Quasi-decidability of a Fragment of the First-order Theory of Real Numbers*

PETER FRANEK and STEFAN RATSCHAN†
Institute of Computer Science, Academy of Sciences of the Czech Republic
and
PIOTR ZGLICZYNSKI
Jagellonian University in Krakow

August 15, 2016

Abstract

In this paper we consider a fragment of the first-order theory of the real numbers that includes systems of n equations in n variables, and for which all functions are computable in the sense that it is possible to compute arbitrarily close interval approximations. Even though this fragment is undecidable, we prove that—under the additional assumption of bounded domains—there is a (possibly non-terminating) algorithm for checking satisfiability such that (1) whenever it terminates, it computes a correct answer, and (2) it always terminates when the input is robust. A formula is robust, if its satisfiability does not change under small continuous perturbations. We also prove that it is not possible to generalize this result to the full first-order language— removing the restriction on the number of equations versus number of variables. As a basic tool for our algorithm we use the notion of degree from the field of topology.

1 Introduction

It is well known that, while the theory of real numbers with addition and multiplication is decidable [42], any periodic function makes the problem undecidable, since it allows encoding of the integers. The root existence problem for uni-variate functions defined by addition, multiplication, the

---

sine function and the constant π is also undecidable [43]. This even holds if we consider only functions on bounded domains, because an algorithm deciding it could be used to compute a fixed point of a continuous function from a ball to itself which is known to be non-computable for some computable functions [4, 33].

Recently, several papers [19, 35, 37, 13] have argued, that in continuous domains (where we have notions of neighborhood, perturbation etc.) such undecidability results do not always have much practical relevance. The reason is, that real-world manifestations of abstract mathematical objects in such domains will always be exposed to perturbations (imprecision of production, engineering approximations, unpredictable influences of the environment etc.). Engineers take these perturbations into account by coming up with *robust* designs, that is, designs that do not change essentially under such perturbations. Hence, in this context, it is sufficient to come up with algorithms that are able to decide such robust problem instances. They are allowed to run forever in non-robust cases, but *must not* return incorrect results, in whatever case. In a recent paper we called problems possessing such an algorithm *quasi-decidable* [38].

The main contribution of this paper can be summarized as follows:

- We show quasi-decidability of a certain fragment of the first-order theory of the reals (Theorem 1). The basic building blocks are existentially quantified disjunctions of systems of n equalities over at most n variables and arbitrarily many inequalities. Those blocks may be combined using universal quantifiers, conjunctions, and disjunctions. All variables are assumed to range over closed and bounded intervals.

- We show that the result cannot be extended to the full first-order language. More specifically, in the basic building blocks (systems of equalities and inequalities) it is impossible to remove the restriction that the number of variables has to be at most the number of equalities (Theorem 2). Still, while we show that this restriction cannot be removed completely, this leaves open the possibility to replace the restriction by a weaker constraint on the number of variables and equations.

The allowed function symbols include addition, multiplication, exponentiation, and sine. More specifically, they have to be continuous, and for compact intervals $I_1, \ldots, I_n$, we need to be able to compute an interval $J \supseteq f(I_1 \times \cdots \times I_n)$ such that the over-approximation of J over $f(I_1 \times \ldots \times I_n)$ can be made arbitrarily small.

The main tool we use is the notion of the *degree of a continuous function* that comes from differential topology. For continuous functions $f : [a, b] \to \mathbb{R}$, the degree $\deg(f, [a, b], 0)$ is 0 iff $f(a)$ and $f(b)$ have the same sign, otherwise the degree is either 1 or -1, depending on whether the sign changes

from negative to positive or the other way round. If f is continuous and the degree is nonzero, then the equation $f(x) = 0$ has a solution by the intermediate value theorem. For higher dimensional functions, the degree is a computable [1, 17] integer whose value may be greater than 1, and a nonzero degree still indicates the existence of a root of f. The converse is not true and the existence of a root does not imply nonzero degree in general. We show how, for robustly satisfiable formulas built up from certain blocks of n equations in n variables, to make the degree test eventually succeed, while at the same time handling inequalities and logical symbols.

The proof of our second contribution—the class of equations and inequalities with no relation between the number of equations and variables is *not* quasi-decidable—is based on a reduction from a recent undecidability result [16] for a related robust satisfiability problem, cited in Theorem 10.

Even though this work applies results from a quite distant field—topology—to automated reasoning, the paper is largely self-contained. Usage of results from topology that are not explicitly delineated in this paper is concentrated exclusively in Section 6.

The content of the paper is as follows: In Section 2, we define the notions of robustness and quasi-decidability, and state the two main theorems of the paper. In Section 3, we provide the quasi-decision procedure whose existence is claimed by the first main theorem. In Section 4, we present the notion of topological degree and describe its main properties. In Section 5, we show that the quasi-decision procedure always returns a correct result. In Section 6 we show some non-algorithmic properties of the degree that will be the essential for showing termination for robust inputs in Section 7. In Section 8 we prove the second main theorem. In Section 9 we discuss related work. Finally, in Section 10, we conclude the paper.

2 Statement of the Results

We will start this section with informal discussion of a motivating example. Consider the first-order predicate logic formula

$$\exists x \mathbin{.} [x \geq -1 \wedge x \leq 1 \wedge \sin x = 0]$$

with the usual interpretation over the real numbers. This formula is true, and remains true, even if it is perturbed a little bit. On the other hand, the formula

$$\exists x \mathbin{.} [x \geq 1 \wedge x \leq 2 \wedge \sin x = 1]$$

is also true, but does not remain true when perturbing it, for example by increasing the right-most number 1 a little bit. We will later call formulas of the first type robust, and formulas of the second type non-robust. Our first theorem will state that, for a certain class of formulas over the reals that

includes function symbols such as sin, there exists an algorithm (a "quasi-decision procedure") that decides whether a given formula is true, but that is only required to terminate for robust inputs while it may run forever for non-robust inputs.

In the rest of the section, after fixing notation, we define the class of functions that we consider (Definition 1). Then we will formalize the notion of perturbing predicate-logical formulas (Definition 2) which results in a precisely defined notion of a formula being robust (Definition 3). Finally, we state Theorem 1 that ensures the existence of such a quasi-decision procedure and the negative Theorem 2 that puts a limit on generalization of the approach.

We define a *box* in $\mathbb{R}^n$ (or also n-box) to be the Cartesian product of n closed intervals of finite length (i.e., a hyper-rectangle). The *width* width(B) of a box B is the maximum of the width of the constituting intervals of B. For $x \in \mathbb{R}^n$, $|x|$ will refer to its *maximum norm* $|x| := \max\{|x_1|, \ldots, |x_n|\}$ and for a continuous function $f : \Omega \to \mathbb{R}^n$, we use the supremum norm $||f||_\Omega := \sup\{|f(x)|; \ x \in \Omega\}$. If $||f - g||_\Omega \leq \alpha$ for some $\alpha > 0$, we say that g is an α-*perturbation* of f in Ω. If Ω is clear from the context then we will simply write $||f||$, or say that g is an α-*perturbation* of f, without explicitly mentioning Ω. For a set $\Omega \subseteq \mathbb{R}^n$, $\bar{\Omega}$ is its closure, Ω° its interior and $\partial\Omega = \bar{\Omega}\backslash\Omega^\circ$ its boundary with respect to the Euclidean topology. We will call the closure $\bar{\Omega}$ of an open connected bounded set Ω a *closed region*.

For defining the class of formulas, we will first fix the class of functions that we handle. Intuitively, we allow functions whose range can be arbitrarly closely approximated by boxes:

Definition 1 *Let $\Omega \subseteq \mathbb{R}^m$ be a box with rational vertices. We say that a function $f : \Omega \to \mathbb{R}^n$ is* interval computable, *iff there exists a corresponding algorithm $\mathcal{I}(f)$ that computes, for any box $B \subseteq \Omega$ with rational vertices, an n-box $\mathcal{I}(f)(B) \subseteq \mathbb{R}^n$ with rational vertices such that*

- $\mathcal{I}(f)(B) \supseteq \{f(x) \mid x \in B\}$, *and*

- *for every $\varepsilon > 0$ there is a $\delta > 0$ such that for every box B with $0 < \text{width}(B) < \delta$, $\text{width}(\mathcal{I}(f)(B)) < \varepsilon$.*

Each interval computable function is uniformly continuous. Moreover, a function $f : \Omega \to \mathbb{R}^n$, with $\Omega \subseteq \mathbb{R}^m$ a box with rational vertices, is interval computable iff it is computable in the sense of computable analysis [8] (for seeing this, note especially that a function that is computable in the sense of computable analysis has a computable modulus of continuity [27, Theorem 2.13]).

For common function symbols that can be written in terms of symbolic expressions containing symbols denoting rational constants, the constant π, addition, multiplication, exponentiation, trigonometric functions and square

root, the algorithm $\mathcal{I}(f)$ can be implemented from the expression by interval arithmetic [30, 29] with arbitrary precision interval endpoints.

In the rest of the paper, we assume that a set of function and predicate symbols is given, together with structure assigning to each function symbol an interval computable function and to each predicate symbol a corresponding relation over the real numbers. We assume that this symbol set contains at least all rational constants, addition, multiplication, and the predicate symbols $=$ and $\geq$ with their usual interpretation. Whenever we will write concrete function or predicate symbols, this structure will assign their standard meaning over the real numbers. From now on, we will restrict ourselves to formulas from the first-order language corresponding to the given symbol set.

We also assume that a map $\mathcal{I}$ is given that assigns, to each function symbol f, an algorithm $\mathcal{I}(f)$ satisfying the specification in Definition 1. This map $\mathcal{I}$ is assumed to be algorithmic. Such assignment $\mathcal{I}$ naturally extends to terms of the language via composition of interval functions: if t is a term of the language, then the algorithm $\mathcal{I}(t)$ represents the corresponding function and satisfies both assumptions of Definition 1. In addition, we will assume that every variable ranges over a closed bounded interval introduced by a corresponding quantifier of the form $\exists x \in I$ or $\forall x \in I$. Throughout the paper we will require those bounds to be small enough to avoid any function application outside of the domain of any interval computable function. In a similar way, whenever we introduce bounds on the free variables of a formula, we assume them to be small enough to avoid such function applications.

As usual, a sentence will refer to a formula without free variables. Now we formalize perturbations of formulas by defining some notion of distance on sentences.

Definition 2 *Let F, G be two sentences. We say that F and G have the same structure iff one can be obtained from the other by only exchanging terms (i.e., they have the same Boolean and quantification structure including bounds of quantified variables, and the same predicate symbols).*

We define the distance d on sentences as follows. If two sentences F and G do not have the same structure, then $d(F, G) := \infty$. In the case where they do have the same structure, assume that the sentence F contains terms denoting functions $f_1, \ldots, f_p$ and the sentence G contains in the corresponding places terms denoting the functions $g_1, \ldots, g_p$. We define the distance

$$d(F, G) := \max_{i \in \{1, \ldots, p\}} ||f_i - g_i||_{\Omega_i},$$

where Ω_i denotes the respective domain of those functions, that is, the box defined by the quantification of all the variables.

For example, the sentences

$$\exists x \in [0, 1] \, \forall y \in [0, 1] \, . \, x^2 - y = xy \wedge x = y$$

and
$$\exists x \in [0,1] \, \forall y \in [0,1] \, . \, x^2 - y = xy + 1 \wedge x = y^2$$
have the same structure, because the only difference is in the terms involved. The distance $d(F, G) = 1$, because—with $(x, y) \in [0, 1]^2$—we have that $\max |(x^2 - y) - (x^2 - y)| = 0$, $\max |xy - (xy + 1)| = 1$, $\max |x - x| = 0$ and $\max |y - y^2| = 1/4$. As another example, the sentences $1 \geq 0$ and $\neg\neg 1 \geq 0$ do not have the same structure, and hence their distance is ∞.

Definition 3 *Let S be a sentence and $\varepsilon > 0$. We say that S is ε-robust iff for every sentence S', $d(S', S) < \varepsilon$ implies that S' and S have the same truth value. We say that the sentence S is* robust *iff there is an $\varepsilon > 0$ such that S is ε-robust. We say that a sentence S is* robustly true *iff it is both robust and true. We say that a sentence S is* robustly false *iff it is both robust and false.*

Note that, since we restricted ourselves to formulas with function symbols denoting interval-computable functions, all functions involved in the above definitions are interval computable, hence uniformly continuous.

Also note that equivalence of two formulas does not necessarily imply the same robustness. For example, the formula $\exists x \in [0, 2] \, . \, x - 1 = 0$ is robust, but the formula $\exists x \in [0, 2] \, . \, x - 1 = 0 \wedge x - 1 = 0$ is not, since both occurrences of the function $x - 1$ can be perturbed independently.

Definition 4 *A* quasi-decision procedure *for some class $\mathcal{B}$ of formulas is an algorithm that takes as inputs a sentence φ from $\mathcal{B}$ and an algorithm $\mathcal{I}$ converting function symbols f to algorithms $\mathcal{I}(f)$. The algorithm computes the truth value of φ whenever φ is robust. If φ is non-robust, the algorithm may run forever but must not return an incorrect result.*

If such a quasi-decision procedure exists for some class $\mathcal{B}$, then we say that $\mathcal{B}$ is quasi-decidable.

Now we are ready to state our first result.

Theorem 1 *The following class of formulas $\mathcal{B}$, defined recursively below, is quasi-decidable:*

(a) $\mathcal{B}$ contains all formulas of the form

$$\exists x \in B \, . \quad [f_1 = 0 \wedge f_2 = 0 \wedge \ldots \wedge f_n = 0 \wedge g_1 \geq 0 \wedge g_2 \geq 0 \wedge \ldots \wedge g_k \geq 0]$$

where $f_1, \ldots, f_n, g_1, \ldots, g_k$ are terms denoting interval-computable functions, B is an m-box (the expression $\exists x \in B$ denoting a block of m existential quantifiers) with rational vertices and either $n \geq m$ or $n = 0$. The integer k may be arbitrary and we also admit $k = 0$ (i.e., the case without inequalities).

6

(b) Let $I \subseteq \mathbb{R}$ be a closed bounded interval with rational endpoints. If U is in $\mathcal{B}$, then

$$\forall x \in I \,.\, U$$

is also in $\mathcal{B}$.

