University of West Bohemia
Faculty of Applied Sciences

D O C T O R A L T H E S I S

Plzeň, 2011

Mgr. Marek Byrtus

FACULTY
OF APPLIED SCIENCES
UNIVERSITY
OF WEST BOHEMIA

www.KMA.zcu.cz
SINCE 1954

PARAMETRIZATION METHODS OF ALGEBRAIC VARIETIES

Mgr. Marek Byrtus

A thesis for the degree of Doctor of Philosophy in the subject of Applied Mathematics

Supervisor:
Doc. RNDr. František Ježek, CSc.

Co-supervisor:
Ing. Bohumír Bastl, Ph.D.

Plzeň, 2011

FAKULTA
APLIKOVANÝCH VĚD
ZÁPADOČESKÉ
UNIVERZITY
V PLZNI

www.KMA.zcu.cz
SINCE 1954

METODY PARAMETRIZACE

ALGEBRAICKÝCH VARIET

Mgr. Marek Byrtus

Disertační práce k získání akademického titulu doktor v oboru aplikovaná matematika

Školitel:
Doc. RNDr. František Ježek, CSc.

Školitel specialista:
Ing. Bohumír Bastl, Ph.D.

Plzeň, 2011

“If you can’t explain it to a six year old, you don’t understand it yourself.”
Albert Einstein

Declaration

I hereby declare that this Ph.D. thesis is completely my own work and that I used only the cited sources.

.....

Marek Byrtus

Acknowledgments

I would like to thank to my co-supervisor Ing. Bohumír Bastl, Ph.D. for his guidance, valuable and very often fruitful comments and suggestions. I would like to thank to my supervisor Doc. RNDr. František Ježek, CSc. and to all members at the department of geometry.

I would like to express my thanks to Prof. Bert Jüttler for his guidance through my one year stay at Johannes Kepler University in Linz, Austria. I am also thankful to DI. Dr. Mario Kapl with whom I cooperated and I would like to thank the Austrian Science Fund, which supported me through the project S09202.

I would like to thank to my wife Dominika for her moral support and love.

Annotation

This thesis is devoted to special techniques for interpolation of planar (points and associated tangent vectors) and spatial (quadrilateral mesh of points with associated normal vectors) data.

In the first theoretical part, we study Hermite interpolation by cubic Pythagorean hodograph (PH) curves. Inspired by Walton and Meek (see [80]), we corrected and extended their results and described all input Hermite data for which an interpolating arc of PH cubic exists. Moreover, we analyze a number of solutions and existence of a loop on an interpolant for given data. Further, we prove that arbitrary G^1 Hermite data can be interpolated by at most two interpolating arcs of PH cubic and there are infinitely many such pairs for any input data. Finally, we focus on C^1 Hermite interpolation by PH cubic. Similarly to G^1 interpolation, any C^1 Hermite data can be interpolated by at most two arcs of PH cubics and we present a method which gives all four possible solutions. We also discuss an appearance of a loop on interpolating arcs.

The second theoretical part of the thesis deals with Bubble patches as a new method for generating an interpolation G^n-surface from a quadrilateral mesh with normals. The method is based on a local construction which works uniformly for vertices of arbitrary valency. For each quadrilateral we construct a surface patch, represented by a bubble patch, in such a way that these patches are pieced together with G^n continuity. The construction of a single patch is based on Gordon-Coons interpolation. The obtained surface is piecewise rational with arbitrary smoothness and interpolates the vertices and normals. In the case of G^0, G^1 and G^2-surface, the construction is described in detail. The method can be generalized to G^n-surfaces for any $n \geq 3$. We also show different examples of obtained continuity and verify the corresponding smoothness with the help of reflection lines.

Keywords:
G^1 Hermite interpolation, Pythagorean hodograph cubic, Tschirnhausen cubic, C^1 Hermite interpolation, Bézier curve, G^n-surface, Interpolating surface, Bubble patch, Quadrilateral mesh, Gordon-Coons interpolation.

Anotace

Disertační práce se zabývá speciálními interpolačními technikami rovinných (zadané body s tečnými vektory) a prostorových (čtyřúhelníková síť bodů s normálovými vektory) geometrických dat.

V první teoretické části práce se věnujeme Hermitově interpolaci rovinnou kubikou s Pythagorejským hodografem (PH). Práce opravuje a rozšiřuje výsledky z článku Waltona a Meeka (viz [80]) a popisuje všechna vstupní Hermitovská data, pro které existuje PH kubický interpolant. Navíc je provedena analýza počtu a kvality (zda-li daný interpolant obsahuje samoprůnik či ne) řešení pro vstupní data. Vzhledem k tomu, že libovolná G^1 Hermitova data není možné interpolovat pouze jedním PH interpolantem, je v práci dokázáno, že libovolná vstupní G^1 data je možné vždy interpolovat dvěmi částmi PH kubiky a že těchto dvojic interpolantů existuje pro daná vstupní data nekonečně mnoho. Dále se práce zabývá C^1 Hermitovou interpolací PH kubikami a podobně jako u G^1 interpolace, libovolná C^1 data je možné interpolovat pomocí dvou oblouků PH kubiky. V závěru první části je ukázán postup, jak nalézt všechna čtyři možná řešení a je provedena diskuze ohledně kvality každého interpolantu, tj. výskytu samoprůniku.

Druhá teoretická část práce se zabývá novou G^n interpolační metodou – Bubble plátování – na čtyřúhelníkových sítích s asociovanými normálovými vektory. Metoda je založena na lokální konstrukci a lze ji použít pro vrcholy libovolné valence. Pro každý čtyřúhelník v síti je konstruován takový plát, že je se sousedními pláty napojen v G^n spojitosti. Konstrukce každého dílčího plátu je založena na Gordon-Coonsově interpolaci a výsledný plát má racionální popis. Pro G^0, G^1 a G^2 plochy je konstrukce popsána detailněji a odpovídající spojitost je ověřena pomocí tzv. metody "reflection lines".

Klíčová slova:
G^1 Hermitova interpolace, kubika s Pythagorejským hodografem, Tschirnhausenova kubika, C^1 Hermitova interpolace, Bézierova křivka, G^n plocha, Interpolační plocha, Bubble plát, Čtyřúhelníková síť, Gordon-Coonsova interpolace.

Contents

Part I

INTRODUCTION

1 Introduction

Current CAD/CAM systems mostly rely on NURBS representation of curves and surfaces, i.e., objects are represented by polynomial or rational parameterizations. A natural requirement is to have also the derived objects, such as offset curves and surfaces or convolution curves and surfaces represented by rational parameterizations. Unfortunately, the rationality of these objects is not generally preserved. This problem motivated the study of special classes of objects with rational offsets. In 1990, Farouki and Sakkalis in [37] introduced an important subclass of polynomial parametric curves called Pythagorean Hodograph (PH) curves. The most significant properties of these curves are that their arc-length function is piecewise polynomial and they possess rational offset curves. The state of art of Pythagorean hodograph curves and related topics is written down in Chapter 3. Moreover, Chapter 2 is devoted to the historical overview of standard approaches and techniques in Computer Aided Geometric Design.

In the second part of the thesis (Chapters 4 and 5), we focus on the simplest non-trivial polynomial PH curve, which is the only one PH cubic know as Tschirnhausen cubic. Walton and Meek in [80] studied G^1 Hermite interpolation by arcs of Tschirnhausen cubic, but the results in their paper were not precise. This fact inspired us to describe for which input G^1 Hermite data an interpolating arc of PH cubic exists (Proposition 6). Moreover, we discuss a number of interpolating PH cubics for given input data. Further, in Section 4.2, we analyze a quality of PH cubic arc in the manner whether it contains a loop or not. A discussion dealing with the quality of PH interpolant is influenced by the results from a paper written by Stone and DeRose (see [110]), who analyzed a quality of a standard Bézier cubic.

Since PH cubic interpolant does not exist for arbitrary G^1 Hermite data, we also investigate a number of interpolants necessary to match any input data. We prove that any G^1 Hermite data can be interpolated by at most two interpolating arcs of PH cubics (Theorem 14). Although PH cubic does not have enough flexibility in comparison to a standard cubic, it is always possible to interpolate any C^1 Hermite data by a pair of PH cubic interpolants joined with C^1 continuity. This surprising result is summarized in Theorem 17. Further, the proof of Theorem 17 is considered as a construction how to obtain desired four pairs of interpolants.

The third part of the thesis, i.e., Chapter 6, is devoted to the interpolation of quadrilateral mesh given by points with associated normal vectors. We design a new technique for interpolation of such data with arbitrary G^n smoothness. A constructed interpolation is called Bubble patch and is mainly

based on Gordon–Coons interpolation. We present several advantages of the construction, e.g., the Bubble patch construction possesses rational parametrization, it works for the meshes of arbitrary valency (this is not usual in standard approaches), the determining compatible twist vectors at the vertices is given as a solution of the system of linear equation. In Subsection 6.4 we study G^0, G^1 and G^2 smoothness in detail and in Subsection 6.6 we present our method on several examples and we verify the desired smoothness by reflection lines.

Main results of the thesis can be summarized as follows:

- We precisely describe for which input G^1 Hermite data an interpolating arc of PH cubic exists. Moreover, we specify the quality of a PH cubic interpolant, i.e., if it contains a loop or not. These results have been already published in journal, see [9].
- We show that at most two interpolating arcs of PH cubic are needed to interpolate any G^1 Hermite data. Further, we formulate a criteria how to find two arcs of PH cubic matching C^1 Hermite data and we show a construction how to obtain all four pairs of interpolating PH cubics joined with C^1 smoothness.
- We design a new technique for interpolation of quadrilateral mesh with associated normal vectors. This approach is based on Gordon–Coons interpolation and has several advantages, e.g., it possesses rational parametrization, it works for meshes with arbitrary valency.

2

Computer Aided Geometric Design

A development of geometry and related sciences is as old as a human being. The earliest beginnings go back to ancient Mezopotamia and Egypt around 3000 BC. The geometry was considered as a collection of empirically discovered principles concerning lengths, angles, areas, and volumes, which were developed to meet some practical needs in surveying, construction, astronomy, and various crafts. One of the branch which has been developed through centuries is a shipbuilding. The other were car and aeronautical industry, which have arisen after industrialization at the end of nineteenth century and a big progress was noticed one century later. Usually, research in geometric modelling have been motivated by technical needs in industrial production. This fact caused that several today's know descriptions have been discovered separately, because the companies kept them in a secret.

The foundation of Computer Aided Geometric Design (CAGD) dated back to 1974, when R. Barnhill and R. Riesenfeld organized the first CAGD conference for researches from Europe and U.S. Ten years later, CAGD journal was established by R. Barnhill and W. Boehm.

2.1 Historical Overview

The first use of curves is recorded in AD Roman times, where they served as a tool in shipbuilding. The construction consists of ship's ribs, which were wooden planks and which created a template for ship construction. The advantage was that they could be reused several times, but the disadvantage was that they could not be recreated. In the Renaissance, Venetians improved ribs techniques in a way that they defined ribs as tangent continuous circular arc and consequently they got a ship hull by changing the ribs' shapes along the keel. The first mention of today's known spline is referred to [24]. Another branch, where a spline appeared, was aeronautical engineering. R. Liming, who was working for North American Aviation, wrote in his book[1] an approach how to use conics in building aircrafts.

In 1963, a French mathematician Paul de Faget de Casteljau introduced his algorithm in technical report, see [11], for the car company Citroën[2], where he was employed since 1959. He used an

[1] Analytical Geometry with Application to Aircraft

[2] Founded in 1919 by André-Gustave Citroën, Citroën was the first mass-production car company outside of the USA

unusual idea to combine control polygon and Bernstein polynomials[3] instead of defining a curve through points on it. It was used to change a curve directly and this concept of curves gave a possibility to change or control a curve by its control polygon. The Casteljau's construction had been already introduced by Blaschke in [6], but at that time nobody saw a contribution of such definition of a curve. Although the algorithm have been established by the author in sixties, it was named after him in seventies when W. Boehm[4] started to use it.

Independently, Pierre Bézier in Rénault realized the need for computer representations of curves and surfaces. His main idea was to represent a curve as the intersection of two elliptic cylinders, which were defined inside a parallelepiped and therefore affine transformation of a curve has been allowed. With the help of polynomial representation he gave a description of a curve, which was later rewritten using Bernstein polynomials by A. R. Forrest, see [42].

An important tool in geometrical modeling are B-splines, which were introduced by I. Schoenberg[5] in 1946, see [106], where he investigated an equidistant (which is considered as an offset in nowadays CAGD). Since Schoenberg studied B-spline only on uniform knots, it did not take so long and H. Curry generalized them to nonuniform knots in 1947. In sixties, C. de Boor, who was working for General Motors, was using B-splines for geometry representation and derived a recursive evaluation of B-splines (known as de Boor algorithm), which possesses a good numerical stability in evaluation.

