we glimpse further extensions. The nearest two amount to a *product operator* and *multi-variable functions*. The product operator concerns the assembly of new function terms by multiply function terms and numerical terms. This new construct will enable us to treat literals of the form $\left(\mathfrak{f} = t\mathfrak{g}\right)_A$ and paves the way to state stronger analytic theorems, such as Cauchy's mean value theorem. Whereas this "syntactical scalar product" seems easily treatable, combining product of two function terms, such as $\mathfrak{h} = \mathfrak{f} \cdot \mathfrak{g}$ with function sum is likely to disrupt decidability (much as Presburger's vs. Peano's arithmetic [9, p.4]), or required at least deep changes in the algorithm. By multi-variable functions we mean the possibility to treat continuous real functions with multiple arguments, such as $f : \mathbb{R}^n \rightarrow \mathbb{R}$. A similar enrichment has already been introduced for the $RMCF$ in [4].

References

1. Buriola G., Cantone D., Cincotti G., Omodeo E.G., Spartà G.T., A decidable theory of differentiable functions with convexities and concavities on real intervals. In Francesco Calimeri, Simona Perri, and Ester Zumpano, editors, *Proceedings of the 35th Italian Conference on Computational Logic - CILC 2020, Rende, Italy, October 13-15, 2020*, volume 2710 of *CEUR Workshop Proceedings*, pages 231–247. CEUR-WS.org, 2020.
2. Cantone D., Cincotti G., Gallo G., *Decision algorithms for fragments of real analysis. I. Continuous functions with strict convexity and concavity predicates*. J. Symb. Comput., 41(7):763-789, 2006.
3. Cantone D., Cincotti G., *Decision algorithms for fragments of real analysis. II. A theory of differentiable functions with convexity and concavity predicates*. List of CILC 2007 papers, 14 pp., 2007. `https://www.programmazionelogica.it/wp-content/uploads/2014/10/cilc2007.pdf`
4. Cantone D., Ferro A., Omodeo E.G., Schwartz J.T., *Decision algorithms for some fragments of analysis and related areas*. Comm. Pure Appl. Math., 40(3):281-300, 1987.
5. Cantone D., Omodeo E.G., Spartà G.T., *Solvable (and unsolvable) cases of the decision problem for fragments of analysis*. Rend. Istit. Mat. Univ. Trieste, 44:313-348, 2012.
6. Cincotti G., Decision algorithms for fragments of real analysis and graph theory, Ph.D. Thesis, Università degli Studi di Catania, Catania, Italy, ix+136 pp., 2000.
7. Collins G., *Quantifier elimination for real closed fields by cylindrical algebraic decomposition*. In: Second GI Conference on Automata Theory and Formal Languages, LNCS Vol.33, Springer-Verlag, Berlin, 1975.
8. Ferro A., Omodeo E.G., Schwartz J.T., *Decision procedures for elementary sublanguages of set theory. I. Multi-level syllogistic and some extensions*, Comm. Pure Appl. Math, 33(5):599–608, 1980.
9. Fisher M. J. and Rabin M. O., *Super-exponential complexity of Presburger arithmetic*. Complexity and Computation, Vol. VII, SIAM-AMS, Philadelphia (1974), 27–41.
10. Friedman H. and Seress Á., *Decidability in elementary analysis. I*, Adv. Math. 76 (1989), no. 1, 94–115.
11. Friedman H. and Seress Á., *Decidability in elementary analysis. II*, Adv. Math. 79 (1990), no. 1, 1–17.
12. Grigoriev D., *Complexity of deciding Tarski algebra,* J. Symbolic Comput. 5 (1988), 65–108.
13. Renegar J., *A faster PSPACE algorithm for deciding the existential theory of the reals*, 29th Annual Symposium on Foundations of Computer Science (FOCS 1988, Los Angeles, Ca., USA), IEEE Computer Society Press, Los Alamitos (1988), pp. 291–295.
14. Richardson D., *Some undecidable problems involving elementary functions of a real variable*, J. Symbolic Logic 33 (1968), 514–520.
15. Schwartz JT, Cantone D, Omodeo EG. Computational logic and set theory: Applying formalized logic to analysis. Springer-Verlag, 2011. ISBN 978-0-85729-807-2. https://doi.org/10.1007/978-0-85729-808-9. Foreword by M. Davis.
16. Tarski A., *The completeness of elementary algebra and geometry*, Institut Blaise Pascal, Paris, 1967, iv+50 pp. (Late publication of a paper which had been submitted for publication in 1940)

17. Tarski A., *A decision method for elementary algebra and geometry*, prepared for publication by J.C.C McKinsey, U.S Air Force Project RAND, R-109, the RAND Corporation, Santa Monica, 1948, iv+60 pp.; a second, revised edition was published by the University of California Press, Berkeley and Los Angeles, CA, 1951, iii+63 pp.