Faster algorithms for computing real isolated points of an algebraic hypersurface

Huu Phuoc Le

Faster algorithms for computing real isolated points of an algebraic hypersurface

Huu Phuoc Le[1]

Sorbonne Université, CNRS, LIP6, *F-75005, Paris, France*

Abstract

Let $\mathbb{R}$ be the field of real numbers. We consider the problem of computing the real isolated points of a real algebraic set in $\mathbb{R}^n$ given as the vanishing set of a polynomial system. This problem plays an important role for studying rigidity properties of mechanism in material designs. In this paper, we design two algorithms of whose complexities are respectively $O^{\sim}\left(D^{8n}\right)$ and $O^{\sim}\left(D^{6n}\right)$ for computing the real isolated points of real algebraic hypersurfaces in $\mathbb{R}^n$ of degree D. We also propose several heuristic optimizations to avoid the most costly computation in our algorithms in most of the cases, which makes our complexity reduced to $O^{\sim}\left(D^{3n}\right)$ for those cases. These algorithms lead to an implementation which is able to solve instances which were out of reach.

Keywords: Real algebraic geometry; Polynomial system solving; Computational geometry

1. Introduction

Given $f \in \mathbb{Q}[x_1, \ldots, x_n]$, we denote by $\mathcal{V} \subset \mathbb{C}^n$ the algebraic hypersurface defined by $f = 0$ and by $\mathcal{H}$ the real trace of $\mathcal{V}$, i.e., $\mathcal{H} = \mathcal{V} \cap \mathbb{R}^n$. We aim at computing the *isolated points* of $\mathcal{H}$, i.e. the set of points $\boldsymbol{x} \in \mathcal{H}$ such that for some positive r, $B(\boldsymbol{x}, r) \cap \mathcal{H} = \{\boldsymbol{x}\}$ where $B(\boldsymbol{x}, r) \subset \mathbb{R}^n$ the open ball centered at $\boldsymbol{x}$ of radius r. We denote this set of real isolated points by $\mathscr{I}(\mathcal{H})$.

This problem is a particular instance of a more general problem of computing the isolated points of a *semi-algebraic* set. Such problems arise naturally and frequently in the design of rigid mechanism in material design (see [18]). Those are modeled canonically with semi-algebraic constraints, and isolated points to the semi-algebraic set under consideration are related to rigidity properties of the mechanism. A particular example is the study of *auxetic* materials, i.e., materials that shrink in all directions under compression. These materials appear in nature (first discovered in [16]) e.g., in foams, bones or propylene; see e.g. [28], and have various potential applications. They are an active field of research, not only on the practical side, e.g., [15, 13], but also with respect to mathematical foundations; see e.g. [6, 7]. On the constructive side, these materials are closely related to *tensegrity frameworks*, e.g., [19, 10], which can possess various sorts of rigidity properties.

Email address: huu-phuoc.le@lip6.fr (Huu Phuoc Le)

[1]Huu Phuoc Le is supported by the ANR grants ANR-18-CE33-0011 Sesame, the ANR-19-CE40-0018 De Rerum Natura, the joint ANR-FWF ANR-19-CE48-0015 ECARP project, the PGMO grant CAMiSAdo, the European Union's Horizon 2020 research and innovative training network program under the Marie Skłodowska-Curie grant agreement N° 813211 (POEMA) and the Grant FA8665-20-1-7029 of the EOARD-AFOSR.

Hence, we aim to provide a practical algorithm for computing these real isolated points in the particular case of real traces of complex hypersurfaces first. This simplification allows us to improve the state-of-the-art complexity for this problem and to establish a new algorithmic framework for such computations which is more efficient in practice.

Prior works.. Let $\mathcal{H}$ be a hypersurface defined by $f = 0$ with $f \in \mathbb{Q}[x_1, \ldots, x_n]$ of degree D.

A first approach would be to compute a cylindrical algebraic decomposition adapted to $\mathcal{H}$ [9]. It partitions $\mathcal{H}$ into connected *cells*, i.e. subsets which are homeomorphic to $]0, 1[^i$ for some $1 \leq i \leq n$. Next, one needs to identify cells which correspond to isolated points using adjacency information (see e.g. [1]). Such a procedure is at least doubly exponential in n and polynomial in D.

A better alternative is to encode real isolated points with quantified formula over the reals. Using e.g. [2, Algorithm 14.21], one can compute isolated points of $\mathcal{H}$ in time $D^{O(n^2)}$. Note also that [27] allows to compute isolated points in time $D^{O(n^3)}$.

A third alternative is to use [2, Algorithm 12.16] to compute sample points in each connected component of $\mathcal{H}$ and then decide whether spheres, centered at these points, of infinitesimal radius, meet $\mathcal{H}$. Note that these points are encoded with parametrizations of degree $D^{O(n)}$ (their coordinates are evaluations of polynomials at the roots of a univariate polynomial with infinitesimal coefficients). Applying [2, Alg. 12.16] on this last real root decision problem would lead to a complexity $D^{O(n^2)}$ since the input polynomials would have degree $D^{O(n)}$. Another approach would be to run [2, Alg. 12.16] modulo the algebraic extension used to define the sample points. That would lead to a complexity $D^{O(n)}$ but this research direction requires modifications of [2, Alg. 12.16] since it assumes the input coefficients to lie in an *integral domain*, which is not satisfied in our case. Besides, we report on practical experiments showing that using [2, Alg. 12.16] to compute *only* sample points in $\mathcal{H}$ does not allow us to solve instances of moderate size.

Since the topological nature of this problem is related to connectivity. Computing isolated points of $\mathcal{H}$ is equivalent to computing those connected components of $\mathcal{H}$ which are reduced to a single point (see [17, Lemma 1]). Hence, one can consider computing *roadmaps*: these are algebraic curves contained in $\mathcal{H}$ which have a non-empty and connected intersection with all connected components of the real set under study. The authors of [25] designed a probabilistic algorithm that computes roadmaps for smooth bounded real algebraic sets. This algorithm runs in time $O((nD)^{12n \log_2 n})$, which makes it the state-of-the-art complexity for roadmap computation. These results makes plausible to obtain in [17] an algorithm running in time $(nD)^{O(n \log n)}$ to compute the isolated points of $\mathcal{H}$.

Recently, [3] presents a new algorithm for computing local semi-algebraic paths of a complexity $D^{O(n)}$. Their algorithm also allows to solve the algorithmic algorithm we target.

Main results.. We present two probabilistic algorithms which take as input a polynomial $f \in \mathbb{Q}[x_1, \ldots, x_n]$ and compute the set of isolated points $\mathscr{I}(\mathcal{H})$ of $\mathcal{H} = V(f) \cap \mathbb{R}^n$. Our output consists of a *zero-dimensional parametrization* $\mathscr{C} = (w(t), v_1(t), \ldots, v_n(t)) \in \mathbb{Q}[t]^{n+1}$ such that

$$\{(v_1(t), \ldots, v_n(t)) \mid w(t) = 0\}$$

is a finite set containing $\mathscr{I}(\mathcal{H})$ and a set $\mathscr{B} = (I_1, \ldots, I_s)$ of intervals in $\mathbb{R}$ that satisfies:

- Each I_i has rational endpoints and contains exactly one real root of $w(t)$, namely t_i.
- The set of isolated points of $\mathcal{H}$ is exactly

$$\{(v_1(t_i), \ldots, v_n(t_i)) \mid 1 \leq i \leq s\}.$$

These data represent symbolically the set of isolated points of $\mathcal{H}$ in the sense that one can derive from the pair $(\mathscr{C}, \mathscr{B})$ numerical values of the elements of $\mathscr{I}(\mathcal{H})$ with any required precision.

We sketch now the geometric ingredients which allow us to compute the real isolated points of an algebraic hypersurface defined by $f = 0$.

Assume that f is non-negative over $\mathbb{R}^n$ (if this is not the case, we can replace it by its square) and let $\boldsymbol{x} \in \mathscr{I}(\mathcal{H})$. Since $\boldsymbol{x}$ is isolated and f is non-negative over $\mathbb{R}^n$, the intuition is that for a small enough $e > 0$, the real solution set to $f = e$ looks like a ball around $\boldsymbol{x}$, hence a bounded and closed connected component $C_{\boldsymbol{x}}$. Then the restriction of every projection on the x_i-axis to the algebraic set in $\mathbb{C}^n$ defined by $f = e$ intersects $C_{\boldsymbol{x}}$. When e tends to 0, these critical points in $C_{\boldsymbol{x}}$ "tend to $\boldsymbol{x}$". This first process allows us to compute a subset of $\mathcal{H}$ containing $\mathscr{I}(\mathcal{H})$; we call the elements of this set the candidates. Of course, one would like that this set of candidates is finite and this will be the case up to some generic linear change of coordinates, using e.g. [24].

At this stage, we dispose of a zero-dimensional parametrization $\mathscr{C} = (w(t), v_1(t), \ldots, v_n(t))$ encoding the candidates. One would naturally check whether a ball of infinitesimal radius centered at each candidate intersects $\mathcal{H}$. More precisely, let $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n) \in \mathfrak{C} \cap \mathbb{R}^n$ be a candidate, we decide whether the system

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} (x_i - \eta_i)^2 - \varepsilon = 0 \tag{1}$$

has at least one solution in $\mathbb{R}^n$.

However, this direct approach faces immediately a complexity issue. Writing a quantified formula to solve this decision problem leads to a complexity of $D^{O(n^2)}$ since those points are encoded by a zero-dimensional parametrization of degree $D^{O(n)}$. To bypass this difficulty, we carry out the computation over the quotient ring $\mathbb{Q}[t]/\langle w(t)\rangle$.

This computation relies on the results of [25, Appendix J], which provide an adaptation of the geometric resolution [14] to polynomial systems with coefficients in $\mathbb{Q}[t]/\langle w(t)\rangle$. Using this version of geometric resolution, the resolution of polynomial systems over $\mathbb{Q}[t]/\langle w(t)\rangle$ induces only an additional cost of $O^{\sim}(\deg(w))$ arithmetic operations of $\mathbb{Q}$ comparing to the classic geometric resolution. We will see that the degree of $w(t)$ is actually bounded by $2D(D-1)^{n-1}$. This allows us to obtain an algorithm that uses $D^{O(n)}$ arithmetic operations in $\mathbb{Q}$.

On the algorithmic side, we go further exploiting the geometry of the problem to avoid using infinitesimals. We apply the algorithm to compute a rational number $e_0 > 0$ that replaces the infinitesimals in the system above.

These ingredients allow us to obtain the complexity result below.

Theorem I. *Let $f \in \mathbb{Q}[x_1, \ldots, x_n]$ of degree D and $\mathcal{H} \subset \mathbb{R}^n$ be the real algebraic set defined by $f = 0$. There exists a probabilistic algorithm which, on input f, computes a data $(\mathscr{C}, \mathscr{B})$ encoding $\mathscr{I}(\mathcal{H})$ in case of success using $O^{\sim}\left(64^n D^{8n}\right)$ arithmetic operations in $\mathbb{Q}$ and one call of real root isolation on a univariate polynomial of degree bounded by $2^{2n+2} D^{3n}$.*

Furthermore, we propose an alternative variant that leads to a more efficient algorithm in practice. Once the rational number e_0 is computed, this variant replaces the candidates by their approximations of coordinates in $\mathbb{Q}$ and solves a similar decision problem as the one given by the system (1) for these approximations. Such a strategy allows us to avoid the computation over $\mathbb{Q}[t]/\langle w(t)\rangle$. In Subsection 4.4, we will define rigorously this notion of approximations.

Since this variant makes use of univariate real root isolating algorithms, a complete complexity analysis would require a bound on the bit-size of polynomials given to real root isolating

algorithms. However, we observe that in practice these real root isolating steps are negligible compared to the computation over $\mathbb{Q}[t]/\langle w(t)\rangle$, this variant is therefore much more efficient in practice. A complexity estimate of this variant is given as below.

Theorem II. *Let $f \in \mathbb{Q}[x_1, \ldots, x_n]$ of degree D and $\mathcal{H} \subset \mathbb{R}^n$ be the real algebraic set defined by $f = 0$. There exists a probabilistic algorithm which, on input f, computes a data $(\mathscr{C}, \mathscr{B})$ encoding $\mathscr{I}(\mathcal{H})$ in case of success using $O^{\sim}\left(D^{6n+3}\right)$ arithmetic operations in $\mathbb{Q}$ and two real root isolating calls on a univariate polynomial of degree bounded by $2D(D-1)^{n-1}$.*

Our complexity results above both lie in $D^{O(n)}$ as the algorithm in [3] (up to some root isolation calls). However, we provide an explicit constant in the exponent and the algorithms are designed to enable practical implementation, which constitutes another contribution of this paper.

In Section 6, we present two heuristic subroutines for our implementation. The goal of these subroutines is to avoid as much as possible the most costly computations in our algorithm which include the computation over quotient rings. Taking into account these optimizations, we implement our algorithms in Maple using the libraries FGb [11], RAGLib [23] and msolve [4]. In Section 7, we report on practical experiments showing that they already allow us to solve non-trivial problems which are actually out of reach of [2, Alg. 12.16] which computes sample points in $\mathcal{H}$ only. Unfortunately, the real-life applications coming from material designs still remain intractable.

Organization of the paper.. This paper is structured as follows. Section 2 reviews some fundamental notions in real algebraic geometry that we use further for the geometry analysis of our problem. In Section 3, we prove the auxiliary results which coin the needed theoretical ingredient. Section 4 is devoted to describe our algorithms and their complexities. The heuristic optimizations for implementations are presented in Section 6. Finally, Section 7 reports on the practical performances of our implementation.

Acknowledgment.. The author would like to thank Mohab Safey El Din for helpful discussions and Timo de Wolff for introducing this problem to me.

2. Preliminaries

In what follows, we recall respectively the notions of Puiseux series and critical points.

Puiseux series.. Let ε be an infinitesimal, i.e., a transcendental element over $\mathbb{R}$ such that $0 < \varepsilon < r$ for any positive element $r \in \mathbb{R}$. The field of Puiseux series over $\mathbb{R}$ is defined as all the series with rational exponents

$$\mathbb{R}\langle\varepsilon\rangle = \left\{\sum_{i\geq i_0} a_i \varepsilon^{i/q} \mid i \in \mathbb{N}, i_0 \in \mathbb{Z}, q \in \mathbb{N} - \{0\}, a_i \in \mathbb{R}\right\}.$$

By, e.g., [2, Theorem 2.91], $\mathbb{R}\langle\varepsilon\rangle$ is a real closed field.

