Subtropical Satisfiability

Pascal Fontaine[1], Mizuhito Ogawa[2], Thomas Sturm[1,3], Vu Xuan Tung[2*]

[1] University of Lorraine, CNRS, Inria, and LORIA, Nancy, France
{Pascal.Fontaine,thomas.sturm}@loria.fr
[2] Japan Advanced Institute of Science and Technology
{mizuhito,tungvx}@jaist.ac.jp
[3] MPI Informatics and Saarland University, Saarbrücken, Germany
sturm@mpi-inf.mpg.de

Abstract. Quantifier-free nonlinear arithmetic (QF_NRA) appears in many applications of satisfiability modulo theories solving (SMT). Accordingly, efficient reasoning for corresponding constraints in SMT theory solvers is highly relevant. We propose a new incomplete but efficient and terminating method to identify satisfiable instances. The method is derived from the subtropical method recently introduced in the context of symbolic computation for computing real zeros of a single very large multivariate polynomial. Our method takes as input conjunctions of strict polynomial inequalities, which represent more than 40% of the QF_NRA section of the SMT-LIB library of benchmarks. The method takes an abstraction of polynomials as exponent vectors over the natural numbers tagged with the signs of the corresponding coefficients. It then uses, in turn, SMT to solve linear problems over the reals to heuristically find suitable points that translate back to satisfying points for the original problem. Systematic experiments on the SMT-LIB demonstrate that our method is not a sufficiently strong decision procedure by itself but a valuable heuristic to use within a portfolio of techniques.

1 Introduction

Satisfiability Modulo Theories (SMT) has been blooming in recent years, and many applications rely on SMT solvers to check the satisfiability of large and numerous formulas [3, 2]. Many of those applications use arithmetic, and linear arithmetic has been one of the first theories considered in SMT.

Several SMT solvers also handle non-linear arithmetic theories. To be precise, some SMT solvers now support constraints of the form $p \bowtie 0$, where $\bowtie \in \{=, \leq, <\}$ and p is a polynomial over real or integer variables. Various techniques are used to solve these constraints over reals, e.g., cylindrical algebraic decomposition (RAHD [21, 20], Z3 4.3 [17]), virtual substitution (SMT-RAT [9], Z3 3.1), interval constraint propagation [4] (HySAT-II [10], dReal [15, 14], RSolver [22], RealPaver [16], raSAT [25]), and CORDIC (CORD [12]). Bit-blasting (MiniSmt [26]) and linearization (Barcelogic [5]) can be used for integers.

* The order of authors is strictly alphabetic.

We here present an incomplete but efficient method to detect the satisfiability of large conjunctions of constraints of the form $p > 0$ where p is a polynomial over real strictly positive variables. Although seemingly restrictive, 40% of the quantifier-free non-linear real arithmetic (QF_NRA) category of the SMT-LIB is easily reducible to this fragment. The heuristic builds on a *subtropical* technique that has been found very effective to find roots of extremely large polynomial stemming from chemistry and systems biology [24]. Intuitively, the subtropical approach somehow generalizes and adapts to any dimension the simple property that a univariate polynomial diverges positively when x increases to infinity, if the coefficient of the monomial with the highest degree is positive.

We first provide a short presentation of the original method [24] and give some new insights for its foundations in Sect. 3. A second contribution of the paper, presented in Sect. 4, is the extension of the method to multiple polynomials. We then show in Sect. 5 that satisfiability modulo linear theory is particularly adequate to check the applicability conditions of the method. Finally, we provide experimental evidence that the method is totally suited as a heuristic to use together with other complete decision procedures for non-linear arithmetic in Satisfiability Modulo Theories. It is indeed extremely fast to detect satisfiability or to fail, and finds solutions for problems for which state-of-the-art non-linear arithmetic SMT solvers timeout.

2 Basic Facts and Definitions

For $a \in \mathbb{R}$, vector $\mathbf{x} = (x_1, \ldots, x_d)$ of variables, and $\mathbf{p} = (p_1, \ldots, p_d) \in \mathbb{R}^d$, we use notations $a^{\mathbf{p}} = (a^{p_1}, \ldots, a^{p_d})$ and $\mathbf{x}^{\mathbf{p}} = (x_1^{p_1}, \ldots, x_d^{p_d})$. The frame F of a multivariate polynomial $f \in \mathbb{Z}[x_1, \ldots, x_d]$ in sparse distributive representation

$$f = \sum_{\mathbf{p} \in F} f_{\mathbf{p}} x^{\mathbf{p}}, \quad f_{\mathbf{p}} \neq 0, \quad F \subset \mathbb{N}^d,$$

is uniquely determined, and written frame(f). It can be partitioned into a positive and a negative frame, according to the sign of $f_{\mathbf{p}}$:

$$\text{frame}^+(f) = \{\, \mathbf{p} \in \text{frame}(f) \mid f_{\mathbf{p}} > 0 \,\}, \quad \text{frame}^-(f) = \{\, \mathbf{p} \in \text{frame}(f) \mid f_{\mathbf{p}} < 0 \,\}.$$

For $\mathbf{p}$ and $\mathbf{q} \in \mathbb{R}^d$, $\overline{\mathbf{pq}}$ denotes $\{\, \lambda\mathbf{p} + (1-\lambda)\mathbf{q} \in \mathbb{R}^n \mid \lambda \in [0,1] \,\}$. Recall that $S \subseteq \mathbb{R}^d$ is *convex* if $\overline{\mathbf{pq}} \subseteq S$ for all $\mathbf{p}, \mathbf{q} \in S$. Furthermore, given any $S \subseteq \mathbb{R}^d$, the *convex hull* $\text{co}(S) \subseteq \mathbb{R}^d$ is the unique inclusion-minimal convex set containing S.

The *Newton polytope* of a polynomial f is the convex hull of its frame, newton$(f) = \text{co}(\text{frame}(f))$. Fig. 1a illustrates the Newton polytope of

$$y + 2xy^3 - 3x^2y^2 - x^3 - 4x^4y^4 \in \mathbb{Z}[x, y],$$

i.e., the convex hull of its frame $\{(0,1), (1,3), (2,2), (3,0), (4,4)\} \subset \mathbb{N}^2$. As a convex hull of a finite set of points, it is bounded and then a polytope [23].

The *face* [23] of a polytope $P \subseteq \mathbb{R}^d$ with respect to a vector $\mathbf{n} \in \mathbb{R}^d$ is

$$\text{face}(\mathbf{n}, P) = \{\, \mathbf{p} \in P \mid \mathbf{n}^T\mathbf{p} \geq \mathbf{n}^T\mathbf{q} \text{ for all } \mathbf{q} \in P \,\}.$$

Faces of dimension 0 are called *vertices*. We denote by $\mathrm{vx}(P)$ the set of all vertices of P, and $\mathbf{p} \in \mathrm{vx}(P)$ if and only if there exists $\mathbf{n} \in \mathbb{R}^d$ such that $\mathbf{n}^T\mathbf{p} > \mathbf{n}^T\mathbf{q}$ for all $\mathbf{q} \in P \setminus \{\mathbf{p}\}$. In Fig.1a, $(4, 4)$ is a vertex of the Newton polytope with respect to $(1, 1)$.