To find the father of NURBS (Non-Uniform Rational B-Splines) description is not a simple task. In 1968, A.R. Forrest wrote his Ph.D. thesis about the curves and surfaces for CAGD, which deeply inspired Coons research in rational curves. Consequently Coons' student K. Versprille first dealt with NURBS description. Even though NURBS is a powerful tool in CAGD, it has its disadvantages. Recently, A. Bakenov gave a new technique based on NURBS, which is called T-spline, see [1]. A main difference between T-Splines and NURBS is the existence of T-points. A T-point is a vertex where on one side, there is an isoparm[6], and on the other side, there isn't. This allows lines to end elegantly and also it simplifies the control meshes of a surface, i.e., to reduce a number of control points, see Fig. 2.1.

Although Liminig's conic construction was used to design an airplane fuselage in US aircraft company Boeing, a wing construction had used a different kind of curves developed by J. Ferguson and D. Maclaren. They joined cubic spatial curves together to create curves which were overall twice differentiable. The unassailable contribution of these curves was to interpolate a set of points. Ferguson applied piecewise monomial form and further he used cubic Hermite form defined by two endpoints with associated tangent vectors.

One side of a coin is to find good representation of a curve and the second is to focus on surfaces. Several techniques or approaches suit for curves and as well for surfaces. But there are methods only for surfaces. The most popular approach has became tensor product surface, which was first introduced by C. de Boor by his bicubic spline interpolation in 1962, see [19], and two years later followed by J. Ferguson, who was working with an array of bicubic patches interpolating a grid of points. At the same year, Coons in his technical report studied a simple formula how to fit a patch

and pioneered the modern concept of creating a sales and services network that complements the motor car.

[3]Polynomials in Bernstein form were first used by Sergei Natanovich Bernstein in a constructive proof for the Stone–Weierstrass approximation theorem. With the advent of computer graphics, Bernstein polynomials, restricted to the interval $t \in [0, 1]$, became important in the form of Bézier curves.

[4]Barry W. Boehm (1935) is an American software engineer.

[5]A Romanian mathematician (1903-1990).

[6]Lines on a NURBS surface connecting points of constant u or v coordinate values, and representing cross sections of the NURBS surface in the u or v directions.

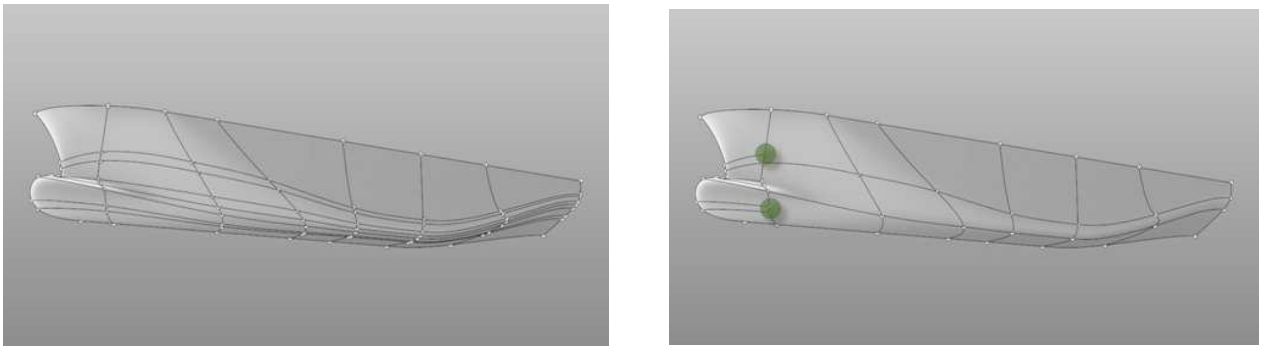


Figure 2.1: A ship hull. Left: NURBS description by 72 control points; Rigth: T-spline description by 42 control points. The pictures are undertaken from [115].

between any four arbitrary boundary curves. The Coons' method is known as the bilinearly blended Coons patch and it was used by Ford, where Coons was a consultant although he was working at MIT. A generalization was done by W. Gordon at General Motors in 1969 and in 1974 J. Gregory applied cubic boundary curves and cubic derivatives to obtain rational description.

Another method in CAGD are triangular patches, which are based on barycentric coordinates[7]. The first mention goes back to Finite Elements Method[8]. The simplest type is a linear element, which was first mentioned in Ritz-Galerkin method[9]. There are several well known techniques on triangular patches which provide desired smoothness. One of the popular is Clough and Tocher element, which uses cubic polynomial and was originally designed for FEM method. Such constructed patches ensure C^2 continuity. Another patches introduced in 1977 by Powell and Sabin are constructed by piecewise quadratic polynomials and ensures a global C^1 continuity, see [100]. A lot of geometers and mathematicians paid attention to triangular meshes, e.g., Bézier triangles, which were constructed by an automotive researcher and which started to be used in 1980's. Other approaches were developed by S. Coons, G. Farin, R. Barnhill. Today, this topic is still worth studying and is an active research area.

Further technique is subdivision curves and surfaces, which have been widely investigated. As usual there is also not clear who first introduced this method. In 1974 at the conference in Utah, G. Chaikin first presented a new technique how to generate a curve although the similar algorithm had been already done in the work of G. de Rham in 1947. This idea motivated several scientists in geometric modeling. In 1978, E. E. Catmull[10] and J. H. Clark[11] published subdivision scheme based on bi-cubic uniform B-spline, which yields C^2 continuity except at extraordinary vertices, see [10]. At the same year, D. Doo and M. Sabin came out with bi-quadratic uniform B-spline generating subdivision scheme, which is extended Chaikin's corner-cutting method for curves to surfaces

[7] Barycentric coordinates are a form of homogeneous coordinates. The system was introduced (1827) by August Ferdinand Möbius.

[8] The finite element method (FEM) is a numerical technique for finding approximate solutions of partial differential equations (PDE) as well as of integral equations. The development of the finite element method began in the middle 1950s for airframe and structural analysis and gathered improvement at the University of Stuttgart through the work of John Argyris and at Berkeley through the work of Ray W. Clough in the 1960s for use in civil engineering.

[9] Ritz-Galerkin methods are a class of methods for converting a continuous operator problem (such as a differential equation) to a discrete problem.

[10] A computer scientist and current president of Walt Disney Animation Studios and Pixar Animation Studios

[11] A prolific entrepreneur and former computer scientist. He founded several notable Silicon Valley technology companies.

and which gives C^1 continuity, cf. [23]. Both schemes work on quadrilateral meshes and thus in 1987, C. T. Loop proposed subdivision scheme on triangular meshes which provides C^2 continuity, see [75]. J.Peters and U. Reif developed mid-edge subdivision scheme working on quadrilateral mesh, see [94]. This scheme was also independently established by Habib and Warren. The previous subdivision schemes are approximation techniques. A interpolation subdivision schemes have been developed a little bit latter. The first interpolating subdivision approach was showen by Dyn, Levin and Gregory in 1990. This scheme is constructed for triangular meshes and is known as butterfly scheme. It was generalized for irregular triangulation by Zorin, Schröder and Schwelden in 1996. Recently, L. Kobbelt introduced $\sqrt{3}$ subdivision scheme for triangular meshes and also subdivision scheme for quadrilateral meshes known as Kobelt method. All interpolating subdivision techniques possess at least C^1 continuity. In 1998, Sederberg et al. in [109] introduced a new technique called NURSS (Non Uniform Rational Subdivision Surface) based on a knot insertion or recursive subdivision and used Catmull Clark or Doo Sabin subdivision technique. The idea of NURSS was improved by T-splines to T-NURCC (Non Uniform Rational Catmull-Clark surfaces with T-junctions), which gives more possibilities in geometric modeling, see [108].

During last two decades, a lot of effort was dedicated to the study of objects with Pythagorean hodograph property, which were first introduced by Farouki and Sakkalis in [37]. The state of art of this still worth studying topic is written down in the Chapter 3.

3

State of Art of Pythagorean Hodograph

> "Geometry has two great treasures: one is the Theorem of Pythagoras, and the other the division of a line into extreme and mean ratio; the first we may compare to a measure of gold, the second we may name a precious jewel."
>
> Johannes Kepler

3.1 The theorem $a^2 + b^2 = c^2$

Theorem 1.

The sum of the squares of the lengths of the two other sides of any right triangle will equal the square of the length of the hypotenuse.

Proof. The theorem can be proved geometrically using four copies of a right triangle with sides a, b and c, set inside a square with side c. The four triangles have the same area $ab/2$ and the small

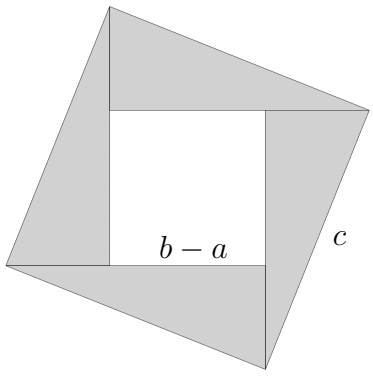


Figure 3.1: Proof of the theorem $a^2 + b^2 = c^2$

square inside has an area $(b-a)^2$. The area of the large square is therefore

$$(b-a)^2 + 4\frac{ab}{2} = (b-a)^2 + 2ab = a^2 + b^2,$$

which is a square with side c and area c^2, and therefore

$$c^2 = a^2 + b^2.$$

□

Around 4000 years ago, the Chinese and the Babylonians were aware of the fact that a triangle with the sides of $3, 4$ and 5 have to be a right triangle. Around 2500 BC, the Megalithic monuments in Egypt and Northern Europe comprised of right triangles with integer sides. During the reign of Hammurabi the Great (1790 – 1750 BC), the Mesopotamian tablet Plimpton consisted of many entries closely related to Pythagorean triples. In the period from eighth to second century BC an Indian book Baudhayana Sulba Sutra contains a list of Pythagorean triples and several statements, theorems and the geometrical proofs of the theorems for an isosceles right triangle.

Pythagoras (569–475 BC), used algebraic methods to construct Pythagorean triples. According to Sir Thomas L. Heath, there was no ascription of the theorem for nearly five centuries after Pythagoras lived. However, authors like Plutarch and Cicero attributed the theorem to Pythagoras in such a way that the attribution was widely known and accepted. In 400 BC, Plato established a method for finding Pythagorean triples which joined both algebra and geometry. Around 300 BC, in the Euclid's Elements, the oldest existing axiomatic proof of the theorem is presented.

The Chinese text Chou Pei Suan Ching written between 500 BC and 200 AD contained the visual proof of the Pythagorean theorem or "Gougu theorem" for the (3,4,5) triangle. During the Han Dynasty (202 BC – 220 AD), the Pythagorean triples appear in the Nine Chapters on the Mathematical Art together with the right triangles. The first recorded use of the theorem was in China as "Gougu theorem", and in India as the "Bhaskara theorem".

However, it is not yet confirmed whether Pythagoras was the first person, who founded the relationship between the sides of the right triangles, as no texts written by him were found. Nevertheless, the theorem has still got his name credited to it.

3.2 The use of Pythagorean Hodograph

In the middle of the last century, new engineering disciplines arose, which gave several fields of interesting problems to the scientists. One of such fields was the area of tool path and motion planing, NC[1] and CNC machining and branches close to them. These problems have been solved using Minkowski sum[2] and by theory of offsets. The Minkowski sum has been widely studied and a lot of efficient algorithms were introduced, see [26, 72].

Let us focus on an offset[3], which is often called classical offset. The term offset was known for centuries as parallel curves and surfaces (first mentioned by G. W. Leibnitz) or as an envelope curve

[1]The birth of NC (numerical control) is generally credited to John T. Parsons, a machinist and salesman at his father's machining company, Parsons Corp. In 1942, he was told that helicopters were going to be the "next big thing" by the former head of Ford Trimotor production, Bill Stout.

[2]Minkowski sum was established by Hermann Minkowski in 1903.

[3]There are other branches where the word offset is used, e.g. printing, greenhouse gas emission (carbon offset), computer science, electronic engineering (DC offset).

and surface (first studied in optics) or as a canal surface (in classical differential geometry, e.g. in [25]). In 1960s, an offset was introduced by M. Sabin in [104] and an analysis of offsets has started by J. Hoschek in [52, 53] and was extended by R.T. Farouki in [30], Y. J. Chen and B. Ravani in [12]. An offset was defined to be a curve or a surface (for these cases we can shortly use hypersurfaces) given by following expression

$$\mathbf{p}_o = \mathbf{p} + d\frac{\mathbf{n}}{|\mathbf{n}|}, \quad d \in \mathbb{R}, \tag{3.1}$$

where $\mathbf{p}$ is a parametric expression of an input hypersurface and $\mathbf{n}$ is a normal vector field of a generating hypersurface $\mathbf{p}$. In this way, we can construct an offset at arbitrary distance d. The disadvantage of such description is that for a polynomially or rationally described hypersurface we do not obtain polynomial or rational expression of an offset in general. Therefore, all techniques concerning offset computation mentioned in [12, 30, 52, 53] are based on approximation or interpolation of offsets.