One can also define $\mathbb{C}\langle\varepsilon\rangle$ as for $\mathbb{R}\langle\varepsilon\rangle$ but taking the coefficients of the series in $\mathbb{C}$. By [2, Theorem 2.17], the field $\mathbb{C}\langle\varepsilon\rangle$ is an algebraic closure of $\mathbb{R}\langle\varepsilon\rangle$.

Consider $\sigma = \sum_{i\geq i_0} a_i \varepsilon^{i/q} \in \mathbb{R}\langle\varepsilon\rangle$ with $a_{i_0} \neq 0$. Then, a_{i_0} is called the *valuation* of σ. When $i_0 \geq 0$, σ is said to be *bounded over* $\mathbb{R}$ and the set of bounded elements of $\mathbb{R}\langle\varepsilon\rangle$ is denoted by $\mathbb{R}\langle\varepsilon\rangle_b$. One defines the function $\lim_\varepsilon : \mathbb{R}\langle\varepsilon\rangle_b \to \mathbb{R}$ that maps σ to a_0 (which is 0 when $i_0 > 0$) and writes $\lim_\varepsilon \sigma = a_0$; note that $\lim_\varepsilon$ is a ring homomorphism from $\mathbb{R}\langle\varepsilon\rangle_b$ to $\mathbb{R}$. All these definitions

extend to $\mathbb{R}\langle\varepsilon\rangle^n$ componentwise. For a semi-algebraic set $\mathcal{S} \subset \mathbb{R}\langle\varepsilon\rangle^n$, we naturally define the limit of $\mathcal{S}$ as $\lim_\varepsilon \mathcal{S} = \{\lim_\varepsilon \boldsymbol{x} \mid \boldsymbol{x} \in \mathcal{S} \text{ and } \boldsymbol{x} \text{ is bounded over } \mathbb{R}\}$.

We refer to [2, Chap. 2] for more details on infinitesimals and real Puiseux series.

Critical points.. Let $\mathcal{S} \subset \mathbb{R}^n$ be a semi-algebraic set defined by a semi-algebraic formula Φ. We denote by $\mathrm{ext}(\mathcal{S}, \mathbb{R}\langle\varepsilon\rangle)$ the semi-algebraic set of points which are solutions of Φ in $\mathbb{R}\langle\varepsilon\rangle^n$.

Let $\mathbb{K}$ be an algebraically closed field, let $\phi \in \mathbb{K}[x_1, \ldots, x_n]$ which defines the polynomial mapping $(x_1, \ldots, x_n) \mapsto \phi(x_1, \ldots, x_n)$ and $V \subset \mathbb{K}^n$ be a smooth equidimensional algebraic set. We denote by $\mathrm{crit}(\phi, V)$ the set of critical points of the restriction of ϕ to V. If c is the codimension of V and $(g_1, \ldots, g_s)$ generates the vanishing ideal associated to V, then $\mathrm{crit}(\phi, V)$ is the subset of V at which the Jacobian matrix associated to $(g_1, \ldots, g_s, \phi)$ has rank less than or equal to c (see e.g., [25, Subsection 3.1]).

3. Geometric results

3.1. The candidates

As above, let $f \in \mathbb{Q}[x_1, \ldots, x_n]$, $\mathcal{V} \subset \mathbb{C}^n$ be the algebraic hypersurface defined by $f = 0$ and $\mathcal{H} = \mathcal{V} \cap \mathbb{R}^n$. We recall that $\mathscr{I}(\mathcal{H})$ denotes the set of isolated points of $\mathcal{H}$.

By, e.g., [17, Lemma 1], the set $\mathscr{I}(\mathcal{H})$ is the finite union of the connected components of $\mathcal{H}$ which are singletons. A natural start for computing those components is to consider the critical points of the restrictions to $\mathcal{H}$ of the projections

$$\pi_i : (x_1, \ldots, x_n) \mapsto x_i.$$

Since we do not assume that $\mathcal{V}$ is smooth, to apply the critical point method, we retrieve the smoothness through deformation techniques by introducing an infinitesimal ε. More precisely, we consider the hypersurface $\mathcal{V}_\varepsilon$ in $\mathbb{C}\langle\varepsilon\rangle^n$ defined by the equation $f^2 = \varepsilon^2$. Note that $\mathcal{V}_\varepsilon$ is the union of two disjoint algebraic sets $V(f - \varepsilon)$ and $V(f + \varepsilon)$ in $\mathbb{C}\langle\varepsilon\rangle^n$. By e.g., [20, Lemma 3.5], $\mathcal{V}_\varepsilon$ is a smooth algebraic set in $\mathbb{C}\langle\varepsilon\rangle^n$.

Let $\mathcal{H}_\varepsilon = \mathcal{V}_\varepsilon \cap \mathbb{R}\langle\varepsilon\rangle^n$. Two lemmas below will be used regularly in this paper.

Lemma 1. *[20, Lemma 3.6] For every $\boldsymbol{x} \in \mathcal{H}$, there exists a point $\boldsymbol{x}_\varepsilon \in \mathcal{H}_\varepsilon$ such that $\boldsymbol{x}_\varepsilon$ is bounded over $\mathbb{R}$ and $\lim_\varepsilon \boldsymbol{x}_\varepsilon = \boldsymbol{x}$.*

Lemma 2. *[2, Proposition 12.51] Given a point $\boldsymbol{x}$ lying in a bounded connected component of $\mathcal{H}$ and $\boldsymbol{x}_\varepsilon \in \mathcal{H}_\varepsilon$ such that $\boldsymbol{x}_\varepsilon$ is bounded over $\mathbb{R}$ and $\lim_\varepsilon \boldsymbol{x}_\varepsilon = \boldsymbol{x}$, let C_ε be the connected component of $\mathcal{H}_\varepsilon$ containing $\boldsymbol{x}_\varepsilon$. Then, C_ε is bounded over $\mathbb{R}$.*

[17, Lemma 2] that we cite below gives explicitly a subset of $\mathcal{V}$ that contains $\mathscr{I}(\mathcal{H})$.

Proposition 3 ([17, Lemma 2])**.** *Assume that $\mathscr{I}(\mathcal{H})$ is not empty and let $\boldsymbol{x} \in \mathscr{I}(\mathcal{H})$. There exists a semi-algebraically connected component C_ε of $\mathcal{H}_\varepsilon$ such that C_ε is bounded over $\mathbb{R}$ and $\lim_\varepsilon C_\varepsilon = \{\boldsymbol{x}\}$. Consequently, for $1 \leq i \leq n$, there exists a point $\boldsymbol{x}_\varepsilon \in \mathrm{crit}(\pi_i, \mathcal{V}_\varepsilon) \cap C_\varepsilon$ such that $\lim_\varepsilon \boldsymbol{x}_\varepsilon = \boldsymbol{x}$. Then, we have*

$$\mathscr{I}(\mathcal{H}) \subset \cap_{i=1}^n \lim_\varepsilon \mathrm{crit}(\pi_i, \mathcal{V}_\varepsilon) \cap \mathbb{R}^n.$$

To obtain a finite set of candidates, we need to introduce a generic linear change of variable to f. For any $A \in \mathrm{GL}(n, \mathbb{Q})$, let $f^A(\boldsymbol{x}) = f(A \cdot \boldsymbol{x})$ and $\mathcal{V}^A = V(f^A)$. For a generically chosen A, the set $\cap_{i=1}^n \lim_\varepsilon \mathrm{crit}(\pi_i, \mathcal{V}_\varepsilon^A)$ is finite by [22, Theorem 1]. Since $\mathscr{I}(\mathcal{H})$ is the image of the real isolated solutions of $f^A = 0$ by the linear map induced by A^{-1}, we have the proposition below.

Proposition 4. *There exists a Zariski open dense subset of* $\mathrm{GL}(n, \mathbb{Q})$ *such that for any* $A \in \mathcal{A} \cap \mathrm{GL}(n, \mathbb{Q})$*, the image* $\mathfrak{C}$ *of*

$$\cap_{i=1}^n \lim_\varepsilon \mathrm{crit}(\pi_i, \mathcal{V}_\varepsilon^A)$$

by the linear map induced by A^{-1} *is finite and contains* $\mathscr{I}(\mathcal{H})$.

3.2. Identification of the real isolated points

Once the set $\mathfrak{C}$ of candidates is computed, we need to identify whether a candidate $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n) \in \mathfrak{C} \cap \mathbb{R}^n$ is an isolated point of $\mathcal{H}$. To do so, one can check whether a sphere centered at $\boldsymbol{\eta}$ of infinitesimal radius intersects $\mathcal{H}$, which leads one to solve the system

$$f(x_1, \ldots, x_n) = \sum_{i=1}^n (x_i - \eta_i)^2 - \varepsilon = 0$$

over $\mathbb{R}\langle\varepsilon\rangle$. This computation involves an infinitesimal ε, which would prevent a practically efficient algorithm. The objective of this subsection is to present a workaround to avoid the use of infinitesimals.

Let $\boldsymbol{a} = (a_1, \ldots, a_n)$ be an n-uple of positive rational numbers. We consider the function d, depending on $\boldsymbol{a}$, defined by

$$\begin{array}{rccc} d: & \mathbb{R}^n \times \mathbb{R}^n & \to & \mathbb{R} \\ & (x_1, \ldots, x_n, y_1, \ldots, y_n) & \mapsto & \sqrt{\sum_{i=1}^n a_i (x_i - y_i)^2} \end{array}.$$

The function d defines a metric in $\mathbb{R}^n$ which can be extended to $\mathbb{R}\langle\varepsilon\rangle^n$. Further, the notations below denote respectively spheres, open balls and closed balls with respect to the metric d:

- $S(\boldsymbol{x}, r) = \{\boldsymbol{y} \in \mathbb{R}^n \mid d(\boldsymbol{x}, \boldsymbol{y}) = r\}$,
- $B(\boldsymbol{x}, r) = \{\boldsymbol{y} \in \mathbb{R}^n \mid d(\boldsymbol{x}, \boldsymbol{y}) < r\}$,
- $\overline{B}(\boldsymbol{x}, r) = \{\boldsymbol{y} \in \mathbb{R}^n \mid d(\boldsymbol{x}, \boldsymbol{y}) \leq r\}$.

For each $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{C}\langle\varepsilon\rangle^n$, we consider the function $d_{\boldsymbol{x}}$ below:

$$\begin{array}{rccc} d_{\boldsymbol{x}}: & \mathbb{C}\langle\varepsilon\rangle^n & \to & \mathbb{C}\langle\varepsilon\rangle \\ & (y_1, \ldots, y_n) & \mapsto & \sum_{i=1}^n a_i (y_i - x_i)^2. \end{array}$$

Recall that the algebraic set $\mathcal{V}_\varepsilon$ defined by $f^2 = \varepsilon^2$ in $\mathbb{C}\langle\varepsilon\rangle^n$, is smooth. By an algebraic variant of Sard's theorem (see [25, Prop. B.2], [5, Theorem 9.6.2]), the critical values of the restriction of $d_{\boldsymbol{x}}$ to $\mathcal{V}_\varepsilon$ form a finite subset of $\mathbb{C}\langle\varepsilon\rangle$. Therefore, for every candidate $\boldsymbol{x} \in \mathfrak{C} \cap \mathbb{R}^n$, the set

$$\mathfrak{D}(\boldsymbol{x}) = \{d_{\boldsymbol{x}}(\lim_\varepsilon \boldsymbol{y}_\varepsilon) \mid \boldsymbol{y}_\varepsilon \in \mathrm{crit}(d_{\boldsymbol{x}}, \mathcal{V}_\varepsilon) \cap \mathbb{R}\langle\varepsilon\rangle_b^n,\ \lim_\varepsilon \boldsymbol{y}_\varepsilon \neq \boldsymbol{x}\}$$

is a finite set of positive elements of $\mathbb{R}$. When $\mathfrak{D}(\boldsymbol{x})$ is not empty, Lemma 5 below allows us to identify whether the point $\boldsymbol{x}$ is isolated in $\mathcal{H}$.

Lemma 5. *Let $\boldsymbol{x} \in \mathcal{H}$ and $C_{\boldsymbol{x}}$ be the connected component of $\mathcal{H}$ containing $\boldsymbol{x}$. Assume that the set $\mathfrak{D}(\boldsymbol{x})$ defined as above is not empty. Let $e_{\boldsymbol{x}} \in \mathbb{R}$ such that*

$$0 < e_{\boldsymbol{x}} < \min \mathfrak{D}(\boldsymbol{x}).$$

Then, the following statements are equivalent:

i) $\boldsymbol{x}$ is an isolated point of $\mathcal{H}$.

ii) There exists $e \in]0, e_{\boldsymbol{x}}[$ such that $\mathcal{H} \cap S(\boldsymbol{x}, \sqrt{e}) = \emptyset$.

iii) For every $e \in]0, e_{\boldsymbol{x}}[$, $\mathcal{H} \cap S(\boldsymbol{x}, \sqrt{e}) = \emptyset$.

Moreover, if $\boldsymbol{x}$ is not an isolated point of $\mathcal{H}$, then $C_{\boldsymbol{x}}$ intersects $S(\boldsymbol{x}, \sqrt{e})$ for every $e \in]0, e_{\boldsymbol{x}}[$.

Proof. By the definition of real isolated points, we immediately have that (i) implies (ii) and (iii) implies (i). It remains to demonstrate that (iii) is a consequence of (ii), which we separate into two statements: (ii) leads to (i) and then (i) leads to (iii).

We now show that (ii) implies (i) by contradiction. We assume that the point $\boldsymbol{x}$ is not an isolated point of $\mathcal{H}$. If $C_{\boldsymbol{x}}$ is not bounded, $C_{\boldsymbol{x}}$ intersects $S(\boldsymbol{x}, \sqrt{e})$ for every $e > 0$ and there is nothing to be proved. We now assume that $C_{\boldsymbol{x}}$ is bounded.

