

Learning, processing and optimising shapes

GRAPES Book

Extended abstracts of PhD Theses
of the GRAPES fellows

Athens - July 15, 2024

Introduction

GRAPES aims at considerably advancing the state of the art in Mathematics, Computer-Aided Design, and Machine Learning in order to promote game changing approaches for generating, optimising, and learning 3D shapes, along with a multisectoral training for young researchers. Recent advances in the above domains have solved numerous tasks concerning multimedia and 2D data. However, automation of 3D geometry processing and analysis lags severely behind, despite their importance in science, technology and everyday life, and the well-understood underlying mathematical principles. The CAD industry, although well established for more than 20 years, urgently requires advanced methods and tools for addressing new challenges.

The scientific goal of GRAPES is to bridge this gap based on a multidisciplinary consortium composed of leaders in their respective fields. Top-notch research is also instrumental in forming the new generation of European scientists and engineers. Their disciplines span the spectrum from Computational Mathematics, Numerical Analysis, and Algorithm Design, up to Geometric Modelling, Shape Optimisation, and Deep Learning. This allows the 15 PhD candidates to follow either a theoretical or an applied track and to gain knowledge from both research and innovation through a nexus of intersectoral secondments and Network-wide workshops.

Horizontally, our results lead to open-source, prototype implementations, software integrated into commercial libraries as well as open benchmark datasets. These are indispensable for dissemination and training but also to promote innovation and technology transfer. Innovation relies on the active participation of SMEs, either as a beneficiary hosting an ESR or as associate partners hosting secondments. Concrete applications include simulation and fabrication, hydrodynamics and marine design, manufacturing and 3D printing, retrieval and mining, reconstruction and visualisation, urban planning and autonomous driving.

GRAPES trained 19 ESR fellows; this book collects the extended abstracts of the PhD Theses of the fellows that have completed their doctoral studies by July 2024. The complete theses will be available upon completion at <http://grapes-network.eu/index.php/publications/>.

Project facts

Duration: 54 Months (1/12/2019 — 31/05/2024)

Project Coordinator: Ioannis Z. Emiris (emiris AT athenarc.gr)

Research Coordinator: Laurent Busé (Laurent.Buse AT inria.fr)

Technical Coordinator: Christos Konaxis (ckonaxis AT athenarc.gr)

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 860843



Marie Skłodowska-Curie
Actions

Contents

1	Carles Checa: Multihomogeneous and sparse polynomial systems: resultants, Groebner bases and regularity	5
2	Eriola Hoxhaj: Reconstruction of Surfaces and Planar Maps from their Branching Curves and Planar Sections	11
3	Shahroz Khan: IDEAS@MI: Intelligent Data-driven systEms for digitAl deSign in Maritime Industry	27
4	Andriamahenina Ramanantoanina: Barycentric rational interpolation	33
5	Krunal Raval: Non-uniform degree and Tchebycheffian spline technologies for adaptive isogeometric analysis	45
6	Jean Michel Menjanahary: Dupin Cyclidic Splines of Arbitrary Topology	55
7	Rao FU: Shape Reconstruction from 3D Point Clouds	61

Chapter 1

Carles Checa: Multihomogeneous and sparse polynomial systems: resultants, Groebner bases and regularity

1.1 Algorithms in algebraic elimination

Solving polynomial systems is one of the oldest and most ubiquitous problems arising computational science and engineering. These problems are inherently hard: their complexity will grow exponentially in the number of variables. Thus, we cannot just tell the computer "solve this" and expect it to respond in reasonable time. We ought to look at the structure coming from those polynomials and see if we manage to find better algorithms that take advantage of the properties of each family of systems. Let's see the particularities of systems encountered within certain applications.



Figure 1.1: Implicit surface of a map given by three bilinear forms.

- Computer vision: A variety of polynomial systems arising in vision consists of matching problems between snapshots captured by cameras (see Fig. 1.2). Thousands of polynomial systems will have to be simultaneously solved [12, 13] so small differences in the computations will be helpful in the final result. As one thinks of the cameras as linear projections,

interesting algebraic objects with multi-homogeneous or sparse structure such as Chow forms [14] and distortion varieties [15] arise.

- Geometric modelling: On the one hand, transforming polynomial maps into implicit equations is a fundamental problem of computer-aided design (see Fig. 1.1). These maps may have a natural multi-homogeneous or toric structure [16]. One might also want to compute directly with the parametric form and study its topology [17]. On the other hand, Bezier surfaces play a central role in the manipulation of the algebraic objects and they can also be seen from the perspective of toric geometry [18, 19].
- Biology: In the study of chemical reaction networks, systems depending on parameters appear. The study of these parameters is helpful to decide a property called multistationarity. Another fundamental problem is the study of the generic dimension depending on those parameters [20, 21]. These systems also exhibit some toric structure [22].

Other applications in which there's a lot of structure to be exploited include kinematics [23], cryptography [24], algebraic statistics [25, 26], coding theory [27] or data analysis [28]. But, how can we describe this structure?

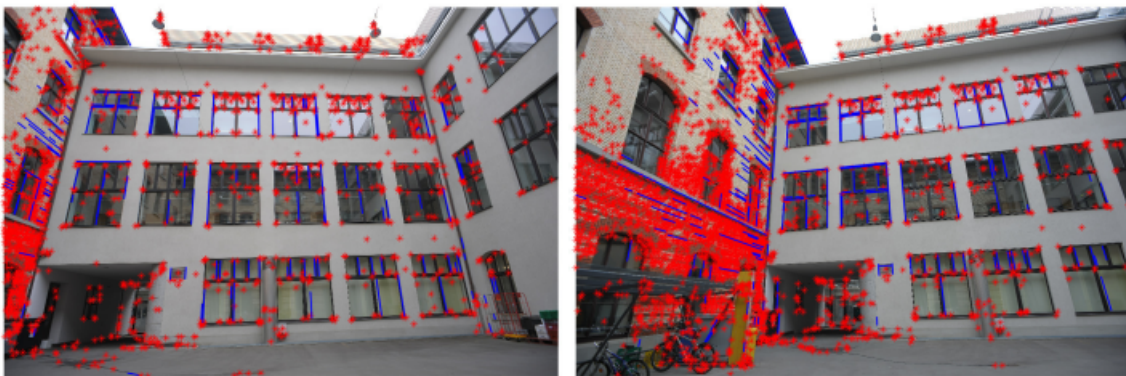


Figure 1.2: Minimal problems in vision [1].

- Geometric structure: Perhaps the solution set has some symmetry, we know one solution that we can exploit to get the rest of them, or the polynomials come from a given geometric predicate (cylinders, cones...).
- Algebraic structure: Polynomial systems can naturally be seen from the point of view of commutative algebra. Therefore, the constructions that one attaches to a polynomial ideal are part of the structure to be exploited (minimal free resolutions, saturations, complete intersections...).

Without losing sight on the first one, let me focus on the second type of structures. A common way of using algebraic structure is by studying the Newton polytopes. Namely, we focus on which terms in the polynomials of the system can have non-zero coefficients. This type of structure also relates to asking in which space we are looking for the solutions: we would like that small perturbations on the solutions on the systems do not change too much the solutions, thus we usually prefer to work in compact spaces. The theory of toric varieties provides a very natural choice for this space given its relation with the Newton polytopes structure (see Fig 1.3). One of the most universal ways of noting the importance of the sparsity in polynomial systems is the theorem of Bernstein-Khovanskii-Kushnirenko that counts the number of solutions in terms of the mixed volume of the polytopes.




Type of systems	Newton polytope	Polynomials	Underlying compact space
Dense polynomial systems		$1 + x + x^3 + y + yx^2 + y^2 + y^3$	\mathbb{P}^n
Multihomogeneous polynomial systems		$1 + x + y + xy + y^2 + y^2x$	$\mathbb{P}^{n_1} \times \dots \times \mathbb{P}^{n_r}$
Sparse polynomial systems		$1 + x + y + xy + y^2$	A toric variety

Figure 1.3: Systems with given Newton polytope arise in applications.

Once we established the setting of which kind of structure we want to exploit, we ask ourselves which type of methods or algorithms of computation fit into the exploitation of the underlying algebra. A big family of methods called "symbolic-numeric" is usually considered. This type of methods combine the fact that the coefficients of the systems are given in a numerical form (here one deals with precision errors, condition stability, etc...) and the fact that one can consider the coefficients as symbols and provide the solutions as algebraic manipulation of these symbols (here one deals with linearization, finding bases of algebraic spaces, etc...). At this point, the objects that I explored during my PhD are welcome in the room.

1.2 Resultants and elimination matrices

In the previous Newton polytope context, let's assume that the polynomials we are studying are given in the following form:

$$F_i = \sum_{a \in \mathcal{A}_i} c_{i,a} x^a \quad i = 1, \dots, l \quad (1.1)$$

where the \mathcal{A}_i are the sets of supports in a lattice. When we have $n + 1$ polynomials and n variables, we will usually have no solutions: the sparse resultant is a polynomial in the coefficients $c_{i,a}$ whose vanishing locus indicates precisely for which values of $c_{i,a}$ we get systems of polynomials that have a solution. The formulas for computing the resultant usually appear by taking the ratio of two determinants.

These formulas are not only beautiful but also practical: the matrices appearing in resultant formulas can be used for solving the polynomial systems either by posing an eigenvalue problem, hiding a variable or using different methods that exploit linear algebra [29]. If we reach an exact determinantal formula, we also avoid many practical problems. Note also that we are eliminating all the variables associated to the system, thus the resultant can fit in the context described by the applications.

One family of matrices are those whose numerator and denominator arise as minors of the Macaulay matrix:

$$\mathcal{M}_{\mathbf{v}} : (\mathcal{S}^{n+1} \rightarrow \mathcal{S})_{\mathbf{v}} \quad (G_0, \dots, G_n) \rightarrow \sum_{i=0}^n G_i F_i \quad (1.2)$$

where \mathbf{v} is a multi-degree, \mathcal{S} is a suitable \mathbb{Z}^r -graded polynomial ring (the Cox ring [30]) and the G_i are polynomials of multi-degree \mathbf{v} minus the multi-degree of the F_i for $i = 0, \dots, n$. An interesting computational problem is to find \mathbf{v} such that we can recover geometric properties of the system from the rank of $\mathcal{M}_{\mathbf{v}}$ (number of solutions, dimension...).

The Canny-Emiris formula: A possible choice of the minors of $\mathcal{M}_{\mathbf{v}}$ providing the resultant formulas comes from a combinatorial rule given by Canny and Emiris [31] which resembles the classical formula of Macaulay [32]. In this formula, the rows of the minor correspond to lattice points in a translation of the polytope $\Delta = \sum_{i=0}^n \Delta_i$. Providing a mixed subdivision on Δ corresponds to matching the rows and some of the columns of $\mathcal{M}_{\mathbf{v}}$ and giving a maximal minor of this matrix. The mixed subdivision also indicates the rows and columns that form the matrix appearing in the denominator of the formula.

PhD results: In [33], we gave a family of subdivisions for which this formula holds, using the proof in [34]. Moreover, different possible algorithms for dealing with the lattice points may provide even smaller matrices. We considered using a greedy algorithm for the previous family of subdivisions under suitable hypotheses on the Newton polytopes and fully characterized the lattice points labeling the rows and columns of these matrices. In a newer version of the same paper

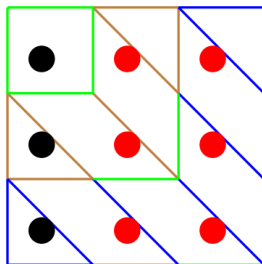


Figure 1.4: A mixed subdivision providing a formula for three bilinear polynomials. Using the greedy algorithm, we only need the **red** lattice points.

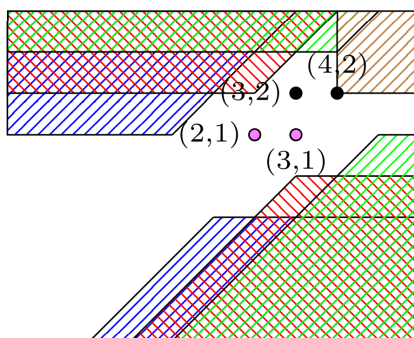


Figure 1.5: The four dots in this table correspond to multi-degrees providing Macaulay matrices for a system of a three homogeneous polynomials in a Hirzebruch surface. In **pink**, you can see the degrees in which the basis of Sylvester forms is used. Some of these matrices provide exact determinantal formulas for the sparse resultant.

[35], we conjectured that which could be the minimal size of the matrices relating it to the Hilbert function and the degree reverse lexicographical monomial order.

Sylvester forms: For dense polynomial systems, it is possible to reduce the size of $\mathcal{M}_{\mathbf{v}}$ to the cost of introducing forms in the saturation $I^{\text{sat}} = (I : \mathfrak{m}^{\infty})$, where I is the ideal generated by the homogenization of the polynomials in 1.1 and \mathfrak{m} is the irrelevant ideal of \mathbb{P}^n [36]. The construction of these forms consists on noticing that under suitable hypotheses on \mathbf{v} , the module $(I^{\text{sat}}/I)_{\mathbf{v}}$ is free and explicitly finding a basis in terms of some elements of I^{sat} known as Sylvester forms. With these, we transform the matrices of 1.2 into:

$$\mathcal{H}_{\mathbf{v}} : (S^{n+1} \oplus I^{\text{sat}}/I \rightarrow S)_{\mathbf{v}}(G_0, \dots, G_n, l_{\mu}) \rightarrow \sum_{i=0}^n G_i F_i + \sum_{\mu \in \mathcal{I}} l_{\mu} \text{sylv}_{\mu} \quad (1.3)$$

where \mathcal{I} labels the basis of $(I^{\text{sat}}/I)_{\mathbf{v}}$.

PhD results: This construction was first extended to the multiprojective case in [37] and

we reproduced it for any smooth projective variety satisfying a certain hypothesis [38]. We also considered the role of Sylvester forms in the construction of toric residues [39].

1.3 Gröbner bases and the Castelnuovo-Mumford regularity

Once a polynomial system is encoded in an ideal I , a natural thing to do is to try to find a set of generators for I with good properties. One of the most established ways to do so are Gröbner bases. Namely, one fixes a monomial order $<$ which allows us to choose leading terms ($\text{in}(f)$) for each polynomial f and aims to find generators $\{g_i\}_{i \in \mathcal{J}}$ of I such that their initial forms generate the initial ideal.

$$\text{in}(I) = (\text{in}(f) \mid f \in I) \text{ is generated by } \{\text{in}(g_i)\}_{i \in \mathcal{J}} \quad (1.1)$$

There are multiple algorithms to compute Gröbner bases (Buchberger, F4, F5...) and they can also be seen from the point of view of exploiting the Newton polytope structure [40]. In this context, more than trying to develop new constructions or techniques, my work focused on the description of the invariants that govern these computations. For instance, we aim to control the degree of the elements in a Gröbner bases in order to understand number of operations involved in their computation. The algebraic invariant that allows us to do this study is the Castelnuovo-Mumford regularity.

From the commutative algebra point of view, the Castelnuovo-Mumford regularity $\text{reg}(I)$ measures the degrees involved in a minimal free resolution of I . A famous paper by Bayer and Stillman [41] established the relation between $\text{reg}(I)$ and the Gröbner basis for dense polynomial systems. Using generic coordinates (i.e. the generic initial ideal $\text{gin}(I)$) and the degree reverse lexicographical order, $\text{reg}(I)$ is the maximal degree of a minimal generator of $\text{gin}(I)$. In particular, they showed that $\text{reg}(I) = \text{reg}(\text{gin}(I))$.

These results imply that the complexity of computing Gröbner bases only depends on the algebro-geometric properties of I . This complexity can be doubly exponential on the number of variables [42], but under suitable assumptions (for example, generic polynomials) it behaves reasonably.

PhD results: An important part of my research was devoted to studying this type of results in the multi-homogeneous setting. We studied the generators of the multi-homogeneous analogue of the generic initial ideal (i.e. the multi-generic initial ideal $\text{mgin}(I)$) and derived a relation with the multi-graded Castelnuovo-Mumford regularity [43]. We compared our results with other type of bounds appearing in the literature [44] and certified the presence of elements of certain multi-degrees. Under suitable assumptions, a part of the picture described by these multi-degrees is described with invariants of only one group of variables.

Chapter 2

Eriola Hoxhaj: Reconstruction of Surfaces and Planar Maps from their Branching Curves and Planar Sections

Abstract:

This study centers on reconstructing particular surfaces and planar maps from a provided planar curve. The planar curve emerges either as a projection of a surface onto the plane or as the planar section of a surface. If the surface is smooth, the projected planar curve is termed the branching curve; otherwise, it consists of the branching curve and the image of the surface's singularities, referred to as the apparent contour. Both the defining equations of the branching curve and the apparent contour are the discriminant of the equation of the surface with respect to the direction of the projection. By resolving the singularities of the projected planar curve, a space curve is obtained, known as the ramification curve or contour—defined by the vanishing locus of the surface's equation and the partial derivative with respect to the direction of the projection. Using straightforward techniques from Linear Algebra, is derived the equation of the surface from the contour/ramification curve. Efforts so far are dedicated to successfully reconstructing various surfaces, including smooth surfaces, those with ordinary singularities, rational surfaces, and Darboux cyclides.

In the realm of rational (parametrizable) surfaces, another approach involves reconstructing a planar map from a provided branching curve. This involves using the linear normalization of a branching curve composed with a parametrization of the surface.

In addition, our analysis extends to the reconstruction of Dupin cyclides, which are surfaces also known as channel or kanal surfaces in two distinct ways: they are the envelope of two families of generating oriented spheres. When an oriented plane intersects a Dupin cyclide, it touches four spheres of the family, paired up on each side of the plane, with at least two of these intersections being real. Here, orientation contact refers to tangency compatible with normals. These points of

contact serve as the focal points for the bicircular quartics that arises as planar sections of the Dupin cyclide. With this understanding, one can reconstruct the family of Dupin cyclides using a provided bicircular quartic as its planar section. The self-duality of Dupin cyclides establishes a connection between the apparent contour and the dual of the planar section, facilitating characterization based on specific planar sections.

2.1 Introduction

Algebraic Geometry is the discipline of Mathematics that solves systems of polynomials. The simple question arises: Why is Algebra associated with Geometry? The first relation between them is the Fundamental Theorem of Algebra, which states that a polynomial of degree d has at most d roots. The notions of polynomials and roots have geometric meanings, and the degree corresponds to Algebra. The vanishing locus of a polynomial defines a hypersurface, specifically a variety of codimension one with degree d . In three-dimensional space, this is referred to as an algebraic surface, and in a two-dimensional plane, it is known as an algebraic curve. Algebraic surfaces have huge applications in Computer Vision and are very helpful in solving many problems. One of the most important tasks for researchers in Computer Vision and Machine Learning is the reconstruction of surfaces from two-dimensional pictures. Knowledge of algebraic surfaces provides many advantages in this study. One advantage is that they are rigid; namely, with a small perturbation, another surface is obtained, which is not isomorphic to the previous one. Additionally, algebraic surfaces have a small number of parameters, making computations easy.

Considering the projection of smooth algebraic surfaces onto a plane, two specific curves are obtained: the ramification curve and the branching curve. The ramification curve is the locus of points on the surface whose the tangent space passes through the center of projection. It is defined as the vanishing locus of the equation of the surface and its partial derivative with respect to the direction of the projection. By projecting the ramification curve, one can obtain the branching curve.

The reconstruction of a surface from the given branching curve depends on resolving the singularities of the branching curve, which leads to the ramification curve. Singularities occur at points where the curve lacks smoothness, and in algebraic computations, they correspond to points where the determinant of the Jacobian matrix equals zero. The entries of the Jacobian matrix represent the first partial derivatives of the curve's equation.

In cases where the surface is not smooth, the projected curve is called the apparent contour, which is the union of the branching curve with the image of the singular locus of the surface, known as the singular image. It is crucial to investigate all the singularities that appear in the apparent contour, including the singularities of the branching curve, the singularities of the singular image, and any common intersections between the branching curve and the singular image. Resolving the

singularities of the apparent curve yields a curve lying on the surface, known as the contour. This contour represents the union of the ramification curve with the singular locus of the surface. It is defined as the vanishing locus of the equation of the surface and its partial derivatives with respect to the direction of the projection.

The final step involves determining the equation of the smooth (or singular) surface by identifying forms that vanish on the ramification curve (or contour). These forms should have the same degree as the surface equation and its partial derivative, and linear algebra techniques are used for this purpose.

In addition to all that, the aim of this research was to find solutions for real problems. However, sometimes it is very difficult to perform computations within our perception of space. Therefore, the right place to conduct Algebraic Geometry is in projective space. The projective space, denoted as \mathbb{P}^n , is the set of $(n + 1)$ tuples of numbers, not all zero, modulo the equivalence relation where a point is equivalent to all its non-zero scalar multiples. Geometrically, \mathbb{P}^n can be understood as the set of rays passing through the origin of an $(n + 1)$ -dimensional space. Drawing a hyperplane (an n -dimensional subspace) at a unit distance from the origin, almost all the rays intersect the hyperplane at points, which serve as representatives of the rays. Projective space extends Euclidean space by adding points at infinity, which arise as the intersection of parallel lines. It does not make sense to speak of a polynomial vanishing at a point in \mathbb{P}^n , as the same polynomial may not vanish for some scalar multiples of that point. However, the notion of vanishing at a point in \mathbb{P}^n is well-defined for homogeneous polynomials, i.e polynomials all of whose monomials have the same degree.

In this context, determining the working space and understanding the properties of the objects are crucial for applying the appropriate tools in the study. One category of surfaces under consideration is Dupin cyclides. To facilitate their examination, Laguerre geometry is employed, the latter is a classical spherical geometry grounded in oriented spheres, oriented hyperplanes, and their oriented contact. Oriented contact refers to tangency with compatible normals.

In particular, the cyclographic model \mathbb{R}^4 of Laguerre geometry is used here. In this model, each sphere is represented as a point (m_1, m_2, m_3, r) in 4-dimensional Minkowski space $\mathbb{M} = \mathbb{R}_{3,1}^4$, where m_1, m_2, m_3 are the coordinates of the center and r is the radius of the sphere. The Euclidean space is identified as the hyperplane $r = 0$.

A Dupin cyclide is the envelope of all spheres that are touching three given spheres. A planar section of a Dupin cyclide yields a curve called a bicircular quartic. The focal points of bicircular quartics are the same as the touching points of the three generating spheres with the plane section. Thus, the entire process of reconstruction depends on the properties of the focal points of the planar section curve, which are defined as the intersection points of the isotropic lines tangent to the curve itself.

While different techniques are employed for the reconstruction of surfaces in this work, there exists a relation among them. Notably, the curve on the planar section of the Dupin cyclide serves as the dual curve of the apparent contour, obtained through the central projection of the Dupin

cyclide, as demonstrated in [45, Theorem 3.1]. This duality arises from the property of the Dupin cyclide being self-dual (see [46]). The following statement connects the reconstruction of Dupin cyclides with the reconstruction of other surfaces examined in this work.

Proposition 2.1.1. Let S be a surface in \mathbb{P}^3 , and let its dual surface be denoted as $S^* \subset \mathbb{P}^{3*}$. The apparent contour obtained from a central projection $S \rightarrow \mathbb{P}^2$ and the curve obtained from a planar section of the dual surface S^* are dual to each other.

Proof. Let $S \subset \mathbb{P}^3$ be a surface and a point $o \in \mathbb{P}^3 \setminus S$ be the center of the projection

$$\pi : (\mathbb{P}^3 \setminus \{o\}) \rightarrow \mathbb{P}^2.$$

Let $B \subset \mathbb{P}^2$ be the apparent contour of the surface S with respect to the projection π . Additionally, let $C \subset S$ be the contour, which consists of points $x \in S$ where the tangent plane $T_x S$ passes through the center of the projection o . In simpler terms, for any point $y \in B$, there exists a corresponding point $x \in C$ such that the center of projection o lies on the tangent plane $T_x S$. Consequently, when the tangent plane $T_x S$ is projected via π , it becomes a line l within the plane that tangentially intersects the contour C at point x . The dual of this tangent to C corresponds to the point $y \in C^*$. Together, these points represent the tangential planes to any point on S that pass through the center of the projection o . The entire set of tangential planes is denoted as S^* . The requirement for these planes to pass through the center of the projection imposes a linear condition, meaning the dual surface S^* is intersected by a hyperplane, known as a planar section. \square

Outline

The thesis is organized with an introduction followed by three main chapters. The introduction provides a broad overview of the addressed topic, briefly discussing the problems tackled in subsequent chapters. Initial emphasis is placed on three primary motivations behind the work. The first motivation originates from the field of computer vision, where surfaces are constructed from images captured by cameras. These images can be considered as central projection with the camera's position as the center. The second motivation explores the conditions determining the existence of such projection, referencing Chisini's conjecture. The final motivation is drawn from insights presented in the paper [47], specifically concerning the Dupin coordinate system. This work serves as a foundation for the reconstruction of Dupin cyclides from planar sections. Additionally, the motivation highlights the broader application of Dupin cyclide coordinate systems in addressing diverse challenges within various problems in physics.

After discussing the motivations, a survey of prior works on surface reconstruction, including cubic surfaces, smooth surfaces, and surfaces with ordinary singularities, is presented. This background sets the stage for the second chapter, which is also published in the paper [48], focusing on the reconstruction of Darboux cyclides. The chapter delves into the planar curve obtained through

projection onto a plane using a calibrated camera. The singularities of that curve are thoroughly examined, and a method involving resolving special points is employed to reconstruct the desired surface.

In the third chapter, which is also published in the paper [49], attention turns towards the reconstruction of a Veronese surface. The study advances through parameterization, culminating in the generation of a planar map that has the same branching curve as the Veronese surface. Singular points on the branching curve are examined, and then they are resolved to deduce a space curve, i.e., the linear normalization embedded within the Veronese surface.

The final chapter investigates the reconstruction of a Dupin cyclide from a curve obtained by planar cross-section, as published in the paper [50]. This particular curve possesses focal properties, which are used to determine the families of Dupin cyclides that have the given curve as their planar section. Through these comprehensive explorations, the thesis contributes to the understanding and application of various surface reconstruction techniques.

2.2 Motivation

2.2.1 Computer vision

The advancements in computer vision problem-solving have introduced another field called algebraic vision, which translates computer vision problems into the language of algebraic geometry. A key element in this approach is the use of projective varieties in multiview geometry, as highlighted in [51]. The primary focus has been on developing algorithms for constructing correspondences between 2D images, identifying the 3D positions of matched points derived from 2D points, and extracting related camera parameters. The studies delve into various camera types, emphasizing the importance of camera calibration, which is closely related to measurements and coordinate systems in Euclidean geometry.