It is easy to see that for finite $S \subset \mathbb{R}^d$ we have

$$\mathrm{vx}(\mathrm{co}(S)) \subseteq S \subseteq \mathrm{co}(S). \tag{1}$$

The following lemma gives a characterization of $\mathrm{vx}(\mathrm{co}(S))$:

Lemma 1. Let $S \subset \mathbb{R}^d$ be finite, and let $\mathbf{p} \in S$. The following are equivalent:

(i) $\mathbf{p}$ is a vertex of $\mathrm{co}(S)$ with respect to $\mathbf{n}$.
(ii) There exists a hyperplane $H : \mathbf{n}^T\mathbf{x} + c = 0$ that strictly separates $\mathbf{p}$ from $S \setminus \{\mathbf{p}\}$, and the normal vector $\mathbf{n}$ is directed from H towards $\mathbf{p}$.

Proof. Assume (i). Then there exists $\mathbf{n} \in \mathbb{R}^d$ such that $\mathbf{n}^T\mathbf{p} > \mathbf{n}^T\mathbf{q}$ for all $\mathbf{q} \in S \setminus \{\mathbf{p}\} \subseteq \mathrm{co}(S) \setminus \{\mathbf{p}\}$. Choose $\mathbf{q}_0 \in S \setminus \{\mathbf{p}\}$ such that $\mathbf{n}^T\mathbf{q}_0$ is maximal, and choose c such that $\mathbf{n}^T\mathbf{p} > -c > \mathbf{n}^T\mathbf{q}_0$. Then $\mathbf{n}^T\mathbf{p} + c > 0$ and $\mathbf{n}^T\mathbf{q} + c \leq \mathbf{n}^T\mathbf{q}_0 + c < 0$ for all $\mathbf{q} \in S \setminus \{\mathbf{p}\}$. Hence $H : \mathbf{n}^T\mathbf{p} + c = 0$ is the desired hyperplane.

Assume (ii). It follows that $\mathbf{n}^T\mathbf{p} + c > 0 > \mathbf{n}^T\mathbf{q} + c$ for all $\mathbf{q} \in S \setminus \{\mathbf{p}\}$. If $\mathbf{q} \in S \setminus \{\mathbf{p}\}$, then $\mathbf{n}^T\mathbf{p} > \mathbf{n}^T\mathbf{q}$. If, in contrast, $\mathbf{q} \in (\mathrm{co}(S) \setminus S) \setminus \{\mathbf{p}\} = \mathrm{co}(S) \setminus S$, then $\mathbf{q} = \sum_{\mathbf{s} \in S} t_\mathbf{s}\mathbf{s}$, where $t_\mathbf{s} \in [0, 1]$, $\sum_{\mathbf{s} \in S} t_\mathbf{s} = 1$, and at least two $t_\mathbf{s}$ are greater than 0. It follows that

$$\mathbf{n}^T\mathbf{q} = \mathbf{n}^T \sum_{\mathbf{s} \in S} t_\mathbf{s}\mathbf{s} < \mathbf{n}^T\mathbf{p} \sum_{\mathbf{s} \in S} t_\mathbf{s} = \mathbf{n}^T\mathbf{p}. \quad \square$$

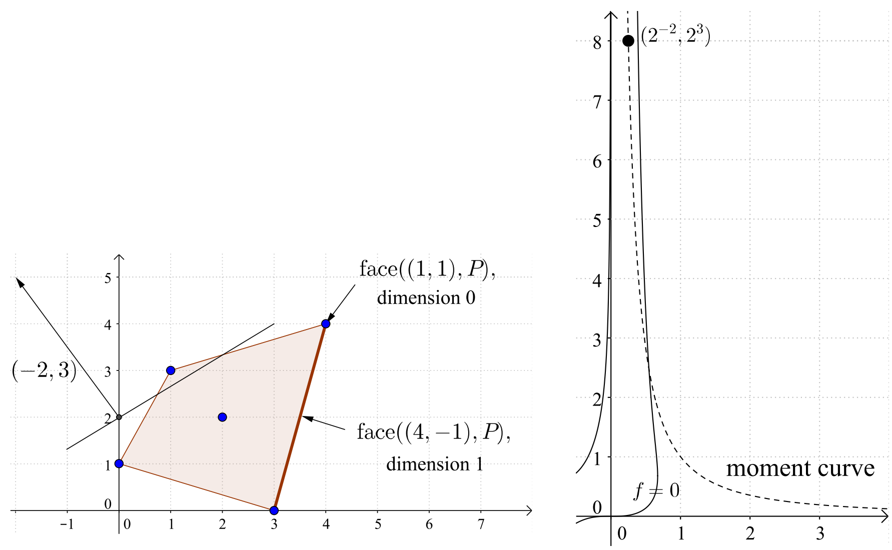
Let $S_1, \ldots, S_m \subseteq \mathbb{R}^d$, and let $\mathbf{n} \in \mathbb{R}^d$. If there exist $\mathbf{p}_1 \in S_1, \ldots, \mathbf{p}_n \in S_m$ such that each $\mathbf{p}_i$ is a vertex of $\mathrm{co}(S_i)$ with respect to $\mathbf{n}$, then $(\mathbf{p}_1, \ldots, \mathbf{p}_m)$ is the (unique) *vertex cluster* of $\{S_i\}_{i \in \{1,\ldots,m\}}$ with respect to $\mathbf{n}$.

3 Subtropical Real Root Finding Revisited

This section improves on the *original method* described in [24]. It furthermore lays some theoretical foundations to better understand the limitations of the heuristic approach. The method finds real zeros with all positive coordinates of a polynomial f in three steps:

1. Evaluate $f(1, \ldots, 1)$. If this is 0, we are done. If this is greater than 0, then consider $-f$ instead of f. We may now assume that we have found $f(1, \ldots, 1) < 0$.
2. Find $\mathbf{p}$ with all positive coordinates such that $f(\mathbf{p}) > 0$.
3. Use the intermediate value theorem and construct a root of f on the line segment $\overline{\mathbf{1}\mathbf{p}}$.

We focus here on Step 2. Our technique builds on [24, Lemma 4], which we restate here in a slightly generalized form. The original lemma required that $\mathbf{p} \in \mathrm{frame}(f) \setminus \{0\}$, but inspection of the proof show that this limitation is not necessary:



(a) The frame and the Newton polytope P of f

(b) The variety of f and the moment curve (a^{-2}, a^3)

Fig. 1: Example 3: $f = y + 2xy^3 - 3x^2y^2 - x^3 - 4x^4y^4$

Lemma 2. Let f be a polynomial, and let $\mathbf{p} \in \text{frame}(f)$ be a vertex of newton(f) with respect to $\mathbf{n} \in \mathbb{R}^d$. Then there exists $a_0 \in \mathbb{R}^+$ such that for all $a \in \mathbb{R}^+$ with $a \geq a_0$, the following holds

1. $|f_{\mathbf{p}}\ a^{\mathbf{n}^T\mathbf{p}}| > |\sum_{\mathbf{q}\in\text{frame}(f)\setminus\{\mathbf{p}\}} f_{\mathbf{q}}\ a^{\mathbf{n}^T\mathbf{q}}|$,
2. $\text{sign}(f(a^{\mathbf{n}})) = \text{sign}(f_{\mathbf{p}})$. □

In order to find a point with all positive coordinates where $f > 0$, the original method iteratively examines each $\mathbf{p} \in \text{frame}^+(f) \setminus \{\mathbf{0}\}$ to check if it is a vertex of newton(f) with respect to some $\mathbf{n} \in \mathbb{R}^d$. In the positive case, Lemma 2 guarantees for large enough $a \in \mathbb{R}^+$ that $\text{sign}(f(a^{\mathbf{n}})) = \text{sign}(f_{\mathbf{p}}) = 1$, in other words, $f(a^{\mathbf{n}}) > 0$.