(c) If U, V are in $\mathcal{B}$, then

$$U \wedge V \qquad U \vee V$$

are also in $\mathcal{B}$.

The formulas corresponding to (a) represent systems of equations and inequalities. However, we assume that there are no more existential quantifiers than equations in (a), corresponding to the condition $n \geq m$.

The following sentence is an example of a formula in class $\mathcal{B}$:

$$\forall x \in [-1, 1]$$
$$\exists y \in [-1, 1] \; \exists z \in [-1, 1]$$
$$[x^2 - y^2 - z^2 = 0 \;\wedge\; x^3 - y^3 - z^3 = 0].$$

The following sentence is an example of a sentence not in $\mathcal{B}$

$$\exists x \in [0, 1] \; \exists y \in [0, 1] \,.\, x - y = 0$$

because the domain of the particular function is a 2-dimensional box and there is only one equation, so the assumptions in (a) are violated.

Throughout we will use the convention that logical connectives bind stronger than quantifiers. Moreover, we use brackets to denote Boolean structure of formulas. Sometimes we will use line breaks instead of brackets for this purpose. We will use the symbol $\equiv$ to denote equality of first-order formulas.

If $\exists x \in B \,.\, F_1$ and $\exists x \in B \,.\, F_2$ are in the class $\mathcal{B}$, then $\exists x \in B \,.\, [F_1 \vee F_2]$ is robust if and only if the formula $[\exists x \in B \,.\, F_1] \vee [\exists x \in B \,.\, F_2]$ is robust and they are equi-satisfiable. Hence a quasi-decision procedure for $\mathcal{B}$ can handle disjunctions within existential quantification, too. In the following, however, we will restrict ourselves to the class $\mathcal{B}$.

The following theorem shows a limitation of possible extension of quasi-decidability of the class $\mathcal{B}$ to the whole first-order theory removing the restriction on the number of equations versus number of variables:

Theorem 2 *Assume that the our symbol set is rich enough to contain function symbols for all piecewise linear functions defined on rational triangulations of boxes with rational values in the vertices. Then there is no algorithm Q with the following specification:*

- *Q is quasi-decision procedure for the class of sentences of the form*

$$\exists x \in [0, 1]^d \,.\, f_1(x) = 0 \wedge \ldots \wedge f_n(x) = 0 \wedge g(x) \geq 0$$

where $(f, g) : [0, 1]^d \to \mathbb{R}^n \times \mathbb{R}$ and d and n are arbitrary.

- *Q can access all functions f_j, g in the formula only via the oracle $\mathcal{I}(f_j)$, resp. $\mathcal{I}(g)$. That is, Q can call $\mathcal{I}(f_j)$ and $\mathcal{I}(g)$ arbitrary many times but has no access to the syntactical representation of f_j and g.*

As will be seen from the proof in Section 8, the second condition in Theorem 2 may be replaced by the alternative condition:

- Q does not terminate whenever the input is non-robust.

Whether or not the second condition in Theorem 2 can be omitted completely is—up to the best of our knowledge—an open problem.

3   The Quasi-decision Procedure

In this section, we construct an algorithm that decides, whether a robust sentence in $\mathcal{B}$ is true. The algorithm serves purely for proving Theorem 1. We do not claim it to be practically efficient whatsoever and leave a practically efficient quasi-decision procedure for future work.

For any formula $U \in \mathcal{B}$, variable x and $x_0 \in \mathbb{R}$ we denote by $U[x \leftarrow x_0]$ the formula derived from U by substituting x_0 for x in every free occurrence of x in U. We also allow x to be an n-tuple of variables, and $x_0 \in \mathbb{R}^n$, in which case $U[x \leftarrow x_0]$ denotes the parallel substitution of entries of x_0 with their corresponding entries of x.

In our algorithms, we use an alternative form of the Cartesian product that concatenates tuples from the argument sets, instead of forming pairs. That is, for sets $X \subseteq \mathbb{R}^n$ and $Y \subseteq \mathbb{R}^m$ it produces the set $\{(x_1, \ldots, x_n, y_1, \ldots, y_n) \mid (x_1, \ldots, x_n) \in X, (y_1, \ldots, y_m) \in Y\}$. Especially, for the set $\{()\}$ containing the 0-tuple, $\{()\} \times X$ will be X. The width of $\{()\}$, viewed as a box, is zero by definition.

We construct an auxiliary algorithm $\mathrm{CheckSat}(S, P, r)$ with the following specification:

Input:

- a formula S from $\mathcal{B}$ in l free variables p,
- an l-box P bounding the free variables of S,
- $r \in \mathbb{Q}^{>0}$,

such that the width of P is at most r.

Output: a nonempty subset of $\{\mathbf{T}, \mathbf{F}\}$

with the following two properties:

Correctness: If the algorithm returns $\{\mathbf{T}\}$ ($\{\mathbf{F}\}$), then for all $p_0 \in P$, $S[p \leftarrow p_0]$ is robustly true (robustly false).

Definiteness: If for a given l-box P_0 bounding the free variables of S, either for all $p_0 \in P_0$ the sentence $S[p \leftarrow p_0]$ is robustly true or for all $p_0 \in P_0$ the sentence $S[p \leftarrow p_0]$ is robustly false, then there exists an $\varepsilon > 0$ such that for every $r \leq \varepsilon$ and every sub-box $P \subseteq P_0$ with width smaller than r, the algorithm returns $\{\mathbf{T}\}$ or $\{\mathbf{F}\}$ (as opposed to $\{\mathbf{T}, \mathbf{F}\}$).

CheckSat(S, P, r) terminates always, but may return the indefinite result $\{\mathbf{T}, \mathbf{F}\}$. The existence of such an algorithm immediately implies Theorem 1, because then the algorithm below is a quasi-decision procedure for $\mathcal{B}$.

$\varepsilon \leftarrow 1$
loop
    $R \leftarrow \text{CheckSat}(S, \{()\}, \varepsilon)$
    if $|R| = 1$ **then**   // R is either $\{\mathbf{T}\}$ or $\{\mathbf{F}\}$
        return s s.t. $s \in R$
    else
        $\varepsilon \leftarrow \varepsilon/2$

Note that the specification of CheckSat does not only result in a quasi-decision procedure, but also checks robustness of the input.

We will now define the algorithm CheckSat(S, P, r) in detail. We will leave the proof that it fulfills the specification to Sections 5 (correctness) and 7 (definiteness). The algorithm is recursive, following the definition of class $\mathcal{B}$. We will now describe the parts corresponding to the individual cases of this definition.

3.1   System of Equations and Inequalities

We first consider the case (a) of class $\mathcal{B}$, that is, a formula S of the form

$$\exists x \in B . [f_1 = 0 \wedge \ldots \wedge f_n = 0 \wedge g_1 \geq 0 \wedge \ldots \wedge g_k \geq 0]$$

where B is an m-box. In an abuse of notation we also use $f_1, \ldots, f_n$ and $g_1, \ldots, g_k$ for the functions denoted by those terms. They are functions in $P \times B \to \mathbb{R}$ with $P \times B \subseteq \mathbb{R}^{l+m}$, where l is the number of free variables of S. We assume that the order of the arguments of those functions is the same as the order in which the respective variables are quantified in the overall formula. Finally, we denote by $f : P \times B \to \mathbb{R}^n$ the function defined by the components $(f_1, \ldots, f_n)$ and by $g : P \times B \to \mathbb{R}^k$ the function defined by the components $(g_1, \ldots, g_k)$.

Disproving the formula is straight-forward using the information given by $\mathcal{I}(f)$ and $\mathcal{I}(g)$. However, in order to ensure that the computed over-approximation is not too big, instead of working with $\mathcal{I}(f)(B)$ and $\mathcal{I}(g)(B)$ we work with elements of a partition S_r of B into small enough pieces, where

"small enough" is determined by the parameter r (Line 2 of the algorithm SoEI below). For this, we will call a set of boxes S_r a *grid covering* B iff $\bigcup_{B' \in S_r} B' = B$ and for every $B^1 \in S_r$ and $B^2 \in S_r$, $\mathsf{int}(B^1) \cap \mathsf{int}(B^2) = \emptyset$.

The core of the algorithm for proving the formula is a test whether a system of equations $f = 0$ has a solution in a bounded region. The test analyzes the boundary of the region and exploits continuity to deduce existence of a zero in the interior.

In the one-dimensional case, a bounded region is simply a closed interval. If f has opposite sign on the two end-points of the interval, the intermediate value theorem tells us, that f has a solution in the interior. Here f has to be non-zero on both interval endpoints (since f is in general non-polynomial, we cannot verify that f is zero on an interval endpoint, we can only exclude this). In general, we use the notion of the degree from the field of differential topology [28, 32]. For a continuous function $f : \Omega \to \mathbb{R}^n$ where Ω is a bounded open set and $p \notin f(\partial\Omega)$, the degree of f with respect to Ω and a point $p \in \mathbb{R}^n$ is an integer denoted by $\deg(f, \Omega, p)$. If $\deg(f, \Omega, p) \neq 0$ then the equation $f = p$ has a solution in Ω. Since the degree is a non-trivial mathematical notion, we defer more details on the degree to Section 4 below.

For ensuring that the test $\deg(f, \Omega, p) \neq 0$ eventually succeeds we have to make sure that Ω encloses a robust zero closely enough (the notion "closely enough" will be made precise in Sections 6 and 7). So, also in this case, we work with the partition S_r of B, and we compute the degree of the individual pieces. However, for ensuring that f is non-zero on the boundary of the pieces, we merge those pieces of the partition S_r for which we cannot prove that (Line 6).

Checking the inequalities is straight-forward (Lines 11 to 13) using $\mathcal{I}(g)$. In order to ensure that the used boxes are small enough, we undo the mergings before the check (Line 11) and apply $\mathcal{I}(g)$ to the individual boxes (Line 12).

The algorithm looks as follows:

Algorithm $\mathrm{SoEI}(S, P, r)$ // System of equations and inequalities

1: Let B be the m-box for the domain of the quantified variables in S.
2: Let S_r be a grid of boxes covering B s.t.
        each grid element has width at most r.
3: **if** for every box $A \in S_r$
        either $0 \notin \mathcal{I}(f)(P \times A)$ or $\mathcal{I}(g)(P \times A) \cap [0, \infty)^k = \emptyset$ **then**
4:         **return** $\{\mathbf{F}\}$ // $f = 0 \wedge g \geq 0$ has no solution
5: **if** $m = n$ **then**
6:     Merge all boxes in S_r containing a common face C s.t. $0 \in \mathcal{I}(f)(P \times C)$.
7:     Remove all grid elements in S_r containing a face C s.t. $C \subseteq \partial B$ and $0 \in \mathcal{I}(f)(P \times C)$.
8:     Let p_0 be an arbitrary element of P
9:     **for** each grid element $A \in S_r$ **do**

10:        **if** $\deg\left(f(p_0), A, 0\right) \neq 0$ **then** // equations hold, so check inequalities
11:            Let $S_r(A)$ be a grid of boxes covering A of width at most r
12:            **if** for all $E \in S_r(A)$, $\mathcal{I}(g)(P \times E) \subseteq (0, \infty)^k$ **then**
13:                **return** $\{\mathbf{T}\}$
14:**return** $\{\mathbf{T}, \mathbf{F}\}$ // no test succeeded, or $n > m$

Here we suppose that f is present in the formula (i.e., $n > 0$). The algorithm can be easily adapted to the case, where it is not. In the case $n > m$, the algorithm can simply return $\{\mathbf{T}, \mathbf{F}\}$, see Lemma 5 below. An illustration of the algorithm is shown in Figure 1.
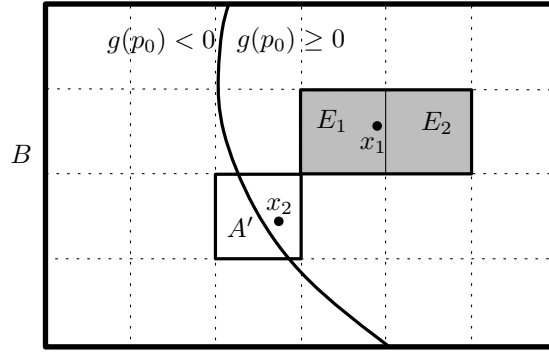

Figure 1: Illustration of the SoEI algorithm. Assume that $f(p_0)$ has two zeros x_1 and x_2, and assume that $\{x \mid g(p_0, x) \geq 0\}$ is to the right of the thick curve. The algorithm creates a grid of boxes S_r (line 2). If each element of the grid provably does not contain a solution (check at line 3), it returns $\{\mathbf{F}\}$. If this is not the case, then it checks whether f is non-zero on all boundaries of grid elements (line 6). In our example, f is close to zero on the common boundary of E and E' and so the algorithm merges them into one grid element A_1. If $\deg\left(f(p_0), A_1, 0\right) \neq 0$, then it checks whether for each p, $g(p) \geq 0$ on E_1 and E_2 (line 12). If this is true as well, then $f = 0 \wedge g \geq 0$ is robustly satisfiable on B and the algorithm terminates with $\{\mathbf{T}\}$. In case of another box A_2 containing a robust zero of f, the given partition may not provide enough evidence for the claim that $g(p, x_2) \geq 0$ for each p (in which case the condition on line 12 is not satisfied).

3.2   Universal Quantifiers

The recursive call corresponding to Case (b) of class $\mathcal{B}$ looks as follows:

Algorithm $\mathrm{Univ}(\forall x \in I \ . \ S, P, r)$:

Let I_r be a grid of sub-intervals of I of width at most r
return $\tilde{\bigwedge}_{I' \in I_r} \mathrm{CheckSat}(S, P \times I', r)$

11

Here, in the return statement, the symbol $\tilde{\bigwedge}$ denotes the lifting of Boolean conjunction to sets of Boolean values:

$$U \tilde{\wedge} V := \{u \wedge v \mid u \in U, v \in V\}.$$

3.3 Conjunctions and Disjunctions

Finally, the recursive call corresponding to Case (c) of class $\mathcal{B}$ looks as follows:

Algorithm $\mathrm{Conj}(S \wedge T, P, r)$

return $\mathrm{CheckSat}(S, P_1, r) \tilde{\wedge} \mathrm{CheckSat}(T, P_2, r)$
$\qquad\qquad$ where P_1 (P_2) is the projection of P
$\qquad\qquad\qquad$ to the free variables of S (T, respectively).