In the real plane, two conics intersect at two points instead of four algebraic solutions. Two of these points are complex: $(1 : \mathbf{i} : 0)$ and $(1 : -\mathbf{i} : 0)$, known as circular points, located on the line at infinity. Analogously, in 3D space, two spheres surfaces intersect through a conic rather than a quartic curve. This implies the existence of an imaginary conic on the plane at infinity, known as the absolute conic, represented by the equation $x^2 + y^2 + z^2 = 0$.

The absolute conic is a fundamental concept in Euclidean Geometry, providing completeness to the framework closely related to the perpendicularity of lines. Two lines are perpendicular when their directions are conjugate with respect to the absolute conic. In the context of a camera positioned away from the plane at infinity, the world's plane at infinity is mapped one-to-one with the image plane, and the absolute conic must project onto a conic in the image plane. The knowledge of this corresponding conic within the image plane is useful for camera calibration.

Capturing plane images from a space surface can be seen as a central projection $p : \mathbb{P}^3 \rightarrow \mathbb{P}^2$, with the camera serving as the projection center. Considering, without loss of generality, the center of projection to be the point $(0:0:0:1)$, then the projection is defined by a 3×4 matrix of rank three with the block structure represented as $P = [I_{3 \times 3} | 0_{3 \times 1}]$. In this configuration, $I_{3 \times 3}$ represents the identity matrix, and $0_{3 \times 1}$ signifies a zero 3-vector. This matrix, referred to as the camera matrix, is fundamental for 3D reconstruction from a single view. The center of a camera P is the point defined by the kernel $\ker(P) \subset \mathbb{P}^3$. In particular, the calibrated camera is determined by a 3×4 matrix whose left 3×3 portion lies in the special orthogonal group $SO(3)$.

There are also reconstructions observed in two views, using the fundamental matrix F , which matches points $x' \leftrightarrow x$ from two images and fulfills the condition $x'Fx = 0$. These images are projections from two cameras that have captured the same object in space. Once the fundamental matrix is computed, it is possible to determine the two camera matrices, up to projective transformations. This, in turn, enables the deduction of spatial coordinates through a technique known as triangulation. All these aspects are explored in Epipolar Geometry [51].

Outside pointwise reconstruction, there are some notable outcomes (see [52]) in reconstructing surfaces from specific apparent contours in a discrete manner. Initially, the challenge is tackled by addressing the mapping between successive apparent contours through the use of epipolar correspondence. Subsequently, surface approximation is accomplished

2.2.2 Chisini's conjecture

As mentioned in the preceding section, numerous studies have been conducted to reconstruct 3D objects from 2D images. A significant focus of these investigations has been on reconstructing algebraic surfaces. Initially, the emphasis was on the challenge of reconstructing smooth surfaces by demonstrating the existence of a projection onto the plane that generates a given branching curve.

The question of whether a surface can be reconstructed from its branching curve when projected onto the plane was posed by Chisini (see [53],[54], [55]). He examined a smooth compact complex surface, denoted as S , and a generic morphism $f : S \rightarrow \mathbb{P}^2$, with B as the branching curve with nodes and cusps as singularities that fulfills the following conditions:

- (a) the ramification curve R of f is smooth and reduced;
- (b) $f|_R : R \rightarrow B$ has degree one;
- (c) $f^*(B) = 2R + C$, where R is irreducible and non-singular, and C is reduced;

Two generic morphisms (S_1, f_1) and (S_2, f_2) with the same branching curve B are said to be equivalent if there exists an isomorphism $\phi : S_1 \rightarrow S_2$ such that $f_1 = f_2 \circ \phi$. Therefore, when it

is said "f is unique," it means "f is unique up to equivalence." The fundamental inquiry was to determine the degree of f to which the curve B characterizes the morphism f.

Based on these assumptions, Chisini then introduced a proposition and offered an alleged proof, which later was considered as conjecture.

Conjecture 2.2.1. (Chisini). Let B be the branching curve of a a generic morphism $f : S \rightarrow \mathbb{P}^2$ of degree $\deg f \geq 5$. Then f is unique up to equivalence.

The necessity of imposing a bound on the degree of f is illustrated by a counterexample presented by Chisini and Catanese (refer to [55], [56], [54]). Consider the curve B as a sextic curve with nine cusps; it is the dual curve of a nonsingular plane cubic. This curve serves as the branch curve for four distinct non-equivalent generic morphisms. Among these morphisms, three are of degree four and represent planar maps obtained by projections of the Veronese surface (for more details, see and section 2.3). The fourth one, of degree three, is the projection onto \mathbb{P}^{2*} of the elliptic ruled surface obtained as the preimage of C in the incidence variety $\mathbb{P}^2 \times \mathbb{P}^{2*}$

There were some attempts, but Kulikov proved that the Chisini conjecture holds for generic morphisms of degree greater than or equal to 5 with branch curves that are cuspidal, i.e., characterized by the absence of nodes (see [56]). Additionally, as outlined in [54], Kulikov established a condition for the validity of the conjecture. He identified that the degree of f must exceed a specific function, depending on the degree, genus, and the number of cusps present in the branching curve.

Theorem 2.2.1. Consider a generic morphism $f : S \rightarrow \mathbb{P}^2$ of degree $\deg f = N$ with branch curve $B \subset \mathbb{P}^2$. Denote by $2d$ the degree of B (it is always even), by g the genus of the desingularization of B, and by c the number of cusps of B. The morphism $f : S \rightarrow \mathbb{P}^2$ is uniquely determined by B if

$$N > \frac{4(3d + g - 1)}{2(3d + g - 1) - c} \tag{2.1}$$

Thus, in this case the Chisini Conjecture holds for B

In general, as deduced from [54], the number of non-equivalent generic morphisms with a given branching curve B is less than or equal to 2^{2g+c-1} .

Here, some results and cases obtained by Kulikov are presented.

- (a) Let $f : S \rightarrow \mathbb{P}^2$ be any generic morphism of $S = \mathbb{P}^1 \times \mathbb{P}^1$. Then, for the branching curve B of f, the generic morphism f is unique.
- (b) Let S be a K3 surface and $f : S \rightarrow \mathbb{P}^2$ any generic morphism. Then, for the branching curve B of f, the generic morphism f is unique.
- (c) Let S be an Enriques surface and $f : S \rightarrow \mathbb{P}^2$ any generic morphism. Then, for the branching curve B of f, the generic morphism f is unique except, possibly, for $\deg f = 4$. In the exceptional case where $\deg B = 12$, $g = 19$, $c = 36$, $n = 0$, if such a morphism exists, then:

- (a) For B , there exist at least two non-equivalent generic morphisms.
- (b) Any generic morphism f' with such a branching curve B has $\deg f' \leq 4$.
- (d) Let S be an abelian surface and $f : S \rightarrow \mathbb{P}^2$ any generic morphism. Then, for the branching curve B of f , the generic morphism f is unique except, possibly, for $\deg f = 6$. In the exceptional case where $\deg B = 18$, $g = 28$, $c = 72$, $n = 36$, if there exists a generic morphism f' which is not equivalent to f , then $\deg f' \leq 6$.
- (e) Let $S \subset \mathbb{P}^N$ be a complete intersection and $f : S \rightarrow \mathbb{P}^2$ the restriction of a generic projection. Then, for the branching curve B of f , the generic morphism f is unique.
- (f) The Chisini Conjecture holds for the dual curve B of a nodal plane curve except, possibly, for:
 - (a) $\deg B = 30$, $g = 10$, $c = 72$, $n = 324$;
 - (b) $\deg B = 20$, $g = 6$, $c = 45$, $n = 120$;
 - (c) $\deg B = 18$, $g = 5$, $c = 39$, $n = 92$;
 - (d) $\deg B = 16$, $g = 4$, $c = 33$, $n = 68$.

In all the exceptional cases, if there exist non-equivalent generic morphisms for B , then these morphisms have degree ≤ 6 .

- (g) The Chisini conjecture holds for a curve B of genus $g \leq 3$.
- (h) The Chisini conjecture holds for B satisfying the inequality $d > 3(g - 1)$.

2.2.3 Cyclide coordinates

The investigation and analysis of phenomena in physics necessitate the application of various mathematical concepts, giving rise to the interdisciplinary field known as physical mathematics. An integral aspect of this field involves solving problems related to the theory of potentials, with a significant emphasis on solving Laplace's equations. In his dissertation on Potential Theory [57], Maxime Böcher presented a captivating introduction, tracing the evolution of cyclide coordinates and their application in physics.

Initial efforts were made to determine potentials for homogeneous bodies, closely resembling spherical bodies. In these endeavors, the use of spherical functions introduced by Legendre played a crucial role in addressing the gravitational attraction of rotating bodies. Importantly, solutions to these problems needed to satisfy specific values within the boundaries of these bodies. Establishing boundary conditions required consideration of general bodies, exemplified by one of the six confocal cyclides.

The term 'cyclide' was initially introduced by Dupin (1822) to describe a surface where all lines of curvature are circles or straight lines. However, following Darboux's approach, the term 'cyclide' is now used to denote a more general category of surfaces. Specifically, any fourth-degree surface having an absolute conic as a double curve falls under this broader definition. Conversely, the term 'Dupin's cyclid' is reserved for a surface that is the inverse of the torus, as originally referred to by Dupin.

Particularly noteworthy was the simultaneous discovery by Moutard and Darboux of the orthogonal system composed of general confocal cyclids. Subsequently, Darboux used this triple orthogonal set of surfaces to establish a system of curvilinear coordinates, now known as cyclid coordinates. One particular case of cyclid coordinates is a class called offset Dupin cyclide systems that were introduced in [58] and used for the separation of variables in the Laplace equation (see an overview in [59]).

Inspired by these developments, the idea emerged to classify Dupin cyclide coordinate systems based on their singularities, which were bicircular quartics obtained as planar sections of cyclides in each of three directions, as detailed in the article [47]. During the study of these systems, certain conclusions were drawn regarding the reconstruction of a Dupin cyclide from planar section curves, i.e., bicircular quartics, and its focal properties.

2.3 Reconstruction of smooth surfaces

Following Chisini's conjecture regarding the existence of a projection of a surface onto a plane with a given branching curve, attempts were made to find the complete construction, including both the surface and the projection. The simplest example and initial construction involved smooth cubic surfaces (see [60]).

Example 2.3.1. Smooth Cubic Surfaces. Consider a central projection $p : S \rightarrow \mathbb{P}^2$ of a cubic surface from a point. Both the ramification curve and the branching curve have degree six. Notably, the branching curve is determined as the discriminant of the cubic surface's equation.

The complexity of the discriminant is elementary since the equation of a smooth cubic surface can be simplified into a straightforward form, known as the Tischirnhaus form. This transformation is achieved through projective automorphisms within the projective space \mathbb{P}^3 . The equation of S can be expressed as:

$$F = w^3 + A(x, y, z)w + B(x, y, z),$$

with the discriminant given by

$$\Delta = -(4A^3 + 27B^2).$$

The equation $\Delta = 0$ represents the branching curve, which consists of six cusps. There is a unique conic passing through the six cusps, defined by the polynomial A . Once the conic is identified, the

next step involves searching for the cubic curve defined by B that passes through these six cusps. Let C be a cubic form in the ideal of the six cusps, linearly independent of multiples of the conic form. One can then express $B = \lambda C + LA$, where $\lambda \in \mathbb{C}$ and L is a linear form. The next step involves imposing and solving the following equation:

$$-(4A^3 + 27(\lambda C + LA)^2) = 0$$

Finally, one can determine the equation of the surface using the defining polynomial F .

For surfaces with equations of higher degrees, proceeding in the same way becomes more challenging. Finding the polynomials that pass through all the singularities of the branching curve requires a different technique.

According to Chisini, the ramification curve of a smooth surface under a generic projection is smooth, and the branching curve has nodes and cusps as singularities, which are obtained from the projection. A cusp in a branching curve occurs when the ray passing through the projection's center is tangential to the surface, while a node appears when the ray intersects the surface at two distinct points.

Drawing upon the properties and knowledge outlined above, D'Almeida successfully demonstrated the existence of smooth surfaces of higher degrees from branching curves.

Theorem 2.3.2 (D'Almeida [61]). Let B be a plane curve of degree $d(d-1) \geq 3$. The necessary and sufficient condition for the existence of a smooth surface $S \subset \mathbb{P}^2$ and a generic point p in \mathbb{P}^3 such that B is the branching curve of the projection of S through p is as follows: The curve B has $n = \frac{d(d-1)(d-2)(d-3)}{2}$ ordinary double points, $k = d(d-1)(d-2)$ cusps, and no other singularities. There are two curves, G_1 and G_2 , of degrees $d^2 - 3d + 2$ and $d^2 - 3d + 3$ respectively, without a common component, passing through the singular points of B . The minimal degree of a plane curve containing the singular points of B is $d^2 - 3d + 2$.

This result, obtained by D'Almeida, is explained and presented in a computationally convenient manner in the paper [60]. An algorithm is introduced for determining the polynomials G_1 and G_2 while defining a mapping from the branching curve to the ramification curve.

The branching curve serves as a crucial tool in surface determination, providing essential information for reconstructing the ramification curve (up to projective equivalence) as the normalization curve. This is achieved through the property of the ramification curve being a complete intersection, ensuring its linear normality, i.e., the ramification curve is not the projection of a non-degenerate curve residing in a larger projective space. All of these aspects are summarized in the following proposition.

Proposition 2.3.1. [60] The ramification curve R of a projection is linearly normal. This means that the standard map $H^0(\mathbb{P}^3, \mathcal{O}_{\mathbb{P}^3}) \rightarrow H^0(R, \mathcal{O}_R)$ is an isomorphism. In particular, $H^0(R, \mathcal{O}_R)$ is 4-dimensional.

The normalization of singularities of the branching curve makes it possible to reconstruct the ramification curve, which can be embedded in \mathbb{P}^3 . To achieve this embedding, the analysis involves considering the twisted invertible sheaves $\mathcal{O}_R(1)$, which are twists of the canonical sheaf ω_R by $(-2d+6)$ [[62] Exercise II.8.4e].

On the other hand, the branching curve is the image under the projection of the ramification curve. The projection determines an isomorphism between the global sections of the canonical sheaves ω_R and ω_B . Furthermore, $\omega_B(-2d+6) \cong \omega_R(-2d+6) \cong \mathcal{O}_R(1)$.

From the last isomorphism, the associated complete linear series $|\omega_B(-2d+6)|$ embeds the curve B into \mathbb{P}^3 , such that the image is linearly normal and coincides (up to projective transformations) with R .

Additionally, it is proven in [60] that the sheaf ω_B is isomorphic to the ideal sheaf $\mathcal{I}(d^2-d-3)$. This sheaf is the restriction to the curve B of the ideal sheaf \mathcal{K} on \mathbb{P}^2 , of the singularities of the curve B . Moreover, there is the following sequence of isomorphisms:

$$\mathcal{O}_R(1) \cong \omega_B(-2d+6) \cong \mathcal{I}(d^2-d-3)(-2d+6) = \mathcal{I}(d^2-3d+3).$$

As emphasized in Proposition 2.3.1, the dimension of the global sections, $H^0(\mathbb{P}^3, \mathcal{O}_R(1))$, is four. Consequently, within the defining ideal J of the ideal sheaf \mathcal{I} , there are precisely four forms of degree d^2-3d+3 . Furthermore, considering the following isomorphisms:

$$\mathcal{O}_R \cong \omega_B(-2d+5) \cong \mathcal{I}(d^2-d-3)(-2d+5) = \mathcal{I}(d^2-3d+2),$$

it is evident that within the invertible sheaf $\mathcal{I}(d^2-3d+2)$, there exists a single global section denoted by G_1 , of degree d^2-3d+2 . Multiplying G_1 by the three monomials x, y, z produces three other linearly independent global sections: xG_1 , yG_1 , and zG_1 , all of degree d^2-3d+3 . Therefore, within the defining ideal of the singularities of the branching curve $\mathcal{I}(d^2-3d+3)$, only a single unique global section G_2 can occur, up to multiples of G_1 and scalar factors.

With the four forms of degree d^2-3d+3 mentioned earlier, one can construct the rational map as follows:

$$\phi : (x_0 : x_1 : x_2) \rightarrow (x_0G_1 : x_1G_1 : x_2G_1 : G_2) = (x_0 : x_1 : x_2 : \frac{G_2}{G_1})$$

It is important to note that the map becomes degenerate at points where both G_1 and G_2 are zero. To address these missing points, it becomes necessary to consider the closure of its image. Furthermore, by composing the projection with ϕ , the resulting map is the identity.

$$\pi \circ \phi : (x_0 : x_1 : x_2) \rightarrow (x_0G_1 : x_1G_1 : x_2G_1) = (x_0 : x_1 : x_2)$$

Then, the ramification curve is obtained as the image of the map ϕ when restricted to the branching curve. To continue with the process of reconstruction, it is necessary to search in the defining ideal

of the ramification curve for two polynomials of degrees d and $d - 1$. Indeed, there exists a unique polynomial T of degree $d - 1$, which is considered as the partial derivative of the surface equation with respect to the direction of the projection. Moreover, there is another polynomial F of degree d , which is unique up to multiples of u_0T , u_1T , u_2T , and scalars. This polynomial is the equation of the surface.

2.4 Reconstruction of surfaces with ordinary singularities

This section presents an extract from the construction of a surface $S \subset \mathbb{P}^3$ with ordinary singularities, as discussed in [60]. The approach employed here is similar to that used for smooth surfaces, with some minor distinctions. Surfaces with ordinary singularities are obtained by projecting a surface in higher-dimensional space to three-dimensional space $\pi : \Sigma \rightarrow \mathbb{P}^3$. The projected surface is requested either to be smooth or to have a finite number of double points and is defined as follows.

Definition 2.4.1. An irreducible, projective surface $S \subset \mathbb{P}^3$ is said to have ordinary singularities if its singular locus Z is either empty or it is a curve Γ , called the double curve of S , with the following properties:

- (a) Γ has at most finitely many ordinary triple points, such that the germ of S there is analytically equivalent to the one of the affine surface in \mathbb{C}^3 with equation $xyz = 0$ at the origin;
- (b) every non-singular point of Γ is either a nodal point, i.e., the germ of S there is analytically equivalent to the one of the surface with equation $x^2 - y^2 = 0$ at the origin, or a pinch point, i.e. the germ of S there is analytically equivalent to the one of the surface with equation $x^2 - zy^2 = 0$ at the origin;
- (c) for every irreducible component Γ' of Γ , the general point of Γ is a nodal point of S , in particular, there are only finitely many pinch points for S .

Consider the projection $\pi : S \rightarrow \mathbb{P}^2$ of the surface with ordinary singularities S from a point $p \in \mathbb{P}^3 \setminus S$ to a plane \mathbb{P}^2 . Then the contour, denoted by Y , is the union of the singular locus Z of the surface with the ramification curve R , where the latter consists of all points where tangents to the surface pass through the center of projection. The contour Y is determined by the surface's equation and the partial derivative in the direction of the projection.

The apparent contour, denoted by C , is the union of the image of the singular locus W with the branching curve. Furthermore, the apparent contour is defined by the discriminant equation of the surface concerning the direction of the projection.

It is crucial to emphasize that, unlike the case of smooth surfaces, the surface with ordinary singularities cannot be obtained through the normalization process of the branching curve with the

provided data. Indeed, as noted in [63], if $\nu : X \rightarrow S$ is the normalization map, one has the following commutative diagram:

$$\begin{array}{ccc}
 X & \xrightarrow{\nu} & S \subset \mathbb{P}^3 \\
 & \searrow \psi & \downarrow \pi \\
 & & \mathbb{P}^2
 \end{array}$$

In this context, the surface X projected along ψ has the same branching curve as S projected along π . Consequently, it becomes crucial to examine the image of the singularities of the surface and their positioning in relation to the branching curve.

It is also noteworthy that the focus will be on investigating favorable projections $S \rightarrow \mathbb{P}^2$, referred to as good projections, which satisfy the following conditions:

- (a) the restriction of the projection to the contour is injective, except for at most finitely many points;
- (b) the ramification curve R is smooth, and the branching curve B has at most nodes and ordinary cusps;
- (c) the line through the center of projection and a point in the ramification curve R intersects S with multiplicity exactly 2 at that point, except for preimages of cusps and singular points on the surface;
- (d) the singular image W has only nodes and ordinary triple points (D_4 singularities), the latter arising as images of spatial triple points;
- (e) the singular image W and the branching curve B meet either transversally, or tangentially with order 2 at smooth points; in particular, ensure that pinch points are mapped to transversal intersections.

It is proved in [60] that a general projection π is indeed a good projection, with potential singularities that may arise in the apparent contour $C = B \cup W$, as detailed below:

- (a) nodes or cusps of B ,
- (b) nodes or triple points of W ,
- (c) tangential intersections of B and W ,
- (d) transversal intersections of B and W whose preimages are distinct,
- (e) transversal intersections of B and W coming from pinch points.

The good projection is an isomorphism everywhere except at these special points. Identifying the special points of the apparent contour makes it possible to continue with the process of reconstruction. Therefore, a detailed local analysis at these special points is necessary, and the application of sheaf theory becomes crucial to achieve a comprehensive global solution.

Global reconstruction

Referring to the paper [60], the task is to determine a rational map that sends C to Y , which is equivalent to computing the space of homogeneous forms that satisfy particular vanishing conditions at the special points of C . This necessitates establishing a connection between the global sections of $\mathcal{O}_Y(1)$ and those of a sheaf in C . As proven in [60], there exists an isomorphism between the global sections $H^0(\mathbb{P}^3, \mathcal{O}_{\mathbb{P}^3})$ and $H^0(Y, \mathcal{O}_Y)$, indicating that the space of global sections is four-dimensional. On the other hand there exist the isomorphism

$$H^0(Y, \mathcal{O}_Y(1)) \cong H^0(C, \mathcal{H}om_{\mathcal{O}_C}(\pi_* \mathcal{O}_Y, \mathcal{O}_C)(d^2 - 3d + 3)).$$

The latter, $\mathcal{H}om_{\mathcal{O}_C}(\pi_* \mathcal{O}_Y, \mathcal{O}_C)$, is an ideal sheaf and is supported at the special points of the curve C . Indeed, this sheaf is an ideal sheaf because the map $\Phi : \mathcal{H}om_{\mathcal{O}_C}(\pi_* \mathcal{O}_Y, \mathcal{O}_C) \rightarrow \mathcal{O}_C$ is injective, induced by the injectivity on the stalks for every point of the apparent contour. The map Φ is an isomorphism for smooth points, but for the special points, it is necessary to consider computing the conductor ideal between the localizations of the completions as shown below.

Let $c \in C$ be a closed point on C that is a special point. Define $E = \hat{\mathcal{O}}_{C,c}$ and $F = \bigoplus_{y_i: \pi(y_i)=c} \hat{\mathcal{O}}_{Y,y_i}$. Then the homomorphism $\phi : E \rightarrow F$ induced by projection is injective, and its image is equal to the conductor ideal $\{w \in E : wF \subset E\}$. The map ϕ becomes an isomorphism when localized by the non-zero elements of E , concluding that $\mathcal{H}om_{\mathcal{O}_C}(\pi_* \mathcal{O}_Y, \mathcal{O}_C)$ is an ideal sheaf.

The final step involves taking the completions of the stalks of the ideal sheaf for every point $c \in C$ and considering the following map:

$$\Phi : \mathcal{H}om_{\mathcal{O}_C}(\pi_* \mathcal{O}_Y, \mathcal{O}_C)_c \otimes \hat{\mathcal{O}}_{C,c} \rightarrow \hat{\mathcal{O}}_{C,c}$$

whose image is the ideal sheaf \mathcal{I} . The four independent global sections of the ideal sheaf \mathcal{I} of degree $d^2 - 3d + 3$ define a map from the apparent contour C to the contour Y .

All these arguments are well-explained in the algorithm `ReconstructionGeneralSurfaces` presented in [60]. Various results for different surfaces are summarized in the following table.

d	B	W	n(B)	c(B)	n(W)	t(W)	t	p	o	time	type
4	8	2	8	12	1	0	0	8	4	12s	ruled(elliptic base)
4	8	2	4	12	0	0	4	4	4	6s	Del Pezzo
4	6	3	4	6	1	0	2	4	6	4s	ruled
4	12	0	12	24	0	0	0	0	0	5s	smooth
4	6	3	0	9	0	1	6	6	3	3s	Veronese
5	20	0	60	60	0	0	0	0	0	180s	smooth
5	10	5	12	18	3	1	18	8	12	400s	DelPezzo
5	8	6	12	9	6	1	12	6	15	130s	ruled

Table 2.1: The table shows the degree of the surface S , of the ramification curve B , and of the singular image W ; then the number of nodes and cusps of B , the number of nodes and triple points of W , the number of tangential intersections, pinch points, and other transversal intersection points, and the computing time in CPU seconds.

Chapter 3

Shahroz Khan: IDEAS@MI: Intelligent Data-driven systems for digital design in Maritime Industry

Executive Abstract

The maritime industry (Mar-I) is currently facing unprecedented pressure from environmental regulatory bodies to radically redesign its assets for decarbonisation. This necessitates intelligent design approaches and efficient simulation tools to revolutionise and reconfigure existing maritime systems. The drive for new systems is propelled by significant regulatory alterations such as the IMO 2020, which mandates a reduction in emissions [64], and the advent of disruptive technologies within the frameworks of Industry 4.0 and 5.0.

Currently, for design tasks in the Mar-I, designers and engineers use extensively off-the-shelf parametric modellers and computational tools. These tools are characterised by conservatism, for they are built to generate shapes lying in the neighbourhood of a successful baseline/parent shape. Next, these modellers are coupled with optimisers for improving the baseline shape against performance criteria (e.g., ship wave resistance, sea-keeping, structural strength, etc.), which involve time-consuming simulations, e.g., computational fluid dynamics (CFD). At the end of the process, the new design is likely a local optimum whose shape is a minor variation of the existing one. Conclusively, the coexistence of conservative parametric modellers with high-cost simulations and a large number of design parameters needed for shape optimisation of complex shapes leads to non-efficient simulation-driven design pipeline that suffers from the curse of high-dimensionality and a limited capability to explore design spaces efficiently for delivering variant, innovative, user-centred and truly optimal designs.

Therefore, aligning maritime design schemes with Industry-4.0 and -5.0 trends, this PhD thesis aims to propel initiatives for the transfer and customisation of a range of novel AI technologies that cover the full spectrum of simulation-driven shape optimisation activities of maritime assets. These include state-of-the-art data-driven approaches such as deep learning (generative adversarial networks), dimension-reduction (DR), parametric sensitivity analysis (PSA) and generative-design techniques coupled with disruptive hydrodynamic simulation paradigms to: i) improve the efficiency of design space exploration, ii) reduce the overall computational cost, iii) develop versatile design parameterisation, and iv) integrate human intelligence in the design process. The objectives of efficient design space exploration and reduction of computational cost are achieved by lowering the design space’s dimensionality and creating high-fidelity surrogate models, which is achieved via i) eliminating the parameters which are less sensitive/significant towards the physical QoI (Quantity of Interest) using PSA and ii) extracting the latent feature with feature extraction/embedding approaches to form a subspace of reduced dimensionality. Figure 3.1 shows the overview of the layout of the approaches proposed to achieve the aforementioned objectives.

