UNIVERSITY OF ALCALÁ

PHD PROGRAM IN SCIENCE
DEPARTMENT OF PHYSIC AND MATHEMATICS

Affine Equivalences, Similarities, and Symmetries of Special Types of Curves and Surfaces

Ph.D.THESIS

Presented by

Emily Nazareth Quintero de D'Alessio

2021

PHD PROGRAM IN SCIENCE
DEPARTMENT OF PHYSIC AND MATHEMATICS

Ph.D.THESIS

Affine Equivalences, Similarities, and Symmetries of Special Types of Curves and Surfaces

Presented by

Emily Nazareth Quintero de D'Alessio

Advised by
Juan Gerardo Alcázar Arribas

Alcalá de Henares, 2021

Mejor es adquirir sabiduría que oro preciado; Y adquirir inteligencia vale más que la plata.

PROVERBIOS 16:16

This thesis has been supported by a grant from the Carolina Foundation and the University of Alcalá.

Agradecimientos

Doy gracias al que me fortaleció en todo momento, a Dios. Gracias por permitirme conocer un poco más lo maravilloso de tu creación por medio de las Matemáticas. Gracias por tu exagerado amor manifestado a través de Jesús, ese controversial judío que revolucionó todo a su paso y que sigue revolucionando vidas, incluyendo la mía.

A la Fundación Carolina y la Universidad de Alcalá por otorgarme la beca que me permitió hacer esta investigación; y al Vicerrectorado de Investigación y Transferencia, especialmente al Vicerrector Dr. D. Francisco J. de la Mata de la Mata, y a Carmen de la Peña, por gestionar y aprobar las solicitudes de beca de alojamiento cada vez que lo necesité. Gracias por invertir en la formación de profesores de América Latina.

A Juange, quien ha sido más que un director para mí. Gracias por tanta paciencia, por tu cariño, tu comprensión, tu disposición y todo el apoyo que desinteresadamente me has dado. Ha sido un privilegio contar con tu experiencia y con tu nivel matemático y personal. Admiro tu pasión por las matemáticas y tu manera de transmitirlas. ¡GRACIAS!

Al personal administrativo y docente del Departamento de Física y Matemáticas, por recibirme y tratarme con tanto cariño desde que llegué.

A Vincenzo, mi esposo y mi novio de siempre, el autor de mis sonrisas diarias. Gracias por tu amor gigante y genuino, por impulsarme cada día a llegar más lejos, por creer y ver lo mejor en mí en cada etapa, y por ser mi cómplice incondicional. Te amo no, ¡lo siguiente!

A mi mami, por levantarme con cada consejo y cada oración, y por su increíble habilidad de sacar fuerzas para enfrentar los retos de la vida. ¡Te amo y te admiro!

A mi papá, mis abuelas, mis amigos, mi familia de sangre y de fe, y a mis pastores, gracias a todos por levantar mis brazos y caminar conmigo durante el proceso, aplanando también el camino.

Resumen

El tema central de este trabajo es detectar y calcular equivalencias afines entre dos curvas y/o superficies biracionalmente parametrizadas con propiedades específicas.

En el Capítulo 1, proveemos un estado del arte sobre el tema, haciendo una revisión de publicaciones recientes que abordan este problema tanto para curvas como para superficies algebraicas definidas implícita o paramétricamente.

En el Capítulo 2 desarrollamos un método para calcular todas las equivalencias afines entre dos superficies racionales regladas, definidas por parametrizaciones racionales y propias (inyectivas en casi todo punto), sin calcular ni hacer uso de sus ecuaciones implícitas. La idea fundamental es encontrar la forma de la transformación de Cremona equivalente en el espacio de parámetros, y se basa en la resolución de sistemas polinómicos.

En el Capítulo 3, describimos un algoritmo eficiente para detectar si dos curvas trigonométricas dadas, es decir, curvas paramétricas cuyas componentes son series truncadas de Fourier, en cualquier dimensión, son afínmente equivalentes. En este caso abordamos tanto equivalencias exactas, como aproximadas. En el caso exacto el algoritmo se reduce al cálculo de un máximo común divisor univariado, mientras que en el caso aproximado, donde los coeficientes de las parametrizaciones están dados con precisión finita, es necesario calcular un gcd aproximado.

Finalmente, en el Capítulo 4 estudiamos la detección de semejanzas entre dos curvas paramétricas acotadas y planas con un enfoque particular para curvas cerradas. El algoritmo es válido para parametrizaciones completamente generales, no sólo racionales, y también se considera en el caso aproximado. La estrategia se basa en el cálculo de los centros de gravedad y tensores de inercia de las curvas o de las regiones planas encerradas por las curvas. Tanto los centros de gravedad como los tensores de inercia tienen buenas propiedades cuando se les aplica una semejanza. En particular, un centro de gravedad es enviado en el otro y las matrices que representan los tensores de inercia

satisfacen una relación simple. Utilizando ambas identidades, y salvo en ciertos casos patológicos, las semejanzas pueden determinarse.

Abstract

The central topic of this thesis is the detection and computation of affine equivalences between two curves or surfaces with specific properties.

Chapter 1 provides a state-of-the-art on the topic, based on recent publications regarding this problem for projective and affine transformations, similarities, and isometries between algebraic curves and surfaces either implicitly or parametrically defined.

In Chapter 2, we develop a method for computing all the affine equivalences between two rational ruled surfaces defined by rational parametrizations without computing or using their implicit equations. The problem is translated into the parameter space, where the general form of the underlying Cremona transformation is discovered, and relies on polynomial system solving.

In Chapter 3, we describe an efficient algorithm to detect whether two given trigonometric curves, i.e., two parametrized curves whose components are truncated Fourier series, in any dimension, are affinely equivalent. In this case, we also deal with approximate affine equivalences. In the exact case, the algorithm boils down to univariate gcd computation, so it is efficient and fast. In the approximate case, where the coefficients of the parametrizations are given with finite precision, the univariate gcd computation is replaced by the computation of approximate gcds.

Finally, in Chapter 4, we provide an algorithm to compute the similarities between two bounded, planar parametrized curves with a particular approach on the case when the curves are closed. The algorithm is valid for completely general parametrizations, not only rational, and the approximate case is also considered. The strategy is based on the computation of centers of gravity and inertia tensors of the considered curves or of the planar regions enclosed by the curves, which have good properties when a similarity transformation is applied: the centers of gravity are mapped onto each other, and the matrices representing the inertia tensors satisfy a simple relationship. Using both properties, and except for certain pathological cases, the similarities can be found.

INDEX

INTRODUCTION

This thesis addresses the problem of detecting when two objects are related by an affine equivalence, in the case of surfaces and curves with specific structures. Particular instances of this problem are similarity and symmetry detection. This question is of interest in applied fields like Computer Aided Geometric Design, Pattern Recognition, and Computer Vision. It has been broadly treated, mainly for objects with a relatively weak structure (e.g., cloud points, images with or without occluded parts, solids, polygons, and polyhedrons). In our case, however, we address objects with a strong structure and use their structure to provide efficient algorithms relying on symbolic and symbolic-numeric methods.

In fields like Pattern Recognition or Computer Vision, recognizing objects up to a certain transformation is important since quite often, the object that one is analyzing has undergone some kind of deformation. In Computer Vision, the picture of an object is the image of the initial object under a projective transformation. In Pattern Recognition, one needs to compare an image with other images stored in a database, but the object to recognize is often placed in a different position or has a different scale than the objects in the database; this implies that a similarity is relating both. In Computer Aided Geometric Design, symmetry computation is important to reduce the amount of memory needed to store an image and certify that the geometry of the object is correct since symmetries are essential elements of the shape.

As a result, in the literature, there is a wide variety of techniques approaching these problems. The list is very long; to quote just a few: affine moments and algebraic invariants [62, 103, 104, 111], B-splines [62], differential invariants [36, 40, 115], Fourier descriptors [98], spherical harmonic analysis [76], statistics [32, 35, 78, 88, 52, 71, 77], spectral analysis [74], extended Gauss images [105], discrete methods [23, 38, 64, 72].

However, until recent years in most of the references on the topic, there were very few assumptions on the structure of the objects to analyze. Some exceptions are [112, 116, 117] and [70, 69, 107], where algebraic curves are considered, sometimes working with a complex representation of the curves. In 2014 Alcázar et al. started a series of papers [2, 12, 11, 13, 9, 14, 1, 15, 10], first regarding symmetries, then moving to similarities, where this topic was explored for planar and space rational curves, some specific types of rational surfaces, and implicit algebraic curves, where a strong use of the structure of the variety was made. These papers were followed by works by other authors, e.g., [33, 34, 56, 57, 63, 94], extending the problem to affine and projective equivalences for rational curves in any dimension, rational surfaces, and implicit curves and surfaces. In all these references, in contrast to works more focused on applied fields, tools from symbolic and symbolic-numeric computation are massively used, as well as notions and results from Algebraic Geometry.

This thesis follows the path of the references in the previous paragraph, sometimes exploiting results developed in some of those papers, and addresses three questions related to the computation of affine equivalences and similarities:

(i) Exact affine equivalences between two rational ruled surfaces.

(ii) Exact and approximate affine equivalences between two trigonometric curves in any dimension, i.e., parametrized curves whose components are truncated Fourier series.

(iii) Approximate similarities between two bounded, parametric, planar curves, defined by non-necessarily rational parametrizations, using the notions, well-known in Mechanics, of center of gravity and inertia tensor.

In more detail, the structure of this thesis is the following. The first chapter, Chapter 1, is introductory and recalls some basic notions related to projective space, affine and projective equivalences, similarities, symmetries, rational curves and surfaces, and state of the art on the topic. In particular, we review the works published on the topic in detail since 2012, which follow approaches close to the one in this thesis.

In Chapter 2, we provide an algorithm to compute the affine equivalences between two ruled rational surfaces. To do this, we first discover the structure of the Cremona transformation, which is associated, in the parameter space (the plane), to any affine equivalences between the surfaces. Using this and taking advantage of ideas in [56], where the problem of computing projective equivalences between rational curves is addressed, we provide an algorithm to solve the problem. For isometries and symmetries, the algorithm has extra advantages, which we analyze. Furthermore, we also show the efficiency of the method in an abundance of examples.

In Chapter 3, we study the computation of exact and approximate affine equivalences between two trigonometric curves, namely parametric curves whose components are truncated Fourier series, widely used in applications. By using a well-known trick, these curves admit a rational parametrization. However, this parametrization has special properties that can be used to improve the computation, compared to the more general method suggested in [56]. As in Chapter 2, any affine equivalence between the curves has an underlying transformation in the parameter space (the line, in this case), which is a Möbius transformation. However, we show that this transformation has a very special form. From here, we present an algorithm that resorts to univariate GCDS. In the approximate case, GCDS are replaced by approximate GCDS. The algorithm is efficient, and evidence of this efficiency is provided through numerous examples. The extension to more general parametrizations (e.g., non-rational) using truncations of their Fourier expansions is also discussed; however, this idea, although natural, is not really efficient.

In Chapter 4, we address the computation of similarities between two parametric, bounded, planar curves defined by parametrizations that are not necessarily rational. The algorithm we provide uses the center of gravity and inertia tensor, well-known in

Mechanics, and their properties under similarities. Although the approach is presented for planar curves, it can be generalized to curves in any dimension and even to surfaces. The main idea is that under a similarity, the centers of gravity are mapped to each other, and the inertia tensors obey the transformation law of Euclidean tensors; by using this, a superset of the similarities between the curves can be computed. If additionally, the curves are closed, then we have extra tools that we also discuss. In the presence of inaccuracies, we use approximate GCDS to solve the problem. Again, we provide examples to show the performance of the algorithm.

The thesis closes with a section containing some ideas for further work.

The results in this thesis have given rise to the following publications [17, 18, 19, 20]; the last three correspond to papers published in journals included in the Journal of Citation Reports:

(1) Alcázar J.G., Quintero E. (2018), *Computing Symmetries of Ruled Rational Surfaces*, Actas de los Encuentros de Álgebra Computacional y Aplicaciones 2018. Monografías de la Real Academia de Ciencias. Zaragoza. Vol. 43, pp. 35–38. ISSN: 1132-6360.

(1) Alcázar J.G., Quintero E. (2020), *Affine equivalences, isometries and symmetries of ruled rational surfaces,* Journal of Computational and Applied Mathematics Vol. 364, 112339.

(2) Alcázar J.G., Quintero E. (2020), *Affine Equivalences of Trigonometric Curves,* Acta Applicandae Mathematicae Vol. 170, pp. 691–708.

(3) Alcázar J.G., Quintero E. (2020), *Exact and approximate similarities of non-necessarily rational planar, parametrized curves, using centers of gravity and inertia tensors,* International Journal of Algebra and Computation, to appear.

Furthermore, the results in this thesis have also been presented by the author in the following conferences:

- 2018: First BYMAT Conference: "Bringing Young Mathematicians Together". ICMAT, Madrid, Spain. Title of the talk: "Computing Symmetries of Ruled

Rational Surfaces".

- 2019: Second BYMAT Conference: "Bringing Young Mathematicians Together", ICMAT, Madrid, Spain. Poster: "Affine equivalences, isometries and symmetries of ruled rational surfaces".
- 2020: 5th EACA International School on Computer Algebra and its Applications. Basque Center for Applied Mathematics (BCAM), Bilbao, Spain. Title of the talk: "Affine equivalences of trigonometric curves".
- 2020: Third BYMAT Conference: "Bringing Young Mathematicians Together", ICMAT, Madrid, Spain. Title of the talk: "Affine equivalences of trigonometric curves".

CHAPTER 1

AFFINE AND PROJECTIVE EQUIVALENCES, SIMILARITIES AND ISOMETRIES

This chapter provides preliminary concepts regarding projective and affine transformations, from which isometries and similarities are particular instances. Furthermore, the chapter accounts for the state of the art of the problem treated in this thesis.

1.1 Projective space

The *projective space* $\mathbb{P}^n(\mathbb{R})$ (see for instance [44]) is defined as the set of equivalence classes of $\sim$ on $\mathbb{R}^{n+1}\backslash\{0\}$,

$$\mathbb{P}^n(\mathbb{R}) = (\mathbb{R}^{n+1}\backslash\{0\})/\sim,$$

where $(a_1, a_2, \ldots, a_{n+1}) \sim (b_1, b_2, \ldots, b_{n+1})$ if and only if there exists $\lambda \in \mathbb{R}\backslash\{0\}$ such that $(a_1, a_2, \ldots, a_{n+1}) = \lambda(b_1, b_2, \ldots, b_{n+1})$. Given $(x_1, \ldots, x_{n+1}) \in \mathbb{R}^{n+1}\backslash\{0\}$, its corresponding equivalence class in $\mathbb{P}^n(\mathbb{R})$ is denoted as $[x_1 : x_2 : \cdots : x_{n+1}]$; the numbers $x_1, \ldots, x_{n+1}$ are called the *homogeneous coordinates* of the point $[x_1 : x_2 : \cdots : x_{n+1}] \in \mathbb{P}^n(\mathbb{R})$. Notice that since $[x_1 : \cdots : x_{n+1}]$ and $[\lambda x_1 : \cdots : \lambda x_{n+1}]$, with $\lambda \in \mathbb{R} - \{0\}$, represent the same point in $\mathbb{P}^n(\mathbb{R})$, the elements of $\mathbb{P}^n(\mathbb{R})$ can be seen geometrically as lines in $\mathbb{R}^{n+1}$ through the origin.

Additionally, if $x_{n+1} \neq 0$ the projective point $[x_1 : \cdots : x_n : x_{n+1}]$ coincides with $\left[\frac{x_1}{x_{n+1}} : \cdots : \frac{x_n}{x_{n+1}} : 1\right]$, which can be identified with the affine point $\left(\frac{x_1}{x_{n+1}}, \cdots, \frac{x_n}{x_{n+1}}\right)$. In particular, $\mathbb{R}^n \subset \mathbb{P}^n(\mathbb{R})$. If $x_{n+1} = 0$ we say that $[x_1 : \cdots : x_n : 0]$ is a *point at infinity*. Thus, $\mathbb{P}^n(\mathbb{R})$ is the union of $\mathbb{R}^n$ and the hyperplane $x_{n+1} = 0$, which is the hyperplane consisting of the points at infinity.

1.2 Affine and projective equivalences

A *projective* transformation, also called a *projectivity*, is a mapping $\widehat{f} : \mathbb{P}^n(\mathbb{R}) \to \mathbb{P}^n(\mathbb{R})$, where $\widehat{f}(\widehat{\mathbf{x}}) = \mathbf{Q}\widehat{\mathbf{x}}$ with

$$\mathbf{Q} = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} & a_{1,n+1} \\ \vdots & \cdots & \vdots & \vdots \\ a_{n,1} & \cdots & a_{n,n} & a_{n,n+1} \\ a_{n+1,1} & \cdots & a_{n+1,n} & a_{n+1,n+1} \end{bmatrix} \tag{1.1}$$

a nonsingular matrix, and $\widehat{\mathbf{x}} = [x_1 : \cdots : x_n : x_{n+1}]$. Notice that $\widehat{f}$ corresponds to a mapping $f : \mathbb{R}^n \to \mathbb{R}^n$ of the type $f(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_n(\mathbf{x}))$ with

$$f_i(\mathbf{x}) = f_i(x_1, \ldots, x_n) = \frac{a_{i,1}x_1 + \cdots + a_{i,n}x_n +, a_{i,n+1}}{a_{n+1,1}x_1 + \cdots + a_{n+1,n}x_n + a_{n+1,n+1}}, \tag{1.2}$$

for $i = 1, \ldots, n$.

An *affine transformation*, also called an *affinity*, is a mapping $f : \mathbb{R}^n \longrightarrow \mathbb{R}^n$,

$$f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}, \qquad \mathbf{x} \in \mathbb{R}^n, \tag{1.3}$$

with $\boldsymbol{b} \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$ a nonsingular square matrix. Obviously, every affinity is a projectivity, but the converse is false. Notice also that affinities preserve the hyperplane at infinity.

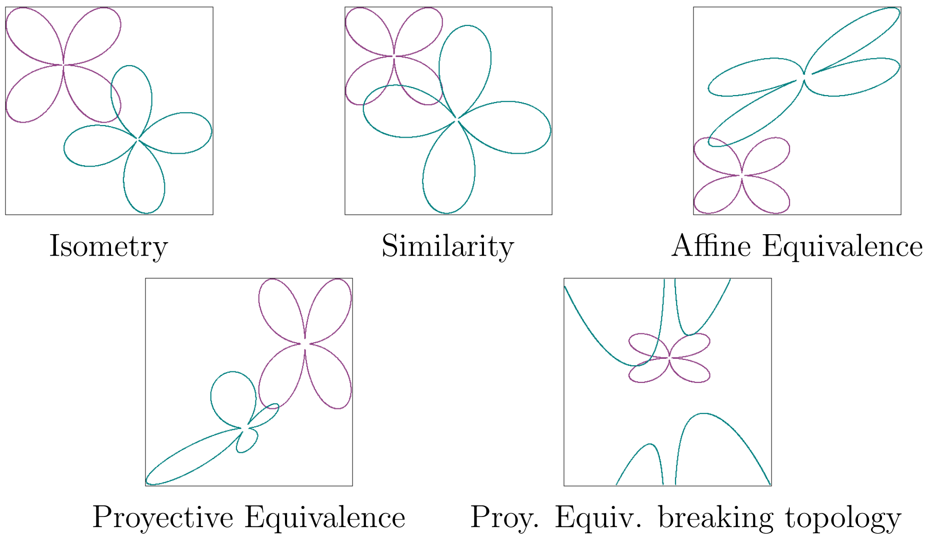


Figure 1.1: Different transformations

Definition 1.1. *Let $\mathcal{V}_1, \mathcal{V}_2 \subset \mathbb{R}^n$. We say that $\mathcal{V}_1, \mathcal{V}_2$ are projectively (resp.* affinely*) equivalent if there exists a projectivity (resp. affinity) f such that $f(\mathcal{V}_1) = \mathcal{V}_2$. Furthermore, we say that f is a projective (resp. affine) equivalence between $\mathcal{V}_1$ and $\mathcal{V}_2$.*

Affine transformations model smooth deformations, so they preserve topology. However, this is not necessarily true for projective transformations, since affine points can be mapped to points at infinity, and viceversa. Figure 1.1 shows, in green, the image of a four-leaved rose, in purple, under different affine and projective transformations. In particular, the right-most picture in the second row of Fig. 1.1 corresponds to the image of the four-leaved rose under a projective transformation that maps the four-fold singularity of the rose onto a point at infinity, which breaks the topology of the curve (in this case, the image of the rose is a curve with four connected components). In the first row of Fig. 1.1, the purple curve and the green curve are affinely (and therefore also projectively) equivalent. In the second row of Fig. 1.1, the purple curve and the green curve are projectively, but not affinely, equivalent.

Fig. 1.1 also shows two special cases of affine transformations, namely *isometries* and *similarities*, which have extra metric properties: isometries preserve distances and similarities preserve angles, although similarities do not necessarily preserve distances. In more detail:

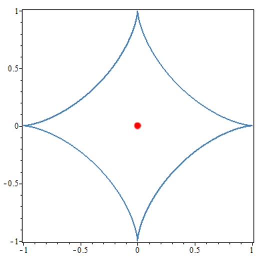
(i) If the matrix $\boldsymbol{A}$ in Eq. (1.3) is orthogonal, i.e., $\boldsymbol{A}^T\boldsymbol{A} = I$ where I is the identity $n \times n$ matrix, we say that f defines an *isometry*. In this case, $\det(\boldsymbol{A}) = \pm 1$. An isometry is said to be *direct* if it preserves the orientation, in which case $\det(\boldsymbol{A}) = 1$, and is called *opposite* when it reverses the orientation, in which case $\det(\boldsymbol{A}) = -1$. Isometries preserve Euclidean distances and are also called *rigid motions*. An isometry $f : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ is an *involution* if $f \circ f = \mathrm{id}_{\mathbb{R}^n}$. In this case, $\boldsymbol{A}^2 = I$ is the identity matrix and $\boldsymbol{b} \in \ker(\boldsymbol{A} + I)$.

(ii) If $\boldsymbol{A} = \lambda\boldsymbol{Q}$ with $\boldsymbol{Q}$ an orthogonal matrix and $\lambda > 0$, we say that f defines a *similarity*. Similarities form a group under composition from which isometries are a subgroup. Observe that a similarity is the composition of a rigid motion and a homothety; thus, two objects related by a similarity only differ in position and scaling.

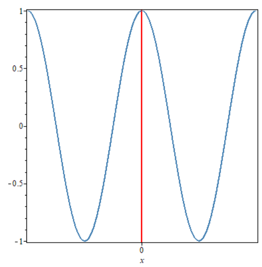
Two objects $\mathcal{V}_1, \mathcal{V}_2 \subset \mathbb{R}^n$ are *isometric* (resp. *similar*) when there exists an isometry (resp. similarity) f such that $f(\mathcal{V}_1) = \mathcal{V}_2$. If $\mathcal{V}_1 = \mathcal{V}_2 = \mathcal{V}$, and if $f : \mathcal{V} \longrightarrow \mathcal{V}$ defines an isometry, we say that $\mathcal{V}$ is *symmetric*, and that f is a symmetry of $\mathcal{V}$.

When $n = 2$, i.e., for $\mathcal{V} \subset \mathbb{R}^2$, notable symmetries are *rotations* around a point and *reflections*in a line. When $n = 3$, i.e., for $\mathcal{V} \subset \mathbb{R}^3$, notable symmetries are *reflections* in a plane, *rotations* about an axis, and *central inversions* (symmetries with respect to a point or *central symmetries*). Rotations by an angle of π are called *half-turns* or *axial symmetries*. Furthermore, central symmetries, reflections, and axial symmetries are involutions; rotations, in general, are not. Also, rotations are direct isometries, while reflections and central symmetries are opposite.

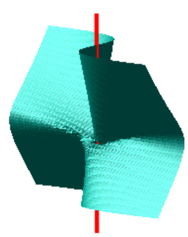
Fig. 1.2 shows some examples of planar and space symmetries. In the first row of Fig. 1.2 we have curves with rotational symmetry (left; the center of rotation is plotted in red, and the angles of rotation are $\frac{\pi}{2}, \pi, \frac{3\pi}{2}$) and reflectional symmetry (right; the reflection axis is plotted in red). In the second row of Fig. 1.2 we have surfaces with axial symmetry (left; the axis is plotted in red), planar symmetry (middle; the reflection



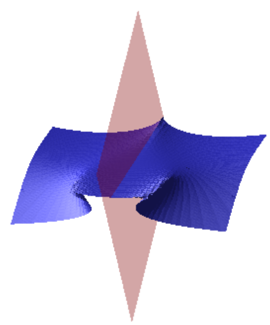
Rotational symmetry



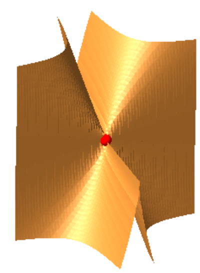
Reflection (line) symmetry



Axial symmetry
(Rotation of π)



Mirror symmetry



Central symmetry

Figure 1.2: Plane and space symmetries

plane is shown) and central symmetry (right; the center of symmetry is plotted in red).

1.3 Brief review of rational curves and surfaces

We say that $\mathcal{V} \subset \mathbb{R}^n$ is *algebraic*, if $\mathcal{V}$ is the common set of zeroes of finitely many polynomials $f_1(x_1, \ldots, x_n), \ldots, f_p(x_1, \ldots, x_n)$, which implicitly define $\mathcal{V}$. When $n = 2$ a nonconstant polynomial $f(x, y)$ implicitly defines the algebraic curve $f(x, y) = 0$. Similarly, when $n = 3$ a nonconstant polynomial $f(x, y, z)$ implicitly defines the algebraic surface $f(x, y, z) = 0$. We can have algebraic curves in any dimension (as algebraic sets of dimension one); in order to implicitly define a curve in $\mathbb{R}^n$ we need at least $n-1$ polynomials without common factors.

A rational function is a quotient of relatively prime polynomials. We say that an algebraic curve $\mathcal{C} \subset \mathbb{R}^n$ is *rational* if it admits a *rational* parametrization, i.e., a parametrization

$$\boldsymbol{x}(t) = (x_1(t), \ldots, x_n(t)) \tag{1.4}$$

where each $x_i(t)$ is a rational function. Likewise, an algebraic surface $S \subset \mathbb{R}^3$ is rational, if it has a rational parametrization

$$\boldsymbol{x}(t, x) = (x(t, x), y(t, s), z(t, s)) \tag{1.5}$$

where $x(t, s)$, $y(t, s)$, $z(t, s)$ are rational functions. Notice that if a variety admits a rational parametrization then it admits infinitely many rational parametrizations: indeed, every composition of a parametrization with a rational function provides a new parametrization. Thus, rational parametrizations are not unique.

The rational parametrization $\boldsymbol{x}(t)$ in Eq. (1.4) is said to be *proper* if it has a rational inverse $t = t(x_1, \ldots, x_n)$; in particular, this means that $\boldsymbol{x}(t)$ is invertible, so that there are just finitely many pairs $(t, \bar{t})$, $t \neq \bar{t}$, satisfying that $\boldsymbol{x}(t) = \boldsymbol{x}(\bar{t})$. The same notion extends to rational parametrizations of surfaces. In other words, the notion of *properness* implies that $\boldsymbol{x}^{-1}$ exists and is rational.

For curves, there are efficient algorithms (see [97]) to check whether a given parametrization like Eq. (1.4) is proper, and to properly reparametrize it in case it is not proper. Additionally, if $\tilde{\boldsymbol{x}}(t)$, $\boldsymbol{x}(t)$ are proper parametrizations of a same curve then there exists a *Möbius transformation* verifying that $\tilde{\boldsymbol{x}} = \boldsymbol{x} \circ \varphi$. Recall that a Möbius transformation is a mapping

$$\varphi : \mathbb{R} \longrightarrow \mathbb{R}, \quad \varphi(t) = \frac{at+b}{ct+d}, \quad \text{with} \quad \Delta := ad - bc \neq 0. \tag{1.6}$$

The same problem, i.e., checking properness and proper reparametrizing, for rational surfaces is harder, and not completely solved yet. One can check [83, 84] and the references therein for further details.

1.4 State of the art

Detecting symmetries, similarities and, with more generality, affine or projective equivalences between two objects is a classical problem that has received much attention in the literature. Many references concerning this problem have been provided in the Introduction. In this section, we account for the state of the art of recent approaches to the problem using tools from Symbolic Computation, and assuming that the objects involved in the computation have a strong structure, namely that they are algebraic (mostly curves and surfaces). We consider first the case of rational curves, then rational surfaces, and finally implicit algebraic curves and surfaces.

1.4.1 The case of rational curves

In this subsection we review the literature on the problem for rational curves. The first paper in this direction, under the approach we are considering here, was [2], published in 2014. In [2] the problem of computing the symmetries of a polynomially parametrized planar curve $\mathcal{C} \subset \mathbb{R}^2$, parametrized by $\boldsymbol{x}(t)$, is addressed. The main idea, which will be generalized later to other situations, is the following. Suppose that the polynomial parametrization $\boldsymbol{x}(t)$ is proper; this is a key assumption. If $\boldsymbol{x}$ is proper, then $\boldsymbol{x}^{-1}$ exists and is a rational mapping. Now if $f : \mathcal{C} \to \mathcal{C}$ is a symmetry of $\mathcal{C}$, then there exists a function φ that makes commutative the diagram below:

$$\begin{tikzcd} \mathcal{C} \arrow[r, "f"] & \mathcal{C} \\ \mathbb{C} \arrow[u, dashed, "\boldsymbol{x}"] \arrow[r, dashed, "\varphi"'] & \mathbb{C} \arrow[u, dashed, "\boldsymbol{x}"'] \end{tikzcd} \tag{1.7}$$

Indeed, $\varphi = \boldsymbol{x}^{-1} \circ f \circ \boldsymbol{x}$. This commutative diagram is absent in [2], but it will come up in subsequent publications on the matter of the same author, and is somehow at the core of [2].

Observe that the function φ in Eq. (1.7) is rational, because it is the composition of three rational functions. Furthermore, $\varphi^{-1} = \boldsymbol{x}^{-1} \circ f^{-1} \circ \boldsymbol{x}$ exists, so φ is a birational mapping of the complex line. However, the only birational mappings of the complex line (see [97]) are the Möbius transformations.

Thus, from Eq. (1.7), in order to find the symmetries f of $\mathcal{C}$ we need to find the Möbius transformations φ such that

$$f(\boldsymbol{x}(t)) = \boldsymbol{x}(\varphi(t)). \tag{1.8}$$

Moreover, because of Eq. (1.8) and since $\boldsymbol{x}(t)$ is polynomial we deduce that φ must be linear, i.e., $\varphi(t) = at + b$. Now for each type of symmetry (rotational, reflectional) Eq. (1.8) leads to a triangular system from which the parameters a, b and the symmetry itself are derived. Furthermore, the results in [2] provide several general results on the existence of symmetries of polynomial curves and some prohibitions. Also, the algorithm derived from the main results of [2] to compute the symmetries of a polynomial curve is high-speed and allows to compute the symmetries of a polynomial curve with high coefficients and degree > 50 in just a few seconds.

The ideas in [2] were later generalized in [12] to compute the symmetries of rational, not necessarily polynomial, planar and space curves. Again, the idea is to exploit the commutative diagram in Eq. (1.7), which is, in fact, valid for rational curves in any dimension whenever they are defined by a proper parametrization $\boldsymbol{x}$. The key is to first find the Möbius transformation φ in Eq. (1.7), although now φ is not necessarily linear anymore. In order to do this, since f is a symmetry, Eq. (1.8) is written as

$$\boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{b} = \boldsymbol{x}(\varphi(t)),$$

with $\boldsymbol{A}$ an orthogonal matrix. Differentiating once and twice and using the fact that $\boldsymbol{A}$ is orthogonal, i.e., that $\boldsymbol{A}$ preserves the Euclidean norm of a vector, the parameters of φ are written in terms of just one of them, and a univariate equation for this parameter is derived. Thus, φ is computed, and $\boldsymbol{A}, \boldsymbol{b}$ are derived from φ. The method is valid for computing the symmetries of planar rational curves and the involutions of space rational curves; however, in general the method in [12] is not enough to compute all the rotational symmetries of space rational curves.

The method of [12] for space rational curves is improved in [13]. The algorithm in this paper is valid for computing all the symmetries of the curve, and makes use of the fact that the *curvature* and the *torsion* of a space rational curve

$$\kappa_{\boldsymbol{x}} = \frac{\|\boldsymbol{x}' \times \boldsymbol{x}''\|}{\|\boldsymbol{x}'\|^3}, \ \tau_{\boldsymbol{x}} = \frac{\langle \boldsymbol{x}' \times \boldsymbol{x}'', \boldsymbol{x}''' \rangle}{\|\boldsymbol{x}' \times \boldsymbol{x}''\|^2}$$

are pointwise invariant under a symmetry, except perhaps for the sign in the case of the torsion. In other words, for a symmetry f we have

$$\kappa_{f\circ\boldsymbol{x}} = \kappa_{\boldsymbol{x}}, \ \tau_{f\circ\boldsymbol{x}} = \pm\tau_{\boldsymbol{x}}$$

where the sign in the case of τ depends on whether f preserves or reverses the orientation. Observe that $\kappa_{\boldsymbol{x}}$ is a rational function; $\tau_{\boldsymbol{x}}$ is not, but however $\tau_{\boldsymbol{x}}^2$ is a rational function. Furthermore, for a Möbius function φ,

$$\kappa_{\boldsymbol{x}\circ\varphi} = \kappa_{\boldsymbol{x}}, \ \tau_{\boldsymbol{x}\circ\varphi} = \tau_{\boldsymbol{x}}.$$

Thus, setting $\kappa(t) = \kappa(s)$ and $\tau^2(t) = \tau^2(s)$, we get two bivariate polynomials. Then, the existence of "Möbius-like" factors $F(t, s) = (ct + d)s - (at + b)$ in the gcd of these two polynomials allows to compute the Möbius functions, and in turn, the symmetries of the curve. The proof of this fact relies heavily in the Fundamental Theorem of Space Curves, which states that the curvature and torsion functions characterize a space curve up to rigid motions. The complexity of the resulting algorithm is $\mathcal{O}(d^4)$, where d represents the maximum degree in the numerators and denominators of the parametrization.

The preceding ideas to detect symmetries are generalized in [11] and [14] for detecting the similarities between two rational planar and space curves, respectively, $\mathcal{C}_1$ and $\mathcal{C}_2$, properly parametrized by rational parametrizations $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. Because of the properness of the parametrizations, we also have a commutative diagram with a Möbius transformation φ at the bottom

$$\begin{tikzcd} \mathcal{C}_1 \arrow[r, "f"] & \mathcal{C}_2 \\ \mathbb{R} \arrow[u, dashed, "\boldsymbol{x}_1"] \arrow[r, dashed, "\varphi"'] & \mathbb{R} \arrow[u, dashed, "\boldsymbol{x}_2"'] \end{tikzcd} \tag{1.9}$$

where f is a similarity between the curves. The commutative diagram implies that

$$f \circ \boldsymbol{x}_1 = \boldsymbol{x}_2 \circ \varphi.$$

In [11], the parametrizations are treated as complex numbers depending on the parameter of the curves, and the above relationship is exploited to write all the parameters of the Möbius function in terms of just one of them. Then an algorithm using univariate gcds is obtained. The complexity of the algorithm is analyzed; this complexity is $\tilde{\mathcal{O}}(d^4)$, where $\tilde{\mathcal{O}}$ represents the complexity neglecting logarithmic factors, and d is the maximum degree of the numerators and denominators of the parametrizations.

In [14], similarities between space rational curves are addressed; the strategy also involves the curvature and torsion of the curves, although the special case of *helical* curves, i.e., curves where the quotient κ/τ is constant, arises as a special situation, also investigated in the paper. Furthermore, in both [11] and [14] the methods are generalized to Bézier curves, and to B-spline curves and NURBS curves with the same knot vectors and same weights. In fact, in these cases it suffices to compute the similarities between the corresponding control polygons.

However, the commutative diagram in Eq. (1.9) works not only for similarities, but in fact for any invertible transformation f. This observation is somehow present in the generalization done by Hauer et al. in [56] of the ideas in the preceding papers, to compute affine and projective equivalences between two properly parametrized rational curves $\mathcal{C}_1$ and $\mathcal{C}_2$ in any dimension. In [56], the authors consider projective parametrizations $\boldsymbol{x}_1, \boldsymbol{x}_2$ of $\mathcal{C}_1$ and $\mathcal{C}_2$, where

$$\boldsymbol{x}_1(t) = (p_0(t_0,t_1), p_1(t_0,t_1), \ldots, p_d(t_0,t_1)), \text{ with } p_i(\mathbf{t}) = \sum_{j=0}^{n} c_{j,i} t_0^{n-j} t_1^j$$

and

$$\boldsymbol{x}_2(t) = (p'_0(t_0,t_1), p'_1(t_0,t_1), \ldots, p'_d(t_0,t_1)), \text{ with } p'_i(\mathbf{t}) = \sum_{j=0}^{n} c'_{j,i} t_0^{n-j} t_1^j$$

and the p_i, p'_i are homogeneous polynomials of degree n (notice that p'_i does *not* represent here the first derivative of p_i). Denoting the coefficient vectors of both parametrizations by

$$\mathbf{c}_j = (c_{j,0}, c_{j,1}, \ldots, c_{j,d}), \ \mathbf{c}'_j = (c'_{j,0}, c'_{j,1}, \ldots, c'_{j,d}),$$

with $j = 0, \ldots, n$, the authors, first, explore the effect on a coefficient vector, say $\mathbf{c}_j$, of applying a Möbius transformation. Denoting the new coefficient vector by $\hat{\mathbf{c}}_j(\alpha)$, where α denotes the 4-tuple defined by the parameters of a Möbius transformation, it is shown that if $\mathcal{C}_1, \mathcal{C}_2$ are related by a projective transformation f defined by a non-singular matrix M, then

$$M\mathbf{c}'_j = \widehat{\mathbf{c}}_j(\alpha), \ j = 0, \ldots, n. \tag{1.10}$$

This provides a polynomial system in the entries of the matrix M, and the parameters of the Möbius transformation. However, the system is linear in the entries of M, which allows to solve for the entries of M in terms of α. One just needs some equations of the system to do this, so the remaining equations are used, after substituting the expressions of the entries of M in terms of α, to derive a polynomial system in α only. The paper [56] includes a detailed analysis of the polynomial system in Eq. (1.10), as well as three different strategies to solve it whose efficiency depends on the degree of the curves, and the dimension of the curves. The case of affine transformations is analyzed separately, since it has extra advantages.

We end this subsection with a different approach for computing projective equivalences between two rational curves in any dimension, provided in [33]. A first observation in [33] is that projective transformations between two sets of four points in the complex projective line $\mathbb{P}^1(\mathbb{C})$ preserve the cross-ratio. From here, a natural algorithm to detect whether two finite sets of points in the complex projective line are projectively equivalent is given. The algorithm is then extended to sets of points in the complex projective line which are given as roots of polynomials. For rational curves, it is observed that a rational curve in $\mathbb{P}^n(\mathbb{C})$ has finitely many *stall points*, which are the points where the osculating hyperplane has a contact of order higher than expected. If we consider a projective parametrization $\widehat{\boldsymbol{x}}(t,\omega)$ of the curve, with ω an homogenization variable, stall points are given by the condition

$$\Delta_{\widehat{\boldsymbol{x}}}(t,\omega) = \det\left[\frac{\partial^n \widehat{\boldsymbol{x}}(t,s)}{\partial t^n}, \frac{\partial^n \widehat{\boldsymbol{x}}(t,\omega)}{\partial t^{n-1}\omega}, \dots, \frac{\partial^n \widehat{\boldsymbol{x}}(t,\omega)}{\partial \omega^n}\right] = 0.$$

Stall points are always finite, and projective transformations map stalls to stalls. Thus, in order to compute the projective equivalences between two curves, one computes first the stall points of each curve, and then finds the projective equivalences between the (finite) sets of stall points.

1.4.2 The case of rational surfaces

The strategy in the previous section can be generalized to the case of rational surfaces up to a certain extent. However, the difficulty is higher, because while birational transformations of the complex line have a closed form, birational transformations of the complex plane do not. These transformations are called *Cremona transformations*, and are known to be generated by quadratic and linear projective transformations. Thus, in this case symmetries, similiarites, affine and projective transformations also lead to commutative diagrams like the ones we had in the preceding section, but in general we do not know how the function φ at the bottom of the diagram looks like.

Nevertheless, by introducing certain hypotheses on the surfaces, we can get partial results. A first step in this sense is the paper [9], where an algorithm for computing the involutions of a surface $S \subset \mathbb{R}^3$ polynomially and properly parametrized by $\boldsymbol{x}(t,s) = (x(t,s), y(t,s), z(t,s))$, with $(t,s) \in \mathbb{R}^2$, is given. However, it is also assumed that the parametrization is surjective, which is a strong hypothesis. Under these assumptions, we again have a commutative diagram

$$\begin{tikzcd} S \arrow[r, "f"] & S \\ \mathbb{R}^2 \arrow[u, dashed, "\boldsymbol{x}"] \arrow[r, dashed, "\varphi"'] & \mathbb{R}^2 \arrow[u, dashed, "\boldsymbol{x}"'] \end{tikzcd} \tag{1.11}$$

where φ is proved to be a linear affine mapping, i.e., $\varphi(t, s) = (at+bs+c_1, ct+ds+c_2)$, satisfying that $\varphi \circ \varphi = \mathrm{id}_{\mathbb{R}^2}$. This condition, together with the relationship $f \circ \boldsymbol{x} = \boldsymbol{x} \circ \varphi$, is exploited to write all the parameters of $\varphi(t, s)$ in terms of just two of them. In order to do this, the fact that the first fundamental form of the surface is pointwise preserved by an isometry is used. Thus, the method leads to bivariate polynomial systems, from which the involutions of the surface are derived. Symmetries, not only involutions, of algebraic surfaces of revolution are also addressed in [9].

A step forward is given in the paper [57], where an algorithm for computing projective and affine symmetries and equivalences between rational surfaces is provided. The paper requires the parametrizations to be proper and without projective base points, which is, again, a quite strong hypothesis. Under these assumptions and using Elimination Theory, the authors prove that the function φ at the bottom of the commutative diagram in Eq. (1.11) must be linear projective, i.e.,

$$\varphi(t, s) = \left(\frac{a_1 t + b_1 s + c_1}{a_3 t + b_3 s + c_3}, \frac{a_2 t + b_2 s + c_2}{a_3 t + b_3 s + c_3} \right).$$

Then the approach in [56] is generalized to the surface setting. Again, we have a relationship as in Eq. (1.10), and the method relies on polynomial system solving.

Some authors of [57] generalize their results to the case of rational surfaces with projective base points in [63]. In [63], the authors reduce the problem of computing projective equivalences between two rational surfaces to finding projective isomorphisms between surfaces that are covered by lines or conics, and that belong to five possible different types. However, computational or efficiency issues are not discussed in [63].

We finish with some references for equivalences between special types of rational surfaces.

- Projective equivalences between ruled rational surfaces are considered in [33], where the computation of projective equivalences between rational curves is also addressed (see the previous subsection). The main idea is that a ruled rational surface can be seen as a rational curve in the *Plücker quadric*, i.e., the Grassmanian $\mathbb{G}(1,3)$. Thus, the problem of computing projective equivalences between rational ruled surfaces can be translated to the problem of relating rational curves of a higher dimension, and higher degree.

- *Canal surfaces* are surfaces obtained as the envelope of a family of spheres whose centers lie on a curve, the *spine curve*, and whose radii are variable. The spine of a canal surface is unique except for *Dupin cyclides*, which can be generated in two different ways, i.e., with two different spine curves. The symmetries of rational canal surfaces, with rational spine and rational radius function are treated in [3]. In [3], symmetries of canal surfaces are reduced to the computation of the symmetries of the spine curve, when it is unique, or the isometries between the two spine curves, for the special case of Dupin cyclides, which are compatible with the invariance of the radius function. In fact, it is the radius function that provides a very fast algorithm, relying on factoring of bivariate polynomials, to detect the symmetries. Furthermore, in [3] a complete classification of the symmetries of Dupin cyclides, following from the general algorithm, is given.

- *Surfaces of translation*, or *translational surfaces*, are surfaces generated by sliding a space curve onto another one: if $\mathbf{a}(t)$ and $\mathbf{b}(s)$ represent two different parametrizations of space curves, then

$$\boldsymbol{x}(t,s) = \mathbf{a}(t) + \mathbf{b}(s)$$

  is the surface of translation generated by $\mathbf{a}(t)$ and $\mathbf{b}(s)$. Affine equivalences between rational surfaces of translation, with rational generators, are addressed in [16]. The main idea is that the computation of the affine equivalences between two rational surfaces of translation can be reduced to computing the affine equivalences between the generators of the surfaces, thus taking the problem from rational surfaces to rational curves. This can be applied to *minimal surfaces* as well, which can be seen as surfaces of translation. Minimal surfaces are the surfaces with zero mean curvature, and enjoy a very notable optimization property, namely that they span a given space curve with minimal area. Since Lie and Weierstrass it is known that a minimal surface is in fact a surface of translation with complex conjugate generators. Additionally, in [16] several properties of the symmetries of rational minimal surfaces are given, and the symmetries of a special family of minimal surfaces, *Enneper surfaces*, are computed. A method to create minimal surfaces with certain prescribed symmetries is also given.