Here, in the return statement, the symbol $\tilde{\wedge}$ again denotes the lifting of conjunction to sets of Boolean values. The algorithm for disjunction is completely analogous, replacing $\tilde{\wedge}$ with $\tilde{\vee}$ (and its lifting to sets of Boolean values).

4 Degree of a Continuous Function

In this section we describe some basic properties of the topological degree. We already mentioned in the introduction that in the one-dimensional case, that is, for continuous functions $f : [a, b] \to \mathbb{R}$ with $f(a) \neq 0$ and $f(b) \neq 0$, the degree $\deg(f, [a, b], 0)$ is 0 iff $f(a)$ and $f(b)$ have the same sign, otherwise the degree is either -1 or 1, depending on whether the sign changes from negative to positive or the other way round. Hence, in this case, the degree gives the information given by the intermediate value theorem plus some directional information.

In dimension two, the degree of a continuous function f from a disc to $\mathbb{R}^2$ is just the number of times $f(x)$ winds around the origin counter-clockwise as x follows the circle forming the boundary of the disc (i.e., the "winding number"). Again, a non-zero winding number implies that f has a zero.

There are several ways of defining the degree in general. We work with an axiomatic definition, that can be shown to be unique [32, Section I.5]. Let $\Omega \subseteq \mathbb{R}^n$ be open and bounded, $f : \bar{\Omega} \to \mathbb{R}^n$ continuous, and $p \notin f(\partial\Omega)$. Then $\deg(f, \Omega, p)$ is an integer satisfying the following properties [31, Thm. 1.2.6.]:

1. For the identity function I, $\deg(I, \Omega, p) = 1$ iff $p \in \Omega$

2. If $\deg(f, \Omega, p) \neq 0$ then $f(x) = p$ has a solution in Ω

3. If there is a continuous function (a "homotopy") $h : [0,1] \times \bar{\Omega} \to \mathbb{R}^n$ such that $h(0) = f$, $h(1) = g$ and $p \notin h(t, \partial\Omega)$ for all t, then $\deg(f, \Omega, p) = \deg(g, \Omega, p)$

4. If $\Omega_1 \cap \Omega_2 = \emptyset$, $\Omega_1 \subseteq \Omega$, $\Omega_2 \subseteq \Omega$, and $p \notin f(\bar{\Omega} \setminus (\Omega_1 \cup \Omega_2))$, then $\deg(f, \Omega, p) = \deg(f, \Omega_1, p) + \deg(f, \Omega_2, p)$

5. $\deg(f, \Omega, p)$, as a function of p, is constant on any connected component of $\mathbb{R}^n \setminus f(\partial\Omega)$.

The first axiom says that for the identity function, the degree counts the zeros in Ω precisely. Due to the second axiom one can infer existence of a zero from a non-zero degree. Due to the third axiom, the degree is invariant under continuous deformations of the function that do not cause any essential change of the boundary information. From this it can be immediately seen that the degree depends only on the boundary $\partial\Omega$: for two functions f and g that agree on $\partial\Omega$, the function $h(t, x) = tf(x) + (1-t)g(x)$ is a homotopy between f and g, as needed by the premise of Axiom 3.

In the SoEI algorithm, we apply the degree to the triple $(f, A, 0)$ where A is not open but the closure of an open set (it is the union of boxes). For completeness, we define $\deg(f, A, p) := \deg(f, A^\circ, p)$ where A° is the interior of A, whenever $p \notin f(\partial A)$.

Many algorithms for computing the degree have been proposed [15, 25, 7, 1, 17]. More specifically, if B is an n-box, $f : B \to \mathbb{R}^n$ is interval computable, $0 \notin f(\partial B)$ and an algorithm $\mathcal{I}(f)$ is given, then the degree $\deg(f, B, 0)$ can be algorithmically computed. This justifies the use of line 10 of algorithm SoEI in Section 3.1.

The axioms defining the degree only argue about zeros, but not about robustness. Still, a nonzero degree is closely connected with the existence of a robust root:

Lemma 1 *Let $\bar{\Omega} \subseteq \mathbb{R}^n$ be a closed region with interior Ω, $f : \bar{\Omega} \to \mathbb{R}^n$ be continuous, $0 \notin f(\partial\Omega)$ and let $\deg(f, \Omega, 0) \neq 0$.*

Then any continuous $g : \bar{\Omega} \to \mathbb{R}^n$ such that $\|g - f\| < \min_{x \in \partial\Omega} |f|$ has a zero in Ω.

Proof. Let $\varepsilon < \min_{x \in \partial\Omega} |f|$. For any g such that $\|g - f\|_{\bar{\Omega}} < \varepsilon$, we define a homotopy $h(t, x) = tf(x) + (1-t)g(x)$ between f and g. We see that for $x \in \partial\Omega$ and $t \in [0, 1]$,

$$|h(t, x)| = |tf(x) + (1-t)g(x)| = |f(x) + (1-t)(g(x) - f(x))| \geq |f(x)| - \varepsilon > 0$$

so that $h(t, x) \neq 0$ for $x \in \partial\Omega$. From Properties 2 and 3, we see that $g(x) = 0$ has a solution. ∎

In particular, this implies that the sentence $\exists x \in B \cdot f(x) = 0$ is not only true, but also robust, whenever $\deg(f, B^\circ, 0) \neq 0$. The upper bound

on the distance between f and g results in an ε such that this sentence is ε-robust. This allows extensions of the algorithms of this paper to return such an ε, which may be useful in applications.

For proving definiteness, we will need a partial converse of this statement which will be given by Theorem 6 in Section 6.

5 Proof of Correctness

We will prove here that the algorithm CheckSat proposed in Section 3 fulfills the first part of its specification, that is: it always returns a correct result. The proof will again be divided into the cases constituting the definition of class $\mathcal{B}$, from which correctness of the overall, recursive algorithm follows by induction.

Before that, we prove some technical results on the relationship between the class $\mathcal{B}$ and robustness.

Note that, in this section, the assumption that our symbol set contains addition and multiplication, is not used. Hence the algorithm is correct even if we do not have those symbols in the symbol set.

5.1 Robustness and the Class $\mathcal{B}$

First we prove a lemma on the effect of substitution of nearby constants on robustness.

Lemma 2 *Let S be a formula in l free variables, P an l-box bounding the free variables of S and p_0 be a point in the interior of P. If $S[p \leftarrow p_0]$ is a robust sentence, then there exists a neighborhood $U \subseteq \mathbb{R}^l$ of p_0, such that for all $u \in U$, $S[p \leftarrow u]$ is robust and has the same truth value as $S[p \leftarrow p_0]$.*

Proof. Assume that $S[p \leftarrow p_0]$ is robust. Then there is an $\varepsilon > 0$ such that for all formulas T with $d(S[p \leftarrow p_0], T) < \varepsilon$, T and $S[p \leftarrow p_0]$ have the same truth value. Since all functions in S are interval-computable, they are uniformly continuous. Hence for $\varepsilon > 0$, there exists a number $\delta > 0$ such that for each function f occurring in S it holds that $|f(x, u) - f(x, p_0)| < \varepsilon/2$ whenever $u, p_0 \in P$ and $|u - p_0| < \delta$. In other words, there exists a $\delta > 0$ s.t. for all $u \in P$ with $|u - p_0| < \delta$, $d(S[p \leftarrow p_0], S[p \leftarrow u]) < \varepsilon/2$, and hence $S[p \leftarrow p_0]$ and $S[p \leftarrow u]$ have equal truth value. We claim that $S[p \leftarrow u]$ is also robust: this is because if T' is any sentence with $d(S[p \leftarrow u], T') < \varepsilon/2$, then $d(T', S[p \leftarrow p_0]) < \varepsilon$ and T' has still the same truth value as $S[p \leftarrow p_0]$. So the neighborhood $U := \{u \in P \mid |u - p_0| < \delta\}$ of p_0 satisfies the required properties. ∎

Due to the syntactical structure of formulas in the class $\mathcal{B}$ we automatically have robustness in the false case:

Lemma 3 *Let S be a sentence from $\mathcal{B}$. If S is false, then it is robustly false.*

Proof. We proceed by induction, following the cases of class $\mathcal{B}$. Let S be the sentence $\exists x \in B \,.\, f = 0 \land g \geq 0$, where $f = 0$, and $g \geq 0$ are the usual short-cuts for conjunctions of equalities, and inequalities, respectively. Let S be false. If $f = 0$ has no solution in B, then $||f|| > \varepsilon$ for some $\varepsilon > 0$ and $||\tilde{f}|| > 0$ for small enough perturbations $\tilde{f}$ of f. Similarly, if $g < 0$ on B, then the same is true for small enough perturbations of g. Finally, if $f^{-1}(\{0\}) \cap B$ and $g^{-1}[0,\infty)^k \cap B$ are both nonempty, then they are compact and disjoint, which implies that they have a positive distance. For small perturbations $\tilde{f}, \tilde{g}$ of f and g, $\tilde{f}^{-1}\{0\}$ and $\tilde{g}^{-1}[0,\infty)^k$ are still disjoint, which implies that S is robustly false.

Further, assume that $I \subseteq \mathbb{R}$ is a compact interval and $\forall x \in I \,.\, S$ is a false sentence. Then there exists an $x_0 \in I$ such that $S[x \leftarrow x_0]$ is false. From the induction hypothesis, it is robustly false. Let $\varepsilon > 0$ be such that $S[x \leftarrow x_0]$ is ε-robust and let S' be a formula such that $d(\forall x \,.\, S', \forall x \,.\, S) < \varepsilon$. Then $d(S'[x \leftarrow x_0], S[x \leftarrow x_0]) < \varepsilon$ and $S'[x \leftarrow x_0]$ is false. So, $\forall x \in I \,.\, S'$ is false and it follows that $\forall x \in I \,.\, S$ is robustly false.

Finally, let U and V be sentences in $\mathcal{B}$ and $U \land V$ be false. Then either U or V is false and the induction hypothesis says that it is robustly false. So, $U \land V$ is robustly false. Similarly, if $U \lor V$ is false, then both U and V are robustly false and $U \lor V$ is robustly false. ∎

In the case of this lemma, the proof goes through for any number of equalities, independent of the restriction that class $\mathcal{B}$ puts on this number. Further, the last lemma remains true even if we leave the set of interval-computable functions and allow arbitrary, small enough continuous perturbations. Moreover, it holds even if all functions in the original formula S are only continuous and not interval computable. We only have used continuity of the perturbations and the proof does not use any algorithmic input.

Universal quantification preserves robustness in the following sense:

Lemma 4 *Let S be a formula containing a free variable x and let I be a bounded closed interval. Then the sentence $S[x \leftarrow x_0]$ is robustly true for all x_0 in I if and only if the sentence $\forall x \in I \,.\, S$ is robustly true.*

Proof. Let $\forall x \in I \,.\, S$ be ε-robust and true, and let x_0 be an arbitrary, but fixed element of the interval I. Then clearly $S[x \leftarrow x_0]$ is true. For showing that it is also robust, we assume an arbitrary, but fixed sentence X such that $d(X, S[x \leftarrow x_0]) =: \varepsilon' < \varepsilon$ and prove that X is true, as well. Let f_X, resp. g_X be the functions that occur in X on the places corresponding to $S[x \leftarrow x_0]$; this is well-defined, because X and $S[x \leftarrow x_0]$ have the same structure. Consider the formula U that is equal to S except for the fact that every equality of the form $f = 0$ is replaced by $f + f_X - f[x \leftarrow x_0] = 0$ and

$g \geq 0$ is replaced by $g + g_X - g[x \leftarrow x_0] \geq 0$. The distance $d(\forall x \in I \;.\; S, \forall x \in I \;.\; U) = \varepsilon' < \varepsilon$ and so, due to ε-robustness of $\forall x \in I \;.\; S$, $\forall x \in I \;.\; U$ is true. In particular, $U[x \leftarrow x_0] \equiv X$ is true and it follows that $S[x \leftarrow x_0]$ is ε-robust and true.

For the converse, assume that for all $x_0 \in I$, $S[x \leftarrow x_0]$ is robustly true. Let

$$\mu(x_0) := \sup\{\mu > 0;\ S[x \leftarrow x_0] \text{ is } \mu\text{-robust}\}.$$

Clearly, μ is a continuous function in x_0 and has strict lower bound $m > 0$ on the compact interval I. So, for each $x_0 \in I$, $S[x \leftarrow x_0]$ is m-robust. If $d(\forall x \in I \;.\; S, \forall x \in I \;.\; U) < m$, then for each $x_0 \in I$, $d(S[x \leftarrow x_0], U[x \leftarrow x_0]) < m$ and $U[x \leftarrow x_0]$ is true. So, $\forall x \in I \;.\; U$ is true and $\forall x \in I \;.\; S$ is robustly true. ∎

Again, the last lemma remains true in the stronger formulation where we consider a statement robustly true iff any small enough continuous perturbation of its function symbols is true—that is, perturbation by functions that do not necessarily correspond to terms formed from the given set of function symbols or functions that are not necessarily interval computable.

5.2 System of Equations and Inequalities

For proving correctness of the algorithm CheckSat we again start with the case (a) of class $\mathcal{B}$, that is, a formula S of the form

$$\exists x \in B \;.\; [f_1 = 0 \wedge f_2 = 0 \wedge \ldots \wedge f_n = 0 \wedge g_1 \geq 0 \wedge g_2 \geq 0 \wedge \ldots \wedge g_k \geq 0]$$

where B is an m-box. Assuming that the formula has l free variables, we again denote by $f : \mathbb{R}^{l+m} \to \mathbb{R}^n$ the function defined by the components $(f_1, \ldots, f_n)$ and $g : \mathbb{R}^{l+m} \to \mathbb{R}^k$ the function defined by the components $(g_1, \ldots, g_k)$.

Theorem 3 *The algorithm* $\mathrm{SoEI}(S, P, r)$ *fulfills the correctness property of the specification of* $\mathrm{CheckSat}(S, P, r)$ *(defined at the beginning of Section 3).*

Proof.

Assume first that the algorithm terminates with a negative result $\{\mathbf{F}\}$. It follows directly from Definition 1, that the input sentence $S[p \leftarrow p_0]$ is false for any $p_0 \in P$. Lemma 3 implies robustness.