First, a novel intra-sensitivity concept is proposed to study the local behaviour of parametric sensitivities and eliminate instabilities - a parameter can be sensitive in certain local areas of the design space but become insensitive in others. Therefore, the outcome of intra-sensitivity allows designers to construct viable design spaces for the reliable execution of PSA. Afterwards, implementation of PSA or intra-sensitivity is expedited with a new geometric-moment dependent PSA that harnesses the geometric variation in a design space using geometric moments to measure parametric sensitivities. A shape-supervised dimension reduction approach is also developed. It extracts a high-level geometry description as a shape signature vector and uses it as a substitute for physics to construct a physics-informed design subspace. A feature-to-feature learning strategy is also proposed to create a functionally-active subspace for expediting the construction of surrogate models at an off-line stage. For the versatile parameterisation of ship hulls, we developed ShipGAN using deep convolutional generative adversarial networks, so the resulting parametric modeller is generic with the ability to perform feasible and plausible design modifications for a large variety of hulls. Finally, we propose a generative and interactive design tool which aids users during optimisation by guiding the design exploration towards user-centred and physically optimised designs. A detailed overview of each of the proposed approaches is provided below:

Intra-sensitivity. In the proposed pipeline, the first approach is a novel intra-sensitivity [65] concept to study the local behaviour of parametric sensitivities and eliminate instabilities - a parameter can be sensitive in some local regions of the design space but become insensitive in others. Such behaviour makes PSA vulnerable to fluctuations even with slight perturbation in the parametric ranges of the design space. For this purpose, we first appeal to the Active Subspace Method (ASM) and develop an ASM-based regional sensitivity analysis, which investigates parametric sensitivity in local regions of the design space and aids in extracting parameters’ intra-sensitivity. This regional

analysis is applied in conjunction with a Dynamic Propagation Sampling approach for tackling the computational complexity arising when high-dimensional problems are concerned. Once sensitive and intra-sensitive parameters are identified, then free-form features correlated to these parameters are evaluated using a feature saliency map built with the aid of Hausdorff distance. Therefore, the outcome of intra-sensitivity allows designers to construct viable design spaces for the reliable execution of PSA.

Geometric moment-dependent PSA. To leverage the computational burden that is likely to occur from implementing PSA or intra-sensitivity, a new geometric-moment dependent PSA (GMDPSA) [66] is proposed that harnesses the geometric variation of designs in a design space using geometric moments as a geometrical QoI to measure parametric sensitivities. We construct a Shape-Signature-Vector (SSV) and propose to use it as a substitute for physics. SSV is composed of shapes' integral properties, in our case geometric moments and their invariants of varying order, and is used as quantity-of-interest (QoI) for prior estimation of parametric sensitivities. Opting for geometric moments is motivated by the fact that they are intrinsic properties of shapes' underlying geometry, and their evaluation is essential in many physical computations as they act as a medium for interoperability between geometry and physics.

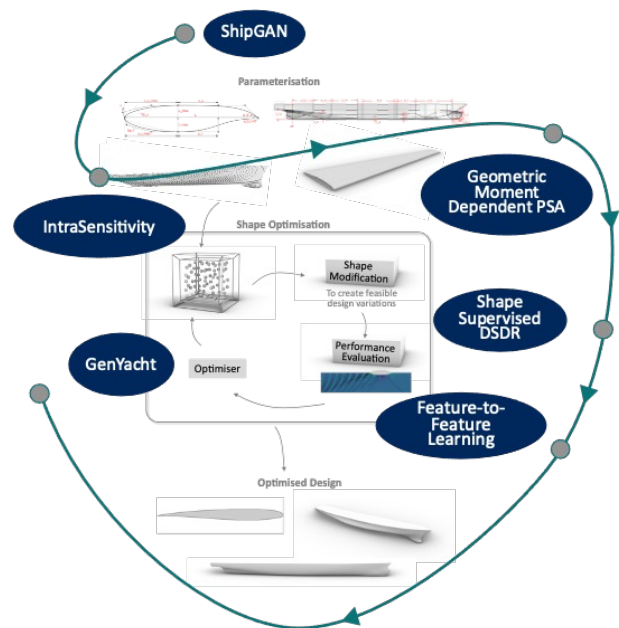


Figure 3.1: Layout of the proposed approaches within IDEAS@MI pipeline.

The proposed GMDPSA has been validated with regard to the capability of global- and composite-SSV to reveal parametric sensitivities of different ship hulls for the wave-making resistance coefficient (C_w), which is a critical QoI towards improving the ship's efficiency and thus decreasing emissions. More importantly, the longitudinal distribution of the volume below the ship's floating waterline, which is measurable via geometric moments, has an impact on C_w . Through extensive experimentation, we show a strong correlation between the sensitive parameters obtained with respect to SSV and those based on C_w . Consequently, we can estimate parameters' sensitivity with considerably reduced computational cost compared to when sensitivity analysis is performed with respect to C_w . Finally, two design spaces are constructed with sensitive parameters evaluated from SSV and C_w , and spaces' quality and richness are analysed in terms of their capability to

provide an optimised solution.

Shape supervised dimension reduction. In shape optimisation problems, subspaces generated with conventional feature extraction-based dimension reduction approaches often fail to extract the intrinsic geometric features of the shape that would allow the exploration of diverse but valid candidate solutions. More importantly, they also lack incorporation of any notion of physics against which shape is optimised. To simultaneously tackle these deficiencies, the proposed shape-supervised dimension reduction (SSDR) [67] uses higher-level information about the shape in terms of its geometric integral properties, such as geometric moments and their invariants. Their usage is based on the fact that moments of a shape are intrinsic features of its geometry, providing a unifying medium between geometry and physics. To enrich the subspace with latent features associated with the shape’s geometrical features and physics, we also evaluate a set of composite geometric moments, using the divergence theorem, for appropriate shape decomposition. These moments are combined with the shape modification function to form a decomposed SSV uniquely representing a shape. Afterwards, the generalised Karhunen–Loève expansion is applied to SSV, embedded in a generalised (disjoint) Hilbert space, which results in a basis of the shape-supervised subspace retaining the highest geometric and physical variance.

Feature-to-feature learning. A feature-to-feature learning strategy [68] is also proposed to create a functionally-active subspace for expediting the construction of physics-informed surrogate models. To achieve this, we adopted a two-step feature-to-feature learning approach to discover a lower-dimensional latent space based on the combination of geometry- and physics-informed principal component analysis and the active subspace method. In the first step, statistical dependencies implicit in the design parameters encode important geometric features of the underlying shape. During the second step, functional features of designs are extracted in terms of previously learned geometric features. Afterwards, geometric and functional features are augmented together to create a functionally-active subspace whose basis captures the geometric variance of designs and induces variability in the designs’ physics. As the new subspace accumulates both the functional and geometric variance, it can be exploited for efficient design exploration and the construction of improved surrogate models for designs’ physics prediction.

ShipGAN. For the versatile parameterisation of ship hulls, we introduce ShipGAN [69], a generic parametric modeller built using deep convolutional generative adversarial networks (GANs) for the versatile representation and generation of ship hulls. At a high level, the new model intends to address the current conservatism in the parametric ship design paradigm, where parametric modellers can only handle a particular ship type. We trained ShipGAN on a large dataset of 52,591 physically validated designs from a wide range of existing ship types, including container ships, tankers, bulk carriers, tugboats, and crew supply vessels. We developed a new shape extraction and representation strategy to convert all training designs into a common geometric representation

of the same resolution, as typically GANs can only accept vectors of fixed dimension as input. A space-filling layer is placed right after the generator component to ensure that the trained generator can cover all design classes. During training, designs are provided in the form of a shape-signature tensor (SST) which harnesses the compact geometric representation using geometric moments that further enable the inexpensive incorporation of physics-informed elements in ship design. We have shown through extensive comparative studies and optimisation cases that ShipGAN can generate designs with augmented features resulting in versatile design spaces that produce traditional and novel designs with geometrically valid and practically feasible shapes.

GenYacht. Finally, we proposed a generative and interactive design tool, GenYacht [70] to empower experienced and novice designers to create various design alternatives. Among them, a user can select a hull design with desirable characteristics based on its appearance and hydrostatics/hydrodynamic performance. GenYacht first explores a given design space using a generative design technique (GDT), which creates uniformly distributed designs satisfying the given design constraints. These designs are then presented to a user, and single or multiple designs are selected based on the user's requirements. Afterwards, based on the selections, the design space is refined using a novel space-shrinking technique (SST). In each interaction, SST shrinks the design space, which is then fed into GDT to create new designs in the shrank space for the next interaction. This shrinkage of design space guides the exploration process and focuses the computational efforts on user-preferred regions. The interactive and generative design steps are repeated until the user reaches a satisfactory design(s). The efficiency of GenYacht is demonstrated via experimental and user studies, and its performance is compared with interactive genetic algorithms.

Chapter 4

Andriamahenina Ramanantoanina: Barycentric rational interpolation

The barycentric form can be traced back to [71] where it was referred to as normalized Lagrangian interpolation. We consider a set of values f_0, \dots, f_n , and a set of scalars a_0, \dots, a_n . A barycentric formula is of the form

$$\frac{\sum_{i=0}^n a_i f_i}{\sum_{i=0}^n a_i}. \quad (4.1)$$

We recall several rational barycentric interpolation techniques in various context.

4.1 Barycentric rational interpolation

We consider a set of interpolation data f_0, \dots, f_n , a set of real numbers x_0, \dots, x_n that we call nodes, and a set of real numbers w_0, \dots, w_n that we call weights. A barycentric rational interpolant $r(x)$ that interpolates f_i at x_i , that is $r(x_i) = f_i$, is given by

$$r(x) = \sum_{i=0}^n \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j} f_i. \quad (4.1)$$

One can verify, by multiplying the numerator and the denominator by $\ell(x) = (x - x_0) \cdots (x - x_n)$, that $r(x)$ is a general rational interpolant such that we have the interpolation property $r(x_i) = f_i$. [72] observe that the interpolation property is satisfied if and only if the weights w_k are non-zero for all k .

As a general rational function, it might occur that $r(x)$ has poles. In the context of curve design, however, it is more desirable to have a curve that is continuous everywhere in $[x_0, x_n]$. Hence, it would be ideal to find some constraints on the weights w_k so that $r(x)$ has no poles. [72] show that it is necessary that the weights w_k have alternating signs. [73] states that the interpolation problem is

well-conditioned if the weights have the same absolute values. By combining these two conditions, [73] deduced that a barycentric interpolant with the particular set of weights $w_k = (-1)^k$ for all k , called Berrut weights, is well-conditioned and has no poles.

There is another specific way to describe a set of weights, called Lagrange weights, that implies a barycentric rational interpolation (4.1) to have no poles. Indeed, since the Lagrange polynomial interpolant, after [74], of degree n is

$$p(x) = \sum_{i=0}^n \frac{(x-x_0)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)} f_i, \quad (4.2)$$

then we can define the Lagrange weights w_k as

$$w_k = \prod_{i=0, i \neq k}^n \frac{1}{x_k - x_i}. \quad (4.3)$$

The Lagrange interpolant (4.2) can be written in first barycentric form, a term that first appeared in [75], as

$$p(x) = \ell(x) \sum_{i=0}^n \frac{w_i}{x - x_i} f_i, \quad (4.4)$$

where $\ell(x) = (x-x_0)\dots(x-x_n)$, and w_k as in (4.3). The alternative, second barycentric form (4.1) is obtained by dividing (4.4) by the interpolant of the constant function 1 written in first barycentric form as $1 = \ell(x) \sum_{i=0}^n \frac{w_i}{x - x_i}$. In general, [76] show that every rational function can be written in barycentric form. Consider a rational function $R(t) = \frac{N(t)}{D(t)}$. We first express $D(t)$ and $N(t) = D(t)R(t)$ in the first barycentric form (4.4) with respect to the same sequence of nodes. Then we recover the barycentric form (4.1) by writing the fraction $\frac{N(t)}{D(t)}$ and canceling out some factors.

For some particular set of nodes, the barycentric weights are given by explicit formulas, and can be deduced from the Lagrange weights [77]. For uniform nodes, $t_k = t_0 + kh$ for some positive scalar h , the weights are given by $w_k = \frac{(-1)^k}{h^n n!} \binom{n}{k}$ [78], which after canceling the common factor $h^n n!$ becomes

$$w_k = (-1)^k \binom{n}{k}.$$

We note that the weights w_k can also be defined as $w_k = \frac{1}{\ell'(x_k)}$ [79, 80]. This induces a formula for the weights with respect to the Chebyshev nodes of the first kind $t_k = \cos \frac{(2k+1)\pi}{2n+2}$ and the Chebyshev nodes of the second kind $t_k = \cos \frac{k\pi}{n}$, respectively as [81]

$$w_k = (-1)^k \sin \frac{(2k+1)\pi}{2n+2}, \quad w_k = (-1)^k \delta_k, \quad \delta_k = \begin{cases} 1/2 & k=0 \text{ or } k=n \\ 1 & \text{otherwise.} \end{cases}$$

An advantage of the barycentric form is that the form of its derivatives are very simple formulas. Indeed, consider a barycentric rational interpolant (4.1) with arbitrary weights $w_k \neq 0$ that have

alternating signs. [72] show the following formulas:

- if $x \neq x_i$

$$\frac{r^{(k)}(x)}{k!} = \frac{\sum_{i=0}^n \frac{w_i}{x-x_i} r[(x)^k, x_i]}{\sum_{i=0}^n \frac{w_i}{x-x_i}},$$

- and

$$\frac{r^{(k)}(x_j)}{k!} = -\frac{\sum_{i=0}^n w_i r[(x_j)^k, x_i]}{w_j}. \quad (4.5)$$

where $r[(x)^k, x_i] = r[x, \dots, x, x_i]$ is a divided difference of order $k + 1$. From these derivatives, we observe that barycentric rational functions, with the restrictions cited before, are C^∞ . This is very important in design and approximation since, for example, we are able to reproduce circular arcs perfectly as opposed to other methods such as splines.

4.2 Trigonometric barycentric rational interpolation

Analogously, we recall the definition of trigonometric barycentric interpolation. In the trigonometric setting, we assume $0 \leq x_0 < \dots < x_n < 2\pi$. A trigonometric barycentric rational interpolant $r(x)$ that interpolates f_i at x_i is given by

$$r(x) = \frac{\sum_{i=0}^n \text{cst} \frac{x-x_i}{2} w_i f_i}{\sum_{i=0}^n \text{cst} \frac{x-x_i}{2} w_i}, \quad (4.1)$$

where

$$\text{cst}(x) = \begin{cases} \csc(x) & \text{if } n \text{ is even,} \\ \cot(x) & \text{if } n \text{ is odd,} \end{cases} \quad \csc(x) = 1/\sin(x).$$

As for the classical case, the continuity of $r(x)$ is also crucial when it comes to curve design. Hence, some conditions on the weights should be satisfied so that the denominator of $r(x)$ does not vanish in $[0, 2\pi)$. For a given equidistant set of nodes, setting $w_k = (-1)^k$ implies the non-existence of poles [82, 73]. Another particular set of weights that induces the non-existence of poles can be extracted from the so-called Gauss's formula for trigonometric interpolation. [83] shows that a trigonometric polynomial $r(x) = a_0 + \sum_{k=1}^N (a_k \cos(kx) + b_k \sin(kx))$ of order N for some coefficients $a_0, \dots, a_N, b_1, \dots, b_N$ can be written in the form

$$r(x) = \sum_{i=0}^n \prod_{j=0, j \neq i}^n \frac{\sin \frac{x-x_j}{2}}{\sin \frac{x_i-x_j}{2}} f_i. \quad (4.2)$$

We note that in this case, $n = 2N$. By factoring out $\ell(x) = \prod_{j=0}^n \sin \frac{x-x_j}{2}$, we can write $r(x)$ in the

first barycentric form [84, 85] as

$$r(x) = \ell(x) \sum_{i=0}^n w_i \csc \frac{x-x_i}{2} f_i \tag{4.3}$$

where

$$w_i = \prod_{j=0, j \neq i}^n \csc \frac{x_i - x_j}{2}, \quad i = 0, \dots, n.$$

It follows that we can write any trigonometric rational function $R(x) = \frac{N(x)}{D(x)}$ in barycentric form (4.2), by expressing both the denominator $D(x)$ and the numerator $N(x) = D(x)R(x)$ in the first barycentric form (4.3). Then by writing the ratio $N(x)/D(x)$, and canceling out the common factor $\ell(x)$, we get the trigonometric barycentric form (4.2) of the rational function $R(x)$. A common property that both of these set of weights admits is that $\text{sgn}(w_k) = (-1)^k$. In fact, [86] show that the alternating sign of the weight is necessary in order to avoid poles.

4.3 Application in open curve design

The interpolation property of a barycentric interpolant can be exploited in the context of curve design. By writing the expression of a curve in barycentric form, we can force a curve to pass through a given point. We explore the idea of creating a curve using its expression written in barycentric form by comparing it with the classical methods.

The classical way to generate and manipulate a curve is by means of Bézier curves. We recall that a rational Bézier curve, defined by the control points P_0, \dots, P_n , and the weights $\alpha_0, \dots, \alpha_n$, is given by

$$P(t) = \frac{\sum_{i=0}^n \alpha_i B_i^n(t) P_i}{\sum_{i=0}^n \alpha_i B_i^n(t)}, \tag{4.1}$$

where $B_i^n(t) = \binom{n}{i} (1-t)^{n-i} t^i$ are the Bernstein polynomials of degree n . There are several ways to manipulate a Bézier curve: either by tweaking the Bézier control points, the weights [87], the so-called Farin points [88], or the shoulder points [89]. Although in order to reproduce a shape with a Bézier curve, we need to have a good intuition on where to place the control points.

On the other hand, the barycentric form can be useful since we can force a curve to pass through specific $n + 1$ point thanks to the interpolation property. Consider a set of points Q_i , a set of nodes t_i , and a set of real number $\beta_i > 0$. Consider a curve given by

$$P(t) = \frac{\sum_{i=0}^n \frac{(-1)^i \beta_i}{t-t_i} Q_i}{\sum_{i=0}^n \frac{(-1)^i \beta_i}{t-t_i}}. \tag{4.2}$$

There are several ways to edit a curve given in barycentric form (4.2). The most direct control

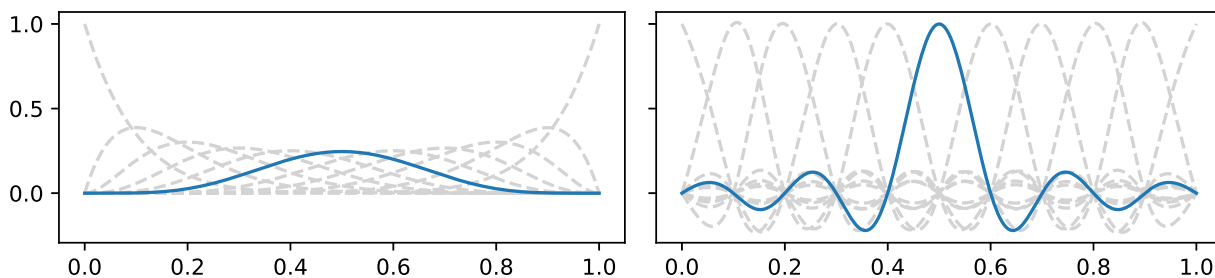


Figure 4.1: This show the normalised basis functions (left) $(B_i^n(t)w_i)/\sum_{j=0}^n B_j^n(t)w_j$, (right) $((-1)^i w_i/(t-t_i))/\sum_{j=0}^n (-1)^j w_j/(t-t_j)$ such that $w_0 = \dots = w_n = 1$.

is by displacing an interpolation point Q_i . Unlike the case for Bézier curves, the basis functions $\frac{(-1)^i \beta_i}{t-t_i} / \sum_{i=0}^n \frac{(-1)^i \beta_i}{t-t_i}$ are not nicely shaped like a bell (see Figure 4.1). Hence large displacements might induce an unpredictable effect on the curve. Therefore, to this point, it is only advised for micro-displacement.

A workflow where we can combine these two methods would be ideal. That is, we use the Bézier control points to design the general shape of the curve, and we use the interpolation points to enhance precision. A correspondence between the rational Bézier expression and its barycentric form would help us to achieve this goal. To describe the correspondence between the rational Bézier expression and its barycentric form, we first write the Bézier expression in barycentric form. One way to achieve that is to write the rational Bézier expression as a polynomial in homogeneous form [90], write this latter in first barycentric form, and finally, project using central projection. Another way is described in Section 4 following the fact that every rational function can be written in barycentric form [76]. These processes result in an equation relating the rational Bézier expression and its barycentric form. We let $\hat{P}_i = (\alpha_i P_i, \alpha_i)$ and $\hat{Q}_i = (z_i Q_i, z_i)$ where $z_i = (-1)^{n+i} \beta_i / w_i$, and w_i are the Lagrange weights (4.3). We have a relation $\hat{Q} = \mathbf{B}\hat{P}$, where

$$\mathbf{B} = \begin{pmatrix} B_0^n(t_0) & \cdots & B_n^n(t_0) \\ \vdots & \ddots & \vdots \\ B_0^n(t_n) & \cdots & B_n^n(t_n) \end{pmatrix}, \quad \hat{P} = \begin{pmatrix} \hat{P}_0 \\ \vdots \\ \hat{P}_n \end{pmatrix}, \quad \hat{Q} = \begin{pmatrix} \hat{Q}_0 \\ \vdots \\ \hat{Q}_n \end{pmatrix}. \quad (4.3)$$

The Bernstein–Vandermonde matrix \mathbf{B} in (4.3) is non-singular, because the Bernstein basis is a Chebyshev system. This guarantees that we can convert back and forth between the rational Bézier expression and its barycentric form, and so be able to exploit the advantages offered by them.

In the context of comparison of the rational Bézier expression and its barycentric form, it is known that a rational Bézier function can be written in standard form [91, 92], that is $\alpha_0 = \alpha_n = 1$. Similarly, we can write a barycentric interpolation in standard form, that is $\beta_0 = \beta_n = 1$. In fact, if

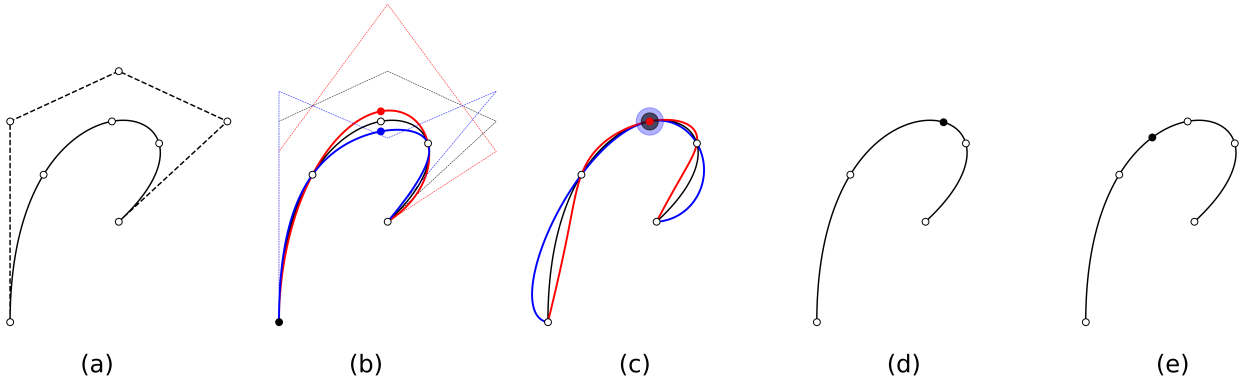


Figure 4.2: a) shows a Bézier curve defined by 5 interpolation points. b) shows the effect of displacing an interpolation point, while c) shows the effect of changing its weight. In d) we slide the black point and in e) we insert an additional point. Notice that the shape of the curve does not change under the last two operations.

we consider the function $\varphi(t)$, defined as

$$\varphi(t) = \frac{(1-\lambda)t}{\lambda(1-t) + (1-\lambda)t},$$

where

$$\lambda = \frac{\beta_n t_0 - \beta_0 t_n}{\beta_0(2t_n - 1) - \beta_n(2t_0 - 1)},$$

then it is stated in [93] that under the linear rational reparametrization $\varphi(t)$, a barycentric rational interpolation (4.1) can be written in standard form.

The barycentric form offers more possibilities of manipulating a curve. The most natural manipulation is by modifying the interpolation points. By modifying Q_k , we can recover the Bézier points as $\mathbf{P} = \mathbf{B}^{-1}\mathbf{Q}$ (see Figure 4.2.b). It can happen that $\hat{P}_k = (\bar{P}_k, \alpha_k)$ where $\alpha_k = 0$ for some k . In this case, we can introduce the so-called infinite control points [94], or we can apply degree elevation to the control polygon \mathbf{P} before applying the central projection. The manipulation of the interpolation points are recommended only for micro-editing since large tweaking may induce undesired effects (see Figure 4.1).

One way to manipulate a curve given in barycentric form (4.2) is by modifying the parameters t_i . Modifying a single parameter t_k can also induce an unpredictable effect on the curve. By constraining $t_k \in (t_{k-1}, t_{k+1})$ for $0 < k < n$, and by automatically updating Q_k and the weights β_i in [93], we present a method to modify a single t_k that results in sliding the interpolation point Q_k along the curve (see Figure 4.2.d). Each time we assign t_k to a new value \bar{t}_k , Q_k will be updated to $Q_k = P(\bar{t}_k)$, and the weights will be updated as

$$\beta_k = \sum_{i=0}^n (-1)^{n+k+i} \frac{\bar{t}_k - t_k}{\bar{t}_k - t_i} \beta_i, \quad \beta_i = \frac{t_i - t_k}{t_i - \bar{t}_k} \beta_i, \quad i \neq k.$$

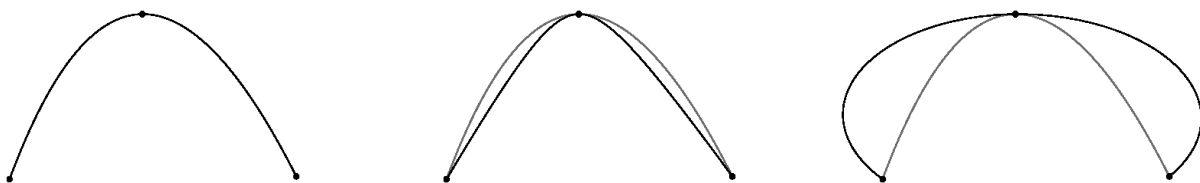


Figure 4.3: Effect of changing the central weight.