By Lemma 1, there exists a point $\boldsymbol{x}_\varepsilon$ such that $\boldsymbol{x}_\varepsilon$ is bounded over $\mathbb{R}$ and $\lim_\varepsilon \boldsymbol{x}_\varepsilon = \boldsymbol{x}$. Let $C_\varepsilon \subset \mathbb{R}\langle\varepsilon\rangle^n$ be a connected component of the real algebraic set $\mathcal{H}_\varepsilon$ containing $\boldsymbol{x}_\varepsilon$. By Lemma 2, C_ε is bounded over $\mathbb{R}$ and, thus, its limit is connected in $\mathbb{R}^n$ by [2, Proposition 12.49]. Hence, $\lim_\varepsilon C_\varepsilon$ is a connected subset of $C_{\boldsymbol{x}}$.

Moreover, as C_ε is closed and bounded over $\mathbb{R}$, the restriction of $d_{\boldsymbol{x}}$ to C_ε reaches its maximum at some point $\boldsymbol{y}_\varepsilon \in C_\varepsilon$ (see [5, Theorem 2.5.8]). Note that $\boldsymbol{y}_\varepsilon \in \operatorname{crit}(d_{\boldsymbol{x}}, \mathcal{V}_\varepsilon) \cap \mathbb{R}\langle\varepsilon\rangle_b^n$. So, we have that $d_{\boldsymbol{x}}(\lim_\varepsilon \boldsymbol{y}_\varepsilon) \geq e_{\boldsymbol{x}}$. Therefore, for any $e \in]0, e_{\boldsymbol{x}}[$, the closed ball $\overline{B}(\boldsymbol{x}, \sqrt{e})$ does not contain $\lim_\varepsilon \boldsymbol{y}_\varepsilon$. Since $\lim_\varepsilon C_\varepsilon$ is connected in $\mathbb{R}^n$ and contains $\boldsymbol{x}$ and $\lim_\varepsilon \boldsymbol{y}_\varepsilon$, there exists a semi-algebraic continuous function $\gamma : [0, 1] \to \lim_\varepsilon C_\varepsilon$ such that $\gamma(0) = \boldsymbol{x}$ and $\gamma(1) = \lim_\varepsilon \boldsymbol{y}_\varepsilon$. As $d_{\boldsymbol{x}}(\boldsymbol{x}) = 0$ and $d_{\boldsymbol{x}}(\lim_\varepsilon \boldsymbol{y}_\varepsilon) \geq e_{\boldsymbol{x}}$, by the intermediate value property [2, Proposition 3.5], there exists $t_0 \in]0, 1[$ such that $d_{\boldsymbol{x}}(\gamma(t_0)) = e$ for any $e \in]0, e_{\boldsymbol{x}}[$. Therefore, the connected component $C_{\boldsymbol{x}}$ intersects $S(\boldsymbol{x}, \sqrt{e})$ at $\gamma(t_0)$.

So, (ii) does not hold either when $C_{\boldsymbol{x}}$ is bounded. We conclude that (ii) leads to (i).

Finally, it remains to show that (i) implies (iii). Again, we prove this by contradiction. Assume that there exists $e \in]0, e_{\boldsymbol{x}}[$ such that $\mathcal{H} \cap S(\boldsymbol{x}, \sqrt{e}) \neq \emptyset$. Equivalently, there exists a point $z \in \mathcal{H}$ such that $d(z) = e \in]0, e_{\boldsymbol{x}}[$.

By Lemma 1, there exists a point $z_\varepsilon \in \mathcal{H}_\varepsilon$ such that $\lim_\varepsilon z_\varepsilon = z$. Let $C_{z,\varepsilon}$ be the connected component of $\mathcal{H}_\varepsilon$ containing z_ε.

In the closed and connected semi-algebraic set $C_{z,\varepsilon}$, there exists a point z'_ε at which the restriction of $d_{\boldsymbol{x}}$ to $C_{z,\varepsilon}$ reaches it minimum. So, z'_ε belongs to $\operatorname{crit}(d_{\boldsymbol{x}}, \mathcal{V}_\varepsilon) \cap \mathbb{R}\langle\varepsilon\rangle_b^n$. Thus, we have that

$$d_{\boldsymbol{x}}(\lim_\varepsilon z'_\varepsilon) \leq d_{\boldsymbol{x}}(\lim_\varepsilon z_\varepsilon) < e_{\boldsymbol{x}}.$$

Using the definition of $e_{\boldsymbol{x}}$, we deduce that $d_{\boldsymbol{x}}(\lim_\varepsilon z'_\varepsilon) = 0$, which is equivalent to $\lim_\varepsilon z'_\varepsilon = \boldsymbol{x}$.

So, both z_ε and z'_ε lie in the connected component $C_{z,\varepsilon}$ of $\mathcal{H}_\varepsilon$ and $\lim_\varepsilon z'_\varepsilon = \boldsymbol{x}$. If $C_{z,\varepsilon}$ is not bounded over $\mathbb{R}$, by Lemma 2, we have that $C_{\boldsymbol{x}}$ is not bounded, which implies immediately that $\boldsymbol{x}$ is not an isolated point of $\mathcal{H}$.

Otherwise, when $C_{z,\varepsilon}$ is bounded over $\mathbb{R}$, by [2, Proposition 12.49], $\lim_\varepsilon C_{z,\varepsilon}$ is a connected subset of $\mathcal{H}$ that contains $\boldsymbol{x}$ and z. In this case, we also conclude that $\boldsymbol{x}$ is not isolated in $\mathcal{H}$. Therefore, (i) leads to (iii), which finishes our proof. □

Lemma 6 handles the remaining case when a point $\boldsymbol{x} \in \mathcal{H}$ such that $\mathfrak{D}(\boldsymbol{x})$ is empty exists.

Lemma 6. *Assume that there exists $\boldsymbol{x} \in \mathcal{H}$ such that $\mathfrak{D}(\boldsymbol{x})$ is empty. Then, exactly one among the two statements below holds:*

i) $\mathcal{H}$ is connected and not bounded, so it does not have any isolated point.

ii) $\mathcal{H}$ is a single point $\boldsymbol{x}$.

Proof. Assume that $\mathcal{H}$ has at least two connected components. Then, there exists a connected component of $\mathcal{H}_\varepsilon$ that does not contain any point whose limit is $\boldsymbol{x}$. Therefore, the restriction of $d_{\boldsymbol{x}}$ to this connected component admits a critical point over this connected component. In consequence, $\mathfrak{D}(\boldsymbol{x})$ is not empty, which contradicts the assumption of Lemma 6. Therefore, $\mathcal{H}$ has exactly one connected component.

Assume now if $\mathcal{H}$ is bounded and is not a single point $\boldsymbol{x}$. As a consequence of Lemma 2, there exists a connected component $C_{\boldsymbol{x},\varepsilon}$, that is bounded over $\mathbb{R}$, of $\mathcal{H}_\varepsilon$ such that $\{\boldsymbol{x}\} \subsetneq \lim_\varepsilon C_{\boldsymbol{x},\varepsilon}$. Thus, the distance function $\delta_{\boldsymbol{x}}$ attains its maximum, which is non-zero, over $\mathcal{H}$. This contradicts the assumption that $\mathfrak{D}(\boldsymbol{x})$ is empty. Thus, we conclude the proof. □

When $\mathfrak{D}(\boldsymbol{x})$ is empty, we define by convention $\min \mathfrak{D}(\boldsymbol{x}) = +\infty$. Let $e_0 \in \mathbb{R}$ such that

$$0 < e_0 < \min_{\boldsymbol{x} \in \mathfrak{C} \cap \mathbb{R}^n} \min \mathfrak{D}(\boldsymbol{x}).$$

We deduce from Lemmas 5 and 6 the following proposition, which is the main criteria for designing our algorithms in Section 4.

Proposition 7. *Let $\boldsymbol{a} = (a_1, \ldots, a_n) \in \mathbb{R}_+^n$ and $e_0 \in \mathbb{R}$ defined as above. Then, for any candidate $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathfrak{C} \cap \mathbb{R}^n$, $\boldsymbol{x}$ is an isolated point of $\mathcal{H}$ if and only if the polynomial system*

$$f(y_1, \ldots, y_n) = \sum_{i=1}^{n} a_i (y_i - x_i)^2 - e_0 = 0$$

admits at least one solution in $\mathbb{R}^n$.

Proof. We first assume that $\mathfrak{D}(\boldsymbol{x})$ is not empty. Using Lemma 5, we have that $\boldsymbol{x}$ is an isolated point of $\mathcal{H}$ if and only if $\mathcal{H}$ intersects the sphere $S(\boldsymbol{x}, \sqrt{e_0})$.

Otherwise, if $\mathfrak{D}(\boldsymbol{x})$ is empty, by Lemma 6, $\mathcal{H}$ is either a single point $\boldsymbol{x}$ or an unbounded connected component containing $\boldsymbol{x}$. The similar conclusion follows immediately. □

4. Algorithms

4.1. Overview of the algorithms

Our algorithms take as input $f \in \mathbb{Q}[x_1, \ldots, x_n]$ and compute:

- A zero-dimensional parametrization $\mathscr{C} = (w, v_1, \ldots, v_n)$ encoding the candidates such that

$$\mathscr{I}(\mathcal{H}) \subset \{(v_1(t), \ldots, v_n(t)) \mid w(t) = 0\};$$

- A finite set $\mathscr{B}$ of intervals isolating the real solutions of $w(t)$ which correspond to $\mathscr{I}(\mathcal{H})$.

Both algorithms share the first step of computing the set of candidates $\mathfrak{C}$. This is done by the subroutine Candidates that takes as input the polynomial f and returns a zero-dimensional parametrization $\mathscr{C}$ encoding $\mathfrak{C}$. The design of this subroutine is given in [17, Subsection 3.2], which mimics the computation in [22]. Note that Candidates is probabilistic. As explained in Proposition 4, it computes the candidates for $f^A = f(A \cdot \boldsymbol{x})$ where A is randomly chosen from $\mathrm{GL}(n, \mathbb{Q})$ then reverses the change of variables to obtain the candidates for f.

Next, we pick randomly an n-tuple $\boldsymbol{a} = (a_1, \ldots, a_n)$ from $\mathbb{Q}_+^n$, Proposition 7 requires us to compute a value $e_0 \in \mathbb{Q}$ such that for every $\boldsymbol{x} \in \mathfrak{C} \cap \mathbb{R}^n$,

$$0 < e_0 < \min\{d_{\boldsymbol{x}}(\lim_{\varepsilon} \boldsymbol{y}_\varepsilon) \mid \boldsymbol{y}_\varepsilon \in \mathrm{crit}(d_{\boldsymbol{x}}, \mathcal{V}_\varepsilon) \cap \mathbb{R}\langle\varepsilon\rangle_b^n,\ \lim_{\varepsilon} \boldsymbol{y}_\varepsilon \neq \boldsymbol{x}\},$$

where $d_{\boldsymbol{x}}$ is defined as

$$(y_1, \ldots, y_n) \mapsto a_1(x_1 - y_1)^2 + \ldots + a_n(x_n - y_n)^2.$$

We call ComputeE0 a subroutine that takes as input $f \in \mathbb{Q}[x_1, \ldots, x_n]$ and $\boldsymbol{a} \in \mathbb{Q}_+^n$ and returns such an $e_0 \in \mathbb{Q}_+$. The explicit description of ComputeE0 is given in Subsection 4.2. From this value e_0, Proposition 7 identifies whether a candidate $\boldsymbol{\eta} \in \mathfrak{C} \cap \mathbb{R}^n$ is isolated in $\mathcal{H}$ by deciding the emptiness of $\mathcal{H} \cap S(\boldsymbol{\eta}, \sqrt{e_0})$. This leads us to solve the following polynomial system over $\mathbb{R}^n$ for each candidate $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n) \in \mathfrak{C} \cap \mathbb{R}^n$:

$$f(x_1, \ldots, x_n) = a_1(x_1 - \eta_1)^2 + \ldots + a_n(x_n - \eta_n)^2 - e_0 = 0. \tag{2}$$

Therefore, Algorithm 1 below gives the outline of our algorithms in which the subroutine IsIsolated that takes as input f, $\mathscr{C}$, $\boldsymbol{a}$ and e_0 and computes the isolating intervals $\mathscr{B}$. It will be designed differently for each of our algorithms.

Algorithm 1: IsolatedPoints

Input: A polynomial $f \in \mathbb{Q}[x_1, \ldots, x_n]$
Output: A zero-dimensional parametrization $\mathscr{C}$ and a set $\mathscr{B}$ of intervals in $\mathbb{R}$

1 A chosen randomly from $\mathrm{GL}(n, \mathbb{Q})$
2 $\mathscr{C} \leftarrow$ Candidates(f, A)
3 $\boldsymbol{a}$ chosen randomly in $\mathbb{Q}_+^n$
4 $e_0 \leftarrow$ ComputeE0$(f, \mathscr{C}, \boldsymbol{a})$
5 $\mathscr{B} \leftarrow$ IsIsolated$(f, \mathscr{C}, \boldsymbol{a}, e_0)$
6 **return** $(\mathscr{C}, \mathscr{B})$

Since the candidates are given by $\mathscr{C}$, we cannot treat directly the system (2) and need some workarounds. The first variant of IsIsolated considers the system

$$f(x_1, \ldots, x_n) = a_1(x_1 - v_1(t))^2 + \ldots + a_n(x_n - v_n(t))^2 - e_0 = w(t) = 0. \tag{3}$$

We need to identify for which real roots of $w(t)$, the above system has at least one real solution. To do so, we basically compute a polynomial $Q(t, z) \in \mathbb{Q}[t, z]$ such that for any real root t_0 of w, $Q(t_0, z)$ admits real solutions if and only if the system (3) does too. The problem is hence reduced to a bivariate setting, which can be solved easily by classical real root counting algorithms. The design details of this variant is explained in Subsection 4.3.

The second variant of IsIsolated is explained in Subsection 4.4. The idea is to take advantage of the knowledge of e_0, we replace the candidate $\boldsymbol{\eta}$ in the system (2) by an "approximation"

$\tilde{\boldsymbol{\eta}} \in \mathbb{Q}^n$ of $\boldsymbol{\eta}$ and establish a similar result as Proposition 7 for these approximations (see Lemma 10). Briefly, we claim that a candidate $\boldsymbol{\eta}$ is an isolated point of $\mathcal{H}$ if and only if the system

$$f(x_1, \ldots, x_n) = a_1(x_1 - \tilde{\eta}_1)^2 + \ldots + a_n(x_n - \tilde{\eta}_n)^2 - \frac{e_0}{4} = 0$$

has no real solution. Therefore, once those approximations are identified, one can apply classical real root finding algorithms to the above system.