Now assume that it terminates with a positive result $\{\mathbf{T}\}$. Then there exists a point $p_0 \in P \subseteq \mathbb{R}^l$ and a connected grid element $A \subseteq \mathbb{R}^m$ such that $\deg(f(p_0), A, 0) \neq 0$. For any $p \in P$, p and p_0 can be connected by a curve $\phi : [0, 1] \to P$, and $f \circ \phi$ is then a homotopy between $f(p_0)$ and $f(p)$ nowhere zero on ∂A. So, $\deg(f(p), A, 0) \neq 0$ and it follows from Lemma 1 that $f(p) = 0$ has a robust solution in A. Moreover, the successful check

whether for all $E \in S_r(A)$, $\mathcal{I}(g)(P \times E) \subseteq (0, \infty)^k$ implies that for some small enough $d > 0$, for all $p \in P$, $x \in A$ and $j = 1, \ldots, k$, $g_j(p, x) > d$. It follows that the input formula is robustly true for all parameter values in P. ∎

5.3 Universal Quantifiers

Theorem 4 *Let S be a formula containing free variables p. Let P be an l-box and I a closed interval. Assume that an algorithm CheckSat fulfilling the correctness property is given. Then also the algorithm* $\mathrm{Univ}(\forall x \in I . S, P, r)$ *fulfills the correctness property.*

Proof. If $\mathrm{Univ}(\forall x \in I . S, P, r)$ returns $\{\mathbf{F}\}$, then $\mathrm{CheckSat}(S, P \times I', r')$ returned $\{\mathbf{F}\}$ for some $I' \in I_r$ and it follows that for all $p_0 \in P$ and $x_0 \in I'$, $S[p \leftarrow p_0][x \leftarrow x_0]$ is robustly false. Then $\forall x \in I . S[p \leftarrow p_0]$ is false for each $p_0 \in P$ and it follows from Lemma 3 that it is robustly false.

If the algorithm returns $\{\mathbf{T}\}$, then $\mathrm{CheckSat}(S, P \times I', r')$ returned $\{\mathbf{T}\}$ for all $I' \in I_r$ and the sentence $S[p \leftarrow p_0][x \leftarrow x_0]$ is robustly true for all $x_0 \in I$ and $p_0 \in P$. It follows from Lemma 4 that for each $p_0 \in P$, $\forall x \in I . S[p \leftarrow p_0]$ is robustly true, so the result is correct. ∎

5.4 Conjunction and Disjunction

Theorem 5 *Let S and T be two formulas in $\mathcal{B}$ and assume that CheckSat fulfills the correctness property both when applied to S, and when applied to T. Then* $\mathrm{Conj}(S \wedge T, P, r)$ *also fulfills the correctness property.*

Proof. Let p_S, and p_T, respectively, be the function that projects any l-tuple corresponding to the free variables of $S \wedge T$ to those components corresponding to the free variables of S, and T, respectively.

If Conj returned $\{\mathbf{T}\}$ then the recursive calls for both S and T returned $\{\mathbf{T}\}$. Hence, by correctness of the result of the recursive calls, for all $p_0 \in P$, $S[p_S(p) \leftarrow p_S(p_0)]$ and $T[p_T(p) \leftarrow p_T(p_0)]$ are robustly true, and hence also $(S \wedge T)[p \leftarrow p_0]$.

If Conj returned $\{\mathbf{F}\}$ then the recursive calls for either S or T returned $\{\mathbf{F}\}$. Hence, by correctness of the result of the recursive calls, either for all $p_0 \in P$, $S[p_S(p) \leftarrow p_S(p_0)]$ is robustly false, or for all $p_0 \in P$, $T[p_T(p) \leftarrow p_T(p_0)]$ is robustly false. Hence, also for all $p_0 \in P$, $(S \wedge T)[p \leftarrow p_0]$ is robustly false. ∎

For disjunctions the situation is analogous.

6 From Robustness To Non-Zero Degree

For proving that the algorithm CheckSat fulfills the second part of its specification, definiteness, we need to prove that for a robust system of equations, the test provided by a non-zero topological degree eventually succeeds. While the algorithmic aspects of the proof are part of the next section, in this section we prove two properties of the degree necessary for this (Lemma 5 and Theorem 6). The first property, Lemma 5, simply says that in the case overdetermined system of n equations in $m < n$ variables, the input cannot be robust, and hence the implication (robust input implies succeeding test for non-zero degree) holds vacuously. The second property, Theorem 6, shows that robustness implies existence of a region for which the degree is non-zero. More precisely, we will show a partial converse to Lemma 1, that is, that a robust solution of $f = 0$ on Ω implies the existence of a region $U \subseteq \Omega$ s.t. $0 \notin f(\partial U)$ and $\deg(f, U, 0) \neq 0$.

The rest of the paper will only refer to the two mentioned properties, so a reader can safely skip this section after noting Lemma 5 and Theorem 6. The proofs in the section are the only place in the paper that uses results from topology that are not explicitly delineated in this paper.

Lemma 5 *Let $\bar{\Omega}$ be a closed region in $\mathbb{R}^m$, $n > m$ and $f : \Omega \to \mathbb{R}^n$ be continuous. Then for each $\varepsilon > 0$ there exists a function $g : \bar{\Omega} \to \mathbb{R}^n$, $\|g - f\| < \varepsilon$, with no root.*

Proof. We assume that for some ε, it holds that each g closer to f than ε has a root, and derive a contradiction. It follows from the Stone-Weierstrass theorem that the continuous function f may be approximated arbitrarily precisely with a smooth function (even with a polynomial), and so we can approximate it by a smooth function $\tilde{f}$ closer than $\varepsilon/2$ to f. Moreover, each such $\tilde{f}$ with $\|\tilde{f} - f\| < \varepsilon/2$ has a root. In particular, $\tilde{f}(x) - c$ has a root for any constant c, $|c| < \varepsilon/2$ and so $\tilde{f}(\Omega)$ contains a neighborhood of $0 \in \mathbb{R}^n$. However, all values in $\tilde{f}(\Omega)$ are critical values (that is, for each $x \in \bar{\Omega}$, the rank of $f'(x)$—a matrix $n \times m$—is smaller than n). Due to Sard's theorem [28, Chapter 2] the set of critical values of a smooth function has zero measure in $\mathbb{R}^n$, and so $\tilde{f}(\Omega)$ cannot contain a neighborhood of $0 \in \mathbb{R}^n$, a contradiction. ∎

The rest of the section considers the case of equal dimensions $m = n$. First we show that a zero degree of a function implies that any possible zero of the function can be removed by a change of the function only in the interior. Moreover, the result of the change will be small in a certain sense.

Lemma 6 *Let $\bar{\Omega}$ be a closed region in $\mathbb{R}^n$, $f : \bar{\Omega} \to \mathbb{R}^n$ continuous, $0 \notin f(\partial\Omega)$ and $\deg(f, \Omega, 0) = 0$. Then there exists a continuous nowhere zero function $g : \bar{\Omega} \to \mathbb{R}^n$ such that $g = f$ on $\partial\Omega$ and $\|g\|_{\bar{\Omega}} \leq \|f\|_{\bar{\Omega}}$.*

Proof. If $0 \notin f(\Omega)$, we may take $g = f$. Otherwise, take a neighborhood $U \subseteq \Omega$ of $f^{-1}(0)$ such that ∂U is an $(n-1)$-manifold (i.e. locally homeomorphic to $\mathbb{R}^{n-1}$). Such a neighborhood U might be constructed as a finite union of balls. It follows from the degree axioms that $\deg(f, U, 0) = \deg(f, \Omega, 0)$ and it is a well-known fact in differential topology that $f/|f| : \partial U \to S^{n-1}$ can be extended to a function $g_1 : U \to S^{n-1}$ iff the degree is zero [24, Theorem 8.1.]. Let $h : \bar{U} \to \mathbb{R}^+$ be an extension of $|f| : \partial U \to \mathbb{R}^+$ (such extension exists due to Tietze's Extension Theorem [9, Thm. 4.22]) and let $i : S^{n-1} \to \mathbb{R}^n \setminus \{0\}$ be the inclusion. Then $g_2 := h(i \circ g_1) : \bar{U} \to \mathbb{R}^n \setminus \{0\}$ is a nowhere zero extension of $f|_{\partial U}$. Define $g : \bar{\Omega} \to \mathbb{R}^n \setminus \{0\}$ by $g(x) = g_2(x)$ for $x \in U$ and $g(x) = f(x)$ for $x \notin U$. This function is continuous, nowhere zero and coincides with f on ∂U. Possibly multiplying g by a positive scalar valued function that equals 1 on ∂U and is small inside U°, we achieve that $||g||_\Omega \leq ||f||_\Omega$. ∎

Now we show that for a smooth function f, we might change it within a small region N where the function is nonzero, to produce arbitrary many regular zero points, both orientation-preserving and orientation-reversing.

Lemma 7 *Let U be an open set in $\mathbb{R}^n$, $f : U \to \mathbb{R}^n$ be smooth. Let N be a neighborhood of $x^0 \in U$ such that $0 \notin f(N)$ and let $k \in \mathbb{N}$. Then there exists a function f_1 such that the following conditions are satisfied:*
(1) $f_1 = f$ on $U \setminus N$
(2) $||f_1|| \leq ||f||$
(3) 0 is a regular value of $f_1|_N$
(4) N contains $2k$ points $x_1, \ldots, x_k, y_1, \ldots, y_k$ such that $f_1(x_i) = f_1(y_i) = 0$, f_1 is orientation-preserving in the neighborhood of x_i and orientation-reversing in the neighborhood of y_i.

Proof. Choose $\delta > 0$ such that $x^0 + [-2\delta, 2\delta]^n \subseteq N$. We construct f_1 such that $f_1(x) = f(x)$ for $x \notin (x^0 + [-2\delta, 2\delta]^n)$. For $x \in (x^0 + [-\delta, \delta]^n)$ we set

$$(f_1)_i(x) = \left(\frac{|x_i - x_i^0|}{\delta} - \frac{1}{2} \right) f_i(x^0).$$

It is easy to see that $f_1^{-1}(0)$ contains in $(x^0 + [\pm\delta])$ 2^n points of the form $(x_1^0 \pm \delta/2, x_2^0 \pm \delta/2, \ldots, x_n^0 \pm \delta/2)$, half of them preserve orientation and half reverse orientation. Clearly, $|f_1(x)| \leq |f(x^0)| \leq ||f||$ on $x_0 + [\pm\delta]$. Because $\deg(f_1, x^0 + [\pm\delta], 0) = \deg(f, x^0 + [\pm 2\delta]) = 0$, it is easy to see that f_1 may be extended to $x^0 + [\pm 2\delta]$ so that $f_1 = f$ on $\partial(x^0 + [\pm 2\delta])$, f_1 is nonzero in $x_0 + [\pm 2\delta] \setminus (x_0 + [\pm\delta])$ and the norm $||f_1|| \leq ||f||$. The only zero points of f_1 in N are $(x_1^0 \pm \delta/2, \ldots, x_n^0 \pm \delta/2)$, so 0 is a regular value of $f_1|_N$. The details are left to the reader.

To produce more zeros we can choose any point $x_1 \in N$ s.t. $f_1(x_1) \neq 0$ and a small neighborhood of x_1 in N where f_1 is nonzero and continue in the same way. ∎

Finally, we prove the following theorem that will be used in the proof of definiteness of the CheckSat procedure.

Theorem 6 *Let $\bar{\Omega}$ be a non-empty closed region in $\mathbb{R}^n$ with interior $\Omega \subseteq \mathbb{R}^n$ and $f : \bar{\Omega} \to \mathbb{R}^n$ be continuous. Then there exists an $\varepsilon > 0$ such that each continuous g, $\|g - f\| < \varepsilon$, has a zero in $\bar{\Omega}$ if and only if there exists an open set $U \subseteq \Omega$ such that $0 \notin f(\partial U)$ and $\deg(f, U, 0) \neq 0$.*

The assumption that Ω is the interior of $\bar{\Omega}$ is necessary to exclude some degenerate cases such as $\Omega = (-1, 1) \setminus \{0\}$ and $f(x) = x$; in this case, f has a robust zero in $\bar{\Omega} = [-1, 1]$ but for any $U \subseteq \Omega$ with $0 \notin f(\partial U)$, $\deg(f, U, 0) = 0$.

Proof. If the dimension is $n = 1$, then $\bar{\Omega}$ is a compact interval and clearly there exists an $\varepsilon > 0$ such that each continuous ε-perturbation of f has a zero iff there exists $x, y \in \bar{\Omega}$ s.t. $f(x) < 0 < f(y)$, and the statement follows. In the rest of the proof we assume that $n \geq 2$.

If $\deg(f, U, 0) \neq 0$ for some U, then we may choose $\varepsilon := \min_{x \in \partial U} |f|$ by Lemma 1 which proves one implication.

For proving the other direction, we assume that for each open $U \subseteq \Omega$ s.t. $0 \notin f(\partial U)$, $\deg(f, U, 0) = 0$. We choose a positive $\varepsilon > 0$ and will show that there exists a continuous 4ε-perturbation g of f with no root.

Let $\Omega_\varepsilon := \{x \mid |f(x)| < \varepsilon\}$. This is an open set in $\bar{\Omega}$. Let $x \in f^{-1}(0) \cap \Omega$. Then there exists a ball $U(x) \subseteq \mathbb{R}^n$ open in $\mathbb{R}^n$ such that $U(x) \subseteq \Omega_\varepsilon$. For $y \in f^{-1}(0) \cap \partial\Omega$, we choose $U(y) \subseteq \mathbb{R}^n$ to be an open ball in $\mathbb{R}^n$ such that $U(y) \cap \bar{\Omega} \subseteq \Omega_\varepsilon$. We assumed that Ω is the interior of $\bar{\Omega}$, which implies $\partial\Omega = \partial\bar{\Omega}$. So, for each such $U(y)$, the set $U(y) \setminus \bar{\Omega}$ is a nonempty open set in $\mathbb{R}^n$.