Another way to manipulate a curve given in barycentric form (4.2) is by modifying the weights. We deduce from (4.5) that the tangent at Q_k is given by

$$P'(t_k) = \frac{\sum_{i=0, i \neq k}^n (-1)^{k+i+1} \frac{\beta_i}{t_k - t_i} (Q_k - Q_i)}{\beta_k}.$$

We observe that only the denominator depends on β_k . It follows that modifying β_k adjusts the length of the tangent vector at Q_k . Geometrically, it means that modifying β_k adjusts the local flatness of the curve around Q_k (see Figure 4.3, Figure 4.2.c).

It is important to preserve the non-singularity of the curve. In order to conserve this non-singularity, we recall that it is necessary for all weights β_i to be strictly positive. In [93], we present a stronger restriction for the new value of β_k . It is shown that the curve remains non-singular as long as there is no t_\star such that $\beta_k = S_k(t_\star)$ where (see Figure 4.4)

$$S_k(t) = \sum_{i=0, i \neq k}^n (-1)^{k+i+1} \frac{t - t_k}{t - t_i} \beta_i.$$

This reduces the new possible value of β_k to be in the range $\beta_k \in (M_\star, M^\star)$, where

$$M_\star = \max\{\dots, M_{k-2}, M_k, M_{k+2}, \dots\}, \quad M^\star = \min\{\dots, M_{k-3}, M_{k-1}, M_{k+1}, M_{k+3}, \dots\},$$

with

$$M_{k+i} = \begin{cases} \max\{S_k(t) : t \in (t_{k+i+i_\star}, t_{k+i+i^\star})\}, & i \text{ even,} \\ \min\{S_k(t) : t \in (t_{k+i+i_\star}, t_{k+i+i^\star})\}, & i \text{ odd,} \end{cases} \quad i_\star = \begin{cases} -1, & i \leq 0, \\ 0, & i > 0, \end{cases} \quad i^\star = \begin{cases} 0, & i < 0, \\ 1, & i \geq 0. \end{cases}$$

To this point, we have explored some methods to manipulate a curve by tweaking the existing data of an interpolation point. One situation that might occur, however, is that a user wants to tweak a part of a curve where there is no control point. It would be advantageous if we can just insert a point on the curve without changing its shape (see Figure 4.2.e). In [93], we present a method to update all the data simultaneously to achieve this goal. Consider that we want to add

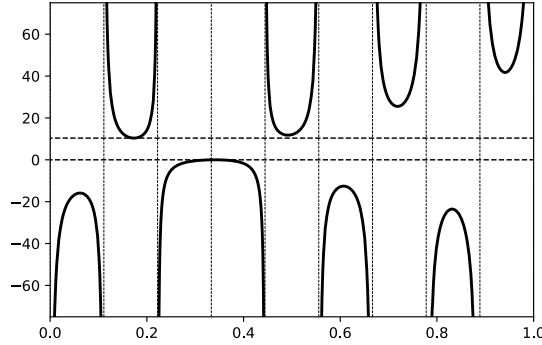


Figure 4.4: An example of the plot of the function $S_k(t)$ for $k = 3$. The horizontal lines represent M_* and M^* .

a new point Q_* at t_* , that is $Q_* = P(t_*)$. Suppose that $t_* \in (t_{k-1}, t_k)$, our new data is then given by

$$\check{t}_i = \begin{cases} t_i, \\ t_*, \\ t_{i-1}, \end{cases} \quad \check{Q}_i = \begin{cases} Q_i, \\ Q_* = P(t_*), \\ Q_{i-1}, \end{cases} \quad \check{\beta}_i = \begin{cases} \frac{\beta_i}{t_* - t_i}, \\ \sum_{i=0}^n (-1)^{n+k+i} \frac{\beta_i}{t_i - t_*}, \\ \frac{\beta_{i-1}}{t_{i-1} - t_*}, \end{cases} \quad \text{if } \begin{cases} i < k, \\ i = k, \\ i > k. \end{cases}$$

We also observe an advantage of using the barycentric form for the evaluation of a rational Bézier curve (4.1). The classical de Casteljau algorithm would evaluate (4.1) in $O(n^2)$ [95]. [96] developed a linear-time algorithm to evaluate (4.1). We also propose to convert it into a linear form, in this case, linear barycentric form. In the first studies of the barycentric form, [71], [97], and [98] studied the efficiency of the evaluation of the barycentric form (4.1). They described the form (4.1) as rapid as it only involves a “cumulative” multiplication and a division. Thus we propose to convert the rational Bézier expression to barycentric form and evaluate this later. This conversion can be done efficiently and accurately in $O(n^2)$ [99]. We compared the evaluation time using the classical de Casteljau algorithm, the linear-time algorithm, and our proposed method by implementing them in C (see Figure 4.5).

4.4 Application in periodic curve design

In this chapter we follow the same idea as in the open curve design. A trigonometric rational Bézier form is introduced and studied in a sequel [100, 101, 102, 103, 104]. We consider the shape control tool given by the rational Bézier setting, and we create complementary tools to have more intuitive and direct control over a curve.

We recall that a periodic Bézier curve is defined by an odd number of points P_0, \dots, P_n such that

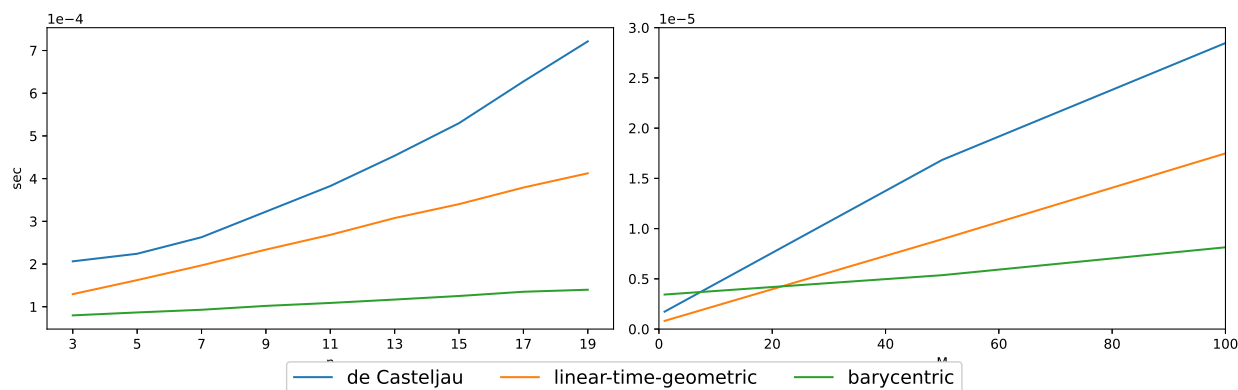


Figure 4.5: Runtime of some algorithms for computing a rational Bézier curve with $P_i = 100i\binom{1}{1} + \binom{1}{1}$ and $\alpha_i = i \bmod 2 + 1$. (left) We first consider $M = 2500$ evaluation points for $n = 3, 5, 7, \dots, 19$. (right) we fix the degree at $n = 20$ and vary the number of evaluation points $M = 1, 50, 100$.

$n = 2N$ for some positive integer N , and is given by

$$P(t) = \sum_{i=0}^n B_n(t - \phi_i) P_i$$

where $B_n(t) = K_n \cos^n \frac{t}{2}$, $K_n = \frac{2^n}{n+1} \binom{n}{N}^{-1}$, $\phi_i = \frac{2i\pi}{n+1}$ for $i = 0, \dots, n$. Furthermore, by associating scalars $\alpha_0, \dots, \alpha_n$ to the points respectively, we define a rational periodic Bézier curve as

$$P(t) = \frac{\sum_{i=0}^n B_n(t - \phi_i) \alpha_i P_i}{\sum_{i=0}^n B_n(t - \phi_i) \alpha_i}. \quad (4.1)$$

Given this restriction on the number of control points, the corresponding barycentric interpolating form that we investigate needs to have the same restriction. Hence the analogue of (4.2), for a given sequence of control points Q_0, \dots, Q_n and their associated weights β_0, \dots, β_n , is given by

$$P(t) = \frac{\sum_{i=0}^n (-1)^i \csc \frac{t-t_i}{2} \beta_i Q_i}{\sum_{i=0}^n (-1)^i \csc \frac{t-t_i}{2} \beta_i}, \quad (4.2)$$

where $\csc(t) = 1/\sin(t)$. The shape controls arise from the manipulation of the degrees of freedom of this formula (the control points Q_k , the weights β_k , and the nodes t_k). As for the classical case, the basis of the trigonometric barycentric rational interpolation is not nicely shaped as the basis of the periodic Bézier form, hence, the recommended manipulation is only for micro-editing.

To achieve the combination of the two methods, we explore the relation between the periodic Bézier form (4.1) and trigonometric interpolating form (4.2). We can write (4.1) in barycentric form (4.2), by writing its numerator and its denominator in the first barycentric form, by writing the ration between the resulting expression and canceling some common factors. These steps can be written more compactly as follow. Denote by $z(t)$ the denominator of (4.1). We let $z_i = z(t_i)$,

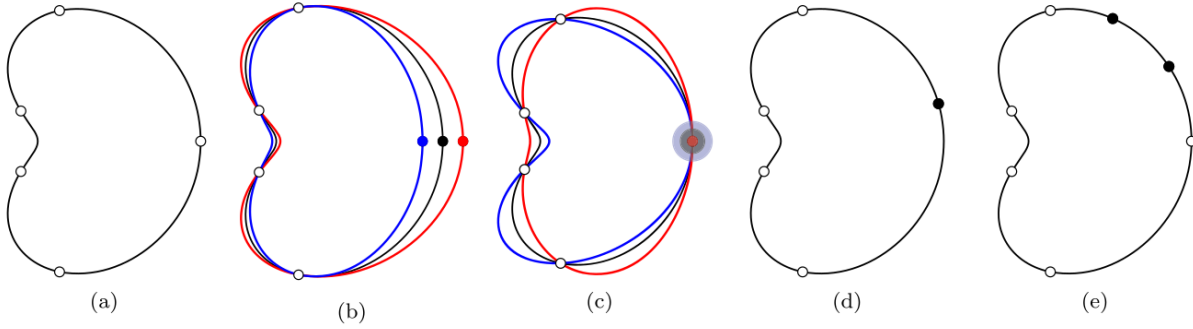


Figure 4.6: a) shows a periodic Bézier curve defined by 5 interpolation points. b) shows the effect of displacing an interpolation point, while c) shows the effect of changing its weight. In d) we slide the black point and in e) we insert additional points. Notice that the shape of the curve does not change under the last two operations.

$\hat{P}_i = (\alpha_i P_i, \alpha_i)$ and $\hat{Q}_i = (z_i Q_i, z_i)$ for $i = 0, \dots, n$. And we have the relation $\hat{Q} = \mathbf{B}\hat{P}$, where

$$\mathbf{B} = \begin{pmatrix} B_n(t_0 - \phi_0) & \cdots & B_n(t_0 - \phi_n) \\ \vdots & \ddots & \vdots \\ B_n(t_n - \phi_0) & \cdots & B_n(t_n - \phi_n) \end{pmatrix}, \quad \hat{P} = \begin{pmatrix} \hat{P}_0 \\ \vdots \\ \hat{P}_n \end{pmatrix}, \quad \hat{Q} = \begin{pmatrix} \hat{Q}_0 \\ \vdots \\ \hat{Q}_n \end{pmatrix}.$$

Since the radial basis $\{B_n(t - \phi_k)\}_k$ forms a Chebyshev system, the matrix \mathbf{B} is non-singular. This guarantees us that we can convert back-and-forth between the Bézier form (4.1) and the barycentric form (4.2). The weight β_k is then given by $\beta_k = (-1)^k w_k z_k$. Hence we can analyse the effect of manipulating the control points, the weights, and the nodes.

The interpolatory property of the form (4.2) induces a snapping tool, which means we can force a curve to pass through a certain set of points. However, the trigonometric basis is not as nicely shaped as the trigonometric Bernstein-like basis. Hence large displacement are not advised. We can see the effect of displacing an interpolation point in Figure 4.6.b).

In addition, [86] show that the tangent of a curve at a point Q_k is given by

$$P'(t_k) = \frac{1}{2\beta_k} \sum_{i=0, i \neq k}^n (-1)^{k+i+1} \csc \frac{t_k - t_i}{2} \beta_i (Q_k - Q_i).$$

We observe that the term in the summation, which determines the direction of the tangent, does not depend on β_k . Since β_k only appear in the denominator, it influences naturally the strength of the tangent at Q_k . Geometrically, by decreasing the value of β_k , the curve is flatten, while by increasing the value of β_k , the curve bends more tightly around Q_k .

The advantage of the interpolation method is that we can manipulate the curve by manipulating the interpolation points. It is possible that we want to rectify a part where there is no interpolation point. [86] devise some formula to update the data of the barycentric form such that we can slide an interpolation point Q_k , that is change t_k to a new value $\tilde{t}_k \in (t_{k-1}, t_{k+1})$. We have $\tilde{Q}_k = Q(\tilde{t}_k)$

$$\tilde{\beta}_k = \sin \frac{\tilde{t}_k - t_k}{2} \sum_{j=0}^n (-1)^{j+k} \csc \frac{\tilde{t}_k - t_j}{2} \beta_j, \quad \tilde{\beta}_i = \csc \frac{t_i - \tilde{t}_k}{2} \sin \frac{t_i - t_k}{2} \beta_i, \quad i \neq k.$$

Alternatively, we also develop some formulas that update the data of the barycentric form, so that we can insert additional interpolation points leaving the shape of the curve unchanged. Assume that we insert two new parameters $\tilde{t}_k, \tilde{t}_{k+1} \in (t_{k-1}, t_k)$. We have

$$\tilde{t}_i = \begin{cases} t_i, \\ \tilde{t}_k, \\ \tilde{t}_{k+1}, \\ t_{i-2}, \end{cases} \quad \tilde{Q}_i = \begin{cases} Q_i, \\ Q(\tilde{t}_k), \\ Q(\tilde{t}_{k+1}), \\ Q_{i-2}, \end{cases} \quad \tilde{\beta}_i = \begin{cases} \csc \frac{t_i - \tilde{t}_k}{2} \csc \frac{t_i - \tilde{t}_{k+1}}{2} \beta_i, \\ \csc \frac{\tilde{t}_k - \tilde{t}_{k+1}}{2} \sum_{j=0}^n (-1)^{j+k} \csc \frac{\tilde{t}_k - t_j}{2} \beta_j, \\ \csc \frac{\tilde{t}_{k+1} - \tilde{t}_k}{2} \sum_{j=0}^n (-1)^{j+k+1} \csc \frac{\tilde{t}_{k+1} - t_j}{2} \beta_j, \\ \csc \frac{t_{i-2} - \tilde{t}_k}{2} \csc \frac{t_{i-2} - \tilde{t}_{k+1}}{2} \beta_{i-2} \end{cases} \quad \text{if } \begin{cases} i < k, \\ i = k, \\ i = k + 1, \\ i > k + 1. \end{cases}$$

Lastly, since we are now able to slide points, that is, we change the parameter values. [86] analyse the limit case where two interpolation points collapse into one interpolation point. Fortunately, the shape of the curve is preserved. Assume that we have double node at t_{k+1} , the formula (4.2) turns into the form of a Hermite rational interpolation as

$$Q(t) = \frac{\sum_{i=0, i \neq k}^n (-1)^i \csc \frac{t - t_i}{2} \hat{\beta}_i Q_i + (-1)^{k+1} \csc \frac{t - t_{k+1}}{2} \cot \frac{t - t_{k+1}}{2} \hat{\beta}'_{k+1} \left(Q_{k+1} + 2 \tan \frac{t - t_{k+1}}{2} Q'_{k+1} \right)}{\sum_{i=0, i \neq k}^n (-1)^i \csc \frac{t - t_i}{2} \hat{\beta}_i + (-1)^{k+1} \csc \frac{t - t_{k+1}}{2} \cot \frac{t - t_{k+1}}{2} \hat{\beta}'_{k+1}},$$

where

$$\hat{\beta}_i = \csc \frac{t_i - t_{k+1}}{2} \sin \frac{t_i - t_k}{2} \beta_i, \quad i \neq k, k+1, \\ \hat{\beta}_{k+1} = \sin \frac{t_{k+1} - t_k}{2} \sum_{i=0, i \neq k+1}^n (-1)^{k+1+i} \csc \frac{t_{k+1} - t_i}{2} \beta_i + \cos \frac{t_{k+1} - t_k}{2} \beta_{k+1}, \quad \hat{\beta}'_{k+1} = \sin \frac{t_{k+1} - t_k}{2} \beta_{k+1},$$

and $Q'_{k+1} = Q'(t_{k+1})$.

For consistency, it is preferable that the number of interpolation points and the number of Bézier points are the same. To this end, [86] devise a framework that increase the number of Bézier control points by 2 while preserving the shape of the curve. Indeed, as for the classical counterpart [87], a periodic rational Bézier curve can be considered as a central projection Π of a spatial periodic Bézier curve defined by $\hat{P}_0, \dots, \hat{P}_n$. The control points of the degree-raised spatial periodic Bézier curve are given by $\hat{P}_{N+1} = \mathbf{C}^{-1} \mathbf{D} \hat{P}_N$ where $\mathbf{C} \in \mathbb{R}^{(n+3) \times (n+3)}$ and $\mathbf{D} \in \mathbb{R}^{(n+3) \times (n+1)}$ are

$$\mathbf{C}_{i,j} = B_{n+2}(\psi_i - \psi_j), \quad \mathbf{D}_{j,k} = B_n(\psi_j - \phi_k), \quad i, j = 0, \dots, n+2, \quad k = 0, \dots, n,$$

with $\psi_i = \frac{2i\pi}{n+3}$, $i = 0, \dots, n+2$ and $\phi_i = \frac{2i\pi}{n+1}$, $i = 0, \dots, n$. The degree of (4.1) can be raised from N to

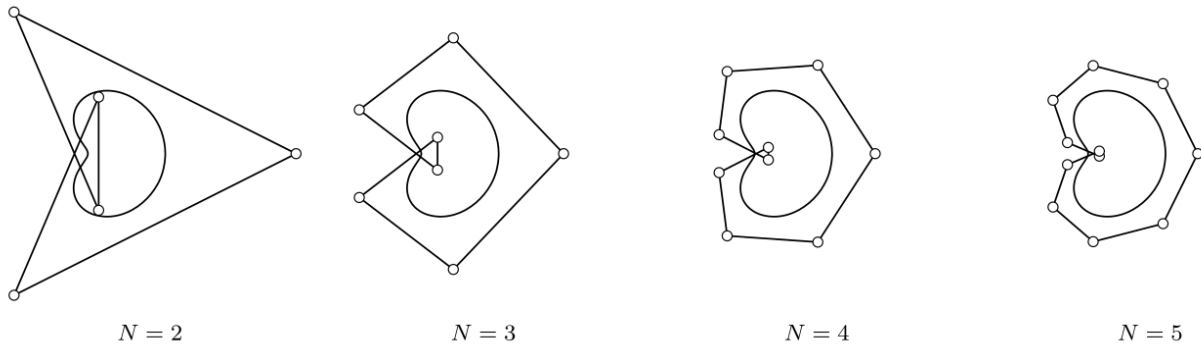


Figure 4.7: Raising the degree (from left to right) of a periodic rational Bézier curve.

$N + 1$ (see Figure 4.7) in three steps

$$\begin{array}{ccc}
 P_N(t) & & P_{N+1}(t) \\
 \Pi^{-1} \downarrow & & \uparrow \Pi \\
 \hat{P}_N(t) & \xrightarrow[\mathbf{C}^{-1}\mathbf{D}]{\text{multiply with}} & \hat{P}_{N+1}(t)
 \end{array}$$

Chapter 5

Krunal Raval: Non-uniform degree and Tchebycheffian spline technologies for adaptive isogeometric analysis

Isogeometric Analysis (IgA) [105] emerged about two decades ago as a methodology aimed at streamlining the interactions between geometric modeling and numerical simulation. It sought to address significant discrepancies between Computer-Aided Design (CAD) packages, used for modeling geometries, and Finite Element Analysis (FEA) software developed in a complete independent way for numerical simulation. Indeed, traditionally geometric modeling relied on highly smooth piecewise polynomial functions known as splines, whereas numerical simulations utilized C^0 finite elements. By bringing FEA and CAD together, IgA enhanced efficiency in solving complex engineering problems, providing a truly integrated design-through-analysis process. The idea behind IgA revolves around employing an isoparametric approach, utilizing the same kind of functions both to represent the geometry of the domain and to approximate the solution of the differential problem defined on it. This approach yields several advantages over classical FEA:

- Enhanced accuracy in describing the geometry: It provides a better approximation of complex geometries, and even achieves exact representation in the case of conic sections; leading to increased precision in analysis.
- Streamlined refinement process: With IgA, the description of geometry is exact (or well approximated) at the coarsest mesh level and it does not change with refinement; eliminating the necessity of communication with CAD systems during refinement.
- Higher accuracy per degree of freedom: For a given degree, spline functions of high and low smoothness provide spaces with the same approximation order but of different dimensions: with a clear gain for the former ones, [106]. This rejuvenated the study of higher order

methods, proving to be a superior alternative in various applications; see [107], and references therein.

These advantages have attracted significant interest in the isogeometric paradigm across various applications, see [108], validating the claim that IgA is a viable alternative to standard, polynomial based finite element analysis, [107].

The initial classical formulation of IgA is based on tensor-product B-splines and their rational extension (NURBS), [109]. As a consequence, it inherited some drawbacks and limitations of these tools. For instance, NURBS are a legacy of CAD industry where they are of main interest because they allow for exact representation of conic sections. Still NURBS lack an exact description of transcendental curves of interest in application, and their parametrization of conic sections does not correspond to –and it is often far from– the arc length. In addition, they are defined in a rational form along with “weights”, resulting into a very poor behavior under differentiation and integration, which are key operations in analysis.

However, the properties of B-splines and NURBS which are crucial for design (positivity, partition of unity, compact support,...) are not confined to the polynomial setting, and can be extended in an elegant way beyond piecewise polynomial and/or rational functions.

In this thesis, we explore the use in IgA of a generalization of the polynomial splines, called Tchebycheffian splines, both to overcome the drawbacks of the rational model and to exploit their superior description of the fundamental solutions of some differential operators.

Another limitation in the primitive formulation of IgA pertains to local refinement in the general multivariate setting and it is due to the tensor-product structures, typically adopted to generate multivariate splines from univariate spline basis functions. Tensor-product spline spaces can be refined through knot insertion and/or degree elevation in each coordinate direction separately. However, this often results in excessive refinement in areas that do not require it due to the global nature of the refinement process. This lack of locality in refinement of tensor-product splines is visualized in Figure 5.1.

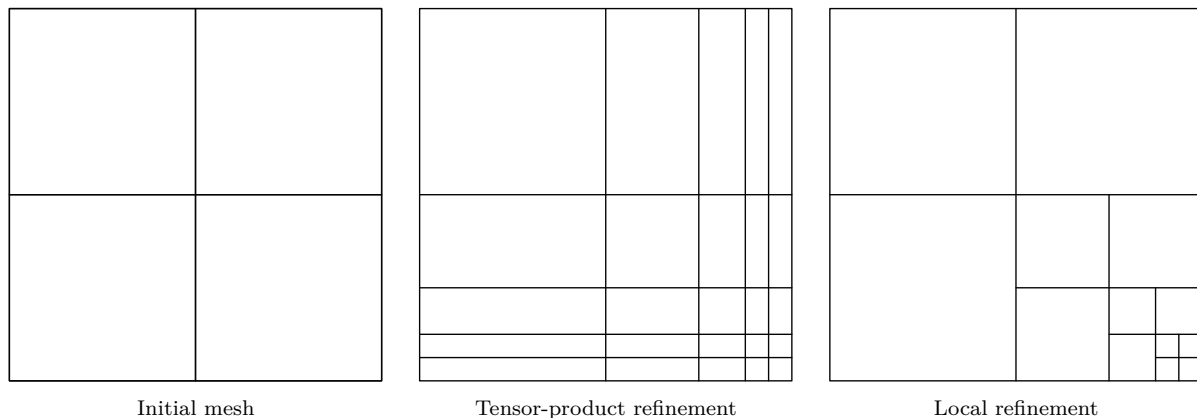


Figure 5.1: Lack of local refinement in tensor-product splines.

To address the limitations associated with local refinement, various strategies have been devised. Here we focus on local tensor-product structures and, in particular, on Locally Refined (LR) meshes and on the construction of different spline spaces over such meshes. Leveraging the structural similarities between standard polynomial B-splines and Tchebycheffian B-splines, we can construct Tchebycheffian spline spaces over LR meshes, combining the benefits inherent to Tchebycheffian spline setting and to local tensor-product structures. This is the second line of investigation presented in this thesis.

Furthermore, when tackling higher-order problems with localized features, integrating higher degree B-splines into the solution space can be advantageous. However, incorporating local degree elevation in B-splines presents a challenge, as uniformly higher degree splines throughout the domain may be unnecessary. In such scenarios, LR meshes with non-uniform degree B-splines, constructed through a combination of local \underline{h} - and \underline{p} -refinement, could offer a more efficient solution. This is a third line of investigation carried out in this thesis.

We shortly summarize the state of the art in the fields of (1) Tchebycheffian (B-)splines as a generalization of the standard polynomial (B-)splines and (2) spline constructions on unstructured meshes with local tensor-product structure for adaptive refinement, and we sum up our contributions on the use of Tchebycheffian B-splines and non-uniform degree splines in isogeometric (adaptive) Galerkin methods.

Tchebycheffian B-splines

Spline functions are ubiquitous in numerical methods. Besides their theoretical interest, they have application in several branches of the sciences including geometric modeling, signal processing, data analysis, visualization and numerical simulation just to mention a few. Splines, in the broad sense of the term, are functions consisting of pieces of smooth functions glued together in a certain smooth way. There is a large variety of spline species, often referred to as the zoo of splines. The most popular species is the one where the pieces are algebraic polynomials of a given degree p and inter-smoothness is imposed by means of equality of derivatives up to a certain order. Their popularity can be mainly attributed to their representation in terms of the so-called B-splines. B-splines enjoy properties such as local linear independence, minimal support, non-negativity, and partition of unity; they can be computed through a stable recurrence relation and can even be seen as the geometrically optimal basis for piecewise polynomial spaces.

Tchebycheffian splines are smooth piecewise functions whose pieces are drawn from (possibly different) Extended Tchebycheff (ET-) spaces which are natural generalizations of algebraic polynomial spaces [110, 111]. Any non-trivial element of an ET-space of dimension $p + 1$ has at most p zeros counting multiplicity. We will refer to p as the degree, in analogy with the polynomial case. Extended Complete Tchebycheff (ECT-) spaces are an important subclass that can be generated

through a set of positive weight functions [112, 113] and are spanned by generalized power functions [111]; the latter are the natural extension of the monomial basis functions for algebraic polynomials; see [114, Section 2.1]. Relevant examples are null-spaces of linear differential operators on suitable intervals [113]; see [114, Section 2.2]. On bounded and closed intervals the concepts of ET-space and ECT-space coincide [115], so from now on we will focus on ECT-spaces.