1.4.3 Implicit case

Compared to the rational case, implicit curves and surfaces have been much less considered in the literature. In fact, the problem for implicit surfaces is still open. Nevertheless, we can mention some contributions.

Symmetries of plane algebraic curves are studied in [70]. Given a planar curve $\mathcal{C}$ implicitly defined by $f(x,y)=0$, the authors consider the complex substitution

$$(x,y) \to \left(\frac{(z+\overline{z})}{2}, \frac{(z-\overline{z})}{2\mathbf{i}}\right),$$

where $z = x + \mathbf{i}y$ and $\mathbf{i}^2 = -1$, obtaining the *implicit complex form* of the curve $F(z,\overline{z}) = 0$. This expression has a certain matrix form, and the properties of the matrix form of a curve admitting rotational or reflectional symmetries are analyzed. The analysis benefits from the fact that such symmetries have a simple expression when working over the complex numbers, and leads to an algorithm to detect the symmetries of the curve. The algorithm is fast for rotational symmetries around the origin and reflections in lines through the origin, and requires some more effort for general rotations and reflections. This matrix form is later recovered in [34] in order to address the case when the coefficients in the implicit equation of the curve are given with finite precision.

The implicit complex form is also used in [5] to detect whether two planar curves $\mathcal{C}_1$, $\mathcal{C}_2$, implicitly given by $f_1(x,y) = 0$, $f_2(x,y) = 0$, are similar. Since a planar similarity corresponds to an affine transformation of the complex plane $h(z) = \boldsymbol{a}z + \boldsymbol{b}$ or $h(z) = \boldsymbol{a}\overline{z} + \boldsymbol{b}$ (for orientation-reversing similarities), it is observed that certain parameters of the similarity can be derived by comparing the forms of highest degree of $f_1(x, y)$ and $f_2(x, y)$. After computing these parameters, in order to compute the similarities in general we need to solve a bivariate polynomial system with complex coefficients, although with the advantage of having a univariate equation in one of the variables. For the case of symmetries, one just needs to compute univariate gcds of polynomials with complex coefficients.

Symmetries and similarities between implicit planar algebraic curves are also studied in [15]. The crucial fact that is exploited here is that the Laplacian operator commutes with orthogonal transformations. Thus, if $f(x, y) = 0$ defines a curve $\mathcal{C}$, ϕ is a symmetry of the curve, and Δ denotes the Laplacian operator, then

$$\Delta(f \circ \phi) = \Delta f \circ \phi.$$

If Δf is not identically zero, this implies that any symmetry of f is also a symmetry of Δf, which is a polynomial of degree $\deg(f) - 2$. For a similarity ϕ with scaling constant λ,

$$\Delta(f \circ \phi) = \lambda^2(\Delta \circ \phi),$$

and we have a similar phenomenon. By repeateadly applying this, we get a *Laplacian chain* that ends with either a quadratic polynomial, or a linear polynomial, or a *harmonic* polynomial, i.e., a polynomial h such that $\Delta h = 0$. This reduces the problem of computing symmetries and similarities to conics and harmonic polynomials and some special cases, easy to treat. The most complicated part is the analysis for harmonic polynomials. For such a polynomial h, we consider the singular points of the vector field

$$\vec{v}(x, y) = (h_x, -h_y).$$

When written in complex form, $\vec{v}(x, y) = h_x - \mathbf{i}h_y$. Using Cauchy-Riemann conditions, it turns out that $\vec{v}$ is in fact a polynomial $g(z)$ in the complex variable z; the singular points of $\vec{v}$ are the zeroes of this polynomial in z. It is shown that the symmetries of

h are, in fact, symmetries of the set of singular points of $\vec{v}$, which can be efficiently computed and admit a closed form in terms of the coefficients of $g(z)$. For similarities, one checks that the similarities between two harmonic polynomials are also similarities between the singular point sets of the vector fields associated with the polynomials. The derived algorithms are very fast.

Finally, in [33] some hints on affine equivalences between implicit algebraic curves, and on symmetries and similarities of implicit algebraic surfaces are given. As in [5], it is observed that part of the information on symmetries and similarities can be computed by just analyzing the form of highest degree of an implicit curve $f(x, y) = 0$, or the form of highest degree of an implicit surface $f(x, y, z) = 0$. In the first case, the zeroes of the form of highest degree of $f(x, y)$ give finitely many points (the points at infinity) of the curve, so one can take advantage of the algorithms for computing symmetries, affine and projective equivalences of finite sets of points, which are also addressed in [33]. For surfaces, one intersects the surface defined by the form of highest degree $f_d(x, y, z)$ of the surface with the absolute conic $x^2 + y^2 + z^2 = 0$. This provides finitely many points (at infinity), and, again, one reduces the search of candidates for potential symmetries or similarities to this finite set of points.

CHAPTER 2

AFFINE EQUIVALENCES OF RULED RATIONAL SURFACES

This chapter describes an algorithm for computing the affine equivalences between two rational ruled surfaces. Isometries and symmetries are treated as special cases enjoying extra advantages. The ideas are also applied to computing certain types of symmetries of an implicit algebraic surface under additional hypotheses.

The main question is to reduce the problem, as it is also done in many works reviewed in the previous chapter, to computing the mapping corresponding to each affine equivalence in the parameter space, which is now two-dimensional. However, this problem is harder, because such mapping, which is a birational transformation of the plane, does not have, unlike birational transformation of the line, a simple and universal form. However, in this case we can guess and make precise the structure of this mapping, and from here recover the affine equivalences.

Furthermore, here we also take advantage of several ideas of [56], where projective mappings between rational curves in any dimension are studied. Indeed, when seen projectively, our problem has a certain correspondence with projective equivalences between certain curves, related to the directions of the rulings of the surfaces involved.

The problem we treat here has also been addressed by other authors in the last few years. We briefly provide a comparison with these works.

2.1 Rational ruled surfaces

Definition 2.1. *An algebraic surface $S \subset \mathbb{R}^3$ is* ruled *if for every point $p \in S$ there exists a line $\mathcal{L}_p$ through p, completely contained in S. The lines $\mathcal{L}_p$ are called the* rulings *of S. Furthermore, if $\mathcal{D} \subset S$ is a curve contained in S which intersects all the rulings, we say that $\mathcal{D}$ is a directrix of S.*

Here we will consider *rational* ruled surfaces, i.e., surfaces $S \subset \mathbb{R}^3$ admitting a rational parametrization $\boldsymbol{x}(t,s)$. We say that S is parametrized in *standard form* if

$$\boldsymbol{x}(t,s) = \boldsymbol{p}(t) + s\cdot \boldsymbol{q}(t), \tag{2.1}$$

where $\boldsymbol{p}(t)$ and $\boldsymbol{q}(t)$ are rational parametrizations. Notice that $\boldsymbol{p}(t)$ defines the directrix curve, while $\boldsymbol{q}(t)$ defines the direction of the ruling through each point $\boldsymbol{p}(t)$. In [87] it is proved that any rational ruled surface can be brought into standard form, although the parametrization might not be real (e.g., quadrics). An algorithm to do it is also provided in [87].

If all the rulings of S intersect at a point p_0, we say that S is *conical*, and that p_0 is the vertex of S. In this case, we have

$$\boldsymbol{x}(t,s) = p_0 + s\cdot \boldsymbol{q}(t). \tag{2.2}$$

If all the rulings of S are parallel to a vector $\bar{\mathbf{v}} \in \mathbb{R}^3$, i.e., they intersect at a point at infinity, we say that S is *cylindrical*. In this case, we have

$$\boldsymbol{x}(t,s) = \boldsymbol{p}(t) + s \cdot \bar{\mathbf{v}}. \tag{2.3}$$

2.2 Affine equivalences of ruled surfaces

2.2.1 Assumptions on the surfaces.

Let S_1, S_2 be real ruled surfaces, defined by parametrizations $\boldsymbol{x}_1, \boldsymbol{x}_2$, where

$$\boldsymbol{x}_i(t,s) = \boldsymbol{p}_i(t) + s \cdot \boldsymbol{q}_i(t), \tag{2.4}$$

and $\boldsymbol{p}_i(t), \boldsymbol{q}_i(t)$ are rational for $i = 1, 2$. In this subsection we will precise some additional hypotheses that we will require on the surfaces S_1, S_2.

First, we will assume that S_1, S_2 are not cylindrical. Cylindrical surfaces are addressed in Subsection 2.3.3. Thus, the $\boldsymbol{q}_i(t)$ are not multiples of constant vectors $\bar{\mathbf{v}}_i \in \mathbb{R}^3$.

Furthermore, we will also assume that no S_i is not doubly-ruled, i.e., that there are not two different families of rulings contained in S_i. It is well-known that the doubly-ruled surfaces are planes, hyperbolic paraboloids, and single-sheeted hyperboloids (see [59, §I.3]), so all of them are either planes or quadrics. For paraboloids and hyperboloids, one can study affine equivalences by first computing the implicit equation, which is easy to do for quadrics, and then applying matrix methods.

Additionally, we will suppose that for $i = 1, 2$, $\boldsymbol{x}_i(t,s)$ is proper in the sense of Sec. 1.3.

Finally, we will assume that each $\boldsymbol{q}_i(t)$ is polynomial. One can always achieve this. Indeed, observe first that if $\boldsymbol{q}_i$ is not polynomial, we can multiply it by $\mu_i(t) = \frac{\mu_{1,i}(t)}{\mu_{2,i}(t)}$, where $\mu_{1,i}(t)$ is the least common multiple of the denominators of the components of $\boldsymbol{q}_i(t)$, and $\mu_{2,i}(t)$ is the greatest common divisor of the numerators of the components of $\boldsymbol{q}_i(t)$. Now since $\mu_i(t)\boldsymbol{q}_i(t)$ is parallel to $\boldsymbol{q}_i(t)$ for all t, the parametrizations $\boldsymbol{p}_i(t)+s\boldsymbol{q}_i(t)$ and $\boldsymbol{p}_i(t)+s\cdot\mu_i(t)\boldsymbol{q}_i(t)$ define the same surface S_i, because both of them have the same rulings. In other words, when moving to $\boldsymbol{p}_i(t)+s\cdot\mu_i(t)\boldsymbol{q}_i(t)$ we are changing the norm of the vector parallel to each ruling of S_i, but not its direction.

However, one needs to check that this last assumption is compatible with the other assumptions; in particular, with the assumption on the properness of the parametrizations (the others are obvious). In order to see this, we set

$$\widehat{\boldsymbol{q}}_i(t) := \mu_i(t)\boldsymbol{q}_i(t),\ \ \mu_i(t) = \frac{\mu_{1,i}(t)}{\mu_{2,i}(t)}, \tag{2.5}$$

and

$$\widehat{s} := s/\mu_i(t). \tag{2.6}$$

This way, the parametrization

$$\widehat{\boldsymbol{x}}_i(t,s) = \boldsymbol{p}_i(t) + \widehat{s}\cdot\widehat{\boldsymbol{q}}_i(t),$$

defines the same surface S_i. The following lemma guarantees that this new parameterization is still proper.

Lemma 2.1. *Let S be a rational ruled surface parametrized by $\boldsymbol{x}(t,s)$ as in Eq. (2.4). If $\boldsymbol{x}(t,s)$ is proper, then $\widehat{\boldsymbol{x}}(t,s)$ is also proper.*

Proof. Assume that $\widehat{\boldsymbol{x}}(t,s)$ is not proper. Then a generic point $P_0 \in S$ is generated via $\widehat{\boldsymbol{x}}(t,s)$ by at least two distinct pairs (t_1,s_1) and (t_2,s_2). But then P_0 is reached via $\boldsymbol{x}(t,s)$ by $(t_1, s_1\cdot\mu(t_1))$ and $(t_2, s_2\cdot\mu(t_2))$ since

$$P_0 = \widehat{\boldsymbol{x}}(t_1,s_1) = \boldsymbol{p}(t_1) + s_1\cdot\mu(t_1)\boldsymbol{q}(t_1) = \boldsymbol{x}(t_1,\mu(t_1)\cdot s_1),$$

and

$$P_0 = \widehat{\boldsymbol{x}}(t_2,s_2) = \boldsymbol{p}(t_2) + s_2\cdot\mu(t_2)\boldsymbol{q}(t_2) = \boldsymbol{x}(t_2,\mu(t_2)\cdot s_2).$$

From the two previous equations, since $\boldsymbol{x}(t,s)$ is proper, we get that $t_1 = t_2$ and $\mu(t_1)\cdot s_1 = \mu(t_2)\cdot s_2$. But since $t_1 = t_2$, and $\mu(t)$ is not identically zero, we deduce that $s_1 = s_2$. Thus, the pairs (t_1,s_1) and (t_2,s_2) are equal, contradicting our initial hypothesis that they were distinct.

■

2.2.2 Main theorems

Let S_1, S_2 be real rational ruled surfaces parametrized by $\boldsymbol{x}_1, \boldsymbol{x}_2$ as in Eq. (2.4), satisfying that: (1) $\boldsymbol{x}_i(t,s)$ is proper; (2) $\boldsymbol{q}_i(t)$ is polynomial with relatively prime components; (3) S_i is not doubly-ruled; (4) S_i is not cylindrical. Our goal in this subsection is to develop some results that will lead to a method to detect whether S_1, S_2 are affinely equivalent, and in the affirmative case to compute the affine equivalences between S_1, S_2.

The following result is crucial for us.

Theorem 2.1. *Let S_1, S_2 be two rational real ruled surfaces properly parametrized by $\boldsymbol{x}_1, \boldsymbol{x}_2$ as in Eq. (2.4). A mapping $f : \mathbb{R}^3 \to \mathbb{R}^3$, $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x}+\boldsymbol{b}$, with $\boldsymbol{A} \in \mathbb{R}^{3\times 3}$, $\boldsymbol{b} \in \mathbb{R}^3$ and $\boldsymbol{A}$ nonsingular, satisfies $f(S_1) = S_2$, so that S_1, S_2 are affinely equivalent, if and only if there exists a birational transformation $\varphi : \mathbb{R}^2 \to \mathbb{R}^2$, such that the diagram*

$$\begin{tikzcd} S_1 \arrow[r, "f"] & S_2 \\ \mathbb{R}^2 \arrow[u, dashed, "\boldsymbol{x}_1"] \arrow[r, dashed, "\varphi"'] & \mathbb{R}^2 \arrow[u, dashed, "\boldsymbol{x}_2"'] \end{tikzcd} \tag{2.7}$$

is commutative. In particular, for a generic point $(t,s) \in \mathbb{R}^2$ we have

$$f \circ \boldsymbol{x}_1 = \boldsymbol{x}_2 \circ \varphi. \tag{2.8}$$

Proof. ($\Rightarrow$) Since $\boldsymbol{x}_2$ is proper by hypothesis, $\boldsymbol{x}_2^{-1}$ exists and is rational. Therefore, $\varphi = \boldsymbol{x}_2^{-1} \circ f \circ \boldsymbol{x}_1$ is birational, because φ is the composite of birational transformations.

($\Leftarrow$) Since $f \circ \boldsymbol{x}_1 = \boldsymbol{x}_2 \circ \varphi$, whenever $\boldsymbol{x}_1(t,s)$ and $(\boldsymbol{x}_2 \circ \varphi)(t,s)$ are well-defined $(f \circ \boldsymbol{x}_1)(t,s) \in S_2$, so $f(S_1) \subset S_2$. Since f is nonsingular, $f(S_1)$ defines a rational surface, i.e., $f(S_1)$ does not degenerate into a curve. Additionally $f(S_1), S_2$ are both rational, and therefore irreducible; since $f(S_1) \subset S_2$ and $f(S_1), S_2$ are irreducible, $f(S_1) = S_2$, i.e., S_1, S_2 are affinely equivalent. ∎

Additionally, from Eq. (2.8) one can easily see that each affine mapping f is associated with a different φ. Indeed, let f_1, f_2 be two different affine equivalences between S_1 and S_2, and let φ_1, φ_2 be the birational planar transformations associated with f_1, f_2 according to Eq. (2.7). Since the diagram is commutative, we have $\varphi_i = \boldsymbol{x}_2^{-1} \circ f_i \circ \boldsymbol{x}_1^{-1}$. Thus, $f_1 \neq f_2$ implies that $\varphi_1 \neq \varphi_2$.

Remark 2.1. *It can happen that at a base point only one of the sides of Eq.* (2.8) *is defined. For instance, let S be the conical surface parametrized by*

$$\boldsymbol{x}(t,s) = s\cdot(-t^4 - 6t^2 + 3, 8t^3, (t^2+1)^2).$$

One can check that this surface is invariant by a rotation of $\frac{2\pi}{3}$ degrees about the z-axis given by $f(x,y,z) = \boldsymbol{A}\cdot(x,y,z)$, i.e., $f(S) = S$, where

$$\boldsymbol{A} = \begin{pmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Furthermore, one can also check that in this case $f \circ \boldsymbol{x} = \boldsymbol{x} \circ \varphi$, with

$$\varphi(t,s) = \left(\frac{-\sqrt{3}t - 3}{3t - \sqrt{3}}, \frac{9}{16} s \left(t - \frac{\sqrt{3}}{3} \right)^4 \right), \tag{2.9}$$

holds for a generic point (t,s) of the parameter space. Now take $(t_0, s_0) = \left(\frac{\sqrt{3}}{3}, 1\right)$, so $\boldsymbol{x}(t_0,s_0) = \left(\frac{8}{9}, \frac{8\sqrt{3}}{9}, \frac{16}{9}\right)$ and

$$(f \circ \boldsymbol{x})(t_0,s_0) = f\left(\frac{8}{9}, \frac{8\sqrt{3}}{9}, \frac{16}{9}\right) = \left(-\frac{16}{9}, 0, \frac{16}{9}\right).$$

In particular, the left hand-side of Eq. (2.8) *is well defined. However, for $(t_0, s_0) =$*

$(\frac{\sqrt{3}}{3}, 1)$ *the denominator of the first component of* $\varphi(t_0, s_0)$ *is zero, and* $(\boldsymbol{x} \circ \varphi)(t_0, s_0)$ *is not defined. One can check that in this case, the symmetry* f *maps the point* $(\frac{8}{9}, \frac{8\sqrt{3}}{9}, \frac{16}{9})$*, generated by* (t_0, s_0)*, to the point* $(-\frac{16}{9}, 0, \frac{16}{9})$*, which is a point of* S *missed by the parametrization* $\boldsymbol{x}$*, i.e., not generated by* $\boldsymbol{x}$ *for any pair* (t, s)*.*

If we consider the situation in a projective setup, we observe that Eq. (2.8) *fails at the projective point corresponding to* $(t_0, s_0) = (\frac{\sqrt{3}}{3}, 1)$*. Indeed, let us represent by* $[t : s : \omega]$ *the elements of the parameter space, which is now* $\mathbb{P}^2(\mathbb{R})$ *(the last coordinate corresponds to the homogenization variable). Then the point* $[\frac{\sqrt{3}}{3} : 1 : 1]$ *of the parameter space, corresponding to the affine point* $(t_0, s_0) = (\frac{\sqrt{3}}{3}, 1)$*, maps to* $[576 : 0 : 0]$*, which is a base point of*

$$\widehat{\boldsymbol{x}}(t, s, \omega) = [s(-t^4 - 6t^2\omega^2 + 3\omega^4) : 8st^3\omega : s(t^2 + \omega^2)^2 : \omega^5],$$

the parametrization of the projective closure $\widehat{S}$ *of* S*. Thus, the left hand-side of Eq.* (2.8) *is* $[-\frac{16}{9} : 0 : \frac{16}{9} : 1]$*, while the right hand-side of Eq.* (2.8) *is* $[0 : 0 : 0 : 0]$*.*

From Theorem 2.1 we observe that φ is a birational transformation of the plane, i.e., a Cremona transformation. Since Cremona transformations do not have a generic closed form, in order to describe φ we need to make use of the properties of the surfaces we are investigating; in this case, of the fact that they are ruled. The following result provides a first clue in this direction.

Proposition 2.1. *Let* S_1, S_2 *be rational ruled surfaces properly parametrized as in Eq.* (2.4)*, which are not doubly ruled. Let* $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ *be a nonsingular affine mapping satisfying* $f(S_1) = S_2$*, and let* $\varphi : \mathbb{R}^2 \to \mathbb{R}^2$ *be the birational transformation making the diagram in Eq.* (2.7) *commutative. Then*

$$\varphi(t, s) = (\psi(t), a(t) \cdot s + c(t)), \tag{2.10}$$

where $\psi(t)$ *is a Möbius transformation and* $a(t), c(t)$ *are rational functions.*

Proof. Since f is an affine mapping, f maps rulings of S_1 onto rulings of S_2. Let $\varphi(t, s) = (\varphi_1(t, s), \varphi_2(t, s))$. A generic ruling of S_i, with $i = 1, 2$ is defined by $\boldsymbol{x}_i(t_{a_i}, s)$, where t_{a_i} is a constant. Since S_2 is not doubly ruled, the ruling parametrized by $\boldsymbol{x}_1(t_{a_1}, s)$ is mapped by f onto the ruling parametrized by $\boldsymbol{x}_2(t_{a_2}, s)$. Using Eq. (2.8), we get

$$f(\boldsymbol{x}_1(t_{a_1}, s)) = \boldsymbol{x}_2(\varphi(t_{a_1}, s)) = \boldsymbol{x}_2(\varphi_1(t_{a_1}, s), \varphi_2(t_{a_1}, s)),$$

so $\varphi_1(t_{a_1}, s) = t_{a_2}$, i.e., $\varphi_1(t_{a_1}, s)$ does not depend on s. Since this independence happens for a generic t_{a_1}, we deduce that $\varphi_1(t, s) = \varphi_1(t)$. Since φ is birational, φ_1 is birational as well; in particular, we deduce that φ_1 is a birational transformation of the line, so φ_1 must be a Möbius transformation, which we represent by $\psi(t)$. From Eq. (2.8), taking into account that $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ we have

$$\boldsymbol{A}\cdot \boldsymbol{p}_1(t) + \boldsymbol{b} + s\cdot \boldsymbol{A}\cdot \boldsymbol{q}_1(t) = \boldsymbol{p}_2(\psi(t)) + \varphi_2(t, s)\cdot \boldsymbol{q}_2(\psi(t)).$$

Writing this expression in components, we get $\varphi_2(t, s) = a(t)\cdot s + c(t)$, for $a(t), c(t)$ rational.

■

Let us now investigate the structure of the function $a(t)$ in Eq. (2.10). Recall that $\boldsymbol{x}_i(t, s) = \boldsymbol{p}_i(t) + s\cdot \boldsymbol{q}_i(t)$, where $\boldsymbol{q}_i(t) = (q_{i,1}(t), q_{i,2}(t), q_{i,3}(t))$, each $q_{i,j}(t)$ is polynomial and $\gcd(q_{i,1}, q_{i,2}, q_{i,3}) = 1$. Also, let

$$n_i = \max\{\deg(q_{i,1}(t)), \deg(q_{i,2}(t)), \deg(q_{i,3}(t))\}, \tag{2.11}$$

and let us write

$$a(t) = \frac{A(t)}{B(t)},\ \psi(t) = \frac{\alpha t + \beta}{\gamma t + \delta},$$

where $A, B \in \mathbb{R}[t]$, $\gcd(A, B) = 1$, and $\alpha\delta - \beta\gamma \neq 0$. Combining Eq. (2.10) and Eq. (2.8) with $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$, we have

$$\begin{aligned}\boldsymbol{A}\cdot \boldsymbol{x}_1(t,s)+\boldsymbol{b} &= \boldsymbol{x}_2(\varphi(t,s))\\ \boldsymbol{A}\cdot \boldsymbol{p}_1(t)+s\cdot \boldsymbol{A}\cdot \boldsymbol{q}_1(t)+\boldsymbol{b} &= \boldsymbol{p}_2(\psi(t))+(a(t)\cdot s+c(t))\cdot \boldsymbol{q}_2(\psi(t))\\ &= \boldsymbol{p}_2(\psi(t))+s\cdot a(t)\boldsymbol{q}_2(\psi(t))+c(t)\cdot \boldsymbol{q}_2(\psi(t)).\end{aligned}$$

Comparing the coefficients of s, we get

$$\boldsymbol{A}\cdot \boldsymbol{p}_1(t)+\boldsymbol{b} = \boldsymbol{p}_2(\psi(t))+c(t)\cdot \boldsymbol{q}_2(\psi(t)), \tag{2.12}$$

and

$$\boldsymbol{A}\cdot \boldsymbol{q}_1(t) = a(t)\cdot \boldsymbol{q}_2(\psi(t)). \tag{2.13}$$

Since $\boldsymbol{q}_i(t)$, $i=1,2$, is polynomial, the left hand-side of Eq. (2.13) is polynomial as well, so the right hand-side of Eq. (2.13) must also be polynomial. This observation yields the following results; here, we denote the entries of the matrix $\boldsymbol{A}$ by $\boldsymbol{A}_{ij}$.

Lemma 2.2. *$(\gamma t+\delta)^{n_2}$ divides $A(t)$.*

Proof. From Eq. (2.13), for $i=1,2,3$ we get

$$\boldsymbol{A}_{i1}\cdot q_{1,1}(t)+\boldsymbol{A}_{i2}\cdot q_{1,2}(t)+\boldsymbol{A}_{i3}\cdot q_{1,3}(t) = a(t)\cdot q_{2,i}(\psi(t)), \tag{2.14}$$

where $q_{2,i}(t)=a_{\ell_i}t^{\ell_i}+a_{\ell_i-1}t^{\ell_i-1}+\cdots+a_0$, with $\ell_i\le n_2$ for $i\in\{1,2,3\}$. Furthermore, $\ell_i=n_2$ for at least one $i\in\{1,2,3\}$. Additionally,

$$q_{2,i}(\psi(t)) = \frac{a_{\ell_i}(\alpha t+\beta)^{\ell_i}+a_{\ell_i-1}(\alpha t+\beta)^{\ell_i-1}(\gamma t+\delta)+\cdots+a_0(\gamma t+\delta)^{\ell_i}}{(\gamma t+\delta)^{\ell_i}}. \tag{2.15}$$

Since $\gamma t+\delta$ does not divide $\alpha t+\beta$, the numerator and denominator of $q_{2,i}(\psi(t))$ are relatively prime. Since the left hand-side of Eq. (2.14) is a polynomial, $a(t)\cdot q_{2,i}(\psi(t))$ must be a polynomial as well, so $(\gamma t+\delta)^{\ell_i}$ divides $A(t)$. Since $\ell_i=n_2$ for some

$i \in \{1, 2, 3\}$, the statement follows. ■

Lemma 2.3. *$B(t)$ is a constant.*

Proof. Recall that $\gcd(q_{2,1}, q_{2,2}, q_{2,3}) = 1$. Let $N_i(t)$ be the numerator of $q_{2,i}(\psi(t))$. Since the left hand-side of Eq. (2.14) is a polynomial, $B(t)|N_i(t)$ for $i = 1, 2, 3$. Thus, $B(t)|G(t)$, where $G = \gcd(N_1, N_2, N_3)$. Now suppose that $G(t)$ is not constant. Then N_1, N_2, N_3 have a common root t_0. Moreover, since the numerators and denominators of the $q_{2,i}(\psi(t))$ are relative prime, $\gamma t_0 + \delta \neq 0$. Therefore, $\psi(t_0)$ is well defined and $\psi(t_0)$ is a common root of the $q_{2,i}(t)$, because $q_{2,i}(\psi(t_0)) = \frac{N_i(t_0)}{(\gamma t_0+\delta)^{\ell_i}}$. But this contradicts the fact that $\gcd(q_{2,1}, q_{2,2}, q_{2,3}) = 1$. Thus, $G(t)$ is constant and since $B(t)|G(t)$, $B(t)$ must be a constant. ■

Finally, we get the following proposition about the form of the function $a(t)$.

Proposition 2.2. *The function $a(t)$ satisfies $a(t) = k\cdot (\gamma t + \delta)^{n_2}$, where k is a nonzero constant.*

Proof. From the two previous lemmas we have $a(t) = k(t)\cdot (\gamma t + \delta)^{n_2}$ for some polynomial $k(t)$. Additionally, from Eq. (2.13) and lemma 2.2,

$$\boldsymbol{A}\cdot \boldsymbol{q}_1(t) = k(t)\cdot (\gamma t + \delta)^{n_2}\cdot \boldsymbol{q}_2(\psi(t)). \tag{2.16}$$

Taking Eq. (2.15) into account, we observe that $(\gamma t + \delta)^{n_2}\cdot \boldsymbol{q}_2(\psi(t))$ is polynomial. If $k(t)$ is not a constant, then the components of $\boldsymbol{A} \cdot \boldsymbol{q}_1(t)$ are not relatively prime, i.e., $\boldsymbol{A} \cdot \boldsymbol{q}_1(t) = r(t)\boldsymbol{q}_1^\star(t)$, with $r(t)$ nonconstant, and $\boldsymbol{q}_1^\star(t)$ a polynomial parametrization with relatively prime components. However, since $\boldsymbol{A}$ is nonsingular, in that case we have $\boldsymbol{q}_1(t) = r(t)\boldsymbol{A}^{-1}\boldsymbol{q}_1^\star(t)$, which implies that the components of $\boldsymbol{q}_1(t)$ are not relatively prime either. Since by hypothesis the components of $\boldsymbol{q}_1(t)$ are relatively prime, $k(t)$ must be a constant k. Finally, since $\boldsymbol{A}$ is nonsingular, from Eq. (2.16) we get that $k \neq 0$. ■

Taking Proposition 2.2 and Eq. (2.16) into account, we get the following corollary.

Corollary 2.1. *If S_1, S_2 are affinely equivalent, then $n_1 = n_2$.*

We summarize the previous results in the following theorem. In the rest of the chapter, we denote, according to Corollary 2.1, $n_1 = n_2 = n$.

Theorem 2.2. *Let S_1, S_2 be two rational ruled surfaces, which are not doubly ruled, properly parametrized as in Eq.* (2.4)*. Let $\boldsymbol{q}_i(t) = (q_{i,1}(t), q_{i,2}(t), q_{i,3}(t))$, with $q_{i,j}(t) \in \mathbb{R}[t]$ for $i = 1, 2$ and $j = 1, 2, 3$, and $n_1 = n_2 = n$. Let $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$, with $\boldsymbol{A}$ nonsingular, such that $f(S_1) = S_2$, and let $\varphi : \mathbb{R}^2 \to \mathbb{R}^2$ be the birational transformation making the diagram in Eq.* (2.7) *commutative. Then*

$$\varphi(t, s) = (\psi(t), k\cdot (\gamma t + \delta)^n \cdot s + c(t)), \tag{2.17}$$

where $\psi(t)$ is a Möbius transformation, k is a constant, and $c(t)$ is a rational function. Moreover,

$$\boldsymbol{A}\cdot \boldsymbol{q}_1(t) = k\cdot (\gamma t + \delta)^n \cdot \boldsymbol{q}_2(\psi(t)). \tag{2.18}$$

Eq. (2.18) can be interpreted in geometric terms. In order to do this, it is preferable to write Eq. (2.18) projectively. Let $\tilde{\boldsymbol{q}}_i(t, \omega) = [q_{i,1}(t, \omega) : q_{i,2}(t, \omega) : q_{i,3}(t, \omega)] \in \mathbb{P}^2(\mathbb{R})$, where $i = 1, 2$ and ω is a homogenization variable. Then Eq. (2.18) can be written as

$$\boldsymbol{A} \cdot \tilde{\boldsymbol{q}}_1(t, \omega) = k \cdot \tilde{\boldsymbol{q}}_2(\alpha t + \beta\omega, \gamma t + \delta\omega). \tag{2.19}$$

This means that the projective curves defined by $\tilde{\boldsymbol{q}}_1(t, \omega)$ and $\tilde{\boldsymbol{q}}_2(t, \omega)$ are projectively equivalent, and even more, that $\boldsymbol{A}$ defines a projectivity mapping the projective curve defined by $\tilde{\boldsymbol{q}}_1(t, \omega)$ onto the projective curve defined by $\tilde{\boldsymbol{q}}_2(t, \omega)$ (or $k \cdot \tilde{\boldsymbol{q}}_2(t, \omega)$, since projectively $\tilde{\boldsymbol{q}}_2(t, \omega)$ and $k \cdot \tilde{\boldsymbol{q}}_2(t, \omega)$ can be identified). This observation makes perfect sense from a geometric point of view: affine equivalences map rulings of S_1 onto rulings of S_2, as observed in the proof of Proposition 2.1, and $\tilde{\boldsymbol{q}}_1(t, \omega)$, $\tilde{\boldsymbol{q}}_2(t, \omega)$ define the *directions* of these rulings. The matrix $\boldsymbol{A}$ defines the map sending the direction of each ruling of S_1 onto the direction of a ruling of S_2.

Projective equivalences between curves in any dimension and particular systems of equations like Eq. (2.19) (and therefore Eq. (2.18)) are studied in great detail in

[56]. We will benefit from the study carried out in [56] in the next section, where we address the computation of the affine equivalences between S_1, S_2.

We will see how to exploit Eq. (2.18) and Eq. (2.12) in the next section.

2.3 Computation of the affine equivalences

The computation of the affine equivalences between S_1, S_2 is based on the following result, which in turn follows from the results of the previous section.

Proposition 2.3. *The affine equivalences* $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ *between* S_1, S_2 *correspond to the* $\boldsymbol{A} \in \mathbb{R}^{3\times 3}$, $\boldsymbol{b} \in \mathbb{R}^3$ *satisfying* Eq. (2.18) *and* Eq. (2.12), *where* $\det(\boldsymbol{A}) \neq 0$, $k \neq 0$, $\psi(t) = \frac{\alpha t+\beta}{\gamma t+\delta}$ *and* $\alpha\delta - \beta\gamma \neq 0$.

Notice that since the components of $\boldsymbol{q}_2(t)$ are polynomials of degree at most n, Eq. (2.18) involves only polynomials, and provides equations which are linear in the entries $\boldsymbol{A}_{ij}$ of the matrix $\boldsymbol{A}$. Furthermore, the coefficients of the $\boldsymbol{A}_{ij}$ in these linear equations are constants, while the constant terms of these linear equations depend on $\alpha, \beta, \gamma, \delta$ and k. However, Eq. (2.12) involves rational functions, i.e., polynomial denominators.

2.3.1 Reducing the numbers of coefficients of φ

Since $\alpha\delta - \beta\gamma \neq 0$ we can always either assume that $\alpha\delta - \beta\gamma = 1$, or separate the analysis in two different cases, namely the case $\gamma = 1$, and the case $\gamma = 0$, $\delta = 1$. This last possibility allows us to perform the computation with fewer variables (although twice).

More precisely, if n is odd, or n is even and $k \geq 0$, we have that

$$k\cdot(\gamma t + \delta)^n = [\sqrt[n]{k}(\gamma t + \delta)]^n = (\sqrt[n]{k}\gamma t + \sqrt[n]{k}\delta)^n.$$

On the other hand,

$$\frac{\alpha t + \beta}{\gamma t + \delta} = \frac{\sqrt[n]{k}\alpha t + \sqrt[n]{k}\beta}{\sqrt[n]{k}\gamma t + \sqrt[n]{k}\delta}.$$

Now, if n is even and $k < 0$, then

$$k\cdot (\gamma t + \delta)^n = -|k|(\gamma t + \delta)^n = -[\sqrt[n]{|k|}(\gamma t + \delta)]^n = -(\sqrt[n]{|k|}\gamma t + \sqrt[n]{|k|}\delta)^n,$$

and

$$\frac{\alpha t + \beta}{\gamma t + \delta} = \frac{\sqrt[n]{|k|}\alpha t + \sqrt[n]{|k|}\beta}{\sqrt[n]{|k|}\gamma t + \sqrt[n]{|k|}\delta}.$$

So renaming the coefficients, we finally have that

$$\varphi(t, s) = \left(\frac{\alpha t + \beta}{\gamma t + \delta}, \pm s\cdot (\gamma t + \delta)^n + c(t)\right). \tag{2.20}$$

In both cases we still have $\alpha\delta - \gamma\beta \neq 0$ since $\sqrt[n]{k^2}\alpha\delta - \sqrt[n]{k^2}\gamma\beta = \sqrt[n]{k^2}(\alpha\delta - \gamma\beta) \neq 0$. We might take $a(t)$ as $k\cdot (\gamma t + \delta)^n$ or as $\pm(\gamma t + \delta)^n$. In our implementations we observed that it is more efficient to use the expression involving the constant k. Now if we take $a(t) = k\cdot (\gamma t + \delta)^n$, we can leave out one of the variables α, β, γ or δ, let us take γ. We consider two cases here:

- If $\gamma = 0$, then we can set $\delta = 1$, and after renaming the coefficients, we have

$$\varphi(t, s) = (\alpha t + \beta, s\cdot k + c(t)). \tag{2.21}$$

- If $\gamma \neq 0$, then

$$\varphi(t, s) = \left(\frac{\alpha t + \beta}{\gamma t + \delta}, s\cdot k\cdot (\gamma t + \delta)^n + c(t)\right) = \left(\frac{\frac{\alpha}{\gamma}t + \frac{\beta}{\gamma}}{t + \frac{\delta}{\gamma}}, s\cdot k\cdot \gamma^n \left(t + \frac{\delta}{\gamma}\right)^n + c(t)\right).$$

Again, renaming the coefficients, we have

$$\varphi(t,s) = \left(\frac{\alpha t+\beta}{t+\delta}, s\cdot k\cdot (t+\delta)^n + c(t)\right). \tag{2.22}$$

The computation of the affine equivalences proceeds in three different steps, (A), (B), (C). Let us describe these steps in detail.

Step (A): ***Writing A in terms of*** $\alpha, \beta, \gamma, \delta$***, and*** k**.**

At this step we exploit Eq. (2.18), which has been studied in great detail in Section 3 of [56]. Writing Eq. (2.18) in components, we get

$$\left\{\begin{array}{lcl} \boldsymbol{A}_{11}\cdot q_{1,1}(t) + \boldsymbol{A}_{12}\cdot q_{1,2}(t) + \boldsymbol{A}_{13}\cdot q_{1,3}(t) & = & k(\gamma t+\delta)^n q_{2,1}(\psi(t)), \\ \boldsymbol{A}_{21}\cdot q_{1,1}(t) + \boldsymbol{A}_{22}\cdot q_{1,2}(t) + \boldsymbol{A}_{23}\cdot q_{1,3}(t) & = & k(\gamma t+\delta)^n q_{2,2}(\psi(t)), \\ \boldsymbol{A}_{31}\cdot q_{1,1}(t) + \boldsymbol{A}_{32}\cdot q_{1,2}(t) + \boldsymbol{A}_{33}\cdot q_{1,3}(t) & = & k(\gamma t+\delta)^n q_{2,3}(\psi(t)). \end{array}\right. \tag{2.23}$$

Since the $q_{2,j}(t)$ have degree at most n, the expressions on the right hand-side of Eq. (2.23) are, in fact, polynomials. Equating the coefficients of t^ℓ, for $\ell = 0, 1, \ldots, n$, on both sides of Eq. (2.23), we get a system $\mathcal{L}$, linear in the $\boldsymbol{A}_{ij}$, where the coefficients of the $\boldsymbol{A}_{ij}$ are constant numbers, and where the constant terms are polynomials in $\alpha, \beta, \gamma, \delta$ and k. Let us write $\boldsymbol{q}_1(t)$ as

$$\boldsymbol{q}_1(t) = \mathbf{v}_0 + \mathbf{v}_1 t + \cdots + \mathbf{v}_n t^n, \tag{2.24}$$

where $\mathbf{v}_\ell \in \mathbb{R}^3$, for $\ell = 0, \ldots, n$, is a numeric row vector whose components are the coefficients in t^ℓ of $q_{1,1}(t), q_{1,2}(t)$ and $q_{1,3}(t)$. Then the system $\mathcal{L}$ has the form:

$$
\underbrace{\begin{bmatrix}
\mathbf{v}_0 & & \\
\vdots & & \\
\mathbf{v}_n & & \\
& \mathbf{v}_0 & \\
& \vdots & \\
& \mathbf{v}_n & \\
& & \mathbf{v}_0 \\
& & \vdots \\
& & \mathbf{v}_n
\end{bmatrix}}_{\mathcal{A}}
\cdot
\begin{bmatrix}
\boldsymbol{A}_{11} \\
\boldsymbol{A}_{12} \\
\boldsymbol{A}_{13} \\
\boldsymbol{A}_{21} \\
\vdots \\
\boldsymbol{A}_{33}
\end{bmatrix}
=
\begin{bmatrix}
\bullet_1 \\
\vdots \\
\bullet_{n+1} \\
\vdots \\
\bullet_{2(n+1)} \\
\vdots \\
\bullet_{3(n+1)}
\end{bmatrix}
\tag{2.25}
$$

Here we see that $\mathcal{A} \in \mathbb{R}^{3(n+1)\times 9}$ is a block matrix with three nonzero blocks of size $(n+1)\times 3$, consisting of the row vectors $\mathbf{v}_0, \ldots, \mathbf{v}_n$. The constant terms $\bullet_j$, where $j = 1, \ldots, 3(n+1)$, are products of k by a homogeneous polynomial in $\alpha, \beta, \gamma, \delta$ of degree n, a structure observed in Section 3.2 of [56]. Notice also that the number $3(n+1)$ of the equations is in agreement with the observations raised in Section 3 of [56] (compare to Table 2 in Section 3 of [56], taking into account that we are dealing with projective curves, defined by $\tilde{\boldsymbol{q}}_1, \tilde{\boldsymbol{q}}_2$, in the projective plane).

Let $r = \operatorname{rank}(\mathbf{v}_0, \ldots, \mathbf{v}_n)$; notice that since $\mathbf{v}_\ell \in \mathbb{R}^3$, we get $r \leq 3$. Furthermore, if $r = 2$ then the directions of all the rulings of S_1 are parallel to a plane. If $r = 1$ then all the rulings of S_1 are parallel to a same vector $\mathbf{v}$, i.e., S_1 is a cylindrical surface; this special case is much easier to solve, see Subsection 2.3.3.

Now by the structure of the matrix $\mathcal{A}$ we get $\operatorname{rank}(\mathcal{A}) = 3r$. Let us address the cases $r = 3$ and $r = 2$. The case $r = 3$ is analyzed in detail in Section 3.2 of [56]; here we adapt several results of [56] to our case. However, the case $r = 2$ is, apparently, not addressed in [56].

(1) **Case** $r = 3$: since $\operatorname{rank}(\mathcal{A}) = 3r$, for $r = 3$ we get $\operatorname{rank}(\mathcal{A}) = 9$, so we can solve the system $\mathcal{L}$ and write the $\boldsymbol{A}_{ij}$ in terms of $\alpha, \beta, \gamma, \delta$ and k. Additionally, applying the Gauss-Jordan method to the system $\mathcal{L}$, we get $3(n+1) - 3r$ additional

conditions on $\alpha, \beta, \gamma, \delta$ and k that must hold for $\mathcal{L}$ to be consistent. When $r = 3$, we get $3n - 6$ conditions of this type. Each such condition is a product of k by a homogeneous polynomial in $\alpha, \beta, \gamma, \delta$. Since $k \neq 0$, we can factor out k and get $3n-6$ homogeneous conditions on $\alpha, \beta, \gamma, \delta$ alone, of degree n. Since $\alpha\delta - \beta\gamma \neq 0$, one can add the extra condition $\alpha\delta - \beta\gamma = 1$.

This way we get a polynomial system $P_{\mathcal{A}}$ in $\alpha, \beta, \gamma, \delta$: if this polynomial system is not consistent, the surfaces S_1, S_2 are identified as non-affinely equivalent, and the computation stops. Otherwise we can get either tentative values for $\alpha, \beta, \gamma, \delta$ that may or may not give rise to an affine equivalence between S_1, S_2 (this must be tested later), or a number of relations between the $\alpha, \beta, \gamma, \delta$. If these relations allow writing some of these parameters in terms of the others, we can reduce the number of parameters in the subsequent computations.

Notice that when $n = 2$, we get $3n - 6 = 3 \cdot 2 - 6 = 0$, so no extra conditions in $\alpha, \beta, \gamma, \delta$ are generated. However, we can still write the $\boldsymbol{A}_{ij}$ in terms of $\alpha, \beta, \gamma, \delta$ and k.

(2) **Case** $r = 2$: in this case, since $r = 2$ applying the Gauss-Jordan method to the system $\mathcal{L}$ we get $3(n + 1) - 3 \cdot 2 = 3n - 3$ additional conditions on $\alpha, \beta, \gamma, \delta, k$ that must hold for $\mathcal{L}$ to be consistent, with the same properties as in the previous case. As before, we denote the collection of all these polynomial conditions by $P_{\mathcal{A}}$. However, since $\mathrm{rank}(\mathcal{A}) = 6$ is less than the number of $\boldsymbol{A}_{ij}$, we cannot, in this case, write all the $\boldsymbol{A}_{ij}$ only in terms of $\alpha, \beta, \gamma, \delta, k$, i.e., three of the $\boldsymbol{A}_{ij}$ must be considered as parameters as well. This observation makes sense from a geometric point of view: if $r = 2$ then $\boldsymbol{q}_1(t), \boldsymbol{q}_2(t)$ parametrize projective lines, and there are infinitely many projective transformations mapping a projective line onto another projective line.

Observe that when the components of $\boldsymbol{q}_1(t)$ are linear we are always either in the case $r = 1$, or in the case $r = 2$. In this last case, we do not get any extra conditions on $\alpha, \beta, \gamma, \delta$, because since $n = 1$ the number $3n - 3$ of extra conditions vanishes.