Example 3. Consider $f = y+2xy^3-3x^2y^2-x^3-4x^4y^4$. Figure 1a illustrates the frame and the Newton polytope of f, of which $(1, 3)$ is a vertex with respect to $(-2, 3)$. Lemma 2 ensures that $f(a^{-2}, a^3)$ is strictly positive for a sufficiently large positive a. For example, $f(2^{-2}, 2^3) = \frac{51193}{256}$. Figure 1b shows that the moment curve (a^{-2}, a^3) with $a \geq 2$ will not leave the sign invariant region of f containing $(2^{-2}, 2^3)$.

An exponent vector $\mathbf{0} \in \text{frame}(f)$ corresponds to an absolute summand $f_{\mathbf{0}}$ in f. Its above-mentioned explicit exclusion in [24, Lemma 4] originated from the false intuition that one cannot achieve $\text{sign}(f(a^{\mathbf{n}})) = \text{sign}(f_{\mathbf{0}})$ because the monomial $f_{\mathbf{0}}$ is invariant under the choice of a. However, inclusion of $\mathbf{0}$ can yield a normal vector $\mathbf{n}$ which renders all other monomials small enough for $f_{\mathbf{0}}$ to dominate.

Given a finite set $S \subset \mathbb{R}^d$ and a point $\mathbf{p} \in S$, the original method uses linear programming to determine if $\mathbf{p}$ is a vertex of $\mathrm{co}(S)$ w.r.t. some vector $\mathbf{n} \in \mathbb{R}^d$. Indeed, from Lemma 1, the problem can be reduced to finding a hyperplane $H : \mathbf{n}^T\mathbf{x} + c = 0$ that strictly separates $\mathbf{p}$ from $S \setminus \{\mathbf{p}\}$ with the normal vector $\mathbf{n}$ pointing from H to $\mathbf{p}$. This is equivalent to solving the following linear problem with $d+1$ real variables $\mathbf{n}$ and c:

$$\varphi(\mathbf{p}, S, \mathbf{n}, c) \doteq \mathbf{n}^T\mathbf{p} + c > 0 \wedge \bigwedge_{\mathbf{q} \in S \setminus \{\mathbf{p}\}} \mathbf{n}^T\mathbf{q} + c < 0. \tag{2}$$

Notice that with the occurrence of a nonzero absolute summand the corresponding point $\mathbf{0}$ is generally a vertex of the Newton polytope with respect to $-\mathbf{1} = (-1, \dots, -1)$. This raises the question whether there are other special points that are certainly vertices of the the Newton polytope. In fact, $\mathbf{0}$ is a lexicographic minimum in $\mathrm{frame}(f)$, and it is not hard to see that minima and maxima with respect to lexicographic orderings are generally vertices of the Newton polytope.

We are now going to generalize that observation. A *monotonic total preorder* $\preceq \subseteq \mathbb{Z}^d \times \mathbb{Z}^d$ is defined as follows:

(i) $\mathbf{x} \preceq \mathbf{x}$ (reflexivity)
(ii) $\mathbf{x} \preceq \mathbf{y} \wedge \mathbf{y} \preceq \mathbf{z} \longrightarrow \mathbf{x} \preceq \mathbf{z}$ (transitivity)
(iii) $\mathbf{x} \preceq \mathbf{y} \longrightarrow \mathbf{x} + \mathbf{z} \preceq \mathbf{y} + \mathbf{z}$ (monotonicity)
(iv) $\mathbf{x} \preceq \mathbf{y} \vee \mathbf{y} \preceq \mathbf{x}$ (totality)

The difference to a total order is the missing anti-symmetry. As an example in $\mathbb{Z}^2$ consider $(x_1, x_2) \preceq (y_1, y_2)$ if and only if $x_1 + x_2 \leq y_1 + y_2$. Alternatively one could use projections or the maxima of the coordinates. Then $-2 \preceq 2$ and $2 \preceq -2$ but $-2 \neq 2$. Our definition of $\preceq$ on the extended domain $\mathbb{Z}^d$ guarantees a cancellation law $\mathbf{x} + \mathbf{z} \preceq \mathbf{y} + \mathbf{z} \longrightarrow \mathbf{x} \preceq \mathbf{y}$ also on $\mathbb{N}^d$. The following lemma follows by induction using monotonicity and cancellation:

Lemma 4. For $n \in \mathbb{N} \setminus \{0\}$ denote as usual the n-fold addition of $\mathbf{x}$ as $n \odot \mathbf{x}$. Then $\mathbf{x} \preceq \mathbf{y} \longleftrightarrow n \odot \mathbf{x} \preceq n \odot \mathbf{y}$. □

Any monotonic preorder $\preceq$ on $\mathbb{Z}^d$ can be extended to $\mathbb{Q}^d$: Using a suitable principle denominator $n \in \mathbb{N} \setminus \{0\}$ define

$$\left(\frac{x_1}{n}, \dots, \frac{x_d}{n}\right) \preceq \left(\frac{y_1}{n}, \dots, \frac{y_d}{n}\right) \quad \text{if and only if} \quad (x_1, \dots, x_d) \preceq (y_1, \dots, y_d).$$

This is well-defined.

Given $\mathbf{x} \preceq \mathbf{y}$ we have either $\mathbf{y} \not\preceq \mathbf{x}$ or $\mathbf{y} \preceq \mathbf{x}$. In the former case we say that $\mathbf{x}$ and $\mathbf{y}$ are *strictly* preordered and write $\mathbf{x} \prec \mathbf{y}$. In the latter case they are *not* strictly preordered, i.e., $\mathbf{x} \not\prec \mathbf{y}$ although we might have $\mathbf{x} \neq \mathbf{y}$. In particular, reflexivity yields $\mathbf{x} \preceq \mathbf{x}$ and hence certainly $\mathbf{x} \not\prec \mathbf{x}$.

Example 5. Lexicographic orders are monotonic total orders and thus monotonic total preorders. Hence our notion covers our discussion of the absolute summand above. Here are some further examples: For $i \in \{1, \dots, d\}$ we define $\mathbf{x} \preceq_i \mathbf{y}$ if and only if $\pi_i(\mathbf{x}) \leq \pi_i(\mathbf{y})$, where π_i denotes the i-th projection. Similarly, $\mathbf{x} \succeq_i \mathbf{y}$ if and only if $\pi_i(\mathbf{x}) \geq \pi_i(\mathbf{y})$. Next, $\mathbf{x} \preceq_\Sigma \mathbf{y}$ if and only if $\sum_i x_i \leq \sum_i y_i$. Our last example is going to be instrumental with the proof of the next theorem: Fix $\mathbf{n} \in \mathbb{R}^d$, and define for $\mathbf{p}$, $\mathbf{p}' \in \mathbb{Z}^d$ that $\mathbf{p} \preceq_{\mathbf{n}} \mathbf{p}'$ if and only if $\mathbf{n}^T\mathbf{p} \leq \mathbf{n}^T\mathbf{p}'$.

Theorem 6. *Let $f \in \mathbb{Z}[x_1, \dots, x_d]$, and let $\mathbf{p} \in \mathrm{frame}(f)$. Then the following are equivalent:*

(i) $\mathbf{p} \in \mathrm{vx}(\mathrm{newton}(f))$
(ii) There exists a monotonic total preorder $\preceq$ on $\mathbb{Z}^d$ such that

$$\mathbf{p} = \max_{\prec}(\mathrm{frame}(f)).$$

Proof. Let $\mathbf{p}$ be a vertex of $\mathrm{newton}(f)$ specifically with respect to $\mathbf{n}$. By our definition of a vertex in Sect. 2, $\mathbf{p}$ is the maximum of $\mathrm{frame}(f)$ with respect to $\prec_{\mathbf{n}}$.