Most of the results known for splines in the polynomial case extend in a natural way to the Tchebycheffian setting. In particular, under suitable assumptions on the involved ECT-spaces, Tchebycheffian splines admit a representation in terms of basis functions, called Tchebycheffian B-splines (TB-splines), with similar properties to polynomial B-splines. TB-splines were introduced in 1968 by Karlin [116] using generalized divided differences. There are several other ways to define them, including Hermite interpolation [117], de Boor-like recurrence relations [118], integral recurrence relations [119], and blossoming [120]; see also the historical notes in [113, Chapters 9 and 11] for further details. Each of these definitions has advantages according to the properties to be proved and lead to the same functions, up to a proper scaling.

Multivariate extensions of Tchebycheffian splines can be easily obtained by means of tensor-product structures.

Due to the richness of ECT-spaces, Tchebycheffian splines can find applications in several contexts including data approximation/interpolation, geometric modeling and numerical simulation; see [111] and references therein. Thanks to their structural similarities, TB-splines are theoretically plug-to-plug compatible with classical polynomial B-splines, so they can be potentially easily incorporated in any software library supporting polynomial B-splines to enrich its capability.

Unfortunately, despite their theoretical interest and applicative potential, TB-splines have not gained much attention in practice so far. The reason behind this is that TB-splines are generally difficult to compute. The classical approaches mentioned above, based on generalized divided differences, Hermite interpolation, or repeated integration, are computationally expensive and/or numerically unstable. An important step forward was recently made in [121] where the authors proposed a strategy that represents TB-splines as linear combinations of local Tchebycheffian Bernstein functions through a suitable extraction operator. The local Tchebycheffian Bernstein functions form a basis of the local ECT-spaces involved in the definition of the TB-splines. In the polynomial case, these are nothing but the classical Bernstein polynomial basis functions. Following the approach in [121], an object-oriented Matlab toolbox has been developed in [122] for the construction and manipulation of TB-splines whenever they exist. The toolbox supports TB-splines whose pieces belong to ECT-spaces that are null-spaces of constant-coefficient linear differential operators, and is publicly available. Note that both [121] and [122] address the more general setting of Multi-Degree TB-splines (MDTB-splines) where the local ECT-spaces of the Tchebycheffian spline space are not required to be of the same dimension. Having at our disposal a publicly available implementation for such a large class of TB-splines paves the path for their effective use in practical applications. An alternative strategy has been proposed in [123] but no implementation is available.

As mentioned before, the advantages of the isogeometric paradigm are not a distinguishing property of B-splines/NURBS, and B-splines/NURBS are not a requisite ingredient in IgA. An interesting subclass of TB-splines, the so-called Generalized B-splines (GB-splines), has been proposed as an alternative to classical polynomial and rational splines in the context of IgA; see [124] and references therein. GB-splines can be seen as the minimal extension of polynomial B-splines towards the wide variety of TB-splines: their pieces belong to ECT-spaces obtained by enriching an algebraic polynomial space with a pair of functions, typically hyperbolic or trigonometric functions identified by a single shape parameter. GB-splines overcome two issues of the rational NURBS model: they allow for an (almost) exact arc-length parameterization of conic sections and behave with respect to differentiation and integration as nicely as polynomial B-splines (for instance, the derivative of a trigonometric generalized spline with a given phase parameter β and degree p is again a trigonometric generalized spline with the same shape parameter β and degree $p - 1$). These properties make GB-splines an interesting tool to face both geometrical and analytical hurdles in IgA. The effectiveness of hyperbolic or trigonometric GB-splines in isogeometric Galerkin and collocation methods has been illustrated in a sequence of papers, where their properties have been exploited to obtain exact representations of common geometries [125] or to beneficially deal with advection-dominated problems [126]; see also [127] for the related spectral properties.

Being a so minimal extension of the polynomial setting, however, GB-splines are not always flexible enough for practical applications. In particular, in a tensor-product GB-spline space only two hyperbolic or trigonometric functions identified by the same shape parameter are added to polynomials along each parametric direction. Therefore, a given tensor-product GB-spline space does not allow for:

- an exact representation of different arcs of conic sections at opposite sides;
- a proper treatment of different analytic features (like sharp layers) along a given parametric direction;
- a simultaneous treatment of geometrical and analytical features along the same parametric direction.

Tchebycheffian splines with pieces belonging to ECT-spaces that are null-spaces of constant-coefficient linear differential operators are an extension of hyperbolic and trigonometric generalized polynomial splines. They enjoy all the geometrical and analytical features that motivate the interest in GB-splines without suffering from the above mentioned restrictions. They grant the freedom of combining polynomials with exponential and trigonometric functions with any number of individual shape parameters. For this class of Tchebycheffian splines, when the various pieces are drawn from a single ECT-space which contains constants, the existence of TB-splines is always ensured, possibly with some restriction on the partition; see [114] for more details. Furthermore, they are supported by the Matlab toolbox available in [122]. In summary, TB-splines with pieces belonging

to ECT-spaces that are null-spaces of constant-coefficient linear differential operators offer a suitable balance between the immense variety of ECT-spaces and the practical needs of a problem-driven space selection and efficient evaluation procedures for the space elements.

In the first part of this thesis we investigate the use of tensor-product TB-splines, whose pieces belong to ECT-spaces that are null-spaces of constant-coefficient linear differential operators, in Galerkin isogeometric methods. It turns out that such a class of TB-splines provides a powerful and flexible environment for the IgA paradigm, beyond the limits of the rational NURBS model.

Splines over T-meshes

Tensor-product B-splines are probably the most well-known multivariate spline basis functions. They have been profitably applied in different contexts including geometric modeling, approximation theory, and numerical simulation. Their popularity roots in their simple, elegant and efficient construction: they are nothing but tensor-products of univariate B-splines; see, e.g., [128] and references therein.

The tensor structure of the underlying mesh, however, is the major weakness of tensor-product B-splines as it hinders adequate local refinement, forcing the use of unnecessarily large discrete spaces and leading to a significant loss in efficiency. This has been seen as a severe limitation in the context of isogeometric analysis.

To overcome this limitation, in the last decades many alternative spline technologies have been developed for so-called T-meshes. Such meshes are still axis-aligned but T-vertices are allowed in the interior of the domain, in order to support local refinement, while preserving locally the simplicity of the tensor approach; see [129] and references therein. T-splines [130], (truncated) hierarchical B-splines [131], and locally refined B-splines [132] are popular examples of such spline technologies. All these approaches have their own strengths (and weaknesses) depending on the context they are intended to be used. T-splines were introduced by Sederberg [133, 130] to enable local refinement of spline surfaces and handle complex topology, by treatment of so called extraordinary points in the context of CAD. Initially, T-splines were limited to bi-cubic polynomial degree and C^2 continuity between elements. However, the technology has since been extended to arbitrary uniform polynomial degrees [134] and supports mixed continuity [135]. Various applications of T-splines in isogeometric analysis have been reported [136]. To ensure nested spaces and maintain linear independence, a restricted class of analysis-suitable T-splines was introduced [137]. Subsequently, these were characterized as dual-compatible T-splines [138]. Several local refinement algorithms for analysis-suitable T-splines with optimal convergence have been proposed in [139].

Hierarchical B-splines were introduced in 1988 for local h -refinement in geometric modeling [140]. The construction of the hierarchical B-splines guarantees nested spaces and linear independence of the basis functions [141]. The application of hierarchical constructions in isogeometric analysis has

been very promising [142]. However, the standard hierarchical construction does not preserve the partition of unity property. To address this limitation, a new set of hierarchical basis functions, called Truncated Hierarchical B-splines (THB-splines), were introduced [131]. THB-splines, defined as a linear combinations of refined B-splines, form a convex partition of unity, exhibit good stability and approximation properties [143]. By providing a way to define an adaptive extension of the B-spline framework which is also suitable for geometric modeling applications, THB-splines satisfy both the demands of adaptive numerical simulation and geometric design, making them well suited for isogeometric analysis; see [144]. A comprehensive study has been conducted to develop a posteriori error estimators for devising optimal refinement strategies with hierarchical B-splines; see [145].

The definition of LR B-splines is inspired by the knot insertion refinement process of univariate (and tensor-product) B-splines. Their formulation bears a large similarity to classical tensor-product B-splines and this makes them one of the most elegant extensions of univariate B-splines towards T-meshes. Since their introduction in [132], LR B-splines have found interesting applications in several contexts ranging from data approximation [146] to numerical simulations [147], also considering their rational version [148]. More theoretical aspects, mainly related to the issue of linear and local linear independence and related adaptive refinement strategies, have been investigated in [149]. A comparison between LR B-splines, hierarchical, and truncated hierarchical B-splines can be found in [150], while combinations of the LR B-spline framework with the hierarchical approach have been explored in [151].

LR B-spline refinement starts from a multivariate tensor-product spline space spanned by tensor-product B-splines. The refinement is performed successively by splitting individual tensor-product B-splines in one parameter direction at a time. In the bivariate case the splitting is done by introducing an axis parallel line segment with multiplicity. The line segment must split the support of at least one tensor-product B-spline into two disjoint parts. Very often the line segment also splits the support of other tensor-product B-splines. The refinement process is continued until all remaining B-splines have a certain minimal support property as defined in [149, 132].

Figure 5.2 depicts examples of a tensor-mesh, a T-mesh that is not an LR-mesh, a T-mesh that is also an LR-mesh, and a mesh that is neither a T-mesh nor an LR-mesh. While T-splines have a straightforward construction only for odd degrees, hierarchical B-splines and LR B-splines offer independence from the polynomial degree. Moreover, both the definition and implementation of hierarchical B-splines and LR B-splines can be made dimension-independent.

The notion of mixing multiple degrees in (univariate) spline constructions was introduced first in the context of approximation [152] and later rejuvenated in geometric modelling [153]. Such an extension of uniform degree B-splines, allowing for spaces of different dimensions in different intervals glued together with certain smoothness, are called Multi-Degree B-splines (MDB-splines). More recently, an efficient algorithm for the computation of MDB-splines was proposed in [154] and further developed in [155]. The potential of the multi-degree spline spaces was also actualized

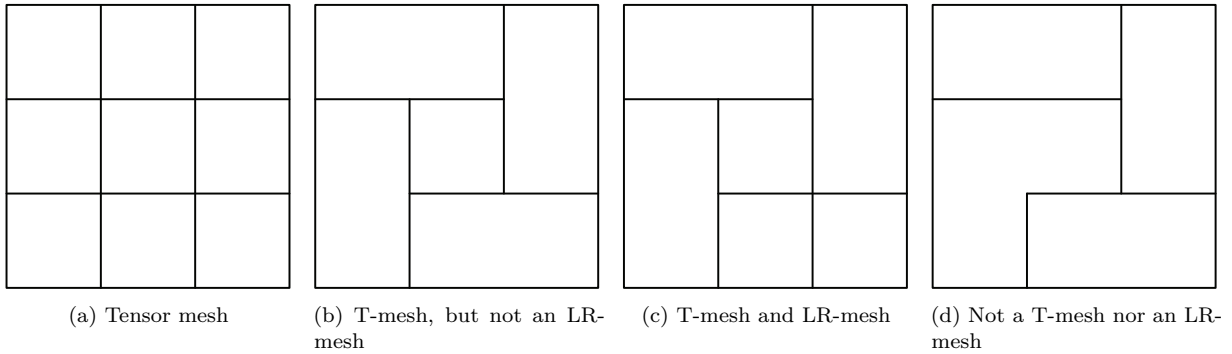


Figure 5.2: The T-mesh in (b) can not be created from single line insertions starting from a tensor mesh. The T-mesh in (c) can, and is therefore also a valid LR-mesh.

in applications [156]. Some alternative constructions to the multi-degree B-splines can be found in [157].

The construction of multivariate spline generalizations combining locally different degrees, with the same flexibility as MDB-splines, is a hard and ambitious task. Already in the early work on hierarchical splines it was noted that different degrees could be mixed in the (truncated) hierarchical framework; see [158, Remark 4]. A full extension of the (truncated) hierarchical setting towards local mesh refinement and degree elevation is described in [159]. Even though not supported by the standard definition of T-splines, there has also been some significant progress in that field, with their applications in collocation [134] and thin shell analysis [160]. Additionally, the study of the effectiveness of \underline{k} -refinement [161] and its comparison with \underline{p} -refinement [162] in isogeometric analysis has garnered considerable attention for its capability to address high-order differential problems accompanied by \underline{hp} -refined meshes.

In the context of LR splines, the construction of non-uniform degrees utilizes the degree elevation operation or \underline{p} -refinement, in which the degree of a spline is increased, while preserving its smoothness [163]. With the local tensor-product structure of LR B-splines, it becomes feasible to perform degree elevation on an individual B-spline without introducing new degrees of freedom along an entire hyper-plane in the parameter domain. This capability allows for local degree elevation, enabling the construction of a solution space with non-uniform degrees, which proves advantageous in addressing higher-order isogeometric methods involving local features. Moreover, due to the two-scale relation, low degree functions are consistently replaced with new higher degree ones, ensuring the nestedness of the space.

In the second part of this thesis, we introduce non-uniform degree LR-splines that allow local adaptive \underline{h} -refinement as well as local \underline{p} - and \underline{hp} -refinement, by combining local knot insertion with local degree elevation, and we discuss their application in isogeometric analysis.

Tchebycheffian splines over T-meshes

As already mentioned, most of the results known for univariate polynomial splines extend in a natural way to the Tchebycheffian setting.

As illustrated in the analysis presented in [114], tensor-product TB-splines are able to outperform tensor-product polynomial B-splines in isogeometric Galerkin methods whenever appropriate problem-driven selection strategies for the underlying ECT-spaces are applied. Nevertheless, tensor-product TB-splines present the same drawbacks of any tensor-product structure when local refinement is of need.

The structural similarity between ECT-spaces and algebraic polynomial spaces enables us to extend popular local refinement technologies, based on local tensor products, towards the Tchebycheffian setting. Tchebycheffian spline spaces over T-meshes have been introduced in their full generality in [164]. The structure of ECT-spaces has been exploited in [165] to fully extend the dimension study carried out in the polynomial case in [166]. Some earlier generalizations of the polynomial setting towards particular Tchebycheffian spline spaces or peculiar T-meshes have been considered in [167]. In particular, [168] outlines the construction of GB-splines on LR-meshes while hierarchical GB-splines have been presented in [169].

In the third line of investigation in this thesis, we define LR TB-splines as a generalization of LR B-splines and we analyze their performance in the context of adaptive isogeometric Galerkin methods.

The definition of LR TB-splines is driven by the knot insertion refinement process of tensor-product TB-splines, in complete analogy to the polynomial setting. In the bivariate tensor case, inserting a new knot in a pair of (global) knot vectors results in inserting a line segment in the mesh crossing the entire domain, thus refining all the TB-splines whose supports are crossed. On the contrary, LR TB-splines are defined on local knot vectors, and consequently the insertion of a new knot is always performed with respect to a particular LR TB-spline and results in refining only few basis functions.

The theoretical construction of LR TB-splines is independent of the particular ECT-spaces where the various pieces are drawn from. However, in the applicative context we confine ourselves to Tchebycheffian splines identified by ECT-spaces that are null-spaces of constant-coefficient linear differential operators containing constants because, as already mentioned:

- they already grant the freedom of combining algebraic polynomials with exponential and trigonometric functions with any number of individual shape parameters;
- when the various pieces are drawn from a single ECT-space, the existence of TB-splines is always ensured, possibly with some restriction on the partition; see [114, Section 2.4];
- the corresponding TB-splines are supported by the Matlab toolbox available in [122].

Contributions

To summarize, this dissertation has explored a diverse set of extensions of the theory and application of isogeometric analysis.

- The idea of Tchebycheffian splines as a generalization of classical algebraic polynomial splines was investigated and a problem dependent strategy for solving differential problems through isogeometric analysis was devised to design a better suited solution space based on the problem at hand.
- The idea of splines on unstructured meshes, and in particular on LR-meshes, has been investigated in two directions: non-uniform degrees and Tchebycheffian spline spaces.

As for the generalization of splines beyond the polynomial setting, we have extensively considered the subclass of Tchebycheffian splines whose pieces are drawn from a single ECT-space which is the null space of a differential operator with real constant coefficients, and includes constants. This subclass of Tchebycheffian splines already provides a large variety of combinations of polynomial, exponential, and trigonometric functions equipped with a wide spectrum of shape parameters and can be represented in terms of a B-spline like basis (TB-splines) at least for sufficiently fine partitions. It turns out that, such TB-splines can outperform polynomial B-splines whenever appropriate problem-driven selection strategies for the underlying ECT-spaces are applied. In particular, they can be beneficial both from the geometrical and the analytical point of view offering a perfect fit to the isogeometric approach; see [114].

Furthermore, the TB-splines on LR-meshes are a viable spline technology on unstructured meshes and offer a valid alternative to classical LR B-splines in adaptive isogeometric analysis. We have shown that the adaptive strategy combined with problem-oriented approximation spaces may create a synergistic effect and may produce results of similar quality with less levels of refinement, and so fewer degrees of freedom, compared to the polynomial setting; see [170].

Finally, the potential of non-uniform degrees in LR B-splines was realized using local \underline{h} -, \underline{p} -, and \underline{hp} -adaptive refinement, and was investigated in a variety of isogeometric applications. We introduced \underline{h} -refinement and \underline{p} -refinement algorithms for LR splines. Depending on the order of operations, combinations of these allow for \underline{hp} -refinement. We then introduced a set of data structures to manage the increased complexity due to the presence of non-uniform degrees. We investigated the efficacy of this approach for problems with smooth solutions as well as for problems with sharp local features. As anticipated, the local combinations of \underline{h} - and \underline{p} -refinement restored faster rates of convergence compared to the uniform degree \underline{h} -refined LR meshes; see [171].

The outcomes of this thesis are documented in the following articles: [171, 114, 170]. These publications provide comprehensive details on the methodologies, algorithms, and applications discussed in this work.

Chapter 6

Jean Michel Menjanahary: Dupin Cyclidic Splines of Arbitrary Topology

Dupin cyclides are algebraic surfaces of degree at most 4 in Euclidean space, discovered by Charles Dupin [172] in the early 1820s. They are initially defined as the envelope of a one parameter family of spheres touching three fixed spheres (in two ways); see Figure 6.1. They are remarkable in the sense that their curvature lines are either circles or straight lines.

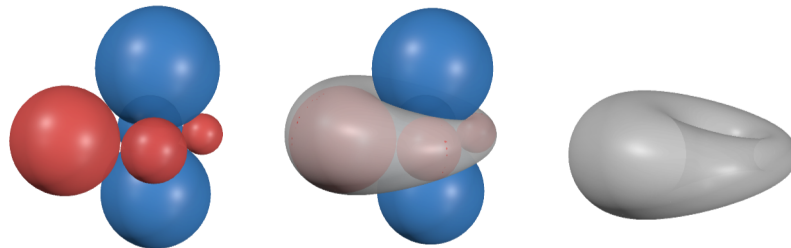


Figure 6.1: A Dupin cyclide as the envelope surface of a variable sphere in blue (resp. in red) touching three fixed spheres in red (resp. in blue).

One of the useful properties of Dupin cyclides is their closure under offsets, a feature that significantly enhances their utility in geometric modeling by allowing for the creation of parallel surfaces that maintain the elegant properties of the original cyclide and thus simplifying the design process. The well-known methods for designing free-form surfaces with Dupin cyclides was introduced by Martin in [173] with the use of quad patches, known as principal patches, bounded by 4 curvature lines that are circles. Dupin cyclide principal patches admit rational bidegree (2,2) parametrization along the curvature lines. It is well-known that the 4 corner points of a principal patch are always co-circular, and that a principal patch is uniquely determined by the 4 corner points and a frame at one corner point; see Figure 6.2. The frames at the other 3 corner points can be obtained by reflection across the underlying edges. A closed formula in the quaternionic framework, which will be more convenient for our approach, has been given in [174] and [175] to compute the

parametric representation of a principal patch. It has been reported in [176] that cyclidic splines based on principal patches are not flexible enough for the needs of modeling applications such as surface approximation. Our goal is to increase this flexibility by studying all possible blending of Dupin cyclides. The motivation comes from [177] that certain attempts to extended regular cyclidic splines to surfaces of arbitrary topology find empirical justification in architecture.

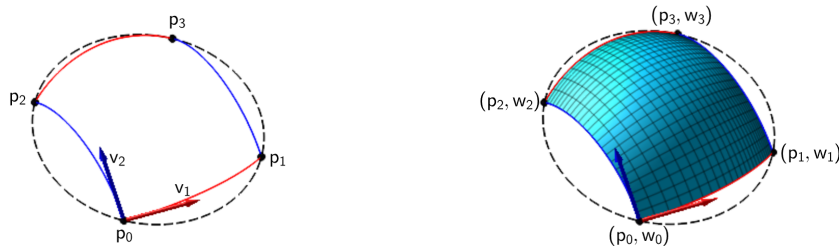


Figure 6.2: Left – a single quad patch mesh suitable for a Dupin cyclide principal patch. Right – a principal patch together with their control points and quaternionic weights associated with the initial data.

From the Laguerre geometry approaches in [178], it was known that there is a certain duality between parabolic Dupin cyclides (deg 3) and quadratic splines. The use of quadratic splines allows the well known Powell–Sabin subdivision procedure, which brings a subdivision procedure on the cyclidic spline construction. The thesis will investigate the generalization of this spline construction by allowing foldings and branchings of the Gaussian map to the unit sphere. This is illustrated in Figure 6.3, where the Monkey saddle is approximated with parabolic Dupin cyclides. Note that this subdivision allows the use of triangular patches bounded by algebraic curves of degree 3. It was proved later in [179] that Dupin cyclides can be blended with G^1 smoothness along algebraic curves of degree 3 or 4, and this is the only possible blending apart from the basic smooth blending along circles. There is a one parameter family of Dupin cyclides satisfying the smooth blending along those curves of degree 3 or 4.

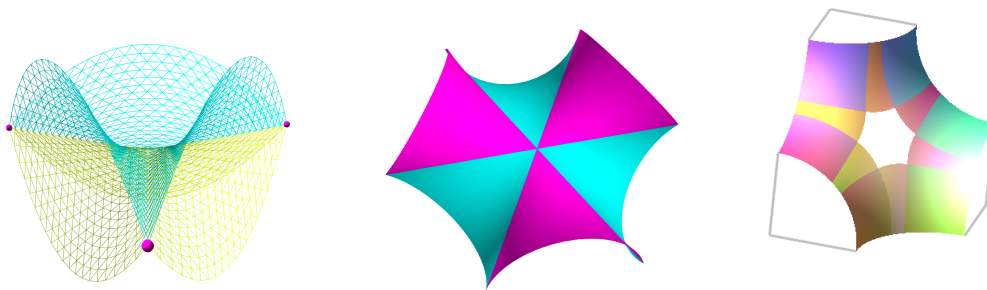


Figure 6.3: Left – foldings of quadratic splines. Middle – the Monkey saddle composed of triangular patches of parabolic cyclides as dual of the quadratic splines. Right – a filling construction of a hexagonal hole using principal patches of Dupin cyclides and a spherical patch in the middle.

In the quaternionic representation approach [175], the blending curves of degree 3 or 4 are nothing else than the diagonal curves $P(t,t)$ of principal patches $P(s,t)$. This approach allows filling in with Dupin cyclides a 4-sided circular boundary hole with two opposite right and supplementary angles. We shall call such a patch a hybrid patch; see Figure 6.4. The quaternionic representation of hybrid patches will be investigated together with their applications in the $2n$ -sided hole filling problem advertised in [177]; see the rightmost Figure 6.3 for the case $n = 3$.

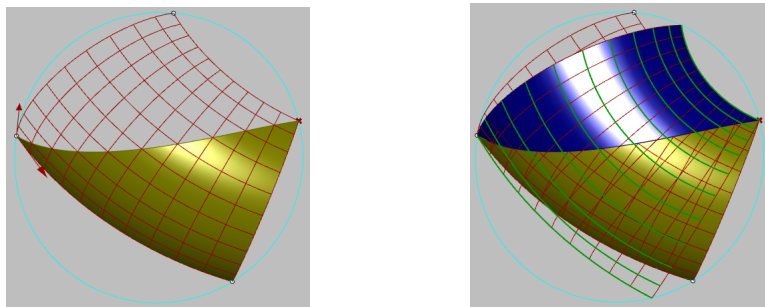


Figure 6.4: Left – a principal patch is trimmed along a principal diagonal curve of degree 4. Right – another half of a principal patch is blended smoothly along the same diagonal curve, resulting into a new shape of cyclidic spline constructions.

With the use of all possible blending of Dupin cyclides, cyclidic splines still have the following topological restrictions. Let S be a G^1 surface composed of Dupin cyclide patches. If S is closed without boundary, then it has to be of torus topology, i.e., $\chi(S) = 0$. Besides, if S is simply connected and bounded by principal circles, then the sum of its angles is the same as for a polygon on a plane. This clarifies that some spherical or planar patches have to be used to fulfil certain surface constructions. A good example, where a spherical patch has to be used, is to fill a $2n$ -sided hole, $n \geq 4$. This suggests that improving the existing algorithm on the hole filling problem, possibly with the use of hybrid patches, is unavoidable for the generalization for the multi-sided patch construction. As an infinite filling procedure was used in [177], we will improve this by only finitely many procedure and insert a multi-sided spherical patch in the middle, as illustrated by the rightmost Figure 6.3.

As a generalization of Dupin cyclidic splines into volume objects, we will also explore the quaternionic representation of 3D generalizations of principal patches called Dupin cyclidic (DC) cubes; see the leftmost Figure 6.5. A general procedure of building DC cubes was studied in [174]. DC cubes have unique extension to DC systems, i.e., triply orthogonal coordinate systems with Dupin cyclides as coordinate surfaces. It turned out that singularities of DC systems can be reduced up to Möbius transformations to focal 1-oval or 2-oval bicircular quartic curves on orthogonal planes, see Figure 6.5. This generalizes the notion of focal conics appearing as singularity of a DC system obtained by offsetting a Dupin cyclide. Each branch of the singular curves are lying on a sphere or a plane that is a single degeneration of coordinate surfaces in one family. Those spheres and/or planes are necessarily spheres and/or planes of symmetry of the coordinate system. The

full classification of DC systems via the quaternionic Möbius geometry approach of DC cubes will be studied. Preliminary results are available in [174].