Because all CAD (Computer Aided Design) and CAM (Computer Aided Manufacture) systems use NURBS (Non-Uniform Rational B-Spline) description, which is a standard form how to keep or represent curves, surfaces and solid objects, the motivation was to identify such curves and surfaces which possess an offset with rational parametrization. Despite it looks like an effortless task the scientists have spent twenty years solving this topic.

In [37], Farouki and Sakkalis came out with a quite simple idea. Only one term, i.e., $|\mathbf{n}|$ in the expression (3.1), influences the rationality of an offset parametrization. They investigated polynomial curves and arrived at the condition

$$x'^2(t) + y'^2(t) = \sigma^2(t), \tag{3.2}$$

where $(x(t), y(t))^\top, t \in \mathcal{I} \subset \mathbb{R}$, describes a polynomial parametric curve and $\sigma(t) \in \mathbb{R}[t]$. Consequently, according to theorem for polynomial triples (see [69]), they derived the functions $x'(t), y'(t)$ fulfilling the condition (3.2), namely

$$\begin{aligned} x'(t) &= w(t)[u^2(t) - v^2(t)], \\ y'(t) &= 2w(t)u(t)v(t), \end{aligned} \tag{3.3}$$

where $w(t), u(t), v(t)$ are polynomials. Since the condition (3.2) is identical with the Pythagorean theorem, it also gave to rise to the name **Pythagorean Hodograph**[4] (or abbreviated to PH) curves.

This interesting and useful property started a deep investigation of PH curves. They do not only possess a rational description of offsets but also a polynomial arc-length function, which is useful in mechanical engineering, e.g., an easier way how to control the speed of the cutting tool in CNC machining.

In [37], it has been also shown that the simplest polynomial PH curves are cubics (except lines). Moreover, it has been proved that PH cubic is only one (Tschirnhausen cubic). Unfortunately, it does not have enough flexibility in a practical use in comparison with a standard cubic. This PH cubic behavior has started the investigation of PH curves of higher degrees, especially five, seven and nine. In [13, 29, 36, 40, 58, 80, 81, 83, 85, 112, 113] several techniques of PH splines construction can be found, typically with G^1, C^1 or C^2 continuity. Recently, a quartic Pythagorean Hodograph were derived from control polygon and were used for G^1 Hermite interpolation, see [116]. In [55], Jaklič

[4]The word Hodograph means a curve of which the radius vector represents the velocity of a moving particle, from Greek hodos is a way.

et al. studied Lagrange interpolation and they formulated a conjecture that PH curve of degree n can, under some natural restrictions on data points, interpolate up to $n+1$ points.

It did not take so long and an idea to investigate rational curves possessing rational offsets appeared. Pottmann in [98] studied such curves and gave a bright new construction in a dual form and arrived at the expression

$$\begin{aligned} x(t) &= 2ab(a'b-ab')fg - \tfrac{1}{2}(a^4-b^4)(f'g-fg'), \\ y(t) &= (a^2-b^2)(a'b-ab')fg + ab(a^2+b^2)(f'g-fg'), \\ w(t) &= (a^2+b^2)(a'b-ab')g^2, \end{aligned}$$

where a plane curve is expressed in the form $(x(t)/w(t), y(t)/w(t))^\top$ and the $a(t), b(t)$ and $f(t), g(t)$ are prime polynomials. Recently, Šír et al. in [111] shown that all hypocycloids and epicycloids yield rational offset.

Although the polynomial and rational PH curves have rational offsets, they differ significantly in their arc-length function. The integration of the polynomial speed $\sigma(t)$ yields a polynomial arc length function for polynomial PH curves, but for rational PH curves, integration of the rational function $\sigma(t)$ needs partial fraction expansion and an arc length function contains transcendental and rational term in general.

In [89], Peternell and Pottmann gave an interpretation for the construction of the rational curves with rational offsets using Laguerre geometry, where oriented lines and circles in plane are basis elements. The orientation of each element is fixed by associating a field of normal vectors with it. Points are treated as circles with zero radius.

In [38], Farouki and Sakkalis have introduced the spatial PH curves. Although the generalization of standard curves from a plane to a space is not a big deal, to shift up PH curves to a space is not so easy. Similar task was solved, i.e., we need to find polynomial solutions of the spatial Pythagorean hodograph condition

$$x'^2(t) + y'^2(t) + z'^2(t) = \sigma^2(t),$$

where $(x(t), y(t), z(t))^\top$, $t \in I \subset \mathbb{R}$ describes spatial polynomial parametric curve and $\sigma(t)$ is polynomial.

The spatial PH curves are distinguished from plane PH curve in practical use. The offset of a spatial curve is meant as a canal surface (or pipe or tubular surface), which can be described as an envelope of spheres. Peternell and Pottmann in [90] showed that any rational spine curve $\mathbf{r}(t)$ and a rational radius function $d(t)$ possesses rational parametrization of a canal surface. Let us emphasize that to get a real envelope surface, the derivative of the spine curve and radius function have to satisfy the condition $|\mathbf{r}'(t)|^2 \geq d'^2(t)$ and moreover the non-negative function $|\mathbf{r}'(t)|^2 - d'^2(t)$ has to be possible to rewrite as a sum of squares, which Peternell and Pottmann proved in [90]. This decomposition can be determined exactly only for PH curves.

Another significant property of the spatial PH curves is that they are automatically equipped with the rational frames, which have been widely studied in [14, 31, 34, 35, 79, 117, 118]. Therefore, the spatial PH curves are used for construction of PH spline curves, as in planar case. There are similar approaches as for plane PH curves, i.e., they are investigated in the manner of control polygon. The simplest nontrivial spatial PH curves are PH cubics and they admit characterization directly in the term of the geometry of their Bézier control polygons. This is consequence of special intrinsic geometry of spatial PH cubics, i.e., they are all helical curves.

In [38], there is further mentioned how the first derivatives have to look like to fulfill the Pythagorean-hodograph condition, i.e.

$$\begin{aligned} x'(t) &= h(t)[u^2(t) - v^2(t) - w^2(t)], \\ y'(t) &= 2h(t)u(t)v(t), \\ z'(t) &= 2h(t)u(t)w(t), \end{aligned} \tag{3.4}$$

for polynomials $h(t), u(t), v(t), w(t)$. However, the condition (3.4) is a sufficient but not a necessary condition for a spatial hodograph $(x'(t), y'(t), z'(t))$ to satisfy the Pythagorean condition. For example, the hodograph given by $x'(t) = (1-t)^2, y'(t) = t^2, z'(t) = 1$ vanishes Pythagorean-hodograph condition, i.e. $\sigma(t) = \sqrt{2}(t^2 - t + 1)$ but it can not be written by above introduced equations (3.4). The reason is that it is invariant under rotation about the x axis, but not about the remaining y and z axes, or axes of arbitrary orientation in space.

Dietz et al. in [22] gave a characterization of Pythagorean condition in the form

$$\begin{aligned} x'(t) &= u^2(t) + v^2(t) - p^2(t) - q^2(t), \\ y'(t) &= 2[u(t)q(t) + v(t)p(t)], \\ z'(t) &= 2[v(t)q(t) - u(t)p(t)], \\ \sigma(t) &= u^2(t) + v^2(t) + p^2(t) + q^2(t), \end{aligned} \tag{3.5}$$

for prime real polynomials $u(t), v(t), p(t), q(t)$. This result can be reformulated using quaternions[5] with basis elements $\mathbf{i}, \mathbf{j}, \mathbf{k}$, which multiplication is determined by the set of rules

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$$

with noncommutative multiplication

$$\mathbf{ij} = -\mathbf{ji} = \mathbf{k}, \quad \mathbf{jk} = -\mathbf{kj} = \mathbf{i}, \quad \mathbf{ki} = -\mathbf{ik} = \mathbf{j}.$$

The hodograph of a curve is obtained as

$$\begin{aligned} (x'(t), y'(t), z'(t))^\top = \mathcal{A}(t)\mathbf{i}\mathcal{A}^*(t) &= [u^2(t) + v^2(t) - p^2(t) - q^2(t)]\mathbf{i} \\ &\quad +2[u(t)q(t) + v(t)p(t)]\mathbf{j} \\ &\quad +2[v(t)q(t) - u(t)p(t)]\mathbf{k}, \end{aligned} \tag{3.6}$$

where $\mathcal{A} = u(t) + v(t)\mathbf{i} + p(t)\mathbf{j} + q(t)\mathbf{k}$ and $\mathcal{A}^* = u(t) - v(t)\mathbf{i} - p(t)\mathbf{j} - q(t)\mathbf{k}$ is conjugated.

Recent investigation of spatial Pythagorean curves gave a birth to double Pythagorean hodograph (shortly DPH) introduced by Beltran and Monterde in [4]. The DPH curves are such spatial curves $\mathbf{r}(t)$ with the property that $|\mathbf{r}'(t)|$ and $|\mathbf{r}'(t) \times \mathbf{r}''(t)|$ are both polynomial in parameter t. Farouki et al. in [32] and [33] have studied spatial DPH curves using quaternion and Hopf[6] map. They have found out that all helical PH curves are DPH curves, which encompass all PH cubics and all helical PH quintics, although non-helical DPH curves of higher order exist.

[5] Quaternion algebra was introduced by Irish mathematician Sir William Rowan Hamilton in 1843 and is a four dimensional extension of complex numbers. Important precursors to this work included Euler's four-square identity (1748) and Olinde Rodrigues' parameterization of the general rotation by four parameters (1840), but neither of these authors treated the four-parameter rotations as an algebra. Gauss had also discovered quaternions in 1819, but this work was only published in 1900

[6] In the mathematical field of topology, the Hopf fibration (also known as the Hopf bundle or Hopf map) describes a 3-sphere (a hypersphere in four-dimensional space) in terms of circles and an ordinary sphere. Discovered by Heinz Hopf in 1931, it is an influential early example of a fiber bundle.

As in a plane where polynomial PH curves were generalized to rational PH curves, it has been done also for spatial PH curves. Recently, Farouki and Šír in [41] have introduced rational spatial PH curves as an edge of regression of envelope of osculating plane.

Rotation Minimizing Frame (abbreviated to RMF) is another topic where spatial PH curves are used. RMF was introduced in [61] and commonly used in computer graphics, sweep surface, tube surface etc. The idea is to investigate an adapted frame (it is the frame $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$, where $\mathbf{e}_1$ is the tangent of the curve and $\mathbf{e}_i \cdot \mathbf{e}_i = 1$ and $\mathbf{e}_i \cdot \mathbf{e}_j = 0$) such that the rotation rate of $\mathbf{e}_2$ and $\mathbf{e}_3$ along the curve is minimal. The computation of the RMF means to solve an ordinary differential equation (for more information see [5] and [59]), which is solvable in the case of PH curves, i.e., we obtain the result as rational or transcendental function.

Choi and Han in [14] have used Euler-Rodriques parameters to describe frame of PH curve by rational function and using quaternion calculus they have arrived at

$$\mathbf{e}_1(t) = \frac{\mathcal{A}(t)\,\mathbf{i}\,\mathcal{A}^*(t)}{|\mathcal{A}(t)|^2},\ \mathbf{e}_2(t) = \frac{\mathcal{A}(t)\,\mathbf{j}\,\mathcal{A}^*(t)}{|\mathcal{A}(t)|^2},\ \mathbf{e}_3(t) = \frac{\mathcal{A}(t)\,\mathbf{k}\,\mathcal{A}^*(t)}{|\mathcal{A}(t)|^2}. \tag{3.7}$$

Further, in [79], it has been noted the existence of rational frame on spatial PH curves. Unlike the Frenet frame, the Euler-Rodrigues frame (for brevity ERF, which is described by Euler-Rodrigues parameters) is uniquely defined at each point of regular spatial PH curve, including inflection and varies smoothly along the curve.

Moreover, Choi and Han in [14] have characterized the angular velocity of the ERF relative to RMF for spatial cubic and quintic curves. They have also shown for PH cubic that among Frenet frame and ERF the constant angle occurs, and the ERF coincides with RMF if and only if the PH cubic is planar. They proved that no RMF exists on spatial PH cubics and the simplest non-planar PH curves with ERF that can be RMF is thus of degree five.

The interesting branch in RMF topic is to focus on the rationality of RMF. Han in [48] has proved that there are no rational RMF on cubic curves. Recently, Farouki et al. in [31] have studied quintic spatial curves which provide Rational RMF (abbreviated to RRMF) and Farouki and Sakkalis in [39] have studied polynomial spatial curves, which possess RRMF.