Summarizing, at this step we write either all the $\boldsymbol{A}_{ij}$, when $r = 3$, or only six of the

$\boldsymbol{A}_{ij}$, when $r=2$, in terms of $\alpha,\beta,\gamma,\delta,k$. Furthermore, except in the case $r=3,n=2$ and the case $n=1$, we get polynomial conditions on $\alpha,\beta,\gamma,\delta$, which may help either to detect that the surfaces are not affine equivalent (when these conditions are not compatible), or to reduce the number of parameters.

(B) ***Writing b in terms of*** $\alpha,\beta,\gamma,\delta$***, and*** k***, and computing*** $c(t)$***.***

Writing Eq. (2.12) in components, we get

$$\left\{\begin{array}{lcl} \boldsymbol{A}_{11}\cdot p_{1,1}(t)+\boldsymbol{A}_{12}\cdot p_{1,2}(t)+\boldsymbol{A}_{13}\cdot p_{1,3}(t)+b_1 & = & p_{2,1}(\psi(t))+c(t)q_{2,1}(\psi(t)),\\ \boldsymbol{A}_{21}\cdot p_{1,1}(t)+\boldsymbol{A}_{22}\cdot p_{1,2}(t)+\boldsymbol{A}_{23}\cdot p_{1,3}(t)+b_2 & = & p_{2,2}(\psi(t))+c(t)q_{2,2}(\psi(t)),\\ \boldsymbol{A}_{31}\cdot p_{1,1}(t)+\boldsymbol{A}_{32}\cdot p_{1,2}(t)+\boldsymbol{A}_{33}\cdot p_{1,3}(t)+b_3 & = & p_{2,3}(\psi(t))+c(t)q_{2,3}(\psi(t)), \end{array}\right. \tag{2.26}$$

where we assume that the $\boldsymbol{A}_{ij}$, or some of the $\boldsymbol{A}_{ij}$, have already been written in terms of $\alpha,\beta,\gamma,\delta,k$. Now we proceed as follows:

(i) Eliminating $c(t)$ between the first and second equations of Eq. (2.26) provides an equation E_1 linear in b_1,b_2, with coefficients that are rational functions of t.

(ii) Proceeding in the same way with the second and third equations, we get an equation E_2, linear in b_2,b_3.

(iii) Evaluating E_1 and E_2 at several random t-values we get a linear system in b_1,b_2,b_3, whose solution provides $\boldsymbol{b}$ in terms of k.

(iv) Finally, we compute $c(t)$ from any equation of Eq. (2.26).

We will refer later to this procedure as "the steps (i)-(iv)".

(C) ***Deriving a polynomial system*** $\mathcal{S}$***, and computing the affine equivalences.***

Substituting the expressions for $\boldsymbol{A},\boldsymbol{b}$ and $c(t)$ computed in steps (A) and (B) into Eq. (2.8), we get a polynomial system $\mathcal{S}$. If $r=3$, the unknowns of $\mathcal{S}$ are, at most, $k,\alpha,\beta,\gamma,\delta$, and we can have fewer unknowns if the polynomial conditions $P_{\mathcal{A}}$ in step (A) allow us to write some of these variables in terms of the others. If $r=2$, we can have at most three more unknowns besides $k, \alpha, \beta, \gamma, \delta$, namely three of the $\boldsymbol{A}_{ij}$. Again, the polynomial system $P_{\mathcal{A}}$ may help reduce the total number of parameters, and therefore of unknowns in $\mathcal{S}$. Thus, the number of unknowns in $\mathcal{S}$ is ≤ 5, if $r = 3$, and ≤ 8, if $r = 2$.

The solutions of this polynomial system provide the affine equivalences between S_1, S_2. We summarize the whole procedure to find the affine equivalences between S_1, S_2 in Algorithm `Affine-Eq-Ruled`.

Algorithm 1 Affine-Eq-Ruled

Require: Two ruled surfaces S_1, S_2, properly parametrized by $\boldsymbol{x}_i(t,s) = \boldsymbol{p}_i(t) + s\boldsymbol{q}_i(t)$, $i = 1, 2$, where each $\boldsymbol{q}_i(t)$ is polynomial with relatively prime components of degree $\leq n$.

Ensure: The affine equivalences $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ between S_1, S_2.

1: Compute the system $\mathcal{L}$ in Eq. (2.25).
2: Apply the Gauss-Jordan method on the system $\mathcal{L}$.
3: **if** $r = 3$ and $n \geq 3$, or $r = 2$ and $n \geq 2$ **then**
4:    solve the polynomial system $P_{\mathcal{A}}$ in $\alpha, \beta, \gamma, \delta$.
5:    **if** $P_{\mathcal{A}}$ is not consistent **then**
6:       **return** S_1 `and` S_2 `are not affinely equivalent`, and **stop**
7:    **end if**
8: **end if**
9: Solve the system $\mathcal{L}$
10: Write the solutions of $\mathcal{L}$ with as few variables as possible, using, if any, the solutions of $P_{\mathcal{A}}$
11: Follow steps (i)-(iv) to write $\boldsymbol{b}$ in terms of the variables in step 10, and to compute $c(t)$
12: Substitute $\boldsymbol{A}, \boldsymbol{b}, c(t)$ and the φ in Eq. (2.17) into Eq. (2.8)
13: Derive from the preceding substitution a polynomial system $\mathcal{S}$ in the parameters appearing in step 9
14: **if** no solution is found **then**
15:    **return** S_1 `and` S_2 `are not affinely equivalent.`
16: **else**
17:    **for** each solution found **do**
18:       compute the corresponding mapping $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$
19:    **end for**
20: **end if**

Example 2.1. Let S_1 and S_2 be the rational ruled surfaces parametrized by $\boldsymbol{x_1}(t,s) = \boldsymbol{p_1}(t) + s \cdot \boldsymbol{q_1}(t)$ and $\boldsymbol{x_2}(t,s) = \boldsymbol{p_2}(t) + s \cdot \boldsymbol{q_2}(t)$, where

$$\begin{aligned}
&\boldsymbol{p}_1(t) = (t^4+t^2+t, t^6+t^3, t^5+t^3+t^2+3t),\\
&\boldsymbol{q}_1(t) = (t^3+t, t^5, t^4+t^2+3),\\
&\boldsymbol{p}_2(t) = (5t^4+5t^2+5t-1, 3t^5+3t^3+3t^2+9t+5, -t^6+t^4-t^3+t^2+t),\\
&\boldsymbol{q}_2(t) = (5t^3+5t, 3t^4+3t^2+9, -t^5+t^3+t)\,.
\end{aligned}$$

In this case, $n=5$. Furthermore, when we write $\boldsymbol{q}_1(t)$ as in Eq. (2.24), we observe that we fall in the case $r=3$. The surfaces S_1, S_2 are shown in Fig. 2.1.

In order to check whether S_1, S_2 are affinely equivalent, we apply `Algorithm 1`. The polynomial system $P_{\mathcal{A}}$ in this case is given by:

- $10\alpha^3\delta\gamma - 15\alpha^2\beta\delta^2 + 15\alpha^2\beta\gamma^2 - 30\alpha\beta^2\delta\gamma - 20\alpha\delta^3\gamma + 20\alpha\delta\gamma^3 - 5\beta^3\gamma^2 - 30\beta\delta^2\gamma^2 + 5\beta\gamma^4 = 0.$
- $3\alpha^4\delta + 12\alpha^3\beta\gamma - 18\alpha^2\beta^2\delta - 3\alpha^2\delta^3 + 9\alpha^2\delta\gamma^2 - 12\alpha\beta^3\gamma - 18\alpha\beta\delta^2\gamma + 6\alpha\beta\gamma^3 - 9\beta^2\delta\gamma^2 - 90\delta^3\gamma^2 + 45\delta\gamma^4 = 0.$
- $-5\alpha^4\beta + 2\alpha^3\delta\gamma + 10\alpha^2\beta^3 - 3\alpha^2\beta\delta^2 + 3\alpha^2\beta\gamma^2 - 6\alpha\beta^2\delta\gamma - 4\alpha\delta^3\gamma + 4\alpha\delta\gamma^3 - \beta^3\gamma^2 - 6\beta\delta^2\gamma^2 + \beta\gamma^4 = 0.$

Since $\alpha\delta - \beta\gamma \neq 0$, we add the equation $(\alpha\delta-\beta\gamma)u - 1 = 0$. Using these equations together with the $3\cdot(5+1) = 18$ equations of the system $\mathcal{L}$ we obtain expressions for $\alpha, \beta, \gamma, \delta$ depending only on k; the same thing happens with $\boldsymbol{b}$ and $c(t)$.

In particular, going back to Eq. 2.16 we get two sets of possible expressions for the entries $\boldsymbol{A}_{ij}$ in terms of k, namely

$$\begin{aligned}
&\mathrm{Sol}_1 = \{\boldsymbol{A}_{11} = 5k, \boldsymbol{A}_{12} = \boldsymbol{A}_{13} = \boldsymbol{A}_{21} = \boldsymbol{A}_{22} = 0, \boldsymbol{A}_{23} = 3k, \boldsymbol{A}_{31} = k, \boldsymbol{A}_{32} = -k, \boldsymbol{A}_{33} = 0\},\\
&\mathrm{Sol}_2 = \{\boldsymbol{A}_{11} = -5k, \boldsymbol{A}_{12} = \boldsymbol{A}_{13} = \boldsymbol{A}_{21} = \boldsymbol{A}_{22} = 0, \boldsymbol{A}_{23} = 3k, \boldsymbol{A}_{31} = -k, \boldsymbol{A}_{32} = k, \boldsymbol{A}_{33} = 0\}.
\end{aligned}$$

From Sol_1, applying step (B) we get

$$E_1 = \frac{3b_1t^4 - 5b_2t^3 + 3t^4 + 3b_1t^2 + 25t^3 - 5b_2t + 45kt + 3t^2 + 9b_1 - 20t + 9}{15t(t^2+1)(t^4+t^2+3)},$$
$$E_2 = \frac{b_2t^5 - 3kt^5 + 3b_3t^4 - 2t^5 - b_2t^3 - 9kt^3 + 3b_3t^2 + 14t^3 - b_2t + 9kt + 9b_3 - 4t}{3t(t^4+t^2+3)(t^4-t^2-1)}.$$

Evaluating E_1 and E_2 in $t = - \pm 1, 3$ we get

$$\boldsymbol{b}_1 = \left(-1, \frac{1}{2} + \frac{9}{2}k, \frac{3}{62} - \frac{3}{62}k\right)^T,$$
$$c_1(t) = \frac{62kt^6 - 62t^6 - 62kt^4 + 62kt^3 + 62t^4 - 62kt^2 - 62t^3 - 62kt + 62t^2 + 3k + 62t - 3}{62t(t^4-t^2-1)}.$$

Replaicing the entries of $\boldsymbol{A}, \boldsymbol{b}_1$ and $c_1(t)$ into Eq. (2.8), with φ as in Eq. (2.17), we get a system $\mathcal{S}$ given by

$$\mathcal{S} = \{-15k+15, -310k+310, -27k+27, 279k-279, 93k-93, -837k+837, -9k+9\},$$

whose solution is $k = 1$. Here $\varphi_1(t,s) = (t,s)$.

From Sol_2 we have

$$E_1 = -\frac{3b_1t^4 + 5b_2t^3 + 3t^4 + 3b_1t^2 - 25t^3 + 5b_2t - 45kt + 3t^2 + 9b_1 + 20t + 9}{15t(t^2+1)(t^4+t^2+3)},$$
$$E_2 = \frac{b_2t^5 - 3kt^5 - 3b_3t^4 - 2t^5 - b_2t^3 - 9kt^3 - 3b_3t^2 + 14t^3 - b_2t + 9kt - 9b_3 - 4t}{3t(t^4+t^2+3)(t^4-t^2-1)}.$$

Using again $t = \pm 1, 3$ we get the same expression for vector $\boldsymbol{b}$ as above, and

$$c_2(t) = \frac{62kt^6 + 62t^6 - 62kt^4 + 62kt^3 - 62t^4 - 62kt^2 - 62t^3 - 62kt - 62t^2 + 3k + 62t - 3}{62t(t^4 - t^2 - 1)}.$$

Here we get the same linear system $\mathcal{S}$ in k. Hence, in this case we also obtain $k = 1$ but this time with $\varphi(t, s) = (-t, s + 2t)$.

In conclusion, we get two φ's corresponding to affine equivalences, namely

$$\varphi_1(t, s) = (t, s), \; \varphi_2(t, s) = (-t, s + 2t).$$

The mapping $\varphi_1(t, s)$ corresponds to the affine mapping $f_1(\mathbf{x}) = \boldsymbol{A}_1\mathbf{x} + \boldsymbol{b}_1$, where

$$\boldsymbol{A}_1 = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 0 & 3 \\ 1 & -1 & 0 \end{pmatrix}, \quad \boldsymbol{b}_1 = \begin{pmatrix} -1 & 5 & 0 \end{pmatrix}^T. \tag{2.27}$$

The mapping $\varphi_2(t, s)$ corresponds to the affine mapping $f_2(\mathbf{x}) = \boldsymbol{A}_2\mathbf{x} + \boldsymbol{b}_2$, where

$$\boldsymbol{A}_2 = \begin{pmatrix} -5 & 0 & 0 \\ 0 & 0 & 3 \\ -1 & 1 & 0 \end{pmatrix}, \quad \boldsymbol{b}_2 = \begin{pmatrix} -1 & 5 & 0 \end{pmatrix}^T. \tag{2.28}$$

Therefore, S_1, S_2 are related by two affine mappings f_1, f_2. Notice that this result is consistent with the fact that S_1 has a non-trivial symmetry (axial with respect to the z-axis); in fact, one can check that $f_2 = f_1 \circ f_0$, where f_0 represents the axial symmetry of S_1.

2.3.2 The special case of conical surfaces

Recall that S is a conical surface if all the rulings of S intersect at a point $\mathbf{p}_0 \in S$, called the vertex, which can be computed by using the results in [7]. By applying a

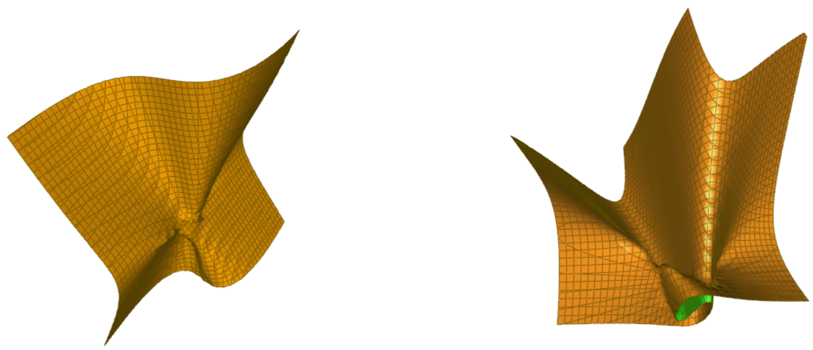


Figure 2.1: S_1 (left) and S_2 (right).

translation if necessary, we can always assume that $\mathbf{p}_0$ is the origin. In this case, S is parametrized by $\boldsymbol{x}(t,s) = s \cdot \boldsymbol{q}(t)$, where $\boldsymbol{q}(t)$ is polynomial.

Now given two rational conical surfaces S_1, S_2 parametrized by $\boldsymbol{x}_i(t,s) = s \cdot \boldsymbol{q}_i(t)$, with $\boldsymbol{q}_i(t)$ polynomial for $i = 1, 2$, any affine equivalence between S_1, S_2 has the form $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$, so $\boldsymbol{b} = \mathbf{0}$. Since $p_1(t), p_2(t)$ are identically zero and $f \circ \boldsymbol{x} = \boldsymbol{x} \circ \varphi$, we have

$$\boldsymbol{A} \cdot s \cdot \boldsymbol{q}_1(t) = [a(t) \cdot s + c(t)] \cdot \boldsymbol{q}_2(\psi(t)) = s \cdot a(t) \cdot \boldsymbol{q}_2(\psi(t)) + c(t) \cdot \boldsymbol{q}_2(\psi(t)).$$

Thus, $c(t) \cdot \boldsymbol{q}_2(\psi(t))$ must be equal to zero, and then the function $c(t)$ is identically zero as well.

Therefore Eq. (2.12) is reduced to $0 = 0$. Thus, the computation of the affine equivalences between S_1, S_2 reduces to solving Eq. (2.18). Notice that the system derived from Eq. (2.18) is homogeneous in k and the entries of the matrix $\boldsymbol{A}$, which implies that $\boldsymbol{A}$ is defined only up to a multiplicative constant. This observation makes perfect sense since any conical surface is invariant by homotheties where the homothety center is the vertex of the surface.

We summarize all this in the proposition below.

Proposition 2.4. *Let S_1, S_2 be rational conical surfaces, not doubly ruled, properly parametrized by $\boldsymbol{x}_i(t,s) = s \cdot \boldsymbol{q}_i(t)$. Let $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$ be a nonsingular affine mapping satisfying $f(S_1) = S_2$, and let $\varphi : \mathbb{R}^2 \to \mathbb{R}^2$ be the birational transformation making the diagram in Eq.* (2.7) *commutative. Then*

$$\varphi(t) = (\psi(t), s \cdot k \cdot (\gamma t + \delta)^n), \tag{2.29}$$

where $\psi(t)$ is a Möbius transformation.

2.3.3 The special case of cylindrical surfaces

Under the assumption that $\boldsymbol{q}_1(t), \boldsymbol{q}_2(t)$ are polynomials with relatively prime components, S_1, S_2 are cylindrical iff the $\boldsymbol{q}_i(t)$ are constant vectors. These vectors define the directions of all the rulings of S_1, S_2. Then in order to check whether S_1, S_2 are affinely equivalent, it suffices to check whether the planar curves $\mathcal{C}_1, \mathcal{C}_2$, obtained by intersecting S_1, S_2 with planes Π_1, Π_2 respectively normal to $\boldsymbol{q}_1(t), \boldsymbol{q}_2(t)$, are affinely equivalent. This can be done, for instance, by using the algorithm in [56]. Notice that the affine equivalences of S_1, S_2 are, in this case, the affine equivalences of the plane sections followed by any translations along the direction of the rulings of S_2, and any dilatation in the same direction.

2.3.4 Computing isometries and symmetries

Let us address now the case when the affine mapping $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ is an isometry, that is, $\boldsymbol{A}$ is an orthogonal matrix. For finding the isometries between S_1, S_2 we can certainly apply Algorithm `Affine-Eq-Ruled`, with the extra condition that $\boldsymbol{A}$ is orthogonal. However, in this case, we have additional conditions, which may be an advantage for simplifying the computation. Indeed, since orthogonal mappings preserve norms, taking norms in Eq. (2.13), with k a constant, we get the condition

$$\|\boldsymbol{q}_1(t)\|^2 - k^2 \cdot (\gamma t + \delta)^{2n} \cdot \|\boldsymbol{q}_2(\psi(t))\|^2 = 0. \tag{2.30}$$

Equating to zero all the coefficients in t at the left hand-side of Eq. (2.30), we get a polynomial system $\mathcal{P}$ of $2n+1$ equations, each one consisting of a homogeneous polynomial of degree $2n$ in the variables $\alpha, \beta, \gamma, \delta$ multiplied by k^2, plus a constant. These equations have a higher degree than the equations of the polynomial system $P_{\mathcal{A}}$, all of degree n. However, collecting the equations in $P_{\mathcal{A}}$ and $\mathcal{P}$ provides a bigger polynomial system in $\alpha, \beta, \gamma, \delta, k$, which may help to reduce the total number of parameters in the polynomial system $\mathcal{S}$, and/or the number of tentative values for $\alpha, \beta, \gamma, \delta, k$. In particular, in the cases $r = 3, n = 2$ and $n = 1$ applying Algorithm `Affine-Eq-Ruled` does not provide extra conditions on $\alpha, \beta, \gamma, \delta, k$; however, Eq. (2.30) does.

If $S_1 = S_2 = S$, the isometries leaving S invariant are the symmetries of S. We can find the symmetries of S by proceeding as before with $S_1 = S_2$. However, recall from Section 2.1 that certain notable symmetries, like central symmetries, axial symmetries and reflections in a plane, are affine involutions, i.e., affine mappings f satisfying $f \circ f = \mathrm{id}_{\mathbb{R}^3}$. If we are interested only in affine involutions (isometric or non-isometric) we can improve the computation as follows. First, from Eq. (2.8) one can see that $f \circ f = \mathrm{id}_{\mathbb{R}^3}$ iff the corresponding φ satisfies $\varphi \circ \varphi = \mathrm{id}_{\mathbb{R}^2}$. By Theorem 2.2, one has

$$\varphi(t,s) = (\varphi_1(t,s), \varphi_2(t,s)) = (\psi(t), s \cdot k(\gamma t + \delta)^n + c(t)),$$

and the condition $(\varphi \circ \varphi)(t,s) = (t,s)$ introduces two constraints:

(i) $(\varphi_1 \circ \varphi_1)(t,s) = t$, i.e., $(\psi \circ \psi)(t) = t$. In turn, this constraint implies that

$$\alpha^2 - \delta^2 = 0,\ \beta(\alpha + \delta) = 0,\ \gamma(\alpha + \delta) = 0.$$

Therefore, either $\alpha = -\delta$, or $\alpha + \delta \neq 0$ and $\alpha = \delta$, $\beta = \gamma = 0$.

(ii) $\varphi_2(\varphi_1(t), \varphi_2(t,s)) = s$, which implies

$$[s \cdot k(\gamma t + \delta)^n + c(t)] \cdot k \cdot \left[\gamma \cdot \frac{\alpha t + \beta}{\gamma t + \delta} + \delta\right]^n + c(\psi(t)) = s.$$

Comparing coefficients of s, we deduce that

$$k^2 \cdot \left[\gamma(\alpha + \delta)t + (\gamma\beta + \delta^2)\right]^n = 1,$$

which in turn yields

$$\gamma(\alpha + \delta) = 0,\ k^2(\gamma\beta + \delta^2)^n = 1.$$

Thus, either $\alpha = -\delta$ and $k^2(\gamma\beta + \delta^2)^n = 1$, or $\alpha = \delta$, $\gamma = 0$ and $k^2\delta^{2n} = 1$.

Putting (i) and (ii) together, we get the following result, which allows decreasing the total number of parameters, and therefore of unknowns in the polynomial system $\mathcal{S}$. Notice that this result is applicable to any affine involution (in particular, isometric involutions).

Theorem 2.3. *Let S be a rational ruled surface, which is not doubly ruled, properly parametrized as in Eq. (2.4). Let $\boldsymbol{q}(t) = (q_1(t), q_2(t), q_3(t))$, with $q_i(t) \in \mathbb{R}[t]$ for $i = 1, 2, 3$, and*

$$n = max\{\deg(q_1(t)), \deg(q_2(t)), \deg(q_3(t))\}.$$

Finally, let $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$, with $\boldsymbol{A} \in \mathbb{R}^3$, $\boldsymbol{b} \in \mathbb{R}^3$, be an affine involution leaving S invariant. With the notation of Theorem 2.2, one has:

(I) $\alpha = -\delta$ and $k^2(\gamma\beta + \delta^2)^n = 1$, in which case

$$\varphi(t,s) = \left(\frac{\alpha t + \beta}{t - \alpha}, \pm s \cdot \frac{(t-\alpha)^2}{\sqrt{(\beta + \alpha^2)^n}} + c(t)\right),$$

or

(II) $\varphi(t,s) = (t, -s + c(t))$, with $c(t)$ a rational function.

So we need to deal with a bivariate system at most.

Observe that in case (II) f fixes each line of the ruling and acts on these lines as an affine involution.

Remark 2.2. Since any similarity can be written as $f(\mathbf{x}) = \lambda \boldsymbol{Q} + \boldsymbol{b}$, where $\lambda \neq 0$ is the scaling constant, taking norms in Eq. (2.13), with k a constant, we get the condition

$$\lambda^2 \|\boldsymbol{q}_1(t)\|^2 - k^2 \cdot (\gamma t + \delta)^{2n_2} \cdot \|\boldsymbol{q}_2(\psi(t))\|^2 = 0. \tag{2.31}$$

The analysis, in this case, is very similar to that of isometries, although the polynomial system has one more variable, namely λ.

Example 2.2. In this example we computed all the symmetries of the ruled surface S (shown in Fig. 2.2) parametrized by

$$\boldsymbol{x}(t,s) = \left(\frac{2t^8 - 10t^6 - 10t^4 + 5t^2 + 1}{t^2+1}, -\frac{t^9 - 6t^7 + 6t^3 + t^2 - 3t + 1}{t^2+1}, t^7 + 3t^5 + 3t^3 + t + 5\right)$$
$$+ s \cdot (2t^5 - 12t^3 + 2t, (-t^2+1)(t^4 - 6t^2 + 1), (t^2+1)^3).$$

- The function $\varphi(t,s) = (-t,-s)$ corresponds to the symmetry with respect to the axis $[t+2, -1, 5]$,
- the function $\varphi(t,s) = \left(\frac{1}{t}, -st^6 - \frac{t^8+1}{t}\right)$ corresponds to the symmetry with respect to axis $[2, t-1, 5]$,
- the function $\varphi(t,s) = \left(-\frac{1}{t}, st^6 + \frac{t^8+1}{t}\right)$ corresponds to the symmetry with respect to the axis $[2, -1, t+5]$,
- the function

  $$\varphi(t,s) = \left(\frac{t-1}{t+1}, -\frac{s(t+1)^6}{8} - \frac{1}{8}\frac{t^8 + 7t^7 + 21t^6 + 35t^5 + 35t^4 + 21t^3 + 7t^2 + 9t - 8}{t+1}\right)$$

  corresponds to a rotation of $\frac{\pi}{2}$ with respect to the axis $[2, -1, t+5]$ plus a reflection on the plane $[t+2, s-1, 5]$,

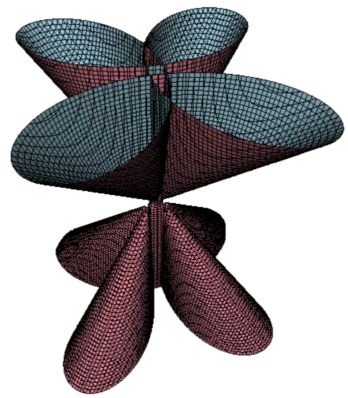


Figure 2.2: Surface S.

- the function

$$\varphi(t,s)=\left(\frac{-t+1}{t+1},\frac{s(t+1)^6}{8}+\frac{1}{8}\frac{t^8+7t^7+21t^6+35t^5+35t^4+21t^3+7t^2+9t-8}{t+1}\right)$$

corresponds to a symmetry with respect to the plane $[t+2,-t-1,s+5]$,

- and the function

$$\varphi(t,s)=\left(\frac{t+1}{t-1},\frac{s(t-1)^6}{8}+\frac{1}{8}\frac{t^8-7t^7+21t^6-35t^5+35t^4-21t^3+7t^2-9t-8}{t-1}\right)$$

corresponds to a symmetry with respect to the plane $[t+2,t-1,s+5]$.

Example 2.3. Let S_1 and S_2 be the rational ruled surfaces parametrized by $\boldsymbol{x_1}(t,s)=\boldsymbol{p_1}(t)+s\cdot\boldsymbol{q_1}(t)$ and $\boldsymbol{x_2}(t,s)=\boldsymbol{p_2}(t)+s\cdot\boldsymbol{q_2}(t)$, where

$$\boldsymbol{p}_1(t)=\left(t+\frac{3}{4},4t^2+3,t\right),$$
$$\boldsymbol{q}_1(t)=(t^3+2t^2+1,-t^3+t^2+t,-t^3+t^2+t),$$
$$\boldsymbol{p}_2(t)=\left(\frac{(\sqrt{3}+1)t}{2}+\frac{3\sqrt{3}}{8}-\frac{1}{2},4t^2+5,\frac{(\sqrt{3}-1)t}{2}-\frac{\sqrt{3}}{2}-\frac{3}{8}\right),$$

and

$$\boldsymbol{q}_2(t) = \left(\frac{(\sqrt{3}-1)t^3}{2} + \left(\sqrt{3}+\frac{1}{2}\right)t^2 + \frac{t}{2} + \frac{\sqrt{3}}{2}, -t^3+t^2+t, \right.$$
$$\left. -\left(\frac{\sqrt{3}+1}{2}\right)t^3 + \left(\frac{\sqrt{3}}{2}-1\right)t^2 + \frac{\sqrt{3}}{2}t - \frac{1}{2} \right),$$

Here, $n = 3$. Furthermore, when we write $\boldsymbol{q}_1(t)$ as in Eq. (2.24), we observe that we are in the case $r = 2$.

In this case, we analyze the isometries mapping S_1 onto S_2. There is only one isometry, associated with $\varphi(t, s) = (t, s)$, defined by $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$, where

$$\boldsymbol{A} = \begin{pmatrix} \frac{\sqrt{3}}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{\sqrt{3}}{2} \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} -\frac{1}{2} & 2 & -\frac{\sqrt{3}}{2} \end{pmatrix}^T, \tag{2.32}$$

corresponding to a rotation of $\frac{\pi}{6}$ around the y-axis. Applying Algorithm 1 with the additional equations corresponding to Eq. (2.30), we need to test only two tentative solutions. If, instead of Eq. (2.30), we use the orthogonality conditions on the columns of the matrix $\boldsymbol{A}$, we need to test four tentative solutions, and the computation time is higher. The surface S_1 is shown in Fig. 2.3.

2.4 Experimentation and performance of the method

We have implemented the methods described in Section 2.3 in the computer algebra system Maple 18, and we have tried the examples in an Intel(R) Core(TM) i5-7500, CPU 3.40 GHz and 32 Gb RAM. We have analyzed affine equivalences, isometries, and symmetries. For isometries, we used the conditions derived from Eq. (2.30), because this tends to speed up the computation.

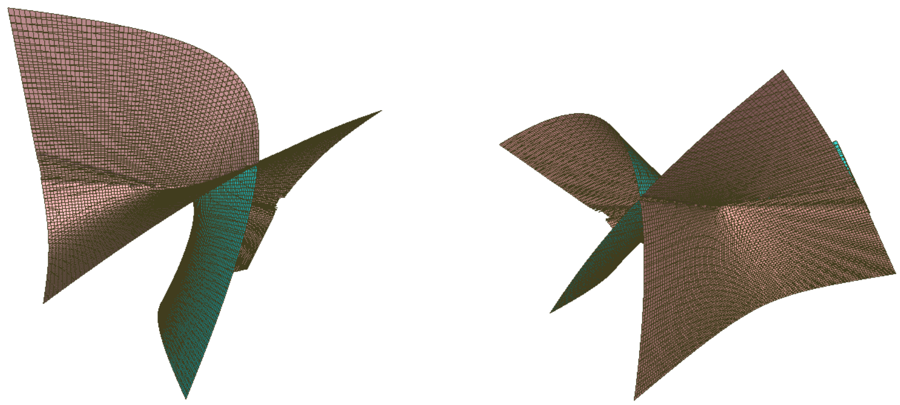


Figure 2.3: Surface S_1 from two points of view.

The results for affine equivalences of some representative examples are summarized in Tables 2.1 and 2.2. When the surfaces are affinely equivalent, the second surface is the result of applying to the first surface an affine equivalence with matrix

$$\begin{pmatrix} -1/2 & -1 & 0 \\ 0 & 1 & 1 \\ 0 & 2 & 3 \end{pmatrix}.$$

For each example, we have included: (1) a picture of the surface S_1, (2) the degree (deg.) of the parametrization, i.e., the maximum power of t appearing in the numerators and denominators of $\boldsymbol{p}(t), \boldsymbol{q}(t)$, (3) the computation time (in CPU seconds) of the method for all the affine mappings, and the computation time using the implicit equation of the surfaces (in red), (4) the number of affine equivalences between the two surfaces and (5) the parametrizations of both S_1, S_2.

The examples with more than one affine equivalence correspond to surfaces with symmetries. Furthermore, in some cases, we identify infinitely many equivalences, implying that the surfaces are invariant under infinitely many affine mappings. In the column of timings, we highlight in red the worst time between our method and the naive method mentioned in the Introduction using the implicit equation. This last timing does not include the time for computing the implicit equation, i.e., we assume that the implicit equation is already known. Only in one of the examples shown, where the implicit equation is straightforward ($F(x, y, z) = x^3 - 27yz^2$), and for the case of symmetries, is the method using the implicit equation faster.

The results for symmetries and isometries for several representative examples are summarized in Tables 2.3 and 2.4: for each example, we include data also included in the affine equivalences table, plus the computation time (in CPU seconds) of our method for computing all the symmetries of the surface ("all sym."), for computing only the involutive symmetries of the surface ("involutions"), and for computing the isometries ("isometries") between each surface and its image under an orthogonal transformation with associated matrix

$$\begin{pmatrix} 0 & 1 & 0 \\ 4/5 & 0 & -3/5 \\ 3/5 & 0 & 4/5 \end{pmatrix}. \tag{2.33}$$

In tables 2.3 and 2.4 only the parametrization of surface S_1 is shown. We have observed that almost all the time is spent solving the polynomial system $\mathcal{P}$, arising from Eq. (2.30). We used the Maple instruction `solve` to find the solutions of this system. The complexity of the method is dominated by the solution of the polynomial system $\mathcal{P}$. We have also observed that $\mathcal{P}$ is usually zero-dimensional. A recent, polynomial, bound for solving a zero-dimensional polynomial system is given in [37]. Although the case when $\mathcal{P}$ is not zero-dimensional is much less frequent, it can happen as well, for instance, when the components of $\boldsymbol{q}(t)$ are linear. In this case, up to our knowledge, there is no algorithm other than Gröbner bases to solve the problem; the best-known complexity, in this case, is exponential [26].

Compared to our approach, computing the implicit equation of the surface and applying the naive method mentioned in the Introduction to the paper provides worse timings, even if the time to compute the implicit equation is not considered. In fact, in many of the examples, Maple cannot find the solution of the polynomial system derived from the naive method in a reasonable amount of time.

We also include the type of symmetries found. In some cases, the symmetries detected are composites of rotations and reflections, denoted as “rotation+reflection”.

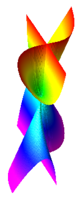
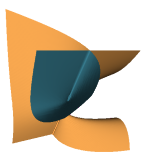
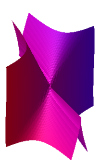
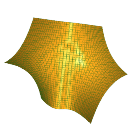
Picture of S_1	deg.	computation time (secs.)/imp.	Affine equivalences
	5	1.453 / > 90	2
$\tilde{\boldsymbol{x}}_1(t,s) = \left(\frac{t^2}{t^2+1}, \frac{t^4}{t^2+1}, \frac{t^5}{t^2+1}\right) + s\cdot(t, t^3, 1)$ $\tilde{\boldsymbol{y}}_1(t,s) = \left(-\frac{1}{2}\frac{t^2(2t^2+1)}{t^2+1}, \frac{t^4(t+1)}{t^2+1}, \frac{t^4(3t+2)}{t^2+1}\right) + s\cdot\left(-\frac{1}{2}(2t^2+1)t, (t+1)(t^2-t+1), 2t^3+3\right)$			
	2	7.125 / > 90	∞
$\tilde{\boldsymbol{x}}_2(t,s) = (4, 1, t) + s\cdot((t+1)^2, t+1, 1)$ $\tilde{\boldsymbol{y}}_2(t,s) = (-3, t+1, 3t+2) + s\cdot\left(-\frac{1}{2}(t+3)(t+1), t+2, 2t+5\right)$			
	3	1.297 / > 90	∞
$\tilde{\boldsymbol{x}}_3(t,s) = s\cdot(3(t+1)^2(t-1), (t-1)^3, (t+1)^3)$ $\tilde{\boldsymbol{y}}_3(t,s) = s\cdot\left(-\frac{1}{2}(t-1)(5t^2+2t+5), 2t(t^2+3), 5t^3+3t^2+15t+1\right)$			
	3	1.562 / > 90	2
$\tilde{\boldsymbol{x}}_4(t,s) = (t^3+t, t^2-3, t^3+t) + s\cdot(-t^3, t^2-8, 2t^3-t)$ $\tilde{\boldsymbol{y}}_4(t,s) = \left(\frac{t^3}{2} - \frac{t}{2} - t^2 + 3, t^3+t^2+t-3, 3t^3+2t^2+3t-6\right) + s\cdot\left(\frac{t^3}{2} - t^2 + 8, 2t^3+t^2-t-8, 6t^3+2t^2-3t-16\right).$			

Table 2.1

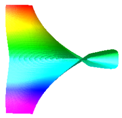
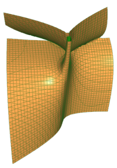
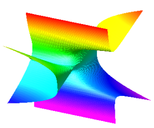
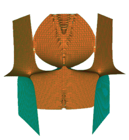
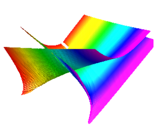
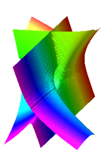
Picture of S_1	deg.	Computation time (secs.)/imp.	Affine equivalences
	4	1.641 / > 90	2
		$\tilde{\boldsymbol{x}}_5(t,s) = (t^4+t^2, t^2-3, t^3+t) + s\cdot(-t^4+2, t^2-8, 2t)$	
		$\tilde{\boldsymbol{y}}_5(t,s) = \left(-\frac{t^4}{2}-\frac{3t^2}{2}+3, t^3+t^2+t-3, 3t^3+2t^2+3t-6\right)+s\left(\frac{t^4}{2}-t^2+7, t^2+2t-8, 2t^2+6t-16\right)$	
	3	4.078 / > 90	1
		$\tilde{\boldsymbol{x}}_6(t,s) = \left(\frac{t^3+t}{t^2+1}, -3, t+t^2\right) + s\cdot(-t^3, t^2-8, t^2-t)$	
		$\tilde{\boldsymbol{y}}_6(t,s) = \left(3-\frac{t}{2}, t^2+t-3, 3t^2+3t-6\right) + s\cdot\left(-t^2+8+\frac{t^3}{3}, 2t^2-t-8, 5t^2-3t-16\right)$	
	4	1.578 / > 90	2
		$\tilde{\boldsymbol{x}}_7(t,s) = \left(\frac{t^4+t^2}{t^2+3}, \frac{t^2-3}{t^2+3}, \frac{t^3+t}{t^2+3}\right) + s\cdot(-t^4+2, t^2-8, 2t)$	
		$\tilde{\boldsymbol{y}}_7(t,s) = \left(-\frac{1}{2}\cdot\frac{t^2(t^2+5)}{t^2+3}, \frac{t^3+t^2+t-3}{t^2+3}, \frac{3t^3+6t^2+3t+6}{t^2+3}\right) + s\cdot\left(\frac{1}{2}t^4-t^2+7, t^2+2t-8, 2t^2+6t-16\right)$	
	7	4.516 / > 90	2
		$\tilde{\boldsymbol{x}}_8(t,s) = (t^6-6t^4+t^2+2t, -t^7+6t^5-t^3+t^2+t, t^3+t) + s\cdot(t^5-6t^3+t, -t^6+6t^4-t^2+1, t^2+1)$	
		$\tilde{\boldsymbol{y}}_8(t,s) = \left(t^3-\frac{3}{2}t^2-2t-\frac{1}{2}t^6+3t^4+t^7-6t^5, -t^7+6t^5+t^2+2t, -2t^7+12t^5+t^3+2t^2+5t\right) + s\cdot\left(3t^3-\frac{1}{2}t+t^2-1+t^6-6t^4-\frac{1}{2}t^5, -t^6+6t^4+2, -2t^6+12t^4+t^2+5\right)$	
 S_1  S_2	4	1.032 / > 90	0
		$\tilde{\boldsymbol{x}}_9(t,s) = \left(\frac{1}{t^4}, t, 1\right) + s\cdot(1, t, t^3)$	
		$\tilde{\boldsymbol{y}}_9(t,s) = \left(-\frac{2}{t^4}, 4, 3t\right) + s\left(\frac{1}{2}, t^3+t, 3\right)$	

Table 2.2

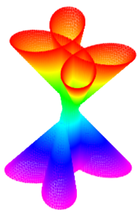
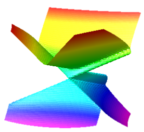
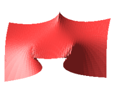
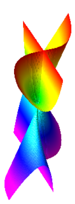
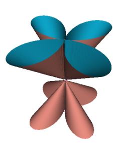
Picture of S_1	deg.	computation time (secs.) all sym. / involutions / isometries / implicit	Symmetries and isometries
	9	4.781 / 3.422 / 4.891 / > 90	3 axial 2 mirror 2 rot. + reflect. 8 isometries
$\boldsymbol{x}_1(t,s) = \left(\frac{2t^8-10t^6+3t^2-1}{t^2+1}, \frac{t^9-6t^7+6t^3-3t}{t^2+1}, t(t^2+1)^3\right)$ $+s\cdot(2t(t^4-6t^2+1), -t^6+7t^4-7t^2+1, (t^2+1)^3)$			
	7	4.981 / 2.391 / 4.016 / > 90	1 reflect. 2 isometries
$\boldsymbol{x}_2(t,s) = \left(-\frac{t^7+7t^5+3t^3-t^2-3t+1}{t^2+1}, \frac{2t(4t^5+4t^3+1)}{t^2+1}, t(t^2+1)^2\right)$ $+s\cdot(-t^4-6t^2+3, 8t^3, (t^2+1)^2)$			
	7	1.046 / 1.109 / 1.172 / > 90	1 reflect. 2 isometries
$\boldsymbol{x}_3(t,s) = (t^6-6t^4+t^2+2t, -t^7+6t^5-t^3+t^2+t, t^3+t)$ $+s\cdot(t^5-6t^3+t, -t^6+6t^4-t^2+1, t^2+1)$			
	5	0.906 / 0.969 / 1.140 / > 90	1 reflect. 2 isometries
$\boldsymbol{x}_4(t,s) = \left(\frac{t^2}{t^2+1}, \frac{t^4}{t^2+1}, \frac{t^5}{t^2+1}\right) + s\cdot(t, t^3, 1)$			
	6	2.015 / 1.641 / 5.250 / > 90	5 reflect. 5 axial sym. 1 central 2 rot. sym. 2 rot. + reflect. 16 isometries
$\boldsymbol{x}_5(t,s) = s\cdot(2t(t^4-6t^2+1), (-t^2+1)(t^4-6t^2+1), (t^2+1)^3)$			

Table 2.3

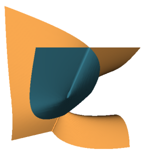
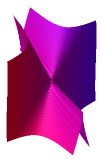
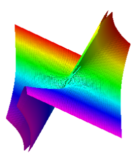
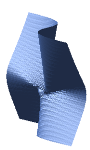
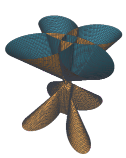
Picture of S_1	deg.	Computation time (secs.) all sym. / involutions isometries / implicit	Symmetries and isometries
	2	1.704 / 1.703 / 2.203 / 23.671	1 axial sym. 2 isometries
$\boldsymbol{x}_6(t,s) = (4,1,t) + s\cdot((t+1)^2, t+1, 1)$			
	3	0.531 / 0.703 / 1.469 / 0.156	central 1 reflection 1 axial sym. 4 isometries
$\boldsymbol{x}_7(t,s) = s\cdot(3(t+1)^2(t-1), (t-1)^3, (t+1)^3)$			
	7	1.016 / 0.938 / 1.219 / > 90	central 2 isometries
$\boldsymbol{x}_8(t,s) = \left(\frac{t^3}{t^2+1}, \frac{t^5}{t^2+1}, \frac{t^7}{t^2+1}\right) + s\cdot(-t^5+t, 3t^7, -2t^3)$			
	6	0.906 / 1.032 / 1.110 / > 90	1 axial 2 isometries
$\boldsymbol{x}_9(t,s) = (t^4+t^2+t, t^6+t^3, t^5+t^3+t^2+3t) + s\cdot(t^3+t, t^5, t^4+t^2+3)$			
	17	4.406 / 3.329 / 4.969 / > 90	4 reflect. 1 axial sym. 2 rotational 8 isometries
$\boldsymbol{x}_{10}(t,s) = \left(-\frac{t^{17}-6t^{15}+6t^{11}-6t^7+6t^3-t^2-t+1}{t^2+1}, \frac{2t(t^{15}-5t^{13}-5t^{11}+t^9+t^7-5t^5-5t^3+t+1)}{t^2+1}, t(t^2+1)^3(t^8+1)\right)$ $+s\cdot(-t^6+7t^4-7t^2+1, 2t(t^4-6t^2+1), (t^2+1)^3)$			

Table 2.4

2.5 Application to implicit algebraic surfaces under certain conditions

In this section, we will see how to apply the method developed in the previous sections to find the reflections and rotational symmetries (in particular, axial symmetries) of an implicit algebraic surface under certain conditions. Let $F(x,y,z)$ define an irreducible, implicit algebraic surface S of total degree N, and let

$$F(x,y,z) = F_N(x,y,z) + F_{N-1}(x,y,z) + \cdots + F_0(x,y,z),$$

where $F_i(x,y,z)$ denotes the homogeneous form of $F(x,y,z)$ of degree $i = 0,1,\ldots,N$. Thus, $F_i(x,y,z)$ is a homogeneous polynomial of degree i. In particular, we refer to $F_N(x,y,z)$ as the *highest order form* of $F(x,y,z)$.

Let $\mathbf{x} = (x,y,z)$, and let $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ be a symmetry of S. The following two lemmas show the connections of the problem treated in this section with the ideas developed in previous sections.