4.2. Computing an appropriate value for e_0

In this subsection, we describe a subroutine that computes a value e_0 introduced in Proposition 7. Recall that, for each $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n) \in \mathbb{C}^n$, the function $d_{\boldsymbol{\eta}}$ is defined as

$$\begin{array}{rccc} d_{\boldsymbol{\eta}}: & \mathbb{C}^n & \to & \mathbb{C}, \\ & \boldsymbol{x} = (x_1, \ldots, x_n) & \mapsto & \sum_{i=1}^n a_i(x_i - \eta_i)^2. \end{array}$$

To apply Lemma 5, we need to compute a value $e_0 \in \mathbb{Q}$ such that for every $\boldsymbol{\eta} \in \mathfrak{C} \cap \mathbb{R}^n$

$$0 < e_0 < \min\{d_{\boldsymbol{\eta}}(\lim_{\varepsilon} \boldsymbol{x}_\varepsilon) \mid \boldsymbol{x}_\varepsilon \in \mathrm{crit}(d_{\boldsymbol{\eta}}, \mathcal{V}_\varepsilon) \cap \mathbb{R}\langle\varepsilon\rangle_b^n,\ \lim_{\varepsilon} \boldsymbol{x}_\varepsilon \neq \boldsymbol{\eta}\}.$$

Lemma 8 shows that, for a generic choice of $\boldsymbol{a}$, every critical locus $\mathrm{crit}(d_{\boldsymbol{\eta}}, \mathcal{V}_\varepsilon)$ is finite.

Lemma 8. *Let $\boldsymbol{\eta} \in \mathfrak{C}$ be a candidate. Then there exists a non-empty Zariski open subset $\mathcal{A}$ of $\mathbb{C}^n$ such that, for $\boldsymbol{a} = (a_1, \ldots, a_n) \in \mathcal{A} \cap \mathbb{Q}_+^n$, the critical locus $\mathrm{crit}\left(d_{\boldsymbol{\eta}}, \mathcal{V}_\varepsilon\right)$ is finite.*

Proof. Since $V(f - \varepsilon) \subset \mathbb{C}\langle\varepsilon\rangle^n$ is smooth, the critical locus $\mathrm{crit}\left(d_{\boldsymbol{\eta}}, V(f - \varepsilon)\right)$ is defined by

$$f - \varepsilon = y \cdot \frac{\partial f}{\partial x_1} - 2a_i(x_1 - \eta_1) = \cdots = y \cdot \frac{\partial f}{\partial x_n} - 2a_n(x_n - \eta_n) = 0. \tag{4}$$

Now we consider $\boldsymbol{a} = (a_1, \ldots, a_n)$ and y as indeterminates. Let $\varphi : \mathbb{C}^n \times \mathbb{C}^n \times \mathbb{C}$ be the polynomial mapping defined as

$$(\boldsymbol{x}, \boldsymbol{a}, y) \mapsto \left(f - \varepsilon, y \cdot \frac{\partial f}{\partial x_1} - 2a_1(x_1 - \eta_1), \ldots, y \cdot \frac{\partial f}{\partial x_n} - 2a_n(x_n - \eta_n)\right).$$

Let $\mathcal{X}$ be the non-empty Zariski open subset of $\mathbb{C}$ defined as

$$(x_1 - \eta_1) \cdots (x_n - \eta_n) \neq 0.$$

The Jacobian matrix of φ with respect to $(\boldsymbol{x}, \boldsymbol{a}, y)$

$$\begin{bmatrix} \frac{\partial f}{\partial x_1} & \cdots & \frac{\partial f}{\partial x_n} & 0 & 0 & \cdots & 0 \\ * & \cdots & * & \frac{\partial f}{\partial x_1} & -2(x_1 - \eta_1) & \cdots & 0 \\ \vdots & \ddots & * & \vdots & \vdots & \ddots & \vdots \\ * & \cdots & * & \frac{\partial f}{\partial x_n} & 0 & \cdots & -2(x_n - \eta_n) \end{bmatrix}$$

has full rank when $\boldsymbol{x} \in \mathcal{X}$ and

$$f - \varepsilon = y \cdot \frac{\partial f}{\partial x_1} - 2a_i(x_1 - \eta_1) = \cdots = y \cdot \frac{\partial f}{\partial x_n} - 2a_n(x_n - \eta_n) = 0.$$

By Thom's weak transversality theorem [8, Theorem 3.7.4], there exists a non-empty Zariski open subset $\mathcal{A}_\emptyset$ of $\mathbb{C}^n$ such that, for $\boldsymbol{a} \in \mathcal{A}_\emptyset$, $\mathbf{0}$ is a regular value of the restriction of $\varphi_{\boldsymbol{a}}$ to $\mathcal{X}$. Thus, for $\boldsymbol{a} \in \mathcal{A}_\emptyset$, by Jacobian criterion, the restriction of the solutions of

$$f - \varepsilon = y \cdot \frac{\partial f}{\partial x_1} - 2a_i(x_1 - \eta_1) = \cdots = y \cdot \frac{\partial f}{\partial x_n} - 2a_n(x_n - \eta_n) = 0$$

to $\mathcal{X}$ is a finite set, i.e., $\mathrm{crit}(d_{\boldsymbol{\eta}}, V(f-\varepsilon)) \cap \mathcal{X}$ is finite.

Now we study the restriction of $\mathrm{crit}(d_{\boldsymbol{\eta}}, V(f-\varepsilon))$ to $\mathbb{C}^n \setminus \mathcal{X}$. We choose $\boldsymbol{a} \in \mathbb{Q}_+^n$. Let I be a non-empty proper subset of $\{1, \ldots, n\}$ and $\mathcal{X}_I$ be the subset of $\mathbb{C}^n$ defined by

$$x_i = \eta_i \quad \text{for } i \in I \quad \text{and} \quad x_i \neq \eta_i \quad \text{for } i \notin \{1, \ldots, n\} \setminus I.$$

Let $\boldsymbol{x} \in \mathrm{crit}(d_{\boldsymbol{\eta}}, V(f-\varepsilon)) \cap \mathcal{X}_I$. As $f(\boldsymbol{\eta}) = 0$, $\boldsymbol{\eta} \notin V(f-\varepsilon)$ and $\boldsymbol{x} \neq \boldsymbol{\eta}$. Hence, $y \neq 0$ in the system (4). Hence, $y \cdot \frac{\partial f}{\partial x_i} - 2a_1(x_i - \eta_i) = 0$ implies that $\frac{\partial f}{\partial x_i} = 0$. Since $V(f-\varepsilon)$ is smooth, $\frac{\partial f}{\partial x_i}$ for $i \in \{1, \ldots, n\} \setminus I$ cannot vanish simultaneously at $\boldsymbol{x}$. This means

$$\{\boldsymbol{x} \mid \boldsymbol{x} \in \mathrm{crit}(d_{\boldsymbol{\eta}}, V(f-\varepsilon)), x_i = \eta_i \text{ for } i \in I\}$$

coincides with the critical locus $\mathrm{crit}(d_{\boldsymbol{\eta},I}, V(f_I - \varepsilon))$ where

$$d_{\boldsymbol{\eta},I} : (x_j)_{j \in \{1,\ldots,n\} \setminus I} \mapsto \sum_{j \in \{1,\ldots,n\} \setminus I} a_j (x_j - \eta_j)^2$$

and f_I is the polynomial obtained from f by substituting $x_i = \eta_i$ for $i \in I$. Therefore, we can use the same arguments as above to prove the following.

There exists a non-empty Zariski open subset $\mathcal{A}_I$ of $\mathbb{C}^n$ such that for $\boldsymbol{a} \in \mathcal{A}_I \cap \mathbb{Q}_+^n$, the restriction of $\mathrm{crit}(d_{\boldsymbol{\eta}}, V(f-\varepsilon))$ to $\mathcal{X}_I$ is finite.

Let $\mathcal{A}_+ = \cap_{I \subsetneq \{1,\ldots,n\}} \mathcal{A}_I$ which is a non-empty Zariski open subset of $\mathbb{C}^n$. Given any $\boldsymbol{a} \in \mathcal{A}_+ \cap \mathbb{Q}_+^n$, $\mathrm{crit}(d_{\boldsymbol{\eta}}, V(f-\varepsilon)$ is a finite set. Similarly for $V(f+\varepsilon)$, we obtain a non-empty Zariski open subset $\mathcal{A}_-$. Taking the intersection $\mathcal{A} = \mathcal{A}_+ \cap \mathcal{A}_-$ ends the proof. □

Since the set of candidates $\mathfrak{C}$ is encoded by a zero-dimensional parametrization $\mathscr{C}$, we do the whole computation at once through the function δ defined as

$$\begin{array}{rccc} \delta : & \mathbb{C}^n \times \mathbb{C} & \to & \mathbb{C}, \\ & (x_1, \ldots, x_n, t) & \mapsto & \sum_{i=1}^n a_i (x_i - v_i(t))^2. \end{array}$$

The following lemma is immediate.

Lemma 9. *Let $\mathcal{V}_{\varepsilon,t} \subset \mathbb{C}\langle\varepsilon\rangle^{n+1}$ be the algebraic set defined by $f^2 - \varepsilon^2 = w(t) = 0$. Then, the set of critical values $\delta(\mathrm{crit}(\delta, \mathcal{V}_{\varepsilon,t}))$ is the union of $d_{\boldsymbol{\eta}}(\mathrm{crit}(d_{\boldsymbol{\eta}}, \mathcal{V}_\varepsilon))$ for $\boldsymbol{\eta} \in \mathfrak{C}$.*

Proof. The set $\mathrm{crit}(\delta, \mathcal{V}_{\varepsilon,t})$ are defined by the points of $\mathcal{V}_{\varepsilon,t}$ at which the matrix

$$\begin{bmatrix} \frac{\partial f^2}{\partial x_1} & \cdots & \frac{\partial f^2}{\partial x_n} & 0 \\ \frac{\partial \delta}{\partial x_1} & \cdots & \frac{\partial \delta}{\partial x_n} & \frac{\partial \delta}{\partial t} \\ 0 & \ldots & 0 & w'(t) \end{bmatrix}$$

has rank at most 2.

As $w(t)$ is square-free, for every t_0 such that $w(t_0) = 0$, $w'(t_0)$ is not zero. Therefore, the condition above restricted to $\mathcal{V}_{\varepsilon,t}$ is equivalent to

$$\operatorname{rank} \begin{bmatrix} \frac{\partial f^2}{\partial x_1} & \cdots & \frac{\partial f^2}{\partial x_n} \\ \frac{\partial \delta}{\partial x_1} & \cdots & \frac{\partial \delta}{\partial x_n} \end{bmatrix} \leq 1.$$

For every complex root t_0 of $w(t)$, let $\boldsymbol{\eta}_0 = (v_1(t_0), \ldots, v_n(t_0))$. By fixing $t = t_0$, the rank condition above is reduced to

$$\operatorname{rank} \begin{bmatrix} \frac{\partial f^2}{\partial x_1} & \cdots & \frac{\partial f^2}{\partial x_n} \\ \frac{\partial \delta_{\boldsymbol{\eta}_0}}{\partial x_1} & \cdots & \frac{\partial \delta_{\boldsymbol{\eta}_0}}{\partial x_n} \end{bmatrix} \leq 1,$$

which defines the set $\operatorname{crit}(\delta_{\boldsymbol{\eta}_0}, \mathcal{H}_\varepsilon)$.

Thus, $\operatorname{crit}(\delta, \mathcal{H}_{\varepsilon,t}) = \cup_{w(t_0)=0}\{(\boldsymbol{x}, t_0) \mid \boldsymbol{\eta}_0 = (v_1(t_0), \ldots, v_n(t_0)),\ \boldsymbol{x} \in \operatorname{crit}(\delta_{\boldsymbol{\eta}_0}, \mathcal{H}_\varepsilon)\}$. This concludes the proof. □

Now we aim to compute the limit of the critical points and their corresponding values of the restriction of δ to the algebraic set $\mathcal{H}_{\varepsilon,t}$ defined by $f^2 - \varepsilon^2 = w(t) = 0$. Note that Lemmas 8 and 9 imply that, for a generic $\boldsymbol{a} \in \mathbb{Q}_+^n$, the set of critical points $\operatorname{crit}(\delta, \mathcal{H}_{\varepsilon,t})$ is finite.

By [22, Theorems 1, 2], for generic values of $\boldsymbol{a} \in \mathbb{Q}^n$, we have that

$$\lim_{\varepsilon} \operatorname{crit}(\delta, \mathcal{H}_{\varepsilon,t}) \subset \langle f \rangle + \left\langle w(t), y \cdot \frac{\partial f}{\partial x_i} - \frac{\partial \delta}{\partial x_i} \text{ for every } 1 \leq i \leq n \right\rangle \cap \mathbb{Q}[\boldsymbol{x}, t]$$

and the ideal on the right-hand side is zero-dimensional.

From the above inclusion, one can follow a similar computation as in [22] using the geometric resolution algorithm [14]. However, as the degree of $w(t)$ is bounded by $2D(D-1)^{n-1}$ (see Section 5), such a computation would lead to an arithmetic complexity $D^{O(n^2)}$.

A workaround to obtain a better complexity is to use a variant of geometric resolution over the quotient ring $\mathbb{A} = \mathbb{Q}[t]/\langle w(t) \rangle$ as explained in [25, Appendix J]. Note that $w(t)$ is not necessarily irreducible, the extension $\mathbb{A}$ is only a product of fields and doing the computation over the ring $\mathbb{A}$ is not trivial. We will see in Subsection 5 that this approach allows us to obtain an algorithm with arithmetic complexity lying in $D^{O(n)}$.