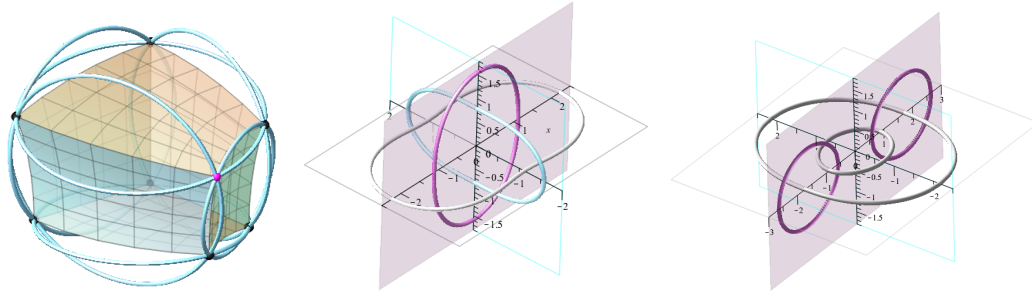


Figure 6.5: Left – a Dupin cyclidic cube. Middle – three focal 1-oval bicircular quartics on mutually orthogonal planes. Right – two focal 2-oval bicircular quartics on orthogonal planes and an empty curve on the third plane.

From the Laguerre geometry approach, more theories on Dupin cyclides and their spherical or planar sections were derived in [180]. First, a formula suggested by Gaston Darboux [181] to compute the implicit equation for a Dupin cyclide with given 3 generating spheres was successfully generalized. An unexpected relation between slices of Dupin cyclidic systems and families of Dupin cyclides with fixed planar sections was revealed. Also, it turned out that with a given planar or spherical sectional curve, there are at most 3 one parameter families of Dupin cyclides sharing the same sectional curve. Each of those families can be considered as coordinate surfaces of certain Dupin cyclidic systems with particular singularity on the sectional curve. This approach will be helpful for the investigation of 2D and 3D cyclidic spline constructions.

A chapter in the thesis will also explore Dupin cyclides via their implicit equations, particularly about the problem of recognition of Dupin cyclides from more general cyclides called Darboux cyclides as studied in [182]. In particular, the implementation and efficiency of the recognition algorithm will be highlighted. This includes a new proof of the Darboux formula for a Dupin cyclide, and formulation of a proof of equivalent definitions of Dupin cyclides in literature. This theory is also helpful to avoid singularities because a Möbius invariant of the Dupin cyclide can be computed right from the implicit equation as addressed in the same paper. The recognition results were first applied to the basic problem of smooth blending of Dupin cyclides along circles in [183]. The practicability of this approach will also be investigated together with the combination of the quaternionic representation approach.

In the last chapter of the thesis, I will report two (3 months) secondment works at Johannes Kepler University (JKU), Linz and at ModuleWorks, Aachen. At JKU, the problem of reconstruction of Darboux cyclides from their single view on the plane was considered. It was known that this problem is solved already for special cases of surfaces such as those smooth ones or those with ordinary singularities. An extended algorithm reconstruction has been addressed in [184] particularly for cyclide surfaces. At ModuleWorks, the quaternionic representation of Dupin cyclides was

implemented in the design tool Rihnoceros 3D with Grasshopper. A Grasshopper plugin has been developed to handle cyclidic splines. The examples demonstrating the efficiency of this plugin are Figure 6.4 and Figure 6.6 about offset of a principal patch, automated fair triangulation, and the ability to design free-form cyclide splines.

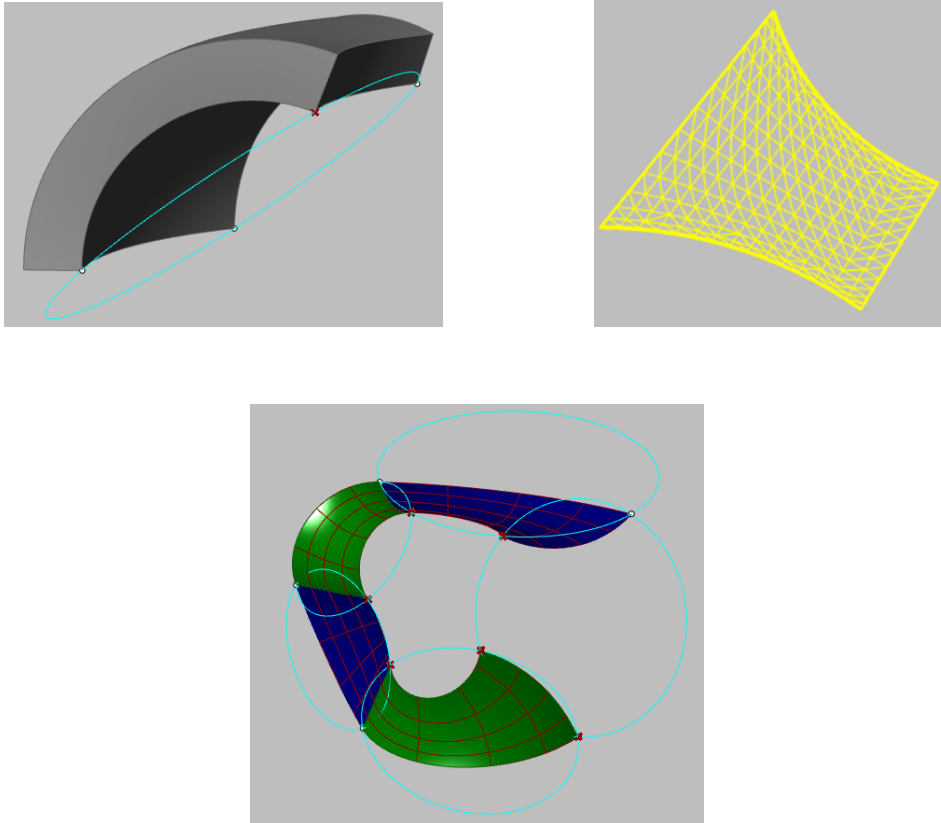


Figure 6.6

Chapter 7

Rao FU: Shape Reconstruction from 3D Point Clouds

Shape reconstruction from raw 3D point clouds is a core challenge in the field of computer graphics and computer vision. This thesis explores new approaches to tackle this multifaceted problem, with a specific focus on two aspects: primitive segmentation and mesh-based surface reconstruction.

For primitive segmentation, we introduce BPNet, a deep-learning framework designed to segment 3D point clouds according to Bézier primitives. Unlike conventional methods that address different primitive types in isolation, BPNet offers a more general and adaptive approach. Inspired by Bézier decomposition techniques commonly employed for Non-Uniform Rational B-Spline (NURBS) models, we employ Bézier decomposition to guide the segmentation of point clouds, removing the constraints posed by specific categories of primitives. More specifically, we contribute a joint optimization framework via a soft voting regularizer that enhances primitive segmentation, an auto-weight embedding module that streamlines the clustering of point features, and a reconstruction module that refines the segmented primitives. We validate the proposed approach on the ABC and AIM@Shape datasets. The experiments show superior segmentation performance and shorter inference time compared to baseline methods.

On surface reconstruction, we contribute a method designed to generate isotropic surface triangle meshes directly from unoriented 3D point clouds. The benefits of this approach lie in its adaptability to local feature size (LFS). Our method consists of three steps: LFS estimation, implicit function reconstruction and LFS-aware mesh sizing. The LFS estimation process computes the minimum of two geometric properties: local curvature radius and shape diameter, determined via jet-fitting and a Lipschitz-guided dichotomic search. The implicit function reconstruction step proceeds in three sub-steps: constructing a tetrahedron multi-domain from an unsigned distance function, signing the multi-domain via data fitting, and generating a signed robust distance function. Finally, the mesh sizing function, derived from the locally estimated LFS, controls the Delaunay refinement process used to mesh the zero-level set of the implicit function. This approach yields

isotropic meshes directly from 3D point clouds while offering an LFS-aware mesh density. Our experiments demonstrate the robustness of this approach, showcasing its ability to handle noise, outliers or missing data.

7.1 Context

In recent years, there has been a growing demand from practitioners to reconstruct shapes from 3D point clouds. This demand spans various fields, including manufacturing, video games and specialized applications like autonomous driving and digital twinning. This increasing demand signifies a substantial commercial market with promising investment prospects, capturing the attention of both industry and academia.

In manufacturing, shape reconstruction is a crucial functionality for computer-aided design (CAD) software, which is widely employed in mechanical prototyping, aerospace analysis and urban planning. In this context, shape reconstruction techniques transform raw 3D point cloud data, often acquired through 3D scanning or other measurement methods, into CAD models. This process facilitates shape reconstruction, converting real-world objects into digital copies. Figure 7.1 presents an example of a CAD software that utilizes shape reconstruction to design mechanical components.

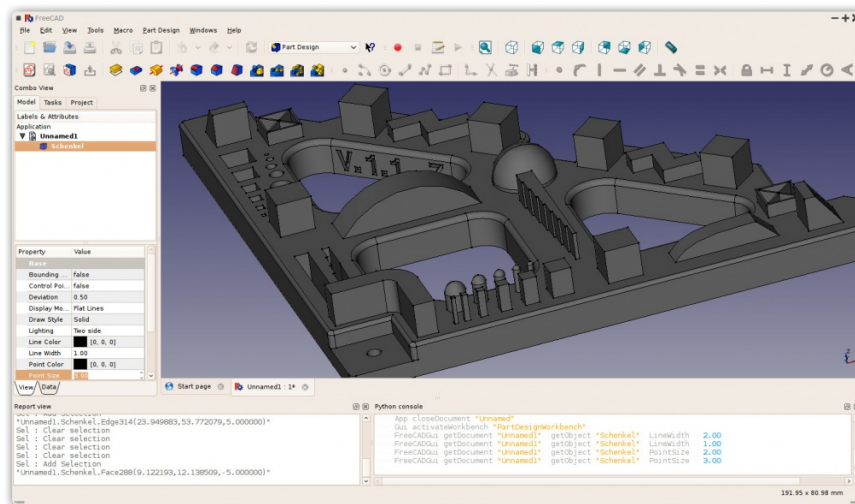


Figure 7.1: Shape reconstruction in CAD. Applying FreeCAD to design mechanical parts. Image taken from FreeCAD documentation [2].

In geography, shape reconstruction plays a pivotal role in geographic information systems (GIS) software for analyzing and visualizing geographical data. GIS applications depend on precise shape representations for terrain modeling and environmental simulations. In this scenario, raw point cloud data, often obtained through LiDAR, serves as the source for shape reconstruction. These

reconstructed shapes assist in downstream GIS-based flood modeling, land-use planning and disaster management analysis. Figure 7.2 shows an example of applying a GIS software for geological map analysis.

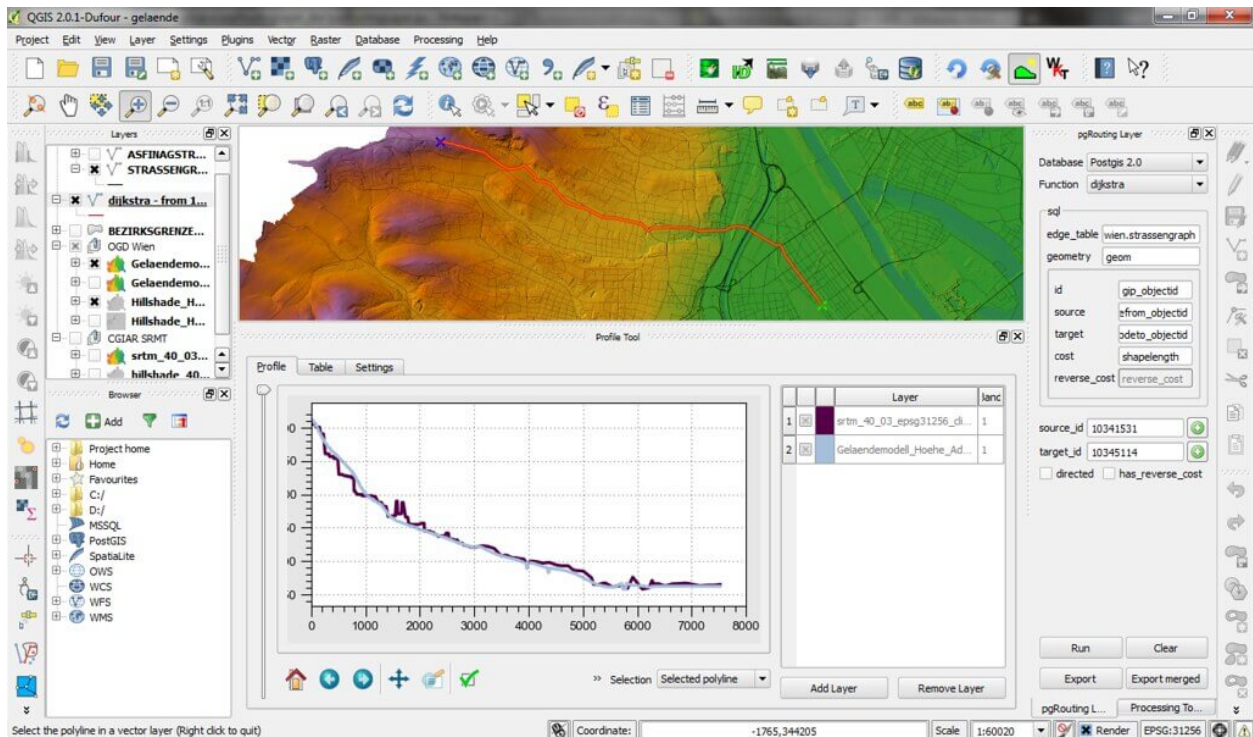


Figure 7.2: Shape reconstruction in GIS. Applying QGIS to generate a geological map. Image taken from the QGIS documentation [3].

The task of shape reconstruction from 3D point clouds is a challenging problem and remains a rapidly evolving area of research. It is an inherently ill-posed problem by nature. The illness arises because a point cloud represents a discrete sampling of an original surface, which means some information about the original surface is inevitably lost. In contrast, shape reconstruction seeks to infer the original continuous surface from this limited and discrete point cloud data. This mismatch between the discrete nature of the point cloud data and the objective of reconstruction of a continuous surface makes the shape reconstruction a complex scientific problem.

Existing solutions for shape reconstruction often follow two primary directions. The first direction involves fitting multiple Non-Uniform Rational B-Spline (NURBS) surface patches to approximate the original shape using point cloud data. NURBS surfaces are defined by control points and blending functions, offering a smooth and precise representation of shapes. However, this direction can be challenging when dealing with complex shapes that contain rich details, requiring the segmentation of point clouds before fitting. Nevertheless, combining segmentation with NURBS fitting remains a complex problem, and no mature packages or algorithms are currently readily available for industrial applications.

The second direction in shape reconstruction involves constructing a surface triangle mesh to approximate the original shape, a widely employed approach supported by mature packages that offer algorithms for mesh generation. For instance, the CGAL library [185] provides various surface reconstruction algorithms built upon the half-edge data structure [186] for mesh representation. However, challenges persist when dealing with real-world 3D point cloud data, which may be imperfect due to limitations in scanning equipment. Additionally, achieving high-fidelity mesh reconstructions while maintaining simplicity and well-shaped triangles remains an open and active area of research.

Before delving into the detailed scientific challenges associated with shape reconstruction, we first discuss how to obtain 3D point clouds, which serve as the direct input source for our shape reconstruction task. These 3D point clouds are assemblies of three-dimensional coordinates, often complemented by additional attributes like normals or RGB colors. To provide context, we explore popular techniques for acquiring 3D point clouds. Examining the principles and technologies behind these data-acquisition techniques offers valuable insights into the scientific challenges that arise for the shape reconstruction task.

Laser Scanning. Laser scanning (Figure 7.3) can capture precise 3D point clouds of the original shape. This process involves emitting laser beams toward the target and measuring the time it takes for the laser pulses to bounce back to a sensor. Laser scanners generate dense point cloud data representing the original shape by analyzing these measurements. However, certain challenges arise with laser scanning. High-reflection materials like glass can pose difficulties, leading to missing data in the point clouds. Additionally, occlusions caused by objects obstructing the scanner’s line of sight can result in gaps in the acquired data, requiring additional processing to address these issues. The sampling density, often measured in points per square meter, and accuracy, usually in millimeters, are critical factors in the quality of data acquired through laser scanning. They influence the success of subsequent shape reconstruction efforts.

Structured Light Scanning. Structured light scanning (Figure 7.4) utilizes a projector to cast a known pattern onto the surface of the target and a camera to capture how that pattern deforms, resulting in a structured point cloud representing the original shape. Unlike laser scanning, which is employed to capture large-scale scenes, structured light scanning is suitable for small to medium-sized shapes. However, structured light scanning still struggles with highly reflective or transparent materials. The accuracy and density of the generated point cloud depend on factors such as the projector, camera quality and the complexity of the projected pattern.

Photogrammetry. Photogrammetry (Figure 7.5) relies on capturing and analyzing multiple 2D images of a shape taken from different viewpoints. It leverages image feature matching and stereo principles to reconstruct the 3D point coordinates from multiple images. However, 3D point clouds



Figure 7.3: Laser scanning. A laser scanner is used to scan point clouds from a construction field. Image taken from [4].

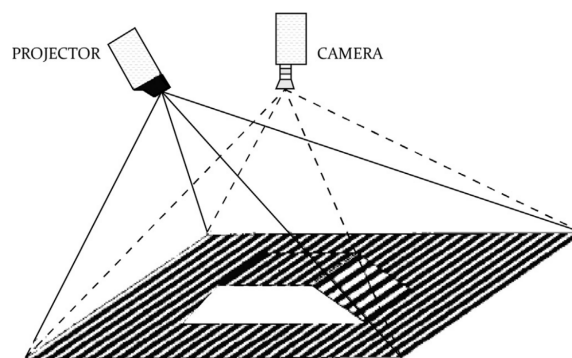


Figure 7.4: Structured Light Scanning. Standard projector-camera configuration used in structured light profilometry techniques. Image taken from [187].

obtained by photogrammetry are usually noisy, as the lighting conditions and reflections can affect the quality of the images and, consequently, the accuracy of the reconstruction. Calibration and synchronization of multiple cameras are also critical for precise results. In addition, the 3D point clouds suffer from extensive missing data because parts of the shape are occluded or not captured clearly in some images. This results in gaps or discontinuities in the point cloud data, challenging the shape reconstruction task. Despite these defects, photogrammetry remains a valuable tool, particularly when dealing with scenarios that are more amenable to image-based data capture.

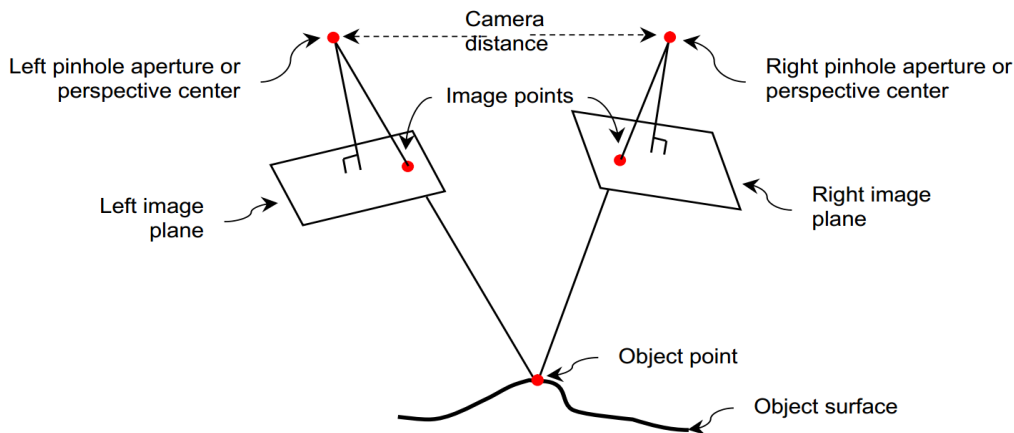


Figure 7.5: Photogrammetry. A schematic of the stereophotogrammetry technique showing the two cameras and how the location of a point is identified. Image taken from [5].

Depth Sensors. Depth sensors (Figure 7.6), like Laser scanners, also operate by emitting infrared light pulses and measuring their time-of-flight (ToF) to calculate distances. It then generates depth maps containing the shape structure based on the ToF, where we can obtain the 3D point cloud from the depth image. However, noise is a notable concern in depth sensors. Noise levels typically vary with distance, with greater distances resulting in less accurate depth measurements. Environmental factors like ambient light can also introduce noise, particularly in outdoor settings. Despite these limitations, depth sensors are favored for their cost-effectiveness and have found widespread usage in various applications, especially for scenarios where real-time depth perception is indispensable.

Simulation. Simulation generates raw point clouds digitally, emulating real-world scanning scenarios. It allows for controlled experiments and preserves the raw essence of intended environments or objects. However, simulations rely upon accurate modeling and assumptions, which can introduce errors. They may also lack the richness and defects of real-world data, limiting their real-world applicability.

Different acquisition methods yield raw 3D point clouds, which are crucial in various applications. Laser scanning excels in high-precision data capture, making it suitable for metrology and



Figure 7.6: Depth Sensor: The Azure Kinect DK depth camera. Image taken from Microsoft [6].

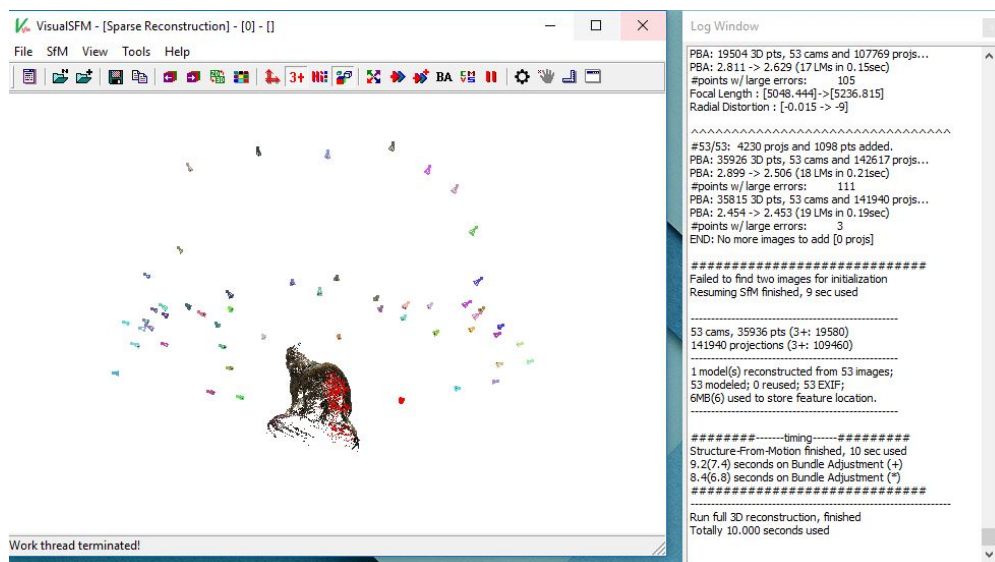


Figure 7.7: Simulation: point clouds with synthesized cameras. Image taken from VisualSFM [7].

large-scale scanning. Structured light offers real-time capabilities, making it valuable in interactive scenarios like gaming and virtual reality. Photogrammetry, while sometimes noisy, can handle city-scale outdoor environments, making it important for geographic information systems. Depth sensors are cost-effective and find applications in consumer devices like gaming consoles. However, the raw point clouds obtained from the aforementioned scanning technologies often come with imperfections due to inherent limitations. These imperfections motivate the design of robust shape reconstruction algorithms.

7.2 Scientific Challenges

Shape reconstruction refers to reconstruct a surface S from a given point cloud $P = \{p \in \mathbb{R}^3\}$. The problem lies in developing robust and efficient algorithms capable of transforming these unprocessed point clouds into accurate and complete 3D representations of objects. Addressing this problem involves tackling issues such as noise reduction, surface reconstruction, hole filling and ensuring geometric fidelity. The problem is multifaceted, marked by intricate challenges in those pivotal aspects: sampling conditions, reconstruction priors, geometric fidelity, topology, memory usage and computation, automation, generalization and evaluation criteria. Within the scope of this thesis, the focus is exclusively on unstructured 3D point clouds.

Sampling conditions. In the context of sampling conditions, the complexities arise from the diverse nature of data acquisition methods, each imparting specific characteristics to the raw point clouds. Challenges include variations in point density, presence of non-uniform sampling, the inherent noise in the acquired datasets, outliers from data inaccuracies or sensor errors, misalignment from multiple scans and missing data from occlusions. Figure 7.8 illustrates those challenges in 2D.

Reconstruction priors. These varied reconstruction priors correspond to different methodologies into the shape reconstruction problem, reflecting the assumption of priors about the nature of the unknown surface. Some methods assume global smoothness, expecting the surface to have a continuous and smooth variation. Others consider piecewise smoothness, where the surface can exhibit distinct regions of smoothness separated by discontinuities. There are also approaches assuming piecewise linearity, where a set of triangles directly interpolates the unknown surface from the input points. However, in the presence of noise in the point data, global smooth priors may lead to over-smoothing by approximation, piecewise smooth priors may struggle to identify the distinct regions, while piecewise linear priors often fail to smooth the noise. These trade-offs and considerations in choosing priors highlight the challenges of shape reconstruction, especially when dealing with real-world data characterized by imperfections and complexities. Please refer to Figure 7.9.

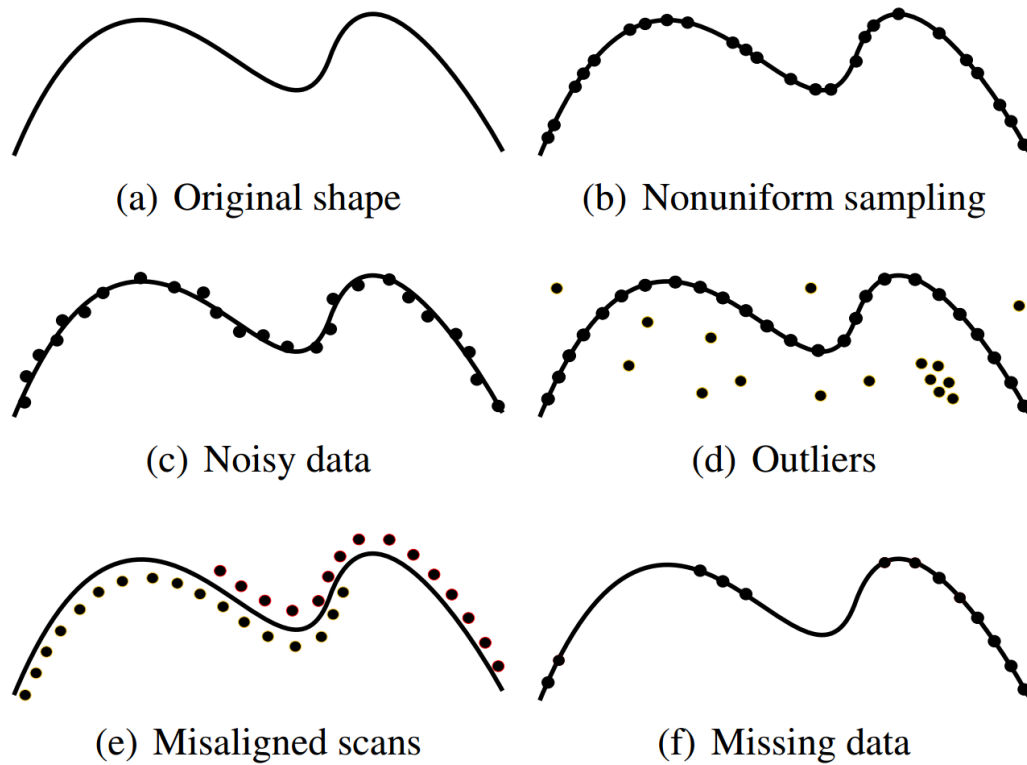


Figure 7.8: Different sampling conditions. The point cloud artifacts include non-uniform sampling, noisy data, outliers, misaligned scans and missing data. Image taken from [8].