The set $\{U(x) \mid x \in f^{-1}(0)\}$ is an open cover of the compact set $f^{-1}(0)$, so we may take finitely many of these sets $U_1, \ldots, U_k$ that still cover $f^{-1}(0)$. Each U_i is either contained in Ω_ε, or has a nontrivial intersection with $\partial\Omega$. Let $V_1, \ldots, V_l, W_1, \ldots, W_m$ be the pairwise disjoint connected components of $\cup_i U_i$ such that $V_i \subseteq \Omega_\varepsilon$ and $W_j \cap \partial\Omega \neq \emptyset$ for each i, j.

If $x \in \partial V_i$, then $f(x) \neq 0$, otherwise x would be contained in the interior of the same connected component V_i of $\cup_i U_i$. In particular, $0 \notin f(\partial V_i)$ and due to the assumption above $\deg(f, V_i, 0) = 0$. V_i is connected and it follows from Lemma 6 that we may change f inside V_i, without changing it on $\bar{\Omega} \setminus V_i$, to construct a function $f_1 : \bar{\Omega} \to \mathbb{R}^n$, $0 \notin f_1(V_i)$ and $\|f_1\|_{V_i} \leq \|f\|_{V_i}$. The inequalities $\|f_1\|_{V_i} \leq \|f\|_{V_i} \leq \varepsilon$ imply that f_1 is a continuous 2ε-perturbation of f. This can be done independently for each i, so we may assume that $0 \notin f_1(\cup_i V_i)$.

Let us extend f_1 to a continuous function $f_2 : \bar{\Omega} \cup_j \bar{W}_j \to \mathbb{R}^n$ (such an extension exists by Tietze's Theorem). Possibly multiplying f_2 by a positive scalar valued function that equals 1 on $\bar{\Omega}$ and is small outside $\bar{\Omega}$, we may assume that $\|f_2\|_{\cup_j W_j} \leq \varepsilon$. The zero set of f_2 is contained in $\cup_j \bar{W}_j$

and if $f_2(x) = 0$ for some $x \in \partial W_j$, then $x \notin \bar{\Omega}$ (otherwise, x would be contained in the same connected component of $\cup_i U_i$ as W_j, contradicting $x \in \partial W_j$). Therefore, f_2 is nowhere zero on the compact set $\bar{\Omega} \setminus \cup_j W_j$ and there exists some $0 < \varepsilon_1 < \varepsilon$ s.t. $|f(x)| > \varepsilon_1$ for $x \in \bar{\Omega} \setminus \cup_j W_j$. Let f_3 be a continuous ε_1-perturbation of f_2 that is smooth and 0 is a regular value of f_3 (such a perturbation exists by Stone-Weierstrass and Sard's theorems). The set $f_3^{-1}(0)$ is finite and contained in $\cup_j \bar{W}_j$. For each j and each $x \in f_3^{-1}(0) \cap \partial W_j$, we may find a small neighborhood O_x of x such that x is the only zero point of f_3 on $\bar{O}_x$, $\bar{O}_x \cap \bar{\Omega} = \emptyset$, $W_j \setminus \bar{O}_x$ is still connected, and replace W_j by $W_j \setminus \bar{O}_x$. So, we can assume that $0 \notin f_3(\partial W_j)$ for each j. Let $A^+(W_j) = \{x \in W_j \mid f_3(x) = 0, \det(f_3'(x)) > 0\}$ and $A^-(W_j) = \{x \in W_j \mid f_3(x) = 0, \det(f_3'(x)) < 0\}$.

$W_j \setminus \bar{\Omega}$ is open and nonempty, and we can use Lemma 7 to create at least $2\|\,|A^+(W_j)| - |A^-(W_j)|\,\|$ zeros in $W_j \setminus \bar{\Omega}$ of f_3 in which f_3 is orientation-preserving, resp. orientation-reversing, without changing f_3 in $W_j \cap \bar{\Omega}$. We can then pair all points in $A^+(W_j) \cap \bar{\Omega}$ with points in $A^-(W_j) \setminus \bar{\Omega}$ and points in $A^-(W_j) \cap \bar{\Omega}$ with $A^+(W_j) \setminus \bar{\Omega}$ (some zeros of f_3 outside $\bar{\Omega}$ may still remain unpaired). We suppose that the dimension $n \geq 2$, so we may connect each pair of points x_a^+ and x_a^- by a curve c_a so that the curves do not intersect themselves and the complement of these curves in W_j is still connected. Further, there exist connected and pairwise disjoint open neighborhoods N_a of these curves such that the only zero points of f_3 in N_a are x_a^+ and x_a^- for each a. The degree $\deg(f_3, N_a, 0) = 0$, so we may change f_3 inside N_a to a continuous function f_4 s.t. $\|f_4\|_{N_a} \leq \|f_3\|_{N_a}$, and $0 \notin f_4(N_a)$. In this way, we destroy all zeros of f_3 in $\bar{\Omega}$ (although some zeros may still exists outside $\bar{\Omega}$). We assumed that $\|f_2\|_{W_j} \leq \varepsilon$, so $\|f_3\|_{W_j} \leq \varepsilon + \varepsilon_1 \leq 2\varepsilon$ and $f_4|_{W_j}$ is a continuous 4ε-perturbation of $f|_{W_j}$. Changing f_3 independently in each N_a, the resulting function $f_4|_{\bar{\Omega}}$ is a nowhere zero continuous 4ε-perturbation of f. ∎

7   Proof of Definiteness

We will prove here that the algorithm CheckSat proposed in Section 3 fulfills the second part of its specification, that is, definiteness. This will complete the proof of Theorem 1. The definiteness proof will again be divided into the cases constituting the definition of class $\mathcal{B}$, from which correctness of the overall, recursive algorithm follows by induction.

Unlike in Section 5, in this section, the assumption that the symbol set of our language contains rational constants, addition, and multiplication, and consequently all polynomials with rational coefficients is needed: it will allow us to construct terms representing functions that are arbitrarily close to a given continuous function.

7.1 System of Equations and Inequalities

We again start with the case (a) of class $\mathcal{B}$, that is, a formula S of the form

$$\exists x \in B \; . \; [f_1 = 0 \;\wedge\; f_2 = 0 \;\wedge\; \ldots \wedge\; f_n = 0 \;\wedge\; g_1 \geq 0 \;\wedge\; g_2 \geq 0 \;\wedge\; \ldots \wedge\; g_k \geq 0]$$

where B is an m-box. Assuming that the formula has l free variables, we again denote by $f : \mathbb{R}^{l+m} \to \mathbb{R}^n$ the function defined by the components $(f_1, \ldots, f_n)$ and $g : \mathbb{R}^{l+m} \to \mathbb{R}^k$ the function defined by the components $(g_1, \ldots, g_k)$.

Theorem 7 *The algorithm* $\mathrm{SoEI}(S, P, r)$ *described in Section 3.1 fulfills the definiteness property of the specification of* CheckSat *(defined at the beginning of Section 3).*

Proof.

Let P_0 be an l-box bounding the free variables of S. We divide the proof into two parts:

Negative case:

Assume that $\exists x \in B \; . \; f(p_0) = 0 \;\wedge\; g(p_0) \geq 0$ is robustly false for each $p_0 \in P_0$. We construct an $\varepsilon > 0$ such that for every $r \leq \varepsilon$ and every sub-box $P \subseteq P_0$ with width smaller than r, the algorithm returns $\{\mathbf{F}\}$:

The sets $X = \{(p, x) \in P_0 \times B \mid f(p, x) = 0\}$ and $Y = \{(p, x) \in P_0 \times B \mid g(p, x) \geq 0\}$ are compact and disjoint, so they have a positive distance. For a small enough $\alpha > 0$, the sets $X' = \{(p, x) \in P_0 \times B \mid |f(p, x)| \leq \alpha\}$ and $Y' = \{(p, x) \in P_0 \times B \mid g(p, x) \geq (-\alpha, \ldots, -\alpha)\}$ are still disjoint and have a positive distance $d > 0$.[1] If ε_0 is small enough, any box of width smaller than ε_0 either has an empty intersection with X' or an empty intersection with Y'.

The second property of interval computability implies that for α there exists a $\delta > 0$ such that any box $A \subseteq B$ with $\mathrm{width}(A) < \delta$ and box $P \subseteq P_0$ with $\mathrm{width}(P) < \delta$ have the following properties:

- If $P \times A$ has empty intersection with X', then $0 \notin \mathcal{I}(f)(P \times A)$.

- If $P \times A$ has empty intersection with Y', then $\mathcal{I}(g)(P \times A) \cap [0, \infty)^k = \emptyset$.

So, if we call the CheckSat algorithm with $r \leq \varepsilon := \min\{\delta, \varepsilon_0\}$ and $P \subseteq P_0$ of width smaller than r, then for every $A \subseteq B$ in the resulting S_r-grid, either $P \times A$ has empty intersection with X' or it has empty intersection with Y' and due to the above properties, A satisfies that $0 \notin \mathcal{I}(f)(P \times A)$ or $\mathcal{I}(g)(P \times A) \cap [0, \infty)^k = \emptyset$. So the test at Line 3 of the algorithm succeeds

---

[1]This follows from the fact that X resp. Y can be separated by open ε'-neighborhoods $U(X)$ resp. $U(Y)$ with positive distance from each other, and the fact that using the uniform continuity of $|f|$ and g, $X' \subseteq U(X)$ and $Y' \subseteq U(Y)$ for α small enough.

and the algorithm terminates with $\{\mathbf{F}\}$.

Positive Case:

Assume now that $\exists x \in B . f(p_0) = 0 \wedge g(p_0) \geq 0$ is robustly true for each $p_0 \in P_0$. We prove that there exists an $\varepsilon > 0$ such that for every $r \leq \varepsilon$ and every sub-box $P \subseteq P_0$ with width smaller than r, the algorithm returns $\{\mathbf{T}\}$.

Exploiting that our given set of functions symbols allows us to form polynomials with rational coefficients, it follows that for some $\alpha > 0$, each α-perturbation $\tilde{f}$ of $f(p_0)$ and $\tilde{g}$ of $g(p_0)$ such that each component of $\tilde{f}$ and of $\tilde{g}$ is a polynomial with rational coefficients, satisfies that $\exists x \in B . \tilde{f} = 0 \wedge \tilde{g} \geq 0$ is true. In particular, each polynomial α-perturbation of $f(p_0) = 0$ with rational coefficients has a root in the compact set $C := \{x \in B \mid g(p_0, x) \geq \alpha\}$.

Now we show that $m = n$. Otherwise $m < n$ and by Lemma 5 there exist arbitrary close continuous perturbations $\tilde{f}$ of $f(p_0)$ with no root in C. The absolute value of each such $\tilde{f}$ has a positive minimum on C and arbitrary close to $\tilde{f}$ are rational polynomials with no root in C. But then arbitrary close to $f(p_0)$ would be polynomials with no root in C which contradicts our assumption. Therefore, $m = n$.

We will now prove that for all $p_0 \in P_0$ there is an open neighborhood $U(p_0)$ of p_0 and $\varepsilon(p_0) > 0$ such that for all $P' \subseteq U(p_0)$, SoEI$(S, P', \varepsilon(p_0))$ terminates with $\{\mathbf{T}\}$. So let $p_0 \in P_0$ be arbitrary, but fixed, for which we will now construct such a $U(p_0)$ and $\varepsilon(p_0)$.

Let $\Omega_1 \subseteq B$ be an open neighborhood of C in B such that $\Omega_1 \subseteq \{x \in B \mid g(p_0, x) \geq \alpha/2\}$ and let Ω be the interior of $\bar{\Omega}_1$ in $\mathbb{R}^n$. We already know that each small enough polynomial perturbation of $f(p_0)$ has a zero in $\bar{\Omega}$.

By construction, $\bar{\Omega} = \bar{\Omega}_1$ and $\bar{\Omega}$ is the closure of its interior, so we are now ready to use Theorem 6. It implies that there exists an open $U \subseteq \Omega$ such that $0 \notin f(p_0)(\partial U)$ and $\deg(f(p_0), U, 0) \neq 0$. Otherwise, by Theorem 6 there would exist continuous perturbations of $f(p_0)$ with no zero in $\bar{\Omega}$ arbitrary close to $f(p_0)$ and it easily follows that there would also exist rational polynomial perturbations arbitrary close to $f(p_0)$ with no zero in $\bar{\Omega}$.

While $\deg(f(p_0), U, 0) \neq 0$ and the inequalities of S strictly hold for all elements of $\{p_0\} \times U$, the set U is not a union of boxes, and hence the algorithm will, in general, not come up with this set. So our goal is now to construct $U(p_0)$ and $\varepsilon(p_0)$ in such a way that for all $P' \subseteq U(p_0)$, SoEI$(S, P', \varepsilon(p_0))$ approximates U closely enough for the degree test (Line 10 of the algorithm) and the test of inequality satisfaction (Line 12) to succeed.

Let $U(p_0) \subseteq P_0$ be an open neighborhood of $\{p_0\}$ in P_0 such that $(U(p_0) \times \Omega) \subseteq \{(p, x) \mid g(p, x) \geq \alpha/4\}^2$ and let $\varepsilon_g(p_0)$ be so small that for every box $K \subseteq U(p_0) \times \Omega$ of width less than $\varepsilon_g(p_0)$,

$$\mathcal{I}(g)(K) \subseteq (0, \infty)^k \tag{1}$$

which exists due to the second property of the definition of interval computability.

Possibly making $U(p_0)$ smaller, we may assume that $0 \notin f(\overline{U(p_0)} \times \partial U)$. Let $V \subseteq \Omega$ be a neighborhood of ∂U open in B such that $0 \notin f(\overline{U(p_0)} \times \bar{V})$ (in these constructions we exploit the compactness of ∂U, resp. $\overline{U(p_0)}$). We will further assume that $U(p_0)$ is connected (if it were not, we could replace it by the connected component of p_0 in $U(p_0)$). The compactness of $\overline{U(p_0)} \times \bar{V}$ implies that $|f|$ has a positive minimum on this set and the second property of the definition of interval computability implies that there exists an $\varepsilon_f(p_0)$ such that for every sub-box $K \subseteq U(p_0) \times V$ of width smaller than $\varepsilon_f(p_0)$,

$$0 \notin \mathcal{I}(f)(K). \tag{2}$$

Let $\varepsilon_V(p_0)$ be such that each box of width less than $\varepsilon_V(p_0)$ that has a nonempty intersection with ∂U lies in V. Let $\varepsilon(p_0)$ be $\min\{\varepsilon_f(p_0), \varepsilon_g(p_0), \varepsilon_V(p_0)\}$.