Our subroutine ComputeE0 is designed as follows.

a) First, we call a subroutine ParametricCurve that takes as input f, $\mathscr{C}$, $\boldsymbol{a} \in \mathbb{Q}_+^n$ and $i \in \{1, \ldots, n\}$ and computes a one-dimensional parametrization $\mathscr{J}_i$ over $\mathbb{Q}[t]/\langle w(t) \rangle$ of the system

$$\left(\frac{\partial \delta}{\partial x_j} \cdot \frac{\partial f}{\partial x_i} - \frac{\partial \delta}{\partial x_i} \cdot \frac{\partial f}{\partial x_j} = 0 \right)_{j \in \{1,\ldots,n\} \setminus \{i\}} \quad \text{and} \quad \frac{\partial f}{\partial x_i} \neq 0.$$

An explicit description of this subroutine can be found in [25, Appendix J.5].

b) Next, we compute a zero-dimensional parametrization $\mathscr{E}_i$ of the intersection of $\mathcal{H} = V(f)$ with the sets of solutions defined by the parametrizations $\mathscr{J}_i$ above.

This is done by calling a subroutine IntersectCurve on the input $\mathscr{J}_i$ and f, which is described also in [25, Appendix J.5].

c) We then call a subroutine Union that computes a zero-dimensional parametrization $\mathscr{E}$ that defines $\cup_{i=1}^n Z(\mathscr{E}_i)$.

d) Finally, taking as input the zero-dimensional parametrization $\mathscr{E}$, we call a subroutine GetE0 that computes the required value e_0. This can be done by calling FGLM algorithm [12] to compute a polynomial $P(e)$ whose solutions encode the values $e = \sum_{i=1} a_i(x_i - v_i(t))^2$ for $\boldsymbol{x} \in Z(\mathscr{E})$ and $w(t) = 0$. Next, we evaluate a lower bound of the minimal distance between the roots of $P(e)$ using [2, Proposition 10.23].

Algorithm 2: ComputeE0

Input: $f \in \mathbb{Q}[x_1, \ldots, x_n]$, $\mathscr{C} = (w(t), v_1(t), \ldots, v_n(t))$ and $\boldsymbol{a} \in \mathbb{Q}_+^n$
Output: $e_0 \in \mathbb{Q}$

```
δ ← a_1(x_1 − v_1(t))^2 + ... + a_n(x_n − v_n(t))^2
for 1 ≤ i ≤ n do
    J_i ← ParametricCurve(f, a, C, i)
    E_i ← IntersectCurve(J_i, f)
E ← Union(E_1, ..., E_n)
e_0 ← GetE0(E)
return e_0
```

4.3. The first variant of IsIsolated

In this subsection, we explain the details of the first variant of IsIsolated.

Using the value e_0 output by Algorithm 2, Proposition 7 allows one to identify the isolated points of $\mathcal{H}$ among the candidates by checking whether the polynomial system

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} a_i(x_i - \eta_i)^2 - e_0 = 0$$

admits real solutions for each candidate $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n) \in \mathfrak{C} \cap \mathbb{R}^n$. Again, one can consider the system

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} a_i(x_i - v_i(t))^2 - e_0 = w(t) = 0 \tag{5}$$

to handle all the candidates at once.

Let $\mathcal{W}_t \subset \mathbb{C}^{n+1}$ be the algebraic set defined the equation (5). Our strategy is to compute a finite subset of $\mathcal{W}_t \cap \mathbb{R}^{n+1}$ that intersects every connected component of $\mathcal{W}_t \cap \mathbb{R}^{n+1}$. Then, all the real t-coordinates of those sample points correspond to the isolated points of $\mathcal{H} \cap \mathbb{R}^n$.

We consider the polynomial

$$F = f(x_1, \ldots, x_n)^2 + \left(\sum_{i=1}^{n} a_i(x_i - v_i(t))^2 - e_0 \right)^2.$$

Note that $F + w(t)^2$ defines also the real algebraic set $\mathcal{W}_t \cap \mathbb{R}^{n+1}$. Therefore, the sample points above can be computed using the algorithm of [22] on the input $F + w(t)^2 \subset \mathbb{Q}[t, x_1, \ldots, x_n]$. Such an algorithm returns a zero-dimensional parametrization over $\mathbb{Q}$ that defines a finite set intersects every connected component of $\mathcal{W}_t$. Since the total degree of $F + w(t)^2$ can go up to $O(D^n)$, this computation faces the same complexity issue as in Subsection 4.2. Again, we can bypass this problem by solving over $\mathbb{A}[x_1, \ldots, x_n]$ where $\mathbb{A} = \mathbb{Q}[t]/\langle w(t) \rangle$.

Let B be a matrix randomly chosen from $\mathrm{GL}(n, \mathbb{Q})$ and $F^B(\boldsymbol{x}) = F(B \cdot \boldsymbol{x})$. We apply the geometric resolution algorithm over $\mathbb{A}$ on the system of equations:

$$F^B = 0, \quad \frac{\partial F^B}{\partial x_j} = 0, \quad \frac{\partial F^B}{\partial x_1} \neq 0.$$

This algorithm returns a zero-dimensional parametrization $(U(z), V_1(z), \ldots, V_n(z))$ over the ring $\mathbb{A}$, which means that $V_1, \ldots, V_n$ and U are elements of $\mathbb{A}[z]$, such that, for any real solution t_0 of $w(t)$, the finite set defined by

$$\{(V_1(t_0, z), \ldots, V_n(t_0, z)) \mid z \in \mathbb{R}, U(t_0, z) = 0\}$$

intersects every connected component of

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} a_i (x_i - v_i(t_0))^2 - e_0 = 0.$$

Hence, the isolated points of $\mathcal{H} \cap \mathbb{R}^n$ are indeed

$$\{(v_1(t), \ldots, v_n(t)) \mid (t, z) \in \mathbb{R}^2 : \ w(t) = U(t, z) = 0\}.$$

Our problem boils down to solving the bivariate system $w(t) = U(t, z) = 0$ over $\mathbb{R}^2$.

In Algorithm 3 below, we introduce two subroutines:

- BivariatePolynomial takes as input the polynomials F and $w(t)$ and returns the eliminating polynomial $U(t, z)$. It uses the geometric resolution algorithm over $\mathbb{A}$ described in [25, Appendix J].
- BivariateSolve takes as input $w(t)$ and $U(t, z)$ and returns the set $\mathscr{B}$ of intervals that isolate the real roots of $w(t)$ corresponding to $\mathscr{I}(\mathcal{H})$. Such a subroutine can be designed efficiently with resultants.

Algorithm 3: The first variant of IsIsolated

Input: f, $\mathscr{C} = (w(t), v_1(t), \ldots, v_n(t))$, $\boldsymbol{a}$ and e_0
Output: A set $\mathscr{B}$ of isolating intervals

1 $F \leftarrow f(x_1, \ldots, x_n)^2 + \left(\sum_{i=1}^{n} a_i (x_i - v_i(t))^2 - e_0\right)^2$
2 $U(t, z) \leftarrow$ BivariatePolynomial$(F, w(t))$
3 $\mathscr{B} \leftarrow$ BivariateSolve$(w(t), U(t, z))$
4 **return** $\mathscr{B}$

4.4. Approximations of the candidates

This subsection describes the design of the second variant of IsIsolated. This variant does not require solving polynomial systems in the quotient ring $\mathbb{Q}[t]/\langle w(t)\rangle$ but is based mostly on isolating the candidates from the parametrization $\mathscr{C}$. The main idea is to replace the candidate $\boldsymbol{\eta}$ in the criteria provided by Proposition 7 by a rational approximation $\tilde{\boldsymbol{\eta}} \in \mathbb{Q}^n$. This allows the subroutine presented in Algorithm 4 to involve only the real points defining by $\mathscr{C}$ and not its whole complex solution set.

To compute the approximations, we first identify how close the points $\boldsymbol{\eta}$ and $\tilde{\boldsymbol{\eta}}$ need to be. The lemma below shows that requiring $d_{\boldsymbol{a}}(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}) < \sqrt{e_0}/2$ is enough.

Lemma 10. *Let $\boldsymbol{\eta} \in \mathfrak{C} \cap \mathbb{R}^n$ and $\tilde{\boldsymbol{\eta}}$ be a point in $\mathbb{R}^n$ satisfying $d_{\boldsymbol{a}}(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}) < \sqrt{e_0}/2$. Then, $\boldsymbol{\eta}$ is an isolated point of $\mathcal{H}$ if and only if $\mathcal{H}$ does not intersect the sphere $S(\tilde{\boldsymbol{\eta}}, \sqrt{e_0}/2)$.*

Proof. If the set $\{\boldsymbol{x}_\varepsilon \in \mathrm{crit}(\delta_{\boldsymbol{\eta}}, \mathcal{V}_\varepsilon) \cap \mathbb{R}\langle\varepsilon\rangle_b^n,\ \lim_\varepsilon \boldsymbol{x}_\varepsilon \neq \boldsymbol{\eta}\}$ is empty, then, by Lemma 6, $\mathcal{H}$ is either a single point $\boldsymbol{\eta}$ or an unbounded connected set containing $\boldsymbol{\eta}$. In either case, the conclusion of Lemma 10 is immediate. Thus, in what follows, $\{\boldsymbol{x}_\varepsilon \in \mathrm{crit}(\delta_{\boldsymbol{\eta}}, \mathcal{V}_\varepsilon) \cap \mathbb{R}\langle\varepsilon\rangle_b^n,\ \lim_\varepsilon \boldsymbol{x}_\varepsilon \neq \boldsymbol{\eta}\}$ is assumed to be non-empty.

We prove now the necessary implication. Assume that $\boldsymbol{\eta}$ is an isolated point of $\mathcal{H}$. By Lemma 5, the intersection of $\mathcal{H}$ and $S(\boldsymbol{\eta}, \sqrt{e})$ is empty for every $e \in]0, e_0[$. So, $\boldsymbol{\eta}$ is the only point of $\mathcal{H}$ lying in the open ball $B(\boldsymbol{\eta}, \sqrt{e_0})$. Since $d_{\boldsymbol{a}}(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}) < \sqrt{e_0}/2$, the candidate $\boldsymbol{\eta}$ does not lie on the sphere $S(\tilde{\boldsymbol{\eta}}, \sqrt{e_0}/2)$. Moreover, $S(\tilde{\boldsymbol{\eta}}, \sqrt{e_0}/2)$ is contained in the open ball $B(\boldsymbol{\eta}, \sqrt{e_0})$. Then, $S(\tilde{\boldsymbol{\eta}}, \sqrt{e_0}/2) \cap \mathcal{H} = \emptyset$.

Now we turn to the sufficient implication. Assume by contradiction that $\boldsymbol{\eta}$ is not isolated in $\mathcal{H}$. By Lemma 5, the connected component $C_{\boldsymbol{\eta}}$ of $\mathcal{H}$ containing $\boldsymbol{\eta}$ intersects the sphere $S(\boldsymbol{\eta}, \sqrt{e_0})$. So, there exists a semi-algebraic continuous function $\gamma : [0, 1] \to C_{\boldsymbol{\eta}}$ such that $\gamma(0) = \boldsymbol{\eta}$ and $\gamma(1)$ lying on the sphere $S(\boldsymbol{\eta}, \sqrt{e_0})$. We have that

$$d_{\boldsymbol{a}}(\gamma(1), \tilde{\boldsymbol{\eta}}) \geq d_{\boldsymbol{a}}(\gamma(1), \boldsymbol{\eta}) - d_{\boldsymbol{a}}(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}) > \sqrt{e_0} - \sqrt{e_0}/2 = \sqrt{e_0}/2.$$

As $d_{\boldsymbol{a}}(\gamma(0), \tilde{\boldsymbol{\eta}}) < \sqrt{e_0}/2$ and $d_{\boldsymbol{a}}(\gamma(1), \tilde{\boldsymbol{\eta}}) > \sqrt{e_0}/2$, by the intermediate value property [2, Proposition 3.5], there exists $t_0 \in]0, 1[$ such that $d_{\boldsymbol{a}}(\gamma(t_0), \tilde{\boldsymbol{\eta}}) = \sqrt{e_0}/2$. This implies that the intersection of $\mathcal{H}$ and $S(\tilde{\boldsymbol{\eta}}, \sqrt{e_0}/2)$ is not empty, which concludes our proof. □

Let $t_{\boldsymbol{\eta}}$ be the real root of $w(t)$ corresponding to $\boldsymbol{\eta}$, i.e., $\boldsymbol{\eta} = \big(v_1(t_{\boldsymbol{\eta}}), \ldots, v_n(t_{\boldsymbol{\eta}})\big)$. To apply Lemma 10, we need to choose $t_{\tilde{\boldsymbol{\eta}}} \in \mathbb{Q}$ such that the rational point $\tilde{\boldsymbol{\eta}} = (v_1(t_{\tilde{\boldsymbol{\eta}}}), \ldots, v_n(t_{\tilde{\boldsymbol{\eta}}}))$ satisfies that $d_{\boldsymbol{a}}(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}) < \sqrt{e_0}/2$. This leads us to identify $\rho > 0$ such that $|t_{\boldsymbol{\eta}} - t_{\tilde{\boldsymbol{\eta}}}| < \rho$ implies

$$a_1(v_1(t_{\boldsymbol{\eta}}) - v_1(t_{\tilde{\boldsymbol{\eta}}}))^2 + \ldots + a_n(v_n(t_{\boldsymbol{\eta}}) - v_n(t_{\tilde{\boldsymbol{\eta}}}))^2 < e_0/4.$$

Lemma 11 below allows us to compute explicitly an appropriate value for ρ.

Lemma 11. *Let $\{t_1, \ldots, t_\ell\}$ be the distinct real roots of $w(t) = 0$ and $\{\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_\ell\}$ be the corresponding candidates. We consider a set of intervals $(I_j)_{1 \leq j \leq \ell}$ such that*

- *The intervals I_j are pairwise disjoint.*
- *The interval I_j contains only t_j as a real root of $w(t)$.*

For each $1 \leq i \leq n$, let $K_i = \max_{j=1}^{\ell} \max_{t \in I_j} |v_i'(t)|$. Then, for any $1 \leq j \leq \ell$ and $t_{\boldsymbol{\theta}}$ such that $t_{\boldsymbol{\theta}} \in I_j$ and $|t_{\boldsymbol{\theta}} - t_j| < \frac{1}{K_i} \cdot \sqrt{\frac{e_0}{4na_i}}$, we have the following inequality:

$$|v_i(t_{\boldsymbol{\theta}}) - v_i(t_j)| < \sqrt{\frac{e_0}{4na_i}}.$$

Let $\rho \leq \min_{i=1}^{n} \frac{1}{K_i} \cdot \sqrt{\frac{e_0}{4na_i}}$. For any real root $t_{\boldsymbol{\eta}}$ of $w(t)$ and $t_{\boldsymbol{\theta}} \in I_j$, $|t_{\boldsymbol{\theta}} - t_{\boldsymbol{\eta}}| < \rho$ implies

$$d_{\boldsymbol{a}}(\boldsymbol{\theta}, \boldsymbol{\eta}) < \sqrt{e_0}/2.$$

Proof. For $1 \leq j \leq \ell$ and any $t_{\boldsymbol{\theta}} \in \mathbb{Q}$, we have that

$$v_i(t_{\boldsymbol{\theta}}) - v_i(t_j) = v_i'(\tilde{t}_j)(t_{\boldsymbol{\theta}} - t_j),$$

where $\tilde{t} \in \mathbb{R}$ lies between $t_{\boldsymbol{\theta}}$ and t_j.