Let, vice versa, $\preceq$ be a monotonic total preorder on $\mathbb{Z}^d$, and let $\mathbf{p} = \max_{\prec}(\mathrm{frame}(f))$. Shortly denote $V = \mathrm{vx}(\mathrm{newton}(f))$, and assume for a contradiction that $\mathbf{p} \notin V$. Since $\mathbf{p} \in \mathrm{frame}(f) \subseteq \mathrm{newton}(f)$, we have

$$\mathbf{p} = \sum_{\mathbf{s} \in V} t_{\mathbf{s}}\mathbf{s}, \quad \text{where} \quad t_{\mathbf{s}} \in [0, 1] \quad \text{and} \quad \sum_{\mathbf{s} \in V} t_{\mathbf{s}} = 1.$$

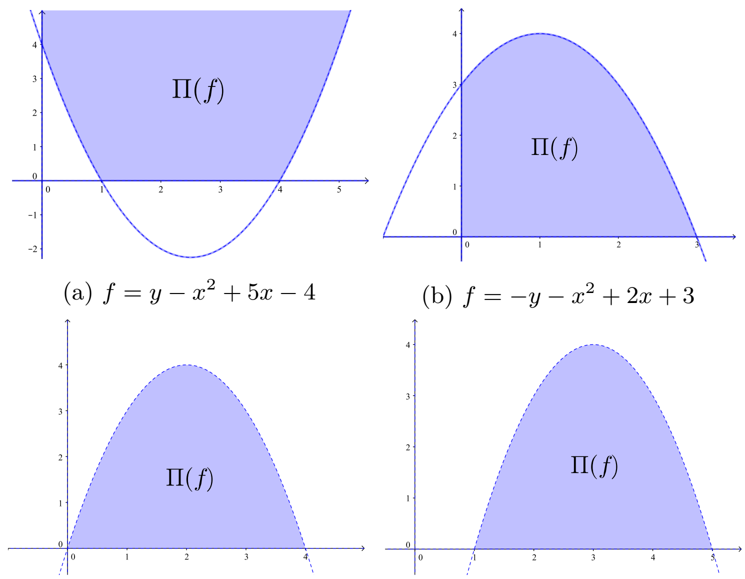
By Eq. (1) in Sect. 2 we know that $V \subseteq \mathrm{frame}(f) \subseteq \mathrm{newton}(f)$. It follows that $\mathbf{s} \prec \mathbf{p}$ for all $\mathbf{s} \in V$, and using monotony we obtain

$$\mathbf{p} \prec \sum_{\mathbf{s} \in V} t_{\mathbf{s}}\mathbf{p} = \left(\sum_{\mathbf{s} \in V} t_{\mathbf{s}}\right)\mathbf{p} = \mathbf{p}.$$

On the other hand, we know that generally $\mathbf{p} \not\prec \mathbf{p}$, a contradiction. □

In Fig.1a we have $(0, 1) = \max_{\succeq_1}(\mathrm{frame}(f))$, $(3, 0) = \max_{\succeq_2}(\mathrm{frame}(f))$, and $(4, 4) = \max_{\preceq_1}(\mathrm{frame}(f)) = \max_{\preceq_2}(\mathrm{frame}(f))$. This shows that, besides contributing to our theoretical understanding, the theorem can be used to substantiate the efficient treatment of certain special cases in combination with other methods for identifying vertices of the Newton polytope.

It is an important research goal to identify and characterize those polynomials where the subtropical heuristic succeeds in finding positive points. We are now going to give a necessary criterion. Let $f \in \mathbb{Z}[x_1, \dots, x_d]$, define $\Pi(f) = \{\, \mathbf{r} \in\,]0, \infty[^d \mid f(\mathbf{r}) > 0 \,\}$, and denote by $\overline{\Pi(f)}$ its closure with respect to the natural topology. In Lemma 2, when a tends to ∞, $a^{\mathbf{n}}$ will tend to some $\mathbf{r} \in \{0, \infty\}^d$. If $\mathbf{r} = \mathbf{0}$, then $\mathbf{0} \in \overline{\Pi(f)}$. Otherwise, $\Pi(f)$ is unbounded. Consequently, for the method to succeed, Π must have at least one of those two properties. Figure 2 illustrates four scenarios: the subtropical method succeeds



(a) $f = y - x^2 + 5x - 4$ (b) $f = -y - x^2 + 2x + 3$

(c) $f = -y - x^2 + 4x$ (d) $f = -y - x^2 + 6x - 5$

Fig. 2: Scenarios of polynomials for the subtropical method

in the first three cases while it fails to find a point in $\Pi(f)$ in the last one. The first sub-figure presents a case where $\Pi(f)$ is unbounded. The second and third sub-figures illustrate cases where the closure of $\Pi(f)$ contains $(0, 0)$. In the fourth sub-figure where neither $\Pi(f)$ is unbounded nor its closure contains $(0, 0)$, the method cannot find any positive value of the variables for f to be positive.

4 Positive Values of Several Polynomials

The subtropical method as presented in [24] finds zeros with all positive coordinates of one single polynomial, which requires to find such a point with a positive value of the polynomial. In the sequel we restrict ourselves to finding points with positive values. This will allow us generalize from one polynomial to simultaneous positive values of finitely many polynomials.

4.1 A Sufficient Condition

For one polynomial, the existence of a positive vertex of the Newton polytope guarantees the existence of positive real choices for the variables with a positive value of the polynomial. With several polynomials we introduce the more general notion of a *vertex cluster*. The existence of a positive vertex cluster will guarantee the existence of positive real choices of the variables such that all polynomials are simultaneously positive. A sequence of $(\mathbf{p}_1, \ldots, \mathbf{p}_m)$ is a *positive vertex* *cluster* of $\{f_i\}_{i\in\{1,\dots,m\}}$ with respect to $\mathbf{n} \in \mathbb{R}^d$ if and only if it is a vertex cluster of $\{\text{frame}(f_i)\}_{i\in\{1,\dots,m\}}$ with respect to $\mathbf{n}$ and for all $i \in \{1,\dots,m\}$, $\mathbf{p}_i \in \text{frame}^+(f_i)$.

The following lemma is an extension of Lemma 2 for multiple polynomials f_1, $\dots$, f_m.

Lemma 7. If there exist a vertex cluster $(\mathbf{p}_1,\dots,\mathbf{p}_m)$ of $\{\text{frame}(f_i)\}_{i\in\{1,\dots,m\}}$ with respect to $\mathbf{n} \in \mathbb{R}^n$, then there exists $a_0 \in \mathbb{R}^+$ such that for all $a \in \mathbb{R}^+$ with $a \geq a_0$ the following holds for each $i \in \{1,\dots,m\}$:

1. $|(f_i)_{\mathbf{p}_i}\ a^{\mathbf{n}^T\mathbf{p}_i}| > |\sum_{\mathbf{q}\in\text{frame}(f_i)\setminus\{\mathbf{p}_i\}}(f_i)_{\mathbf{q}}\ a^{\mathbf{n}^T\mathbf{q}}|$, and
2. $\text{sign}(f_i(a^{\mathbf{n}})) = \text{sign}((f_i)_{\mathbf{p}_i})$.

Proof. From [24, Lemma 4], for each $i \in \{1,\dots,m\}$, there exist $a_{0,i} \in \mathbb{R}^+$ such that for all $a \in \mathbb{R}^+$ with $a \geq a_{0,i}$ the following holds:

1. $|(f_i)_{\mathbf{p}_i}\ a^{\mathbf{n}^T\mathbf{p}_i}| > |\sum_{\mathbf{q}\in\text{frame}(f_i)\setminus\{\mathbf{p}_i\}}(f_i)_{\mathbf{q}}\ a^{\mathbf{n}^T\mathbf{q}}|$, and
2. $\text{sign}(f_i(a^n)) = \text{sign}((f_i)_{\mathbf{p}_i})$.