Lemma 2.4. *Let $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x}+\boldsymbol{b}$ be a symmetry of the surface S defined by $F(x,y,z) = 0$, where F is irreducible. Then $\tilde{f}(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$ is a symmetry of the surface defined by $F_N(x,y,z) = 0$.*

Proof. Since $F(x,y,z)$ is irreducible by hypothesis, if $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ is a symmetry of S then

$$F(\boldsymbol{A}\mathbf{x} + \boldsymbol{b}) = \lambda F(x,y,z) = \lambda F_N(\mathbf{x}) + \lambda F_{N-1}(\mathbf{x}) + \cdots + \lambda F_0(\mathbf{x}),$$

with λ a constant. Since

$$F(\boldsymbol{A}\mathbf{x} + \boldsymbol{b}) = F_N(\boldsymbol{A}\mathbf{x}) + F_{N-1}(\boldsymbol{A}\mathbf{x} + \boldsymbol{b}) + \cdots ,$$

we conclude that $F_N(\boldsymbol{A}\mathbf{x}) = \lambda F_N(\mathbf{x})$, and the result follows. ■

Lemma 2.5. *The surface $F_N(x, y, z) = 0$ is a conical surface, with vertex at the origin.*

Proof. Since $F_N(x, y, z)$ is a homogeneous polynomial, for any constant β we have $F_N(\beta x, \beta y, \beta z) = F_N(\beta \mathbf{x}) = \beta^N F_N(x, y, z)$. Thus, for any point (x, y, z) of the surface $F_N(x, y, z)$, the line connecting (x, y, z) with the origin is included in the surface. ■

Thus, the matrices $\boldsymbol{A}$ defining the symmetries $\tilde{f}(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$ of $F_N(x, y, z) = 0$ provide a superset for the matrices $\boldsymbol{A}$ defining the symmetries $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ of $F(x, y, z) = 0$.

Now we can make precise the conditions under which the ideas in previous sections can be applied to the problem in this section. Whenever $F_N(x, y, z)$: (1) is irreducible and (2) defines a real, rational surface, from Lemma 2.5 we know that $F_N(x, y, z) = 0$ is a conical surface S_N, with vertex at the origin, whose symmetries can, therefore, be found by applying the method in this paper (in particular, using Prop. 2.4). Additionally, if $F_N(x, y, z) = 0$ defines a rational surface, since S_N is a conical surface the intersection of the surface with a generic plane is a rational curve that we can parametrize rationally with well-known methods. As a consequence, a parametrization of S_N of the type $\boldsymbol{x}(t, s) = s\boldsymbol{A}(t)$ can be computed. In turn, the symmetries of S_N can be obtained by applying our methods; taking Lemma 2.4 into account, the rotational symmetries and reflections in planes of S can be found from here. The next lemma sheds some light on this last step.

Lemma 2.6. *(1) If $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ represents a rotational symmetry about an axis $\mathcal{A}$, then $\tilde{f}(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$ represents a symmetry of the same kind, with axis $\mathcal{A}'$ parallel to $\mathcal{A}$ through the origin.*

(2) If $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ represents a symmetry with respect to a symmetry plane Π, then $\tilde{f}(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$ represents a symmetry with respect to a symmetry plane Π' through the origin.

Proof. We prove (1); the proof of (2) is similar. Let $\mathcal{A}$ be the rotational axis corresponding to f, let P_0 be a point on the axis $\mathcal{A}$, and let $\vec{v} \in \mathbb{R}^3$ be a vector parallel to

$\mathcal{A}$. Since f leaves $\mathcal{A}$ invariant, for any $\lambda \in \mathbb{R}$ we have $f(P_0 + \lambda\vec{v}) = P_0 + \lambda\vec{v}$. Since $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$, we get

$$f(P_0 + \lambda\vec{v}) = \boldsymbol{A} \cdot (P_0 + \lambda\vec{v}) + \boldsymbol{b} = \boldsymbol{A} \cdot P_0 + \lambda\boldsymbol{A} \cdot \vec{v} + \boldsymbol{b} = P_0 + \lambda\vec{v}.$$

Since the above equality holds for any λ, we conclude that $\boldsymbol{A} \cdot \vec{v} = \vec{v}$. Thus, $\widehat{f}(\beta\vec{v}) = \beta\vec{v}$ for any $\beta \in \mathbb{R}$, i.e., $\widehat{f}$ leaves the line $\mathcal{A}'$ parallel to $\mathcal{A}$ through the origin invariant. Since the nature of the symmetry depends upon the eigenvalues of $\boldsymbol{A}$, and this matrix is common to f and $\widehat{f}$, the result follows. ■

Recall that an axial symmetry is nothing else than a rotational symmetry of angle π, so Lemma 2.6 includes axial symmetries as well. Therefore, whenever $F_N(x, y, z)$ satisfies the hypotheses mentioned before, we can proceed as follows:

(1) Compute a parametrization $\boldsymbol{x}(t, s) = s\boldsymbol{A}(t)$ of the surface S_N defined by $F_N(x, y, z) = 0$.

(2) Compute the rotational symmetries and reflections of $\boldsymbol{x}(t, s) = s\boldsymbol{A}(t)$.

(3) [rotational] Let $\mathcal{A}'$ be the axis of rotational symmetry of S_N, and let $\widehat{f}(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$ be the corresponding symmetry.

   (i) Apply a rigid motion to the surface S defined by $F(x, y, z) = 0$, so that $\mathcal{A}'$ is the z-axis.

   (ii) If there exists $\boldsymbol{b} \in \mathbb{R}^3$ such that $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ is a rotational symmetry of S about an axis $\mathcal{A}$, parallel to $\mathcal{A}'$, then for a generic plane Π_{z_0}, normal to the z-axis, the intersection curve $S \cap \Pi_{z_0}$ exhibits central symmetry around the point $P_0 = \mathcal{A} \cap \Pi_{z_0}$. Central symmetry of $S \cap \Pi_{z_0}$ can be detected with the algorithm in [15].

   (iii) Check whether or not S has rotational symmetry with respect to the line $\mathcal{A}$ parallel to $\mathcal{A}'$ through P_0.

(4) [planar] Let Π' be a symmetry plane of S_N, and let $\widehat{f}(\mathbf{x}) = \boldsymbol{A}\mathbf{x}$ be the corresponding symmetry.

(i) Apply a rigid motion to the surface S defined by $F(x,y,z)=0$, so that Π' is the xz-plane.

(ii) If there exists $\boldsymbol{b}\in\mathbb{R}^3$ such that $f(\mathbf{x})=\boldsymbol{A}\mathbf{x}+\boldsymbol{b}$ is a reflection in a plane Π, parallel to Π', then for a generic plane Π_{z_0}, normal to the z-axis, the intersection curve $S\cap\Pi_{z_0}$ exhibits axial symmetry with respect to the line $\ell=\Pi\cap\Pi_{z_0}$. Axial symmetry of $S\cap\Pi_{z_0}$ can be detected with the algorithm in [15].

(iii) Check whether or not S is symmetric with respect to the plane parallel to Π through ℓ.

Remark 2.3. If the surface S_N defined by $F_N(x,y,z)=0$ is a surface of revolution, not a sphere, S_N has an axis $\mathcal{A}$ of revolution and has also infinitely many symmetry planes intersecting at $\mathcal{A}$. Whenever we apply an orthogonal change of coordinates mapping $\mathcal{A}$ to the z-axis, the proposed method is also valid in this case.

Example 2.4. Let S be an algebraic surface implicitly defined by $F(x,y,z)=x^6+y^5z+6x^5+14x^4+16x^3+8x^2+z^2$. The highest order form of S is $F_N(x,y,z)=x^6+y^5z$. The polynomial $F_N(x,y,z)$ is irreducible and defines a rational surface (in fact, a conic surface with vertex at the origin), which can be parametrized, for instance, as

$$\left(x,y,\frac{-x^6}{y^5}\right).$$

In order to compute a parametrization of the form $\boldsymbol{x}(t,s)=s\boldsymbol{A}(t)$, we intersect S_N with the plane $y=2$. This yields the planar curve

$$\{x^6+32z=0,\ y=2\},$$

which can be parametrized as $\left(t,2,-\frac{1}{32}t^6\right)$. In turn, S_N can be parametrized as $\boldsymbol{x}(t,s)=s\left(t,2,-\frac{1}{32}t^6\right)$. Using the method in Section 3, one can check that S_N has symmetries with respect to the plane $x=0$, and with respect to the x-axis.

To check whether S is symmetric with respect to a plane parallel to $x=0$, we

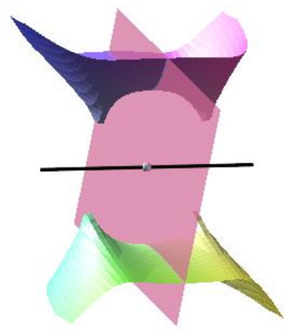


Figure 2.4: Symmetry plane and symmetry axis of $F(x, y, z) = x^6 + y^5 z + 6x^5 + 14x^4 + 16x^3 + 8x^2 + z^2 = 0$.

intersect S with the plane $z = 3$. The resulting curve is

$$\{x^6 + 6x^5 + 3y^5 + 14x^4 + 16x^3 + 8x^2 + 9 = 0,\ z = 3\}.$$

To analyze the symmetries of this curve, we use the method in [15] which shows that it is symmetric with respect to the line $x = -1$. Finally, we can easily check that S is certainly symmetric with respect to the plane $x + 1 = 0$.

Similarly, we check that S is symmetric with respect to the x-axis; in fact, one can straightforwardly check this observing that $F(x, y, z)$ is invariant when we apply the transformation $\{x := x,\ y := -y,\ z := -z\}$.

One can observe the symmetries of this surface in Figure 2.4. The symmetry axis is shown in black; the solid sphere corresponds to the intersection point of the symmetry axis and the symmetry plane.

The method proposed here, however, does not allow to find the central symmetries of the surface. For instance, one can check that the surface in Example 2.4 is symmetric with respect to the origin, but we cannot read this from $F_N(x, y, z)$; in fact, the form of the highest degree of *any* polynomial in x, y, z defines a surface symmetric with respect to the origin (since it is a conic surface with vertex at the origin).

2.6 Observations on the computation of projective equivalences

Projective equivalences between S_1, S_2 correspond to rational mappings $f(\mathbf{x})$ from $\mathbb{R}^3$ to $\mathbb{R}^3$ satisfying $f(S_1) = S_2$, where the components of f have the form

$$\frac{a_{i1}x + a_{i2}y + a_{i3}z + b_i}{a_{41}x + a_{42}y + a_{43}z + b_4}, \tag{2.34}$$

for $i = 1, 2, 3$. Whenever f is invertible, Theorem 2.1 is also valid for this case, so each projective equivalence between S_1, S_2 has an associated mapping $\varphi(t, s) = (\varphi_1(t, s), \varphi_2(t, s))$ in parameter space. Additionally, projective mappings are collineations, i.e., they map lines to lines. Thus, we can argue as in the first part of Proposition 2.1 to conclude that $\varphi_1(t, s) = \psi(t)$, where $\psi(t)$ is a Möbius transformation. However, in general, the form of $\varphi_2(t, s)$ is not the same as in Proposition 2.1. Indeed, using Eq. (2.34) one has that

$$\varphi_2(t, s) = \frac{\xi_1(t) + s\xi_2(t)}{\xi_3(t) + s\xi_4(t)},$$

where the $\xi_j(t)$ are polynomials. As a consequence, the remaining results of Section 2.2.2, and in particular the form of φ predicted by Theorem 2.2, cannot be easily generalized. Therefore, an approach analogous to the one in this paper for projective equivalences requires further work.

2.7 Comparison with other works

In this section we compare our results with the results in other related works. There are three papers that we need to consider here: [57], which appeared before our results were made public, [33], which was developed independently and was made public almost simultaneously to our own results, and [63], which was recently uploaded to the

ArXiv.

The paper [57] provides an algorithm for computing the projective and affine equivalences between two rational surfaces without projective base points. Notice that the algorithm in [57] is, however, not applicable to the case of ruled surfaces. Indeed, the components of a rational parametrization of a ruled surface in standard form can be written, using projective coordinates, as

$$A(t, \omega) + sB(t, \omega),$$

where A, B are homogeneous polynomials (with ω as the homogenization variable). Thus, [0:1:0] is always a base point of the surface, so rational ruled surfaces in standard form always have base points.

The paper [63] addresses projective equivalences between special types of algebraic varieties, and includes the case of ruled rational surfaces. While the paper treats a more general problem compared to ours (projective equivalences versus affine equivalences), it is unclear whether the algorithm in [63] has a better or worse performance. The idea in [63] is to reduce the computation of projective equivalences between ruled surfaces to exploring the relationship between the rational curves corresponding to these surfaces in the Plücker quadric, i.e., in five-dimensional projective space. Thus, we move to a higher dimension, and the degrees of the curves to analyze are also higher compared to the degrees in the original parametrizations. In [63] there are no details on the algorithm and no timings are given, so it is difficult to compare their results with ours.

The results in [63], recently uploaded to the ArXiv, however, are applicable to our case. Nevertheless, the algorithm in [63] seems difficult to implement. Although the algorithm is certainly very general, no timings are given, and again the comparison of the performance of their algorithm with the algorithm in this chapter is unclear.

CHAPTER 3

AFFINE EQUIVALENCES OF TRIGONOMETRIC CURVES

In this chapter, we provide an efficient algorithm to detect whether two parametrized curves whose components are given as finite linear combinations of sines and cosines, i.e., truncated Fourier series, in any dimension, are affinely equivalent. If the coefficients of the parametrizations are exact (the *exact case*), the solutions are obtained by computing univariate gcds. If the coefficients of the parametrizations are known as floating-point numbers (the *approximate case*), we need to compute approximate gcds.

In some references [60, 61], these curves receive the name of *trigonometric curves* or *generalized trigonometric curves*. In other, more applied, references (see for instance [119]), these curves are called elliptic *Fourier descriptor (EFD) representations* and are often used to describe closed planar and space curves (see, for instance, the references in [119]).

In particular, for these curves one can compute *shape descriptors* (see [46, 47, 53], among many others), which are invariants that can be computed from the parametrization, and that can be used for curve recognition, in particular similarity recognition. However, the approach that we use to solve the affine equivalence problem for trigonometric curves is similar to the approach used in papers like [11, 13, 56], that we discussed in previous chapters. In these papers, involving rational curves, the main idea is to reduce the computation to finding a Möebius transformation from which the problem is solved.

For trigonometric curves, it is a well-known trick to compute a rational parametrization depending on one complex parameter taking values in the unit circle, and thus the techniques of the papers [11, 13, 56] are applicable. However, the rational parametrization of a trigonometric curve has special properties that one can exploit, resulting in more advantageous algorithms. In particular, we can prove that the associated Möebius transformation has a predictable shape only depending on one parameter, so that the final computation reduces to computing the greatest common divisor of univariate complex polynomials.

In the presence of floating-point numbers, due to numerical inaccuracies, this greatest common divisor can be constant, and thus the method must be adapted. Because of this, we replace greatest common divisors with approximate common divisors (see, for instance, [28, 65, 82, 81, 122]); in our experiments, we use the MATLAB implementation for computing approximate gcds provided in [123].

Furthermore, we explore the possibility of computing approximate affine equivalences between non-necessarily rational parametric curves, replacing the components of the curves by truncated Fourier series of high degree.

3.1 Preliminaries on trigonometric curves

A *trigonometric curve* $\mathcal{C} \subset \mathbb{R}^n$, following [61], is a parametric curve whose components are truncated Fourier series, i.e.,

$$\boldsymbol{x}(t) = (x_1(t), \dots, x_n(t)), \tag{3.1}$$

where

$$x_i(t) = \sum_{\ell=0}^{m_i} [a_\ell^{(i)} \cos(\ell t) + b_\ell^{(i)} \sin(\ell t)], \quad t \in [0, 2\pi],\ i = 1, \dots, n. \tag{3.2}$$

We refer to a parametrization of this kind as a *trigonometric* parametrization. A proper trigonometric parametrization $\boldsymbol{x}(t)$ (see Sec. 1.3) is also called *simple*; for instance, $(\sin(t), \cos(t))$ is a simple parametrization of a circle, while $(\sin(2t), \cos(2t))$ is not.

We will assume that at least one of the m_i is different from 1 since otherwise the curve is an ellipse. Affine equivalences between ellipses can be detected from their implicit equations.

Furthermore, a trigonometric parametrization $\widehat{\boldsymbol{x}}(t)$ is a *simplification* of another trigonometric parametrization $\boldsymbol{x}(t)$, if $\widehat{\boldsymbol{x}}(t)$ is simple and both $\boldsymbol{x}(t), \widehat{\boldsymbol{x}}(t)$ parametrize a same curve $\mathcal{C} \subset \mathbb{R}^n$; we also say that $\widehat{\boldsymbol{x}}(t)$ is the result of simplifying $\boldsymbol{x}(t)$. In [61] it is shown (see Theorem 2.1 in [61]) that for a given trigonometric curve, there always exists either a trigonometric or a polynomial simplification. Furthermore, algorithms for simplifying a trigonometric curve are also provided in [61].

Given $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^2$ trigonometric curves, if there exists a polynomial simplification for $\mathcal{C}, \mathcal{D}$ then we can use the results in [56] to find the affine equivalences between them. Notice that if there exists a trigonometric simplification for $\mathcal{C}$ and a polynomial one for $\mathcal{D}$, then they cannot be affinely equivalent. Hence, we will only consider trigonometric curves with simple parametrizations.

We will assume that $\mathcal{C}$ is not contained in a hyperplane of $\mathbb{R}^n$ since in that case the problem can be reduced to a lower dimension. Additionally, for $n = 2$ we will assume that $\mathcal{C}$ is not a conic; the problem for conics is easy and can be reduced to matrix operations.

Any trigonometric curve admits infinitely many simplifications. However, the following result, which is a reformulation of Theorem 2.5 in [61], shows that all the simplifications of the same trigonometric curve are related by an explicit type of transformation.

Lemma 3.1. *Let $\boldsymbol{x}_1(t), \boldsymbol{x}_2(t)$ be two simple trigonometric parametrizations of a same trigonometric curve $\mathcal{C} \subset \mathbb{R}^n$. Then $\boldsymbol{x}_2 = \boldsymbol{x}_1 \circ \psi$, where $\psi(t) = \alpha \pm t$.*

When dealing with trigonometric curves, a common technique is to use a rational parametrization of the curves employing the change $z = e^{\mathbf{i}t}$, where $\mathbf{i}^2 = -1$ and z belongs to the unit circle S^1 (see for instance [60, 61, 119]). Since $e^{\mathbf{i}t} = \cos t + \mathbf{i} \sin t$, and taking into account that for $z \in S^1$ the conjugate $\overline{z}$ satisfies that $\overline{z} = \frac{1}{z}$, we deduce that

$$\cos t = \frac{z^2+1}{2z}, \ \sin t = \frac{z^2-1}{2\mathbf{i}z}, \ \cos(Mt) = \frac{z^{2M}+1}{2z^M}, \ \sin(Mt) = \frac{z^{2M}-1}{2\mathbf{i}z^M}, \tag{3.3}$$

where $M \in \mathbb{N}$. Substituting these relationships into Eq. (3.2), we get a *rational* complex parametrization (i.e., a parametrization whose components are quotients of polynomials)

$$\tilde{\boldsymbol{x}}(z) = (\tilde{x}_1(z), \ldots, \tilde{x}_n(z)), \tag{3.4}$$

where each component satisfies that

$$\tilde{x}_i(z) = \frac{P_i(z)}{z^{m_i}}, \ i = 1, \ldots, n, \tag{3.5}$$

with $P_i(z)$ complex polynomials of degree $2m_i$, such that $\gcd(P_1(z), \cdots, P_n(z), z^{\mathbf{m}}) = 1$, with $\mathbf{m} = gcd(m_1, \ldots, m_n)$ and $z \in S^1$.

We refer to $\tilde{\boldsymbol{x}}(z)$ as the *rational complex parametrization* associated with $\boldsymbol{x}(t)$. Denoting $N = \max\{m_i | i = 1, \ldots, n\}$, we say that the degree of $\tilde{\boldsymbol{x}}(z)$ is $2N$. Observe that not every $P_i(z)$ has degree $2N$, but there always exists $i \in \{1, \ldots, n\}$ such that the degree of $P_i(z)$ is $2N$.

Remark 3.1. *One can easily see that*

$$P_i(z) = \frac{1}{2} \sum_{\ell=0}^{m_i} [A_\ell z^{m_i+\ell} + B_\ell z^{m_i-\ell}]$$

where $A_\ell = a_\ell^{(i)} - \mathbf{i} b_\ell^{(i)}$, $B_\ell = a_\ell^{(i)} + \mathbf{i} b_\ell^{(i)}$. *In particular, since* a_{m_i}, b_{m_i} *are real and nonzero, then* $B_{m_i} \neq 0$, *so no cancellation in* $\tilde{x}_i(z) = \frac{P_i(z)}{z^{m_i}}$ *is possible.*

Remark 3.2. *An alternative possibility to work with trigonometric curves is to apply* *the classical rational change of variables*

$$(\cos(t), \sin(t)) \to \left(\frac{1-s^2}{1+s^2}, \frac{2s}{1+s^2}\right). \tag{3.6}$$

However, this produces parametrizations with more terms and higher coefficients. For instance, consider the function

$$f(t) = \cos(t) + 3\cos(2t) - 2\cos(3t) + 4\cos(5t) - \cos(8t).$$

While the change in Eq. (3.6) *produces*

$$\frac{-s^{16} - 74s^{14} + 1758s^{12} - 8306s^{10} + 12780s^8 - 7838s^6 + 1842s^4 - 166s^2 + 5}{s^{16} + 8s^{14} + 28s^{12} + 56s^{10} + 70s^8 + 56s^6 + 28s^4 + 8s^2 + 1},$$

the change in Eq. (3.3) *yields*

$$\frac{4z^{16} + z^{14} + 2z^{12} - 2z^{10} + 4z^6 - 1}{z^{16}},$$

which is a simpler expression, with smaller coefficients and fewer terms.

Since for $t \in [0, 2\pi]$ the mapping $z = e^{\mathbf{i}t} : [0, 2\pi] \to S^1$ is invertible, and since we are assuming that Eq. (3.1) is a simple trigonometric parametrization, we get that $\tilde{\boldsymbol{x}}(z)$ in Eq. (3.4), seen as mapping from S^1 to $\mathcal{C}$, is proper. Furthermore, we get the following result as a corollary of Lemma 3.1.

Corollary 3.1. *Let $\tilde{\boldsymbol{x}}_1(z), \tilde{\boldsymbol{x}}_2(z)$ be two rational parametrizations of a same trigonometric curve $\mathcal{C} \subset \mathbb{R}^n$, associated with two simple trigometric parametrizations $\boldsymbol{x}_1(t), \boldsymbol{x}_2(t)$ of $\mathcal{C}$. Then $\tilde{\boldsymbol{x}}_2 = \tilde{\boldsymbol{x}}_1 \circ \xi$, where $\xi(z) = kz$ or $\xi(z) = \dfrac{k}{z}$, and $k, z \in S^1$.*

Proof. From Lemma 3.1, $\boldsymbol{x}_2 = \boldsymbol{x}_1 \circ (\alpha \pm t)$. Since $\boldsymbol{x}_j = \tilde{\boldsymbol{x}}_j \circ e^{\mathbf{i}t}$ for $j = 1, 2$, we get $\tilde{\boldsymbol{x}}_2 \circ e^{\mathbf{i}t} = \tilde{\boldsymbol{x}}_1 \circ e^{\mathbf{i}t} \circ (\alpha \pm t)$. Thus,

$$\tilde{\boldsymbol{x}}_2 \circ e^{\mathbf{i}t} = \tilde{\boldsymbol{x}}_1 \circ (e^{\mathbf{i}\alpha} \cdot e^{\pm \mathbf{i}t}).$$

Calling $k = e^{\mathbf{i}\alpha}$ and since $z = e^{\mathbf{i}t}$, the result follows. ■

Notice that $\varphi(z) = kz$ and $\varphi(z) = \frac{k}{z}$ are, in particular, Möbius transformations of S^1. Moreover, we also have the following corollary of Lemma 3.1, which follows from Corollary 3.1.

Corollary 3.2. *Let $\tilde{\boldsymbol{x}}_1(z), \tilde{\boldsymbol{x}}_2(z)$ be two rational parametrizations of a same trigonometric curve $\mathcal{C} \subset \mathbb{R}^n$, associated with two simple trigometric parametrizations $\boldsymbol{x}_1(t), \boldsymbol{x}_2(t)$ of $\mathcal{C}$. Then the degrees of both $\tilde{\boldsymbol{x}}_1(z), \tilde{\boldsymbol{x}}_2(z)$ are the same.*

3.2 Affine equivalences of trigonometric curves

Recall that two trigonometric curves $\mathcal{C}, \mathcal{D}$ are affinely equivalent if there exists a nonsingular affine mapping $f : \mathbb{R}^n \longrightarrow \mathbb{R}^n$,

$$f(\mathrm{x}) = \boldsymbol{A}\mathrm{x} + \boldsymbol{b}, \qquad \mathrm{x} \in \mathbb{R}^n, \tag{3.7}$$

with $\boldsymbol{b} \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$ a nonsingular square matrix, such that $f(\mathcal{C}) = \mathcal{D}$.

Henceforth, we will consider trigonometric curves $\mathcal{C}, \mathcal{D}$ defined by simple parametrizations

$$\boldsymbol{x}(t) = (x_1(t), \dots, x_n(t)), \;\; \boldsymbol{y}(t) = (y_1(t), \dots, y_n(t)) \tag{3.8}$$

where $x_i(t), y_i(t)$ are as in Eq. (3.2). We denote by $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$, with $z \in S^1$, the rational parametrizations associated with $\boldsymbol{x}(t), \boldsymbol{y}(t)$, so the components of $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ are as in Eq. (3.5). Our goal is to detect whether $\mathcal{C}$ and $\mathcal{D}$ are affinely equivalent, i.e., to check whether they are related by a mapping like Eq. (3.7), and in the affirmative case to find the affine equivalences between $\mathcal{C}$ and $\mathcal{D}$. We first need the following lemma.

Lemma 3.2. *Let $\boldsymbol{x}(t)$ be a simple trigonometric parametrization as in Eq.* (3.1)*, let $\tilde{\boldsymbol{x}}(z)$ be its associated rational parametrization, and let $2N$ be the degree of $\tilde{\boldsymbol{x}}(z)$. Let $f(\mathrm{x}) = \boldsymbol{A}\mathrm{x} + \boldsymbol{b}$, $\mathrm{x} \in \mathbb{R}^n$, with $\boldsymbol{b} \in \mathbb{R}^n$ and $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ a nonsingular square matrix.*

(1) $\boldsymbol{x}^\star(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{b}$ is a simple trigonometric parametrization, with associated rational complex parametrization $\tilde{\boldsymbol{x}}^\star(z) = \boldsymbol{A}\tilde{\boldsymbol{x}}(z) + \boldsymbol{b}$.

(2) The degrees of $\tilde{\boldsymbol{x}}(z)$ and $\tilde{\boldsymbol{x}}^\star(z)$ are the same.

Proof. Let us see (1). Since the components of $\boldsymbol{x}^\star(t)$ are linear combinations of the components of $\boldsymbol{x}(t)$, it is clear that $\boldsymbol{x}^\star(t)$ is trigonometric. Furthermore, since $\boldsymbol{A}$ is regular, f is an injective mapping. Thus, $\boldsymbol{x}^\star(t)$ is simple because it is the composition of a simple trigonometric parametrization with an injective mapping. Finally, since $\tilde{\boldsymbol{x}}^\star = \boldsymbol{x}^\star \circ e^{\mathbf{i}t}$, we easily deduce that $\tilde{\boldsymbol{x}}^\star(z) = \boldsymbol{A}\tilde{\boldsymbol{x}}(z) + \boldsymbol{b}$.

Now let us see (2). It is clear that the degree of $\tilde{\boldsymbol{x}}^\star(z)$ cannot be greater than $2N$; so let us see that the degree of $\tilde{\boldsymbol{x}}^\star(z)$ cannot be less than $2N$. Following the notation in Eq. (3.2), for $i = 1, \dots, n$ let $a_N^{(i)}, b_N^{(i)}$ denote the coefficients of $\cos(Nt), \sin(Nt)$ in the i-th component of $\boldsymbol{x}(t)$, $x_i(t)$. Of course $a_N^{(i)}, b_N^{(i)}$ are zero when $m_i < N$. Notice, however, that since the degree of $\tilde{\boldsymbol{x}}(z)$ is $2N$, not all the $a_N^{(i)}, b_N^{(i)}$ can vanish. The coefficients of $\cos(Nt), \sin(Nt)$ in the i-th component of $\boldsymbol{x}^\star(t)$ are

$$\begin{gathered} \boldsymbol{A}_{i1}a_N^{(1)} + \boldsymbol{A}_{i2}a_N^{(2)} + \cdots + \boldsymbol{A}_{in}a_N^{(n)}, \\ \boldsymbol{A}_{i1}b_N^{(1)} + \boldsymbol{A}_{i2}b_N^{(2)} + \cdots + \boldsymbol{A}_{in}b_N^{(n)}. \end{gathered}$$

Now if the degree of $\tilde{\boldsymbol{x}}^\star(z)$ is less than $2N$, then the above expressions must vanish for all $j = 1, \dots, n$. Since not all the $a_N^{(i)}, b_N^{(i)}$ are zero, this implies that there exists $\mathbf{v} \in \mathbb{R}^n$, $\mathbf{v} \neq \mathbf{0}$, such that $\boldsymbol{A} \cdot \mathbf{v} = \mathbf{0}$. But this is impossible because $\boldsymbol{A}$ is a regular matrix. ∎

Then we have the following result.

Theorem 3.1. *Let $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$ be two trigonometric curves, defined by rational complex parametrizations $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$, with $z \in S^1$, associated with simple trigonometric parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$. Let $f : \mathcal{C} \to \mathcal{D}$ be an affine mapping $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$, where $\mathbf{x} \in \mathbb{R}^n$, with $\boldsymbol{b} \in \mathbb{R}^n$ and $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ a nonsingular square matrix, such that $f(\mathcal{C}) = \mathcal{D}$. Then there exists $k \in S^1$ and $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$, such that the diagram*

$$\begin{tikzcd} \mathcal{C} \arrow[r, "f"] & \mathcal{D} \\ S^1 \arrow[u, dashed, "\tilde{\boldsymbol{x}}"] \arrow[r, dashed, "\varphi"'] & S^1 \arrow[u, dashed, "\tilde{\boldsymbol{y}}"'] \end{tikzcd} \tag{3.9}$$

is commutative, i.e., for $z \in S^1$ *we get that* $f \circ \tilde{\boldsymbol{x}} = \tilde{\boldsymbol{y}} \circ \varphi$, *or equivalently*

$$\boldsymbol{A}\tilde{\boldsymbol{x}}(z) + \boldsymbol{b} = \tilde{\boldsymbol{y}}(\varphi(z)). \tag{3.10}$$

Furthermore, the degrees of $\tilde{\boldsymbol{x}}(z)$ *and* $\tilde{\boldsymbol{y}}(z)$ *are the same.*

Proof. Since $f(\mathcal{C}) = \mathcal{D}$ and by statement (1) of Lemma 3.2, $\boldsymbol{x}^\star(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{b}$ is also a simple trigonometric parametrization of $\mathcal{D}$. Furthermore, also by statement (1) of Lemma 3.2, the rational complex parametrization $\tilde{\boldsymbol{x}}^\star(z)$ associated with $\boldsymbol{x}^\star(t)$ is $\tilde{\boldsymbol{x}}^\star(z) = \boldsymbol{A}\tilde{\boldsymbol{x}}(z) + \boldsymbol{b}$. Then the results follow from Corollary 3.1, Corollary 3.2 and the statement (2) of Lemma 3.2. ■

Theorem 3.1 provides the following corollary on the involutional symmetries of a trigonometric curve $\mathcal{C}$.

Corollary 3.3. *In the hypotheses of Theorem 3.1, if* $\mathcal{C} = \mathcal{D}$ *and* f *is a nontrivial involutional symmetry (i.e., different from the identity) then* $\varphi(z) = -z$ *or* $\varphi(z) = \frac{k}{z}$ *with* $k \in S^1$.

Proof. From Theorem 3.1, assuming $\mathcal{C} = \mathcal{D}$ we get $\varphi = \tilde{\boldsymbol{x}}^{-1} \circ f \circ \boldsymbol{x}$. Thus, if $f \circ f = \mathrm{id}_{\mathbb{R}^n}$ then $\varphi \circ \varphi = \mathrm{id}_{\mathbb{R}^n}$ as well, so φ is an involution of S^1. Now from Lemma 3.1, $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$. The mapping $\varphi(z) = \frac{k}{z}$ is always an involution. However, $\varphi(z) = kz$ is an involution only when $k = \pm 1$. Since $\varphi(z) = z$ implies that f is the identity, the result follows. ■

Theorem 3.1 can be exploited in order to find the affine equivalences between $\mathcal{C}$ and $\mathcal{D}$. The general idea is to write first the entries $\boldsymbol{A}_{ij}$ of the matrix $\boldsymbol{A}$ and the components of the vector $\boldsymbol{b}$ as rational functions of k by using Eq. (3.10), and then find, if any, the values $k \in S^1$ such that Eq. (3.10) is satisfied. In particular, we get polynomial conditions in the variable k that must have a common root, belonging to S^1, for $\mathcal{C}, \mathcal{D}$ to be affinely equivalent.

The following result shows that under our hypotheses, in particular by excluding the possibility that $\mathcal{C}$ lies in a hyperplane, writing the $\boldsymbol{A}_{ij}$ and the components of $\boldsymbol{b}$ in terms of k by using Eq. (3.10) is always possible.

Lemma 3.3. *If $\mathcal{C}$ is not contained in a hyperplane, Eq.* (3.10) *allows to write the $\boldsymbol{A}_{ij}$ and $\boldsymbol{b}$ in terms of k.*

Proof. We focus on proving that the entries $\boldsymbol{A}_{ij}$ of $\boldsymbol{A}$ can be written as rational functions of k. Once this is done, from Eq. (3.10) we get $\boldsymbol{b} = \tilde{\boldsymbol{y}}(\varphi(\mathbf{a})) - \boldsymbol{A}\tilde{\boldsymbol{x}}(\mathbf{a})$ for any $\mathbf{a} \in S^1$.

A possibility to write $\boldsymbol{A}$ in terms of k is to choose $n+1$ distinct complex numbers $z_0, z_1, \dots, z_n \in S^1$, and then consider the matrix equations $\boldsymbol{A}\tilde{\boldsymbol{x}}(z_i) + \boldsymbol{b} = \tilde{\boldsymbol{y}}(\varphi(z_i))$, $i = 0, 1, \dots, n$. By subtracting the first equation from the last n equations, we get n matrix equations of the form

$$\boldsymbol{A}(\tilde{\boldsymbol{x}}(z_i) - \tilde{\boldsymbol{x}}(z_0)) = \tilde{\boldsymbol{y}}(\varphi(z_i)) - \tilde{\boldsymbol{y}}(\varphi(z_0)). \tag{3.11}$$

Let W be the $n \times n$ matrix whose columns are the vectors $\mathbf{v}_i = \tilde{\boldsymbol{x}}(z_i) - \tilde{\boldsymbol{x}}(z_0)$, for $i = 1, \dots, n$, and let Z be the matrix whose columns are the vectors $\mathbf{w}_i = \tilde{\boldsymbol{y}}(\varphi(z_i)) - \tilde{\boldsymbol{y}}(\varphi(z_0))$. From Eq. (3.11), we get the matrix equation $\boldsymbol{A} \cdot W = Z$. If the $\mathbf{v}_i$ are linearly independent, then W^{-1} exists, and $\boldsymbol{A} = Z \cdot W^{-1}$; thus, all the $\boldsymbol{A}_{ij}$ can be written as rational functions of k.

So the only possibility for not succeeding in writing the $\boldsymbol{A}_{ij}$ in terms of k, is that we fail to find n vectors $\mathbf{v}_i$ which are linearly independent. In this case, for any choosing of distinct complex numbers $z_0, z_1, \dots, z_{n-1} \in S^1$, the vector $\tilde{\boldsymbol{x}}(z) - \tilde{\boldsymbol{x}}(z_0)$ is linearly dependent with the $\mathbf{v}_i = \tilde{\boldsymbol{x}}(z_i) - \tilde{\boldsymbol{x}}(z_0)$, for $i = 1, \dots, n-1$. In turn, this implies that there exist functions $\lambda_1(z), \dots, \lambda_n(z)$ such that

$$\lambda_1(z)\mathbf{v}_1 + \ldots + \lambda_{n-1}(z)\mathbf{v}_{n-1} + \lambda_n(z)(\tilde{\boldsymbol{x}}(z) - \tilde{\boldsymbol{x}}(z_0)) = \mathbf{0}$$

for $z \in S^1$. But then $\tilde{\boldsymbol{x}}(z)$ belongs to the hyperplane through $\tilde{\boldsymbol{x}}(z_0)$, spanned by $\mathbf{v}_1, \dots, \mathbf{v}_{n-1}$, i.e., $\mathcal{C}$ is contained in a hyperplane. ∎

The proof of Lemma 3.3 suggests a strategy to write $\boldsymbol{A}$, and then $\boldsymbol{b}$, in terms of k by substituting random values $z \in S^1$ in Eq. (3.10). However, in order to write $\boldsymbol{A}, \boldsymbol{b}$ in terms of k we can proceed directly from Eq. (3.10). To make the process more clear, let us write

$$\tilde{\boldsymbol{x}}(z) = \left(\frac{\widehat{P}_1(z)}{z^N}, \ldots, \frac{\widehat{P}_n(z)}{z^N} \right). \tag{3.12}$$

The $\widehat{P}_i(z)$ are polynomials of degree at most $2N$, although there must be some i for which the degree of $\widehat{P}_i(z)$ is precisely $2N$. For this reason, some of the $\widehat{P}_i(z)$, but not all of them, can have z as a factor, with some multiplicity. Also, let us write

$$\tilde{\boldsymbol{y}}(z) = \left(\frac{\widehat{Q}_1(z)}{z^N}, \ldots, \frac{\widehat{Q}_n(z)}{z^N} \right). \tag{3.13}$$

Again, the $\widehat{Q}_i(z)$ are polynomials of degree at most $2N$, and not all of them can have z as a factor with some multiplicity. Furthermore, by Theorem 3.1 we have $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$. Thus, we get

$$\tilde{\boldsymbol{y}}(\varphi(z)) = \left(\frac{Q_1(k,z)}{z^N}, \ldots, \frac{Q_n(k,z)}{z^N} \right), \tag{3.14}$$

where the $Q_i(k,z)$ are polynomials in z, of degree $2N$, with coefficients polynomially depending on k, regardless of whether $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$. Also, for $i = 1, \ldots, n$ let us write

$$\begin{array}{rcl} \widehat{P}_i(z) & = & \alpha_0^{(i)} + \alpha_1^{(i)} z + \ldots + \alpha_N^{(i)} z^N + \cdots + \alpha_{2N}^{(i)} z^{2N}, \\ Q_i(k,z) & = & \beta_0^{(i)}(k) + \beta_1^{(i)}(k) z + \ldots + \beta_N^{(i)}(k) z^N + \cdots + \beta_{2N}^{(i)}(k) z^{2N}, \end{array} \tag{3.15}$$

where the coefficients of $Q_i(k,z)$, seen as a polynomial in z, are polynomials in k of degree at most $2N$.

Now Eq. (3.10) can be written as

$$\boldsymbol{A} \cdot \left(\frac{\widehat{P}_1(z)}{z^N}, \dots, \frac{\widehat{P}_n(z)}{z^N} \right)^T + \boldsymbol{b} = \left(\frac{Q_1(k,z)}{z^N}, \dots, \frac{Q_n(k,z)}{z^N} \right)^T . \tag{3.16}$$

Multiplying by z^N, we get

$$\boldsymbol{A} \cdot \left(\widehat{P}_1(z), \dots, \widehat{P}_n(z) \right)^T + z^N \boldsymbol{b} = \left(Q_1(k,z), \dots, Q_n(k,z) \right)^T . \tag{3.17}$$

From Eq. (3.17), equating the coefficients of the terms in z^ℓ, $\ell \neq N$, at both sides of the equation, we get linear equations

$$\boldsymbol{A}_{i1} \alpha_\ell^{(1)} + \cdots + \boldsymbol{A}_{in} \alpha_\ell^{(n)} = \beta_\ell^{(i)}(k) \tag{3.18}$$

where $i = 1, \dots, n$, $\ell = 0, 1, \dots, N-1, N+1, \dots, 2N$. Thus, we reach $2Nn$ linear equations of this type. Additionally, also from Eq. (3.17), equating the coefficients of the terms in z^N at both sides of the equation we get linear equations

$$\boldsymbol{A}_{i1} \alpha_N^{(1)} + \cdots + \boldsymbol{A}_{in} \alpha_N^{(n)} + b_i = \beta_N^{(i)}(k) \tag{3.19}$$

where $i = 1, \dots, n$. We get n linear equations of this type. Putting together the equations Eq. (3.18) and Eq. (3.19), we obtain a linear system $\mathcal{S}$, whose unknowns are the n^2 entries $\boldsymbol{A}_{ij}$ of the matrix $\boldsymbol{A}$, and the n coordinates of the vector $\boldsymbol{b}$, that must be consistent for some values $k \in S^1$ in the event that the curves $\mathcal{C}, \mathcal{D}$ are affinely equivalent. We refer to $\mathcal{S}$ as the linear system *associated* with Eq. (3.10). Moreover, the coefficient matrix $\mathcal{A}$ of the system $\mathcal{S}$ has the following block structure:

$$\mathcal{A} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{B}_2 & \mathbf{1} \end{bmatrix} \tag{3.20}$$

The block $\mathbf{B}_1$ is a block diagonal $2Nn \times n^2$ matrix, and consists of n copies of the

$2N \times n$ submatrix

$$\begin{bmatrix} \alpha_0^{(1)} & \cdots & \alpha_0^{(n)} \\ \vdots & \ddots & \vdots \\ \alpha_{2N}^{(1)} & \cdots & \alpha_{2N}^{(n)} \end{bmatrix} \tag{3.21}$$

where the row corresponding to the subindex N is missing. The block $\mathbf{B}_2$ is also block diagonal with dimension $n \times n^2$ and consists of n copies of the row matrix

$$\begin{bmatrix} \alpha_N^{(1)} & \cdots & \alpha_N^{(n)} \end{bmatrix}. \tag{3.22}$$

The block $\mathbf{0}$ is corresponds to a $2Nn \times n$ null matrix, and the block $\mathbf{1}$ is the identity matrix of dimension n.

In particular, notice that the number of linear equations we get is $2Nn + n = (2N+1)n$, and the number of unknowns is $n^2 + n$, so $\mathcal{A} \in \mathcal{M}_{(2N+1)n \times (n^2+n)}$.

Lemma 3.4. *If $\mathcal{C}$ is not contained in a hyperplane, then* $\operatorname{rank}(\mathcal{A}) = n^2 + n$.

Proof. We know that $\mathcal{A} \in \mathcal{M}_{(2N+1)n \times (n^2+n)}$. Furthermore, since by hypothesis $\mathcal{C}$ is not contained in a hyperplane, by Lemma 3.3 the system $\mathcal{S}$ is consistent and determined. Since $\mathcal{S}$ has $n^2 + n$ unknowns, then $\operatorname{rank}(\mathcal{A}) = n^2 + n$. ■

Since $\mathcal{A} \in \mathcal{M}_{(2N+1)n \times (n^2+n)}$, Lemma 3.4 implies that $(2N+1)n \geq (n^2 + n)$, i.e., $2N \geq n$. The next result shows that this is exactly what happens in the case when $\mathcal{C}$ is not contained in a hyperplane.

Lemma 3.5. *If $\mathcal{C}$ is not contained in a hyperplane, then $2N \geq n$.*

Proof. The vector $\tilde{\boldsymbol{x}}(z)$ is parallel to the vector

$$\tilde{\boldsymbol{x}}^\star(z) = z^N \tilde{\boldsymbol{x}}(z) = (\widehat{P}_1(z), \ldots, \widehat{P}_n(z)).$$

In turn, we can write $\tilde{\boldsymbol{x}}^\star(z)$ as

$$\tilde{\boldsymbol{x}}^\star(z) = \mathbf{a}_0 + \mathbf{a}_1 z + \cdots + \mathbf{a}_{2N} z^{2N},$$

where $\mathbf{a}_j \in \mathbb{R}^{2N}$ for $j = 0, \ldots, 2N$. If $2N < n$, then $\tilde{\boldsymbol{x}}^\star(z)$, and therefore, for every $z \in S^1$, $\tilde{\boldsymbol{x}}(z)$ belongs to a subspace of $\mathbb{R}^n$ of dimension less or equal than $n-1$. Thus, $\mathcal{C}$ is contained in a hyperplane. ■

Since $\mathcal{A} \in \mathcal{M}_{(2N+1)n \times (n^2+n)}$ and $\text{rank}(\mathcal{A}) = n^2 + n$ by Lemma 3.4, all the columns of $\mathcal{A}$ are linearly independent. However, using the structure of the matrix $\mathcal{A}$ (see Eq. (3.20)), we can also find the rows of $\mathcal{A}$ which are linearly independent. Indeed, observe that all the rows of $\mathcal{A}$ corresponding to the block $\mathbf{B}_2$ are linearly independent. Furthermore, no linear combination of rows of $\mathcal{A}$ corresponding to $\mathbf{B}_2$ and $\mathbf{B}_1$ can produce the zero vector. So linear combinations of rows of $\mathcal{A}$ leading to the zero vector must come from the rows corresponding to $\mathbf{B}_1$ only, and in fact from rows corresponding to the same block of $\mathbf{B}_1$. By the structure of $\mathcal{A}$ and since $\mathbf{B}_1$ is block diagonal, it suffices to find the rows $L_1, ..., L_p$ of the submatrix in Eq. (3.21) which are linearly independent: all the rows of $\mathcal{A}$ corresponding to the L_j (notice that by the block structure of $\mathcal{A}$, there are n rows of $\mathcal{A}$ for each L_j) must also be linearly independent. Since there are n blocks of the submatrix in Eq. (3.21), this yields $p \cdot n$ independent rows, plus the n rows corresponding to $\mathbf{B}_2$. Since $p \cdot n + n = n^2 + n$, we deduce that $p = n$, so the submatrix in Eq. (3.21) must have full rank.