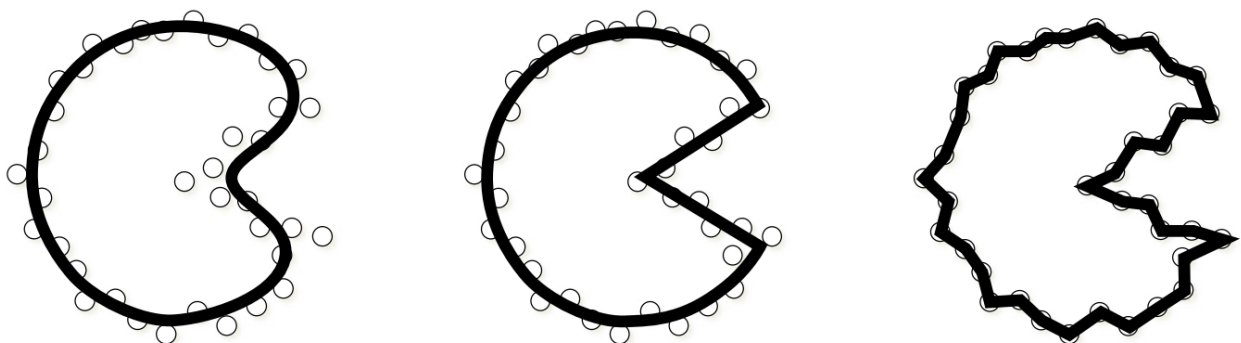


Figure 7.9: Priors for reconstruction methodologies. We identify three main priors: global smoothness, piecewise smoothness and linear interpolation.

Geometric fidelity. Geometric fidelity reflects how the reconstructed shape is close to the original shape or the input point cloud data. It pertains to the accuracy with which the reconstructed shape approximates the true geometry of the original shape. Achieving high geometric fidelity is crucial, especially in applications where precision is paramount, such as aerospace, medical imaging, and manufacturing. The challenge in assessing geometric fidelity arises from the fact that the true original shape is often unknown, and the input point cloud itself may contain imperfections and uncertainties. Therefore, measuring the geometric fidelity of the reconstruction becomes challenging, as there may be no reference shape for direct comparison.

Topology. Topology defines the connectivity and relationships within a shape, reflecting the number of connected components, boundaries, handles, and voids. In 3D shape reconstruction, maintaining the accurate preservation of these topological features is challenging, especially when the input point cloud is defect-laden. Some methods prioritize preserving the watertightness of the reconstructed shape, ensuring that there are no gaps or holes. Others focus on preserving non-manifold properties, which describe singular structures such as objects with multiple connected components or complex boundaries. Figure 7.10 depicts an example of a reconstructed hand with wrong topologies.

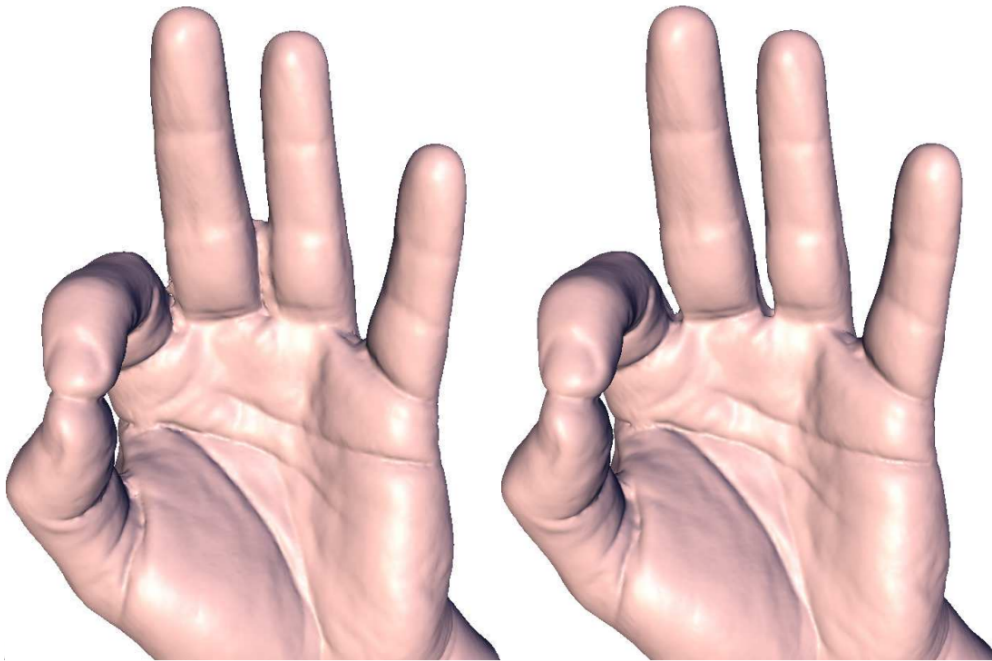


Figure 7.10: Topology issue for shape reconstruction reconstruction. Left: two fingers are connected with wrong topology. Right: two are fingers are separated with correct topology. Image taken from [9].

Memory usage and computation. Memory usage and computational demands present significant challenges in shape reconstruction. Handling extensive 3D point clouds can quickly deplete available

memory resources, potentially leading to inefficiencies. Moreover, the computational requirements for processing large point clouds can result in extended processing times, reducing overall efficiency. Achieving a balance between efficient memory management and computational speed is crucial. While machine learning-based methods have gained popularity, they often struggle with memory and computation issues, making them less practical for industrial applications. Figure 7.11 provides a visual example of memory-intensive shape reconstruction, with the primary source of high memory usage being the 3D convolution operations on voxels.

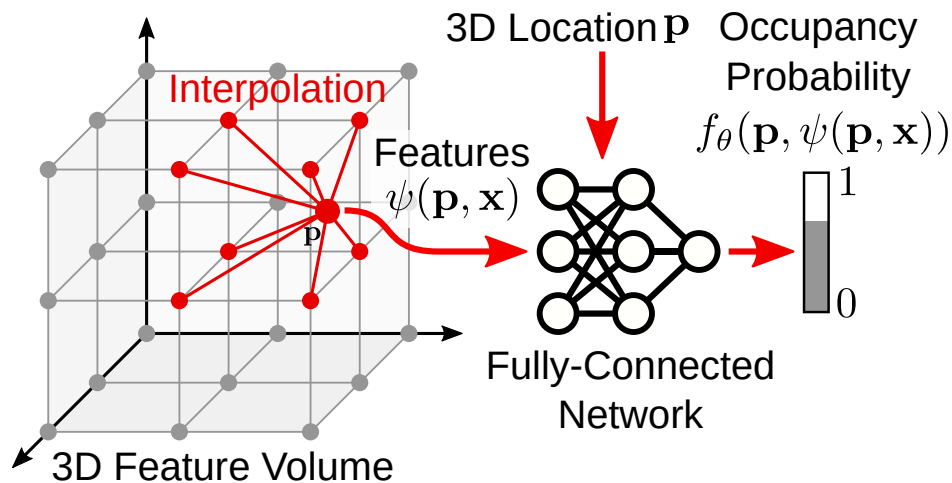


Figure 7.11: Memory-consuming reconstruction. Convolution on a voxel grid consumes large amounts of memory, thus lacking scalability for large-scale point clouds. Image taken from [10].

Automation. Making shape reconstruction automatic is a fundamental goal. While some methods can handle challenging topology cases with human intervention, such an interactive approach is labor-intensive and can produce results that vary based on different individuals' input, making it less reliable for large-scale or standardized applications. Figure 7.12 presents an example of interactive surface reconstruction, which requires users to provide hints to improve the reconstruction quality.

Generalization. Generalization in shape reconstruction refers to an algorithm's capacity to extend its reconstructive abilities beyond specific, limited examples and datasets. This is particularly significant in the context of learning-based methods. In essence, a well-generalized shape reconstruction algorithm should exhibit proficiency in accurately reconstructing surfaces that extend beyond the scope of its training data. However, many learning-based approaches are trained on specific object categories, such as chairs or couches. While they may perform well within these predefined categories, they can fail when faced with shapes significantly differing from their training data. Figure 7.13 highlights an example of the limitation of generalization using DeepSDF [11].

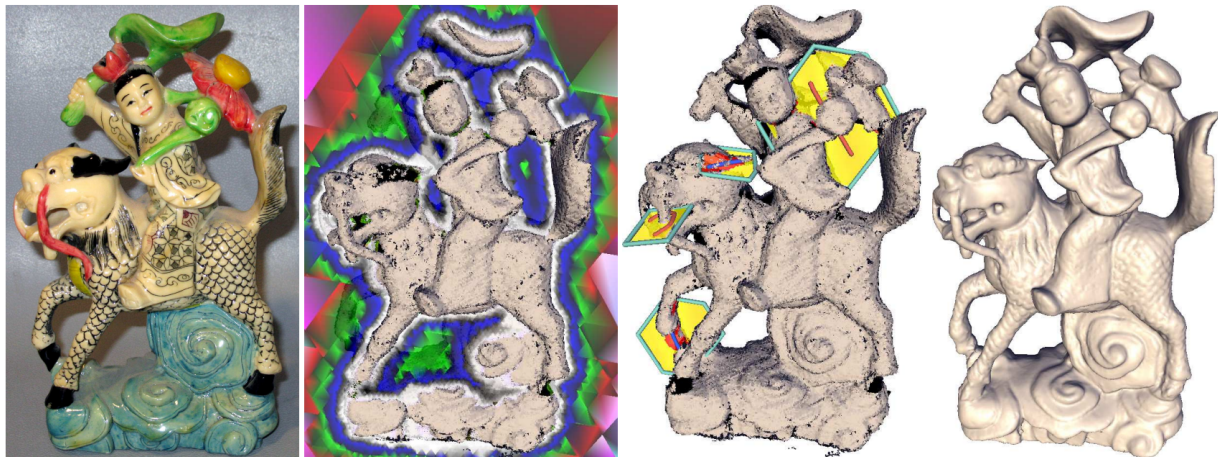


Figure 7.12: Interactive reconstruction. This approach requires users to provide scribbles for ambiguous areas. Image taken from [9].

This specific method is trained and tested on particular models, which restricts its applicability in industrial scenarios where shapes can vary significantly.

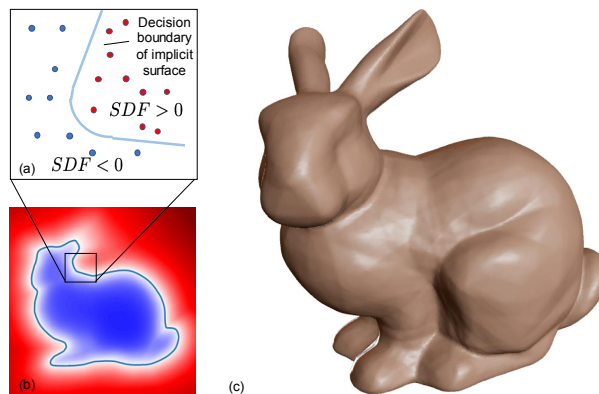


Figure 7.13: Non-general reconstruction. The learned SDF is constrained to specific shapes. Image taken from [11].

Other evaluation criteria. In addition to geometric fidelity and topological correctness, various other evaluation criteria are also crucial for shape reconstruction. These criteria encompass aspects such as visual fidelity, which assesses how closely the reconstructed shape resembles the original from a visual standpoint. Additionally, some evaluations consider view-dependent variants, accounting for how the reconstructed shape appears from different viewpoints. These diverse evaluation criteria provide a comprehensive assessment of the reconstructed shapes, ensuring their suitability for various applications.

7.3 Contributions

We first provide a summary of our contributions to shape reconstruction. Specifically, we explored both explicit and implicit shape reconstruction. In terms of explicit reconstruction, we have concentrated on primitive segmentation. Our proposed method involves segmenting point clouds with guidance from Bézier decomposition, enhancing the genericity of shape recognition. In terms of implicit reconstruction, we have introduced a new mesh reconstruction technique designed to extract LFS-aware (local-feature-size-aware) triangle surface meshes directly from 3D point clouds.

7.3.1 Primitive Segmentation

Existing primitive segmentation methods commonly approach the task by identifying individual primitives from point clouds in a separate manner. These approaches imply that a distinct optimization process is required for each primitive type. Consequently, these methods lack the generality required to handle a wide range of primitive shapes efficiently. Recognizing this limitation, our work aims to overcome the constraints of common approaches by introducing a more general and adaptive solution. We provide a detailed discussion of those existing methods in Chapter 2 of the Thesis.

Our first contribution is to define a general primitive type for the primitive segmentation task. We introduce BPNet, a new end-to-end deep learning framework designed to facilitate Bézier primitive segmentation on 3D point clouds. Unlike existing approaches constrained to finite shape categories because of distinct optimization, BPNet is inspired by the operation of Bézier decomposition techniques applied to NURBS models, which decompose different primitives into rational Bézier patches. We adapt and extend these principles to guide point cloud segmentation, making the Bézier decomposition learnable for point clouds and casting off the constraints of primitive types.

Specifically, our approach employs a joint optimization framework, enabling the concurrent learning of Bézier primitive segmentation and geometric fitting within a cascaded deep learning architecture. Notably, we introduce a soft voting regularizer to enhance primitive segmentation and propose an auto-weight embedding module to cluster point features, augmenting the network’s robustness and generality. Additionally, we introduce a reconstruction module that effectively processes multiple CAD models, each featuring different primitives, simultaneously.

To evaluate the efficacy of BPNet, we conducted extensive experiments on both synthetic ABC datasets and real-scan datasets. The results demonstrate superior segmentation performance compared to previous methods, accompanied by an improvement in inference speed. Furthermore, we demonstrate the generalizability of our method by training the model on the synthetic ABC datasets but testing with free-form points. Our model also yields reasonable segmentations on free-form 3D point sets with smoother boundaries compared to other baseline methods.

7.3.2 Surface Reconstruction

The primary goal in surface reconstruction is to ensure fidelity in reconstructing shapes. Many modern surface reconstruction algorithms place a strong emphasis on achieving high fidelity by closely fitting the reconstructed mesh to the input data. However, this main focus on fidelity can sometimes result in poor-quality meshes, particularly when contouring implicit surfaces using contouring methods like marching cubes or marching tetrahedrons, leading to non-isotropic and dense triangular meshes. Consequently, post-processing steps, such as remeshing, become necessary to convert these non-isotropic and dense meshes into isotropic ones with simplified complexity. Additionally, it is important for the mesh itself to be adaptive, meaning that it should use small triangles in regions with rich details. While post-processing can undoubtedly enhance mesh quality, it also introduces the potential risk of compromising the overall reconstruction quality. This gives rise to another objective in surface reconstruction: Is it possible to obtain an isotropic and adaptive mesh directly from surface reconstruction without the need for post-processing?

Our second contribution addresses the two above objectives altogether by contributing a method for reconstructing an isotropic surface triangle mesh directly from an unoriented 3D point cloud, without remeshing. The key novelty of our approach lies into its adaptability to the local feature size (LFS), resulting in a density that aligns with the local details of the underlying data. Instead of following the conventional path of dense reconstruction followed by remeshing, our method simultaneously reconstructs both an implicit function and an LFS-aware mesh sizing function. These two components cooperate to produce the final LFS-aware reconstructed mesh, which is isotropic with adaptive sizes.

Determining the LFS is a critical step for our surface reconstruction method. We achieve LFS estimation by considering the minimum of two geometric properties: the local curvature radius and half of the shape diameter. We derive the local curvature from a polynomial surface adjusted via least squares fitting (also referred to as jet fitting), while we estimate the shape diameter by our proposed Lipschitz-guided dichotomic search. We then construct the implicit function in three main steps: creating a tetrahedron multi-domain mesh from an unsigned distance function, refining the multi-domain with data fitting, and generating a robust signed distance function.

Finally, we utilize the Delaunay refinement meshing approach to obtain the final LFS-aware mesh. Specifically, we derive a mesh sizing function, which controls the density of the final mesh, from the estimated LFS. We then employ the sizing function with the implicit function to mesh the zero-level set of the implicit function using Delaunay refinement. The distinctive feature of our approach is its ability to generate isotropic meshes from 3D point clouds while maintaining an LFS-aware density. Our experiments demonstrate the method’s robustness in handling defects such as noise, outliers, or missing data. Furthermore, we provide experiments to show that our method can reconstruct shapes with complicated topologies, such as high genus.

This thesis is supported by the following publications:

- Rao Fu, Cheng Wen, Qian Li, Xiao Xiao, Pierre Alliez. BpNet: Bézier Primitive Segmentation on 3D Point Clouds. In Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, 2023.
- Rao Fu, Kai Hormann, Pierre Alliez. Submitted to IEEE Transactions on Visualization and Computer Graphics.

7.4 Outline

This thesis contributes to the shape reconstruction problem by approaching it from the perspectives of primitive segmentation and surface mesh reconstruction. The thesis is structured as follows:

- (a) Chapter 1 provides an overview of shape reconstruction with its applications and challenges. It also discusses 3D point cloud acquisition that will challenge the shape reconstruction task. Finally, this chapter outlines our contributions.
- (b) Chapter 2 presents a thorough survey of the existing literature and related work in the field of shape reconstruction, with a primary focus on primitive segmentation and surface reconstruction.
- (c) Chapter 3 delves into our contributions and methodologies concerning primitive segmentation, accompanied by extensive experiments and comparisons in this domain.
- (d) Chapter 4 investigates our contributions and methodologies related to surface reconstruction, providing an in-depth analysis of the experiments conducted in this area.
- (e) Chapter 5 summarizes the key findings and contributions of the thesis and outlines potential directions for future research.

Bibliography

- [1] T. Duff, K. Kohn, A. Leykin, and T. Pajdla, “Plmp – point-line minimal problems in complete multi-view visibility,” 2019.
- [2] J. Riegel, W. Mayer, and Y. van Havre, “Freecad,” Freecadspec2002. pdf, 2016.
- [3] N. Moyroud and F. Portet, “Introduction to qgis,” QGIS and generic tools, vol. 1, pp. 1–17, 2018.
- [4] J. Wang, T. Yi, X. Liang, and T. Ueda, “Application of 3d laser scanning technology using laser radar system to error analysis in the curtain wall construction,” Remote Sensing, vol. 15, no. 1, p. 64, 2022.
- [5] M. N. Helfrick, C. Niezrecki, P. Avitabile, and T. Schmidt, “3d digital image correlation methods for full-field vibration measurement,” Mechanical systems and signal processing, vol. 25, no. 3, pp. 917–927, 2011.
- [6] Microsoft, “Azure kinect dk depth camera.” <https://learn.microsoft.com/en-us/azure/kinect-dk/depth-camera>, 2023.
- [7] C. Wu, “Visualsfm: A visual structure from motion system.” <http://www.cs.washington.edu/homes/ccwu/vsfm>, 2011.
- [8] M. Berger, A. Tagliasacchi, L. M. Seversky, P. Alliez, G. Guennebaud, J. A. Levine, A. Sharf, and C. T. Silva, “A survey of surface reconstruction from point clouds,” Computer Graphics Forum, vol. 36, no. 1, pp. 301–329, 2017.
- [9] A. Sharf, T. Lewiner, G. Shklarski, S. Toledo, and D. Cohen-Or, “Interactive topology-aware surface reconstruction,” ACM Transactions on Graphics (TOG), vol. 26, no. 3, pp. 43–es, 2007.
- [10] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger, “Convolutional occupancy networks,” in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16, pp. 523–540, Springer, 2020.

- [11] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, “DeepSDF: Learning continuous signed distance functions for shape representation,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 165–174, 2019.
- [12] T. Duff, C. Hill, A. Jensen, K. Lee, A. Leykin, and J. Sommars, “Solving polynomial systems via homotopy continuation and monodromy,” IMA Journal of Numerical Analysis, vol. 39, pp. 1421–1446, 04 2018.
- [13] Z. Kukulova, Algebraic Methods in Computer Vision. PhD thesis, 2013.
- [14] B. Osserman and M. Trager, “Multigraded Cayley-Chow forms,” Advances in Mathematics, vol. 348, pp. 583–606, 2019.
- [15] J. Kileel, Z. Kukulova, T. Pajdla, and B. Sturmfels, “Distortion varieties,” 2016.
- [16] L. Buse, D. Cox, and C. D’Andrea, “Implicitization of surfaces in p^3 in the presence of base points,” 2003.
- [17] C. Katsamaki, F. Rouillier, E. Tsigaridas, and Z. Zafeirakopoulos, “PTOPO: Computing the Geometry and the Topology of Parametric Curves,” Journal of Symbolic Computation, Aug. 2022.
- [18] I. Davies, E. Duarte, I. Portakal, and M.-S. Sorea, “Families of polytopes with rational linear precision in higher dimensions,” Foundations of Computational Mathematics, Aug. 2022.
- [19] R. Krasauskas, “Toric surface patches,” Advances in Computational Mathematics, vol. 17, pp. 89–113, 2002.
- [20] F. Bihan, A. Dickenstein, and M. Giaroli, “Lower bounds for positive roots and regions of multistationarity in chemical reaction networks,” 2018.
- [21] E. Feliu, O. Henriksson, and B. Pascual-Escudero, “Dimension and degeneracy of solutions of parametric polynomial systems arising from reaction networks,” 2023.
- [22] G. Craciun, A. Dickenstein, A. Shiu, and B. Sturmfels, “Toric dynamical systems,” 2007.
- [23] A. Le, F. Rouillier, D. Chablat, and G. Rance, “On the certification of the kinematics of 3-dof spherical parallel manipulators,” 2023.
- [24] A. Caminata and E. Gorla, “Solving degree, last fall degree, and related invariants,” Journal of Symbolic Computation, vol. 114, pp. 322–335, 2023.
- [25] F. Catanese, S. Hosten, A. Khetan, and B. Sturmfels, “The maximum likelihood degree,” American Journal of Mathematics, vol. 128, 07 2004.

-
- [26] L. Garcia-Puente and F. Sottile, “Linear precision for parametric patches,” 2009.
- [27] I. Soprunov, “Toric complete intersection codes,” Journal of Symbolic Computation, vol. 50, pp. 374–385, mar 2013.
- [28] H. Schenck, Algebraic Foundations for Applied Topology and Data Analysis. Mathematics of Data, Springer International Publishing, 2022.
- [29] M. R. Bender, “Solving sparse polynomial systems using groebner bases and resultants,” in Proceedings of the 2022 International Symposium on Symbolic and Algebraic Computation, ISSAC ’22, ACM, July 2022.
- [30] D. A. Cox, “The homogeneous coordinate ring of a toric variety,” Journal of Algebraic Geometry, no. 4, pp. 17–50, 1995.
- [31] J. F. Canny and I. Z. Emiris, “An efficient algorithm for the sparse mixed resultant,” Applied algebra, algebraic algorithms and error-correcting codes, vol. 673, pp. 89 – 104, 1993.
- [32] F. S. Macaulay, “Some formulae in elimination,” 1903.
- [33] C. Checa and I. Emiris, “A greedy approach to the canny-emiris formula,” in Proceedings of the 2022 International Symposium on Symbolic and Algebraic Computation, ISSAC ’22, (New York, NY, USA), pp. 283 – 291, Association for Computing Machinery, 2022.
- [34] C. D’Andrea, G. Jeronimo, and M. Sombra, “The canny-emiris conjecture for the sparse resultant,” Foundations of Computational Mathematics, 2022.
- [35] C. Checa and I. Z. Emiris, “Mixed subdivisions suitable for the canny-emiris formula,” 2023.
- [36] J.-P. Jouanolou, “Formes d’inertie et resultant: Un formulaire,” Advances in Mathematics, vol. 126, p. 119–250, 03 1997.
- [37] L. Busé, M. Chardin, and N. Nemati, “Multigraded sylvester forms, duality and elimination matrices,” Journal of Algebra, vol. 609, pp. 514–546, 2022.
- [38] L. Busé and C. Checa, “Toric sylvester forms,” 2023.
- [39] E. Cattani, A. Dickenstein, and B. Sturmfels, “Residues and resultants,” Journal of Mathematical Sciences, vol. 5, pp. 119–148, 1997.
- [40] M. R. Bender, J.-C. Faugère, and E. Tsigaridas, “Towards mixed gröbner basis algorithms: the multihomogeneous and sparse case,” ISSAC - 43rd Intern. Symp. on Symbolic and Algebraic Computation, 2018.
- [41] D. Bayer and M. Stillman, “A criterion for detecting m-regularity,” Inventiones Mathematicae, vol. 87, pp. 1–11, Feb. 1987.

- [42] G. Caviglia and E. Sbarra, “Characteristic free bounds for the castelnuovo mumford regularity,” Compositio Mathematica, vol. 141, no. 6, pp. 1365–1373, 2005.
- [43] D. Maclagan and G. G. Smith, “Multigraded castelnuovo-mumford regularity,” Journal für die reine und angewandte Mathematik, vol. 2004, no. 571, pp. 179–212, 2004.
- [44] T. Römer, “Homological properties of bigraded algebras,” Illinois Journal of Mathematics, vol. 45, no. 4, pp. 1361 – 1376, 2001.
- [45] J. Vrsek, “Contour curves and isophotes on rational ruled surfaces,” Computer Aided Geometric Design, vol. 65, pp. 1–12, 2018.
- [46] M. M. Young, “Dupin’s cyclide as a self-dual surface,” American Journal of Mathematics, vol. 38, no. 3, pp. 267–286, 1916.
- [47] J. M. Menjanahary, E. Hoxhaj, and R. Krasauskas, “Singularities of Dupin cyclidic cubes. submitted to Computer Aided Geometric Design (2024) and will be available soon on arxiv.”
- [48] E. Hoxhaj, J. M. Menjanahary, and J. Schicho, “Using algebraic geometry to reconstruct a darbox cyclide from a calibrated camera picture,” Applicable Algebra in Engineering, Communication and Computing, Mar 2023.
- [49] E. Hoxhaj and J.Schicho, “How to reconstruct a planar map from its branching curve,” Mathematics of Computation.
- [50] E. Hoxhaj, J. M. Menjanahary, and R. Krasauskas, “Sections of dupin cyclides and their focal properties. submitted to journal of symbolic computations (2024) and will be available soon on arxiv.”
- [51] R. Hartley and A. Zisserman, Multiple view geometry in computer vision. Cambridge University Press, 2004.
- [52] M. Berger and E.Boyer, “3d surface reconstruction using occluding contours,” in Computer Analysis of Images and Patterns (V. Hlaváč and R. Šára, eds.), (Berlin, Heidelberg), pp. 198–205, Springer Berlin Heidelberg, 1995.
- [53] O. Chisini, “Sulla identità birazionale di due funzioni algebriche di più variabili, dotate di una medesima varietà di diramazione,” Ist. Lombardo Sci. Lett. Rend Cl. Sci. Mat. Nat. (3), vol. 77, pp. 339–356, 1944.
- [54] V. S. Kulikov, “On Chisini’s conjecture,” Izv. Ross. Akad. Nauk Ser. Mat., vol. 63, no. no. 6, pp. 83–116, 1999.
- [55] F. Catanese, “On a problem of Chisini,” Duke Math. J., vol. 53, no. 1, pp. 33–42, 1986.