It then suffices to take $a_0 = \max\{a_{0,i} \mid 1 \leq i \leq m\}$. ◻

Similarly to the case of one polynomial, the following Proposition provides a sufficient condition for the existence of a positive value for multiple polynomials.

Proposition 8. *If there exists a positive vertex cluster $(\mathbf{p}_1,\dots,\mathbf{p}_m)$ of $\{f_i\}_{i\in\{1,\dots,m\}}$ with respect to a vector $\mathbf{n} \in \mathbb{R}^d$, then there exists $a_0 \in \mathbb{R}^+$ such that for all $a \in \mathbb{R}^+$ with $a \geq a_0$ the following holds:*

$$\bigwedge_{i=1}^{m} f_i(a^{\mathbf{n}}) > 0$$

Proof. For each $i = \{1,\dots,m\}$ since $\mathbf{p}_i \in \text{frame}^+(f_i)$, Lemma 7 implies that $f_i(a^{\mathbf{n}}) > 0$. ◻

Example 9. Consider three polynomials $f_1 = 2 - xy^2z + x^2yz^3$, $f_2 = 3 - xy^2z^4 - x^2z - x^4y^3z^3$, and $f_3 = 4 - z - y - x + 4$. The exponent vector $\mathbf{0}$ is a vertex of newton(f_1), newton(f_2), and newton(f_3) with respect to $(-1,-1,-1)$. There thus exists $a_0 \in \mathbb{R}$, ($a_0 = 2$ is suitable), such that for all $a \in \mathbb{R}$ with $a \geq a_0$, $f_1(a^{-1},a^{-1},a^{-1}) > 0 \wedge f_2(a^{-1},a^{-1},a^{-1}) > 0 \wedge f_3(a^{-1},a^{-1},a^{-1}) > 0$. ◻

4.2 Existence of Positive Vertex Clusters

Given polynomials f_1, $\dots$, f_m, Proposition 8 provides a sufficient condition, i.e. the existence of a positive vertex cluster of $\{f_i\}_{i\in\{1,\dots,m\}}$, for the satisfiability of $\bigwedge_{i=1}^m f_i > 0$. A straightforward method to decide the existence of such a cluster is to verify whether each $(\mathbf{p}_1,\dots,\mathbf{p}_m) \in \text{frame}^+(f_1) \times \cdots \times \text{frame}^+(f_m)$ is a positive vertex cluster by checking the satisfiability of the formula

$$\bigwedge_{i\in\{1,\dots,m\}} \varphi(\mathbf{p}_i, \text{frame}(f_i), \mathbf{n}, c_i),$$

where φ is defined as in Eq. (2) on p.5. This is a linear problem with $d+m$ variables $\mathbf{n}$, $c_1, \ldots, c_m$. Since $\mathrm{frame}(f_1), \ldots, \mathrm{frame}(f_m)$ are finite, checking all m-tuples $(\mathbf{p}_1, \ldots, \mathbf{p}_m)$ will terminate, provided we rely on a complete algorithm for linear programming, such as the Simplex algorithm [7], the ellipsoid method [19], or the interior point method [18]. This thus provides a decision procedure for the existence of a positive vertex cluster of $\{f_i\}_{i\in\{1,\ldots,m\}}$. However, this requires checking all candidates in $\mathrm{frame}^+(f_1) \times \cdots \times \mathrm{frame}^+(f_m)$.

We propose to use instead state-of-the-art SMT solving techniques over linear real arithmetic to examine whether or not $\{f_i\}_{i\in\{1,\ldots,m\}}$ has a positive vertex cluster with respect to some $\mathbf{n} \in \mathbb{R}^d$. In the positive case, a solution for $\bigwedge_{i=1}^m f_i > 0$ can be constructed as $a^{\mathbf{n}}$ with a sufficiently large $a \in \mathbb{R}^+$.

To start with, we provide a characterization for the positive frame of a single polynomial to contain a vertex of the Newton polytope.

Lemma 10. Given a polynomial f, there exists a vertex $\mathbf{p} \in \mathrm{frame}^+(f)$ of $\mathrm{newton}(f) = \mathrm{co}(\mathrm{frame}(f))$ with respect to $\mathbf{n} \in \mathbb{R}^d$ if and only if there exists a vertex $\mathbf{p'} \in \mathrm{frame}^+(f)$ of $\mathrm{co}(\mathrm{frame}^-(f) \cup \{\mathbf{p'}\})$ with respect to $\mathbf{n'} \in \mathbb{R}^d$.

Proof. $\Rightarrow$ Take $\mathbf{p'} = \mathbf{p}$ and $\mathbf{n'} = \mathbf{n}$. Since $\mathbf{p}$ is a vertex of $\mathrm{newton}(f)$ with respect to $\mathbf{n}$, $\mathbf{n}^T\mathbf{p} > \mathbf{n}^T\mathbf{p}_1$ for all $\mathbf{p}_1 \in \mathrm{frame}(f) \setminus \{\mathbf{p}\}$. This implies that $\mathbf{n}^T\mathbf{p} > \mathbf{n}^T\mathbf{p}_1$ for all $\mathbf{p}_1 \in \mathrm{frame}^-(f) \setminus \{\mathbf{p}\} = \big(\mathrm{frame}^-(f) \cup \{\mathbf{p}\}\big) \setminus \{\mathbf{p}\}$. In other words, $\mathbf{p}$ is a vertex of $\mathrm{co}(\mathrm{frame}^-(f) \cup \{\mathbf{p}\})$ with respect to $\mathbf{n}$.

$\Leftarrow$ Suppose $V = \mathrm{vx}(\mathrm{newton}(f)) \subseteq \mathrm{frame}^-(f)$. Then, $\mathbf{p'} = \sum_{\mathbf{s}\in V} t_{\mathbf{s}}\mathbf{s}$ where $t_{\mathbf{s}} \in [0,1]$, $\sum_{\mathbf{s}\in V} t_{\mathbf{s}} = 1$. It follows that

$$\mathbf{n'}^T\mathbf{p'} = \sum_{\mathbf{s}\in V} t_{\mathbf{s}}\mathbf{n'}^T\mathbf{s} < \sum_{\mathbf{s}\in V} t_{\mathbf{s}}\mathbf{n'}^T\mathbf{p'} = \mathbf{n'}^T\mathbf{p'}\sum_{\mathbf{s}\in V} t_{\mathbf{s}} = \mathbf{n'}^T\mathbf{p'}$$

which is a contradiction. As a result, there must be some $\mathbf{p} \in \mathrm{frame}^+(f)$ which is a vertex of $\mathrm{newton}(f)$ with respect to some $\mathbf{n} \in \mathbb{R}^d$. ☐

Then some $\mathbf{p} \in \mathrm{frame}^+(f)$ is a vertex of the Newton polytope of a polynomial f if and only if the following formula is satisfiable:

$$\begin{aligned}
\psi(f, \mathbf{n'}, c) &\doteq \bigvee_{\mathbf{p}\in\mathrm{frame}^+(f)} \varphi\left(\mathbf{p}, \mathrm{frame}^-(f) \cup \{\mathbf{p}\}, \mathbf{n'}, c\right) \\
&\equiv \bigvee_{\mathbf{p}\in\mathrm{frame}^+(f)} \left(\mathbf{n'}^T\mathbf{p} + c > 0 \wedge \bigwedge_{\mathbf{q}\in\mathrm{frame}^-(f)} \mathbf{n'}^T\mathbf{q} + c < 0\right) \\
&\equiv \left(\bigvee_{\mathbf{p}\in\mathrm{frame}^+(f)} \mathbf{n'}^T\mathbf{p} + c > 0\right) \wedge \bigwedge_{\mathbf{p}\in\mathrm{frame}^-(f)} \mathbf{n'}^T\mathbf{p} + c < 0
\end{aligned}$$

For the case of several polynomials, the following theorem is a direct consequence of Lemma 10.