The Minkowski Pythagorean Hodograph curves, or MPH curves for brevity, was first introduced by Moon in [82]. Their distinctive feature is that Pythagorean condition is treated under metric of the Minkowski space $\mathbb{R}^{2,1}$. The MPH curves are such curves which allow the medial axis transform[7] (abbreviated to MAT) of a planar domain to be specified in such way that the boundary of the domain is exactly expressable by rational curves.

The Pythagorean Hodograph condition is modified in the Minkowski space to the condition

$$x'^2(t) + y'^2(t) - r'^2(t) = \sigma^2(t), \tag{3.8}$$

for a polynomial curve $(x(t), y(t), r(t))^\top$, $\sigma(t) \in \mathbb{R}[t]$, and is satisfied if and only if $x'(t), y'(t), r'(t)$ can be written in the form

$$\begin{aligned} x'(t) &= u^2(t) + v^2(t) - p^2(t) - q^2(t), \\ y'(t) &= 2[u(t)p(t) - v(t)q(t)], \\ r'(t) &= 2[u(t)v(t) - p(t)q(t)], \\ \sigma(t) &= u^2(t) - v^2(t) + p^2(t) - q^2(t), \end{aligned} \tag{3.9}$$

[7] The Medial Axis of an object is the set of all points having more than one closest point on the object's boundary. Originally referred to as the topological skeleton, it was introduced by Blum, see [7], as a tool for biological shape recognition. In mathematics, the closure of the medial axis is known as the cut locus. The Medial Axis together with the associated radius function of the maximally inscribed discs is called the Medial Axis Transform. The Medial Axis Transform is a complete shape descriptor, meaning that it can be used to reconstruct the shape of the original domain.

where $u(t), v(t), p(t), q(t)$ are polynomials as in previous discussion. The computation of medial axis can be found in [15, 62, 63, 66]. Recently, Kosinka and Lávička in [65] generalized polynomial class of Minkowski Pythagorean hodograph curves to rational. They showed that any rational Minkowski Pythagorean hodograph curve can be obtained in terms of its associated planar rational Pythagorean hodograph curve and an additional rational function.

Not only PH curves have been generalized to spatial PH curves and also MPH curves. Choi et al. in [16] have investigated MPH curves using Clifford[8] algebra and they have widely studied their behavior in Minkowski space $\mathbb{R}^{3,1}$.

If we return to the formula of an offset, it is obvious to see that the normal consists of the terms of velocity of a given curve in in the planar case. This property gave the name Pythagorean Hodograph. Looking for the surfaces which possess rational offsets we arrive at the similar condition as (3.1), i.e.,

$$\mathbf{n}(u,v) \cdot \mathbf{n}(u,v) = \sigma^2(u,v), \quad (u,v) \in \mathbb{R}^2, \tag{3.10}$$

where $\mathbf{n}(u,v)$ is normal of a surface $\mathbf{p}(u,v)$ and $\sigma(u,v) \in \mathbb{R}(u,v)$. Such class of surfaces which fulfill the condition (3.10) is called surfaces with **Pythagorean Normal** (shortly PN) and it has been first studied by Pottman in [98]. Further, this class of surfaces was widely investigated and used for interpolating and approximating techniques, for more detail see [77, 67, 68] and [70].

Like PH curves have been generalized to PN surfaces, the same have been done for MPH curves. In [64], Kosinka and Jüttler have established MOS (Medial surface transform which Obeys the Sum-of-square-condition) surfaces. The Medial Surface Transform (MST) of a volume is the set of surface patches (or curves segment) in four dimensional Minkowski space $\mathbb{R}^{3,1}$. Every point of MST represents the center and the radius of a maximal sphere inscribed into the domain. The advantage of MOS surfaces is analogous to MPH curves. If MST of volume is described by MOS surface then associated envelope and all offsets admit exact rational parametrization. Recently, Peternell et al. in [87] have proved that quadratic triangular Bézier surfaces in $\mathbb{R}^{3,1}$ are MOS surfaces. Further, several techniques concerning Minkowski metric have been developed, see [86, 88].

In 1992, Brechner introduced a general offset, which was a generalization of a classical offset, see [8]. The motivation was in 3-axis milling in comparison with 5-axis milling, which was crucial for classical offset. General offset can be expressed in the manner of convolution. The convolution hypersurface $\mathbf{c} = \mathbf{a} \star \mathbf{b}$ is defined as $\mathbf{c} = \{\mathbf{A} + \mathbf{B} | \mathbf{A} \in \mathbf{a}, \mathbf{B} \in \mathbf{b} \text{ and } \alpha(\mathbf{A}) || \beta(\mathbf{B})\}$, where $\alpha(\mathbf{A}), \beta(\mathbf{B})$ are tangent hyperplanes of smooth hypersurfaces $\mathbf{a}, \mathbf{b}$ at the point $\mathbf{A} \in \mathbf{a}, \mathbf{B} \in \mathbf{b}$.

According to the definition of convolution the classical offset can be treated as a convolution of a circle or sphere with arbitrary curve or surface. The general offset can be considered as a convolution of two curves or surfaces (generally hypersurfaces), where one of them often describes the shape of a cutting tool. Also in this point of view, we can ask whether an output object (convolution object or general offset) is polynomial or rational. In general, it is not true that convolution of polynomial or rational hypersurface with polynomial or rational one is again rational. This problem opened several questions, e.g., for which polynomial or rational hypersurface we obtain rational convolution with arbitrary polynomial or rational hypersurface. Sampoli et al. in [105] have introduced a subclass of PN class called surfaces with Linear Normal (abbreviated to LN). The computational advantage is that LN surfaces possess always rational convolution with arbitrary rational surfaces. Inspired by

[8]Clifford algebras are a type of associative algebra. They can be thought of as one of the possible generalizations of the complex numbers and quaternions. The theory of Clifford algebras is intimately connected with the theory of quadratic forms and orthogonal transformations. Clifford algebras have important applications in a variety of fields including geometry and theoretical physics. They are named after the English geometer William Kingdon Clifford.

convolution approach and with the help of Gröbner basis Lávička and Bastl in [71] classified the surfaces with respect to the rationality of the convolution. They identified classes of parametrizations of hypersurfaces which yield always rational convolution with arbitrary rational parametrization of hypersurfaces (General Rational Convolution – GRC) and parametrization of hypersurfaces, which possess rational convolution in special cases (Special Rational Convolution – SRC). Moreover, they have proved that the convolution surfaces of non-developable quadratic Bézier surfaces and an arbitrary rational surface are always rational.

Part II

PYTHAGOREAN HODOGRAPH CURVES

4

G^1 Hermite interpolation

In this chapter we focus on G^1 Hermite interpolation by PH cubic. We extend some results mentioned in the paper of Walton and Meek, see [80], especially we discuss for which input data the interpolating arc of PH cubic exists. Further, we deal with the quality of PH cubic interpolant, i.e., whether it contains a loop or not.

4.1 Hermite interpolation by Tschirnhausen cubic

In this section we want to describe all initial data for G^1 Hermite interpolation problem for which an interpolating arc of the Tschirnhausen cubic exists. We use some basic facts from [37, 80] and [56].

First, we recall some basic notation about PH curves.

Definition 2. *A polynomial parametric curve* $\mathbf{r}(t) = (x(t), y(t))^\top$, $t \in I \subset \mathbb{R}$ *is called a* Pythagorean Hodograph curve *(or* PH curve*) if there exists a polynomial* $\sigma(t) \in \mathbb{R}[t]$ *such that*

$$x'(t)^2 + y'(t)^2 = \sigma(t)^2.$$

It follows from the definition that coordinates of hodographs of PH curves and $\sigma(t)$ form Pythagorean triples. K. K. Kubota proved in [69] the following:

Theorem 3 (Kubota)**.** *Three real polynomials* $a(t), b(t)$ *and* $c(t)$*, where* $\max[\deg(a), \deg(b)] = \deg(c) > 0$*, satisfy the Pythagorean condition* $a^2(t) + b^2(t) = c^2(t)$ *if and only if they can be expressed in terms of real polynomials* $u(t), v(t)$ *and* $w(t)$ *in the form*

$$\begin{aligned} a(t) &= w(t)[u^2(t) - v^2(t)], \\ b(t) &= 2w(t)u(t)v(t), \\ c(t) &= w(t)[u^2(t) + v^2(t)]. \end{aligned}$$

This theorem directly implies the following lemma.

Lemma 4. *The polynomial curve corresponding to the Pythagorean hodograph is of degree* $n = \lambda + 2\mu + 1$*, where* $\lambda = \deg[w(t)]$ *and* $\mu = \max[\deg[u(t)], \deg[v(t)]]$.

Proof. Can be found in [37]. □

According to Lemma 4 the simplest nontrivial PH curves are cubics. Hence, we consider two linear polynomials $u(t), v(t)$ given in Bernstein-Bézier form as

$$\begin{aligned} u(t) &= u_0(1-t) + u_1 t, \\ v(t) &= v_0(1-t) + v_1 t, \quad t \in \mathbb{R}, \end{aligned} \tag{4.1}$$

where we assume that the ratios $u_0 : u_1$ and $v_0 : v_1$ are unequal. We call the equation (4.1) as *preimage* of PH cubic.

As initial data we consider two boundary points $\mathbf{P}_0$, $\mathbf{P}_3$ with two associated unit tangent vectors $\mathbf{t}_0$, $\mathbf{t}_3$. Moreover, we denote angles

$$\theta_0 = \angle(\mathbf{t}_0, \mathbf{P}_3 - \mathbf{P}_0),\ \theta_3 = \angle(\mathbf{P}_3 - \mathbf{P}_0, \mathbf{t}_3).$$

If we fix the angle θ_0 to be in the interval $[-\pi, 0]$, which is always possible, then

$$\theta_0 = -\arccos\left(\mathbf{t}_0 \cdot \frac{\mathbf{P}_3 - \mathbf{P}_0}{||\mathbf{P}_3 - \mathbf{P}_0||}\right). \tag{4.2}$$

The computation of the corresponding $\theta_3 \in [0, 2\pi)$ depends on the orientation of the initial system which is determined by the signs of the plane cross products

$$w_0 = \mathbf{t}_0 \times (\mathbf{P}_3 - \mathbf{P}_0), \quad w_3 = (\mathbf{P}_3 - \mathbf{P}_0) \times \mathbf{t}_3. \tag{4.3}$$

Namely,

$$\theta_3 = \arccos\left(\mathbf{t}_3 \cdot \frac{\mathbf{P}_3 - \mathbf{P}_0}{||\mathbf{P}_3 - \mathbf{P}_0||}\right) \quad \text{or} \quad \theta_3 = 2\pi - \arccos\left(\mathbf{t}_3 \cdot \frac{\mathbf{P}_3 - \mathbf{P}_0}{||\mathbf{P}_3 - \mathbf{P}_0||}\right), \tag{4.4}$$

if the signs of w_0 and w_3 are equal (i.e., $w_0 w_3 > 0$) or not equal (i.e., $w_0 w_3 < 0$), respectively. The special case $w_0 w_3 = 0$ will be discussed below. In the following text, whenever we consider angles θ_0, θ_3 to be the initial data for a Hermite interpolation problem, we always mean that they were obtained using (4.2) and (4.4).

Definition 5. *Let $\mathbf{P}_0$, $\mathbf{P}_3$ be distinct points in $\mathbb{R}^2$, $\mathbf{t}_0$, $\mathbf{t}_3$ two unit tangent vectors associated to these points and θ_0, θ_3 the corresponding angles computed using* (4.2) *and* (4.4)*. Then an arc of Tschirnhausen cubic interpolating these G^1 Hermite interpolation data is called* TC-interpolant*.*

Moreover, the set of all pairs $(\theta_0, \theta_3) \in [-\pi, 0] \times [0, 2\pi)$ taken together with $\mathbf{P}_0$, $\mathbf{P}_3$ as initial data for G^1 Hermite interpolation problem for which TC-interpolant exists is called the domain of definition of TC-interpolant*.*

Farouki and Sakkalis in [37] described conditions on control points $\mathbf{P}_0$, $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$, lengths of the control polygon legs $d_{i,i+1} = ||\mathbf{P}_{i+1} - \mathbf{P}_i||$, $i = 0, 1, 2$, and their angles $\theta_i = \angle(\mathbf{P}_{i-1} - \mathbf{P}_i, \mathbf{P}_{i+1} - \mathbf{P}_i)$, $i = 1, 2$, which ensure that the corresponding Bézier cubic has a Pythagorean hodograph, see Fig. 4.1. Namely, it has to hold that

$$\theta_1 = \theta_2 \quad \text{and} \quad d_{01} d_{23} = d_{12}^2.$$

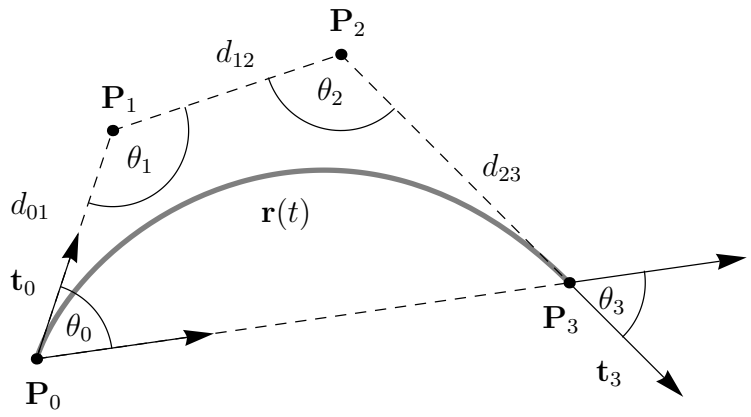


Figure 4.1: G^1 Hermite data and interpolating Bézier cubic with its control points $\mathbf{P}_0, \ldots, \mathbf{P}_3$, control polygon, angles $\theta_0, \ldots, \theta_3$ and control polygon legs d_{01}, d_{12}, d_{23}.