Having constructed $U(p_0)$ and $\varepsilon(p_0)$ we will now show that they are indeed small enough for the algorithm to return a positive result: Let $P' \subseteq U(p_0)$ be a box of width at most $\varepsilon(p_0)$. We will show that $\mathrm{SoEI}(S, P', \varepsilon(p_0))$ terminates with $\{\mathbf{T}\}$. The algorithm creates a grid of boxes S_r such that each grid element has width at most $\varepsilon(p_0)$. It merges boxes containing a face C such that $0 \in \mathcal{I}(f)(P' \times C)$ and removes elements (i.e. merged boxes) containing a face $C \subseteq \partial B$ such that $0 \in \mathcal{I}(f)(P' \times C)$. Let us denote by S_r^{m} the set containing all these merged boxes after the removal. So, elements of S_r^{m} can be identified with unions of boxes in S_r. Let M be the smallest union of elements in S_r^{m} such that $M \supseteq U$. M consists of unions of boxes in S_r that are either contained in U or intersect ∂U and hence are contained in V. It follows that $M \subseteq \Omega$ (by a slight abuse of notation, we denote by M both the set of elements as well as the underlying space). Further, $\partial M \subseteq V$, $0 \notin \mathcal{I}(f)(P' \times C)$ for any boundary box $C \subseteq \partial M$ (due to (2)) and

$$\deg\left(f(p'), M^\circ, 0\right) = \deg\left(f(p_0), M^\circ, 0\right) = \deg\left(f(p_0), U, 0\right) \neq 0$$

for any $p' \in P'$. The first identity follows from the fact that $U(p_0)$ is connected, hence p_0 and p' can be connected by a curve that gives rise to a homotopy between $f(p')$ and $f(p_0)$ that is nowhere zero on the boundary faces of M, see axiom 3 defining the degree in Section 4. The second identity

---

[2]The set $\{(p, x) \mid g(p, x) > \alpha/4\}$ is an open neighborhood of $\{p_0\} \times \bar{\Omega}$ and the compactness of $\bar{\Omega}$ implies that there is a neighborhood $U(p_0)$ of $\{p_0\}$ such that $U(p_0) \times \Omega \subseteq \{(p, x) \mid g(p, x) > \alpha/4\}$.

follows from the fact that $0 \notin f(M \setminus U)$ and axiom 4 of Section 4 applied to $\bar{\Omega} = M$, $\Omega_1 = U$ and $\Omega_2 = \emptyset$.

Let $p' \in P'$ be chosen in the algorithm. There exists a subset $M' \subseteq M$ that consists of elements in S_r where the algorithm finds that $\deg{(f(p'), (M')^{\circ}, 0)} \neq 0$ (otherwise, M would be a union of subsets on which $f(p')$ has zero degree, contradicting $\deg{(f(p'), M^{\circ}, 0)} \neq 0$). Then it splits elements of M' back to the corresponding elements in S_r and checks the condition whether for all boxes $E \in S_r(M')$, $\mathcal{I}(g)(P' \times E) \subseteq (0, \infty)^k$. This is satisfied due to (1) and the algorithm terminates with $\{\mathbf{T}\}$.

So we now know that for all $p_0 \in P_0$, there is an $U(p_0)$ and $\varepsilon(p_0) > 0$ such that for all $P' \subseteq U(p_0)$, $\mathrm{SoEI}(S, P', \varepsilon(p_0))$ terminates with $\{\mathbf{T}\}$. So, we have a covering $\{U(p_0) \,|\, p_0 \in P_0\}$ of the compact set P_0 and can choose a finite sub-covering $\{U(p_1), \dots, U(p_s)\}$. There exists an ε' such that each box $P \subseteq P_0$ of width smaller than ε' is contained in some $U(p_j)$. Let ε be the minimum of ε' and all the $\varepsilon(p_j)$, $j \in \{1, \dots, s\}$. For any $P \subseteq P_0$ of width at most ε, $\mathrm{SoEI}(S, P, \varepsilon)$ terminates with a positive result $\{\mathbf{T}\}$. ∎

7.2   Universal quantifiers

Theorem 8 *Let S be a formula and let I be a closed interval. Let P be an l-box bounding the free variables of the formula $\forall x \in I \,.\, S$. Assume that an algorithm* CheckSat *fulfilling the definiteness property is given. Then also the algorithm* $\mathrm{Univ}(\forall x \in I \,.\, S, P, r)$ *described in Section 3.2 fulfills the definiteness property.*

Proof.   Assume that for all $p_0 \in P_0$, the sentence $\forall x \in I \,.\, S[p \leftarrow p_0]$ is robustly true. Then, by Lemma 4, for all $p_0 \in P_0$ and all $x_0 \in I$, $S[p \leftarrow p_0][x \leftarrow x_0]$ is robustly true and the property follows directly from the assumption on CheckSat.

Assume now that for all $p_0 \in P_0$, $\forall x \in I \,.\, S[p \leftarrow p_0]$ is robustly false. Let $p_0 \in P_0$. Then there exists a $x_0 \in I$ such that $S[p \leftarrow p_0][x \leftarrow x_0]$ is false, and hence, due to Lemma 3, it is also robustly false. From this, Lemma 2 implies that there is a neighborhood $P(p_0)$ of p_0 and I_0 of x_0 such that for all $p_0' \in P(p_0)$ and $x_0' \in I_0$, $S[p \leftarrow p_0'][x \leftarrow x_0']$ is false. It follows from the assumption on CheckSat that there exists an ε_{p_0} such that for all $P' \subseteq P(p_0)$, $I' \subseteq I$ of width at most ε_{p_0}, $\mathrm{CheckSat}(P' \times I', S, \varepsilon_{p_0})$ terminates with $\{\mathbf{F}\}$.

Because P_0 is compact, we can cover it by $\{P(p_0); \, p_0 \in \Lambda\}$ for a finite set Λ. It is easy to see that there exists an ε' such that any box of side-length smaller than ε' is in at least one of these $P(p_0)$. Now, choose ε to be smaller than ε' and smaller than ε_{p_0} for all $p_0 \in \Lambda$. For any box P of side-length at most ε, the algorithm $\mathrm{Univ}(\forall x \in I \,.\, S, P, \varepsilon)$ terminates with $\{\mathbf{F}\}$. ∎

7.3 Conjunction and Disjunction

Theorem 9 *Let S and T be two formulas in $\mathcal{B}$ and assume that* CheckSat *fulfills the definiteness property both when applied to S and when applied T. Then* $\mathrm{Conj}(S \wedge T, P, r)$ *(described in Section 3.3) also fulfills the definiteness property.*

Proof. Let p_S, and p_T, respectively, be the function that projects any l-tuple corresponding to the free variables of $S \wedge T$ to those components corresponding to the free variables of S, and T, respectively.

We first assume that for all $p_0 \in P_0$ the sentence $(S \wedge T)[p \leftarrow p_0]$ is robustly true. Then for all $p_0 \in P_0$, $S[p_S(p) \leftarrow p_S(p_0)]$ is robustly true and for all $p_0 \in P_0$, $T[p_T(p) \leftarrow p_T(p_0)]$ is robustly true. So, by definiteness of the recursive call, there exists an $\varepsilon_1 > 0$ such that if $r \leq \varepsilon_1$ and the width of $P_1 \subseteq P_0$ is less than r, then $\mathrm{CheckSat}(S, p_S(P_1), r)$ terminates with $\{\mathbf{T}\}$. An analogous ε_2 exists for T. For $\varepsilon < \min\{\varepsilon_1, \varepsilon_2\}$, $r \leq \varepsilon$ and $P \subseteq P_0$ of width less than r, $\mathrm{Conj}(S \wedge T, P, r)$ terminates with $\{\mathbf{T}\}$.

Suppose that for all $p_0 \in P_0$, $(S \wedge T)[p \leftarrow p_0]$ is robustly false. Then, for any $p_0 \in P_0$, either $S[p_S(p) \leftarrow p_S(p_0)]$ or $T[p_T(p) \leftarrow p_T(p_0)]$ is robustly false. Let $p_0 \in P_0$. Assume, without loss of generality, that $S[p_S(p) \leftarrow p_S(p_0)]$ is robustly false. By Lemma 2 there exists a neighborhood U of p_0 such that for every $u \in U$, $S[p_S(p) \leftarrow p_S(u)]$ is robustly false. Let $P(p_0) \subseteq P$ be a box neighborhood of p_0 contained in the interior of U. By assumption, there exists an $\varepsilon_{p_0} > 0$ such that if $r \leq \varepsilon_{p_0}$ and $P' \subseteq P(p_0)$ has width at most r, then $\mathrm{CheckSat}(S, p_S(P'), r)$ terminates with $\{\mathbf{F}\}$, hence $\mathrm{Conj}(S \wedge T, P', r)$ terminates with $\{\mathbf{F}\}$ as well.

This can be done for each $p_0 \in P_0$. Let $P(p')^\circ$ be the interior of $P(p')$ in the topology of the box P_0. Then $\{P(p')^\circ \mid p' \in P_0\}$ is an open cover of the compact space P_0 and there exists a finite subcovering $\{P(p_1)^\circ, \ldots, P(p_m)^\circ\}$ of P_0. Take ε to be so small that each box $P \subseteq P_0$ of width at most ε is contained in some $P(p_j)$ and $\varepsilon < \min_i \varepsilon_{p_i}$. Then $\mathrm{Conj}(S \wedge T, P, r)$ terminates with $\{\mathbf{F}\}$ for any $r \leq \varepsilon$ and any box P of width at most r.

For disjunctions the situation is analogous.

Together with the correctness proof from Section 5 this concludes the proof of Theorem 1.

8 Limitations on Generalization

We showed in Lemma 5 that an overdetermined system of equations $(m < n)$ never has a robust solution. In the underdetermined case $(m > n)$, in some cases, we could fix $m - n$ input variables in f to constants $a \in \mathbb{R}^{m-n}$ and try to analyze the formula $\exists x \in \bar{\Omega}^a . f(a, x) = 0$, where $\bar{\Omega}^a = \{x \in \mathbb{R}^n \mid (a, x) \in \bar{\Omega}\}$. If $f(a, \cdot)$ has a robust zero in $\bar{\Omega}^a$, then f has a robust zero in $\bar{\Omega}$. However,

the converse is not true: If $f(a, \cdot)$ does not have a robust zero in $\bar{\Omega}^a$ for any fixed choice of $a \in \mathbb{R}^{m-n}$ (the components of a ranging over all $(m-n)$-subsets of the total number of m variables), f still may have a robust zero in $\bar{\Omega}$.

Indeed, Theorem 2 states that a generalization to the underdetermined case is (under certain weak conditions) impossible, and we will spend the rest of this section to prove this theorem. If Q is a quasi-decision procedure (Def. 4) and $\mathcal{I}$ an algorithmic assignment of $\mathcal{I}(f)$ to all function symbols f, we will denote by $Q_\mathcal{I}$ the algorithm that takes a sentence φ and returns $Q(\varphi, \mathcal{I})$. We need the following:

Lemma 8 *Assume that there exists a quasi-decision procedure Q for some class of formulas such that each function symbol appears in each formula at most once, and such that each term in each formula consists of one single function symbol. Assume that the quasi-decision procedure has access only to the oracle $\mathcal{I}(f)$ for each function symbol f in the formula.*[3]

Then there exists an algorithmic assignment $\mathcal{I}'(f)$ to all function symbols f such that $Q_{\mathcal{I}'}(\varphi)$ terminates if and only if φ is robust.

Proof. Let us define the addition of boxes naturally by $B_1 + B_2 := \{b_1 + b_2 : b_1 \in B_1, b_2 \in B_2\}$. For every function symbol f corresponding to a function $B \to \mathbb{R}^n$, let $\mathcal{I}'(f)$ be the algorithm defined by $\mathcal{I}'(f)(B') := \mathcal{I}(f)(B') + [-\text{width}(B'), \text{width}(B')]^n$. This algorithm is a modification of $\mathcal{I}(f)$, it still represents the function f and satisfies the assumptions of Definition 1. However, for any box $B' \subseteq B$, the output $\mathcal{I}'(f)(B')$ contains $f(B')$ in its interior. We will show that $Q_{\mathcal{I}'}$ terminates if and only if the input is robust.

By definition of Q, $Q_{\mathcal{I}'}(\varphi)$ terminates whenever φ is a robust sentence. It remains to prove that it does not terminate for inputs that are not robust. Let φ be a fixed non-robust sentence. For proving that $Q_{\mathcal{I}'}(\varphi)$ does not terminate, we assume that it does terminate and derive a contradiction.

$Q_{\mathcal{I}'}(\varphi)$ only uses a finite number of evaluations of $\mathcal{I}'(f)(B)$ with f being a function in φ. Let $\tilde{\varphi}$ be a perturbation of φ in which each function f is replaced by an interval computable function $\tilde{f}$ representable by a term in our first-order language such that

- $\tilde{\varphi}$ and φ have different truth values,

- $\tilde{f}(B) \subseteq \mathcal{I}'(f)(B)$ for every B used by $Q_{\mathcal{I}'}(\varphi)$ in a call to $\mathcal{I}'(f)(B)$.

Such functions exist, because φ is non-robust, $\mathcal{I}'(f)(B)$ contains $f(B)$ in its interior and arbitrarily close to f are other functions representable by a term

---

[3]That is, it may call $\mathcal{I}(f)$ with any input an arbitrary number of times, but apart from the results of calling $\mathcal{I}(f)$ it does not use any properties of f, nor does it analyze how $\mathcal{I}(f)$ is computed.

in our first-order language. Now, let $\mathcal{I}''$ be equal to $\mathcal{I}$ with the exception that for every $\tilde{f}$ occurring in $\tilde{\varphi}$, $\mathcal{I}''(\tilde{f})(B) :=$

- $\mathcal{I}'(f)(B)$, for every box B used by $Q_{\mathcal{I}'}(\varphi)$ in a call to $\mathcal{I}'(f)(B)$, and

- $\mathcal{I}(\tilde{f})(B)$, otherwise.