Thus, in practice, in order to solve the system $\mathcal{S}$ it suffices to perform Gaussian elimination on the submatrix in Eq. (3.21), which has full rank. This way we can identify $L_1, ..., L_n$. The n^2 equations of $\mathcal{S}$ corresponding to $L_1, ..., L_n$, plus the equations of $\mathcal{S}$ corresponding to the last n rows of the matrix $\mathcal{A}$, yield the linearly independent equations of the system $\mathcal{S}$. We denote by $\mathcal{S}_0$ the set of these linear equations. Then, solving $\mathcal{S}_0$ allows one to write the entries of $\boldsymbol{A}$ and the elements of $\boldsymbol{b}$ in terms of k. Thus, we have the following result.

Proposition 3.1. *Assume that $\mathcal{C} \subset \mathbb{R}^n$ is not contained in a hyperplane. Then $\mathcal{S}_0$ is a set of linearly independent $n^2 + n$ linear equations, and the solution set of $\mathcal{S}_0$ coincides* *with the solution set of the system* $\mathcal{S}$.

Notice also that since the system $\mathcal{S}$ has $(2N+1)n$ equations, after solving the system $\mathcal{S}_0$ we still have $(2N+1)n-(n^2+n)=(2N-n)n$ linear equations of $\mathcal{S}$ that we have not used. Substituting the entries of the matrix $\boldsymbol{A}$ and the vector $\boldsymbol{b}$ in terms of k, into these remaining equations, we get polynomial conditions in k. We denote by r the number of nonzero polynomial conditions that we get by proceeding this way; when r is greater than zero, we denote these polynomial conditions by

$$g_1(k),\ldots,g_r(k). \tag{3.23}$$

Then we have the following result.

Proposition 3.2. *Assume that* $\mathcal{C}\subset\mathbb{R}^n$ *is not contained in a hyperplane. The linear system* $\mathcal{S}$ *associated with Eq.* (3.9) *provides* $0\le r\le(2N-n)n$ *nonzero polynomial conditions in* k*, of degree bounded by* $2N$.

Proof. From preceding considerations it is clear that $(2N+1)n-(n^2+n)=(2N-n)n$ is an upper bound for r, where $2N-n\ge 0$ because of Lemma 3.5. Since the constant terms of $\mathcal{S}$, i.e., the $\beta_\ell^{(i)}(k)$ (see Eq. (3.18) and Eq. (3.19)), are polynomials in k of degree $\le 2N$, by Cramer's rule the $\boldsymbol{A}_{ij}$ and the components of $\boldsymbol{b}$ are polynomials of degree $\le 2N$. Thus, when substituted in the remaining equations of $\mathcal{S}$, we get polynomials in k of degree $\le 2N$. ■

Finally, we have the following theorem.

Theorem 3.2. *Let* $\mathcal{C},\mathcal{D}\subset\mathbb{R}^n$ *be two trigonometric curves, none of them contained in a hyperplane.*

(1) If $r=0$*, the curves* $\mathcal{C},\mathcal{D}$ *are related by infinitely many affine transformations.*

(2) If $r>0$*, the curves* $\mathcal{C},\mathcal{D}$ *are affinely equivalent if and only if the polynomials* $g_1(k),\ldots,g_r(k)$ *have a common root* $k\in S^1$*, i.e., if and only if the greatest common divisor* $\gcd(g_1(k),\ldots,g_r(k))$ *is not constant, and has a root* $k\in S^1$.

Proof. Let us see (1) first. If $r = 0$, after solving $\mathcal{S}_0$ we can write $\boldsymbol{A}$ and $\boldsymbol{b}$ in terms of k, and Eq. (3.10) is satisfied for all values of k. Hence, every affinity $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$, with $\boldsymbol{A} = \boldsymbol{A}(k)$ and $\boldsymbol{b} = \boldsymbol{b}(k)$, maps $\mathcal{C}$ onto $\mathcal{D}$. Statement (2) follows from Theorem 3.1 and all the preceding constructions. ∎

Corollary 3.4. *Any two trigonometric curves $\mathcal{C}, \mathcal{D}$ in $\mathbb{R}^n$ defined by rational complex parametrizations of degree N with $2N = n$, not contained in a hyperplane, are related by infinitely many affine equivalences.*

Proof. Since $2N = n$, by Proposition 3.2 the number r of polynomial conditions is $r = 0$. Then the result follows from statement (1) in Theorem 3.2. ∎

The preceding ideas are summarized in Algorithm `Affine-Trigonometric`.

The complexity of Algorithm `Affine-Trigonometric` is provided in the following proposition. Here we use the standard Big O notation $\mathcal{O}$ for the time complexity analysis, and the Soft O notation $\tilde{\mathcal{O}}$ to ignore logarithmic factors.

Proposition 3.3. *Let $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$ be two trigonometric curves of degree N, not contained in a hyperplane. The complexity of Algorithm* `Affine-Trigonometric` *is $\tilde{\mathcal{O}}(N^3)$.*

Proof. Writing $\boldsymbol{A}, \boldsymbol{b}$ in terms of k implies solving the linear system $\mathcal{S}$ stemming from Eq. (3.10). This can be done by applying Gaussian Elimination to the system $\mathcal{S}$. The coefficient matrix of this system is $\mathcal{A} \in \mathcal{M}_{(2N+1)n \times (n^2+n)}$. Since $2N \geq n$ because of Lemma 3.5, the rank of $\mathcal{A}$ is bounded by $(2N+1)n$ and thus the complexity of Gaussian Elimination on $\mathcal{S}$ is $\mathcal{O}(N^3 n^3)$ (see for instance [27]). Computing the polynomials $g_1(k), \ldots, g_r(k)$ does not increase the complexity. The degrees of the $g_i(k)$ are bounded by $2N$, and thus computing the gcd of the $g_i(k)$ can be done in $\tilde{\mathcal{O}}(N)$ time (see Corollary 11.6 in [113]). The roots of the gcd can be computed in $\tilde{\mathcal{O}}(N^3)$ time (see [29]), so we get an overall complexity of $\tilde{\mathcal{O}}(N^3)$. ∎

Notice that Proposition 3.3 refers to arithmetic operation counts rather than bit complexity, i.e., it treats coefficient arithmetic as constant-time.

Algorithm 2 Affine-Trigonometric

Require: Two trigonometric curves $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$, defined by simple parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$, of the same degree $2N$.

Ensure: The affine equivalences $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ between $\mathcal{C}, \mathcal{D}$.

1: Compute the rational complex parametrizations $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ associated with the curves.
2: Set $\varphi(z) = kz$
3: Solve the linear system $\mathcal{S}_0$, i.e., write $\boldsymbol{A}, \boldsymbol{b}$ in terms of k.
4: Substitute $\boldsymbol{A}, \boldsymbol{b}$ in terms of k into the remaining equations of $\mathcal{S}$, to find the number r of nonzero polynomial conditions.
5: **if** $r = 0$ **then**
6:     **return** $\mathcal{C}$ `and` $\mathcal{D}$ `are related by infinitely many affine equivalences.`
7: **else**
8:     Compute the polynomial conditions $g_1(k), \ldots, g_r(k)$
9:     Compute the complex roots $k \in S^1$ of the greatest common divisor of $g_1(k), \ldots, g_r(k)$.
10:     **return**, if any, the affine equivalences corresponding to the k found in the step before
11: **end if**
12: Set $\varphi(z) = \frac{k}{z}$, and repeat steps (3-11)
13: **if** no value $k \in S^1$ has been found, and $r \neq 0$ in both cases $\varphi(z) = kz$ and $\varphi(z) = \frac{k}{z}$ **then**
14:     **return** $\mathcal{C}$ `and` $\mathcal{D}$ `are not affinely equivalent.`
15: **end if**

Example 3.1. Let $\mathcal{C}$ and $\mathcal{D}$ be the plane trigonometric curves parametrized by $\boldsymbol{x}(t), \boldsymbol{y}(t)$ respectively, with $t \in [0, 2\pi]$, where

$$\begin{aligned}
x_1(t) &= -\frac{1}{3}\sin(3t) + \frac{2}{3}\cos(t),\\
x_2(t) &= -\sin(5t) - 2\sin(t) - \frac{1}{3}\cos(t),\\
y_1(t) &= -\frac{1}{6}\sin(5t) + \frac{1}{4}\sin(3t) - \frac{1}{3}\sin(t) - \frac{5}{9}\cos(t) + 4,\\
y_2(t) &= -\frac{\sqrt{3}}{2}\sin(5t) + \frac{2}{15}\sin(3t) - \sqrt{3}\sin(t) - \frac{8+5\sqrt{3}}{30}\cos(t) - 2.
\end{aligned}$$

The curves $\mathcal{C}$ and $\mathcal{D}$ are shown in Fig. 3.1.

The associated rational complex parametrizations are

$$\begin{aligned}
\tilde{\boldsymbol{x}}(z) &= \left(\frac{\mathbf{i}z^6 + 2z^4 + 2z^2 - \mathbf{i}}{6z^3}, \frac{3\mathbf{i}z^{10} - (1-6\mathbf{i})z^6 - (1+6\mathbf{i})z^4 - 3\mathbf{i}}{6z^5}\right),\\
\tilde{\boldsymbol{y}}(z) &= (\tilde{y}_1(z), \tilde{y}_2(z)),
\end{aligned}$$

where

$$\tilde{y}_1(z) = \frac{6\mathbf{i}z^{10} - 9\mathbf{i}z^8 - (20-12\mathbf{i})z^6 + 288z^5 - (20+12\mathbf{i})z^4 + 9\mathbf{i}z^2 - 6\mathbf{i}}{72z^5},$$

$$\tilde{y}_2(z) = \frac{15\mathbf{i}\sqrt{3}z^{10} - 4\mathbf{i}z^8 - (5\sqrt{3}+8-30\sqrt{3}\mathbf{i})z^6 - 120z^5 - (5\sqrt{3}+8+30\sqrt{3}\mathbf{i})z^4 + 4\mathbf{i}z^2 - 15\sqrt{3}\mathbf{i}}{60z^5}.$$

Here $N = 5$. We consider first the case $\varphi(z) = kz$.

Step 8 of Algorithm 2 provides the 12 polynomials below:

$$g_1(k) = -(18+168\mathbf{i})k^{10} + (27+54\mathbf{i})k^8 - (156-12\mathbf{i})k^6 + (156+12\mathbf{i})k^4 - (27-54\mathbf{i})k^2 + 18 + 36\mathbf{i},$$

$$g_2(k) = -(54-6\mathbf{i})k^{10} + (81+315\mathbf{i})k^8 - (128+168\mathbf{i})k^6 + (128-168\mathbf{i})k^4 - (81-9\mathbf{i})k^2 + 54 + 6\mathbf{i},$$

$$g_3(k) = -(36-174\mathbf{i})k^{10} + (54-45\mathbf{i})k^8 + (28-180\mathbf{i})k^6 - (28+180\mathbf{i})k^4 - (54-261\mathbf{i})k^2 + 36 - 30\mathbf{i},$$

$$g_4(k) = (36+30\mathbf{i})k^{10} - (54+261\mathbf{i})k^8 - (28-180\mathbf{i})k^6 + (28+180\mathbf{i})k^4 + (54+45\mathbf{i})k^2 - 36 - 174\mathbf{i},$$

$$g_5(k) = (54-6\mathbf{i})k^{10} - (81+9\mathbf{i})k^8 + (128+168\mathbf{i})k^6 - (128-168\mathbf{i})k^4 + (81-315\mathbf{i})k^2 - 54 - 6\mathbf{i},$$

$$g_6(k) = (18-36\mathbf{i})k^{10} - (27+54\mathbf{i})k^8 + (156-12\mathbf{i})k^6 - (156+12\mathbf{i})k^4 + (27-54\mathbf{i})k^2 - 18 + 168\mathbf{i},$$

$$\begin{aligned} g_7(k) = &-\sqrt{3}(45+420\mathbf{i})k^{10} + (12+24\mathbf{i})k^8 - (48+120\sqrt{3}+(24-165\sqrt{3})\mathbf{i})k^6 \\ &+ (48+120\sqrt{3}-(24-165\sqrt{3})\mathbf{i})k^4 - (12-24\mathbf{i})k^2 + 4\sqrt{3} + 90\sqrt{3}\mathbf{i}, \end{aligned}$$

$$\begin{aligned} g_8(k) = &-\sqrt{3}(135-15\mathbf{i})k^{10} + (36+140\mathbf{i})k^8 - (8+275\sqrt{3}+(72+15\sqrt{3})\mathbf{i})k^6 \\ &+ (8+275\sqrt{3}-(72+15\sqrt{3})\mathbf{i})k^4 - (36-4\mathbf{i})k^2 + 135\sqrt{3} + 15\sqrt{3}\mathbf{i}), \end{aligned}$$

$$\begin{aligned} g_9(k) = &-\sqrt{3}(90-435\mathbf{i})k^{10} + (24-20\mathbf{i})k^8 + (40-155\sqrt{3}-(48+180\sqrt{3})\mathbf{i})k^6 \\ &- (40-155\sqrt{3}+(48+180\sqrt{3})\mathbf{i})k^4 - (24-116\mathbf{i})k^2 + 90\sqrt{3} - 75\sqrt{3}\mathbf{i}), \end{aligned}$$

$$\begin{aligned} g_{10}(k) = &\sqrt{3}(90+75\mathbf{i})k^{10} - (24+116\mathbf{i})k^8 - (40-155\sqrt{3}-(48+180\sqrt{3})\mathbf{i})k^6 \\ &+ (40-155\sqrt{3}+(48+180\sqrt{3})\mathbf{i})k^4 + (24+20\mathbf{i})k^2 - 90\sqrt{3} - 435\sqrt{3}\mathbf{i}), \end{aligned}$$

$$\begin{aligned} g_{11}(k) = &\sqrt{3}(135-15\mathbf{i})k^{10} - (36+4\mathbf{i})k^8 + (8+275\sqrt{3}+(72+15\sqrt{3})\mathbf{i})k^6 \\ &- (8+275\sqrt{3}-(72+15\sqrt{3})\mathbf{i})k^4 + (36-140\mathbf{i})k^2 - 135\sqrt{3} - 15\sqrt{3}\mathbf{i}), \end{aligned}$$

$$\begin{aligned} g_{12}(k) = &\sqrt{3}(45-90\mathbf{i})k^{10} - (12+24\mathbf{i})k^8 + (48+120\sqrt{3}+(24-165\sqrt{3})\mathbf{i})k^6 \\ &- (48+120\sqrt{3}-(24-165\sqrt{3})\mathbf{i})k^4 + (12-24\mathbf{i})k^2 - 45\sqrt{3} + 420\sqrt{3}\mathbf{i}). \end{aligned}$$

The gcd of all of them is k^2-1. Thus, we get $k=\pm 1$, i.e., $\varphi_1(z) = z,\ \varphi_2(z) = -z$.

The mapping $\varphi_1(z)$ corresponds to the affine mapping $f_1(\mathbf{x}) = \boldsymbol{A}_1\mathbf{x} + \boldsymbol{b}_1$, where

$$\boldsymbol{A}_1 = \begin{pmatrix} -\dfrac{3}{4} & \dfrac{1}{6} \\ -\dfrac{2}{5} & \dfrac{\sqrt{3}}{2} \end{pmatrix}, \quad \boldsymbol{b}_1 = \begin{pmatrix} 4 \\ -2 \end{pmatrix}. \tag{3.24}$$

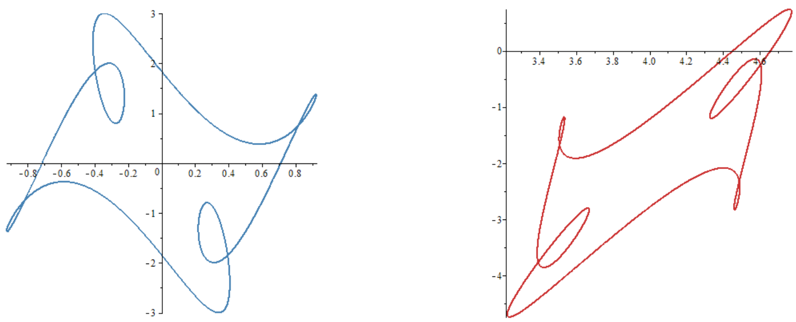


Figure 3.1: $\mathcal{C}$ (left) and $\mathcal{D}$ (right).

The mapping $\varphi_2(z)$ corresponds to the affine mapping $f_2(\mathbf{x}) = \boldsymbol{A}_2\mathbf{x} + \boldsymbol{b}_2$, where

$$\boldsymbol{A}_2 = \begin{pmatrix} \frac{3}{4} & -\frac{1}{6} \\ \frac{2}{5} & -\frac{\sqrt{3}}{2} \end{pmatrix}, \quad \boldsymbol{b}_2 = \begin{pmatrix} 4 \\ -2 \end{pmatrix}. \tag{3.25}$$

When $\varphi(z) = \dfrac{k}{z}$ we obtain no solution. Therefore, we conclude that $\mathcal{C}, \mathcal{D}$ are related by two affine mappings f_1, f_2. Notice that both $\mathcal{C}, \mathcal{D}$ have a nontrivial symmetry τ with respect to a point, which is the reason why we get two affine equivalences.

Example 3.2. Let $\mathcal{C}$ and $\mathcal{D}$ be the space trigonometric curves parametrized by $\boldsymbol{x}(t) = (x_1(t), x_2(t), x_3(t))$ and $\boldsymbol{y}(t) = (y_1(t), y_2(t), y_3(t))$ respectively, with $t \in [0, 2\pi]$, where

$$\begin{aligned}
x_1(t) &= \cos(2t) - 2\sin(2t) + 1,\\
x_2(t) &= -\cos(2t) - \sin(2t),\\
x_3(t) &= 2\cos(t) + 2\sin(t),\\
y_1(t) &= \frac{9}{5}\cos(2t) - \frac{21}{5}\sin(2t) - 6\cos(t) - 6\sin(t) + 3,\\
y_2(t) &= -\cos(2t) + 2\sin(2t) - 8\cos(t) - 8\sin(t) - 1\\
y_3(t) &= -2\cos(2t) - 11\sin(2t) + \sqrt{2}\cos(t) + \sqrt{2}\sin(t) + 4.
\end{aligned}$$

In this case, $N = 2$. The curves $\mathcal{C}, \mathcal{D}$ are shown in Fig. 3.2.

With $\varphi(z) = kz$, step 8 of Algorithm 2 provides three quadratics polynomials in k giving rise to

$$\varphi_1(z) = z,\ \varphi_2(z) = -z.$$

Hence, here we get two affine equivalences between $\mathcal{C}$ and $\mathcal{D}$, more precisely:

- $\varphi_1(z)$ corresponds to the affine equivalence $f_1(\mathbf{x}) = \boldsymbol{A}_1\mathbf{x} + \boldsymbol{b}_1$, where

$$\boldsymbol{A}_1 = \begin{pmatrix} 2 & \frac{1}{5} & -3 \\ -1 & 0 & -4 \\ 3 & 5 & \frac{\sqrt{2}}{2} \end{pmatrix},\quad \boldsymbol{b}_1 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}.$$

- $\varphi_2(z)$ corresponds to the affine equivalence $f_2(\mathbf{x}) = \boldsymbol{A}_2\mathbf{x} + \boldsymbol{b}_2$, where

$$\boldsymbol{A}_2 = \begin{pmatrix} 2 & \frac{1}{5} & -3 \\ -1 & 0 & -4 \\ 3 & 5 & -\frac{\sqrt{2}}{2} \end{pmatrix},\quad \boldsymbol{b}_2 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}.$$

With $\varphi(z) = \dfrac{k}{z}$, step 8 of Algorithm 2 also provides three quadratic polynomials. This time the gcd is $k^2 + 1$, and thus $k = \pm\mathbf{i}$. So we have

$$\varphi_3(z) = \frac{\mathbf{i}}{z},\ \varphi_4(z) = -\frac{\mathbf{i}}{z}.$$

- $\varphi_3(z)$ corresponds to the affine equivalence $f_3(\mathbf{x}) = \boldsymbol{A}_3\mathbf{x} + \boldsymbol{b}_3$, where

$$\boldsymbol{A}_3 = \begin{pmatrix} \frac{4}{5} & \frac{13}{5} & -3 \\ -\frac{1}{3} & -\frac{4}{3} & -4 \\ \frac{13}{3} & \frac{7}{3} & \frac{\sqrt{2}}{2} \end{pmatrix}, \quad \boldsymbol{b}_3 = \begin{pmatrix} \frac{11}{5} \\ -\frac{2}{3} \\ -\frac{1}{3} \end{pmatrix}.$$

- $\varphi_4(z)$ corresponds to the affine equivalence $f_4(\mathbf{x}) = \boldsymbol{A}_4\mathbf{x} + \boldsymbol{b}_4$, where

$$\boldsymbol{A}_4 = \begin{pmatrix} \frac{4}{5} & \frac{13}{5} & 3 \\ -\frac{1}{3} & -\frac{4}{3} & 4 \\ \frac{13}{3} & \frac{7}{3} & -\frac{\sqrt{2}}{2} \end{pmatrix}, \quad \boldsymbol{b}_4 = \begin{pmatrix} \frac{11}{5} \\ -\frac{2}{3} \\ -\frac{1}{3} \end{pmatrix}.$$

If we consider only the symmetries of $\mathcal{C}$, i.e., we directly apply the algorithm with $\mathcal{C} = \mathcal{D}$, we get that $\mathcal{C}$ has three symmetries; in particular:

- $\varphi_2(z)$ corresponds to a symmetry with respect to the xy-plane.
- $\varphi_3(z)$ corresponds to the axial symmetry $f_5(\mathbf{x}) = \boldsymbol{A}_5\mathbf{x} + \boldsymbol{b}_5$, where

$$\boldsymbol{A}_5 = \begin{pmatrix} \frac{1}{3} & \frac{4}{3} & 0 \\ \frac{2}{3} & -\frac{1}{3} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \boldsymbol{b}_5 = \begin{pmatrix} \frac{2}{3} \\ -\frac{2}{3} \\ 0 \end{pmatrix}.$$

- $\varphi_4(z)$ corresponds to the axial symmetry $f_6(\mathbf{x}) = \boldsymbol{A}_6\mathbf{x} + \boldsymbol{b}_6$, where

$$\boldsymbol{A}_6 = \begin{pmatrix} \frac{1}{3} & \frac{4}{3} & 0 \\ \frac{2}{3} & -\frac{1}{3} & 0 \\ 0 & 0 & -1 \end{pmatrix}, \quad \boldsymbol{b}_6 = \begin{pmatrix} \frac{2}{3} \\ -\frac{2}{3} \\ 0 \end{pmatrix}.$$

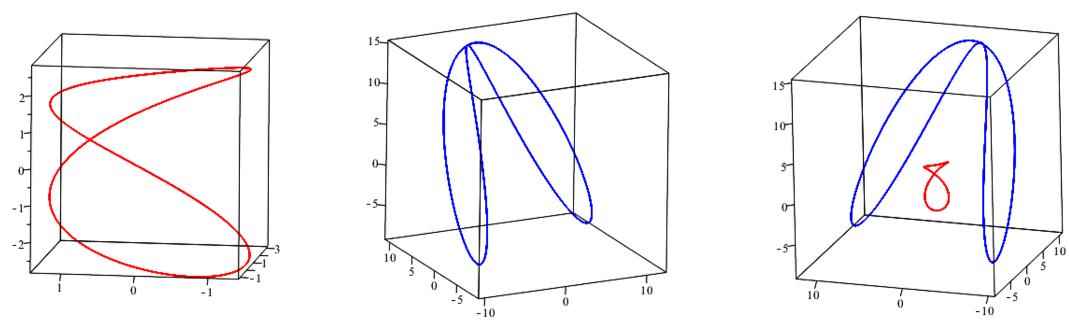


Figure 3.2: Curve $\mathcal{C}$ (left), curve $\mathcal{D}$ (middle) and $\mathcal{C}, \mathcal{D}$ together (right).

Remark 3.3. *In some cases, it can be computationally cheaper to solve the system $\mathcal{S}$ directly, which is polynomial in k although linear in the entries of $\boldsymbol{A}$ and the elements of $\boldsymbol{b}$, rather than computing the $g_1(k), \ldots, g_r(k)$, and find the roots of the gcd of these polynomials.*

3.3 Observations on the computation of projective equivalences

We consider now the detection of projective equivalences $f : \mathcal{C} \longrightarrow \mathcal{D}$ between trigonometric curves $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$, where f is as in Eq. (1.2). We present some considerations and results on this question, but we must say that we could not find any example, other than conics, of two trigonometric curves related by a projectivity that is not an affinity. In fact, we have tried to prove, although we have not succeeded, that two trigonometric curves, not conics, related by a projectivity must be affinely equivalent. This is somehow supported by the fact that, unlike affine transformations, the image of a trigonometric parametrization under a projectivity that is not an affinity is not trigonometric itself.

Regardless, we include two results that, we hope, may help in the future to either prove that projective equivalence implies affine equivalence in this case, or to produce a counterexample of this statement.

First, we can prove that for projectivities Eq. (3.10) also holds.

Proposition 3.4. *Let $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$ be two trigonometric curves, defined by two simple trigonometric parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$, with associated rational complex parametrizations $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$, and let $f : \mathcal{C} \to \mathcal{D}$ be a projective transformation with components as in Eq.* (1.2)*, such that $f(\mathcal{C}) = \mathcal{D}$. Then the relationship Eq.* (2.18) *holds, with $\varphi(z)$ a Möbius transformation, i.e., $\varphi(z) = \frac{az+b}{cz+d}$, $|ad - bc| \neq 0$, with $\varphi(S^1) = S^1$. Furthermore, the degrees of $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ must coincide.*

Proof. Let η be a Möbius transformation mapping S^1 into the real line. This transformation is invertible, and η^{-1} is also a Möbius transformation. Since $f, \tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ are invertible and their inverses are also rational transformations, the diagram

$$
\begin{tikzcd}
\mathcal{C} \arrow[r, "f"] & \mathcal{D} \\
S^1 \arrow[u, dashed, "\tilde{\boldsymbol{x}}"] \arrow[r, dashed, "\varphi"'] & S^1 \arrow[u, dashed, "\tilde{\boldsymbol{y}}"'] \\
\mathbb{R} \arrow[u, dashed, "\eta^{-1}"] \arrow[r, dashed, "\psi"'] & \mathbb{R} \arrow[u, dashed, "\eta^{-1}"']
\end{tikzcd}
\tag{3.26}
$$

is commutative, and ψ is a birational mapping of the real line, and therefore, a Möbius transformation. Since $\varphi = \eta^{-1} \circ \psi \circ \eta$, then φ is a composition of Möbius transformations, and therefore φ is a Möbius transformation itself. By the commutativity of the diagram, we get $f \circ \tilde{\boldsymbol{x}} = \tilde{\boldsymbol{y}} \circ \varphi$.

Finally, let us see that the degrees of $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ must coincide. In order to see this, we observe that the mappings $\tilde{\boldsymbol{x}} \circ \eta^{-1}$ and $\tilde{\boldsymbol{y}} \circ \eta^{-1}$ are injective for almost all points (because they are composites of two mappings which are also injective for almost all points). Since the degree of the composition of two rational functions is multiplicative [21] and the degree of η^{-1} is one, the degree of $\tilde{\boldsymbol{x}} \circ \eta^{-1}$ is $2N$. Since projectivities preserve the degree, we deduce that the degree of $\tilde{\boldsymbol{y}} \circ \eta^{-1}$, and therefore the degree of $\tilde{\boldsymbol{y}}(z)$, must be $2N$ as well. ∎

In this case, we were not able to prove that $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$ as it happens for affinities. However, the following result shows that if indeed $\varphi(z)$ is of these types, then the projectivity must be an affinity.

Proposition 3.5. *Let $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$ be two trigonometric curves, not contained in hyperplanes, defined by two simple trigonometric parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$, with associated rational complex parametrizations $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$. Let $2N$ be the degree of $\tilde{\boldsymbol{x}}$, and let $f : \mathcal{C} \to \mathcal{D}$ be a projective transformation with components as in Eq. (1.2). If the Möbius function $\varphi(z)$ in Proposition 3.4 satisfies that $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$, for $k \in S^1$, then f is an affine transformation.*

Proof. Let us write $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ as in Eq. (3.12) and Eq. (3.13). If $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$ then $\tilde{\boldsymbol{y}}(\varphi(z))$ can be written as in Eq. (3.14). Since by Proposition 3.4 the degrees of $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ coincide, we get

$$\frac{a_{i1}\widehat{P}_1(z) + \cdots + a_{in}\widehat{P}_n(z) + b_i z^N}{a_{n+1,1}\widehat{P}_1(z) + \cdots + a_{n+1,n}\widehat{P}_n(z) + b_{n+1} z^N} = \frac{Q_i(k,z)}{z^N} \tag{3.27}$$

for $i = 1, \ldots, n$. From Eq. (3.27) we get

$$a_{n+1,1}\widehat{P}_1(z) + \cdots + a_{n+1,n}\widehat{P}_n(z) + b_{n+1} z^N = z^N. \tag{3.28}$$

If $a_{n+1,1}, \ldots, a_{n+1,n}$ are not all of them zero, then Eq. (3.28) implies that

$$a_{n+1,1}\frac{\widehat{P}_1(z)}{z^N} + \cdots + a_{n+1,n}\frac{\widehat{P}_n(z)}{z^N} + b_{n+1} - 1 = 0,$$

which means that $\mathcal{C}$ is contained in the hyperplane $a_{n+1,1}x_1 + \ldots + a_{n+1,1}x_n + b_{n+1} - 1 = 0$. However, by hypothesis this cannot happen, and we conclude that $a_{n+1,1} = \cdots = a_{n+1,n} = 0$, i.e., f is an affinity. ∎

3.4 Approximate affine equivalences

In this section we consider the case when the curves $\mathcal{C}$ and $\mathcal{D}$ are defined by means of simple trigonometric parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$ as in Eq. (3.1) and Eq. (3.2), but where the coefficients of $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ are given with finite precision, i.e., as floating-point numbers. We denote them here by $\tilde{\mathcal{C}}, \tilde{\mathcal{D}}$. In this case, and even if the curves $\tilde{\mathcal{C}}$ and $\tilde{\mathcal{D}}$ are very close to being related by an affinity, applying the same procedure as in the exact case yields polynomial conditions $g_1(k), \ldots, g_r(k)$ with a constant gcd, so even though these polynomials have some roots which are very close to each other, no common root of $g_1(k), \ldots, g_r(k)$ is computed.

Thus, here we focus not on the affine equivalences between $\tilde{\mathcal{C}}$ and $\tilde{\mathcal{D}}$, but on *approximate* affine equivalences. In order to do it, we proceed as in the exact case to compute $g_1(k), \ldots, g_r(k)$, and then we find the *approximate* common roots of $g_1(k), \ldots, g_r(k)$.

A possibility to do this is to compute an *approximate gcd* of $g_1(k), \ldots, g_r(k)$.

Definition 3.1. *Given two polynomials $\tilde{p}(k), \tilde{q}(k)$, we say that $\xi(k)$ is an* approximate gcd *or an ϵ-*gcd *of $\tilde{p}(k), \tilde{q}(k)$, if $\xi(k)$ is the exact gcd of two polynomials $p(k), q(k)$, where $\|p - \tilde{p}\|$ and $\|q - \tilde{q}\|$ are both less than ϵ, with ϵ close to zero and $\| \bullet \|$ a certain norm. The value ϵ is called the* tolerance*, and must be fixed in advance.*

This definition can be easily generalized to the case of three or more polynomials. Other versions of this definition can be found, for instance, in [68]. Furthermore, there are methods to evaluate how close two polynomials $p(k), q(k)$ are to having a nontrivial approximate gcd: see for instance [30, 31], and the command `distanceToCommonDivisors` of the package SNAP in Maple. The bibliograhy on the computation of approximate gcds is vast; one can check, among many others, [68, 65, 81, 82, 121, 122]. In our case, we use the `uvGCD` method described in [122]; one can find a publicly available MATLAB implementation of this method in [123].

When there are more than two polynomials, we can compute the gcd of only two of them and then check which of these solutions correspond to approximate roots of the remaining polynomials. Further refinement can be performed using numeric optimization (see also [122]), although we did not use this in our experiments.

The whole procedure is given in `Approximate-Affine-Trigonometric`; this algorithm is essentially equal to the algorithm `Affine-Trigonometric`, replacing exact gcds by approximate gcds.

Algorithm 3 Approximate-Affine-Trigonometric

Require: Two trigonometric curves $\tilde{\mathcal{C}}, \tilde{\mathcal{D}} \subset \mathbb{R}^n$, given by approximate parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$, of the same degree $2N$, and a tolerance ϵ.
Ensure: The approximate affine equivalences $\tilde{f}(\mathbf{x}) = \tilde{\boldsymbol{A}}\mathbf{x} + \tilde{\boldsymbol{b}}$ between $\tilde{\mathcal{C}}, \tilde{\mathcal{D}}$.
1: Compute the rational complex parametrizations $\tilde{\boldsymbol{x}}(z), \tilde{\boldsymbol{y}}(z)$ associated with the curves.
2: Set $\varphi(z) = kz$
3: Solve the linear system $\mathcal{S}_0$, i.e., write $\boldsymbol{A}, \boldsymbol{b}$ in terms of k.
4: Substitute $\boldsymbol{A}, \boldsymbol{b}$ in terms of k into the remaining equations of $\mathcal{S}$, to find the number r of nonzero polynomial conditions.
5: Compute the polynomial conditions $g_1(k), \ldots, g_r(k)$
6: Use the `uvGCD` method with the tolerance ϵ to compute the approximate complex roots $k \in S^1$ of the approximate gcd of $g_1(k), \ldots, g_r(k)$.
7: **return**, if any, the approximate affine equivalences corresponding to the k found in the step before.
8: Set $\varphi(z) = \frac{k}{z}$, and repeat steps (3-7)
9: **if** no approximate gcd has been found, and $r \neq 0$ in both cases $\varphi(z) = kz$ and $\varphi(z) = \frac{k}{z}$ **then**
10: **return** `There are not approximate affine equivalences between` $\tilde{\mathcal{C}}$ `and` $\tilde{\mathcal{D}}$.
11: **end if**

Of course, the remaining question is how to choose the tolerance ϵ. We tried to derive a bound for ϵ from a bound on the error in the coefficients, but we did not succeed. This is therefore left here as an open problem.

The following examples illustrates the method.

Example 3.3. Let us consider the space trigonometric curves $\mathcal{C}$ and $\mathcal{D}$ parametrized by

$$\boldsymbol{x}(t) = (5\sin(t), -\cos(t) + \sin(3t), \cos(3t) - 2\sin(4t))$$

and $\boldsymbol{y}(t) = (y_1(t), y_2(t), y_3(t))$, with $t \in [0, 2\pi]$, where

$$\begin{aligned}
y_1(t) &= 6\sin(4t) - 3\cos(3t) + 5\sin(3t) - 5\cos(t) - 30\sin(t) + 1,\\
y_2(t) &= -8\sin(4t) + 4\cos(3t) - 5\sin(t),\\
y_3(t) &= -2\sin(3t) + 5\sin(t) + 2\cos(t) + 1.
\end{aligned}$$

Its complex forms are given by

$$\tilde{\boldsymbol{x}}(z) = \left(-\frac{5\mathbf{i}(z^2-1)}{2z}, -\frac{z^6+z^4+z^2-\mathbf{i}}{2z^3}, \frac{2\mathbf{i}z^8+z^7+z-2\mathbf{i}}{2z^4}\right)$$

and $\tilde{\boldsymbol{y}}(z) = (\tilde{y}_1(z), \tilde{y}_2(z), \tilde{y}_3(z))$ where

$$\begin{aligned}
\tilde{y}_1(z) &= -\frac{6\mathbf{i}z^8 + (3+5\mathbf{i})z^7 + (5-3\mathbf{i})z^5 - 2z^4 + (5+3\mathbf{i})z^3 + (3-5\mathbf{i})z - 6\mathbf{i}}{2z^4},\\
\tilde{y}_2(z) &= \frac{8\mathbf{i}z^8 + 4z^7 + 5\mathbf{i}z^5 - 5\mathbf{i}z^3 + 4z - 8\mathbf{i}}{2z^4},\\
\tilde{y}_3(z) &= \frac{2\mathbf{i}z^6 + (2-5\mathbf{i})z^4 + 2z^3 + (2+5\mathbf{i})z^2 - 2i}{2z^3}.
\end{aligned}$$

Here, $N = 4$. The curves $\mathcal{C}, \mathcal{D}$ are shown in Fig. 3.3 and are related by one affine equivalence, namely $f_1(\mathbf{x}) = \boldsymbol{A}_1\mathbf{x} + \boldsymbol{b}_1$ corresponding to $\varphi_1(z) = z$, where

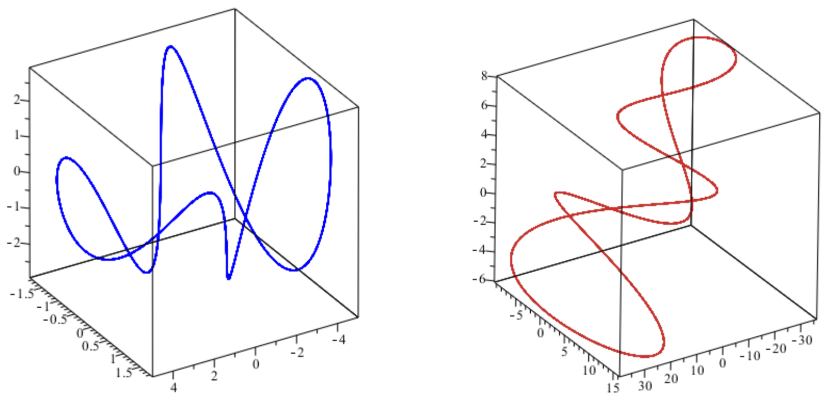


Figure 3.3: C_1 (left) and C_2 (right).

$$\boldsymbol{A}_1 = \begin{pmatrix} -6 & 5 & -3 \\ -1 & 0 & 4 \\ 1 & -2 & 0 \end{pmatrix}, \quad \boldsymbol{b}_1 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}.$$

Now we apply a random perturbation of order 10^{-4} to all the coefficients of the parameterizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$, and we seek approximate affine equivalences between the resulting curves $\tilde{\mathcal{C}}$ and $\tilde{\mathcal{D}}$. Proceeding as in Algorithm 1, for $\varphi(z) = kz$ we get 9 polynomial equations in k of degree 8, two of them are

$$\begin{aligned} g_1(k) &\approx -3.000029k^8 + (2.9999973 + 4.999936\mathbf{i})k^7 - (15.000037 + 2.50001\mathbf{i})k^5 \\ &\qquad + (15.000037 - 2.50001\mathbf{i})k^3, \\ g_2(k) &\approx 3.999956k^8 - 3.99996k^7 - 2.50002k^5 + 2.50002k^3, \end{aligned}$$

whose approximate gcd, using the `uvGCD` method and with a tolerance $\epsilon = 10^{-5}$, is

$$\xi(k) \approx (6 \cdot 10^{-10} - 2.53 \cdot 10^{-9}\mathbf{i})k^4 + (9.3 \cdot 10^{-10} + 3.9 \cdot 10^{-10}\mathbf{i})k^3(1.4 \cdot 10^{-9} - 2.2 \cdot 10^{-9}\mathbf{i})k^2 \\ + (2.474073 - 1.298621\mathbf{i})k - 2.474077 + 1.298614\mathbf{i}.$$

Thus, we get four values for k, but only $k \approx 0.99999985 + 0.00000247\mathbf{i}$ corresponds to an approximate solution. Notice that his modulus is close to one. This value of k give rise to one approximate affine equivalence, namely, $\tilde{f}(\mathbf{x}) = \tilde{\boldsymbol{A}}\mathbf{x} + \tilde{\boldsymbol{b}}$, where

$$\tilde{\boldsymbol{A}} \approx \begin{pmatrix} -6.00002 - 3.35 \cdot 10^{-5}\mathbf{i} & 4.99992 + 6.78 \cdot 10^{-5}\mathbf{i} & -3.000027 - 2.96 \cdot 10^{-5}\mathbf{i} \\ -1.000009 - 4.93 \cdot 10^{-7}\mathbf{i} & 0.000009 + 4.77 \cdot 10^{-6}\mathbf{i} & 3.99995 + 3.94 \cdot 10^{-5}\mathbf{i} \\ 1.000008 - 5.42 \cdot 10^{-6}\mathbf{i} & -2.00003 - 1.48 \cdot 10^{-5}\mathbf{i} & 0 \end{pmatrix},$$

$$\tilde{\boldsymbol{b}} \approx \begin{pmatrix} 0.9999061779 \\ 0 \\ 0.9999050456 \end{pmatrix}.$$

Here we have that $\dfrac{\|\boldsymbol{b} - \tilde{\boldsymbol{b}}\|}{\|\boldsymbol{b}\|} \approx 0.000094$ and $\dfrac{\|\boldsymbol{A} - \tilde{\boldsymbol{A}}\|_2}{\|\boldsymbol{A}\|_2} \approx 0.000015$ where $\|\cdot\|$ is the Euclidean norm of vectors and $\|\cdot\|_2$ is the spectral norm (the largest singular value) of matrices.

With $\varphi(z) = k/z$ we get no solution.

3.4.1 Computing the distance between $\tilde{f}(\tilde{\mathcal{C}})$ and $\tilde{\mathcal{D}}$

Now, after proceeding as before, we have tentative approximate similarities $\tilde{f}$, but we still need to test whether the curves $\tilde{\mathcal{C}}$ and $\tilde{\mathcal{D}}$ are approximately similar. In order to do this, given a tentative similarity $\tilde{f}$ we must evaluate whether $\tilde{f}(\tilde{\mathcal{C}})$ is close to $\tilde{\mathcal{D}}$. The best way to do this is to compute the *Hausdorff distance* between $\tilde{f}(\tilde{\mathcal{C}})$ and $\tilde{\mathcal{D}}$.

Definition 3.2. *The Hausdorff distance between two given objects* $\mathcal{O}_1, \mathcal{O}_2 \in \mathbb{R}^n$ *(see [22, 50]) is given by*

$$D_H(\mathcal{O}_1, \mathcal{O}_2) = \max\{\max_{P\in\mathcal{O}_1}\min_{Q\in\mathcal{O}_2}\|P-Q\|, \max_{Q\in\mathcal{O}_2}\min_{P\in\mathcal{O}_1}\|P-Q\|\}.$$

However, an algorithm to compute the Hausdorff distance between non-rational curves is absent, and even for rational curves the computation is complicated and slow (see [50]). So, instead, we present a different, heuristic approach to evaluate the distance between the curves. We work from the rational complex parametrizations.

We proceed in the following way to evaluate the closeness between $\tilde{f}(\tilde{\mathcal{C}})$ and $\tilde{\mathcal{D}}$:

- We consider a uniform distribution on the interval $[0, 2\pi]$, and then pick a random sample $\mathbb{S}$. Let n be the size of $\mathbb{S}$.
- We substitute each $t \in \mathbb{S}$ into $z = e^{\mathbf{i}t}$. Let $\mathcal{P}$ be the set of all the points obtained after each substitution. Notice that this is equivalent to computing a partition of S^1.
- If $\tilde{f}(\tilde{\mathcal{C}})$ is close to $\tilde{\mathcal{D}}$, then for $z \in \mathcal{P}$, the point $\tilde{\boldsymbol{A}}\tilde{\boldsymbol{x}}(z) + \tilde{\boldsymbol{b}}$ should be close to the point $\tilde{\boldsymbol{y}}(\varphi(z))$. Hence, for each $z_i \in \mathcal{P}$ we compute the distance d_i between $\tilde{\boldsymbol{A}}\tilde{\boldsymbol{x}}(z_i) + \tilde{\boldsymbol{b}}$ and $\tilde{\boldsymbol{y}}(\varphi(z_i))$ for $i = 1, \ldots, n$.
- Finally, we consider the arithmetic mean of all the d_i, and we divide it by the length of $\tilde{\mathcal{D}}$, in order to compare its value with the size of the curve; we represent by ν the resulting number.

For instance, in Ex. 3.3, the distance between $\tilde{f}(\tilde{\mathcal{C}})$ and $\tilde{\mathcal{D}}$ is $\nu \approx 1.3 \cdot 10^{-6}$ using a sample of 100 points.

3.4.2 Experimentation for the approximate case

We have implemented the method described above with the help of the computer algebra systems MATLAB R2020b and Maple 18, using a tolerance for the computation of approximate gcds of $\epsilon = 10^{-5}$. In Fig. 3.4 we show the CPU time in seconds for some representative examples of growing degrees. One can observe that for these data the quadratic polynomial $P(N) = 0.1517N^2 - 0.4645N + 6.3948$ fits very well, with a coefficient of determination (R^2) equal to 98.72 %.