Using these results, Meek and Walton in [80] derived a method how to compute the lengths d_{01}, d_{23} for given initial Hermite data such that these conditions are fulfilled. Assuming that the control points $\mathbf{P}_0$ and $\mathbf{P}_3$ lie on the real axis and $D = \sqrt{d_{23}/d_{01}}$, we denote by $D_\pm(\theta_0, \theta_3)$ the roots of

$$\sin(\theta_0) + D \sin\left(\frac{\theta_0 + \theta_3}{2}\right) + D^2 \sin(\theta_3) = 0. \tag{4.5}$$

Then we get

$$d_{01}(\theta_0, \theta_3) = \frac{||\mathbf{P}_3 - \mathbf{P}_0||}{G_\pm(\theta_0, \theta_3)}, \tag{4.6}$$

where

$$G_\pm(\theta_0, \theta_3) = \cos(\theta_0) + D_\pm(\theta_0, \theta_3) \cos\left(\frac{\theta_0 + \theta_3}{2}\right) + D_\pm^2(\theta_0, \theta_3) \cos(\theta_3).$$

Before we start to investigate the domain of definition of TC-interpolant we have to discuss some special cases. Firstly, if $(\theta_0, \theta_3) = (\alpha, \beta)$ where $\alpha \in \{-\pi, 0\}$ and $\beta \in \{0, \pi\}$, then the corresponding TC-interpolant degenerates to a line segment and we exclude these cases from our considerations.

Concerning other special cases, the equation (4.5) degenerates to the linear one if $\theta_3 \in \{0, \pi\}$. Moreover, if $w_0 w_3 = 0$, we are not able to decide about the orientation of the initial system – this happens when the vectors $\mathbf{t}_0$, $\mathbf{P}_3 - \mathbf{P}_0$ or $\mathbf{t}_3$, $\mathbf{P}_3 - \mathbf{P}_0$ are collinear, i.e., $\theta_0 \in \{-\pi, 0\}$ or $\theta_3 \in \{0, \pi\}$, respectively. The existence of TC-interpolants for these special cases is summarized in Table 4.1.

Further, we want to cover all initial Hermite data which can occur and discuss the existence of TC-interpolants for these data. It is enough to consider initial angles (θ_0, θ_3) from the domain $\Gamma = (-\pi, 0) \times (0, 2\pi)$, as the existence of TC-interpolants for (θ_0, θ_3) on the boundary of Γ is described in Table 4.1. We look for such pairs $(\theta_0, \theta_3) \in \Gamma$ fulfilling $G_+(\theta_0, \theta_3) > 0$ or $G_-(\theta_0, \theta_3) > 0$. The situation is influenced by the fact that the discriminant of (4.5) is not non-negative for all $(\theta_0, \theta_3) \in \Gamma$. We obtain that the domain where $G_+(\theta_0, \theta_3) > 0$ is bounded by the curves (see Fig. 4.2 (left))

$$\begin{aligned}
\theta_3 &= f_1(\theta_0) = \theta_0 + \tfrac{4}{3}\pi, \quad \theta_0 \in (-\pi, -\tfrac{1}{6}\pi), \\
\theta_3 &= f_2(\theta_0) = \theta_0 + \tfrac{8}{3}\pi, \quad \theta_0 \in (-\tfrac{5}{6}\pi, -\tfrac{2}{3}\pi), \\
\theta_3 &= f_3(\theta_0) = \theta_0 + \alpha_+(\theta_0), \quad \theta_0 \in (-\tfrac{5}{6}\pi, 0), \\
\theta_3 &= f_4(\theta_0) = \theta_0 + \alpha_-(\theta_0), \quad \theta_0 \in (-\tfrac{1}{6}\pi, 0),
\end{aligned} \tag{4.7}$$

Table 4.1: Existence of TC-interpolants for (θ_0, θ_3) on the boundary of Γ.

	D	G	TC-interpolant exists for
$\theta_3 = 0$	$-2\cos\left(\frac{\theta_0}{2}\right)$	$1 + 2\cos(\theta_0)$	$\theta_0 \in \left(\frac{-2\pi}{3}, 0\right)$
$\theta_3 = \pi$	$-2\sin\left(\frac{\theta_0}{2}\right)$	$-1 + 2\cos(\theta_0)$	$\theta_0 \in \left(\frac{-\pi}{3}, 0\right)$
$\theta_0 = 0$	$-\frac{1}{2\cos\left(\frac{\theta_3}{2}\right)}$	$1 - \frac{1}{2+2\cos(\theta_3)}$	$\theta_3 \in \left(0, \frac{2\pi}{3}\right) \cup \left(\frac{4\pi}{3}, 2\pi\right)$
$\theta_0 = -\pi$	$\frac{1}{2\sin\left(\frac{\theta_3}{2}\right)}$	$-1 + \frac{1}{4\sin^2\left(\frac{\theta_3}{2}\right)}$	$\theta_3 \in \left(0, \frac{\pi}{3}\right) \cup \left(\frac{5\pi}{3}, 2\pi\right)$

where

$$\begin{aligned}\alpha_\pm(\theta_0) &= 4\arctan\left(-7\cot\left(\tfrac{\theta_0}{2}\right) \pm 4\sqrt{3}\sqrt{\cot\left(\tfrac{\theta_0}{2}\right)^2} + \right.\\ &\left. + \sqrt{\csc\left(\tfrac{\theta_0}{2}\right)^2\left(49 + 48\cos(\theta_0) \pm 28\sqrt{3}\sqrt{\cot\left(\tfrac{\theta_0}{2}\right)^2}\right)\sin(\theta_0)}\right).\end{aligned}$$

Thus, $G_+(\theta_0, \theta_3) > 0$ if and only if $(\theta_0, \theta_3) \in \Gamma_+^{\mathrm{PH}}$, where Γ_+^{PH} is of the form[1]

$$\begin{aligned}\Gamma_+^{\mathrm{PH}} = \Gamma \cap &\left(\left((-\pi, -\tfrac{\pi}{6}) \times (0, f_1)\right) \cup \left((-\tfrac{5}{6}\pi, -\tfrac{2}{3}\pi) \times (f_3, f_2)\right) \cup\right.\\ &\left. \cup \left((-\tfrac{5}{6}\pi, 0) \times (0, f_4)\right) \cup \left((-\tfrac{2}{3}\pi, 0) \times (f_3, 2\pi)\right)\right).\end{aligned}$$

Similarly, the domain where $G_-(\theta_0, \theta_3) > 0$ is bounded by the same curves $\theta_3 = f_1(\theta_0)$, ..., $\theta_3 = f_4(\theta_0)$ (cf. (4.7), only the domain of definition is changed in this case) and one new curve $\theta_3 = f_5(\theta_0) = \theta_0 + \frac{2}{3}\pi, \theta_0 \in (-\frac{2}{3}\pi, 0)$ (see Fig. 4.2 (middle)). Thus, $G_-(\theta_0, \theta_3) > 0$ if and only if $(\theta_0, \theta_3) \in \Gamma_-^{\mathrm{PH}}$, where

$$\begin{aligned}\Gamma_-^{\mathrm{PH}} = \Gamma \cap &\left(\left((-\pi, -\tfrac{5}{6}\pi) \times (f_2, 2\pi)\right) \cup \left((-\tfrac{5}{6}\pi, 0) \times (f_3, 2\pi)\right) \cup\right.\\ &\left. \cup \left((-\tfrac{1}{6}\pi, 0) \times (f_1, f_4)\right) \cup \left((-\tfrac{2}{3}\pi, 0) \times (0, f_5)\right)\right).\end{aligned}$$

Finally, we can formulate

Proposition 6. *For given Hermite interpolation data $\mathbf{P}_0$, $\mathbf{P}_3$, $\mathbf{t}_0$, $\mathbf{t}_3$, where $\mathbf{P}_0$ and $\mathbf{P}_3$ are distinct points on the real axis and $\theta_0 = \angle(\mathbf{t}_0, \mathbf{P}_3 - \mathbf{P}_0) \in (-\pi, 0)$, $\theta_3 = \angle(\mathbf{P}_3 - \mathbf{P}_0, \mathbf{t}_3) \in (0, 2\pi)$ such that*

$$(\theta_0, \theta_3) \in \Gamma^{\mathrm{PH}} = \Gamma_+^{\mathrm{PH}} \cup \Gamma_-^{\mathrm{PH}},$$

there exists at least one TC-interpolant that matches the given initial Hermite data. The domain Γ^{PH} describes all possible initial Hermite data *for which at least one TC-interpolant fulfilling these data exists.*

Remark 7. *Let us point out that*

1. *if we consider $(\theta_0, \theta_3) \in (-\pi, 0) \times (0, \pi)$, a TC -interpolant exists if $-\theta_0 + \theta_3 < \frac{4}{3}\pi$ holds – this result was also proved in [56] where it describes the domain of angles θ_0, θ_3 for TC-interpolant without a loop,*

2. *if $-\theta_0 + \theta_3 < \frac{2}{3}\pi$, then there exist two TC-interpolants fulfilling the given initial Hermite data – one without and one with a loop.*

[1] For the sake of simplicity, we use a notation $(a, b) \times (f, g)$ for the description of the domain $\{(x, y) \in \mathbb{R}^2 : a < x < b \wedge f(x) < y < g(x)\}$ between curves $y = f(x)$ and $y = g(x)$ restricted on the interval (a, b).

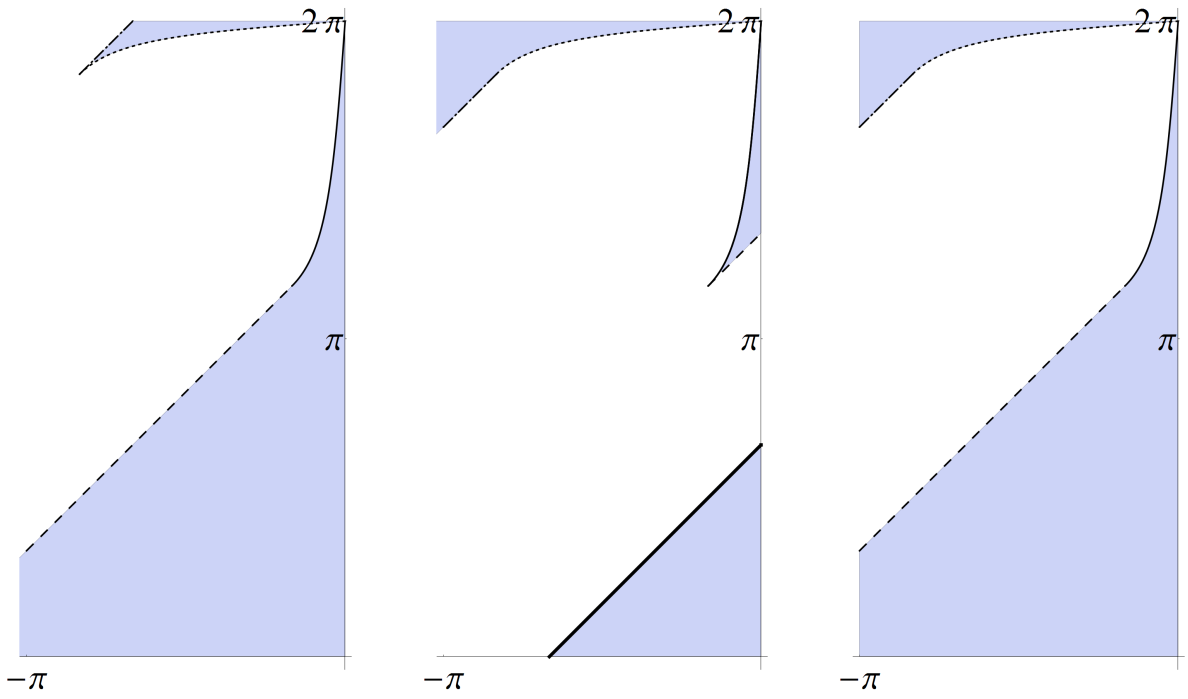


Figure 4.2: Left: Γ^{PH}_{+}; Middle: Γ^{PH}_{-}; Right: Γ^{PH}. All domains (light blue) are in the $\theta_0\theta_3$-plane and are bounded by the curves f_1 (dashed), f_2 (dot-dashed), f_3 (dotted), f_4 (thin solid), f_5 (thick solid).