Theorem 11. *Polynomials $\{f_i\}_{i\in\{1,\ldots,m\}}$ have a positive vertex cluster with respect to $\mathbf{n} \in \mathbb{R}^d$ if and only if $\bigwedge_{i=1}^m \psi(f_i, \mathbf{n}, c_i)$ is satisfiable.* ☐

The formula $\bigwedge_{i=1}^{m} \psi(f_i, \mathbf{n}, c_i)$ can be checked for satisfiability using combinations of linear programming techniques and DPLL(T) procedures [8, 13], i.e., satisfiability modulo linear arithmetic on reals. If it is satisfiable, then $\{f_i\}_{i\in\{1,\ldots,m\}}$ has a positive vertex cluster and we can construct a solution for $\bigwedge_{i=1}^{m} f_i > 0$ as discussed earlier.

Example 12. Consider $f_1 = -12 + 2x^{12}y^{25}z^{49} - 31x^{13}y^{22}z^{110} - 11x^{1000}y^{500}z^{89}$ and $f_2 = -23 + 5xy^{22}z^{110} - 21x^{15}y^{20}z^{1000} + 2x^{100}y^{2}z^{49}$. We have

$$\begin{aligned}\psi(f_1, \mathbf{n}, c_1) &= 12n_1 + 25n_2 + 49n_3 + c_1 > 0 \wedge 13n_1 + 22n_2 + 110n_3 + c_1 < 0 \\ &\quad \wedge 1000n_1 + 500n_2 + 89n_3 + c_1 < 0 \wedge c_1 < 0 \\ \psi(f_2, \mathbf{n}, c_2) &= (n_1 + 22n_2 + 110n_3 + c_2 > 0 \vee 100n_1 + 2n_2 + 49n_3 + c_2 > 0) \\ &\quad \wedge 15n_1 + 20n_2 + 1000n_3 + c_2 < 0 \wedge c_2 < 0\end{aligned}$$

with $\mathbf{n} = (n_1, n_2, n_3)$. The conjunction of the two formulas above is satisfiable. The SMT solver CVC4 returns $\mathbf{n} = (-\frac{238834}{120461}, \frac{2672460}{1325071}, -\frac{368561}{1325071})$ and $c_1 = c_2 = -1$ as a model. Theorem 11 and Proposition 8 guarantee that there exists a large enough $a \in \mathbb{R}^+$ such that $f_1(a^{\mathbf{n}}) > 0 \wedge f_2(a^{\mathbf{n}}) > 0$. Indeed, $a = 2$ already yields $f_1(a^{\mathbf{n}}) \approx 16371.99$ and $f_2(a^{\mathbf{n}}) \approx 17707.27$. □

5 More General Solutions

So far all variables were assumed to be strictly positive, i.e., only the solutions $\mathbf{x} \in]0, \infty[^d$ were considered. This section proposes a method for searching over $\mathbb{R}^d$ by encoding signs conditions along with the condition in Theorem 11 as a quantifier-free formula over linear real arithmetic.

Let $V = \{x_1, \ldots, x_d\}$ be the set of variables. We define a *sign variant* of V as a function $\tau : V \mapsto V \cup \{-x \mid x \in V\}$ such that for each $x \in V$, $\tau(x) \in \{x, -x\}$. We write $\tau(f)$ to denote the substitution $f(\tau(x_1), \ldots, \tau(x_d))$ of τ into a polynomial f. Furthermore, $\tau(a)$ denotes $\left(\frac{\tau(x_1)}{x_1}a, \ldots, \frac{\tau(x_d)}{x_d}a\right)$ for $a \in \mathbb{R}$. A sequence $(\mathbf{p}_1, \ldots, \mathbf{p}_m)$ is a *variant positive vertex cluster* of $\{f_i\}_{i\in\{1,\ldots,m\}}$ with respect to a vector $\mathbf{n} \in \mathbb{R}^d$ and a sign variant τ if $(\mathbf{p}_1, \ldots, \mathbf{p}_m)$ is a positive vertex cluster of $\{\tau(f_i)\}_{i\in\{1,\ldots,m\}}$. Note that the substitution of τ into a polynomial f does not change the exponent vectors in f in terms of their exponents values, but only possibly changes signs of monomials. Given $\mathbf{p} = (p_1, \ldots, p_d) \in \mathbb{N}^d$ and a sign variant τ, we define a formula $\vartheta(\mathbf{p}, \tau)$ such that if it is TRUE, then the sign of the monomial associated with $\mathbf{p}$ is changed after applying the substitution defined by τ:

$$\vartheta(\mathbf{p}, \tau) \doteq \bigoplus_{i=1}^{d} \bigl(\tau(x_i) = -x_i \wedge (p_i \bmod 2 = 1)\bigr).$$

Note that the xor expression becomes TRUE if and only if an odd number of its operands are TRUE. Furthermore, a variable can change the sign of a monomial only when its exponent in that monomial is odd. As a result, if $\vartheta(\mathbf{p}, \tau)$ is TRUE, then applying the substitution defined by τ will change the sign of the monomial associated with $\mathbf{p}$. In conclusion, some $\mathbf{p} \in \text{frame}(f)$ is in the positive frame of $\tau(f)$ if and only if either

- $\mathbf{p} \in \text{frame}^+(f)$ and $\vartheta(\mathbf{p}, \tau) = \text{FALSE}$, or
- $\mathbf{p} \in \text{frame}^-(f)$ and $\vartheta(\mathbf{p}, \tau) = \text{TRUE}$.

In other words, $\mathbf{p}$ is in the positive frame of $\tau(f)$ if and only if the formula $\Theta(\mathbf{p}, f, \tau) \doteq \big(f_{\mathbf{p}} > 0 \wedge \neg\vartheta(\mathbf{p}, \tau)\big) \vee \big(f_{\mathbf{p}} < 0 \wedge \vartheta(\mathbf{p}, \tau)\big)$ holds. Then, the positive and negative frames of $\tau(f)$ parameterized by τ are defined as

$$\begin{aligned} \text{frame}^+(\tau(f)) &= \{\mathbf{p} \in \text{frame}(f) \mid \Theta(\mathbf{p}, f, \tau)\}, \\ \text{frame}^-(\tau(f)) &= \{\mathbf{p} \in \text{frame}(f) \mid \neg\Theta(\mathbf{p}, f, \tau)\}, \end{aligned}$$

respectively. The next lemma provides a sufficient condition for the existence of a solution in $\mathbb{R}^d$ of $\bigwedge_{i=1}^m f_i > 0$.