The features of some of the examples used in Fig. 3.4 using the `uvGCD` method are provided in Table 3.1. In all these examples we considered two curves $\mathcal{C}, \mathcal{D}$ where $\mathcal{D}$ was the result of applying to $\mathcal{C}$ the affine transformation $f(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$, where

$$\boldsymbol{A} = \begin{pmatrix} 2 & 1/5 & -3 \\ -1 & 0 & -4 \\ 3 & 5 & \sqrt{3} \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 3 \\ 1 \\ -2 \end{pmatrix},$$

and introducing afterwards a perturbation of order 10^{-4}. The transformation computed by our method is denoted by $\tilde{f}(\boldsymbol{x}) = \tilde{\boldsymbol{A}}\boldsymbol{x} + \tilde{\boldsymbol{b}}$.

In Table 3.1 we can see the degree N, and the values of $\dfrac{\|\boldsymbol{A} - \tilde{\boldsymbol{A}}\|_2}{\|\boldsymbol{A}\|_2}$ and $\dfrac{\|\boldsymbol{b} - \tilde{\boldsymbol{b}}\|}{\|\boldsymbol{b}\|}$ for each example, which measure the relative error in each case.

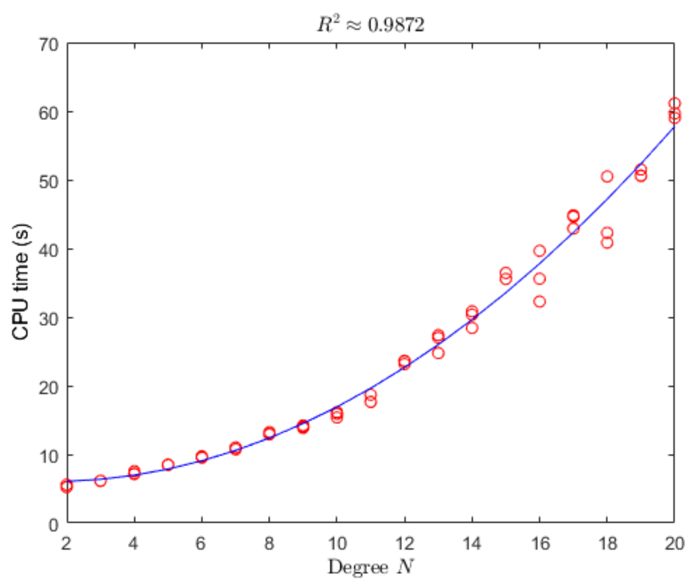


Figure 3.4: CPU time versus degree

Degree N	$\dfrac{\Vert \boldsymbol{A}-\tilde{\boldsymbol{A}}\Vert_2}{\Vert \boldsymbol{A}\Vert_2}$	$\dfrac{\Vert \boldsymbol{b}-\tilde{\boldsymbol{b}}\Vert}{\Vert \boldsymbol{b}\Vert}$	CPU time (secs.)	ν
3	0.00004	0.0006	6.0837	$2.8 \cdot 10^{-6}$
5	0.00001	0.0002	8.6786	$1.1 \cdot 10^{-6}$
8	0.0003	0.001	13.6119	$9.8 \cdot 10^{-6}$
9	0.00003	0.0004	15.0914	$1.8 \cdot 10^{-6}$
10	0.00007	0.001	17.8637	$7.9 \cdot 10^{-7}$
12	0.00005	0.0004	25.4754	$6.1 \cdot 10^{-7}$
14	0.00002	0.0003	35.0920	$3.3 \cdot 10^{-7}$
15	0.00001	0.0002	40.9761	$1.1 \cdot 10^{-7}$
17	0.000007	0.0002	45.1100	$6.9 \cdot 10^{-8}$
20	0.000005	0.0002	65.1659	$5.9 \cdot 10^{-8}$

Table 3.1

3.5 Application to more general types of curves using Fourier series decomposition

We have explored the possibility of applying the method described in the previous section to more general bounded curves with non necessarily trigonometric parametrizations. In order to do this, and assuming that the components of the parametrizations have sufficiently good properties, we replace these components by their truncated Fourier series, up to a certain (high) order.

Recall that if a function $f(t)$, $t \in \mathbb{R}$, is integrable on the interval $\left[t_0 - \dfrac{T}{2}, t_0 + \dfrac{T}{2}\right]$, then

$$f(t) \approx \frac{a_0}{2} + \sum_{m=1}^{n} \left[a_m \cos\left(\frac{2m\pi}{T} \cdot t\right) + b_m \sin\left(\frac{2m\pi}{T} \cdot t\right) \right], \tag{3.29}$$

where a_0, a_m and b_m are the fourier coefficients

$$\begin{aligned} a_0 &= \frac{2}{T} \int_{-T/2}^{T/2} f(t) dt, \\ a_m &= \frac{2}{T} \int_{-T/2}^{T/2} f(t) \cos\left(\frac{2m\pi}{T} \cdot t\right) dt, \\ b_m &= \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin\left(\frac{2m\pi}{T} \cdot t\right) dt. \end{aligned}$$

In general the approximation is better when n is large. Notice in particular that if we take $T = 2\pi$, Eq. (3.29) provides a trigonometric parametrization as in Eq.(3.2).

We have tried this idea for four examples shown in Table 3.2. In each case, the curve $\mathcal{D}_i$ is the result of applying to each $\mathcal{C}_i$ the affine equivalence with matrix $\boldsymbol{A}$ and vector $\boldsymbol{b}$ as follows:

$$\boldsymbol{A} = \begin{pmatrix} -4 & 1 \\ 3 & -1/2 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 4 \\ -2 \end{pmatrix}. \tag{3.30}$$

The CPU time is given in minutes (min). The results are good, but the main difficulty is that in order to get good approximations, we need a large n, which implies time-consuming computations, as one can deduce from the timings in Table 3.2. In our case we needed $n \geq 100$. So even though the idea is natural, the performance is not satisfactory.

$\mathcal{C}_1$	$\boldsymbol{x}_1(t) = \left(\sin(t)\left(e^{\cos(t)} - 2\cos(t) - \sin^5\left(\frac{t}{3}\right)\right), \cos(t)(e^{\sin(t)} + \cos(4t))\right)$ $n = 100$ $\frac{\lVert\boldsymbol{A}-\boldsymbol{A}_1\rVert_2}{\lVert\boldsymbol{A}\rVert_2} \approx 2.7 \cdot 10^{-7}$, $\frac{\lVert\boldsymbol{b}-\boldsymbol{b}_1\rVert}{\lVert\boldsymbol{b}\rVert} \approx 1.2 \cdot 10^{-16}$ $\nu \approx 8.7 \cdot 10^{-7}$ (with 100 points) CPU time (min): 8.5010
$\mathcal{C}_2$	$\boldsymbol{x}_2(t) = \left(\frac{2^t}{2^{3t}+4}, \frac{2^{2t}}{2^{4t}+2}\right)$ $n = 150$ $\frac{\lVert\boldsymbol{A}-\boldsymbol{A}_1\rVert_2}{\lVert\boldsymbol{A}\rVert_2} \approx 1.5 \cdot 10^{-7}$, $\frac{\lVert\boldsymbol{b}-\boldsymbol{b}_1\rVert}{\lVert\boldsymbol{b}\rVert} \approx 1.9 \cdot 10^{-7}$ $\nu \approx 7.4 \cdot 10^{-7}$ (with 100 points) CPU time (min): 26.7515
$\mathcal{C}_3$	$\boldsymbol{x}_3(t) = \left(\frac{t}{t^4+2}, \frac{t^2+1}{e^t \cdot t^2+1}\right)$ $n = 150$ $\frac{\lVert\boldsymbol{A}-\boldsymbol{A}_1\rVert_2}{\lVert\boldsymbol{A}\rVert_2} \approx 1.2 \cdot 10^{-7}$, $\frac{\lVert\boldsymbol{b}-\boldsymbol{b}_1\rVert}{\lVert\boldsymbol{b}\rVert} \approx 3.8 \cdot 10^{-9}$ $\nu \approx 7.8 \cdot 10^{-8}$ (with 100 points) CPU time (min): 27.7606
$\mathcal{C}_4$	$\boldsymbol{x}_4(t) = \left(\frac{e^{3t+5} + e^{-(3t+5)} - 12}{6}, \frac{e^{t+4} + e^{-(t+4)} + 18}{2}\right)$ $n = 150$ $\frac{\lVert\boldsymbol{A}-\boldsymbol{A}_1\rVert_2}{\lVert\boldsymbol{A}\rVert_2} \approx 3.1 \cdot 10^{-10}$, $\frac{\lVert\boldsymbol{b}-\boldsymbol{b}_1\rVert}{\lVert\boldsymbol{b}\rVert} \approx 5.3 \cdot 10^{-8}$ $\nu \approx 2.7 \cdot 10^{-17}$ (with 100 points) CPU time (min): 26.1986

Table 3.2

CHAPTER 4

SIMILARITIES OF NON-NECESSARILY RATIONAL CURVES

In this chapter, we employ two notions of Mechanics, namely those of *center of gravity* and *inertia tensor*, in order to compute the exact or approximate similarities between two parametric, not necessarily rational, bounded curves. For simplicity, we develop our results for planar curves. However, some of the strategies that we present here are generalizable to curves in any dimension.

In Mechanics (see [24, 48, 25]), the center of gravity of a rigid body is the point where the total weight of the body is assumed to be concentrated. If the body has a uniform density, i.e., if the body is homogeneous, the center of gravity matches the geometric center or centroid. On the other hand, while the mass of a body describes the resistance of the body to move under the action of a certain force, the inertia tensor represents, roughly speaking, the resistance of the body to rotate around the axes of a coordinate system centered at the center of gravity. Therefore, the inertia tensor depends on the directions of the axes, and has a *tensorial* nature (see [6, 91, 101] for the notion of *tensor*): this means that the inertia tensor, which is represented by a square matrix, changes in a precise way when the orthonormal basis describing the system of coordinates changes.

These two notions are intrinsic, i.e., they depend on the geometry of the object, and therefore they are preserved (in the case of the inertia tensor, up to a certain extent) when a similarity is applied. Thus, if two objects are similar, the similarity relating them maps the center of gravity of the first object onto the center of gravity of the second object. Additionally, when we consider the same coordinate system, centered at the center of gravity, for both objects, the inertia tensors are related by the law that describes how a tensor changes when an orthonormal change of coordinates is applied. These two relationships provide a method to compute the similarities relating two objects, if any.

We require the curves we work with to be bounded since otherwise, the center of gravity and the inertia tensor are not well-defined. Additionally, for curves finding the center of gravity or the components of the inertia tensor amounts to computing certain univariate integrals. Therefore, we demand that the parametrizations of the curves have sufficiently good properties (e.g., differentiability) so that the integrals exist. In the case of planar closed curves, we can also consider the curves as borders of solid objects, i.e., planar regions, and compute the similarities between these planar regions instead. This is useful in the case of closed rational curves and trigonometric curves without self-intersections because after Green's Theorem, we can compute these integrals by using the Residues' Theorem, which is simpler and more efficient.

If the integrals can be computed exactly, then we can compute exact similarities. However, in most cases the integrals need to be computed numerically. In those cases we seek not exact, but approximate similarities. We do this by using approximate gcds.

Additionally, the relationships between the center of gravity and the inertia tensors of the curves allow us to compute a superset of tentative similarities. However, they are not sufficient to guarantee that the objects we are analyzing are similar. Thus, once we have computed the tentative (exact or approximate) similarities, we need to test whether they actually are similarities between the objects. In order to do this, we present a heuristic strategy to assess the closeness between both objects.

4.1 Center of gravity and inertia tensor

In Mechanics, the *center of gravity* of a rigid body [54] is the point where the entire mass of an object is assumed to be concentrated so that the total weight of the object, as a force, is exerted at this point. If the body has a uniform distribution of mass, the center of gravity coincides with the geometric centroid. In particular, if the body is symmetric with respect to a point, i.e., if the body has *central symmetry*, the center of gravity coincides with the center of symmetry. If the body has an axis of symmetry, the center of gravity lies on this axis.

In this subsection, we will consider a bounded planar curve $\mathcal{C} \subset \mathbb{R}^2$. If $\mathcal{C}$ is, additionally, closed (i.e., it encloses a finite area), then $\mathcal{C}$ is the border of a planar region that we will denote by Δ, i.e., $\partial\Delta = \mathcal{C}$. Furthermore, we will assume that $\mathcal{C}$ can be parametrized by $\boldsymbol{x}(t) = (x_1(t), x_2(t))$, with $t \in I \subset \mathbb{R}$, where I is an interval (possibly infinite). We will also assume that for $i = 1, 2$ the first derivatives of the functions $x_i(t)$ are continuous, so that all the univariate integrals that we will be using in this chapter exist.

Under these assumptions, the *center of gravity* of $\mathcal{C}$ is defined as the point $\mathbf{G} = (\mathbf{x}_1, \mathbf{x}_2)$, where

$$\mathbf{x}_i = \frac{1}{L}\int_I x_i(t)ds, \tag{4.1}$$

for $i = 1, 2$, $ds = \sqrt{x_1'^2(t) + x_2'^2(t)} = \|\boldsymbol{x}'(t)\|dt$, and $L = \int_I ds$ is the *total length* of $\mathcal{C}$; the symbol $\| \bullet \|$ denotes the Euclidean norm.

If $\mathcal{C}$ is closed, in some cases we will also consider the *center of gravity* $\mathbf{G}^\star = (\mathbf{x}_1^\star, \mathbf{x}_2^\star)$ of the *planar region* Δ whose border is $\mathcal{C}$. In this case, for $i = 1, 2$ we have

$$\mathbf{x}_i^\star = \frac{1}{A}\iint_\Delta x_i dA, \tag{4.2}$$

where $dA = dx_1 dx_2$ and $A = \iint_\Delta dA$ is the total area of Δ.

Notice that in general, $\mathbf{G}$ and $\mathbf{G}^\star$ do not need to coincide: in the case of $\mathbf{G}$, all the mass is evenly distributed over $\mathcal{C}$ only; however, in the case of $\mathbf{G}^\star$ the mass is evenly

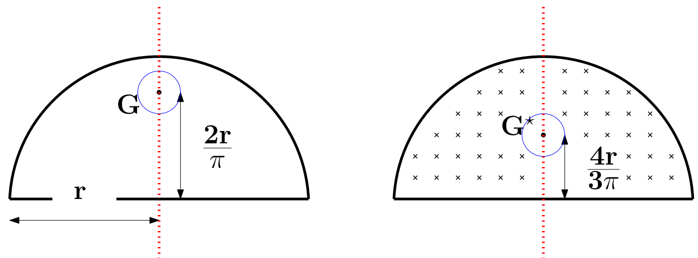


Figure 4.1: Centers of gravity of a semicircle and its border

distributed over the planar region Δ enclosed by $\mathcal{C}$. For instance, the center of gravity of a half circle of radius r is placed on its axis of symmetry, at a height equal to $\frac{4r}{3\pi}$ from its center (see Fig. 4.1, right). In contrast, the center of gravity of the border of the half circle (the union of a half circumference and a segment) is also placed on its axis of symmetry, but at a height equal to $\frac{2r}{\pi}$ from its center (see Fig. 4.1, left).

Furthermore, also in Mechanics, the *moment of inertia* of an object around an axis represents the resistance of the object to rotate around the axis; we refer the interested reader to [66] for further reference on the notion of moment of inertia and related concepts from Mechanics alluded to in this section. Moments of inertia are the main ingredients of the *inertia tensor*, which allows describing the rotational kinetics of an object. In more detail, let $\mathcal{C} \subset \mathbb{R}^2$ be a curve parametrized by $(x_1(t), x_2(t))$, with $t \in I$ as before, and assume, perhaps after a translation, that the center of gravity of $\mathcal{C}$ is the origin. The inertia tensor of $\mathcal{C}$ in the frame $\{O, x, y\}$, where O represents the origin and x, y represent the coordinate axes, is the tensor of order two defined by the matrix

$$\mathbf{T} = \begin{bmatrix} I_{xx} & -I_{xy} \\ -I_{xy} & I_{yy} \end{bmatrix} = \frac{1}{L} \begin{bmatrix} \displaystyle\int_I x_2^2(t)ds & \displaystyle -\int_I x_1(t)x_2(t)ds \\ \displaystyle -\int_I x_1(t)x_2(t)ds & \displaystyle\int_I x_1^2(t)ds \end{bmatrix}. \tag{4.3}$$

The elements in the first diagonal of the matrix in Eq. (4.3) are the moments of inertia I_{xx}, I_{yy} with respect to the x-axis and the y-axis. The element in the second diagonal of the matrix in Eq. (4.3) is minus the *product of inertia* I_{xy}. The matrix $\mathbf{T}$ satisfies that $\vec{M} = \mathbf{T}\vec{\omega}$, where $\vec{M}$ is the *angular momentum*, and $\vec{\omega}$ is the angular velocity; both vectors $\vec{M}, \vec{\omega}$ are related to the rotational kinetics of $\mathcal{C}$.

If $\mathcal{C}$ is closed and we consider the planar region Δ whose border is $\mathcal{C}$, i.e., if we assume that the inside of $\mathcal{C}$ is solid and not hollow, then the expression of the inertia tensor becomes

$$\mathbf{T}^\star = \frac{1}{A}\begin{bmatrix} \iint_\Delta x_2^2 dA & -\iint_\Delta x_1 x_2 dA \\ -\iint_\Delta x_1 x_2 dA & \iint_\Delta x_1^2 dA \end{bmatrix}. \tag{4.4}$$

If $\mathcal{C}$ is a closed curve, as it happens with the center of gravity, the inertia tensor needs not be the same when we consider $\mathcal{C}$ only, or the area Δ enclosed by $\mathcal{C}$. Consider for instance the case of an ellipse $\mathcal{E}$ with $a = 2$ and $b = 1$ paremetrized by $\boldsymbol{x}(t) = \left(\frac{2(1-t^2)}{t^2+1}, \frac{2t}{t^2+1}\right)$. When we consider the border of $\mathcal{E}$ (see Fig. 4.2 left), the inertia tensor is

$$\mathbf{T} \approx \begin{bmatrix} 0.5799 & 0 \\ 0 & 1.6803 \end{bmatrix}. \tag{4.5}$$

However, when we consider Δ with $\partial\Delta = \mathcal{E}$, the inertia tensor is

$$\mathbf{T}^\star = \begin{bmatrix} 0.25 & 0 \\ 0 & 1 \end{bmatrix}. \tag{4.6}$$

Notice that the matrix representing the inertia tensor is diagonal in both cases because of the symmetry of the object, which makes that the integrals in the second diagonal vanish. Furthermore, notice the different values of the inertia moments I_{xx}, I_{yy}: intuively, if $a > b$ then $I_{xx} < I_{yy}$ because it is easier to rotate the ellipse around the x-axis, compared to the y-axis.

The example of the ellipse in Fig. 4.2 allows us to make an observation that will be important later. Because of the intrinsic nature of the inertia tensor, if we apply a

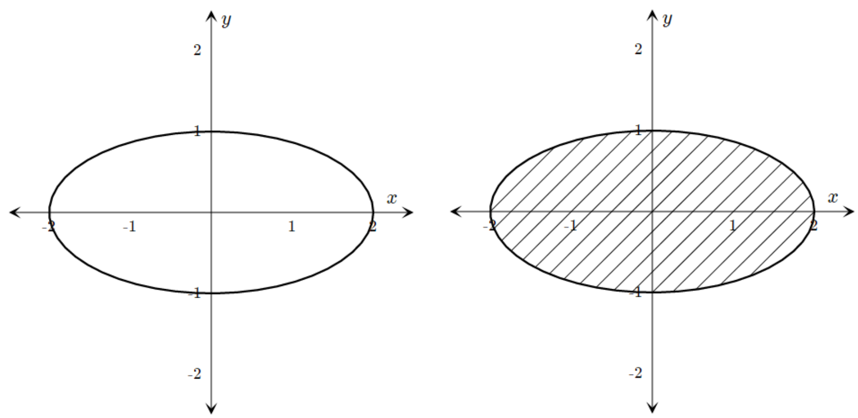


Figure 4.2: Elipse $\mathcal{E}$ on left and the area Δ enclosed by $\mathcal{E}$ on the right

rigid motion f to the ellipse (see for instance Fig. 4.3, where we have applied a rotation f of $\frac{\pi}{4}$ radians with center in the origin to the ellipse in Fig. 4.2), and x', y' are the images of the x-axis and the y-axis under f, the inertia tensor of $f(\mathcal{E})$ with respect to the x'-axis and the y'-axis coincides with the inertia tensor of $\mathcal{E}$ with respect to the x-axis and the y-axis. Thus, in Fig. 4.3 the inertia tensor of the red ellipse, $f(\mathcal{E})$, with respect to the axes x', y' (which are the result of rotating x, y), coincides with the inertia tensor of $\mathcal{E}$ with respect to x, y.

But one can also wonder what the inertia tensor of $f(\mathcal{E})$ with respect to x, y (and not x', y') would be. In fact, this question amounts to wondering about the relationship between the inertia tensors of $\mathcal{E}$ and $f(\mathcal{E})$ in the axes x, y. This is related with the *tensorial* nature of the inertia tensor, which is recalled in the next subsection; in fact, in the next subsection we will come back to this question about Fig. 4.3.

4.1.1 Brief review of tensors

In Physics, given a vector space V of dimension M, a *tensor* is, intuitively, an entity defined with respect to a basis of V that changes in a certain way when the basis changes (see [6, 91, 101]). A tensor depends on several indexes, each one of two possible

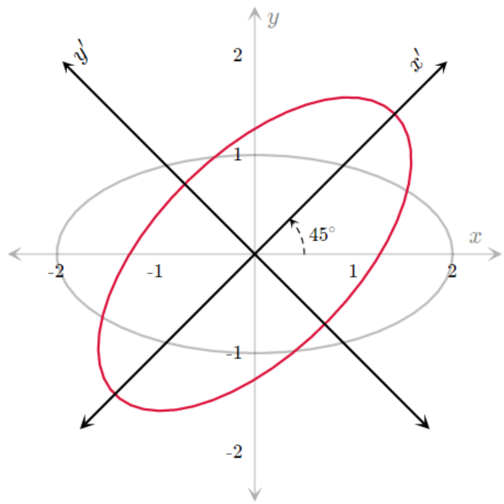


Figure 4.3: Ellipse $f(\mathcal{E})$ after a rotation of $\frac{\pi}{4}$.

different natures, *contravariant* or *covariant*, and can be seen as a multiarray. In Mathematics, tensors are defined as multilinear forms going from the cartesian product of a certain numbers of copies of V and a certain number of copies of the dual $V^\star$, to the field where V is defined. Interestingly, for our purposes is more useful to consider the point of view of Physics.

Let us provide some more details, already adapting the situtation to our case. A *contravariant* vector in V, according to [6], is an entity $\mathcal{T} = \{t^i\}, i = 1, \ldots, M$, such that when an orthonormal change of basis represented by an orthogonal square matrix $\boldsymbol{Q} = (Q_r^s)$ is applied, the components of $\mathcal{T}$ change according to the rule

$$\hat{t}^i = \sum_{r=1,\ldots,M} Q_r^i \cdot t^r.$$

On the other hand, a *covariant vector* is an entity $\mathcal{T} = \{t_j\}$, with $j = 1, \ldots, M$, whose components t_j transform by the rule

$$\hat{t}_j = \sum_{s=1,\ldots,M} (Q^{-1})_j^s \cdot t_s = \sum_{s=1,\ldots,M} (Q^T)_j^s \cdot t_s$$

when an orthogonal change of basis defined by a square matrix $\boldsymbol{Q} = (Q_r^s)$ is applied. Both contravariant and covariant vectors correspond to tensors of *order one*, since they refer to entities depending on just one index. The nature of a tensor of order one can be contravariant or covariant, depending on the transformation rule.

With more generality, also following [6], an *n-contravariant and m-covariant tensor* is a multidimensional array $\mathcal{T} = \{t^{i_1,\dots,i_n}_{j_1,\dots,j_m}\}$, where $i_p, j_q \in \{1,\dots,M\}$ for $p = 1,\dots,n, q = 1,\dots,m$ whose components, under an orthogonal change of basis defined by $\boldsymbol{Q} = (Q_i^r)$, transform into $\hat{t}^{i_1,\dots,i_n}_{j_1,\dots,j_m}$ as

$$\hat{t}^{i_1,\dots,i_n}_{j_1,\dots,j_m} = \sum_{r_1=1}^{M} \cdots \sum_{r_n=1}^{M} \sum_{s_1=1}^{M} \cdots \sum_{s_m=1}^{M} (Q^T)^{i_1}_{r_1} \cdots (Q^T)^{i_n}_{r_n} \cdot Q^{s_1}_{j_1} \cdots Q^{s_m}_{j_m} \cdot t^{r_1,\dots,r_n}_{s_1,\dots,s_m}. \tag{4.7}$$

Notice that the behavior of the first n indexes and the last m indexes is different: while the first ones have a contravariant behavior, the second ones have a covariant behavior. We say that this is an (n,m)-tensor, and that its *order* is $N = n + m$. However, when working in the Euclidean space, as it will be our case, the nature of the indexes is the same (see Section 10.2 in [91]), so we keep the value of N and we need not worry about n, m; we will be assuming this from now on.

Second order tensors, i.e., tensors of order $N = 2$, can be represented by matrices. Furthermore, in this case Eq. (4.7) corresponds to

$$\mathbf{T}' = \mathbf{Q}^{\mathbf{T}} \cdot \mathbf{T} \cdot \boldsymbol{Q}, \tag{4.8}$$

where $\boldsymbol{Q}$ is the matrix defining the orthonormal change of basis.

In particular, the inertia tensor introduced in the previous section is a tensor of order two. Furthermore, the law in Eq. (4.8) is essential for our purposes. In order to illustrate this, let us go back to the question, regarding Fig. 4.3, that we asked ourselves before starting this subsection: the relationship between the inertia tensor of the red ellipse of Fig. 4.3 in the x, y axes, and the inertia tensor of the black ellipse in the x, y axes. To answer this question, we recall the observation that the inertia tensor of the red ellipse in the axes x', y' coincides con the inertia tensor of the black ellipse in the axes x, y, that we denote by $\mathbf{T}$. Now let $\mathbf{T}'$ be the inertia tensor of the red ellipse in the x, y axes. If $\boldsymbol{Q}$ represents the orthogonal transformation taking (x, y) to (x', y'), then $\boldsymbol{Q}^T$ represents the orthogonal transformation taking (x', y') to (x, y). Thus, using Eq. (4.8) and replacing $\boldsymbol{Q}$ by $\boldsymbol{Q}^T$, we get

$$\mathbf{T}' = (\boldsymbol{Q}^T)^{-1} \cdot \mathbf{T} \cdot \boldsymbol{Q}^T = \boldsymbol{Q} \cdot \mathbf{T} \cdot \boldsymbol{Q}^T. \tag{4.9}$$

One can verify this in the case of Fig. 4.3. The inertia tensor $\mathbf{T}$ of $\mathcal{E}$ is given in Eq. (4.5); this is also the inertia tensor of $f(\mathcal{E})$ in the axes x', y'. On the other hand, the inertia tensor of the transformed (rotated) ellipse $f(\mathcal{E})$ in x, y is

$$\mathbf{T}' \approx \begin{bmatrix} 1.1301 & 0.55019 \\ 0.55019 & 1.1301 \end{bmatrix}, \tag{4.10}$$

and we can verify that $\mathbf{T}' = \boldsymbol{Q} \cdot \mathbf{T} \cdot \boldsymbol{Q}^T$ where $\boldsymbol{Q}$ is the rotation matrix given by

$$\boldsymbol{Q} = \begin{bmatrix} \sqrt{2}/2 & \sqrt{2}/2 \\ -\sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix}. \tag{4.11}$$

The idea behind Eq. (4.9), motivated by the example in Fig. 4.3, will be used in the next section.

4.2 Similarities

Let $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$ be two bounded planar curves (possibly equal). We will assume that $\mathcal{C}_1, \mathcal{C}_2$ are respectively parametrized by $\boldsymbol{x}(t) = (x_1(t), x_2(t))$, with $t \in I_1$ and I_1 an interval, $\boldsymbol{y}(t) = (y_1(t), y_2(t))$, with $t \in I_2$ and I_2 also an interval, and where $x_1(t), x_2(t)$ and $y_1(t), y_2(t)$ are bounded functions with continuous first derivatives. Additionally, if $\mathcal{C}_1, \mathcal{C}_2$ are closed curves we will represent the bounded planar regions they enclose by Δ_1, Δ_2, i.e., $\partial \Delta_i = \mathcal{C}_i$ for $i = 1, 2$. We do not request the $x_i(t), y_i(t)$ to be rational functions; in particular, the method is valid for non-rational, parametric curves in the required conditions.

Our goal is to develop methods, using the notions of center of gravity and inertia tensor, to check whether $\mathcal{C}_1, \mathcal{C}_2$ are similar. In other words, we want to check whether there exists

$$f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x} + \boldsymbol{b}, \tag{4.12}$$

with $\boldsymbol{Q}$ orthogonal, $\lambda > 0$, satisfying that $f(\mathcal{C}_1) = \mathcal{C}_2$. Notice that since $\boldsymbol{Q}$ is orthogonal,

$$\boldsymbol{Q} = \begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix}, \quad \text{or} \quad \boldsymbol{Q} = \begin{bmatrix} \alpha & \beta \\ \beta & -\alpha \end{bmatrix}, \tag{4.13}$$

with $\alpha, \beta \in \mathbb{R}$, $\alpha^2 + \beta^2 = 1$.

In order to do this, first we need to know how gravity centers and inertia tensors behave when a similarity is applied. The following result will be needed.

Lemma 4.1. *Let $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$ be two bounded planar curves related by a similarity, and for $i = 1, 2$ let L_i denote the total length of $\mathcal{C}_i$. Then*

$$L_2 = \lambda L_1. \tag{4.14}$$

Furthermore, if $\mathcal{C}_1, \mathcal{C}_2$ are closed curves enclosing planar regions Δ_1, Δ_2, with areas A_1, A_2, then

$$A_2 = \lambda^2 A_1. \tag{4.15}$$

Proof. Let us see Eq. (4.14) first. In order to do that, let $\boldsymbol{x}(t)$, $t \in I$, be a parametrization of $\mathcal{C}_1$, and let $f : \mathcal{C}_1 \to \mathcal{C}_2$ be a similarity as in Eq. (4.12). Since $f(\mathcal{C}_1) = \mathcal{C}_2$, $\boldsymbol{z}(t) = \lambda \boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{b}$, $t \in I$, parametrizes $\mathcal{C}_2$. Therefore,

$$L_2 = \int_I \|\boldsymbol{z}'(t)\| dt = \int_I \|\lambda \boldsymbol{Q}\boldsymbol{x}'(t)\| dt = \lambda \int_I \|\boldsymbol{Q}\boldsymbol{x}'(t)\| dt.$$

Since $\boldsymbol{Q}$ is orthogonal, $\boldsymbol{Q}$ preserves norms, so $\|\boldsymbol{Q}\boldsymbol{x}'(t)\| = \|\boldsymbol{x}'(t)\|$, and we get Eq. (4.14). To prove Eq. (4.15), it suffices to observe that the determinant of the Jacobian matrix of f is equal to $\lambda^2 \cdot \det(\boldsymbol{Q})$. Since $\boldsymbol{Q}$ is orthogonal $\det(\boldsymbol{Q}) = \pm 1$. Thus, the absolute value of the determinant of the Jacobian is equal to λ^2, and Eq. (4.15) follows from the Change of Variables Theorem for double integrals. ∎

Corollary 4.1. *Let $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$ be two bounded planar curves whose centers of gravity lie at the origin, and let Δ_1, Δ_2 be the bounded regions enclosed by $\mathcal{C}_1, \mathcal{C}_2$ in the case when both curves are closed. Let $f(\mathbf{x}) = \lambda \mathbf{x}$, $\lambda > 0$, and assume that $f(\mathcal{C}_1) = \mathcal{C}_2$. Finally, let $\mathbf{T}_1, \mathbf{T}_2$ be the inertia tensors of $\mathcal{C}_1, \mathcal{C}_2$, and let $\mathbf{T}_1^\star, \mathbf{T}_2^\star$ be the inertia tensors of Δ_1, Δ_2. Then*

(1) $\mathbf{T}_2 = \lambda^2 \mathbf{T}_1$.

(2) $\mathbf{T}_2^\star = \lambda^2 \mathbf{T}_1^\star$.

Proof. (1) follows from Eq. (4.3), taking Eq. (4.14) into account. (2) follows from Eq. (4.4), taking Eq. (4.15) into account. ∎

In particular, Lemma 4.1 provides the tentative value of λ of any similarity mapping $\mathcal{C}_1$ onto $\mathcal{C}_2$. Now the following result shows the behaviour of the center of gravity under a similarity.

Theorem 4.1. *Let $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$ be two bounded planar curves related by a similarity f. Furthermore, if both $\mathcal{C}_1, \mathcal{C}_2$ are closed, let Δ_1, Δ_2 be the bounded regions enclosed by $\mathcal{C}_1, \mathcal{C}_2$, i.e., for $i = 1, 2$, $\partial \Delta_i = \mathcal{C}_i$. And for $i = 1, 2$, let $\mathbf{G}_i$ (resp. $\mathbf{G}_i^\star$) be the center of gravity of $\mathcal{C}_i$ (resp. Δ_i). Then (1) $f(\mathbf{G}_1) = \mathbf{G}_2$; (2) $f(\mathbf{G}_1^\star) = \mathbf{G}_2^\star$.*

Proof. We prove it for curves, first. By assumption, $\mathcal{C}_2 = f(\mathcal{C}_1)$, so $\boldsymbol{z}(t) = \lambda \boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{b}$, $t \in I_1$, parametrizes $\mathcal{C}_2$. Then, in compact notation, using Eq. (4.14) in Lemma 4.1

and the fact that $\boldsymbol{Q}$ is orthogonal (which implies that $\|\boldsymbol{Q}\boldsymbol{x}'(t)\| = \|\boldsymbol{x}'(t)\|$), we get

$$\begin{aligned}
\mathbf{G}_2 &= \frac{1}{L_2}\int_{I_1} \boldsymbol{z}(t)\|\boldsymbol{z}'(t)\|dt \\
&= \frac{1}{\lambda L_1}\int_{I_1} (\lambda \boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{b})\lambda\|\boldsymbol{x}'(t)\|dt \\
&= \frac{1}{L_1}\int_{I_1} (\lambda \boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{b})\|\boldsymbol{x}'(t)\|dt.
\end{aligned}$$

Since

$$\frac{1}{L_1}\int_{I_1} (\lambda \boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{b})\|\boldsymbol{x}'(t)\|dt = \lambda \boldsymbol{Q}\frac{1}{L_1}\int_{I_1} \boldsymbol{x}(t)\|\boldsymbol{x}'(t)\|dt + \boldsymbol{b}\underbrace{\left(\frac{1}{L_1}\int_{I_1} \|\boldsymbol{x}'(t)\|dt\right)}_{=\frac{1}{L_1}\cdot L_1 = 1},$$

we get (1).

Now for regions, i.e., for (2), the result follows from Eq. (4.2), using Eq. (4.15) in Lemma 4.1 and the fact that, because $\boldsymbol{Q}$ is orthogonal, the jacobian of the transformation $f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x} + \boldsymbol{b}$ is λ^2. ■

Furthermore, the following result shows the behavior of the inertia tensor under a similarity. In order to follow the proof, it is illustrative to recall the discussion about Fig. Fig. 4.3 that we made at the end of Subsection 4.1.1.

Theorem 4.2. *Let $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$ be two bounded planar curves, and let Δ_1, Δ_2 be the bounded regions enclosed by $\mathcal{C}_1, \mathcal{C}_2$ in the case when both curves are closed.*

(1) If for $i = 1, 2$ the center of gravity of $\mathcal{C}_i$ is the origin O, and $\mathcal{C}_2 = f(\mathcal{C}_1)$ with $f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x}$, $\lambda > 0$ and $\boldsymbol{Q}$ orthogonal, then the relationship between the inertia tensors $\mathbf{T}_1, \mathbf{T}_2$ of $\mathcal{C}_1$ and $\mathcal{C}_2$ in the frame $\{O; x, y\}$ is

$$\mathbf{T}_2 = \lambda^2 \boldsymbol{Q} \cdot \mathbf{T}_1 \cdot \boldsymbol{Q}^T. \tag{4.16}$$

(2) If for $i = 1, 2$ the center of gravity of Δ_i is the origin O, and $\mathcal{C}_2 = f(\mathcal{C}_1)$ with

$f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x}$, $\lambda > 0$ and $\boldsymbol{Q}$ orthogonal, then the relationship between the inertia tensors $\mathbf{T}_1^\star, \mathbf{T}_2^\star$ of Δ_1 and Δ_2 in the frame $\{O; x, y\}$ is

$$\mathbf{T}_2^\star = \lambda^2 \boldsymbol{Q} \cdot \mathbf{T}_1^\star \cdot \boldsymbol{Q}^T. \tag{4.17}$$

Proof. We address (1); (2) is completely analogous. Let x', y' be the images of the axes x, y under $f(\mathbf{x})$. Assume first that $\lambda = 1$, in which case $f(\mathbf{x})$ defines a rigid motion. Because of the intrinsic nature of the inertia tensor, the inertia tensor $\mathbf{T}_2'$ of $\mathcal{C}_2 = f(\mathcal{C}_1)$ with respect to the axes x', y' coincides with the inertia tensor $\mathbf{T}_1$ of $\mathcal{C}_1$ with respect to the axes x, y. Now observe that the tensor $\mathbf{T}_2$ is in fact the inertia tensor of $\mathcal{C}_2$ with respect to the axes x, y. Thus, we just need to relate the inertia tensors $\mathbf{T}_2'$ and $\mathbf{T}_2$. The key observation now is that since $f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x}$, the change of basis mapping x', y' to x, y has $\boldsymbol{Q}^{-1} = \boldsymbol{Q}^T$ (because $\boldsymbol{Q}$ is orthogonal) as change of basis matrix. Thus,

$$\mathbf{T}_2 = \boldsymbol{Q} \cdot \mathbf{T}_2' \cdot \boldsymbol{Q}^T = \boldsymbol{Q} \cdot \mathbf{T}_1 \cdot \boldsymbol{Q}^T.$$

For $\lambda \neq 1$, the result follows from Corollary 4.1. ∎

Since after perhaps an appropriate translation, we can always assume that $\mathbf{G}_1$ and $\mathbf{G}_2$, or $\mathbf{G}_1^\star$ and $\mathbf{G}_2^\star$, coincide and are placed at the origin of the coordinate system, we have the following corollary of Theorem 4.1 and Theorem 4.2.

Corollary 4.2. *Let $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$ be two bounded planar curves, and let Δ_1, Δ_2 be the bounded regions enclosed by $\mathcal{C}_1, \mathcal{C}_2$ in the case when both curves are closed.*

(1) If $\mathbf{G}_1 = \mathbf{G}_2 = (0, 0)$ and $\mathcal{C}_1, \mathcal{C}_2$ are related by a similarity f, then $f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x}$, where $\lambda, \boldsymbol{Q}$ satisfy Eq. (4.14) *and Eq.* (4.16).

(2) If $\mathbf{G}_1^\star = \mathbf{G}_2^\star = (0, 0)$ and $\mathcal{C}_1, \mathcal{C}_2$ are related by a similarity f, then $f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x}$, where $\lambda, \boldsymbol{Q}$ satisfy Eq. (4.15) *and Eq.* (4.17).

Thus, applying Corollary 4.2 we get tentative values for $\lambda, \boldsymbol{Q}$. Unfortunately, the conditions in Corollary 4.2 are necessary for two planar curves to be similar but are not sufficient. The reason behind this is that the inertia tensor does not characterize the

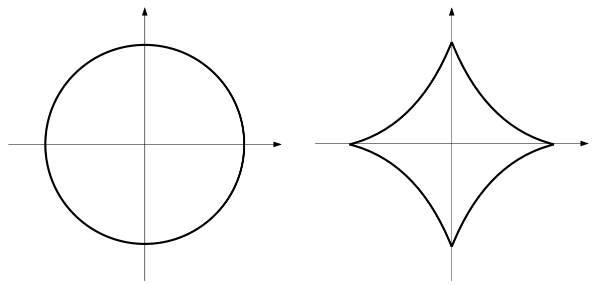


Figure 4.4: The converse of Corollary 4.2 is not true

shape of the curve, i.e., there can be two different curves with the same inertia tensor. For instance, the inertia tensors of a circle centered at the origin and of an astroid curve (see Fig. 4.4, right) whose centroid is the origin, for symmetry reasons, are both defined by matrices that are multiples of the identity matrix. Thus, we can find λ and $\boldsymbol{Q}$ (the identity) satisfying Eq. (4.14) and Eq. (4.16). However, a circle and an astroid curve are not similar (see Fig. 4.4).

Since the conditions of Corollary 4.2 are not sufficient, we must test the tentatives $\boldsymbol{Q}$ to check whether or not they correspond to similarities mapping one curve onto the other. We provide more details in the next sections.

4.2.1 Computation of the similarities

In this subsection we develop in more detail a strategy, based on statement (1) of Corollary 4.2, to compute the similarities, if any, between two bounded curves $\mathcal{C}_1$ and $\mathcal{C}_2$. This strategy can be generalized to curves in higher dimensions. We request our curves to be bounded, but not necessarily closed. Notice that whenever the $\mathcal{C}_i$ are exact and the integrals defining the centers of gravity and inertia tensors can be computed exactly, we get exact similarities as well. Here we will assume that this is the case: we will deal with inaccuracies (what we call the *approximate case*) later.

Now the tentative value of the scaling constant λ is computed from Eq. (4.14), and the centers of gravity of $\mathcal{C}_1$ and $\mathcal{C}_2$ are computed by means of Eq. (4.1). After that, we can apply a translation so that the centers of gravity of the resulting curves are the origin in both cases. Then the inertia tensors $\mathbf{T}_1, \mathbf{T}_2$ of the translated curves can be computed by using Eq. (4.3). From statement (1) in Theorem 4.2, $\mathbf{T}_1, \mathbf{T}_2$ sastisfy Eq. (4.16). Since the matrix $\boldsymbol{Q}$ is orthogonal, Eq. (4.16) gives rise to $\mathbf{T}_2\boldsymbol{Q} = \lambda^2\boldsymbol{Q}\mathbf{T}_1$, which in turn provides two linear systems $\mathcal{S}_1, \mathcal{S}_2$ in α, β (see Eq. (4.13)), depending on which expression of $\boldsymbol{Q}$ in Eq. (4.13) we consider. Writing

$$\mathbf{T}_1 = \begin{pmatrix} t_{11}^{(1)} & t_{12}^{(1)} \\ t_{12}^{(1)} & t_{22}^{(1)} \end{pmatrix}, \quad \mathbf{T}_2 = \begin{pmatrix} t_{11}^{(2)} & t_{12}^{(2)} \\ t_{12}^{(2)} & t_{22}^{(2)} \end{pmatrix},$$

for $i = 1, 2$ the linear system $\mathcal{S}_i$ can be written as

$$\mathbf{B_i} \cdot [\alpha, \beta]^T = \mathbf{0}, \tag{4.18}$$

where the $\mathbf{B}_i$ are the following 4×2 matrices:

$$\mathbf{B}_1 = \begin{bmatrix} t_{11}^{(2)} - \lambda^2 t_{11}^{(1)} & -t_{12}^{(2)} - \lambda^2 t_{12}^{(1)} \\ t_{12}^{(2)} - \lambda^2 t_{12}^{(1)} & t_{11}^{(2)} - \lambda^2 t_{22}^{(1)} \\ t_{12}^{(2)} - \lambda^2 t_{12}^{(1)} & -t_{22}^{(2)} + \lambda^2 t_{11}^{(1)} \\ t_{22}^{(2)} - \lambda^2 t_{22}^{(1)} & t_{12}^{(2)} + \lambda^2 t_{12}^{(1)} \end{bmatrix}, \mathbf{B}_2 = \begin{bmatrix} t_{11}^{(2)} - \lambda^2 t_{11}^{(1)} & t_{12}^{(2)} - \lambda^2 t_{12}^{(1)} \\ -t_{12}^{(2)} - \lambda^2 t_{12}^{(1)} & t_{11}^{(2)} - \lambda^2 t_{22}^{(1)} \\ t_{12}^{(2)} + \lambda^2 t_{12}^{(1)} & t_{22}^{(2)} - \lambda^2 t_{11}^{(1)} \\ -t_{22}^{(2)} + \lambda^2 t_{22}^{(1)} & t_{12}^{(2)} - \lambda^2 t_{12}^{(1)} \end{bmatrix}. \tag{4.19}$$

Recall also that $\alpha^2 + \beta^2 = 1$. Then we have the following proposition.

Proposition 4.1. *Let $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$ be two bounded planar curves, whose centers of gravity $\mathbf{G}_1, \mathbf{G}_2$ lie at the origin. If $\mathcal{C}_1, \mathcal{C}_2$ are similar then there exists $i \in \{1, 2\}$ such that* $\operatorname{rank}(\mathbf{B}_i) \leq 1$.

Proof. If $\mathcal{C}_1$ and $\mathcal{C}_2$ are similar, Eq. (4.18) must have a non-trivial solution. Thus, $\operatorname{rank}(\mathbf{B}_i) \neq 2$ for some $i \in \{1, 2\}$. ∎

Additionally, we have the following result.