4.2 TC-interpolant with and without a loop

In this section, we want to analyze the number of TC-interpolants satisfying given initial data and their quality, i.e., if they contain a loop or not.

The discussion of inflections, cusps, arches and loops of general cubic Bézier curves with respect to a control polygon was done by Stone and DeRose in [110]. Now, we want to summarize results of a general approach used in [110] which we will adapt to our case of TC-interpolants in the next subsections.

Let us consider the control polygon of a cubic Bézier curve given by the points $\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{Q}_3$, where the first three points are fixed to $\mathbf{Q}_0 = (0,0)^\top, \mathbf{Q}_1 = (0,1)^\top, \mathbf{Q}_2 = (1,1)^\top$. By moving the point $\mathbf{Q}_3 = (x,y)^\top$ the control polygon is changed and consequently the cubic Bézier curve. As usual, we deal with standard Bézier curves (cf. e.g. [95]) defined for a parameter in the interval $[0,1]$. The domains where cusps, inflection points, loops or arches occur are bounded by conic sections. Our interest is focused on the domain where a loop arises. This domain is described by the inequalities

$$x^2-3x+3y \geq 0 \quad \wedge \quad \frac{x^2}{4}+\frac{x}{2}-y+\frac{3}{4} > 0 \quad \wedge \quad x^2+y^2+xy-3x \geq 0 \quad \wedge \quad x \leq 1\,, \tag{4.8}$$

see Fig 4.3.

4.2.1 TC-interpolants containing a loop

Now, we want to describe such initial Hermite data for which the corresponding TC-interpolant contains a loop. We use a well-known expressions of the control points of a cubic PH Bézier curve

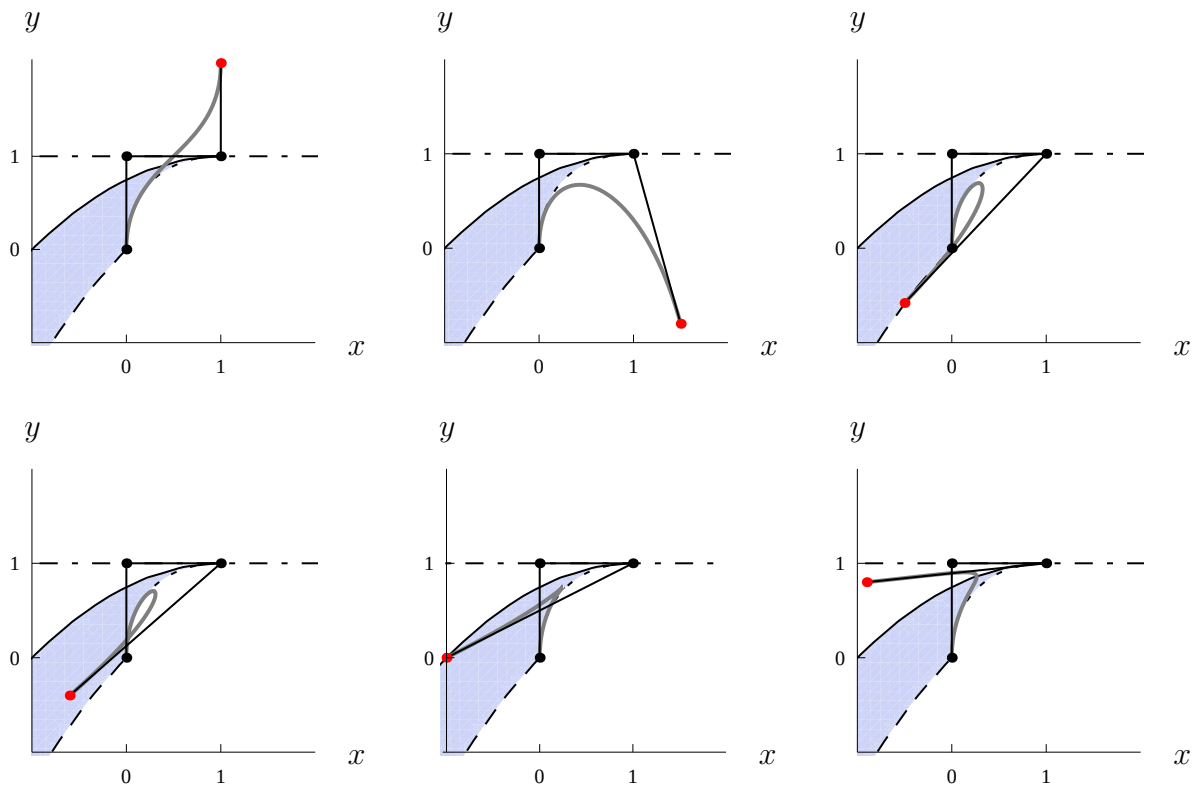


Figure 4.3: Three points $\mathbf{Q}_0 = (0,0)^\top, \mathbf{Q}_1 = (0,1)^\top, \mathbf{Q}_2 = (1,1)^\top$ and moving point $\mathbf{Q}_3$. A lightblue color shows a domain where a loop occurs. Bounding curves from the equation (4.8) are drawn in solid, dashed, dotted and dotdashed curves.

(see [37])

$$\begin{aligned}
\mathbf{P}_1 &= \mathbf{P}_0 + \tfrac{1}{3}\left(u_0^2 - v_0^2, 2u_0v_0\right)^\top, \\
\mathbf{P}_2 &= \mathbf{P}_1 + \tfrac{1}{3}\left(u_0u_1 - v_0v_1, u_0v_1 + u_1v_0\right)^\top, \\
\mathbf{P}_3 &= \mathbf{P}_2 + \tfrac{1}{3}\left(u_1^2 - v_1^2, 2u_1v_1\right)^\top,
\end{aligned} \tag{4.9}$$

where $\mathbf{P}_0$ corresponds to translation. Without loss of generality we can identify points $\mathbf{P}_0$, $\mathbf{Q}_0$ and $\mathbf{P}_1$,$\mathbf{Q}_1$, i.e., $\mathbf{P}_0 = (0,0)^\top, \mathbf{P}_1 = (0,1)^\top$. Since $\mathbf{P}_1$ depends only on two parameters u_0, v_0 (cf. (4.9)) we can solve

$$\mathbf{P}_1 = \mathbf{P}_0 + \frac{1}{3}\left(u_0^2 - v_0^2, 2u_0v_0\right)^\top = (0,1)^\top$$

with respect to u_0, v_0. We obtain two solutions

$$(u_0^1, v_0^1) = (\sqrt{3/2}, \sqrt{3/2}) \quad \text{and} \quad (u_0^2, v_0^2) = (-\sqrt{3/2}, -\sqrt{3/2}).$$

Since both these solutions provide two domains symmetrical with respect to the origin we will consider only the first solution in the remainder of the section. Substituting u_0^1, v_0^1 into $\mathbf{P}_2$, $\mathbf{P}_3$ (cf. (4.9)) we obtain the remaining control points as

$$\begin{aligned}
\mathbf{P}_2 &= \left(\tfrac{1}{\sqrt{6}}(u_1 - v_1), 1 + \tfrac{1}{\sqrt{6}}(u_1 + v_1)\right)^\top \\
\mathbf{P}_3 &= \left(\tfrac{1}{6}(u_1 - v_1)(\sqrt{6} + 2u_1 + 2v_1), \tfrac{1}{6}(6 + \sqrt{6}v_1 + u_1(\sqrt{6} + 4v_1))\right)^\top.
\end{aligned}$$

Transformation from the xy-plane to the u_1v_1-plane

In general, affine transformations do not preserve the PH condition. Nevertheless, we need only to find out, if the TC-interpolant contains a loop or not and this property is preserved under affine transformations. Using the affine transformation which map the points $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2$ to $\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2$ we can map also $\mathbf{P}_3$ to a new point $\mathbf{Q'}_3$, i.e.,

$$\mathbf{Q'}_3 = \begin{bmatrix} \frac{\sqrt{6}}{u_1-v_1} & 0 \\ -\frac{u_1+v_1}{u_1-v_1} & 1 \end{bmatrix} \cdot \mathbf{P}_3 = \left(\frac{1}{\sqrt{6}}(\sqrt{6}+2u_1+2v_1), \frac{1}{3}(3-u_1^2-v_1^2) \right)^\top . \tag{4.10}$$

The inequalities (4.8) describe the domain where $\mathbf{Q}_3$ has to lie in order to get a cubic Bézier curve with a loop. We can use the x- and y-coordinates of $\mathbf{Q'}_3$ given by (4.10) to transform the xy-plane into the u_1v_1-plane, i.e., we can substitute

$$\begin{aligned} x &= \tfrac{1}{\sqrt{6}}(\sqrt{6}+2u_1+2v_1), \\ y &= \tfrac{1}{3}(3-u_1^2-v_1^2) \end{aligned}$$

into (4.8). After some simplifications we obtain a semialgebraic set Σ, defined as a union of two sets Σ_1 and Σ_2 where

$$\begin{aligned} \Sigma_1 = \{&(u_1,v_1) \in \mathbb{R}^2 : \\ &(u_1-v_1)^2 > 0 \wedge u_1(\sqrt{6}+u_1) + v_1(\sqrt{6}+v_1) < 3 + 4u_1v_1 \wedge \\ &3 + \sqrt{6}u_1 + \sqrt{6}v_1 \le 0 \big\}, \end{aligned} \tag{4.11}$$

$$\begin{aligned} \Sigma_2 = \{&(u_1,v_1) \in \mathbb{R}^2 : \\ &9 + (\sqrt{6}-2u_1)u_1 + (\sqrt{6}-2v_1)v_1 > 3\sqrt{9+2\sqrt{6}v_1 - 2(-\sqrt{6}u_1 + u_1^2 + 2u_1v_1 + v_1^2)} \wedge \\ &\wedge (u_1-v_1)^2 > 0 \wedge -3 < \sqrt{6}(u_1+v_1) < 0 \big\}. \end{aligned} \tag{4.12}$$

If we choose parameters $(u_1, v_1) \in \Sigma$ (see Fig. 4.4 (left)) and substitute these parameters to $\mathbf{P}_3$ (cf. (4.9)), we get a control polygon providing TC-interpolant with a loop.

Moreover, since both pairs of parameters (u_1, v_1) and (v_1, u_1) provide the same point $\mathbf{P}_3$, the domain Σ is symmetrical with respect to the line $u_1 = v_1$ and we can use only one half of this domain in the remaining text.

Transformation from the u_1v_1-plane to the $\theta_0\theta_3$-plane

In this section we focus on the last transformation from the u_1v_1-plane to the $\theta_0\theta_3$-plane, i.e., we want to be able to determine whether the TC-interpolant contains a loop directly from angles θ_0, θ_3.

As it was mentioned at the beginning of Section 4.1, the angles θ_0, θ_3 can be computed using (4.2) and (4.4). The computation of θ_3 depends on signs of w_0 and w_3 given by (4.3). Thus, we can divide the u_1v_1-plane into two domains with respect to the signs of w_0, w_3 and also the computation of the corresponding angles θ_0, θ_3 – the domains are (see Fig. 4.4 (middle)):

$$\begin{aligned} \Omega_1 &= \{(u_1,v_1) \in \mathbb{R}^2 : w_0(u_1,v_1)w_3(u_1,v_1) > 0\}, \\ \Omega_2 &= \{(u_1,v_1) \in \mathbb{R}^2 : w_0(u_1,v_1)w_3(u_1,v_1) < 0\}. \end{aligned}$$

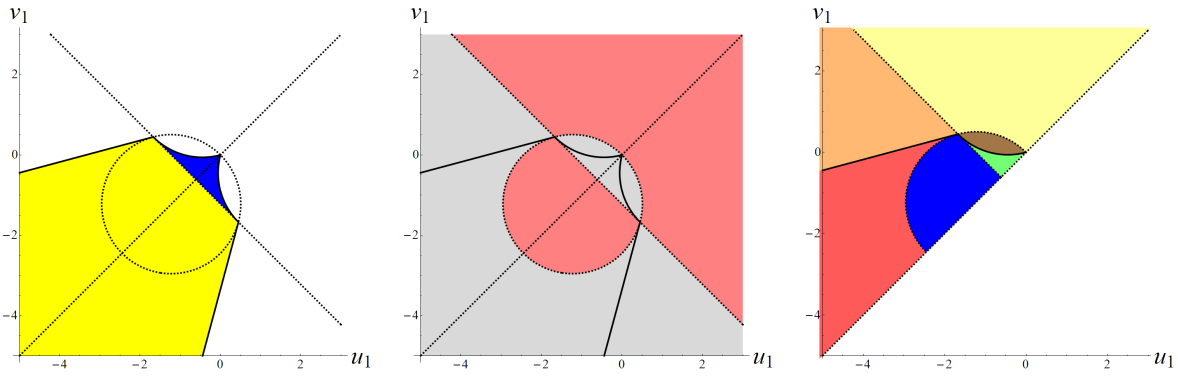


Figure 4.4: Left: Σ_1 (yellow), Σ_2 (blue); Middle: Ω_1 (pink), Ω_2 (light gray); Right: Λ_1 (blue), Λ_2 (green), Λ_3 (red), Λ_4 (light yellow), Λ_5 (orange), Λ_6 (brown). All domains are in the u_1v_1-plane.