For $t \in I_j =]r_j, s_j[$, by the definition of K_i, we have $|v_i'(t)| \leq K_i$. Then, for $t_{\boldsymbol{\theta}} \in I_j$ such that $|t_{\boldsymbol{\theta}} - t_j| < \frac{1}{K_i} \cdot \sqrt{\frac{e_0}{4na_i}}$, we have

$$|v_i(t_{\boldsymbol{\theta}}) - v_i(t_j)| = |v_i'(\tilde{t}_j) \cdot (t_{\boldsymbol{\theta}} - t_j)| \leq K_i \cdot |t_{\boldsymbol{\theta}} - t_j| < \sqrt{\frac{e_0}{4na_i}}.$$

Now we take $\rho \leq \min_{i=1}^{n} \frac{1}{K_i} \cdot \sqrt{\frac{e_0}{4na_i}}$. If $t_{\boldsymbol{\theta}} \in I_j$ and $|t_{\boldsymbol{\theta}} - t_j| < \rho$, then we have

$$d_{\boldsymbol{a}}(\boldsymbol{\theta}, \boldsymbol{\eta}_j) = \sqrt{\sum_{i=1}^{n} a_i (v_i(t_{\boldsymbol{\theta}}) - v_i(t_j))^2} < \sqrt{\sum_{i=1}^{n} \frac{e_0}{4n}} = \frac{\sqrt{e_0}}{2}.$$ □

Lemmas 10 and 11 provides us the ingredients to design Algorithm 4. It requires us to introduce two subroutines Isolate and MaxOverInterval below.

- We need two versions of Isolate. The first one takes as input a polynomial $p \in \mathbb{Q}[t]$ and returns a set of disjoint intervals of rational extremities isolating the real roots of p.

  Besides the polynomial $p \in \mathbb{Q}[t]$, the second version of Isolate requires a positive $\rho \in \mathbb{Q}$ as input and returns the intervals of length at most ρ that isolate the real roots of p.

  The explicit descriptions of both of these real root isolating algorithms are given in [21].

- MaxOverInterval takes as input a polynomial $p \in \mathbb{Q}[t]$ and an interval $[r, s]$ where $r, s \in \mathbb{Q}$ and returns an upper bound of $\max_{t \in [r,s]} |p(t)|$. Such a subroutine can be implemented using the following naive bound:

  $$\max_{t \in [r,s]} |p(t)| \leq \sum_{i=0}^{\deg(p)} |c_i|$$

  where $p\left(\frac{t-r}{s-r}\right) = c_0 \cdot t^{\deg(p)} + \cdots + c_{\deg(p)}$.

Algorithm 4 proceeds through these following steps:

a) We call Isolate on the input $w(t)$ to obtain a set of intervals I_j that isolate the real roots of $w(t)$ and compute $K_i = \max_{j=1}^{\ell} \max_{t \in I_j} |v_i'(t)|$ using the subroutine MaxOverInterval on the input $v_i'(t)$ and each interval I_j.

b) We then compute $\rho \in \mathbb{Q}$ such that $0 < \rho \leq \min_{i=1}^{n} \frac{1}{K_i} \cdot \sqrt{\frac{e_0}{4na_i}}$ and use Isolate on the polynomial $w(t)$ and the precision ρ to obtain a set of intervals $\tilde{I}_j$ such that each $\tilde{I}_j$ contains exactly one real root of $w(t)$ and $|\tilde{I}_j| < \rho$.

c) For $1 \leq j \leq \ell$, we choose a point $\tilde{t}_j$ in $I_j \cap \tilde{I}_j \cap \mathbb{Q}$ and evaluate $\tilde{\boldsymbol{\eta}}_j = (v_1(\tilde{t}_j), \ldots, v_n(\tilde{t}_j))$.

   The set $\tilde{\mathfrak{C}}$ of the approximations is taken as $\{(\tilde{\boldsymbol{\eta}}_j, I_j \mid 1 \leq j \leq \ell\}$.

d) Finally, we decide whether the system

$$f(x_1,\dots,x_n)=\sum_{i=1}^{n} a_i(x_i-\tilde{\eta}_i)^2-e_0/4=0$$

has a real solution for each approximation $\tilde{\boldsymbol{\eta}}$ and return those which do not.

We summarize Section 4 in Algorithm 4 below, which is our second variant of IsIsolated.

Algorithm 4: Algorithm IsIsolated-Approx

Input: $f, \mathscr{C}=(w(t),v_1(t),\dots,v_n(t)), \boldsymbol{a}\in\mathbb{Q}_+^n$ and $e_0\in\mathbb{Q}$
Output: A set of isolating interval $\mathscr{B}$

$\{I_1,\dots,I_\ell\}\leftarrow \mathsf{Isolate}(w(t))$
for $i\in\{1,\dots,n\}$ **do**
  $K_i\leftarrow \max_{j=1}^{\ell}\mathsf{MaxOverInterval}(v_i'(t),I_j)$
$\{\tilde{I}_1,\dots,\tilde{I}_\ell\}\leftarrow \mathsf{Isolate}\left(w(t),\rho=\min_{i=1}^{n}\frac{1}{K_i}\cdot\sqrt{\frac{e_0}{4na_i}}\right)$
for $j\in\{1,\dots,\ell\}$ **do**
  $\tilde{t}_j\in I_j\cap\tilde{I}_j$
  $\tilde{\boldsymbol{\eta}}_j\leftarrow(v_1(\tilde{t}_j),\dots,v_n(\tilde{t}_j))$
$\tilde{\mathfrak{C}}\leftarrow\{(\tilde{\boldsymbol{\eta}}_j,I_j)\mid 1\le j\le\ell\}, \mathscr{B}\leftarrow\emptyset$
for $(\tilde{\boldsymbol{\eta}},I_{\boldsymbol{\eta}})\in\tilde{\mathfrak{C}}$ **do**
  if $\mathsf{HasRealSolutions}(\tilde{\boldsymbol{\eta}},f,\boldsymbol{a},e_0)=$ `false` **then**
    $\mathscr{B}\leftarrow\mathscr{B}\cup I_{\boldsymbol{\eta}}$
return $\mathscr{B}$

Remark 12. *Note that the sphere in Lemma 10 can be replaced by any hypercube that contains the candidate $\boldsymbol{\eta}$ in its interior and is contained in the ball $B(\boldsymbol{\eta},e_0)$. This leads one to decide the emptiness of semi-algebraic sets defined by $f=0$ and some linear polynomial inequalities instead of the quadric defining the sphere. We observe in practice that this helps reduce the bit-size growth and accelerates the computation.*

5. Complexity analysis

We establish now the complexity results for two variants of Algorithm 1, starting with the one using Algorithm 3 for IsIsolated.

Theorem I. *Let $f\in\mathbb{Q}[x_1,\dots,x_n]$. Then, the variant of Algorithm 1 which uses Algorithm 3 computes the real isolated points of the algebraic hypersurface defined by f within $O^{\sim}\left(64^n D^{8n}\right)$ arithmetic operations in $\mathbb{Q}$ and one call of real root isolation on a univariate polynomial of degree bounded by $2^{n+2}D^{2n}$.*

Proof. We start with the subroutine Candidates. Since $\mathrm{crit}(\pi_i,\mathcal{H}_\varepsilon^A)$ is finite and defined by

$$(f^A-\varepsilon)\cdot(f^A+\varepsilon)=0,\quad \frac{\partial f^A}{\partial x_j}=0 \text{ for all } j\neq i,$$

its degree is bounded by $2D(D-1)^{n-1}$ using Bézout bound. Consequently, the degree of the output zero-dimensional parametrization is bounded by $2D(D-1)^{n-1}$.

Using [22, Theorem 4] (which is based on the geometric resolution algorithm in [14]), each zero-dimensional parametrization of $\mathrm{crit}(\pi_i, \mathcal{H}_\varepsilon^A)$ is computed within $O^\sim\left(D^{3n}\right)$ arithmetic operations in $\mathbb{Q}$. The last step which takes intersections of those parametrizations is done using the algorithm in [25, Appendix J.1] ; it does not change the asymptotic complexity.

Hence, computing the parametrization $\mathscr{C}$ encoding the candidates can be done within $O^\sim\left(D^{3n}\right)$ arithmetic operations in $\mathbb{Q}$ and the degrees of the polynomials $w(t), v_1(t), \ldots, v_n(t)$ are bounded by $2D(D-1)^{n-1}$. It remains to estimate the arithmetic complexity of the subroutines ComputeE0 and IsIsolated.

Let κ be the degree of $w(t)$. Algorithm 2 (ComputeE0) relies on computing the limit of $\mathrm{crit}(\delta, \mathcal{H}_{\varepsilon,t}) \cap \mathbb{C}\langle\varepsilon\rangle_b^n$, where $\mathcal{H}_{\varepsilon,t}$ is the algebraic set defined by

$$(f-\varepsilon)\cdot(f+\varepsilon) = 0, \quad w(t) = 0 \tag{6}$$

and δ is the distance function

$$(x_1, \ldots, x_n) \mapsto \sum_{i=1}^{n} a_i (x_i - v_i(t))^2$$

We use the algorithm of [22] on the function δ for the resolution of polynomial systems in the quotient ring $\mathbb{Q}[t]/\langle w(t)\rangle$. Using Bézout's bound on the system

$$f^2 - \varepsilon^2 = y\cdot\frac{\partial f}{\partial x_1} - \frac{\partial \delta}{\partial x_1} = \cdots = y\cdot\frac{\partial f}{\partial x_n} - \frac{\partial \delta}{\partial x_n} = 0$$

defining $\mathrm{crit}(\delta, \mathcal{H}_{\varepsilon,t})$ over $\mathbb{A}$, the degree of $\mathrm{crit}(\delta, \mathcal{H}_{\varepsilon,t})$ in $\mathbb{C}\langle\varepsilon\rangle^n$ is bounded by $2D^{n+1}\kappa \le 4D^{n+2}(D-1)^{n-1} \approx 4D^{2n+1}$.

By [25, Appendix J.5], the arithmetic operations over $\mathbb{A}$ can be done using $O^\sim(\kappa)$ operations in $\mathbb{Q}$. Thus, applying [22, Theorem 5], we obtain the complexity bound $O^\sim\left(\kappa\cdot D^{3n+2}\right) \approx O^\sim\left(D^{4n+2}\right)$ for obtaining the zero-dimensional parametrization $\mathscr{E}$ in Algorithm 2.

The call to GetE0 computes from the zero-dimensional parametrization $\mathscr{E}$ a univariate polynomial $P(e) \in \mathbb{Q}[e]$ whose solutions are the critical values of δ restricted to $\mathcal{V}_\varepsilon$. Since the degree of $P(e)$ is bounded by $4D^{2n+1}$, this can be done using FGLM algorithm [12] within $O^\sim\left(D^{6n+3}\right)$ arithmetic operations over $\mathbb{Q}$. Next, it computes the minimal distance between the real roots of $P(e)$ using [2, Proposition 10.23]. The complexity of this computation is linear in the degree of $P(E)$. Thus, it does not change the asymptotic complexity of Algorithm 2.

Therefore, Algorithm 2 can be done within $O^\sim(D^{6n+3})$ arithmetic operations in $\mathbb{Q}$.

Algorithm 3 is basically computing sample points of the hypersurface

$$F = f(x_1, \ldots, x_n)^2 + \left(\sum_{i=1}^{n} a_i (x_i - v_i(t))^2 - e_0\right)^2$$

over the quotient ring $\mathbb{Q}[t]/\langle w(t)\rangle$. Again, we follow the algorithm of [22] on the input F with the extended version of geometric resolution to the quotient ring $\mathbb{A}$. By [22, Theorem 6] with the overcost $O^\sim(\kappa)$ of arithmetic operations over $\mathbb{A}$, we obtain the complexity bound

$$O^\sim\left(\kappa\cdot(2D)^{3n+2}\right) \approx O^\sim\left(8^n D^{4n+2}\right)$$

for MinimalPolynomial.

The output polynomial $U(t,z)$ has degree at most $(2D)^n$ in z and κ in t so its total degree is bounded by $(2D)^n + \kappa$. Therefore, solving the bivariate system

$$w(t) = U(t,z) = 0$$

can be done within

$$O^{\sim}\left(((2D)^n+\kappa)^4\,\kappa^2\,((2D)^n+\kappa)^2\right) \approx O^{\sim}\left(64^n D^{8n}\right)$$

arithmetic operations in $\mathbb{Q}$ using geometric resolution. In the end, one needs to isolate the real roots of the eliminating polynomial output by the geometric resolution. That polynomial has degree bounded by $\kappa((2D)^n+\kappa) \leq 2^{n+2}D^{2n}$.

Adding up all the steps, we obtain the arithmetic complexity of Algorithm 1, which lies in $O^{\sim}\left(64^n D^{8n}\right)$ with a call to real root isolation on a polynomial of degree bounded by $2^{n+2}D^{2n}$. □

Note that for implementing our algorithm, we would mainly rely on Algorithm 4 (IsIsolated-Approx). Hence, we dedicate the rest of this subsection to discuss its complexity. The complexity result of our algorithm using Algorithm 4 is stated as follows.