Lemma 13. If there exists a variant positive vertex cluster of $\{f_i\}_{i\in\{1,\ldots,m\}}$ with respect to $\mathbf{n} \in \mathbb{R}^d$ and a sign variant τ, then there exists $a_0 \in \mathbb{R}^+$ such that for all $a \in \mathbb{R}^+$ with $a \geq a_0$ the following holds:

$$\bigwedge_{i=1}^{m} f_i\big(\tau(a)^{\mathbf{n}}\big) > 0.$$

Proof. Since $\{\tau(f_i)\}_{i\in\{1,\ldots,m\}}$ has a positive vertex cluster with respect to $\mathbf{n}$, Proposition 8 guarantees that there exists $a_0 \in \mathbb{R}$ such that for all $a \in \mathbb{R}$ with $a \geq a_0$, we have $\bigwedge_{i=1}^m \tau(f_i)(a^{\mathbf{n}}) > 0$, or $\bigwedge_{i=1}^m f_i\big(\tau(a)^{\mathbf{n}}\big) > 0$. □

A variant positive vertex cluster exists if and only if there exist $\mathbf{n} \in \mathbb{R}^d$, $c_1, \ldots, c_m \in \mathbb{R}$, and a sign variant τ such that the following formula becomes TRUE:

$$\Psi(f_1, \ldots, f_m, \mathbf{n}, c_1, \ldots, c_m, \tau) \doteq \bigwedge_{i=1}^{m} \psi\big(\tau(f_i), \mathbf{n}, c_i\big),$$

where for $i \in \{1, \ldots, m\}$:

$$\begin{aligned} \psi\big(\tau(f_i), \mathbf{n}, c_i\big) &= \bigvee_{\mathbf{p}\in\text{frame}^+(\tau(f_i))} \Big(\mathbf{n}^T\mathbf{p} + c_i > 0\Big) \wedge \bigwedge_{\mathbf{p}\in\text{frame}^-(\tau(f_i))} \Big(\mathbf{n}^T\mathbf{p} + c_i < 0\Big) \\ &= \bigvee_{\mathbf{p}\in\text{frame}(f_i)} \Big(\Theta(\mathbf{p}, f_i, \tau) \wedge \mathbf{n}^T\mathbf{p} + c_i > 0\Big) \wedge \\ &\qquad \bigwedge_{\mathbf{p}\in\text{frame}(f_i)} \Big(\Theta(\mathbf{p}, f_i, \tau) \vee \mathbf{n}^T\mathbf{p} + c_i < 0\Big). \end{aligned}$$

The sign variant τ can be encoded as d Boolean variables $b_1, \ldots, b_d$ such that b_i is TRUE if and only if $\tau(x_i) = -x_i$ for all $i \in \{1, \ldots, d\}$. Then, the formula $\Psi(f_1, \ldots, f_m, \mathbf{n}, c_1, \ldots, c_m, \tau)$ can be checked for satisfiability using an SMT solver for quantifier-free logic with linear real arithmetic.

6 Application to SMT Benchmarks

A library *STROPSAT* implementing Subtropical Satisfiability, is available on our web page.[4] It is integrated into veriT [6] as an incomplete theory solver for non-linear arithmetic benchmarks. We experimented on the QF_NRA category of the SMT-LIB on all benchmarks consisting of only inequalities, that is 4917 formulas out of 11601 in the whole category. The experiments thus focus on those 4917 benchmarks, comprising 3265 SAT-annotated ones, 106 UNKNOWNs, and 1546 UNSAT benchmarks. We used the CVC4 SMT solver to handle the generated linear real arithmetic formulas $\Psi(f_1, \ldots, f_m, \mathbf{n}, c_1, \ldots, c_m, \tau)$, and we ran veriT (with STROPSAT as the theory solver) against the clear winner of the SMT-COMP 2016 on the QF_NRA category, i.e., Z3 (implementing nlsat [17]), on a CX250 Cluster with Intel Xeon E5-2680v2 2.80GHz CPUs. Each pair of benchmark and solver was run on one CPU with a timeout of 2500 seconds and 20 GB memory.

Since our method focuses on showing satisfiability, only brief statistics on UNSAT benchmarks are provided. Among the 1546 UNSAT benchmarks, 200 benchmarks are found unsatisfiable already by the linear arithmetic theory reasoning in veriT. For each of the remaining one, the method quickly returns UNKNOWN within 0.002–0.096 seconds, with a total cumulative time of 18.45 seconds (0.014 second on average). This clearly shows that the method can be applied with a very small overhead, upfront of another, complete or less incomplete procedure to check for unsatisfiability.

Table 1 provides the experimental results on benchmarks with SAT or UNKNOWN status, and the cumulative times. The meti-tarski family consists of small benchmarks (most of them contain 3 to 4 variables and 1 to 23 polynomials with degree between 1 and 4) which are proof obligations extracted from the Meti-Tarski project [1], where the polynomials represent approximations of elementary real functions; all of them have defined statuses. The zankl family consists of large benchmarks (large number of variables and polynomials, but small degree) stemming from termination proofs for term-rewriting systems [11].

Table 1: Comparison between STROPSAT and Z3 (times in seconds)

Family	STROPSAT				Z3			
	SAT	Time	UNKOWN	Time	SAT	Time	UNSAT	Time
meti-tarski (SAT - 3220)	2359	32.37	861	10.22	**3220**	88.55	0	0
zankl (SAT - 45)	29	3.77	16	0.59	**42**	2974.35	0	0
zankl (UNKNOWN - 106)	**15**	2859.44	76	6291.33	14	1713.16	23	1.06

Although Z3 clearly outperforms STROPSAT in the number of solved benchmarks, the results also clearly show that the technique is a useful complementing heuristic with little drawback, to be used either upfront or in portfolio with

[4] http://www.jaist.ac.jp/~s1310007/STROPSAT/

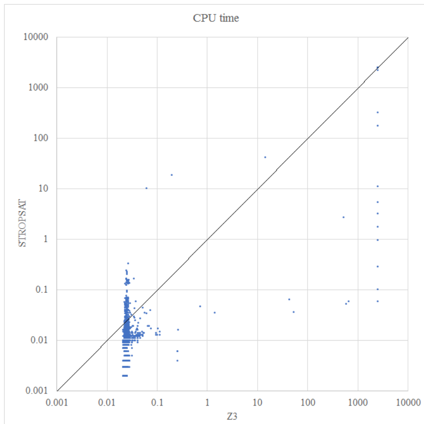


Fig. 3: STROPSAT returns SAT or timeout (2418 benchmarks)

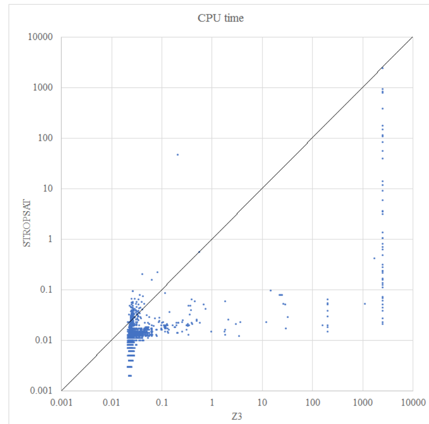


Fig. 4: STROPSAT returns UNKNOWN (2299 benchmarks)

other approaches. As already said, it returns UNKNOWN quickly on UNSAT benchmarks. In particular, on all benchmarks solved by Z3 only, STROPSAT returns UNKNOWN quickly (see Fig. 4).

When both solvers can solve a same benchmark, the running time of STROPSAT is comparable with Z3 (Fig. 3). There are 11 large benchmarks (9 of them have the UNKNOWN status) that are solved by STROPSAT but timed out by Z3. STROPSAT timed out for only 15 problems, on which Z3 also timed out. STROPSAT provided a model for 15 UNKNOWN benchmarks, whereas Z3 times out on 9 of them.

Since the exponents of the polynomials are used as coefficients in the linear formulas, high degrees should not hurt our method. The SMT-LIB does not currently contain inequality benchmarks with high degrees so that the experimental results cannot demonstrate this claim. One might mention that on the formulas in Example 12, Z3 with 20 GB of memory raised after 30 seconds an `(error "out of memory")` for the constraint $f_1 > 0 \wedge f_2 > 0$, while STROPSAT returned SAT within a second.