The domains Ω_1, Ω_2 are bounded by the three implicit curves

$$\begin{aligned} d_1(u_1, v_1) &= u_1 - v_1 = 0, \\ d_2(u_1, v_1) &= 2u_1 + 2v_1 + \sqrt{6} = 0, \\ d_3(u_1, v_1) &= u_1\left(\sqrt{6}u_1 + 6\right) + v_1\left(\sqrt{6}v_1 + 6\right) = 0. \end{aligned}$$

Intersecting the domains Ω_1 and Ω_2 with the domain Σ (only one half of it, according to the symmetry, cf. (4.11) and (4.12)) we get three new domains

$$\begin{aligned} \Lambda_1 &= \Omega_1 \cap \Sigma_1 \cap \{(u_1, v_1) \in \mathbb{R}^2 : u_1 < v_1\}, \\ \Lambda_2 &= \Omega_2 \cap \Sigma_2 \cap \{(u_1, v_1) \in \mathbb{R}^2 : u_1 < v_1\}, \\ \Lambda_3 &= \Omega_2 \cap \Sigma_1 \cap \{(u_1, v_1) \in \mathbb{R}^2 : u_1 < v_1\}, \end{aligned} \tag{4.13}$$

which differ by a computation of θ_0 and θ_3 and which represent subdomains of the u_1v_1-plane providing TC-interpolant(s) with a loop. The domains Λ_i, $i = 1, 2, 3$, are shown in Fig. 4.4 (right).

Further, we can define mappings

$$\begin{aligned} \Phi_1 &: \Omega_1 \to \mathbb{R}(\theta_0, \theta_3) : (u_1, v_1) \mapsto (\theta_0(u_1, v_1), \theta_3(u_1, v_1)), \\ \Phi_2 &: \Omega_2 \to \mathbb{R}(\theta_0, \theta_3) : (u_1, v_1) \mapsto (\theta_0(u_1, v_1), \theta_3(u_1, v_1)), \end{aligned}$$

which differ by a computation of $\theta_3(u_1, v_1)$ (cf. (4.4)). Using Φ_1 and Φ_2 we can transform the domains Λ_i, $i = 1, 2, 3$ (cf. (4.13)) to the $\theta_0\theta_3$-plane in order to obtain a subdomain of Γ^{PH} providing TC-interpolants with a loop. The general approach for the transformation of Λ_i to the $\theta_0\theta_3$-plane consists of two steps:

1. finding the parametric description of the domain Λ_i using rational Bézier surfaces (see e.g. [95] for more details), we obtain a parameterization of the form $(\hat{u}_1(s, t), \hat{v}_1(s, t))^\top$,
2. transformation of Λ_i using Φ_1 (or Φ_2).

To demonstrate these steps in more detail, we transform Λ_2 using Φ_2 into the $\theta_0\theta_3$-plane. The corresponding control net (the third coordinate represents the weight of the control point) is described by the control points

$$\begin{aligned} &\mathbf{R}_1^2 = \left(-\tfrac{\sqrt{\frac{3}{2}}}{2}, -\tfrac{\sqrt{\frac{3}{2}}}{2}, 1\right)^\top, \ \mathbf{R}_2^2 = (0, 0, 1)^\top, \\ &\mathbf{R}_3^2 = \left(-\tfrac{3+\sqrt{3}}{2\sqrt{2}}, \tfrac{1}{2}\sqrt{6 - 3\sqrt{3}}, 1\right)^\top, \mathbf{R}_4^2 = \left(-\tfrac{1}{2}\sqrt{2 + \sqrt{3}}, \tfrac{\sqrt{3}-3}{2\sqrt{6}}, \tfrac{\sqrt{3}}{2}\right)^\top. \end{aligned}$$

Computing a rational Bézier surface for this control net we arrive at the parametric description of the domain Λ_2 in the form

$$(\hat{u}_1^2(s,t),\hat{v}_1^2(s,t)) = \left(\frac{s\left(3\sqrt{2}-2\sqrt{3\left(2+\sqrt{3}\right)}t\right)+\sqrt{2}\left(3+\sqrt{3}\right)(t-1)}{4\left(\sqrt{3}-2\right)(s-1)(t-1)t+4}, \frac{s\left(\left(\sqrt{3}-3\right)t(2t-3)-3\right)-\left(\sqrt{3}-3\right)(t-1)(2t-1)}{2\sqrt{2}\left(\left(\sqrt{3}-2\right)(s-1)(t-1)t+1\right)}\right)^\top,$$

where $(s,t) \in (0,1)\times(0,1)$. Then

$$\Psi_2 = \Phi_2(\hat{u}_1^2(s,t),\hat{v}_1^2(s,t))$$

represents the subdomain of Γ^{PH} corresponding to Λ_2 (see Fig. 4.5 (left)).

Similarly, we can find parametric descriptions $(\hat{u}_1^1(s,t),\hat{v}_1^1(s,t))^\top$ and $(\hat{u}_1^3(s,t),\hat{v}_1^3(s,t))^\top$ of the domains Λ_1 and Λ_3, respectively. Then

$$\Psi_1 = \Phi_1(\hat{u}_1^1(s,t),\hat{v}_1^1(s,t)) \quad \text{and} \quad \Psi_3 = \Phi_2(\hat{u}_1^3(s,t),\hat{v}_1^3(s,t))$$

represent subdomains of Γ^{PH} corresponding to Λ_1 and Λ_3, respectively (see Fig. 4.5 (left)).

Proposition 8. *For given Hermite interpolation data $\mathbf{P}_0$, $\mathbf{P}_3$, $\mathbf{t}_0$, $\mathbf{t}_3$, where $\mathbf{P}_0$ and $\mathbf{P}_3$ are distinct points on the real axis and $\theta_0 = \angle(\mathbf{t}_0, \mathbf{P}_3-\mathbf{P}_0) \in (-\pi,0)$, $\theta_3 = \angle(\mathbf{P}_3-\mathbf{P}_0,\mathbf{t}_3)\in(0,2\pi)$ such that*

$$(\theta_0,\theta_3) \in \Psi_1 \cup \Psi_2 \cup \Psi_3,$$

there exists at least one TC-interpolant with a loop that matches given initial Hermite data.

Remark 9. *It can be seen from Fig. 4.5 (left) that some parts of Ψ_2 and Ψ_3 are "folded". According to Section 4.1, there exist two TC-interpolants for such initial Hermite data. Moreover, initial data taken from Ψ_2 and Ψ_3 always produce TC-interpolants with a loop. This implies that there exist exactly two TC-interpolants with a loop for such initial data.*

4.2.2 TC-interpolants without a loop

To analyze the number of TC-interpolants, it is necessary to map also the remaining subdomains of intersection of Ω_1 and Ω_2 with Σ where a loop does not occur. Similarly to Section 4.2.1, we obtain three domains (see Fig. 4.4 (right))

$$\begin{aligned}
\Lambda_4 &= \Omega_1 \cap \Sigma_1' \cap \{(u_1,v_1)\in\mathbb{R}^2 : u_1 < v_1\},\\
\Lambda_5 &= \Omega_2 \cap \Sigma_1' \cap \Sigma_2' \cap \{(u_1,v_1)\in\mathbb{R}^2 : u_1 < v_1 \wedge d_2(u_1,v_1) < 0\},\\
\Lambda_6 &= \Omega_2 \cap \Sigma_1' \cap \Sigma_2' \cap \{(u_1,v_1)\in\mathbb{R}^2 : u_1 < v_1 \wedge d_2(u_1,v_1) > 0\},
\end{aligned}$$

where $\Sigma_i' = \mathbb{R}^2 \setminus \Sigma_i$, $i=1,2$. Further, we find parametric descriptions $(\hat{u}_1^j(s,t),\hat{v}_1^j(s,t))^\top$ of the domains Λ_j, $j=4,5,6$. Then

$$\Psi_4 = \Phi_1(\hat{u}_1^4(s,t),\hat{v}_1^4(s,t)) \quad \text{and} \quad \Psi_j = \Phi_2(\hat{u}_1^j(s,t),\hat{v}_1^j(s,t)),\ j=5,6,$$

represent subdomains of Γ^{PH} corresponding to Λ_j, $j=4,5,6$ (see Fig. 4.5 (middle)).

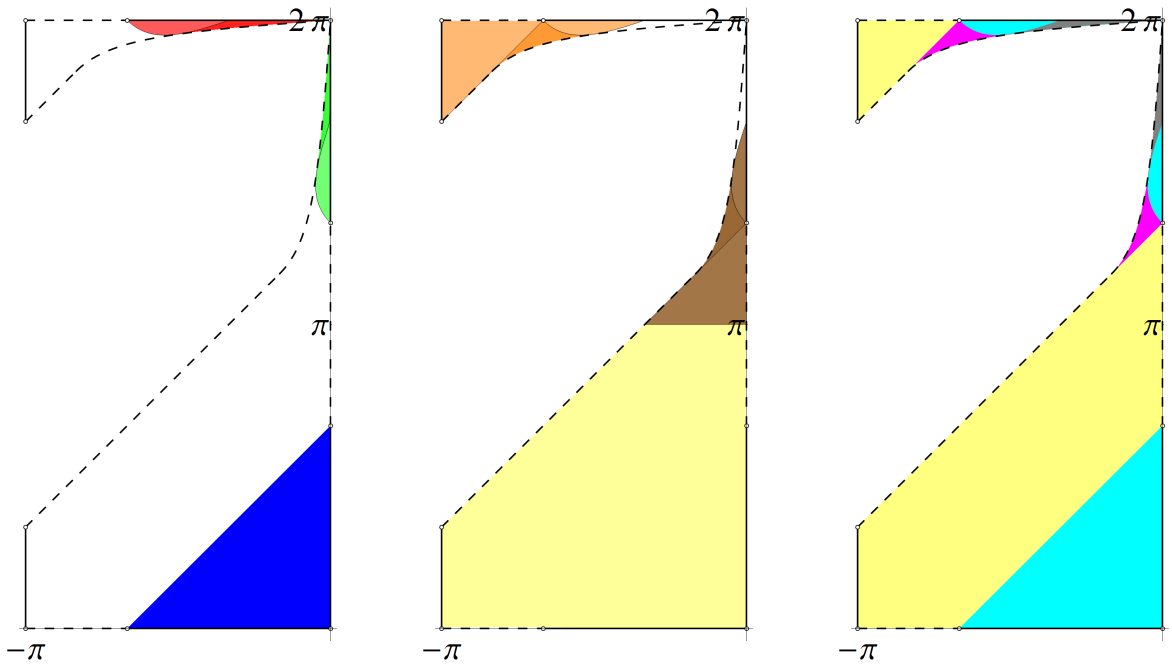


Figure 4.5: Left: Ψ_1 (blue), Ψ_2 (green), Ψ_3 (red); Middle: Ψ_4 (light yellow), Ψ_5 (orange), Ψ_6 (brown); Right: Subdomains of Γ^{PH} where exactly one without a loop (yellow), one with and one without a loop (cyan), two without a loop (magenta) and two with a loop (gray) TC-interpolant(s) exist. All domains are in the $\theta_0\theta_3$-plane.

Proposition 10. *For given Hermite interpolation data $\mathbf{P}_0$, $\mathbf{P}_3$, $\mathbf{t}_0$, $\mathbf{t}_3$, where $\mathbf{P}_0$ and $\mathbf{P}_3$ are distinct points on the real axis and $\theta_0 = \angle(\mathbf{t}_0, \mathbf{P}_3 - \mathbf{P}_0) \in (-\pi, 0)$, $\theta_3 = \angle(\mathbf{P}_3 - \mathbf{P}_0, \mathbf{t}_3) \in (0, 2\pi)$ such that*

$$(\theta_0, \theta_3) \in \Psi_4 \cup \Psi_5 \cup \Psi_6,$$

there exists at least one TC-interpolant without a loop that matches given initial Hermite data.