Theorem II. *Let $f \in \mathbb{Q}[x_1,\ldots,x_n]$. Then, the variant of Algorithm 1 which uses Algorithm 4 requires $O^{\sim}\left(D^{6n+3}\right)$ arithmetic operations in $\mathbb{Q}$ and two real root isolating calls on a univariate polynomial of degree bounded by $2D(D-1)^{n-1}$.*

Proof. Recall that Algorithm 4 computes an approximation $\tilde{\boldsymbol{\eta}} = (\tilde{\eta}_1,\ldots,\tilde{\eta}_n)$ for each candidate $\boldsymbol{\eta} \in \mathfrak{C} \cap \mathbb{R}^n$ and decides whether the system

$$f(x_1,\ldots,x_n) = \sum_{i=1}^{n} a_i(x_i-\tilde{\eta}_i)^2 - \frac{e_0}{4} = 0$$

has a real solution. The arithmetic complexity for solving each of those decision problems lies in $O^{\sim}\left(8^n D^{3n+2}\right)$ using [22]. Since the cardinality of $\mathfrak{C}$ is bounded by $2D(D-1)^{n-1}$, Algorithm 4 runs within

$$O^{\sim}\left(8^n D^{4n+2}\right)$$

arithmetic operations in $\mathbb{Q}$.

Note that all the complexities above are dominated by the complexity

$$O^{\sim}\left(D^{6n+3}\right)$$

of ComputeE0 (Algorithm 2).

It remains to estimate the complexity of computing the approximations, whose main steps consist of calling MaxOverInterval and isolating the real roots of the eliminating polynomial $w(t)$ in the zero-dimensional parametrization encoding $\mathfrak{C}$.

The subroutine MaxOverInterval is called n times for the polynomials $v_i'(t)$; this would require $O^{\sim}(\deg(w)) \approx O^{\sim}(D^n)$ arithmetic operations over $\mathbb{Q}$.

Since each of ℓ evaluations $\tilde{\boldsymbol{\eta}}_j \leftarrow \left(v_1(\tilde{t}_j),\ldots,v_n(\tilde{t}_j)\right)$ takes $O(nD^n)$ arithmetic operations, the cost of getting the approximations is bounded by

$$O\left(nD^{2n}\right).$$

The real root isolation is called twice in Algorithm 4 on the polynomial $w(t)$.

Summing up the above discussion, we conclude that Algorithm 4 requires

$$O^{\sim}\left(8^n D^{4n+2}\right)$$

arithmetic operations in $\mathbb{Q}$ and two calls of real root isolation on a univariate polynomial of degree bounded by $2D(D-1)^{n-1}$. □

Furthermore, the complexity of real root isolation algorithms depends on the degree of the input polynomial, which is $w(t)$ in our case, and its bit-size of coefficients. For instance, using the algorithm of [26] leads to a bit complexity

$$O^{\sim}\left(\deg(w)^3 \tau^2\right)$$

where τ is the largest bit-size of coefficients of $w(t)$. While the degree of $w(t)$ is already bounded by $2D(D-1)^{n-1}$, τ is not estimated yet in this thesis. To identify a bound of τ, one needs to estimate the bit complexity of the subroutine Candidates [17, Subsection 3.2], especially the algorithm for computing at least one point per connected component of a real algebraic set given in [22]. This topic will be studied in future research.

6. Heuristic optimizations

Even though computing the constant e_0 requires at most $D^{O(n)}$ arithmetic operations in $\mathbb{Q}$, its performance depends heavily on an efficient implementation of the geometric resolution algorithm over $\mathbb{Q}[t]/\langle w(t)\rangle$, which remains challenging to obtain. Thus, we aim to avoid such computations as much as possible. In what follows, we present two subroutines which are launched to test whether it is necessary for computing e_0. In most of the case, with theses subroutines, our algorithm will return the set of isolated points without doing any further computation.

6.1. Heuristic identification of real isolated points

The optimization described in what follows identifies efficiently a subset of isolated points of $\mathcal{H}$ from the candidates without computing e_0. The idea is to compute for each candidate $\boldsymbol{x}$ a ball $B \in \mathbb{R}^n$ such that if the boundary of B intersects $\mathcal{H}$, $\boldsymbol{x}$ is isolated in $\mathcal{H}$.

We consider the polynomial f^A obtained from f by a linear change of variable A and denote $\mathcal{V}^A = V(f^A)$ and $\mathcal{H}^A = \mathcal{V}^A \cap \mathbb{R}^n$. Similarly, $\mathcal{V}^A_\varepsilon$ and $\mathcal{H}^A_\varepsilon$ correspond to $(f^A)^2 = \varepsilon^2$.

Lemma 13. *Let C be a bounded connected component of $\mathcal{H}^A$ and $1 \leq j \leq n$ such that $\pi_j(C)$ is not a single point. Then, there exist at least two points in $\lim_\varepsilon \operatorname{crit}(\pi_j, \mathcal{V}^A_\varepsilon) \cap \mathbb{R}^n$ contained in C.*

Proof. Let $C_1, \ldots, C_k$ be the connected components of $\mathcal{H}_\varepsilon \subset \mathbb{R}\langle\varepsilon\rangle^n$ such that $\lim_\varepsilon C_i \subset C$. By Lemma 2, since C is bounded, the C_i's are bounded over $\mathbb{R}$. Then, by [2, Proposition 12.49], $\lim_\varepsilon C_i$ is connected. As $\pi_j(C)$ is not a single point, it must be an interval $[a, b]$ where $a \neq b$.

On the other hand, $\cup_{i=1}^k \lim_\varepsilon C_i = C$. Hence,

$$\pi_j(C) = \cup_{i=1}^k \pi_j(\lim_\varepsilon C_i)$$

and the projections $\pi_j(\lim_\varepsilon C_i)$ are connected. Thus, there exists $1 \leq i \leq k$ such that $\pi_j(\lim_\varepsilon C_i)$ is not a singleton. Hence, the projection of C_i by π_j is a closed interval $[\alpha, \beta] \subset \mathbb{R}\langle\varepsilon\rangle$ where

$\lim_\varepsilon \alpha \neq \lim_\varepsilon \beta$. Then, there exist two points $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}_\beta$ in $\mathrm{crit}(\pi_j, \mathcal{V}_\varepsilon) \cap C_i$ such that $\pi_j(\boldsymbol{x}_\alpha) = \alpha$ and $\pi_j(\boldsymbol{x}_\beta) = \beta$. Since $\pi_j(\lim_\varepsilon \boldsymbol{x}_\alpha) \neq \pi_j(\lim_\varepsilon \boldsymbol{x}_\beta)$, $\lim_\varepsilon \boldsymbol{x}_\alpha \neq \lim_\varepsilon \boldsymbol{x}_\beta$. Then $\lim_\varepsilon \boldsymbol{x}_\alpha$ and $\lim_\varepsilon \boldsymbol{x}_\beta$ are two distinct points of $C \cap \lim_\varepsilon \mathrm{crit}(\pi_j, \mathcal{V}_\varepsilon^A)$. Consequently, there exists two distinct points in $C \cap \lim_\varepsilon \mathrm{crit}(\pi_j, \mathcal{V}_\varepsilon^A)$. □

Lemma 14. *For every bounded connected component C of $\mathcal{H}^A$ that is not a singleton, there exist at least two points in $\bigcup_{i=1}^n \lim_\varepsilon \mathrm{crit}(\pi_i, \mathcal{V}_\varepsilon^A)$ that belong to C.*

Proof. Since C is not a singleton, there exists a coordinate x_j such that the projection $\pi_j(C)$ is not a point. Using Lemma 13, we conclude the proof. □

Let $\mathfrak{C}_2$ be the image of

$$\bigcup_{i=1}^{n} \lim_{\varepsilon} \mathrm{crit}(\pi_i, \mathcal{V}_\varepsilon^A)$$

by the linear map induced by A^{-1}. Note that the set of candidates $\mathfrak{C}$ is a subset of $\mathfrak{C}_2$.

Proposition 15. *Let $\mathfrak{C}_2$ be defined as above. Let $\boldsymbol{x} \in \mathfrak{C} \cap \mathbb{R}^n$ and $B \subset \mathbb{R}^n$ be a ball such that $\mathfrak{C}_2 \cap B = \{\boldsymbol{x}\}$ and $\boldsymbol{x}$ is contained in the interior of B. Then, if the intersection of the boundary of B and $\mathcal{H}$ is empty, $\boldsymbol{x}$ is an isolated point of $\mathcal{H}$.*

Proof. Let C be the connected component of $\mathcal{H}$ containing $\boldsymbol{x}$. If C is unbounded, then $\boldsymbol{x}$ is not an isolated point and the intersection of the boundary B with $\mathcal{H}$ is not empty.

We now assume that C is bounded and is not a singleton. By Lemma 14, there exists $\boldsymbol{y} \in \mathfrak{C}_2 \cap \mathbb{R}^n$ such that $\boldsymbol{y} \neq \boldsymbol{x}$ and $\boldsymbol{y} \in C$. As $\boldsymbol{x}$ and $\boldsymbol{y}$ lie on different sides of B, by intermediate value theorem, $\mathcal{H}$ intersects the boundary of B, which ends the proof. □

Since the subroutine Candidates in [17, Subsection 3.2] computes all the zero-dimensional parametrizations encoding $\lim_\varepsilon \mathrm{crit}(\pi_i, \mathcal{V}_\varepsilon^A)$, the union $\mathfrak{C}_2$ can be obtained easily by taking the union of those zero-dimensional parametrizations. Next, we isolate the candidates in $\mathscr{C} \cap \mathbb{R}^n$ such that each isolating ball contains exactly on point of $\mathfrak{C}_2 \cap \mathbb{R}^n$.

Note also that for a generic matrix $A \in \mathrm{GL}(n, \mathbb{Q})$, the projection of any non-singleton bounded component of $\mathcal{H}^A$ by π_1 is not a singleton. In this case, by Lemma 13, it is enough to take $\mathfrak{C}_2 = \lim_\varepsilon \mathrm{crit}(\pi_1, \mathcal{V}_\varepsilon^A)$. This can accelerate implementations with a small cost of randomness (which is already a part of the subroutine Candidates).

Algorithm 5 contains the description of the subroutine SimpleIdentification. We call to a subroutine BoxIsolate that takes as input a zero-dimensional parametrization encoding a subset of $\mathbb{C}^n$ and computes isolating boxes for its real zeros.

Algorithm 5: SimpleIdentification

Input: A zero-dimensional parametrization $\mathscr{C}_2$
Output: A set $\mathscr{B}_1$ of intervals of $\mathbb{R}$

```
1 B_1 ← ∅
2 Boxes ← BoxIsolate(C_2)
3 for box ∈ Boxes do
4     if box ∩ H = ∅ then
5         B_1 ← B_1 ∪ {t-coordinate of box}
6 return B_1
```

By Proposition 15, for each $\boldsymbol{x} \in \mathfrak{C}$ such that the intersection of the ball isolating $\boldsymbol{x}$ with $\mathcal{H}$ is empty, we conclude that $\boldsymbol{x}$ is an isolated point of $\mathcal{H}$. For the non-empty intersections, we cannot decide whether $\boldsymbol{x}$ is isolated yet. The problem arises when the isolating boxes are not small enough so that they intersect not only the connected component of $\mathcal{H}$ containing $\boldsymbol{x}$ but also some other connected component. When this happens, one could try a smaller size of isolating boxes.

Remark 16. *When every candidate is an isolated point, running* SimpleIdentification *is enough to return a correct and certified output, which is the whole set of candidates. No further computation is required in this case.*

As each candidate is contained in $\lim_\varepsilon \operatorname{crit}(\pi_i, \mathcal{V}_\varepsilon)$ *for every projection* π_i*, we conjecture that every candidate is actually isolated. We are not aware of any counter-example.*

6.2. Limits of critical curves

To compute a set of candidates, we consider the critical loci $\operatorname{crit}(\pi_i, \mathcal{V}_\varepsilon)$ for $1 \le i \le n$. Our second optimization considers the critical loci of the projections on the plane; especially, the limits of those critical loci are curves in $\mathbb{R}^n$ whose real isolated points contain the isolated points of $\mathcal{H}$. Thus, one can compute a superset of $\mathscr{I}(\mathcal{H})$ through computing the real isolated points of limits of critical curves.

More precisely, for $1 \le i < j \le n$, we denote by $\pi_{i,j}$ the projection

$$\pi_{i,j} : (x_1, \ldots, x_n) \mapsto (x_i, x_j).$$

Recall that $\mathcal{H}_\varepsilon$ is a smooth algebraic set defined by

$$(f - \varepsilon) \cdot (f + \varepsilon) = 0.$$

Lemma 17. *Let* $A \in \mathrm{GL}(n, \mathbb{Q})$*. For every* $1 \le i < j \le n$*, the set of isolated points of* $\mathcal{H}^A$ *is contained in set of real isolated points of* $\lim_\varepsilon \operatorname{crit}(\pi_{i,j}, \mathcal{H}_\varepsilon^A)$*.*

Proof. Let $\boldsymbol{x}$ be an isolated point of $\mathcal{H}^A$. By Proposition 3, $\boldsymbol{x} \in \lim_\varepsilon \operatorname{crit}(\pi_i, \mathcal{V}_\varepsilon^A)$. Since $\operatorname{crit}(\pi_i, \mathcal{V}_\varepsilon^A) \subset \operatorname{crit}(\pi_{i,j}, \mathcal{V}_\varepsilon^A)$, $\boldsymbol{x} \in \lim_\varepsilon \operatorname{crit}(\pi_{i,j}, \mathcal{V}_\varepsilon^A)$. Note that $\lim_\varepsilon \operatorname{crit}(\pi_{i,j}, \mathcal{V}_\varepsilon^A)$ is a subset of $\mathcal{V}^A$. Thus, if $\boldsymbol{x}$ is isolated in $\mathcal{H}^A$, it is also an isolated point in $\lim_\varepsilon \operatorname{crit}(\pi_{i,j}, \mathcal{V}_\varepsilon^A) \cap \mathbb{R}^n$. □

Remark 18. *Note that a real isolated point of* $\lim_\varepsilon \operatorname{crit}(\pi_{i,j}, \mathcal{V}_\varepsilon)$ *is not necessarily isolated in* $\mathcal{H}$*. Take for example the degenerate torus, given by the equation*

$$(x_1^2 + x_2^2 + x_3)^2 - 4(x_1^2 + x_2^2) = 0.$$

The real trace of $\lim_\varepsilon \operatorname{crit}(\pi_{1,2}, \mathcal{V}_\varepsilon)$ *is the union of the point* $(0, 0)$ *and the circle given by*

$$x_1^2 + x_2^2 - 4 = x_3 = 0.$$

Hence, Lemma 17 allows us to obtain a superset of $\mathscr{I}(\mathcal{H})$ *only.*

By [24, Theorem 2], for a generic change of variables A, the critical locus $\operatorname{crit}(\pi_{i,j}, \mathcal{H}_\varepsilon^A)$ is an equidimensional algebraic set of dimension one defined by

$$(f - \varepsilon) \cdot (f + \varepsilon) = 0, \quad \frac{\partial f}{\partial x_k} = 0 \quad \text{for } 1 \le k \le n \text{ and } k \ne i, j.$$

The computation of $\lim_\varepsilon \operatorname{crit}(\pi_{i,j}, \mathcal{H}_\varepsilon^A)$ can be done using a similar subroutine of [22]. For each $1 \le i, j \le n$, we denote by $J_{i,j}$ the ideal

$$\left\langle \frac{\partial f}{\partial x_k} = 0 \text{ for } 1 \le k \le n, k \ne i, j \right\rangle.$$

Lemma 19. *Let $\pi_{i,j}$ be defined as above. There exists a non-empty Zariski open subset $\mathcal{A}$ of* $\mathrm{GL}(n,\mathbb{C})$ *such that, for any $A \in \mathcal{A} \cap \mathrm{GL}(n,\mathbb{Q})$, the algebraic set C defined by*

$$V\left(\left\langle f^A \right\rangle + J_k : \left(\frac{\partial f^A}{\partial x_i}\right)^\infty \cap J_k : \left(\frac{\partial f^A}{\partial x_j}\right)^\infty\right)$$

is equi-dimensional of dimension 1 *and contains* $\lim_\varepsilon \mathrm{crit}(\pi_{i,j}, \mathcal{H}^A_\varepsilon)$.