7 Conclusion

We presented some extensions useful to handle SMT problems of a heuristic method to find positive points of a polynomial. In practice, the method is fast, either to succeed or to fail, and it succeeds where state-of-the-art solvers do not. Because of this, it is a valuable heuristic to try either before or in parallel with other more complete methods to deal with non-linear constraints.

To improve the completeness of the method, it could be helpful to not only consider vertices of the Newton polytope, but also faces. Then, the value of the coefficients and not only their sign would matter. Consider $\{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\} =$

face($\mathbf{n}$, newton(f)), then we have $\mathbf{n}^T\mathbf{p}_1 = \mathbf{n}^T\mathbf{p}_2 = \mathbf{n}^T\mathbf{p}_3$. It is easy to see that $f_{\mathbf{p}_1}\mathbf{x}^{\mathbf{p}_1} + f_{\mathbf{p}_2}\mathbf{x}^{\mathbf{p}_2} + f_{\mathbf{p}_3}\mathbf{x}^{\mathbf{p}_3}$ will dominate the other monomials in the direction of $\mathbf{n}$. In other words, there exists $a_0 \in \mathbb{R}$ such that for all $a \in \mathbb{R}$ with $a \geq a_0$, $\mathrm{sign}(f(a^{\mathbf{n}})) = \mathrm{sign}(f_{\mathbf{p}_1} + f_{\mathbf{p}_2} + f_{\mathbf{p}_3})$. We leave for future work the encoding of the condition for the existence of such a face into linear formulas.

In the last paragraph of Section 3, we showed that, for the subtropical method to succeed, the set of values for which the considered polynomial is positive should either be unbounded, or should contain points arbitrarily near $\mathbf{0}$. We believe there is a stronger, sufficient condition, that would bring another insight to the subtropical method.

Finally, on a more practical side, we would like to investigate the use of the techniques presented here for the testing phase of the raSAT loop [25], an extension of the interval constraint propagation with testing and the intermediate value theorem. We believe that this could lead to significant improvements in the solver, since testing is now completely random.

Acknowledgment This research has been partially supported by the Agence nationale de la recherche / Deutsche Forschungsgemeinschaft project SMArT (ANR-13-IS02-0001, STU 483/2-1) and by the European Union project SC^2 (grant agreement No. 712689). The work has also received funding from the European Research Council under the European Union's Horizon 2020 research and innovation program (grant agreement No. 713999, Matryoshka).

References

1. Akbarpour, B., Paulson, L.C.: Metitarski: An automatic theorem prover for real-valued special functions. Journal of Automated Reasoning **44**(3) (2010) 175–205
2. Barrett, C., Kroening, D., Melham, T.: Problem solving for the 21st century: Efficient solvers for satisfiability modulo theories. Technical Report 3, London Mathematical Society and Smith Institute for Industrial Mathematics and System Engineering (2014) Knowledge Transfer Report.
3. Barrett, C., Sebastiani, R., Seshia, S.A., Tinelli, C.: Satisfiability modulo theories. In: Handbook of Satisfiability. Volume 185 of Frontiers in Artificial Intelligence and Applications. IOS Press (2009) 825–885
4. Benhamou, F., Granvilliers, L.: Continuous and interval constraints. In: Handbook of Constraint Programming. Elsevier, New York (2006) 571–604
5. Bofill, M., Nieuwenhuis, R., Oliveras, A., Rodríguez-Carbonell, E., Rubio, A.: The Barcelogic SMT solver. In: Computer Aided Verification. Springer, Berlin (2008) 294–298
6. Bouton, T., Caminha B. De Oliveira, D., Déharbe, D., Fontaine, P.: veriT: An open, trustable and efficient SMT-Solver. In: Proceedings of the 22nd International Conference on Automated Deduction. CADE-22, Berlin, Springer (2009) 151–156
7. Dantzig, G.B.: Linear programming and extensions. Prentice University Press, Princeton, NJ (1963)
8. Dutertre, B., de Moura, L.: A fast linear-arithmetic solver for dpll(t). In: Computer Aided Verification. Springer, Berlin (2006) 81–94
9. Florian, C., Ulrich, L., Sebastian, J., Erika, Á.: SMT-RAT: An SMT-Compliant nonlinear real arithmetic toolbox. In: Theory and Applications of Satisfiability Testing – SAT 2012. Springer, Berlin (2012) 442–448

10. Fränzle, M., Herde, C., Teige, T., Ratschan, S., Schubert, T.: Efficient solving of large non-linear arithmetic constraint systems with complex boolean structure. Journal on Satisfiability, Boolean Modeling and Computation **1** (2007) 209–236
11. Fuhs, C., Giesl, J., Middeldorp, A., Schneider-Kamp, P., Thiemann, R., Zankl, H.: SAT solving for termination analysis with polynomial interpretations. In: Theory and Applications of Satisfiability Testing – SAT 2007. Springer, Berlin (2007) 340–354
12. Ganai, M., Ivancic, F.: Efficient decision procedure for non-linear arithmetic constraints using cordic. In: Formal Methods in Computer-Aided Design, 2009. FMCAD 2009. (2009) 61–68
13. Ganzinger, H., Hagen, G., Nieuwenhuis, R., Oliveras, A., Tinelli, C.: DPLL(T): Fast decision procedures. In: Computer Aided Verification. Springer, Berlin (2004) 175–188
14. Gao, S., Kong, S., Clarke, E.M.: Satisfiability modulo ODEs. In: Formal Methods in Computer-Aided Design (FMCAD), 2013. (2013) 105–112
15. Gao, S., Kong, S., Clarke, E.: dReal: An SMT solver for nonlinear theories over the reals. In: Automated Deduction – CADE-24. Springer, Berlin (2013) 208–214
16. Granvilliers, L., Benhamou, F.: RealPaver: An interval solver using constraint satisfaction techniques. ACM Transactions on Mathematical Software **32** (2006) 138–156
17. Jovanović, D., de Moura, L.: Solving non-linear arithmetic. In: Automated Reasoning. Springer, Berlin (2012) 339–354
18. Karmarkar, N.: A new polynomial-time algorithm for linear programming. Combinatorica **4**(4) (1984) 373–395
19. Khachiyan, L.: Polynomial algorithms in linear programming. USSR Computational Mathematics and Mathematical Physics **20**(1) (1980) 53 – 72
20. Passmore, G.O.: Combined decision procedures for nonlinear arithmetics, real and complex. Dissertation, School of Informatics, University of Edinburgh (2011)
21. Passmore, G.O., Jackson, P.B.: Combined decision techniques for the existential theory of the reals. In: Intelligent Computer Mathematics, Berlin, Springer (2009) 122–137
22. Ratschan, S.: Efficient solving of quantified inequality constraints over the real numbers. ACM Transactions on Computational Logic **7** (2006) 723–748
23. Schrijver, A.: Theory of Linear and Integer Programming. John Wiley & Sons, Inc., New York, NY, USA (1986)
24. Sturm, T.: Subtropical real root finding. In: Proceedings of the ISSAC 2015. ACM (2015) 347–354
25. Tung, V.X., Van Khanh, T., Ogawa, M.: raSAT: An SMT solver for polynomial constraints. In: Automated Reasoning. Springer, Cham (2016) 228–237
26. Zankl, H., Middeldorp, A.: Satisfiability of non-linear (ir)rational arithmetic. In: Logic for Programming, Artificial Intelligence, and Reasoning. Springer, Berlin (2010) 481–500