-
- [56] S. Nemirovski, “Kulikov’s theorem on the chisini conjecture,” Izvestiya: Mathematics, vol. 65, pp. 71–74, 2001.
- [57] B. Maxime, Ueber die Reihenentwickelungen der Potentialtheorie. Leipzig: B. G. Teubner, 1894.
- [58] J. Maxwell, “On the cyclide,” Quarterly Journal of Pure and Applied Mathematics, vol. 9, pp. 111–126, 1868.
- [59] A. Sym and A. Szereszewski, “On darboux’s approach to r-separability of variables,” SIGMA, vol. 7, 2011.
- [60] M. Gallet, N. Lubbes, J. Schicho, and J. Vršek, “Reconstruction of surfaces with ordinary singularities from their silhouettes,” SIAGA, vol. 3, no. 3, pp. 472–506, 2019.
- [61] J. d’Almeida, “Courbe de ramification de la projection sur \mathbf{P}^2 d’une surface de \mathbf{P}^3 ,” Duke Math. J., vol. 65, no. 2, pp. 229–233, 1992.
- [62] R. Hartshornes, Algebraic Geometry. Springer, 1977.
- [63] C. Ciliberto and F. Flamini, “On the branch curve of a general projection of a surface to a plane,” Trans. AMS, vol. 363, no. 7, pp. 3457–3471, 2011.
- [64] T.-H. Joung, S.-G. Kang, J.-K. Lee, and J. Ahn, “The imo initial strategy for reducing greenhouse gas (ghg) emissions, and its follow-up actions towards 2050,” Journal of International Maritime Safety, Environmental Affairs, and Shipping, vol. 4, no. 1, pp. 1–7, 2020.
- [65] S. Khan and P. Kaklis, “From regional sensitivity to intra-sensitivity for parametric analysis of free-form shapes: Application to ship design,” Advanced Engineering Informatics, vol. 49, p. 101314, 2021.
- [66] S. Khan, P. Kaklis, A. Serani, and M. Diez, “Geometric moment-dependent global sensitivity analysis without simulation data: application to ship hull form optimisation,” Computer-Aided Design, vol. 151, p. 103339, 2022.
- [67] S. Khan, P. Kaklis, A. Serani, M. Diez, and K. Kostas, “Shape-supervised dimension reduction: Extracting geometry and physics associated features with geometric moments,” Computer-Aided Design, vol. 150, p. 103327, 2022.
- [68] S. Khan, A. Serani, M. Diez, and P. Kaklis, “Physics-informed feature-to-feature learning for design-space dimensionality reduction in shape optimisation,” in AIAA scitech 2021 forum, p. 1235, 2021.

- [69] S. Khan, K. Goucher-Lambert, K. Kostas, and P. Kaklis, “ShipHullGAN: A generic parametric modeller for ship hull design using deep convolutional generative model,” Computer Methods in Applied Mechanics and Engineering, vol. 411, p. 116051, 2023.
- [70] S. Khan, E. Gunpinar, and B. Sener, “Genyacht: An interactive generative design system for computer-aided yacht hull design,” Ocean Engineering, vol. 191, p. 106462, 2019.
- [71] W. Taylor, “Method of Lagrangian curvilinear interpolation,” Journal of Research of the National Bureau of Standards, vol. 35, Aug. 1945.
- [72] C. Schneider and W. Werner, “Some new aspects of rational interpolation,” Mathematics of Computation, vol. 47, July 1986.
- [73] J.-P. Berrut, “Rational functions for guaranteed and experimentally well-conditioned global interpolation,” Computers and Mathematics with Applications, vol. 15, no. 1, 1988.
- [74] J. L. d. Lagrange, “Leçon cinquième. sur l’usage des courbes dans la solution des problèmes,” Journal de l’École polytechnique, VIIe et VIIIe cahiers, t. II, 1812.
- [75] H. Rutishauser, Vorlesungen über Numerische Mathematik, Band 1: Gleichungssysteme, Interpolation und Approximation. Birkhäuser Basel, 1976.
- [76] J.-P. Berrut and H. Mittelmann, “Lebesgue constant minimizing linear rational interpolation of continuous functions over the interval,” Computers and Mathematics with Applications, vol. 33, Mar. 1997.
- [77] J.-P. Berrut and L. Trefethen, “Barycentric Lagrange interpolation,” SIAM Review, vol. 46, Sept. 2004.
- [78] H. R. Schwarz and N. Köckler, Numerische Mathematik. Vieweg+Teubner Verlag Wiesbaden, 2011.
- [79] D. Gibb, A Course in Interpolation and Numerical Integration for the Mathematical Laboratory. Cornell University Library historical math monographs, G. Bell and Sons, Limited, 1915.
- [80] P. Henrici, Elements of Numerical Analysis. John Wiley and Sons, New York, 1964, 1965.
- [81] H. Salzer, “Lagrangian interpolation at the Chebyshev points $x_{n,v} \equiv \cos(v\pi/n)$, $v = 0(1)n$; some unnoted advantages,” The Computer Journal, vol. 15, 05 1972.
- [82] P. Henrici, “Barycentric formulas for interpolating trigonometric polynomials and their conjugates,” Numerische Mathematik, vol. 33, jun 1979.

-
- [83] J. C. F. Gauss, “Theoria interpolationis methodo nova tractata,” Werke, Königlichen Gesellschaft der Wissenschaften, vol. 3, 1866.
- [84] H. Salzer, “Coefficients for facilitating trigonometric interpolation,” Journal of Mathematics and Physics, vol. 27, Apr. 1948.
- [85] J.-P. Berrut, “Baryzentrische formeln zur trigonometrischen interpolation (I),” Zeitschrift für angewandte Mathematik und Physik, vol. 35, jan 1984.
- [86] A. Ramanantoanina and K. Hormann, “Shape control tools for periodic Bézier curves,” Computer Aided Geometric Design, vol. 103, 2023.
- [87] L. Piegl, “A geometric investigation of the rational Bézier scheme of computer aided design,” Computers in Industry, vol. 7, no. 5, 1986.
- [88] G. Farin, “Curvature continuity and offsets for piecewise conics,” ACM Transactions on Graphics, vol. 8, Apr. 1989.
- [89] J. Sánchez-Reyes, “Shape factors and shoulder points for shape control of rational Bézier curves,” Computer-Aided Design, vol. 157, 2023.
- [90] G. Farin, Curves and Surfaces for CAGD: A Practical Guide. The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling, San Francisco: Morgan Kaufmann, 5th ed., 2001.
- [91] R. Patterson, “Projective transformations of the parameter of a Bernstein–Bézier curve,” ACM Transactions on Graphics, vol. 4, Oct. 1985.
- [92] M. McCool, “Projective reparameterizations of rational Bézier simplices,” in Proceedings of Graphics Interface '93, GI 1993, (Toronto, Ontario, Canada), Canadian Information Processing Society, 1993.
- [93] A. Ramanantoanina and K. Hormann, “New shape control tools for rational Bézier curve design,” Computer Aided Geometric Design, vol. 88, 2021.
- [94] L. Piegl, “On the use of infinite control points in CAGD,” Computer Aided Geometric Design, vol. 4, July 1987.
- [95] G. Farin and D. Hansford, The Essentials of CAGD. New York: A K Peters/CRC Press, 1st ed., 2000.
- [96] P. Woźny and F. Chudy, “Linear-time geometric algorithm for evaluating Bézier curves,” Computer-Aided Design, vol. 118, 2020.

- [97] M. Dupuy, “Le calcul numérique des fonctions par l’interpolation barycentrique,” Comptes Rendus de l’Académie des Sciences. Série I, Mathématique, vol. 226, 1948.
- [98] W. Werner, “Polynomial interpolation: Lagrange versus Newton,” Mathematics of Computation, vol. 43, July 1984.
- [99] A. Marco and J.-J. Martínez, “A fast and accurate algorithm for solving Bernstein–Vandermonde linear systems,” Linear Algebra and its Applications, vol. 422, Apr. 2007.
- [100] A. Róth, I. Juhász, J. Schicho, and M. Hoffmann, “A cyclic basis for closed curve and surface modeling,” Computer Aided Geometric Design, vol. 26, no. 5, 2009.
- [101] A. Róth, “Simple and weighted cyclic proximity curves and surfaces,” Computer-Aided Design, vol. 137, 2021.
- [102] J. Sánchez-Reyes, “Periodic Bézier curves,” Computer Aided Geometric Design, vol. 26, no. 9, 2009.
- [103] I. Juhász and A. Róth, “Closed rational trigonometric curves and surfaces,” Journal of Computational and Applied Mathematics, vol. 234, no. 8, 2010.
- [104] J. Xiao and J. P. Boyd, “Periodized radial basis functions, part I: Theory,” Applied Numerical Mathematics, vol. 86, 2014.
- [105] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, Isogeometric Analysis: Toward Integration of CAD and FEA. John Wiley and Sons, 2009.
- [106] A. Bressan and E. Sande, “Approximation in FEM, DG and IGA: a theoretical comparison,” Numerische Mathematik, vol. 143, pp. 923–942, 2019.
- [107] T. J. R. Hughes, J. A. Cottrel, and Y. Basilev, “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement,” Computer Methods in Applied Mechanics and Engineering, vol. 194, pp. 4135–4195, 2005.
- [108] Y. Bazilevs, L. Beirão da Veiga, J. A. Cottrell, T. J. R. Hughes, and G. Sangalli, “Isogeometric analysis: approximation, stability and error estimates for h -refined meshes,” Mathematical Models and Methods in Applied Sciences, vol. 16, pp. 1031–1090, 2006.
- [109] C. d. Boor, A practical guide to splines, vol. 27. Springer-Verlag New York, 1978.
- [110] S. Karlin and W. J. Studden, Tchebycheff Systems: With Applications in Analysis and Statistics. Interscience Publishers, New York, 1966.

-
- [111] T. Lyche, C. Manni, and H. Speleers, “Tchebycheffian B-splines revisited: An introductory exposition,” in Advanced Methods for Geometric Modeling and Numerical Simulation (C. Giannelli and H. Speleers, eds.), vol. 35 of Springer INdAM Series, pp. 179–216, Springer International Publishing, 2019.
- [112] M. L. Mazure, “Finding all systems of weight functions associated with a given extended Chebyshev space,” Journal of Approximation Theory, vol. 163, pp. 363–376, 2011.
- [113] L. L. Schumaker, Spline Functions: Basic Theory. Cambridge University Press, Cambridge, third ed., 2007.
- [114] K. Raval, C. Manni, and H. Speleers, “Tchebycheffian B-splines in isogeometric Galerkin methods,” Computer Methods in Applied Mechanics and Engineering, vol. 403, p. 115648, 2023.
- [115] M. L. Mazure, “Extended Chebyshev piecewise spaces characterised via weight functions,” Journal of Approximation Theory, vol. 145, pp. 33–54, 2007.
- [116] S. Karlin, Total Positivity. Stanford University Press, Stanford, 1968.
- [117] B. Buchwald and G. Mühlbach, “Construction of B-splines for generalized spline spaces generated from local ECT-systems,” Journal of Computational and Applied Mathematics, vol. 159, pp. 249–267, 2003.
- [118] N. Dyn and A. Ron, “Recurrence relation for Tchebycheffian B-splines,” Journal d’Analyse Mathématique, vol. 51, pp. 118–138, 1988.
- [119] D. Bister and H. Prautzsch, “A new approach to Tchebycheffian B-splines,” in Curves and Surfaces with Applications in CAGD (A. Le Méhauté, C. Rabut, and L. L. Schumaker, eds.), pp. 387–394, Vanderbilt University Press, Nashville, 1997.
- [120] T. Goodman and M. L. Mazure, “Blossoming beyond extended Chebyshev spaces,” Journal of Approximation Theory, vol. 109, pp. 48–81, 2001.
- [121] R. R. Hiemstra, T. J. R. Hughes, C. Manni, H. Speleers, and D. Toshniwal, “A Tchebycheffian extension of multidegree B-splines: Algorithmic computation and properties,” SIAM Journal on Numerical Analysis, vol. 58, pp. 1138–1163, 2020.
- [122] H. Speleers, “Algorithm 1020: Computation of multi-degree Tchebycheffian B-splines,” ACM Transactions on Mathematical Software, vol. 48, p. 12, 2022.
- [123] C. V. Beccari, G. Casciola, and L. Romani, “A practical method for computing with piecewise Chebyshevian splines,” Journal of Computational and Applied Mathematics, vol. 406, p. 114051, 2022.

- [124] C. Manni, F. Pelosi, and M. L. Sampoli, “Generalized B-splines as a tool in isogeometric analysis,” Computer Methods in Applied Mechanics and Engineering, vol. 200, pp. 867–881, 2011.
- [125] A. Aimi, M. Diligenti, M. L. Sampoli, and A. Sestini, “Non-polynomial spline alternatives in isogeometric symmetric Galerkin BEM,” Applied Numerical Mathematics, vol. 116, pp. 10–23, 2017.
- [126] C. Manni, F. Pelosi, and M. L. Sampoli, “Isogeometric analysis in advection–diffusion problems: Tension splines approximation,” Journal of Computational and Applied Mathematics, vol. 236, pp. 511–528, 2011.
- [127] M. L. Cardinali, C. Garoni, C. Manni, and H. Speleers, “Isogeometric discretizations with generalized b-splines: Symbol-based spectral analysis,” Applied Numerical Mathematics, vol. 166, pp. 288–312, 2021.
- [128] T. Lyche, C. Manni, and H. Speleers, “Foundations of spline theory: B-splines, spline approximation, and hierarchical refinement,” in Splines and PDEs: From Approximation Theory to Numerical Linear Algebra (T. Lyche, C. Manni, and H. Speleers, eds.), vol. 2219 of Lecture Notes in Mathematics, pp. 1–76, Springer International Publishing, 2018.
- [129] L. L. Schumaker and L. Wang, “Approximation power of polynomial splines on T-meshes,” Computer Aided Geometric Design, vol. 29, pp. 599–612, 2012.
- [130] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri, “T-splines and T-NURCCs,” ACM Transactions on Graphics, vol. 22, pp. 477–484, 2003.
- [131] C. Giannelli, B. Jüttler, and H. Speleers, “THB-splines: The truncated basis for hierarchical splines,” Computer Aided Geometric Design, vol. 29, pp. 485–498, 2012.
- [132] T. Dokken, T. Lyche, and K. F. Pettersen, “Polynomial splines over locally refined box-partitions,” Computer Aided Geometric Design, vol. 30, pp. 331–356, 2013.
- [133] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, and T. Lyche, “T-spline simplification and local refinement,” ACM Transactions on Graphics, vol. 23, pp. 276–283, 2004.
- [134] H. Casquero, L. Liu, Y. Zhang, A. Reali, and H. Gomez, “Isogeometric collocation using analysis-suitable T-splines of arbitrary degree,” Computer Methods in Applied Mechanics and Engineering, vol. 301, pp. 164 – 186, 2016.
- [135] A. Buffa, D. Cho, and M. Kumar, “Characterization of T-splines with reduced continuity order on T-meshes,” Computer Methods in Applied Mechanics and Engineering, vol. 201, pp. 112–126, 2012.

-
- [136] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, and T. W. Sederberg, “Isogeometric analysis using T-splines,” Computer Methods in Applied Mechanics and Engineering, vol. 199, pp. 229–263, 2010.
- [137] A. Buffa, D. Cho, and G. Sangalli, “Linear independence of the T-spline blending functions associated with some particular T-meshes,” Computer Methods in Applied Mechanics and Engineering, vol. 199, pp. 1437 – 1445, 2010.
- [138] L. Beirão da Veiga, A. Buffa, D. Cho, and G. Sangalli, “Analysis-suitable T-splines are dual-compatible,” Computer Methods in Applied Mechanics and Engineering, vol. 249, pp. 42–51, 2012.
- [139] Z. Jingjing and X. Li, “Local refinement for analysis-suitable++ T-splines,” Computer Methods in Applied Mechanics and Engineering, vol. 342, pp. 32 – 45, 2018.
- [140] D. R. Forsey and R. H. Bartels, “Hierarchical B-spline refinement,” ACM SIGGRAPH Computer Graphics, vol. 22, pp. 205–212, 1988.
- [141] R. Kraft, Adaptive and linearly independent multilevel B-splines. SFB 404, Geschäftsstelle, 1997.
- [142] E. J. Evans, M. A. Scott, X. Li, and D. C. Thomas, “Hierarchical T-splines: Analysis-suitability, Bézier extraction, and application as an adaptive basis for isogeometric analysis,” Computer Methods in Applied Mechanics and Engineering, vol. 284, pp. 1 – 20, 2015.
- [143] C. Giannelli and B. Jüttler, “Bases and dimensions of bivariate hierarchical tensor-product splines,” Journal of Computational and Applied Mathematics, vol. 239, pp. 162 – 178, 2013.
- [144] C. Giannelli, B. Jüttler, S. K. Kleiss, A. Mantzaflaris, B. Simeon, and J. Speh, “THB-splines: An effective mathematical technology for adaptive refinement in geometric design and isogeometric analysis,” Computer Methods in Applied Mechanics and Engineering, vol. 299, pp. 337 – 365, 2016.
- [145] A. Buffa and C. Giannelli, “Adaptive isogeometric methods with hierarchical splines: Error estimator and convergence,” Mathematical Models and Methods in Applied Sciences, vol. 26, pp. 1–25, 2016.
- [146] G. Kermarrec, V. Skytt, and T. Dokken, Optimal Surface Fitting of Point Clouds Using Local Refinement: Application to GIS Data. Springer Nature, 2023.
- [147] K. A. Johannessen, M. Kumar, and T. Kvamsdal, “Divergence-conforming discretization for Stokes problem on locally refined meshes using LR B-splines,” Computer Methods in Applied Mechanics and Engineering, vol. 293, pp. 38–70, 2015.

- [148] C. Zimmermann and R. A. Sauer, “Adaptive local surface refinement based on LR NURBS and its application to contact,” Computational Mechanics, vol. 60, pp. 1011–1031, 2017.
- [149] A. Bressan, “Some properties of LR-splines,” Computer Aided Geometric Design, vol. 30, pp. 778–794, 2013.
- [150] K. A. Johannessen, F. Remonato, and T. Kvamsdal, “On the similarities and differences between classical hierarchical, truncated hierarchical and LR B-splines,” Computer Methods in Applied Mechanics and Engineering, vol. 291, pp. 64–101, 2015.
- [151] A. Bressan and B. Jüttler, “A hierarchical construction of LR meshes in 2D,” Computer Aided Geometric Design, vol. 37, pp. 9–24, 2015.
- [152] G. Nürnberger, L. L. Schumaker, M. Sommer, and H. Strauss, “Generalized Chebyshevian splines,” SIAM Journal on Mathematical Analysis, vol. 15, pp. 790–804, 1984.
- [153] T. W. Sederberg, J. Zheng, and X. Song, “Knot intervals and multi-degree splines,” Computer Aided Geometric Design, vol. 20, pp. 455–468, 2003.
- [154] D. Toshniwal, H. Speleers, R. R. Hiemstra, and T. J. R. Hughes, “Multi-degree smooth polar splines: A framework for geometric modeling and isogeometric analysis,” Computer Methods in Applied Mechanics and Engineering, vol. 316, pp. 1005–1061, 2017.
- [155] H. Speleers, “Algorithm 999: Computation of multi-degree B-splines,” ACM Transactions on Mathematical Software, vol. 45, p. 43, 2019.
- [156] H. Speleers and D. Toshniwal, “A general class of C^1 smooth rational splines: Application to construction of exact ellipses and ellipsoids,” Computer-Aided Design, vol. 132, p. 102982, 2021.
- [157] C. V. Beccari, G. Casciola, and S. Morigi, “On multi-degree splines,” Computer Aided Geometric Design, vol. 58, pp. 8 – 23, 2017.
- [158] C. Giannelli, B. Jüttler, and H. Speleers, “Strongly stable bases for adaptively refined multi-level spline spaces,” Advances in Computational Mathematics, vol. 40, pp. 459–490, 2014.
- [159] N. Engleitner and B. Jüttler, “Patchwork B-spline refinement,” Computer-Aided Design, vol. 90, pp. 168–179, 2017.
- [160] H. Casquero, L. Liu, Y. Zhang, A. Reali, J. Kiendl, and H. Gomez, “Arbitrary-degree T-splines for isogeometric analysis of fully nonlinear Kirchhoff-Love shells,” Computer-Aided Design, vol. 82, pp. 140 – 153, 2017.

-
- [161] J. A. Cottrell, T. J. R. Hughes, and A. Reali, “Studies of refinement and continuity in isogeometric structural analysis,” Computer Methods in Applied Mechanics and Engineering, vol. 196, pp. 4160 – 4183, 2007.
- [162] T. J. R. Hughes, A. Reali, and G. Sangalli, “Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: comparison of p-method finite elements with k-method NURBS,” Computer Methods in Applied Mechanics and Engineering, vol. 197, pp. 4104–4124, 2008.
- [163] H. Prautzsch, “Degree elevation of B-spline curves,” Computer Aided Geometric Design, vol. 1, pp. 193–198, 1984.
- [164] C. Bracco, T. Lyche, C. Manni, F. Roman, and H. Speleers, “On the dimension of Tchebycheffian spline spaces over planar T-meshes,” Computer Aided Geometric Design, vol. 45, pp. 151–173, 2016.
- [165] C. Bracco, T. Lyche, C. Manni, and H. Speleers, “Tchebycheffian spline spaces over planar T-meshes: Dimension bounds and dimension instabilities,” Journal of Computational and Applied Mathematics, vol. 349, pp. 265–278, 2019.
- [166] B. Mourrain, “On the dimension of spline spaces on planar T-meshes,” Mathematics of Computation, vol. 83, pp. 847–871, 2014.
- [167] C. Bracco, D. Berdinsky, D. Cho, M. Oh, and T. Kim, “Trigonometric generalized T-splines,” Computer Methods in Applied Mechanics and Engineering, vol. 268, pp. 540–556, 2014.
- [168] C. Bracco, T. Lyche, C. Manni, F. Roman, and H. Speleers, “Generalized spline spaces over T-meshes: Dimension formula and locally refined generalized B-splines,” Applied Mathematics and Computation, vol. 272, pp. 187–198, 2016.
- [169] C. Manni, F. Pelosi, and H. Speleers, “Local hierarchical h-refinements in IgA based on generalized B-splines,” in Mathematical Methods for Curves and Surfaces 2012 (M. Floater, T. Lyche, M. L. Mazure, K. Mørken, and L. L. Schumaker, eds.), vol. 8177 of Lecture Notes in Computer Science, pp. 341–363, Springer–Verlag, Heidelberg, 2014.
- [170] K. Raval, C. Manni, and H. Speleers, “Adaptive isogeometric analysis based on Locally Refined Tchebycheffian B-splines,” Available at SSRN 4761823, 2024.
- [171] K. Raval, R. R. Hiemstra, K. A. Johannessen, and T. Dokken, “Local h-, p- and k- adaptivity with Locally Refined B-splines,” In preparation.
- [172] C. Dupin, Applications de géométrie et de mécanique. Bachelier, 1822.

- [173] R. R. Martin, “Principal patches—a new class of surface patch based on differential geometry,” 1983.
- [174] J. M. Menjanahary, E. Hoxhaj, and R. Krasauskas, “Classification of Dupin cyclidic cubes by their singularities,” [arXiv preprint arXiv:2405.07225](https://arxiv.org/abs/2405.07225), 2024.
- [175] S. Zube and R. Krasauskas, “Representation of Dupin cyclides using quaternions,” Graphical Models, vol. 82, pp. 110–122, 2015.
- [176] Y. Srinivas, V. Kumar, and D. Dutta, “Surface design using cyclide patches,” Computer-Aided Design, vol. 28, no. 4, pp. 263–276, 1996.
- [177] R. Mesnil, C. Douthe, O. Baverel, and B. Léger, “Generalised cyclidic nets for shape modelling in architecture,” International Journal of Architectural Computing, vol. 15, no. 2, pp. 148–168, 2017.
- [178] H. Pottmann and M. Peternell, “Applications of Laguerre geometry in cagd,” Computer Aided Geometric Design, vol. 15, no. 2, pp. 165–186, 1998.
- [179] R. Krasauskas and C. Mäurer, “Studying cyclides with Laguerre geometry,” Computer Aided Geometric Design, vol. 17, pp. 101–126, 2000.
- [180] E. Hoxhaj, J. M. Menjanahary, and R. Krasauskas, “Sections of Dupin cyclides and their focal properties,” [Available at SSRN 4807452](https://www.ssrn.com/abstract/4807452).
- [181] G. Darboux, “Sur les sections du tore,” Nouvelles annales de mathématiques 2e série, vol. tome 3, pp. 156–165, 1864.
- [182] J. M. Menjanahary and R. Vidunas, “Dupin cyclides as a subspace of Darboux cyclides,” [arXiv preprint arXiv:2212.14015](https://arxiv.org/abs/2212.14015), 2022.
- [183] J. M. Menjanahary and R. Vidunas, “Dupin cyclides passing through a fixed circle,” Mathematics, vol. 12, no. 10, p. 1505, 2024.
- [184] E. Hoxhaj, J. M. Menjanahary, and J. Schicho, “Using algebraic geometry to reconstruct a Darboux cyclide from a calibrated camera picture,” Applicable Algebra in Engineering, Communication and Computing, pp. 1–17, 2023.
- [185] CGAL, “The computational geometry algorithms library.” <https://www.cgal.org/>, 2009. Accessed: 2023-05-15.
- [186] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy, Polygon mesh processing. CRC press, 2010.
- [187] S. Van der Jeught and J. J. Dirckx, “Real-time structured light profilometry: a review,” Optics and Lasers in Engineering, vol. 87, pp. 18–31, 2016.