$\mathcal{I}''$ still satisfies both axioms of Definition 1. All function symbols in both φ and $\tilde{\varphi}$ are mutually different and both $Q_{\mathcal{I}'}$ and $Q_{\mathcal{I}''}$ do not use any other information about the function symbols in φ and $\tilde{\varphi}$ than the evaluations $\mathcal{I}'(f)$ and $\mathcal{I}''(\tilde{f})$, respectively.

However, for every call $\mathcal{I}''(\tilde{f})(B)$ of $Q_{\mathcal{I}''}(\tilde{\varphi})$, and corresponding call $\mathcal{I}'(f)(B)$ of $Q_{\mathcal{I}'}(\varphi)$, $\mathcal{I}''(\tilde{f})(B) = \mathcal{I}'(f)(B)$. Hence $Q_{\mathcal{I}''}(\tilde{\varphi})$ uses exactly the same information about its input as $Q_{\mathcal{I}'}(\varphi)$ and they have to return the same result. But this is impossible, because φ and $\tilde{\varphi}$ have different truth values.

Therefore, $Q_{\mathcal{I}'}$ does not terminate whenever the input is non-robust. ∎

For proving Theorem 2 we use a reduction from a recent undecidability result [16, p. 19]. For this we introduce the following notions: A *triangulation* of the box $[0,1]^d$, with $d \in \mathbb{N}$, is a subdivision of $[0,1]^d$ into a finite set S of simplices such that the intersection of any two simplices in S is again a simplex (possibly empty) in S. A *piecewise linear function* from $[0,1]^d$ to $\mathbb{R}^d$ is a function that is linear on each simplex of some triangulation. It is uniquely determined by values on the vertices of the simplices. If the simplices have rational coordinates and the values of f on the vertices are all rational, then f is interval computable; moreover, for any box $B \subseteq [0,1]^d$ with rational vertices, the image $f(B)$ can be computed exactly by means of linear programming. We summarize the statement given in [16, Inequalities, Section 4].

Theorem 10 *There is no algorithm with the following specification:*

Input:

- $n, d \in \mathbb{N}$,
- *T, a triangulation of $[0,1]^d$ with rational vertices*
- *$(f,g): [0,1]^d \to \mathbb{R}^n \times \mathbb{R}$, piecewise linear with rational values on vertices of T*

Output: *At least one correct answer from the following two options:*

- *$\exists x \in [0,1]^d \, . \, f(x) = 0 \wedge g(x) \leq 0$ is robustly true,*
- *Some 1-perturbation of $\exists x \in [0,1]^d \, . \, f(x) = 0 \wedge g(x) \leq 0$ is false.*

In the cited theorem, the notion of "robustly true" means that for some $\varepsilon > 0$, for *arbitrary continuous* functions $\tilde{f}_i$ and $\tilde{g}$ such that $\|\tilde{f}_i - f\| < \varepsilon$ and $\|\tilde{g} - g\| < \varepsilon$, it holds that the sentence $\exists x \in [0,1]^d . \ \tilde{f}(x) = 0 \wedge \tilde{g}(x) \leq 0$ is true, not only for interval computable functions from a specified language. However, if our first-order language contains all piecewise linear functions with rational values on rational vertices, then both notions of robustness are equivalent. This can be shown as follows:

Assume that all piecewise linear ε-perturbations $\tilde{f}_i^{PL}, \tilde{g}^{PL}$ of f_i, g satisfy $\exists x \in [0,1]^d \ \tilde{f}^{PL}(x) = 0 \wedge \tilde{g}^{PL}(x) \leq 0$, and for some continuous $(\varepsilon/2)$-perturbations f_i', g_i' the sentence $\exists x \in [0,1]^d \ f'(x) = 0 \wedge g'(x) \leq 0$ is false. Then the last sentence is also "robustly false" by the remarks after Lemma 3: "robustly false" here means that any small enough continuous perturbation is false (note that f' and g' doesn't need to be interval computable). However, arbitrary close to f' and g' are some piecewise linear functions, which contradicts our assumption that any piecewise linear ε-perturbation of $\exists x \in [0,1]^d \ f(x) = 0 \wedge g(x) \leq 0$ is true. Therefore, both notions of being robustly true are equivalent and we do not need to distinguish them further.

Further, Theorem 10 still holds, if we assume that the function symbols $\{f_1, \ldots, f_n, g\}$ in the input formula

$$\exists x \in [0,1]^d \ f_1 = 0 \wedge \ldots \wedge f_n = 0 \wedge g \leq 0 \tag{3}$$

are all pairwise different, and that the perturbations consist of formulas in which all functions are pairwise different. This can be seen as follows:

If, in formula (3), two functions f_i and f_j or f_i and g coincide, we can easily construct an arbitrary small perturbation of (3) that is false, because each component can be perturbed independently. So, without loss of generality, we may assume that all function symbols in the input of Theorem 10 are different. It can easily be shown that the sentence (3) is robustly true if and only if for some $\varepsilon > 0$, each ε-perturbation

$$\exists x \in [0,1]^d \ \tilde{f}_1 = 0 \wedge \ldots \wedge \tilde{f}_n = 0 \wedge \tilde{g} \leq 0$$

in which all $n+1$ functions $\tilde{f}_j, \tilde{g}$ are different, is true. Similarly, some 1-perturbation is false, if some 1-perturbation in which all functions are different, is false. Summarizing the previous paragraphs, we obtain the following consequence:

Lemma 9 *Assume that we have a language containing function symbols for all piecewise linear functions on rational triangulations of $[0,1]^d$ with rational values on vertices, and the class of all sentences $\mathcal{A}$ of the type (3) such that in each sentence, all function symbols are different. Then there is no algorithm with the following specification:*

Input:

- *A sentence φ from $\mathcal{A}$.*

Output: *At least one correct answer from the following two options:*

- *φ is robustly true wrt. the class $\mathcal{A}$*
- *Some 1-perturbation of φ from $\mathcal{A}$ is false.*

Now we are ready to prove Theorem 2:

Proof. [of Theorem 2.] We will assume that a quasi-decision procedure for the class of sentences defined in Theorem 2 exists, and derive a contradiction. Let us call the assumed quasi-decision procedure Q. We prove that the existence of Q implies the existence of an algorithm solving the undecidable problem from Lemma 9. For this we will first (Step 1) construct an algorithm computing positive information, then (Step 2) an algorithm computing negative information, and finally (Step 3) run them in parallel to get an algorithm specified in Lemma 9.

Step 1. First we show that the existence of Q implies the existence of an algorithm with input as in Lemma 9 such that it terminates iff $\exists x \in [0,1]^d \,.\, f = 0 \wedge g \leq 0$ is robustly true.

We can easily construct an algorithm assigning to each piecewise linear function $f : [0,1]^d \to \mathbb{R}$ with rational values on the vertices an algorithm $\mathcal{I}(f)$ satisfying the axioms in Definition 1. From the quasi-decision procedure Q for general systems of equations and inequalities we get an algorithm $Q_{\mathcal{I}}$ that takes an input such as in Lemma 9 and decides whether it is robustly true or not, whenever the input is robust. By Lemma 8, we can algorithmically replace $\mathcal{I}$ by $\mathcal{I}'$ and obtain an algorithm $Q_{\mathcal{I}'}$ that terminates iff the input is robust. This procedure can be modified such that instead of terminating with $\{\mathbf{F}\}$, it runs forever. The result is an algorithm that terminates if and only if the input $\exists x \in [0,1]^d \, f = 0 \wedge g \leq 0$ is robustly true.

Step 2. Now we show that there exists an algorithm with input such as in Lemma 9 with the following specification:

- if some $1/2$-perturbation of $\exists x \in [0,1]^d \,.\, f = 0 \wedge g \leq 0$ is false, then it terminates, and

- if it terminates, then some 1-perturbation of the above formula is false.

This algorithm can be described as follows: In the i-th step, it constructs the i-th barycentric subdivision $T^{(i)}$ of the given triangulation T, and further constructs all piecewise linear functions f', g' on this subdivision such that their values on the vertices $\{v_k\}_k$ of $T^{(i)}$ are rational with denominators at most i and such that for each k, the values $f_i'(v_k)$ resp. $g'(v_k)$ differ from $f(v_k)$ resp. $g(v_k)$ by less than 1. For all such piecewise linear functions f', g', the truth value of $\exists x \in [0,1]^d \,.\, f' = 0 \wedge g' \leq 0$ can be computed. Moreover, due to the restriction on the denominators of the values on the vertices,

there exists only a finite number of such functions. So, for all those finitely many f' and g', the algorithm checks whether $\exists x \in [0,1]^d \ . \ f' = 0 \wedge g' \le 0$ is false and terminates as soon as it finds a pair (f', g') for which the formula is false.

In the rest of step 2 of the proof we show that this algorithm satisfies the above specification.

The absolute value $|f_i - f'_i|$ of a linear function $f_i - f'_i$ is a convex function on each simplex $\Delta \in T^{(i)}$, so on each simplex it attains its maximum on a vertex. Therefore, a piecewise linear function f' is a 1-perturbation of f iff its restriction to the vertices is a 1-perturbation of the restriction of f. It follows that $f' = 0 \wedge g' \le 0$ is a 1-perturbation of $f = 0 \wedge g \le 0$ if and only if the differences $|f_i(v_k) - \tilde{f}_i(v_k)| \le 1$ and $|g_i(v_k) - \tilde{g}_i(v_k)| \le 1$ for all i and all vertices v_k. Assume that $f, g : [0,1]^d \to \mathbb{R}^n \times \mathbb{R}$ are piecewise linear on a given triangulation T of $[0,1]^d$ and that some $1/2$-perturbation of $f = 0 \wedge g \le 0$ is unsatisfiable. Each continuous function can be approximated arbitrarily precisely by some piecewise linear function on an iterated barycentric subdivision. So, there exists an iterated barycentric subdivision $T^{(i)}$ of T and piecewise linear functions $\tilde{f}, \tilde{g}$ on $T^{(i)}$ such that $\exists x \in [0,1]^d \ . \ \tilde{f} = 0 \wedge \tilde{g} \le 0$ is a false 1-perturbation of $\exists x \in [0,1]^d \ . \ f = 0 \wedge g \le 0$. The algorithm finds this perturbation in its ith step and terminates.

Conversely, if the algorithm terminates, then it had found a false 1-perturbation of $\exists x \in [0,1]^d \ . \ f = 0 \wedge g \le 0$.

Step 3. Finally, we show that the existence of Q contradicts Lemma 9. Given piecewise linear functions $(f, g) : [0,1]^d \to \mathbb{R}^n \times \mathbb{R}$ with non-repeating function symbols and the quasi-decision procedure Q for systems of equations and inequalities, we could run an algorithm specified in Step 1 that terminates if and only if $\exists x \in [0,1]^d \ . \ f = 0 \wedge g \le 0$ is robustly true. Further, by Step 2, we could run another algorithm that terminates whenever some $1/2$-perturbation of this formula is false. A formula is either robustly true, or has a false $1/2$-perturbation, so at least one of these algorithm would always terminate. If the first algorithm terminates, we know that the formula is robustly true and if the second one terminates, we know that some 1-perturbation is false. Thus, we could choose at least one correct answer from the output specified in Lemma 9, which is impossible. ∎

9   Related Work

From the very beginning of engineering the notion of robustness has played a key role. This is being recognized more and more in several scientific fields: For example, the field of robust control [45, 6] is now considered as a central subject of control engineering. Robustness also plays an increasingly important role in applied and computational mathematics, as shown by

the emerging fields of robust optimization [5] and uncertainty quantification (with a journal of the same name recently having been launched by SIAM).

Also in the computing field, robustness has been a core issue from the very beginning. In computer systems design this is usually captured by the keyword "fault-tolerance" and for numerical algorithms "stability". Robustness also plays an important role in computational geometry [44].

In the complexity analysis of algorithms, the notion of perturbation has helped to explain the good practical behavior of algorithms with exponential worst-case complexity [40]. The present paper in analogy applies Spielman and Deng's [41] motivation "The basic idea is to identify typical properties of practical data, define an input model that captures these properties, and then rigorously analyze the performance of algorithms assuming their inputs have these properties" to undecidable problems, where the main goal then is not performance analysis but finding a terminating algorithm.

Apparently, the first paper that follows this approach of ensuring termination of an algorithm for all robust inputs to an undecidable problem (in this case safety verification of hybrid systems) is due to Fränzle [19]. Since then, a similar approach has been applied several times [19, 20, 35, 37, 13, e.g.] to problems in formal verification.

To the best of our knowledge, the first paper to apply such an approach to decision procedures for the real numbers is by one of the co-authors [34, Theorem 5] (see also [37, Theorem 6]), based on an analysis of robustness of first-order formulas [35]. The main difference to the present paper and—at the same time—main weakness is, that it expresses equalities of the form $f(x) = 0$ as a conjunction of two equalities of the form $f(x) \leq 0 \wedge -f(x) \leq 0$ which—in general—loses robustness, since the two occurrences of f can be perturbed independently and a solution of $f(x) = 0$ can vanish under perturbations of $f(x) \leq 0 \wedge -f(x) \leq 0$. Hence, the corresponding algorithm need not necessarily terminate in such cases of satisfiable equalities.

Recently, Gao et. al. [22] took a similar approach: They model perturbations of formulas by the notion of δ-strengthening which roughly means that inequalities of the form $f \geq 0$ are replaced by inequalities of the form $f \geq \delta$, where $\delta > 0$. However, instead of allowing non-termination in non-robust cases, the approach uses the notion of δ-decidability that requires an algorithm to terminate always, but either decides that the input formula is true, or that a δ-strengthening of the input formula is false. These two answers overlap, and especially for non-robust inputs, for every $\delta > 0$, a δ-strengthening of the input is false, and hence both answers are allowed.

Since δ-decidability cannot give a definite answer for inputs that are false, δ-decidability does not imply quasi-decidability, in general. However, it does imply quasi-decidability for classes of formulas that are closed under negation, since then it is possible to run the corresponding algorithm in parallel on both the input formula and its negation. It would be an easy extension of the algorithm in this paper to return also quantitative information

on robustness (i.e. a value $\varepsilon \in \mathbb{R}$ s.t. the input is ε-robust).