As a consequence, any isolated point of $\mathcal{H}^A$ is also isolated in $C \cap \mathbb{R}^n$.

Proof. The proof of the first statement follows a similar outline of the proof of [22, Theorem 1 and Theorem 2]. From the inclusion

$$\mathcal{I}(\mathcal{H}^A) \subset \lim_\varepsilon \mathrm{crit}\left(\pi_{i,j}, \mathcal{V}^A_\varepsilon\right) \subset C \cap \mathbb{R}^n \subset \mathcal{H}^A,$$

we deduce that every real isolated point of $\mathcal{H}^A$ is also an isolated point of $C \cap \mathbb{R}^n$. □

We define the subroutine CurveLimitCheck that takes as input $f \in \mathbb{Q}[x_1, \ldots, x_n]$ and $A \in \mathrm{GL}(n,\mathbb{Q})$ and returns a set of isolating boxes $\mathscr{B}_2$. It calls to two subroutines:

- CurveLimit that takes as input f, A and a pair of index (i, j) and returns the eliminating polynomial of a rational parametrization encoding $\lim_\varepsilon \mathrm{crit}(\pi_{i,j}, \mathcal{H}^A_\varepsilon)$. The design of this subroutine follows Lemma 19.
- BivariateIsolated that takes as input a bivariate polynomial $U_{i,j}$ and computes the boxes isolating the real isolated points of $V(U_{i,j})$. This can be done by computing a cylindrical algebraic decomposition adapted to $U_{i,j} = 0$.

Algorithm 6: CurveLimitCheck

Input: $f \in \mathbb{Q}[x_1, \ldots, x_n]$, $A \in \mathrm{GL}(n,\mathbb{Q})$
Output: A set $\mathscr{B}_2$ of intervals of $\mathbb{R}$

for $1 \le i < j \le n$ **do**
  $U_{i,j} \leftarrow$ CurveLimit$(f, A, (i, j))$
  $\mathrm{boxes}_{i,j} \leftarrow$ BivariateIsolated$(U_{i,j})$
$\mathscr{B}_2 \leftarrow \cap_{1 \le i,j \le n} \mathrm{boxes}_{i,j}$
return $\mathscr{B}_2$

Summary.. To conclude this section, we show below the pseudo-code of our implementation. The subroutine Candidates is modified so that it returns, besides $\mathscr{C}$ encoding the candidates, a zero-dimensional parametrization $\mathscr{C}_2$ encoding the finite set $\mathfrak{C}_2$.

Algorithm 7: Implementation of IsolatedPoints

Input: A polynomial $f \in \mathbb{Q}[x_1, \ldots, x_n]$
Output: A zero-dimensional parametrization $\mathscr{C}$ and a set $\mathscr{B}$ of intervals of $\mathbb{R}$

A chosen randomly in $\mathrm{GL}(n, \mathbb{Q})$
$\mathscr{C}, \mathscr{C}_2 \leftarrow \mathsf{Candidates}(f, A)$
$\mathscr{B}_1 \leftarrow \mathsf{SimpleIdentification}(\mathscr{C}_2)$
if $|\mathscr{B}_1| = |\mathfrak{C} \cap \mathbb{R}^n|$ **then**
    return $\mathscr{B}_1$
$\mathscr{B}_2 \leftarrow \mathsf{CurveLimitCheck}(f, A)$
if $|\mathscr{B}_1| = |\mathscr{B}_2|$ **then**
    return $\mathscr{B}_1$
$\boldsymbol{a}$ chosen randomly in $\mathbb{Q}_+^n$
$e_0 \leftarrow \mathsf{ComputeE0}(f, \mathscr{C}, \boldsymbol{a})$
$\mathscr{B} \leftarrow \mathsf{IsIsolated\text{-}Approx}(f, \mathscr{C}, \boldsymbol{a}, e_0)$
return $(\mathscr{C}, \mathscr{B})$

7. Experimental results

In this section, we report on practical performances of our algorithms. Computations were done on an Intel(R) Xeon(R) CPU E7-4820 2GHz and 1.5 TB of RAM. Timings are given in seconds (s.), minutes (m.), hours (h.) and days (d.).

We take sums of squares of n random dense quadrics in n variables (with a non-empty intersection over $\mathbb{R}$); we obtain *dense quartics* defining a finite set of points. None of the considered examples can be solved using CAD implementations in Maple within 10 days.

Table 1 below reports on the timings of our implementation (Algorithm 7). Timings for the subroutine Candidates are given in the column CAND below. We use FGB library for computing Gröbner bases in order to perform algebraic elimination in our algorithms. The subroutine HasRealSolutions in Algorithm 4 is done by RAGLIB. Solving of zero-dimensional systems in the whole algorithm is done by MSOLVE and real root isolation is done by the command RootFinding[Isolate] in MAPLE.

The column CAND2 shows the timings for computing the zero-dimensional parametrization $\mathscr{C}_2$ and isolates its zeros (see Subsection 6.1). The column |real sols.|/ deg(w) shows the number of real candidates among the total number of candidates. This motivates the use of approximations, which runs only on candidates in $\mathbb{R}^n$, instead of computing over $\mathbb{Q}[t]/\langle w(t)\rangle$ which takes into account all candidates.

The column TEST1 reports on the timings of the first optimization (Algorithm 5). Exploiting the fact that isolating boxes are given by linear inequalities, we tweak RAGLIB for solving the associated decision problems. As explained in the end of Subsection 6.1, by isolating $\mathscr{C}_2$ with a small enough boxes in the subroutine SimpleIdentification, one can also obtain a certified output without computing e_0. In our examples, it is the case and we do not need to carry out further computations. Timings of other steps are given as an indication for further researches.

Timings for ComputeE0 are given in the column E0. The columns APPROX and RAGLIB respectively give the timings for computing the approximations and solving the decision problem by RAGLIB. Note that the implementation used for two columns APPROX and RAGLIB checks the emptiness of intersections of $\mathcal{H}$ with hypercubes (as explained in the end of Subsection 4.4). This

computation is similar to the one of TEST1 with the main difference coming from the fact that the isolating boxes computed in APPROX requires more precision. This results in linear polynomials, that define hypercubes, of larger bit-sizes, which makes the column RAGLIB slower than TEST1.

At the moment, we do not dispose of a geometric resolution algorithm for $\mathbb{Q}[t]/\langle w(t)\rangle$. The implementation of ComputeE0 relies on available tools such as FGB, MSOLVE that run over $\mathbb{Q}$. The complexity of this subroutine is actually bounded by $D^{O(n^2)}$ and the timings show that it is not practical. The value e_0 in these examples is obtained since we know in advance that the given real algebraic sets are finite.

n	CAND	CAND2	|real sols.|/ deg(w)	TEST1	**total**	E0	APPROX	RAGLIB
2	.1 s	.1 s	1/4	.1 s	.3 s	3 s	.1 s	.1 s
3	.2 s	.3 s	4/8	6 s	7 s	1 m	.1 s	10 s
4	1 s	4 s	2/16	1 m	1 m	20 h	.1 s	2 m
5	20 s	90 s	2/32	10 m	12 m	> 10 d	.2 s	15 m
6	30 m	2.5 h	2/64	4 h	7 h	> 10 d	20 s	6 h

Table 1: Experimental timings of Algorithm 7

References

[1] Arnon, D. S., 1988. A cluster-based Cylindrical Algebraic Decomposition algorithm. J. Symb. Comput. 5 (1/2), 189–212.

[2] Basu, S., Pollack, R., Roy, M.-F., 2006. Algorithms in Real Algebraic Geometry (Algorithms and Computation in Mathematics). Springer-Verlag.

[3] Basu, S., Roy, M.-F., 03 2022. Quantitative curve selection lemma. Mathematische Zeitschrift 300.

[4] Berthomieu, J., Eder, C., Safey El Din, M., 2021. Msolve: A library for solving polynomial systems. In: Proceedings of the 2021 on International Symposium on Symbolic and Algebraic Computation. ISSAC '21. Association for Computing Machinery, New York, NY, USA, pp. 51–58.

[5] Bochnak, J., Coste, M., Roy, M.-F., 1998. Real Algebraic Geometry. Springer.

[6] Borcea, C., Streinu, I., 2015. Geometric auxetics. Proc. R. Soc. Lond., A, Math. Phys. Eng. Sci. 471 (2184), 24.

[7] Borcea, C. S., Streinu, I., 2018. Periodic auxetics: structure and design. Q. J. Mech. Appl. Math. 71 (2), 125–138.

[8] Chillingworth, D., Demazure, M., 2013. Bifurcations and Catastrophes: Geometry of Solutions to Nonlinear Problems. Universitext. Springer Berlin Heidelberg.

[9] Collins, G. E., 1976. Quantifier elimination for real closed fields by cylindrical algebraic decomposition: a synopsis. ACM SIGSAM Bulletin 10 (1), 10–12.

[10] Connelly, R., Whiteley, W., 1992. The stability of tensegrity frameworks. International Journal of Space Structures 7 (2), 153–163.

[11] Faugère, J.-C., September 2010. FGb: A Library for Computing Gröbner Bases. In: Fukuda, K., Hoeven, J., Joswig, M., Takayama, N. (Eds.), Mathematical Software - ICMS 2010. Vol. 6327 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, Berlin, Heidelberg, pp. 84–87.

[12] Faugère, J.-C., Gianni, P., Lazard, D., Mora, T., 1993. Efficient computation of zero-dimensional gröbner bases by change of ordering. Journal of Symbolic Computation 16 (4), 329–344.

[13] Gaspar, N., Ren, X., Smith, C., Grima, J., Evans, K., 2005. Novel honeycombs with auxetic behaviour. Acta Materialia 53 (8), 2439 – 2445.

[14] Giusti, M., Lecerf, G., Salvy, B., 2001. A Gröbner free alternative for polynomial system solving. Journal of Complexity 17 (1), 154 – 211.

[15] Grima, J. N., Evans, K. E., May 2006. Auxetic behavior from rotating triangles. Journal of Materials Science 41 (10), 3193–3196.

[16] Lakes, R., 1987. Foam structures with a negative poisson's ratio. Science 235 (4792), 1038–1040.

[17] Le, H. P., Safey El Din, M., de Wolff, T., 2020. Computing the real isolated points of an algebraic hypersurface. In: Emiris, I. Z., Zhi, L. (Eds.), ISSAC '20: International Symposium on Symbolic and Algebraic Computation, Kalamata, Greece, July 20-23, 2020. ACM, pp. 297–304.

[18] Oster, M., Dias, M. A., de Wolff, T., E. Evans, M., 2021. Reentrant tensegrity: A three-periodic, chiral, tensegrity structure that is auxetic. Science Advances 7.
[19] Roth, B., Whiteley, W., 1981. Tensegrity frameworks. Trans. Am. Math. Soc. 265, 419–446.
[20] Rouillier, F., Roy, M., Safey El Din, M., 2000. Finding at least one point in each connected component of a real algebraic set defined by a single equation. J. Complexity 16 (4), 716–750.
[21] Rouillier, F., Zimmermann, P., 2004. Efficient isolation of polynomial's real roots. Journal of Computational and Applied Mathematics 162 (1), 33–50.
[22] Safey El Din, M., 2005. Computing sampling points on a singular real hypersurface using Lagrange's system. Research Report RR-5464, INRIA.
[23] Safey El Din, M., 2017. Real alebraic geometry library, RAGlib (version 3.4). URL https://www-polsys.lip6.fr/~safey/RAGLib/
[24] Safey El Din, M., Schost, E., 2003. Polar varieties and computation of one point in each connected component of a smooth real algebraic set. In: Proc. of the 2003 Int. Symp. on Symb. and Alg. Comp. ISSAC '03. ACM, p. 224–231.
[25] Safey El Din, M., Schost, É., Jan. 2017. A nearly optimal algorithm for deciding connectivity queries in smooth and bounded real algebraic sets. J. ACM 63 (6), 48:1–48:37.
[26] Sagraloff, M., 2014. On the complexity of the Descartes method when using approximate arithmetic. Journal of Symbolic Computation 65, 79–110.
[27] Vorobjov, N., 1999. Complexity of computing the local dimension of a semialgebraic set. Journal of Symbolic Computation 27 (6), 565–579.
[28] Yang, W., Li, Z.-M., Shi, W., Xie, B.-H., 2004. Review on auxetic materials. Journal of Materials Science 39, 3269–3279.