Lemma 4.2. *The following statements are equivalent:*

(1) The matrix $\mathbf{B}_1$ *is the zero matrix.*

(2) The matrix $\mathbf{B}_2$ *is the zero matrix.*

(3) $\mathbf{T}_1, \mathbf{T}_2$ *are multiples of the identity matrix.*

Proof. (1) $\Rightarrow$ (2) follows directly since the entries of $\mathbf{B}_2$ are the same entries of $\mathbf{B}_1$ up to a change of position and sign.

In order to see that (2) $\Rightarrow$ (3), we first note that if all the entries of $\mathbf{B}_2$ are 0, then, in particular,

$$t_{12}^{(1)} = \frac{t_{12}^{(2)}}{\lambda^2} = -\frac{t_{12}^{(2)}}{\lambda^2}, \quad t_{11}^{(1)} = \frac{t_{11}^{(2)}}{\lambda^2} = t_{22}^{(1)}.$$

From here we get $t_{12}^{(1)} = 0$. Therefore,

$$\mathbf{T}_1 = \frac{1}{\lambda^2} \begin{pmatrix} t_{11}^{(2)} & 0 \\ 0 & t_{11}^{(2)} \end{pmatrix} = \frac{t_{11}^{(2)}}{\lambda^2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

and similarly for $\mathbf{T}_2$.

Finally, if both $\mathbf{T}_1, \mathbf{T}_2$ are multiples of the identity matrix, then

$$t_{11}^{(1)} = t_{22}^{(1)}, \quad t_{12}^{(1)} = 0 = t_{12}^{(2)}, \quad t_{11}^{(2)} = t_{22}^{(2)}.$$

Besides, Eq. (4.16) would imply that

$$\mathbf{T}_1 = \frac{1}{\lambda^2} \begin{pmatrix} t_{11}^{(2)} & 0 \\ 0 & t_{11}^{(2)} \end{pmatrix}.$$

Thus, $t_{11}^{(1)} = \dfrac{t_{11}^{(2)}}{\lambda^2}$. Since all the non-zero entries of $\mathbf{B}_i$, in this case, reduce to $\pm(t_{11}^{(2)} - \lambda^2 t_{11}^{(1)})$, we conclude that (3) $\Rightarrow$ (1). ■

The case in Lemma 4.2 can certainly happen (see for instance Fig. 4.4). In this situation, Corollary 4.2 does not provide finitely many similarities to test, so we will exclude this possibility from our study. Thus, we are left with the case when some of the $\mathbf{B}_i$ has rank 1. Assuming this, an obvious way to solve the systems in Eq. (4.18) is to choose a nonzero row of $\mathbf{B}_i$, write α in terms of β (or conversely), and then use that $\alpha^2 + \beta^2 = 1$. Notice that in this case we get at most four similarities.

This last observation leads to an unexpected result about the inertia tensors of curves with self-similarities. In order to develop this result, one observes first that the number of similarities between two curves is higher than one if and only if the curves are self-similar: indeed, if f_1 and f_2 are two different similarities between two curves $\mathcal{C}_1, \mathcal{C}_2$, then $f_1 \circ f_2^{-1}$ is a self-similarity of $\mathcal{C}_1$, and $f_1^{-1} \circ f_2$ is a self-similarity of $\mathcal{C}_2$; conversely, if g is a self-similarity of, say, $\mathcal{C}_1$, and f is a similarity between $\mathcal{C}_1, \mathcal{C}_2$, then $f \circ g$ is also a similarity between $\mathcal{C}_1, \mathcal{C}_2$. Then we have the following result.

Proposition 4.2. *Let $\mathcal{C} \subset \mathbb{R}^2$ be a bounded planar curve. If the number of non-trivial self-similarities of $\mathcal{C}$ is ≥ 4, then the inertia tensors $\mathbf{T}, \mathbf{T}^\star$ are multiples of the identity matrix.*

Proof. Let $\mathcal{C}_1 = \mathcal{C}$, and let $\mathcal{C}_2 = f(\mathcal{C})$, where f is a similarity of the plane. Since composing any self-similarity of $\mathcal{C}$ with f yields a similarity between $\mathcal{C}_1$ and $\mathcal{C}_2$, we get that the number of similarities between these two curves is at least 5. Since $\mathcal{C}_1$ and $\mathcal{C}_2$ are similar, by Proposition 4.1 we have $\operatorname{rank}(\mathbf{B}_i) \leq 1$. However, if $\operatorname{rank}(\mathbf{B}_i) = 1$ for $i = 1, 2$ then the number of similarities is at most 4, so $\operatorname{rank}(\mathbf{B}_i) = 0$. Thus, the result follows from Lemma 4.2. ∎

In the case of algebraic curves, self-similarities must be isometries (see Proposition 2 in [11]), so self-similar algebraic curves are symmetric. However, for non-algebraic curves, non-isometric self-similarities are possible (e.g., spirals). In particular, Proposition 4.2 implies that the method does not work when the curves $\mathcal{C}_1$ and $\mathcal{C}_2$ are similar and have more than three self-similarities (e.g., symmetries), since in that case, the inertia tensors are multiples of the identity matrix.

Now by using Eq. (4.18) we get tentative similarities f between the curves $\mathcal{C}_1$ and $\mathcal{C}_2$, but we still need to check whether $f(\mathcal{C}_1) = \mathcal{C}_2$. Notice that if $f(\mathcal{C}_1) = \mathcal{C}_2$, then $f(\boldsymbol{x}(t))$ parametrizes $\mathcal{C}_2$, but it does not need to coincide with $\boldsymbol{y}(t)$, since a change of parameters may be involved. For a general parametrization (for instance, with analytic functions), checking whether two different parametrizations define the same curve is, in practice, an unsolved problem. A possibility is to check whether the Hausdorff distance (see Definition (3.2) in Chapter 3) of the curves defined by both parametrizations is zero, but computing the Hausdorff distance is difficult (see also [50]). There are, however, two important cases where the problem of checking whether $f(\mathcal{C}_1) = \mathcal{C}_2$ define the same curve is solvable:

(i) *Rational curves:* if $\boldsymbol{x}(t)$ is rational, we can check whether another rational parametrization $\tilde{\boldsymbol{x}}(t) = (\tilde{x}_1(t), \tilde{x}_2(t))$ defines the same curve as $\boldsymbol{x}(t)$. In order to do this, we need both $\boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t)$ to be proper (see Section 1.3 in Chapter 1). Assuming that, $\boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t)$ both define the same curve iff there exists a Möbius function φ such that $\boldsymbol{x} = \tilde{\boldsymbol{x}} \circ \varphi$ (see also Section 1.3 in Chapter 1). This amounts to checking whether the polynomials obtained after clearing denominators in $x_1(t) - \tilde{x}_1(u)$, $x_2(t) - \tilde{x}_2(u)$ and factoring out the factor $t - u$, have a common gcd of the form $u(ct + d) - (at + b)$ with $ad - bc \neq 0$.

(ii) *Trigonometric curves:* given two trigonometric parametrizations $\boldsymbol{x}(t), \tilde{\boldsymbol{x}}(t)$, they can be transformed into rational complex parametrizations $\boldsymbol{y}(z), \tilde{\boldsymbol{y}}(z)$ following Eq. (3.5). Besides, from lemma 3.1 and corollary 3.1 we get that $\tilde{\boldsymbol{y}} = \boldsymbol{y} \circ \varphi$ where $\varphi(z) = kz$ or $\varphi(z) = \frac{k}{z}$. Thus, $\tilde{\boldsymbol{y}}$ will be rational and we can apply the criterion in (i).

For other cases, we provide in Section 4.3 a heuristic criterion to evaluate how close two curves are.

Finally, one can observe that the integrals in Eq. (4.1) and Eq. (4.3) include ds, which in general involves the square-root of a function. There is an important class of curves, *Pythagorean hodograph (PH) curves* (see [91] for instance), very useful in CAGD, where this square-root is simplified. In more detail, a planar curve $\mathcal{C} \subset \mathbb{R}^2$

parametrized by $\boldsymbol{x}(t) = (x_1(t), x_2(t))$ is a PH curve if there is a function $\sigma(t)$ such that

$$[x_1'(t)]^2 + [x_2'(t)]^2 = \|\boldsymbol{x}'(t)\|^2 = [\sigma(t)]^2. \tag{4.20}$$

Thus, $ds = \|\boldsymbol{x}'(t)\|dt = \sigma(t)dt$. Furthermore, the image of a PH curve under a similarity is also a PH curve. For these curves, the integrals in Eq. (4.1) and Eq. (4.3) can be simplified. If $\boldsymbol{x}(t)$ is, additionally, rational, these integrals become (improper) rational integrals, so they can be computed by using the Residues' Theorem (see Chapter 7 of [39]). In more detail, an integral

$$\int_{-\infty}^{\infty} f(t)dt$$

where $f(t) = \frac{p(t)}{q(t)}$ with p, q relatively prime, is equal to $\sum_{i=1}^{k} Res(f, z_i)$, where the z_i are the complex roots of q with positive complex part, and $Res(f, z_i)$ denotes the residue of f at $z = z_i$. Recall that if z_i is a complex root of multiplicity m_i of z_i, and $\phi(t) = f(t)(t - t_i)^{m_i}$, then

$$Res(f, z_i) = \frac{\phi^{m_i - 1}(z_i)}{(m_i - 1)!},$$

where $\phi^{m_i-1}(z_i)$ is the derivative of $\phi(t)$ of order $m_i - 1$ at $t = z_i$. Example 4.1 illustrates the computation of the similarities between two PH curves.

Example 4.1. *Let the curves $\mathcal{C}_1, \mathcal{C}_2$ be parametrized by*

$$\begin{aligned}
\boldsymbol{x}(t) = &\left(-\frac{2(12t^4 - 56t^3 + 75t^2 - 39t - 23)}{(4t^2 - 24t + 37)(t^2 + 1)^2}, \frac{4(4t^3 - 9t^2 + 6t + 5)}{(4t^2 - 24t + 37)(t^2 + 1)^2} \right), \\
\boldsymbol{y}(t) = &\left(\frac{4(4t^6 - 24t^5 + 45t^4 + 24t^3 - 84t^2 + 84t + 127)}{(4t^2 - 24t + 37)(t^2 + 1)^2}, \right. \\
&\left. \frac{4t^6 - 24t^5 - 387t^4 + 1968t^3 - 2622t^2 + 1380t + 865}{(4t^2 - 24t + 37)(t^2 + 1)^2} \right).
\end{aligned}$$

One can check that both parametrizations correspond to PH curves. In more detail, we get that

$$\|\boldsymbol{x}'(t)\|^2 = [\sigma_1(t)]^2 = \frac{36(8t^5 - 32t^4 + 54t^3 + 7t^2 - 74t + 19)^2}{(t^2+1)^6(4t^2 - 24t + 37)^2},$$
$$\|\boldsymbol{y}'(t)\|^2 = [\sigma_2(t)]^2 = 324 \cdot [\sigma_1(t)]^2,$$

so

$$ds_1 = \sigma_1(t) = \frac{6(8t^5 - 32t^4 + 54t^3 + 7t^2 - 74t + 19)}{(t^2+1)^3(4t^2 - 24t + 37)}, \quad ds_2 = \sigma_2(t) = 18 \cdot \sigma_1(t). \quad (4.21)$$

Now the poles of the components of $\boldsymbol{x}(t), \boldsymbol{y}(t)$ *are* $\pm\mathbf{i}$ *and* $3 \pm \frac{\mathbf{i}}{2}$, *although in order to compute the integrals by means of the Residues' Theorem, it is sufficient to consider the poles* $\mathbf{i}$ *and* $3 + \frac{\mathbf{i}}{2}$, *which belong to the upper half-plane. Using this, we get that* $L_1 = \dfrac{9743377\pi}{2700000}$ *and* $L_2 = \dfrac{9743377\pi}{150000}$. *Hence,*

According to Eq. (4.1), the componenents of the gravity center is given by $\dfrac{1}{L}\displaystyle\int_I x_i(t)ds$. *For* $\int_I x_i(t)ds$, *the pole* $\mathbf{i}$ *has order* 5 *and the pole* $3 + \frac{\mathbf{i}}{2}$ *has order* 2. *Using the Resiue's Theorem we get that the centers of gravity of* $\mathcal{C}_1$ *and* $\mathcal{C}_2$ *are*

$$\mathbf{G}_1 = \left(-\frac{927359}{2605500}, \frac{197291}{144750}\right), \quad \mathbf{G}_2 = \left(\frac{1823869}{72375}, -\frac{1361609}{144750}\right).$$

We translate the curves so that the centers of gravity lie at the origin. Using again the Residues' Theorem we get that

$$\mathbf{T}_1 = \begin{bmatrix} \frac{112086007207}{339431512500} & \frac{9566739442}{28285959375} \\ \frac{9566739442}{28285959375} & -\frac{96523502657}{905150700000} \end{bmatrix}, \quad \mathbf{T}_2 = \begin{bmatrix} -\frac{96523502657}{2793675000} & \frac{38266957768}{349209375} \\ \frac{38266957768}{349209375} & \frac{112086007207}{1047628125} \end{bmatrix}.$$

Thus, we get a similarity between the curves given by

$$\boldsymbol{A} = \begin{pmatrix} 0 & 18 \\ 18 & 0 \end{pmatrix} = 18 \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 2/3 \\ -3 \end{pmatrix}. \quad (4.22)$$

Figure 4.5: Curves $\mathcal{C}_1$ (yellow) and $\mathcal{C}_2$ (blue)

4.2.2 Similarities computation of closed curves without self-intersections

An alternative for computing the similarities between two curves $\mathcal{C}_1, \mathcal{C}_2$, useful in the case when $\mathcal{C}_1, \mathcal{C}_2$ are both closed and do not have self-intersections, is to see them as borders of planar regions Δ_1, Δ_2 respectively, and use statement (2) of Corollary 4.2, which leads to linear systems analogous to Eq. (4.18). In this case we need to compute double integrals over the planar regions Δ_1, Δ_2. However, whenever $\mathcal{C}_i$ does not have self-intersections, we can use Green's Theorem to compute the integrals in Eq. (4.2) and Eq. (4.4). Recall that Green's Theorem states that for a simple (i.e., without self-intersections) curve $\mathcal{C}$ enclosing a planar region Δ, i.e., $\partial\Delta = \mathcal{C}$, and two vector functions $P = P(x_1, x_2)$, $Q = Q(x_1, x_2)$ with continuous partial derivatives, then it holds that

$$\iint_{\Delta} \left(\frac{\partial Q}{\partial x_1} - \frac{\partial P}{\partial x_2} \right) dA = \oint_{\mathcal{C}} (P dx_1 + Q dx_2), \tag{4.23}$$

where the integral at the right-hand side of Eq. (4.23) is the circulation of the vector field $(P(x_1, x_2), Q(x_1, x_2))$ around $\mathcal{C}$, positively oriented.

For instance, if $\mathcal{C}$ is parametrized by $\boldsymbol{x}(t) = (x_1(t), x_2(t))$ with $t \in I$, taking $P = 0$

and $Q = x_1$ we get the following expressions for the area A enclosed by $\mathcal{C}$:

$$A = \int_I x_1(t)x_2'(t)dt. \tag{4.24}$$

Also, using appropriate expressions for P, Q we get that the center of gravity $\mathbf{G}^\star = (\mathbf{x}_1^\star, \mathbf{x}_2^\star)$ of Δ can be computed, for instance, as

$$\mathbf{x}_1^\star = \frac{1}{2}\frac{\displaystyle\int_I [x_1(t)]^2 x_2'(t)dt}{A}, \quad \mathbf{x}_2^\star = -\frac{1}{2}\frac{\displaystyle\int_I [x_2(t)]^2 x_1'(t)dt}{A}. \tag{4.25}$$

The expression is not unique, since other expressions for P, Q are also possible.

Additionally, after translating $\mathbf{G}^\star$ into the origin and using appropriate expressions for P, Q, the inertia tensor can be computed, for instance, in the following way:

$$\mathbf{T}^\star = \frac{1}{A}\begin{bmatrix} -\frac{1}{3}\displaystyle\int_I [x_2(t)]^3 x_1'(t)dt & \frac{1}{2}\displaystyle\int_I [x_1(t)]^2 x_2(t)x_2'(t)dt \\ \frac{1}{2}\displaystyle\int_I [x_1(t)]^2 x_2(t)x_2'(t)dt & \frac{1}{3}\displaystyle\int_I [x_1(t)]^3 x_2'(t)dt \end{bmatrix}. \tag{4.26}$$

In particular, this approach has the advantage that no square roots appear in the integrals of Eq. (4.25) or Eq. (4.26). Furthermore, two special cases, that also appeared in Section 4.2.1, should be mentioned here:

(i) *Rational curves (without self-intersections):* in this case, all the integrals we need to compute are rational integrals. In turn, these integrals can be computed using the Residues' Theorem. Notice that, unlike the previous subsection, in this case this is applicable not only to PH curves, but for all rational compact curves without self-intersections.

(ii) *Trigonometric curves (without self-intersections):* if, as it is common, the parameter t lies in $[0, 2\pi]$, the integrals we need to compute can be reduced to integrals of the type

$$\int_0^{2\pi} F(\sin t, \cos t)dt.$$

Again, these integrals (see Chapter 7 of [39]) can be computed using the Residues'

Theorem. In more detail,

$$\int_0^{2\pi} F(\sin t, \cos t)dt = \oint_C F\left(\frac{z+z^{-1}}{2\mathbf{i}}, \frac{z-z^{-1}}{2}\right)\frac{dz}{\mathbf{i}z}, \tag{4.27}$$

where C is the border of the unit circle, positively oriented. Furthermore, this last integral can be computed by the Residues' Theorem. Notice, first, that in fact Eq. (4.27) amounts to performing the transformation in Eq. (4.30). Thus, the integral in the right-hand side of Eq. (4.27) is the circulation of a rational function, whose denominator is a power of z. Therefore, the rational function in the integrand has just one pole, namely $z = 0$, so the integral can be computed exactly. Hence, we have the following result.

Lemma 4.3. *Let $\mathcal{C}$ be a trigonometric curve parametrized by $\boldsymbol{x}(t) = (x_1(t), x_2(t))$, with $t \in [0, 2\pi]$ and $x_i(t)$ as in Eq. (3.2) for $i = 1, 2$, without self-intersections. Let $\Delta \subset \mathbb{R}^2$ be a planar region satisfying that $\partial\Delta = \mathcal{C}$, and let $\mathbf{G}^\star$ be the center of gravity of Δ, and $\mathbf{T}^\star$ the inertia tensor of Δ. If the coefficients $a_k^{(i)}, b_k^{(i)} \in \mathbb{K}$, with $\mathbb{K}$ a field, then the coordinates of $\mathbf{G}^\star$ and the components of $\mathbf{T}^\star$ also lie in $\mathbb{K}$.*

Example 4.2. *Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be the curves parametrized by*

$$\begin{aligned}
\boldsymbol{x}(t) = \;& (2\cos(4t) + 27\cos(t) - 2\sin(2t) - 6\sin(t), \\
& \cos(t) - \sin(4t) + 0.5\sin(3t) + \sin(2t) - 3\sin(t)), \\
\boldsymbol{y}(t) = \;& (12\cos(t) - 12\sin(4t) + 6\sin(3t) + 12\sin(2t) - 36\sin(t) - 1, \\
& -24\cos(4t) - 324\cos(t) + 24\sin(2t) + 72\sin(t) - 4),
\end{aligned}$$

and let Δ_1, Δ_2 be the planar regions enclosed by $\mathcal{C}_1, \mathcal{C}_2$. Here we have $A_1 = 83\pi$ and $A_2 = 11952\pi$. Hence, from Eq. (4.15) we have

$$\lambda = \sqrt{\frac{A_2}{A_1}} = 12.$$

The curves $\mathcal{C}_1$ and $\mathcal{C}_2$ are shown in Fig. 4.6. Notice that the coefficients of $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ belong to $\mathbb{Q}$. After performing the change in Eq. (4.30), we get complex rational parametrizations, whose components have the pole $z = 0$ with order four.

By using the Residues' Theorem, we get the coordinates of the centers of gravity $\mathbf{G}_1^\star, \mathbf{G}_2^\star$ *of* Δ_1, Δ_2,

$$\mathbf{G}_1^\star = \left(-\frac{115402830925126961583675}{244972761298862845460048}, \frac{383692276733158695802 5}{489945522597725690920 96}\right),$$
$$\mathbf{G}_2^\star = \left(\frac{-475188127338758077109 25}{244972761298862845460 48}, \frac{-463382211131583963392 25}{765539879058946392064}\right).$$

After translating the curves so that the centers of gravity are moved to the origin, the inertia tensors can be computed, again using the Residues' Theorem. The entries of the matrices defining these tensors belong to $\mathbb{Q}$, *but their expressions are long, so we provide floating point approximations:*

$$\mathbf{T}_1^\star \approx \begin{pmatrix} -150.4751 & -11.3433 \\ -11.3433 & -4.9668 \end{pmatrix}, \quad \mathbf{T}_2^\star \approx \begin{pmatrix} -715.2132 & 1633.4379 \\ 1633.4379 & -21668.4178 \end{pmatrix}. \tag{4.28}$$

Solving the system analogous to Eq. (4.18) *but using* $\mathbf{T}_1^\star$ *and* $\mathbf{T}_2^\star$ *instead (with exact coefficients), we get one solution, which corresponds to a similarity between the curves:*

$$\boldsymbol{A} = \begin{pmatrix} 0 & 12 \\ -12 & 0 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} -1 \\ -4 \end{pmatrix}. \tag{4.29}$$

4.3 Computation of approximate similarities

The previous sections are useful when: (1) the curves $\mathcal{C}_1$ and $\mathcal{C}_2$ are defined by exact coefficients (e.g., rational, or belonging to an algebraic extension); (2) $\mathcal{C}_1$ and $\mathcal{C}_2$ are related by an exact similarity; (3) the integrals appearing can be computed exactly. However, in applications, these conditions are rarely satisfied: curves are often defined by floating-point coefficients and are not exactly similar, but *close*, at most, to being

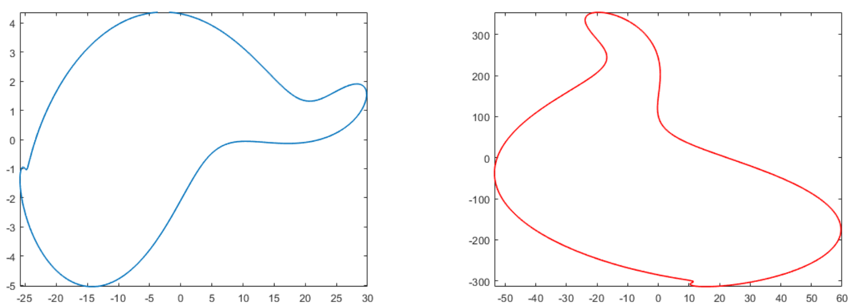


Figure 4.6: $\mathcal{C}_1$ (left) and $\mathcal{C}_2$ (right) of Ex. 4.2

similar. In this situation, which we call the *approximate case*, the approaches in the previous sections must be adapted.

The main consequence of the failure of one or several conditions (1), (2), (3) is that the ranks of the matrices in Eq. (4.18) are equal to two, so the linear systems defined by the matrices in Eq. (4.18), or the corresponding matrices in the case of the approach in Section 4.2.2, only have the trivial solution. In order to address the approximate case, it is useful first to consider an alternative method to computing similarities in the exact case. The method is as follows: since $\alpha^2 + \beta^2 = 1$, we can parametrize α, β as

$$\alpha = \frac{\boldsymbol{a}^2 - 1}{\boldsymbol{a}^2 + 1}, \quad \beta = \frac{2\boldsymbol{a}}{\boldsymbol{a}^2 + 1}. \tag{4.30}$$

Then, for each $i = 1, 2$, Eq. (4.18), or the equivalent system in the approach of Section 4.2.2, provides four quadratic polynomials $R_j(\boldsymbol{a})$, $j = 1, \ldots, 4$, that must have a common nontrivial gcd $\mathcal{R}(\boldsymbol{a})$ for the curves to be similar. The real roots $\boldsymbol{a}_\ell$ of $\mathcal{R}(\boldsymbol{a})$ provide the tentative similarities.

In the approximate case, we still compute the polynomials $R_j(\boldsymbol{a})$ as in the exact case, but because of numerical inaccuracies the exact gcd of the $\mathcal{R}(\boldsymbol{a})$ will be constant. So we replace the exact gcd $\mathcal{R}(\boldsymbol{a})$ by the approximate gcd $\widehat{\mathcal{R}}(\boldsymbol{a})$ of the $R_j(\boldsymbol{a})$, as in Sec. 3.4. Certainly, the problem is how to choose ϵ, which depends on the innacuracies due to the conditions (1), (2), (3) described at the beginning of the section. In Subsection 4.3.2 we will address this problem assuming that conditions (1) and (2) hold, but (3) fails, i.e., that the only source of innacuracy is the fact that the integrals providing the gravity centers and the inertia tensors have been computed numerically.

The next examples illustrate these ideas.

Example 4.3. *Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be the curves parametrized by*

$$
\begin{aligned}
\boldsymbol{x}(t) &= \left(\frac{5^{2t}+2t-4}{5^{4t}+t^2+6}, \frac{3^t-5t}{3^{3t}+8t^2}\right),\\
\boldsymbol{y}(t) &= \left(\frac{3(5^{2t}+2t-4)}{4(5^{4t}+t^2+6)} - \frac{3\sqrt{3}(3^t-5t)}{4(3^{3t}+8t^2)} + 2, \frac{3\sqrt{3}(5^{2t}+2t-4)}{4(5^{4t}+t^2+6)} + \frac{3(3^t-5t)}{4(3^{3t}+8t^2)}\right).
\end{aligned}
$$

The centers of gravity of $\mathcal{C}$ and $\mathcal{D}$ are respectively

$$\mathbf{G}_1 \approx (-0.54247, 0.98405), \quad \mathbf{G}_2 \approx (0.31483, 0.03335).$$

We translate the curves and find their inertia tensor matrices which in this case are given by

$$\mathbf{T}_1 \approx \begin{pmatrix} 0.453975 & 0.076403 \\ 0.076403 & 0.060162 \end{pmatrix}, \quad \mathbf{T}_2 \approx \begin{pmatrix} 0.208009 & 0.2977297 \\ 0.2977297 & 0.948801 \end{pmatrix}. \tag{4.31}$$

The curves $\mathcal{C}_1$ and $\mathcal{C}_2$, shown in Fig. 4.7, are related by the (exact) similarity given by

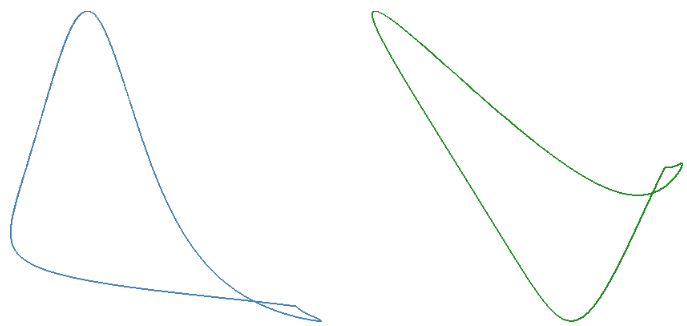


Figure 4.7: $\mathcal{C}_1$ (left) and $\mathcal{C}_2$ (right).

$$\begin{aligned} \boldsymbol{A}_1 &= \frac{3}{2}\begin{pmatrix} 1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{pmatrix} \approx \begin{pmatrix} 0.75 & -1.2990381 \\ 1.2990381 & 0.75 \end{pmatrix}, \\ \boldsymbol{b}_1 &= \begin{pmatrix} 2 \\ 0 \end{pmatrix}. \end{aligned} \tag{4.32}$$

Let us perturb the parametrizations above; the new curves $\mathcal{C}_1$ *and* $\mathcal{C}_2$ *are*

$$\boldsymbol{x}(t) = \left(\frac{5^{2t}+2.000016t-3.999899}{5^{4t}+0.99998t^2+5.99997}, \frac{3^t-4.99991t}{3^{3t}+8.0000101t^2}\right),$$

$$\begin{aligned} \boldsymbol{y}(t) = \Bigg(& \frac{3.000012(5^{2t}+1.99997t-4.00012)}{3.99989(5^{4t}+1.0000025t^2+5.99991)} - \frac{2.9999994\sqrt{3}(3^t-4.99998t)}{4.0000015(3^{3t}+7.999801t^2)} + 1.999902, \\ & \frac{2.999918\sqrt{3}(5^{2t}+2.0000101t-4)}{3.999899(5^{4t}+t^2+6.00003)} + \frac{3.0001012(3^t-5.00004t)}{4.00007(3^{3t}+8.0001004t^2)}\Bigg). \end{aligned}$$

From Eq. (4.18) *we get the following four polynomial equations:*

$$\begin{aligned} -0.469644\boldsymbol{a}^2 + 1.6269146\boldsymbol{a} + 0.469644 &\approx 0, \\ -0.07264187\boldsymbol{a}^2 + 0.25165869\boldsymbol{a} + 0.07264187 &\approx 0, \\ -0.07264353\boldsymbol{a}^2 + 0.25165869\boldsymbol{a} + 0.07264353 &\approx 0 \\ 0.469644\boldsymbol{a}^2 - 1.62691796\boldsymbol{a} - 0.4696440 &\approx 0. \end{aligned}$$

Applying the `uvGCD` *method with a tolerance of* 10^{-5}*, we get the aprroximate gcd*

$$\widehat{\mathcal{R}}(\boldsymbol{a}) \approx 0.222182707719528\boldsymbol{a}^2 - 0.769674334584299\boldsymbol{a} - 0.222182707719528,$$

with two solutions for $\boldsymbol{a}$*, namely,*

$$\boldsymbol{a}_1 \approx -0.267945918844864, \quad \boldsymbol{a}_2 \approx 3.732096403300630.$$

The lower bound for the approximate gcd is $3.2593 \cdot 10^{-6}$*. Finally, we get the solutions below for* α *and* β *:*

$$\alpha \approx \pm 49999470978778656, \quad \beta \approx \pm 0.8660284580683403,$$

which give rise to the approximate similarity defined by

$$\boldsymbol{A}_2 \approx \begin{pmatrix} 0.750012903 & -1.29907878 \\ 1.29907878 & 0.750012903 \end{pmatrix}$$
$$\boldsymbol{b}_2 \approx \begin{pmatrix} 1.9999048583326 \\ -0.00000414266776 \end{pmatrix}.$$

Example 4.4. *Let us consider the curves* $\mathcal{C}_1, \mathcal{C}_2$*, parametrized by*

$$\boldsymbol{x}(t) = (2\cos(t) + \cos(2t), 4\sin(t) - \sin(4t))$$

and $\boldsymbol{y}(t) = (y_1(t), y_2(t))$*, with* $t \in [0, 2\pi]$*, where*

$$y_1(t) = -\frac{39\sqrt{10}}{10}\sin(4t) + \frac{13\sqrt{10}}{10}\cos(2t) + \frac{13\sqrt{10}}{5}\cos(t) + \frac{78\sqrt{10}}{5}\sin(t) - 1,$$
$$y_2(t) = \frac{13\sqrt{10}}{10}\sin(4t) + \frac{39\sqrt{10}}{10}\cos(2t) + \frac{39\sqrt{10}}{5}\cos(t) - \frac{26\sqrt{10}}{5}\sin(t) + 3.$$

The curves $\mathcal{C}_1, \mathcal{C}_2$ are related by

$$\boldsymbol{A}_1 = 13 \begin{pmatrix} \frac{\sqrt{10}}{10} & \frac{3\sqrt{10}}{10} \\ \frac{3\sqrt{10}}{10} & -\frac{\sqrt{10}}{10} \end{pmatrix}, \quad \boldsymbol{b}_1 = \begin{pmatrix} -1 \\ 3 \end{pmatrix},$$
$$\boldsymbol{A}_2 = 13 \begin{pmatrix} \frac{\sqrt{10}}{10} & -\frac{3\sqrt{10}}{10} \\ \frac{3\sqrt{10}}{10} & \frac{\sqrt{10}}{10} \end{pmatrix}, \quad \boldsymbol{b}_2 = \begin{pmatrix} -1 \\ 3 \end{pmatrix}.$$

Now, we apply a random perturbation of order 10^{-6} to all the coefficients of the parameterizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$.

The centers of gravity of $\mathcal{C}_1$ and $\mathcal{C}_2$ are respectively

$$\mathbf{G}_1^\star \approx (0.375, 2.1229), \quad \mathbf{G}_2^\star \approx (0.5416, 7.6248).$$

We translate the curves and find their inertia tensor matrices which in this case are given by

$$\mathbf{T}_1^\star \approx \begin{pmatrix} 0.85937 & 0 \\ 0 & 5.5000009 \end{pmatrix}, \quad \mathbf{T}_2^\star \approx \begin{pmatrix} 851.0735 & -235.2797 \\ -235.2797 & 223.6609 \end{pmatrix}. \tag{4.33}$$

We also get $\lambda \approx 13.0000023314$. We apply `uvGCD` *method with a tolerance of 10^{-5} and we get four solutions for $\boldsymbol{a}$, namely,*

$$\boldsymbol{a}_{1,2} \approx \pm 6.162283129064051, \quad \boldsymbol{a}_{3,4} \approx \pm 0.162277516150395.$$

The lower bound for the approximate gcd is approximately 0.00014. Finally, we get the solutions below for α and β :

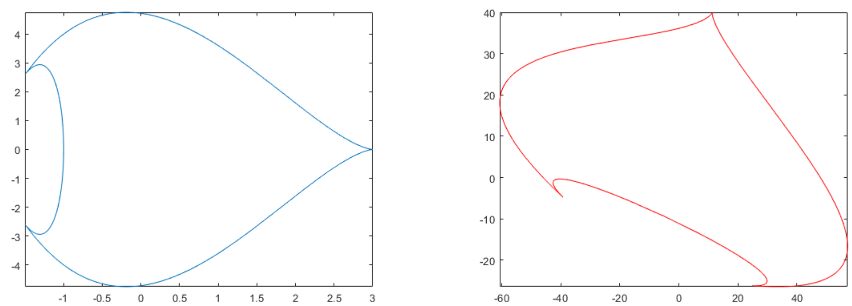


Figure 4.8: $\mathcal{C}_1$ (left) and $\mathcal{C}_2$ (right) of Example 4.4

$$\alpha \approx \pm 0.316227499773, \quad \beta \approx \pm 0.948683386798,$$

which give rise to the similarities below

$$\boldsymbol{A}_1 \approx \begin{pmatrix} 4.110956796573 & 12.33288192694 \\ 12.33288192694 & -4.110956796573 \end{pmatrix}$$
$$\boldsymbol{A}_2 \approx \begin{pmatrix} 4.110956796573 & -12.33288192694 \\ 12.33288192694 & 4.110956796573 \end{pmatrix}$$
$$\boldsymbol{b} \approx \begin{pmatrix} -0.9999997 \\ 2.9999989 \end{pmatrix}.$$

The relative errors with respect to the original similarity are around $3.3 \cdot 10^{-7}$*. The curves* $\mathcal{C}_1, \mathcal{C}_2$ *are shown in Fig. 4.8.*

The suggested method provides tentative approximate similarities f between $\mathcal{C}_1$ and $\mathcal{C}_2$. However, in order to test each f, we need to check whether $f(\mathcal{C}_1)$ and $\mathcal{C}_2$ are "close". This is difficult, and several observations on this matter were already made in the previous chapter. In our case, we developed a heuristic procedure to check closeness, that works as follows:

- Take a partition of points $\{P_1, \ldots, P_n\}$ in the curve $f(\mathcal{C}_1)$. If the interval I_1 where the parameter of $\mathcal{C}_1$ takes values is finite, we can consider a uniform distribution on I_1, and then pick a random sample so that the P_i are the images under f of the points in the sample. If I_1 is infinite, we consider the stereographic projection $\pi : S^1 \to \mathbb{R}$ from the unit circle S^1 onto the real line, consider a uniform distribution on S^1, pick a random sample on S^1, map the points to I_1, and proceed as in the case where I_1 is finite.

- If P_i is close to $\mathcal{C}_2$, then the normal line L_i to $f(\mathcal{C}_1)$ at P_i should be almost normal to $\mathcal{C}_2$.

- We intersect L_i with $\mathcal{C}_2$, compute the closest point to P_i in $L_i \cap \mathcal{C}_2$, and compute the distance d_i between this point, that we denote by Q_i, and P_i.

- Finally, we consider the arithmetic mean of all the d_i, and we divide it by the length of $\mathcal{C}_2$, in order to compare its value with the size of the curve; we represent by ν the resulting number.

For instance, applying this method in Ex. 4.3 we get $\nu = 0.0326$.

4.3.1 Algorithm and experimentation

In Algorithm `Similar-Plane-Curves` we summarize the ideas given in Subsection 4.2.1 and Section 4.3. Here we do not include the ideas about closed curves without self-intersections, since they are more suited for specific cases; nevertheless, we have also tested these other ideas in appropriate examples. Both the exact and approximate versions are included in Algorithm `Similar-Plane-Curves`.

The whole procedure has been implemented in MATLAB. In Table 4.1 we summarize the performance of the algorithm on 12 representative examples; the parametrizations of the curves $\mathcal{C}_1$ are provided in Table 4.2. In each case, $\mathcal{C}_2$ is the result of applying to the curve $\mathcal{C}_1$ the similarity $f(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$ defined by

Algorithm 4 Similar-Plane-Curves

Require: Two bounded curves $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$, parametrized by $\boldsymbol{x}(t), \boldsymbol{y}(t)$.
Ensure: The similarities $f(\mathbf{x}) = \lambda \boldsymbol{Q}\mathbf{x} + \boldsymbol{b}$ (approximate in some cases) between $\mathcal{C}_1, \mathcal{C}_2$.

1: Compute the total lengths L_1, L_2 of $\mathcal{C}_1, \mathcal{C}_2$ and use Eq. (4.14) to compute the value of λ.
2: Compute the gravity centers of both $\mathcal{C}_1, \mathcal{C}_2$ using Eq. (4.1).
3: Translate $\mathcal{C}_1, \mathcal{C}_2$ so that their gravity centers are at the origin.
4: **if** the coefficients of the parametrizations are given exactly **then**
5: compute the inertia tensor matrices of the translated curves using Eq. (4.3).
6: solve the systems $\mathcal{S}_1, \mathcal{S}_2$ derived from Eq. (4.18) in each case, together with the equation $\alpha^2 + \beta^2 = 1$ to find the entries α, β of the posible matrices $\boldsymbol{Q}$, using the `uvGCD` method when necessary.
7: **else**
8: compute the inertia tensor matrices of the translated curves using Eq. (4.35).
9: use the `uvGCD` method to solve the systems $\mathcal{S}_1, \mathcal{S}_2$, this time using the parametrization of the circle $\alpha^2 + \beta^2 = 1$ given by Eq. (4.30) to find the entries α, β of $\boldsymbol{Q}$.
10: **end if**
11: Verify how close $f(\mathcal{C}_1)$ and $\mathcal{C}_2$ are by computing the distance ν.

$$\boldsymbol{A} = \lambda \begin{pmatrix} 3/5 & 4/5 \\ -4/5 & 3/5 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} -1/2 \\ 1/3 \end{pmatrix}, \tag{4.34}$$

with λ a positive random constant, plus a perturbation in the coefficients of order of magnitude less than 10^{-6}. For each curve, in Table 4.1 we include: the type of the curve; the method used to compute the integrals; the total CPU time in seconds; the relative errors between the matrix $\tilde{\boldsymbol{A}}$ and the vector $\tilde{\boldsymbol{b}}$ defining the approximate similarity $\tilde{f}(\boldsymbol{x}) = \tilde{\boldsymbol{A}}\boldsymbol{x} + \tilde{\boldsymbol{b}}$ we get, compared to the matrix $\boldsymbol{A}$ and the vector $\boldsymbol{b}$ in Eq. (4.34), measured as $NA = \dfrac{\|\boldsymbol{A} - \tilde{\boldsymbol{A}}\|_2}{\|\boldsymbol{A}\|_2}$ and $Nb = \dfrac{\|\boldsymbol{b} - \tilde{\boldsymbol{b}}\|}{\|\boldsymbol{b}\|}$, where $\| \bullet \|_2$ represents the 2-norm; and the number ν to measure the geometric distance between the given curve $\mathcal{C}_2$ and the obtained curve $f(\mathcal{C}_1)$.

Type of curve	Par.	Method used	CPU time (secs.)	Norms	ν
Trigonometric with 1 autointersection	$\boldsymbol{x}_1(t)$	Numeric integration	2.1792	$NA \approx 2.3 \cdot 10^{-6}$ $Nb \approx 1.2 \cdot 10^{-6}$	0.0178
Trigonometric with 10 autointersections	$\boldsymbol{x}_2(t)$	Numeric integration	2.5315	$NA \approx 8.3 \cdot 10^{-8}$ $Nb \approx 1.6 \cdot 10^{-6}$	0.0138
PH curve with 2 autointersections	$\boldsymbol{x}_3(t)$	Residue Theorem	5.0199	$NA \approx 3.6 \cdot 10^{-8}$ $Nb \approx 4.6 \cdot 10^{-8}$	0.00000022
Rational with 3 autointersections	$\boldsymbol{x}_4(t)$	Numeric integration	3.3001	$NA \approx 8.5 \cdot 10^{-8}$ $Nb \approx 3.1 \cdot 10^{-5}$	0.000000064
Radical	$\boldsymbol{x}_5(t)$	Numeric integration	2.6718	$NA \approx 5.4 \cdot 10^{-7}$ $Nb \approx 1.4 \cdot 10^{-6}$	0.0043
Exponential	$\boldsymbol{x}_6(t)$	Numeric integration	2.8619	$NA \approx 1.2 \cdot 10^{-6}$ $Nb \approx 2.1 \cdot 10^{-7}$	0.02996
Exponential	$\boldsymbol{x}_7(t)$	Numeric integration	2.7409	$NA \approx 2.3 \cdot 10^{-6}$ $Nb \approx 4.7 \cdot 10^{-5}$	0.106008052
Logarithmic	$\boldsymbol{x}_8(t)$	Numeric integration	2.7590	$NA \approx 9.1 \cdot 10^{-6}$ $Nb \approx 6.2 \cdot 10^{-5}$	0.00625
Expo-trigonometric	$\boldsymbol{x}_9(t)$	Numeric integration	2.8081	$NA \approx 2.6 \cdot 10^{-6}$ $Nb \approx 1.4 \cdot 10^{-6}$	0.02645
Rational with no autointersections	$\boldsymbol{x}_{10}(t)$	Residue and Green's Theorems	5.4636	$NA \approx 1.4 \cdot 10^{-6}$ $Nb \approx 8.5 \cdot 10^{-8}$	0.00000049
Trigonometric with no autointersections	$\boldsymbol{x}_{11}(t)$	Residue and Green's Theorem	2.7412	$NA \approx 1.9 \cdot 10^{-7}$ $Nb \approx 1.5 \cdot 10^{-5}$	0.014156
Trigonometric with no autointersections	$\boldsymbol{x}_{12}(t)$	Residue and Green's Theorem	2.9124	$NA \approx 7.1 \cdot 10^{-8}$ $Nb \approx 1.5 \cdot 10^{-5}$	0.01736

Table 4.1

List of the parametrizations in Table 4.1
$\boldsymbol{x}_1(t) = (\cos(t) - 3\sin(2t), 2\cos(t) - \sin(t))$
$\boldsymbol{x}_2(t) = \Big(5\sin(5t) - \frac{1}{2}\cos(2t) + \sin(2t) + \cos(t) - 3\sin(t) - 7,$ $\sin(4t) - 8\cos(3t) - 8\cos(2t) - \sin(2t) + 9\cos(t) - 2\sin(t)\Big)$
$\boldsymbol{x}_3(t) = \left(-\frac{24t^4 - 112t^3 + 150t^2 - 78t - 46}{(t^2+1)^2(4t^2 - 24t + 37)}, \frac{16t^3 - 36t^2 + 24t + 20}{(t^2+1)^2(4t^2 - 24t + 37)}\right)$
$\boldsymbol{x}_4(t) = \left(\frac{28t^5 + 2t^4 - 3t^3 + 3t - 19}{2t^8 + t^2 + 1}, \frac{37t^6 + 8t^5 + 9t^4 - 4t^3 + 6}{2t^8 + t^2 + 1}\right)$
$\boldsymbol{x}_5(t) = \left(\frac{\sqrt{t}}{t^4+1}, \frac{94\sqrt{t}+1}{18t^2+2}\right)$
$\boldsymbol{x}_6(t) = \left(\frac{2^t}{2^{3t}+4}, \frac{2^{3t}}{2^{4t}+2}\right)$
$\boldsymbol{x}_7(t) = \left(\frac{e^t}{t^6 e^t + 2}, \frac{t^3 - t + 1}{t^6 + 2}\right)$
$\boldsymbol{x}_8(t) = \left(-\frac{\ln(t^2+1)}{t^4+4}, \frac{t^2 - 2t - 4}{t^4+4}\right)$
$\boldsymbol{x}_9(t) = \left(\sin(t)\left(e^{\cos(t)} - 2\cos(t) - \sin^5\left(\frac{t}{3}\right)\right), \cos(t)(e^{\sin(t)} + \cos(4t))\right)$
$\boldsymbol{x}_{10}(t) = \left(\frac{t^5 + 5t^4 + 2t^3 - t^2 + 2t}{(25t^2+9)^2(t^2+14t+50)^3}, \frac{t^6 + 9t^5 + 0.5t^3 - 3t^2 + 4t}{(25t^2+9)^2(t^2+14t+50)^3}\right)$
$\boldsymbol{x}_{12}(t) = \Big(\sin(4t) - 2\cos(3t) - 3\cos(2t) + 4\sin(2t) - 8\cos(t) - 7\sin(t) + 5,$ $-2\cos(4t) + \frac{4}{3}\sin(3t) - 17\sin(2t) + 2\cos(t) - 13\sin(t) - 1\Big)$
$\boldsymbol{x}_{11}(t) = (2\cos(3t) - 3\cos(2t) + 4\sin(2t) - 6\cos(t) - 7\sin(t),$ $\sin(3t) - \sin(2t) + 2\cos(t) - 3\sin(t) + 3)$

Table 4.2

4.3.2 Error analysis for curves with exact coefficients and exact similarities

In this subsection, we will assume that the curves $\mathcal{C}_1$ and $\mathcal{C}_2$ are given exactly and are related by an exact similarity, but that the integrals that we need to calculate are computed numerically, with an error bounded by δ (which depends on the numerical solver). Additionally, we will address the case when $\mathcal{C}_1$ and $\mathcal{C}_2$ are considered as hollow so that the integrals that we need to compute are univariate; the analysis for closed curves and areas is completely analogous.