Gao and co-authors handle equalities of the form $t = 0$ as the non-robust formula $-|t| \geq 0$. Hence—in contrast to the present paper—their approach cannot prove equalities to have a solution. For example, it cannot prove that $\exists x \in [-10, 10] . \ x = 0$ is true since this formula is handled as the non-robust formula $\exists x \in [-10, 10] . \ -|x| \geq 0$ for which, for every $\delta > 0$, the δ-strengthening $\exists x \in [-10, 10] . \ -|x| \geq \delta$ is false. And indeeed, in such cases the approach returns that a δ-strengthening of the formula is false. The paper [22] also studies complexity of such algorithms in some model of computable analysis [8]. Another paper [23] studies δ-decidability in a satisfiability modulo theory (SMT) context, where the approach either returns "unsatisfiable" or "a δ-weakening is satisfiable" with the notion of δ-weakening defined in analogy to δ-strengthening.

Due to the fact that those approaches [37, 22, 23] do not handle equalities directly, but reformulated as inequalities, those algorithms that do not need to, and in fact do not exploit continuity of the involved functions. In contrast to that, in the present paper we use the topological degree as the notion that captures the essential information about the roots of continuous functions under continuous perturbations.

All those approaches can depend on the precise way perturbations of first-order formulas are modeled. There are various possibilities for this, and some have been compared [35], but a comprehensive exploration of this is still missing.

The approach of relaxing the semantics of first-order formulas can be taken even further than just relaxing the dichotomy satisfiable/unsatisfiable. For example, one can weaken the necessity of distinguishing between close values [10], or introduce quantifiers that are weaker than the classical ones [36].

Collins [12] presents similar result to ours for the special case of systems of n equalities in n variables, formulated in the language of computable analysis [8]. However, the paper contains only very rough proof sketches, that we were not able to complete into full proofs.

Franek and Krčál study [16] the problem whether or not each continuous r-perturbation of a system $f(x) = 0$ has a solution or not, where $f : K \to \mathbb{R}^n$ is a piecewise linear function defined on a finite simplicial complex K. This turns out to be decidable whenever $\dim K < 2n - 2$ or n is even and undecidable for a fixed odd $n \geq 3$ and arbitrary K.

Verification of zeros of systems of equations is a major topic in the interval computation community [30, 39, 26, 21]. However, here people are usually not interested in some form of completeness of their methods, but in usability within numerical solvers for systems of equations or global optimization.

Basic existence theorems that are commonly used for proving that an equation $f = 0$ has a solution in B are Kantorovich's, Miranda's and Borsuk's theorem. Among these Borsuk's theorem is the strongest [3, 21], that

is, if the assumptions of the other theorems are fulfilled, then the assumptions of Borsuk's theorem are fulfilled as well.

We will now recall Borsuk's theorem and then compare its power for proving existence of a zero with that of the use of the topological degree:

Theorem 11 (Borsuk's theorem) *If $B \subseteq \mathbb{R}^n$ is open, bounded, convex and symmetric with respect to an interior point x, $f : \bar{B} \to \mathbb{R}^n$ is continuous and non-zero on the boundary ∂B and if for any $x + y \in \partial B$ and $\lambda > 0$,*

$$f(x + y) \neq \lambda f(x - y),$$

then $f = 0$ has a solution in B.

It can be shown that if the assumption of Miranda's theorem are satisfied, then the degree has to be 1 or -1 and if the assumption of Borsuk's theorem are satisfied, then the degree $\deg(f, B, 0)$ has to be an odd number[4]. On the other hand, if f has an isolated zero of even degree, then one cannot prove that using Borsuk's theorem. A simple illustration of this is the complex function $f(z) = z^2$ from $\mathbb{C} \simeq \mathbb{R}^2$ to itself, defined in a symmetric and convex neighborhood B of $0 \simeq (0, 0)$. This function has a robust zero in B and $\deg(f, B, 0) = 2$, so the assumptions of Borsuk's theorem are not fulfilled in any such neighborhood B.

An essential ingredient of our algorithm is the computation of the topological degree. Many papers deal with the question of an effective implementation, e.g. [15, 25, 7, 1, 17]. Our online package `TopDeg`[5] computes $\deg(f, B, 0)$ for a function f defined as an expression containing symbols such as polynomials and sin, and a low-dimensional box B. The degree can also be computed by the use of packages for *simplicial homology* computations, such as Chomp[6], GAP homology packes[7], or a collection of MATLAB routines PLEX [8]. However, to compute the degree with the use of these programs, one has to create first a simplicial approximation of $f/|f| : \partial B \to S^{n-1}$, which can be done by means of interval arithmetic.

A limitation of our approach is the fact that while in the context of real-world problems and engineering applications, the robustness assumption is natural, theorems with a purely mathematical motivation often fail to be robust. In such a context, the only option to automatize theorem proving of first-order sentences of the reals with function symbols such as sin is the

---

[4]This can be shown as follows. The function $\tilde{f} := f/|f| : \partial B \to S^{n-1}$ is homotopic to $g(x) := \frac{\tilde{f}(x) - \tilde{f}(-x)}{|\tilde{f}(x) - \tilde{f}(-x)|}$ via the homotopy $H(t, x) = \frac{\tilde{f}(x) - t\tilde{f}(-x)}{|\tilde{f}(x) - t\tilde{f}(-x)|}$, so $\tilde{f}$ and g have the same degree. Assumptions on B imply that $\partial B \simeq S^{n-1}$ and an odd map $g(-x) = -g(x)$ between spheres has odd degree [14, p. 180].

[5]`http://topdeg.sourceforge.net`

[6]`http://chomp.rutgers.edu`

[7]`http://www.linalg.org/gap.html`

[8]`http://comptop.stanford.edu/u/programs/plex/`

systematic usage of heuristics. This has been successfully implemented in the MetiTarski [2] package.

10   Conclusion

Motivated by the fact that in many application domains robustness is an essential property of formal models, we showed that for an undecidable class of first-order formulas over the real numbers one can algorithmically check satisfiability in all robust cases (under the additional assumption that all variables range over bound intervals). Moreover, we showed that it is not possible to generalize this result to the case without restrictions on the number of variables versus number of equations. Still, it might be possible to find a quasi-decision procedure for certain, specific numbers of variables versus equations. Moreover, it might be possible to find a quasi-decision procedure for a class of formulas with functions that are more specific than general interval computable.

The generalization to arbitrary existential quantification is hindered by the fact that the property that $\forall x \in I \ . \ S$ is robustly true if and only if for each $x_0 \in I$, the sentence $S[x \leftarrow x_0]$ is robustly true (Lemma 4) does not hold in analogy for existential quantifiers. The sentence $\exists x \in [-1, 1] \ . \ x = 0$ is robustly true but for any $x_0 \in [-1, 1]$, the sentence $x_0 = 0$ (x_0 is considered to be a constant function here) is not robustly true. A topological reformulation of adding an existence quantifier to the beginning of a formula would be desirable and could be a subject of future research.

It also remains an open problem to come up with an algorithm that is both a quasi-decision procedure and efficient in practice.

References

[1] O. Aberth. Computation of topological degree using interval arithmetic, and applications. *Mathematics of Computation*, 62(205):171–178, 1994.

[2] B. Akbarpour and L. C. Paulson. MetiTarski: An automatic theorem prover for real-valued special functions. *Journal of Automated Reasoning*, 44, 2010.

[3] G. Alefeld, A. Frommer, G. Heindl, and J. Mayer. On the existence theorems of Kantorovich, Miranda and Borsuk. *Electronic Transactions on Numerical Analysis*, 17:102–111, 2004.

[4] G. Baigger. Die Nichtkonstruktivität des Brouwerschen Fixpunktsatzes. *Archive for Mathematical Logic*, 25(1):183–188, 1985.

[5] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization.* Princeton Series in Applied Mathematics. Princeton University Press, October 2009.

[6] S. P. Bhattacharyya, H. Chapellat, and L. H. Keel. *Robust Control.* Prentice Hall, 1995. `http://www.ece.tamu.edu/~bhatt/books/robustcontrol/`.

[7] T. E. Boult and K. Sikorski. Complexity of computing topological degree of Lipschitz functions in n dimensions. *J. Complexity*, 2:44–59, 1986.

[8] V. Brattka, P. Hertling, and K. Weihrauch. A tutorial on computable analysis. In S. Cooper, B. Lwe, and A. Sorbi, editors, *New Computational Paradigms*, pages 425–491. Springer New York, 2008.

[9] A. M. Bruckner, J. B. Bruckner, and B. S. Thomson. *Real analysis.* Prentice Hall PTR, 1997.

[10] A. Casagrande, C. Piazza, and A. Policriti. Discrete semantics for hybrid automata. *Discrete Event Dynamic Systems*, 19(4):471–493, 2009.

[11] B. F. Caviness and J. R. Johnson, editors. *Quantifier Elimination and Cylindrical Algebraic Decomposition.* Springer, Wien, 1998.

[12] P. Collins. Computability and representations of the zero set. *Electron. Notes Theor. Comput. Sci.*, 221:37–43, December 2008.

[13] W. Damm, G. Pinto, and S. Ratschan. Guaranteed termination in the verification of LTL properties of non-linear robust discrete time hybrid systems. *International Journal of Foundations of Computer Science (IJFCS)*, 18(1):63–86, 2007.

[14] J. Dieudonné. *A History of Algebraic and Differential Topology, 1900 - 1960.* Modern Birkhäuser classics. Springer, 2009.

[15] P. J. Erdelsky. Computing the Brouwer degree in R^2. *Mathematics of Computation*, 27(121):pp. 133–137, 1973.

[16] P. Franek and M. Krčál. Robust satisfiability of systems of equations. In *Proc. Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA)*, 2014. Extended version in arXiv:1402.0858, to appear in JACM.

[17] P. Franek and S. Ratschan. Effective topological degree computation based on interval arithmetic. *Mathematics of Computation*, 84:1265–1290, 2015.

[18] P. Franek, S. Ratschan, and P. Zgliczynski. Satisfiability of systems of equations of real analytic functions is quasi-decidable. In *MFCS 2011: 36th International Symposium on Mathematical Foundations of Computer Science*, volume 6907 of *LNCS*, pages 315–326. Springer, 2011.

[19] M. Fränzle. Analysis of hybrid systems: An ounce of realism can save an infinity of states. In J. Flum and M. Rodriguez-Artalejo, editors, *Computer Science Logic (CSL'99)*, number 1683 in LNCS. Springer, 1999.

[20] M. Fränzle. What will be eventually true of polynomial hybrid automata. In N. Kobayashi and B. C. Pierce, editors, *Theoretical Aspects of Computer Software (TACS 2001)*, number 2215 in LNCS. Springer-Verlag, 2001.

[21] A. Frommer and B. Lang. Existence tests for solutions of nonlinear equations using Borsuk's theorem. *SIAM Journal on Numerical Analysis*, 43(3):1348–1361, 2005.

[22] S. Gao, J. Avigad, and E. Clarke. δ-decidability over the reals. In *LICS*, pages 305–314. IEEE, 2012.

[23] S. Gao, J. Avigad, and E. M. Clarke. δ-complete decision procedures for satisfiability over the reals. In *IJCAR*, volume 7364 of *Lecture Notes in Computer Science*, pages 286–300. Springer Berlin Heidelberg, 2012.

[24] M. Hirsch. *Differential topology*. Springer, 1976.

[25] B. Kearfott. An efficient degree-computation method for a generalized method of bisection. *Numerische Mathematik*, 32:109–127, 1979.

[26] R. B. Kearfott. On existence and uniqueness verification for non-smooth functions. *Reliable Computing*, 8(4):267–282, 2002.

[27] K.-I. Ko. Computational complexity of real functions. In *Complexity Theory of Real Functions*, Progress in Theoretical Computer Science, pages 40–70. Birkhuser Boston, 1991.

[28] J. W. Milnor. *Topology from the differential viewpoint*. Princeton University Press, 1997.

[29] R. E. Moore, R. B. Kearfott, and M. J. Cloud. *Introduction to Interval Analysis*. SIAM, 2009.

[30] A. Neumaier. *Interval Methods for Systems of Equations*. Cambridge Univ. Press, Cambridge, 1990.

[31] D. O'Regan, Y. Cho, and Y.Q.Chen. *Topological Degree Theory and Applications*. Chapman & Hall, 2006.

[32] E. Outerelo and J. M. Ruiz. *Mapping Degree Theory*. American Mathematical Society, 2009.

[33] P. H. Potgieter. Computable counter-examples to the Brouwer fixed-point theorem. *ArXiv e-prints*, Apr. 2008.

[34] S. Ratschan. Continuous first-order constraint satisfaction. In J. Calmet, B. Benhamou, O. Caprotti, L. Henocque, and V. Sorge, editors, *Artificial Intelligence, Automated Reasoning, and Symbolic Computation*, number 2385 in LNCS, pages 181–195. Springer, 2002.

[35] S. Ratschan. Quantified constraints under perturbations. *Journal of Symbolic Computation*, 33(4):493–505, 2002.

[36] S. Ratschan. Convergent approximate solving of first-order constraints by approximate quantifiers. *ACM Transactions on Computational Logic*, 5(2):264–281, 2004.

[37] S. Ratschan. Efficient solving of quantified inequality constraints over the real numbers. *ACM Transactions on Computational Logic*, 7(4):723–748, 2006.

[38] S. Ratschan. Safety verification of non-linear hybrid systems is quasi-semidecidable. In *TAMC 2010: 7th Annual Conference on Theory and Applications of Models of Computation*, volume 6108 of *LNCS*, pages 397–408. Springer, 2010.

[39] S. M. Rump. A note on epsilon-inflation. *Reliable Computing*, 4:371–375, 1998.

[40] D. A. Spielman and S.-H. Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *J. ACM*, 51(3):385–463, May 2004.

[41] D. A. Spielman and S.-H. Teng. Smoothed analysis: An attempt to explain the behavior of algorithms in practice. *Comm. ACM*, 52(10):76–84, 2009.

[42] A. Tarski. *A Decision Method for Elementary Algebra and Geometry*. Univ. of California Press, Berkeley, 1951. Also in [11].

[43] P. S. Wang. The undecidability of the existence of zeros of real elementary functions. *J. ACM*, 21(4):586–589, 1974.

[44] C. K. Yap. Robust geometric computation. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 41, pages 927–952. Chapmen & Hall/CRC, Boca Raton, FL, 2nd edition, 2004. Revised and expanded from 1997 version.

[45] K. Zhou. *Essentials of Robust Control*. Prentice Hall, 1997.