Under these assumptions, the question is how to choose the value of ϵ to compute the ϵ-gcd of the polynomials $R_j(\boldsymbol{a})$. In order to do this, we consider the following notation:

- The *exact* gravity center $\mathbf{G}_i = (\mathbf{x}_1^{(i)}, \mathbf{x}_2^{(i)})$ of each curve $\mathcal{C}_i$ satisfies that $\mathbf{G}_i = (\tilde{\mathbf{x}}_1^{(i)} + \delta_{1,(i)}, \tilde{\mathbf{x}}_2^{(i)} + \delta_{2,(i)})$, where $\tilde{\mathbf{G}}_i = (\tilde{\mathbf{x}}_1^{(i)}, \tilde{\mathbf{x}}_2^{(i)})$ is the approximate gravity center that we actually compute, and $\delta_{1,(i)}, \delta_{2,(i)}$ are numerical errors bounded by δ.

- The *exact* length L_i of $\mathcal{C}_i$ satisfies that $L_i = \tilde{L}_i + \delta_{3,(i)}$, where $\tilde{L}_i$ is the approximate length that we actually compute, and $\delta_{3,(i)}$ is the numerical error of this computation, which is also bounded by δ.

- The *exact* inertia tensors $\mathbf{T}_i$ satisfy that $\mathbf{T}_i = \tilde{\mathbf{T}}_i + \mathbf{E}_i$, where $\mathbf{E}_i$ is a matrix storing the numerical errors in the computation of the elements of the tensor. Notice that the errors in the computation of the inertia tensors are linked to the errors in the computation of the centers of gravity and the lengths of the curve, so $\mathbf{E}_i$ is a matrix whose entries must be derived. We will consider this problem later in the section. The entries $t_{k\ell}^{(i)}$ of $\mathbf{T}_i$ satisfy that $t_{k\ell}^{(i)} = \tilde{t}_{k\ell}^{(i)} + \delta_{k\ell,(i)}$, where $\delta_{k\ell,(i)}$ denotes the error in the computation of $t_{k\ell}^{(i)}$.

In the exact case, we apply a translation to each curve $\mathcal{C}_i$ so that its center of gravity coincides with the origin, and then we compute the inertia tensor according to Eq. (4.3). However, in the approximate case, it is better to work with the general expression for the inertia tensor of a body whose center of gravity is $\mathbf{G} = (\mathbf{x}_1, \mathbf{x}_2)$ (not necessarily the origin), which is

$$\mathbf{T} = \frac{1}{L}\begin{bmatrix} \int_I (x_2(t)-\mathbf{x}_2)^2 ds & -\int_I (x_1(t)-\mathbf{x}_1)(x_2(t)-\mathbf{x}_2) ds \\ -\int_I (x_1(t)-\mathbf{x}_1)(x_2(t)-\mathbf{x}_2) ds & \int_I (x_1(t)-\mathbf{x}_1)^2(t) ds \end{bmatrix}. \tag{4.35}$$

Considering $\mathbf{G}^*$ (not as the origin) the inertia tensor is given by

$$\mathbf{T}^\star = \frac{1}{A}\begin{bmatrix} -\frac{1}{3}\int_I [x_2(t)-\mathbf{x}_2^*]^3 x_1'(t) dt & \frac{1}{2}\int_I [x_1(t)-\mathbf{x}_1^*]^2[x_2(t)-\mathbf{x}_2^*] x_2'(t) dt \\ \frac{1}{2}\int_I [x_1(t)-\mathbf{x}_1^*]^2[x_2(t)-\mathbf{x}_2^*] x_2'(t) dt & \frac{1}{3}\int_I [x_1(t)-\mathbf{x}_1^*]^3 x_2'(t) dt \end{bmatrix}. \tag{4.36}$$

Using these expressions for the inertia tensors of $\mathcal{C}_1, \mathcal{C}_2$, Theorem 4.2 also holds.

Now the polynomials whose approximate gcd we must compute stem from Eq. (4.18), writing α, β as in Eq. (4.30). For instance, the first polynomial is (recall that the variable is $\boldsymbol{a}$)

$$R_1(\boldsymbol{a}) = \left(t_{11}^{(2)} - \lambda^2 t_{11}^{(1)}\right)\boldsymbol{a}^2 + 2\left(t_{12}^{(2)} - \lambda^2 t_{12}^{(1)}\right)\boldsymbol{a} - \left(t_{11}^{(2)} - \lambda^2 t_{11}^{(1)}\right). \tag{4.37}$$

The above expression is written in terms of exact quantities. So we need to study how the constant λ and the elements of the inertia tensors are affected by the numerical errors in the computation of the centers of gravity and the lengths of the curves, and by the numerical errors of the integrals themselves. In order to do this, we will assume that $\delta \ll \tilde{L}_i$. Under this assumption, we have

$$\frac{1}{L_i} = \frac{1}{\tilde{L}_i + \delta_{3,(i)}} = \frac{1}{\tilde{L}_i\left(1 + \delta_{3,(i)}/\tilde{L}_i\right)} \approx \frac{1}{\tilde{L}_i}. \tag{4.38}$$

Furthermore, we also have that

$$\lambda = \frac{L_2}{L_1} = \frac{\tilde{L}_2 + \delta_{3,(2)}}{\tilde{L}_1 + \delta_{3,(1)}} \approx \frac{\tilde{L}_2}{\tilde{L}_1} = \tilde{\lambda}. \tag{4.39}$$

Now we consider the element $t_{11}^{(1)}$ of the inertia tensor $\mathbf{T}_i$. It holds that

$$t_{11}^{(1)} = \frac{1}{L_1}\int_{I_1}(x_1 - \mathbf{x}_1)^2 ds = \frac{1}{L_1}\int_{I_1}(x_1 - \tilde{\mathbf{x}}_1 - \delta_{1,(1)})^2 ds$$

Using Eq. (4.38) and developing $(x_1 - \tilde{\mathbf{x}}_1 - \delta_{1,(1)})^2 = (x_1 - \tilde{\mathbf{x}}_1)^2 - 2(x_1 - \tilde{\mathbf{x}}_1)\delta_{1,(1)} + \delta_{1,(1)}^2$, we get that

$$t_{11}^{(1)} \approx \tilde{t}_{11}^{(1)} - 2\delta_{1,(1)}^2 + \delta_{1,(1)}^2 = \tilde{t}_{11}^{(1)} - \delta_{1,(1)}^2. \tag{4.40}$$

One can obtain similar expressions for the remaining $t_{k\ell}^{(i)}$. Next, using Eq. (4.37), we get that

$$R_1(\boldsymbol{a}) \approx \tilde{R}_1(\boldsymbol{a}) - \delta\tilde{R}_1(\boldsymbol{a}), \tag{4.41}$$

where

$$\delta\tilde{R}_1(\boldsymbol{a}) = (\delta_{1,(2)}^2 - \tilde{\lambda}^2\delta_{1,(1)}^2)\boldsymbol{a}^2 + 2(\delta_{1,(2)}\delta_{2,(2)} + \tilde{\lambda}^2\delta_{1,(1)}\delta_{2,(1)})\boldsymbol{a} + (\delta_{1,(2)}^2 - \tilde{\lambda}^2\delta_{1,(1)}^2) \tag{4.42}$$

and $\tilde{R}_1(\boldsymbol{a})$ corresponds to $R_1(\boldsymbol{a})$, but replacing $t_{k\ell}^{(i)}$ by $\tilde{t}_{k\ell}^{(i)}$, and λ by $\tilde{\lambda}$; thus, $\tilde{R}_1(\boldsymbol{a})$ is the polynomial that we actually compute, and $\delta\tilde{R}_1(\boldsymbol{a})$ measures the error in the computation of this polynomial. Notice that the 1-norm of $\delta\tilde{R}_1(\boldsymbol{a})$ is bounded by $4(1+\tilde{\lambda}^2)\delta^2$, where δ is the bound for the numerical error in the computation of the integrals.

We can proceed similarly for the other polynomials stemming from Eq. (4.18). In all the cases, we get the same result, i.e., in each case, we get an error term corresponding to a polynomial whose 1-norm is bounded by $4(1+\tilde{\lambda}^2)\delta^2$. Thus, we conclude that in the computation of the ϵ-gcd we can use $\epsilon = 4(1+\tilde{\lambda}^2)\delta^2$, whenever we consider the infinity norm.

The following proposition summarizes the preceding ideas.

Proposition 4.3. *Let $\mathcal{C}_1$ and $\mathcal{C}_2$ two parametric bounded curves with exact coefficients, related by an exact similarity. Let L_i be the total length of $\mathcal{C}_i$, for $i = 1, 2$, and let δ be a bound for the numerical error of the integrals involved in the computation of their centers of gravity and inertia tensors. If $\delta \ll L_i$ then the polynomials $R_j(\boldsymbol{a})$ have a nontrivial ϵ-gcd, with $\epsilon \leq 4(1 + \tilde{\lambda}^2)\delta^2$.*

Additionally, given two polynomials $\tilde{p}_1$ and $\tilde{p}_2$ there are algorithms [30, 31], implemented in Maple (see the command `DistanceToCommonDivisors`) to evaluate how close $\tilde{p}_1, \tilde{p}_2$ are to having a nontrivial gcd. More precisely, and denoting $\deg(\tilde{p}_1) = m$, $\deg(\tilde{p}_2) = n$,, we define

$$d(\tilde{p}_1, \tilde{p}_2) := \inf\{\|(\tilde{p}_1 - p_1^*, \tilde{p}_2 - p_2^*)\| : (p_1^*, p_2^*) \,\text{have a common root}, \deg(p_1^*) \leq m, \deg(p_2^*) \leq n\}.$$

Here, $\|(f, g)\|$ denotes the maximum of $\|f\|_1, \|g\|_1$; recall that the polynomial norm $\| \bullet \|_1$ is the sum of the absolute values of the coefficients of $\bullet$. Methods to compute a lower bound τ for $d(\tilde{p}_1, \tilde{p}_2)$ are given in [30, 31]. In particular, if $\tilde{p}_1, \tilde{p}_2$ are coprime, whenever they undergo perturbations smaller than τ, they will remain coprime. Thus, the value of ϵ so that $\tilde{p}_1, \tilde{p}_2$ have a nontrivial ϵ-gcd must be, at least, τ.

Applying this to the polynomials $\tilde{R}_1(\boldsymbol{a})$ and $\tilde{R}_2(\boldsymbol{a})$, and calling $B = 4(1 + \tilde{\lambda}^2)\delta^2$ to the bound in Proposition 4.3, we get that

$$\tau \leq \epsilon \leq B.$$

In particular, if $B < \tau$, no similarity exists.

4.4 Generalizations of the method

The algorithm stemming from Subsection 4.2.1 can be generalized to bounded curves in $\mathbb{R}^n$. In this case, the center of gravity of $\mathcal{C} \subset \mathbb{R}^n$ parametrized by

$$\boldsymbol{x}(t) = (x_1(t), x_2(t), \ldots, x_n(t))$$

is $\mathbf{G} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$, where

$$\mathbf{x}_i = \frac{1}{L}\int_I x_i(t)ds, \tag{4.43}$$

with $i = 1, 2, \ldots, n$, $ds = \sqrt{x_1'^2(t) + x_2'^2(t) + \cdots + x_n'^2(t)} = \|\boldsymbol{x}'(t)\|dt$, and $L = \int_I ds$.

The inertia tensor in this case is the second order tensor defined by the matrix $\mathbf{T} = \frac{1}{L} \cdot \tilde{\mathbf{T}}$, where $\tilde{\mathbf{T}}$ is given by

$$\begin{bmatrix} \int_I [x_2^2(t) + \cdots + x_n^2(t)]ds & -\int_I x_1(t)x_2(t)ds & \cdots & -\int_I x_1(t)x_n(t)ds \\ -\int_I x_1(t)x_2(t)ds & \int_I [x_1^2(t) + x_3^2(t) + \cdots + x_n^2(t)]ds & \cdots & -\int_I x_2(t)x_n(t)ds \\ \vdots & \vdots & \ddots & \vdots \\ -\int_I x_1(t)x_n(t)ds & -\int_I x_2(t)x_n(t)ds & \cdots & \int_I [x_1^2(t) + \cdots + x_{n-1}^2(t)]ds \end{bmatrix}.$$

The results can also be generalized to surfaces, at the cost of computing double integrals. If $S \subset \mathbb{R}^3$ is a surface parametrized by

$$\boldsymbol{x} : \Delta \subset \mathbb{R}^2 \longrightarrow \mathbb{R}^3, \quad \boldsymbol{x}(t,s) = (x_1(t,s), x_2(t,s), x_3(t,s)),$$

then surface area of S is given by

$$A(S) = \iint_\Delta dS = \iint_\Delta \left\| \frac{\partial \boldsymbol{x}(t,s)}{\partial t} \times \frac{\partial \boldsymbol{x}(t,s)}{\partial s} \right\| dtds, \tag{4.44}$$

the center of gravity is $\mathbf{G} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$, where

$$\mathbf{x}_i = \frac{1}{A(S)}\iint_\Delta x_i(t,s)dS, \quad i = 1, 2, 3, \tag{4.45}$$

and the inertia tensor is represented by the $(3,3)$-matrix given by

$$\mathbf{T} = \frac{1}{A(S)} \begin{bmatrix} \iint_\Delta [x_2^2(t,s) + x_3^2(t,s)] dS & -\iint_\Delta x_1(t,s) x_2(t,s) dS & -\iint_\Delta x_1(t,s) x_3(t,s) dS \\ -\iint_\Delta x_1(t,s) x_2(t,s) dS & \iint_\Delta [x_1^2(t,s) + x_3^2(t,s)] dS & -\iint_\Delta x_2(t,s) x_3(t,s) dS \\ -\iint_\Delta x_1(t,s) x_3(t,s) dS & -\iint_\Delta x_2(t,s) x_3(t,s) dS & \iint_\Delta [x_1^2(t,s) + x_2^2(t,s)] dS \end{bmatrix}.$$

CONCLUSION AND FUTURE WORK

We have presented algorithms to compute affine equivalences, similarities and symmetries between special types of curves and surfaces. For surfaces, we have focused on rational ruled surfaces, and we have presented an algorithm for computing affine equivalences between two such surfaces. For curves, we have considered affine equivalences between parametric curves whose components are truncated Fourier series (called trigonometric curves), and similarities between parametric curves which are parametrized by non-necessarily rational functions using the notions of center of gravity and intertia tensor. All the algorithms have been tested in Maple or MATLAB, and show a good performance.

A first question is whether our ideas can be extended to projective equivalences. In the case of ruled surfaces, this does not seem likely. In the case of trigonometric curves, it is not clear yet whether two trigonometric curves (not conics) which are not affinely equivalent may be projectively equivalent: we could not find an example of this situation, but we could not find a proof of impossibility, either, so this is an open question. Also, extending the approach leaning on centers of gravity and inertia tensor to the case of affine or projective equivalences does not seem possible since tensors do not behave well for these kinds of transformations.

A second question is concerned with floating-point inputs, and in general approximate computations. In the algorithm of the approximate case presented in Chapter 3, we suggest using ϵ-gcds to compute approximate affine equivalences. However, the question of relating ϵ with a bound of the precision of the input is still pending. The problem here is that it is difficult to trace the effect of the error in the coefficients of the input on the polynomials whose approximate gcd must be found. Furthermore, even if we could solve that problem, there is still a second problem, namely evaluating whether the obtained affine equivalence is really "good" or not, i.e., to check whether the image of the first curve under the computed affine equivalence is close to the second curve. These problems arise again in the algorithm of Chapter 4 since ϵ-gcds and closeness evaluation are also needed here. In Chapter 4, we provide a criterion to choose ϵ when the only source of inaccuracy is the numeric evaluation of integrals. However, in the case of approximate inputs, it is also difficult to relate the precision of the input and the value of ϵ.

These open questions and difficulties suggest potential lines of research, that we enumerate below.

(1) *Projective and affine equivalences of rational surfaces:* Although there has been recently uploaded an ArXiv paper that addresses this question with great generality (see [63]), it is not clear whether the suggested approach is efficient. It would be interesting to look for specific algorithms for special surfaces that could exploit the properties of the surface. Some particular types of surfaces that could be addressed are *Steiner surfaces*, *translational surfaces* (affine equivalences have been attacked in [16], but projective transformations are still open), *affine rotation surfaces*, which are generalizations of the well-known surfaces of revolution (see for instance [8]), or *monoid surfaces.*

(2) *Approximate equivalences:* the general problem of computing approximate equivalences between curves defined up to a certain precision is almost absent in the literature. The only contributions so far are those of this thesis, and the paper [34], where approximate symmetries of planar implicit algebraic curves are considered. For surfaces the problem is completely open. These questions are

really challenging, and the difficulty has to do with the fact that Algebra and Geometry do not always agree: even if Algebra suggests that two objects are close, this might be false. Take for instance the curves $f_1(x, y) = x^2 - 1 = 0$ and $f_2(x, y) = x^2 + 0.01y^2 - 1 = 0$; the implicit equations of both objects are pretty close, however, the first equation represents the lines $x = -1$, $x = 1$, while the second equation represents an ellipse centered at the origin whose major and minor axes are $a = 1$ and $b = 10$. Thus, there are enormous differences between both objects from the geometric point of view.

Another example of the problem is the notion of ϵ-point, see for instance [85, 86]. Essentially, an ϵ-point of a variety of codimension 1 (see [85, 86] for a formal definition) is a point p that yields a number very close to zero when the polynomial implicitly defining the variety is evaluated at p. Intuition suggests that this means that the point is very close to the variety, but this is not always true. Somehow, the key question is to distinguish between ill-conditioned and well-conditioned problems: for instance, the case of $f_1(x, y) = x^2 - 1 = 0$ and $f_2(x, y) = x^2 + 0.01y^2 - 1 = 0$ seems to be ill-conditioned, and however, $f_1(x, y) = 1.01x^2 + 0.01y^2 - 1 = 0$ and $f_2(x, y) = x^2 + 0.01y^2 - 1 = 0$ seem to be well-conditioned.

(3) *Measuring distances:* the problem of evaluating how close two objects are has to do with determining the Hausdorff distance between these two objects (see Subsection 3.4.1 in Chapter 3), but this computation is not easy. Although there have been several contributions to this question (see [50, 67, 93] and the references in this last paper), there seems to be still space for improvement. On the one hand, existing algorithms are for algebraic curves or surfaces, and it would be nice to address more general objects. On the other hand, in general, the computation is challenging, and it would be interesting to focus on bounds for the Hausdorff distance that could be more easily derived.

Additionally, one can also consider the more general problem of determining the minimal distance between two algebraic objects. Again, the brute-approach to this problem, especially for implicit varieties (e.g., using Lagrange's multipliers), is time-consuming, and it would be desirable to have efficient methods for deriving upper and lower bounds for the distance.

Furthermore, although this thesis mainly focuses on certain parametric varieties, the computation of equivalences and symmetries for implicit curves and surfaces is still very open and suggests other potential research lines. Thus, we add a fourth item to the previous list:

(4) *Implicit case:* for curves, exact symmetries and similarities can be considered as solved since the algorithm in [5] is very efficient. Affine equivalences are considered in [33], but projective equivalences are not. For algebraic space curves, no contributions, up to our knowledge, have been made yet. A possibility to explore in this case, at least for symmetries and affine equivalences, is that the points at infinity of the curves should be mapped to each other; this opens up the possibility of using the techniques in [33] for finite collections of points.

    For algebraic surfaces, the fact that the surfaces defined by the highest degree forms must be mapped to each other could also be a starting point. In fact, these surfaces are conic, so any advance on the solution of the problem for conic surfaces could be used in the general case. Other interesting observation is that the highest degree form is a homogeneous polynomial, and there is a one-to-one correspondence between homogeneous polynomials, and tensors (see Section 2.2 of [6]). In particular, any homogeneous polynomial of degree N corresponds to a tensor of order N. Thus, recognizing whether two surfaces defined by homogeneous polynomials are similar, is equivalent to recognizing a same tensor, written in two different bases. This problem is very well understood for tensors of order two, i.e. matrices, and it amounts to, for instance, diagonalizing. However, the same problem for high order tensors is much more difficult, and apparently still open. Therefore, the approach from multilinear algebra might also be interesting.

    Moreover, certain particular types of implicit surfaces with additional properties could also be investigated. For instance, it is well-known that cubic surfaces contain a certain number of lines, which should be mapped to each other by any affine or projective equivalence. Affine rotation surfaces also have a specific structure in their implicit equation that could be exploited. Furthermore, of course, approximate problems are entirely open as well.

All these potential lines of research are certainly attractive, and might be pursued in the future.

BIBLIOGRAPHY

[1] Alcázar J.G. (2012), *Computing the shapes arising in a family of space rational curves depending on one parameter.* Comput. Aided Geom. Des. Vol. 29, pp. 315–331.

[2] Alcázar J. G. (2014), *Efficient detection of symmetries of polynomially parametrized curves,* Journal of Computational and Applied Mathematics Vol. 255, pp. 715–724.

[3] Alcázar J.G., Dahl H., Muntingh G. (2018), *Symmetries of canal surfaces and Dupin cyclides*, Computer Aided Geometric Design Vol. 59, pp. 68–85.

[4] Alcázar, J. G., Díaz-Toca, G. M., (2010), *Topology of 2D and 3D rational curves,* Comput. Aided Geom. Des. 27 (7), pp. 483–502.

[5] Alcázar J.G., Díaz Toca G.M., Hermoso C. (2019), *The problem of detecting when two implicit plane algebraic curves are similar*, International Journal of Algebra and Computation Vol. 29, No. 05, pp. 775–793.

[6] Alcázar J.G., Goldman R. (2016), *Finding the axis of revolution of an algebraic surface of revolution*, IEEE Transactions on Visualization and Computer Graphics Vol. 22, Issue 9, pp. 2082-2093.

[7] Alcázar J.G., Goldman R. (2017), *Detecting when an implicit equation or a rational parametrization defines a conical or cylindrical surface, or a surface of revolution*, IEEE Transactions on Visualization and Computer Graphics Vol. 23, Issue 12, pp. 2550–59.

[8] Alcázar J.G., Goldman R. (2020), *Recognizing algebraic affine rotation surfaces*, Computer Aided Geometric Design Vol. 81, 101905.

[9] Alcázar J.G., Hermoso C. (2016), *Involutions of polynomially parametrized surfaces*, Journal of Computational and Applied Mathematics Vol. 294, pp. 23–38.

[10] Alcázar J.G., Hermoso C. (2018), *Projective equivalences of elliptic and hyperelliptic planar curves.* PrePrint.

[11] Alcázar J.G., Hermoso C., Muntingh G. (2014), *Detecting similarity of rational plane curves*, Journal of Computational and Applied Mathematics vol. 269, pp. 1–13.

[12] Alcázar J.G., Hermoso C., Muntingh G. (2014), *Detecting symmetries of rational plane and space Curves*, Computer Aided Geometric Design vol. 31, pp. 199-209.

[13] Alcazár J.G., Hermoso C., Muntingh G. (2015), *Symmetry detection of rational space curves from their curvature and torsion*, Computer Aided Geometric Design Vol. 33, pp. 51–65.

[14] Alcázar J. G., Hermoso C., Muntingh G. (2017), *Similarity detection of rational space curves,* ArXiv 1512.02620v5.

[15] Alcázar J. G., Lávička M., Vršek J. (2019), *Symmetries and similarities of planar algebraic curves using harmonic polynomials,* Journal of Computational and Applied Mathematics Vol. 357, pp. 302–318.

[16] Alcázar J.G., Muntingh G. (2021), *Affine equivalences of rational surfaces of translastion, and applications to rational minimal surfaces*, ArXiv: 2103.00151.

[17] Alcázar J.G., Quintero E. (2018), *Computing Symmetries of Ruled Rational Surfaces*, Actas de los Encuentros de Álgebra Computacional y Aplicaciones 2018.

Monografías de la Real Academia de Ciencias. Zaragoza. Vol. 43, pp. 35–38. ISSN: 1132-6360.

[18] Alcázar J.G., Quintero E. (2020), *Affine equivalences, isometries and symmetries of ruled rational surfaces,* Journal of Computational and Applied Mathematics Vol. 364, 112339.

[19] Alcázar J.G., Quintero E. (2020), *Affine Equivalences of Trigonometric Curves,* Acta Appl Math Vol. 170, pp. 691–708.

[20] Alcázar J.G., Quintero E. (2020), *Exact and approximate similarities of non-necessarily rational planar, parametrized curves, using centers of gravity and inertia tensors,* International Journal of Algebra and Computation, to appear.

[21] Alonso C., Gutiérrez J., Recio T. (1995), *A Rational Function decomposition Algorithm by Near-separated Polynomials*, Journal of Symbolic Computation Vol. 19, pp. 527–544.

[22] Alt H., Scharf L. (2004), *Computing the Hausdorff distance between sets of curves*, Proceedings of the 20th European Workshop on Computational Geometry (EWCG), Seville, Spain, pp. 233–236.

[23] Alt H., Mehlhorn K., Wagener H., Welzl E. (1988), *Congruence, similarity and symmetries of geometric objects*, Discrete Comput. Geom. Vol. 3, pp. 237–256.

[24] Andre K. T. Assis (2010), *Archimedes, the Center of Gravity, and the First Law of Mechanics*, Apeiron Montreal, second edition.

[25] Banchoff Thomas F., Lovett Stephen T., (2016), *Differential Geometry of Curves and Surfaces*, Taylor and Francis Group, LLC, second edition.

[26] Bardet M., Faugère J.C., Salvy B. (2015), *On the Complexity of the F5 Gröbner basis Algorithm*, Journal of Symbolic Computation Vol. 70, p. 49–70.

[27] Basu S., Pollack R., Roy M.F. (2003), *Algorithms in real algebraic geometry*, Springer.

[28] Batselier K., Dreesen P., De Moor B. (2013), *A geometrical approach to finding multivariate approximate LCMs and GCDs*, Linear Algebra and its Applications, Vol. 438, Issue 9, pp. 3618–3628.

[29] Becker R., Sagraloff M., Sharma V., Yap C. (2018), *A near-optimal subdivision algorithm for complex root isolation based on the Pellet test and Newton iteration*, Journal of Symbolic Computation Vol. 86, pp. 51–96.

[30] Beckermann B., Labahn G. (1998), *When are two numerical polynomials relative prime?*, J. Symbolic Computation Vol. 26, pp. 677–689.

[31] Beckermann B., Labahn G. (1998), *A Fast and Numerically Stable Euclidean-like Algorithm for Detecting Relatively Prime Numerical Polynomials*, J. Symbolic Computation Vol. 26, pp. 691–714.

[32] Berner A., Bokeloh M., Wand M., Schilling A., Seidel H.P. (2008), *A graph-based approach to symmetry detection*, Symposium on Volume and Point-Based Graphics, pp. 1–8.

[33] Bizzarri M., Lávička M., Vršek J. (2020), *Computing projective equivalences of special algebraic varieties*, Journal of Computational and Applied Mathematics Vol. 367, 112438.

[34] Bizzarri M., Lávička M., Vršek J. (2020), *Approximate symmetries of planar algebraic curves with inexact input*, Computer Aided Geometric Design Vol. 76, 101794.

[35] Bokeloh M., Berner A., Wand M., Seidel H.P., Schilling A. (2009), *Symmetry detection using line features*, Comput. Graph. Forum Vol. 28, No. 2, pp. 697–706.

[36] Boutin M. (2000), *Numerically Invariant Signature Curves*, International Journal of Computer Vision Vol. 40, No. 3, pp. 235–248.

[37] Brand C., Sagraloff M. (2016), *On the Complexity of Solving Zero-Dimensional Polynomial Systems via Projection*, Proceedings ISSAC 2016, pp. 151–158.

[38] Brass P., Knauer C. (2004), *Testing congruence and symmetry for general 3-dimensional objects*, Comput. Geom. Vol. 27, pp. 3–11.

[39] Brown J., Churchill R. (1996), *Complex variables and applications*, Mc Graw Hill.

[40] Calabi E., Olver P.J., Shakiban C., Tannenbaum A., Haker S. (1998), *Differential and numerically invariant signature curves applied to object recognition*, International Journal of Computer Vision Vol. 26, No. 2, pp. 107–135.

[41] Carmichael R.D. (1910), *On r-fold symmetry of plane algebraic curves*, Amer. Math. Monthly Vol. 17, No. 3, pp. 56–64.

[42] Chen F., Zheng J., Sederberg T.W. (2001), *The μ-basis of a rational ruled surface*, Computer Aided Geometric Design Vol. 18, pp. 61–72.

[43] Chen F., Wang W. (2003), *Revisiting the μ-basis of a rational ruled surface*, Journal of Symbolic Computation Vol. 36, pp. 699–716.

[44] Cox D. A., Little J., O'Shea D. (2015), *Ideals, Varieties and Algorithms (4^{th} edition)*, Springer.

[45] Coxeter H. S. M. (1969), *Introduction to Geometry (2^{nd} edition)*, John Wiley & Sons Inc.

[46] Crimmins T. (1982), *A complete set of Fourier descriptors for two-dimensional shapes*, IEE Transactions on Systems, Man and Cybernetics Vol. 12, pp. 848–855.

[47] Dalitz C., Brandt C., Goebbels S., Kolanus D. (2013), *Fourier descriptors for broken shapes*, Journal on Advances in Signal Processing Vol. 161.

[48] Davidovits P. (2013), *Appendix A - Basic Concepts in Mechanics*, Physics in Biology and Medicine, Fourth Edition, pp. 275–287.

[49] Díaz-Toca G.M., González-Vega L. (2006), *Computing greatest common di-visors and squarefree decompositions through matrix methods: the parametric and approximate cases*, Linear Algebra and its Applications Vol. 412, pp. 222–246.

[50] Elber G., Grandine T. (2008), *Hausdorff and Minimal Distances between Parametric Freeforms in $\mathbb{R}^2$ and $\mathbb{R}^3$*. In Geometric Modeling and Processing, GMP 2008. Lecture Notes in Computer Science, vol. 4975. Springer, Berlin, Heidelberg, pp 191–204.

[51] Feng S., Kogan I., Krim H. (2010), *Classification of Curves in 2D and 3D via Affine Integral Signatures*, Acta Applicandae Mathematicae Vol. 109, pp. 903–937.

[52] Gal R., Cohen-Or D. (2006), *Salient geometric features for partial shape matching and similarity*, ACM Trans. Graph. Vol. 25, No. 1, pp. 130–150.

[53] Ghorbel F., Derrode S., Mezhoud R., Bannour T., Dhahbi S. (2006), *Image reconstruction from a complete set of similarity invariants extracted from complex moments*, Pattern Recognition Letters Vol. 27, pp. 1361–1369.

[54] Goldstein H. (1992), *Classical mechanics*, 2nd Edition, Addison-Wesley.

[55] Gray A., Abbena E., Salamon S. (2006), *Modern Differential Geometry of Curves and Surfaces with Mathematica*, Chapman and Hall/CRC.

[56] Hauer M., Jüttler B. (2018), *Projective and affine symmetries and equivalences of rational curves in arbitrary dimension*, Journal of Symbolic Computation Vol. 87, pp. 68–86.

[57] Hauer M., Jüttler B., Schicho J. (2018), *Projective and affine symmetries and equivalences of rational and polynomial surfaces*, Journal of Computational and Applied Mathematics Vol. 349, pp. 424–437.

[58] Hess F. (2004). *An algorithm for computing isomorphisms of algebraic function fields*. Algorithmic Number Theory. Vol. 3076 of LNCS. Springer, Berlin, pp. 263–271.

[59] Hilbert D., Cohn-Vossen S. (1990), *Geometry and the imagination*, Chelsea Publishing Company, New York (USA).

[60] Hong H. (1995), *Implicitization of curves parametrized by generalized trigonometric polynomials*, Proceedings of Applied Algebra, Algebraic Algorithms and Error Correcting Codes (AAECC-11), pp. 285–296.

[61] Hong H., Schicho J. (1998), *Algorithms for Trigonometric Curves (Simplification, Implicitization, Parameterization)*, Journal of Symbolic Computation Vol. 26, Issue 3, pp. 279–300.

[62] Huang Z., Cohen F.S. (1996), *Affine-invariant B-spline moments for curve matching*, IEEE Transactions on Image Processing Vol. 10, No. 5, pp. 1473–1480.

[63] Jüttler B., Lubbes N., Schicho J. (2020), *Projective isomorphisms between rational surfaces*, ArXiv 2010.08393v1.

[64] Jiang X., Yu K., Bunke H. (1996), *Detection of rotational and involutional symmetries and congruity of polyhedra*, Vis. Comput. Vol. 12, No. 4, pp. 193–201.

[65] Karmarkar N., Lakshman Y. N. (1998), *On Approximate GCDs of UnivariatePolynomials*, Journal of Symbolic Computation Vol. 26 (6), pp. 653–666.

[66] Kibble T., Berkshire F. (2004), *Classical Mechanics*, Imperial College Press.

[67] Kim Y. J., Oh Y. T., Yoon S. H., Kim M. S, Elber G. (2013) *Efficient Hausdorff distance computation for freeform geometric models in close proximity*, Computer Aided Design Vol. 45, pp. 270–276.

[68] Nagasaka K. (2020), *Toward the best algorithm for approximate GCD of univariate polynomials*, Journal of Symbolic Computation Vol. 105, pp. 4–27.

[69] Lebmair P. (2009), *Feature detection for real plane algebraic curves*, Ph.D. Thesis, Technische Universität München.

[70] Lebmair P., Richter-Gebert J. (2009), *Rotations, translations and symmetry detection for complexified curves*, Computer Aided Geometric Design Vol. 25 pp. 707–719.

[71] Lei Z., Tasdizen T., Cooper D.B. (1998), *PIMs and invariant parts for shape recognition*, Proceedings Sixth International Conference on Computer Vision, pp. 827–832.

[72] Li M., Langbein F., Martin R. (2008), *Detecting approximate symmetries of discrete point subsets*, Comput.-Aided Des. Vol. 40, No.1, pp. 76–93.

[73] Li M., Langbein F., Martin R. (2010), *Detecting design intent in approximate CAD models using symmetry*, Comput. Aided Des. Vol. 42, No. 3, pp. 183–201.

[74] Lipman Y., Cheng X., Daubechies I., Funkhouser T. (2010). *Symmetry factored embedding and distance*. Proceedings of ACM SIGGRAPH. ACM Trans. Graph. Vol. 29, No. 4.

[75] Loy G., Eklundh J. (2006). *Detecting symmetry and symmetric constellations of features*. Proceedings ECCV, 9th European Conference on Computer Vision, pp. 508–521.

[76] Martinet A., Soler C., Holzschuch N., Sillion F. (2006). *Accurate detection of symmetries in 3D shapes*. ACM Trans. Graph. Vol. 25, No. 2, pp. 439–464.

[77] Mishra R., Pradeep K. (2012), *Clustering web logs using similarity upper approximation with different similarity measures*, Int. J. Mach. Learn. Comput. Vol. 2, No. 3, pp. 219–221.

[78] Mitra N.J., Guibas L.J., Pauly M. (2006), *Partial and approximate symmetry detection for 3D geometry*, ACM Trans. Graph. Vol. 25, No. 3, pp. 560–568.

[79] Mozo-Fernández J., Munuera C. (2002), *Recognition of polynomial plane curves under affine transformations*, Appl. Algebra Eng. Commun. Comput. Vol. 13, pp. 121–136.

[80] Nejhum S., Chi Y-T., Ho J., Yang M-H. (2011), *Higher-Dimensional Affine Registration and Vision Applications*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 33, No. 7.

[81] Noda M.T., Sasaki T. (1991), *Approximate GCD and its application to ill-conditioned equations*, Journal of Computational and Applied Mathematics Vol. 38, pp. 335–351.

[82] Pan, V. (2001), *Numerical computation of a polynomial gcd and extensions*, Information and computation Vol. 167, pp. 71-85.

[83] Pérez-Díaz S. (2013) *A partial solution to the problem of proper reparametrization-for rational surfaces*, Computer Aided Geometric Design Vol. 30, pp. 743–759.

[84] Pérez-Díaz S. (2019) *Recent Advances on the Problem of ProperReparametrization for Rational Surfaces*, Advances in Mathematics Research. Nova Science Publishers (Albert R. Baswell. Editor). Vol 25. pp. 203–225.

[85] Pérez-Díaz S., Sendra R., Sendra J. (2004), *Distance properties of epsilon points on algebraic curves*, Series Mathematics and Visuatization, Springer-Verlag, Computational Methods for Algebraic Spline Surfaces. pp.45–61.

[86] Pérez-Díaz S., Sendra R., Sendra J. (2006), *Distance bounds of ϵ-points on hypersurfaces*, Theoretical Computer Science. Vol 359. N. 1-3. pp. 344–368.

[87] Pérez-Díaz S., Shen L-Y. (2014), *Characterization of Rational Ruled Surfaces*, Journal of Symbolic Computation, Vol. 63, pp. 21–45.

[88] Podolak J., Shilane P., Golovinskiy A., Rusinkiewicz S., Funkhouser T. (2006), *A planar-reflective symmetry transform for 3D shapes*, Proceeding SIGGRAPH, pp. 549–559.

[89] Pottmann H., Wallner J. (2001), *Computational Line Geometry*, Springer-Verlag.

[90] Rahtu E., Salo M., Heikkilä J. (2005), *Affine invariant pattern recognition using Multiscale Autoconvolution*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 27, No. 6, pp. 908–918.

[91] Rida T. Farouki, (2008), *Pythagorean-Hodograph Curves: Algebra and Geometry Inseparable*, Springer.

[92] Rothganger F., Lazebnik S., Schmid C., Ponce J. (2006), *3D Object Modeling and Recognition Using Local Affine-Invariant Image Descriptors and Multi-View Spatial Constraints*, International Journal of Computer Vision Vol. 66, pp. 231–259.

[93] Rueda S.L., Sendra J., Sendra R. (2014), *Bounding and estimating the Hausdorff distance between real space algebraic curves*, Computer Aided Geometric Design Vol. 31, Issues 3-4, pp. 182–198.

[94] Sánchez-Reyes J. (2015), *Detecting symmetries in polynomial Bèzier curves*, Journal of Computational and Applied Mathematics, Vol. 288, pp. 274–283.

[95] Schnabel R., Wessel R., Wahl R., Klein R. (2008). *Shape recognition in 3D-point clouds*. The 16-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision..

[96] Sendra R., Sevilla D., Villarino C. (2017), *Covering Rational Ruled Surfaces*, Mathematics of Computation Vol. 86, No. 308, pp. 2861-2875.

[97] Sendra J.R., Winkler F., Pérez-Díaz S. (2008), *Rational Algebraic Curves*, Springer-Verlag.

[98] Sener S., Unel M. (2005), *Affine invariant fitting of algebraic curves using Fourier descriptors*, Pattern Analysis and Applications Vol. 8, pp. 72–83.

[99] Shahed Nejhum S.M., Chi Y-T, Ho J., Yang M-H (2011), *Higher-Dimensional Affine Registration and Vision Applications*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 7.

[100] Simari P., Kalogerakis E., Singh K. (2006). *Folding meshes: hierarchical mesh segmentation based on planar symmetry*. Proc. Symp. Geometry Processing, pp. 111–119.

[101] Simmonds J.G. (1994), *A brief on tensor analysis*, Undergraduate texts in Mathematics, Springer-Verlag.

[102] Stewart G. W. (2006), *Perturbation of the SVD in the presence of small singular values*, Linear Algebra and its Applications Vol. 419. pp. 53–77.

[103] Suk T., Flusser J. (1993), *Pattern recognition by affine moment invariants*, Pattern Recognition Vol. 26, No. 1, pp. 167–174.

[104] Suk T., Flusser J. (2005), *Affine normalization of symmetric objects*, Proceedings ACIVS 2005, Lecture Notes in Computer Science, pp. 100–107.

[105] Sun C., Sherrah J. (1997), *3D symmetry detection using the extended Gaussian image*, IEEE Trans. Pattern Anal. Mach. Intell. Vol. 19, pp 164–168.

[106] Synge J. L., Schild A. (1949), *Tensor Calculus*, University of Toronto Press, pp 127–136.

[107] Tarel J.P., Cooper D.B. (2000), *The complex representation of algebraic curves and its simple exploitation for pose estimation and invariant recognition*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 22, No. 7, pp. 663–674.

[108] Tasdizen T., Tarel J.P., Cooper D.B. (2000), *Improving the stability of algebraic curves for applications*, IEEE Transactions on Image Processing Vol. 9, pp. 405–416.

[109] Tate S., Jared G. (2003). *Recognising symmetry in solid models*. Comput. Aided Des. Vol. 35, No. 7, pp. 673–692.

[110] Taubin G. (1991), *Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations, with applications to edge and range image segmentation*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 13, No. 11, pp. 1115–1138.

[111] Taubin G., Cooper D.B. (1992), *Object recognition based on moments (or algebraic) invariants*, J.L. Mundy, A. Zisserman (Eds.), Geometric Invariance in Computer Vision, MIT Press, pp. 375–397.

[112] Unel M., Wolowich W.A. (2000), *On the construction of complete sets of geometric invariants for algebraic curves*, Adv. in Appl. Math. Vol. 24. pp. 65–87.

[113] Von Zur Gathen J., Gerhard J. (1999), *Modern Computer Algebra*, Cambridge University Press, Cambridge.

[114] Vršek J., Lávička L. (2015), *Determining surfaces of revolution from their implicit equations*, J. Comput. Appl. Math. 290. pp. 125–135.

[115] Weiss I. (1993), *Noise-resistant invariants of curves*, IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 15, No. 9, pp 943–948.

[116] Wolowich W., Unel M. (1998), *The determination of implicit polynomial canonical curves*, IEEE Trans. Pattern Anal. Mach. Intell. Vol. 20, No. 10, pp. 1080–1089.

[117] Wolowich W., Unel M. (1998), *Vision-based system identification and state estimation*, The Confluence of Vision and Control, Lecture Notes in Control and Information Systems, Springer-Verlag, New York.

[118] Xiao S. G., Shang C. C. (1992), *On the Parameterization of Algebraic Curves*, Journal of Applicable Algebra in Engineering, Communication and Computing, 3, pp. 27-38.

[119] Yalcin H., Unel M., Wolowich W. (2003), *Implicitization of Parametric Curves by Matrix Annihilation*, International Journal of Computer Vision 54, pp. 105-115.

[120] Yang Z., Cohen F. (1999), *Image Registration and Object Recognition Using Affine Invariants and Convex Hulls*, IEEE Transactions on Image Processing Vol. 8, No. 7, pp. 934–946.

[121] Zeng Z. (2004), *The approximate gcd of inexact polynomials. Part I: a univariate algorithm*, Proceedings of the 2004 International Symposium on Symbolic and Algebraic Computation (ISSAC), ACM, (2004), pp. 320–327.

[122] Zeng, Z. (2011), *The Numerical Greatest Common Divisor of Univariate Polynomials*. Contemporary Mathematics Vol. 556, pp. 187–217.

[123] Zeng, Z. (2012), http://homepages.neiu.edu/~zzeng/naclab.html

List of Algorithms

Description:
Input: Two ruled surfaces S_1, S_2, properly parametrized by $\boldsymbol{x}_i(t,s) = \boldsymbol{p}_i(t) + s\boldsymbol{q}_i(t)$, $i = 1,2$, where each $\boldsymbol{q}_i(t)$ is polynomial with relatively prime components.
Output: The affine equivalences $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ between S_1, S_2.

Description:
Input: Two trigonometric curves $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$, defined by simple parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$.
Output: The affine equivalences $f(\mathbf{x}) = \boldsymbol{A}\mathbf{x} + \boldsymbol{b}$ between $\mathcal{C}, \mathcal{D}$.

Description:
Input: Two trigonometric curves $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^n$, given by approximate parametrizations $\boldsymbol{x}(t), \boldsymbol{y}(t)$, and a tolerance ϵ.
Output: The approximate affine equivalences $\tilde{f}(\mathbf{x}) = \tilde{\boldsymbol{A}}\mathbf{x} + \tilde{\boldsymbol{b}}$ between $\mathcal{C}, \mathcal{D}$.

Description:
Input: Two bounded curves $\mathcal{C}_1, \mathcal{C}_2 \subset \mathbb{R}^2$, not necessarily rational, parametrized by $\boldsymbol{x}(t), \boldsymbol{y}(t)$.
Output: The similarities $f(\mathbf{x}) = \lambda\boldsymbol{Q}\mathbf{x} + \boldsymbol{b}$ (approximate in some cases) between $\mathcal{C}_1, \mathcal{C}_2$.