Remark 11. *It can be seen from Fig. 4.5 (middle) that some parts of Ψ_5 and Ψ_6 are again "folded". According to Section 4.1, there exist two TC-interpolants for such initial Hermite data. Moreover, initial data taken from Ψ_5 and Ψ_6 always produce TC-interpolants without a loop. This implies that there exist exactly two TC-interpolants without a loop for such initial data.*

Remark 12. *Let us summarize some consequences of Propositions 8 and 10:*

1. *If $(\theta_0, \theta_3) \in (\Psi_1 \cup \Psi_2 \cup \Psi_3) \cap (\Psi_4 \cup \Psi_5 \cup \Psi_6)$, then there exist one TC-interpolant without a loop and one TC-interpolant with a loop.*

2. *If the angles θ_0, θ_3 fulfil*

$$\frac{2}{3}\pi \leq -\theta_0 + \theta_3 < \frac{4}{3}\pi \quad \text{or} \quad -\theta_0 + \theta_3 > \frac{8}{3}\pi,$$

then there exists exactly one TC-interpolant without a loop.

Remark 13. *It remains to study the number and the quality of TC-interpolants on the boundary of the domain Γ, i.e., for $\theta_0 \in \{-\pi, 0\}$ or $\theta_3 = 0$. According to the discussion of these special cases at the beginning of Section 4.1 and with respect to Propositions 8 and 10, we can write that*

- *for $\theta_3 = 0$, there exists only one TC-interpolant for $\theta_0 \in (-2\pi/3, 0)$ with a loop,*
- *for $\theta_0 = 0$, there exists only one TC-interpolant for $\theta_3 \in (0, 2\pi/3) \cup (4\pi/3, 2\pi)$ with a loop,*
- *for $\theta_0 = -\pi$, there exists only one TC-interpolant for $\theta_3 \in (0, \pi/3) \cup (5\pi/3, 2\pi)$ without a loop.*

4.3 Examples

In this section we show several examples which demonstrate the existence of TC-interpolant(s) for given Hermite data.

Example 1. Let us find TC-interpolant(s) for the following Hermite data

$$\mathbf{P}_0 = (0,1)^\top,\ \mathbf{P}_3 = (6,1)^\top, \quad \mathbf{t}_0 = \left(\tfrac{1}{\sqrt{2}}, -\tfrac{1}{\sqrt{2}}\right)^\top,\ \mathbf{t}_3 = \left(\tfrac{1}{\sqrt{2}}, \tfrac{1}{\sqrt{2}}\right)^\top.$$

We can easily check that w_0 and w_3 have the same signs. Using (4.2) and (4.4) we compute $\theta_0 = -\pi/4$ and $\theta_3 = \pi/4$. Since $-\theta_0 + \theta_3 < 2\pi/3$, we expect two qualitatively different solutions according to Propositions 8 and 10. Further, using (4.6) we can compute d_{01} and also the corresponding d_{23} from $D = \sqrt{d_{23}/d_{01}}$ in the form

$$d_{01}^+ = \tfrac{6}{1+\sqrt{2}},\ d_{01}^- = \tfrac{6}{-1+\sqrt{2}}, \qquad d_{23}^+ = d_{01}^+ D_+ = \tfrac{6}{1+\sqrt{2}},\ d_{23}^- = d_{01}^- D_- = \tfrac{6}{-1+\sqrt{2}}.$$

Finally, we determine the control points of Bézier curves providing TC-interpolants to obtain their parameterizations

$$\mathbf{r}_\pm(t) = \frac{1}{\pm 1 + \sqrt{2}} \left(3t(3\sqrt{2} - 3(\mp 2 + \sqrt{2})t + 2(\mp 2 + \sqrt{2})t^2), \pm 1 + \sqrt{2} - 9\sqrt{2}t + 9\sqrt{2}t^2\right)^\top,$$

$t \in [0,1]$, with PH condition $x'(t)^2 + y'(t)^2 = 162(17 \mp 12\sqrt{2})(\pm 2 + \sqrt{2} \pm 2(-1+t)t)^2$. The TC-interpolants are shown in Fig. 4.6 (left). ♦

Example 2. Let us consider the Hermite data

$$\mathbf{P}_0 = (0,1)^\top,\ \mathbf{P}_3 = (6,1)^\top, \quad \mathbf{t}_0 = \left(\tfrac{\sqrt{2}+\sqrt{6}}{4}, \tfrac{\sqrt{2}-\sqrt{6}}{4}\right)^\top,\ \mathbf{t}_3 = \left(-\sqrt{\tfrac{5}{8} - \tfrac{\sqrt{5}}{8}}, \tfrac{-1-\sqrt{5}}{4}\right)^\top.$$

In this case, $\theta_0 = -\pi/12$ and $\theta_3 = 13\pi/10$. According to Proposition 10, we expect two TC-interpolants without a loop, which we can find analogously to Example 1. The TC-interpolants are shown in Fig. 4.6 (right). ♦

Example 3. Let us consider Hermite data

$$\mathbf{P}_0 = (0,1)^\top,\ \mathbf{P}_3 = (6,1)^\top, \quad \mathbf{t}_0 = \left(-\tfrac{\sqrt{3}}{2}, -\tfrac{1}{2}\right)^\top,\ \mathbf{t}_3 = \left(\tfrac{1+\sqrt{3}}{2\sqrt{2}}, -\tfrac{\sqrt{3}-1}{2\sqrt{2}}\right)^\top.$$

For these Hermite data we get $\theta_0 = -5\pi/6$ and $\theta_3 = 23\pi/12$. Since $-\theta_0 + \theta_3 > 8\pi/3$, we expect exactly one TC-interpolant without a loop according to Remark 12, which we can find analogously to Example 1. This TC-interpolant is shown in Fig. 4.7. ♦

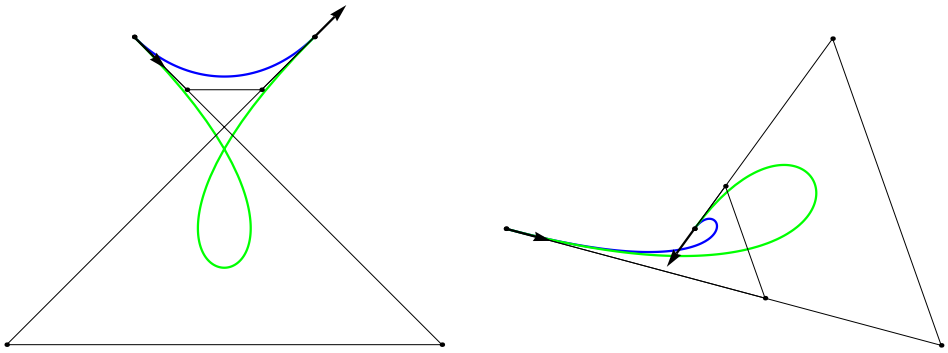


Figure 4.6: TC-interpolants with their control polygons and unit tangent vectors. Left: Example 1; Right: Example 2.

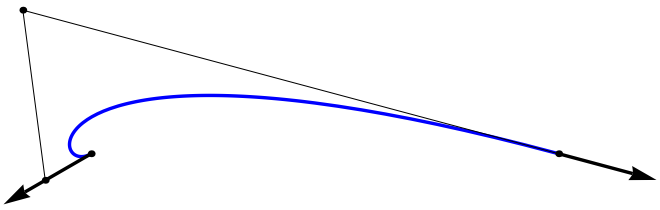


Figure 4.7: Example 3: TC-interpolant with is control polygon and unit tangent vectors.

5

Hermite interpolation by two PH cubics

In this chapter we cover all possible initial Hermite data, including data which is impossible to interpolate by only one TC-interpolant. Moreover, we show that there exist at most two TC-interpolants which match such data. In the first section, we study G^1 Hermite interpolation where the proof is done and in second section we focus on C^1 Hermite interpolation by two TC-interpolants. At the end of this chapter we set several statements how to get good TC-interpolants approximating a given curve.

5.1 G^1 Hermite interpolation

First, we formulate the theorem describing the number of TC-interpolants matching any G^1 Hermite data.

Theorem 14. *Any G^1 Hermite data $\mathbf{P}_0, \mathbf{P}_3, \mathbf{t}_0, \mathbf{t}_3$ can be interpolated by at most **two** TC-interpolants joined with G^1 continuity.*

Proof. (of Theorem 14) Let us suppose connecting point $\overline{\mathbf{P}}$ lying on the line segment $\mathbf{P}_0\mathbf{P}_3$ with associated tangent vector $\overline{\mathbf{t}}$. We denote $\overline{\theta} = \angle(\overline{\mathbf{t}}, \mathbf{P}_0 - \mathbf{P}_3)$. We have to emphasize that we do not care about the orientation! The main idea is to show that $\overline{\theta}$ lies in the interval $\mathcal{I}$, which is always nonempty.

Let us consider any point $(\theta_0, \theta_3) \in \Gamma^{\mathrm{PH}}$. Since we know $\theta_0 \in (-\pi, 0)$ and $\theta_3 \in (0, 2\pi)$ then the point (θ_0, θ_3) determines two line segments in the domain $(-\pi, 0) \times (0, 2\pi)$ parallel with the axis. Such two line segments intersect the domain of definition of TC-interpolant in horizontal line segment $\overline{\mathcal{I}}_0$ and the second one is vertical line segment $\overline{\mathcal{I}}_3$, which can be described as follows

$$\overline{\mathcal{I}}_0 = (0, a_1) \cup \underbrace{(a_2, a_3)}_{\text{for } \theta_3 > \frac{5\pi}{3}}, \quad \overline{\mathcal{I}}_3 = (0, b_1) \cup (b_2, b_3),$$

where a_i and b_i are values lying on boundary of the domain of definition of TC-interpolant and according to its symmetry we arrive at nonempty intersection $\overline{\mathcal{I}} = \overline{\mathcal{I}}_0 \cup \overline{\mathcal{I}}_3$. Since the interval $\overline{\mathcal{I}}$ is nonempty, it means that the input data can be interpolated by infinitely many pairs of TC-interpolants. □

Now, let us discuss all the cases in detail. The discussion can be considered as an outline how to construct a pair of TC-interpolants. Let us emphasize that we do not preserve the sign of $\overline{\theta}_0, \overline{\theta}_3$ in all the outline, but we keep the orientation, which gives the direction of the tangent vector and consequently the sign of the correspondent angles. If we have kept the sign we did not get an nonempty intervals.

We split up the outline into two parts. We consider that on the line segment $\mathbf{P}_0\mathbf{P}_3$ lies such point $\overline{\mathbf{P}}$ with unit tangent vector $\overline{\mathbf{t}}$ which connects arbitrary G^1 Hermite data. The choice of the point $\overline{\mathbf{P}}$ can be done in different way but we do this for simplification of all contruction, i.e., $\overline{\theta} = \angle(\overline{\mathbf{t}}, \mathbf{P}_3 - \mathbf{P}_0)$.

1. $(\theta_0, \theta_3) \notin \Gamma^{\mathrm{PH}} \quad \wedge \quad \theta_3 \leq \pi$
   This condition can be rewritten as $-\theta_0 + \theta_3 \geq 4\pi/3 \wedge \theta_3 \leq \pi$.

   (1a) We have to find first TC-interpolant, which matches $\mathbf{P}_0$ and $\overline{\mathbf{P}}$ and which is determined by the angles $\theta_0, \overline{\theta}_3$. We get all angles $\overline{\theta}_3$ such as

   $$\overline{\theta}_3 \in \overline{\mathcal{I}}_3 = (0, 4\pi/3 + \theta_0),$$

   which describe the family of TC-interpolants.

   According to the orientation of the system (w_0, w_3 from the equation (4.3)) and according to the assumption of $\overline{\theta}_3$ we know that for the second TC-interpolant the angle is $\theta_3^s = 2\pi - \theta_3 > \pi$. Now we have to find all $\overline{\theta}_0$ which create second TC-interpolant. All angles $\overline{\theta}_0$ are

   $$\overline{\theta}_0 \in \overline{\mathcal{I}}_0 = \begin{cases} (-f_4^{-1}(\theta_3^s), 0), & \theta_3^s \in (7\pi/6, 5\pi/3) \\ (-4\pi/3 + \theta_3^s, 0), & \theta_3^s \in (\pi, 7\pi/6\rangle, \end{cases}$$

   where f_4^{-1} is inverse function to the function f_4 from equation (4.7). The interval $\overline{\mathcal{I}}$ of the angle $\overline{\theta}$ is described as

   $$\overline{\theta} \in -\overline{\mathcal{I}}_0 \cap \overline{\mathcal{I}}_3 = -\overline{\mathcal{I}}_0 \neq \emptyset,$$

   see Fig. 5.1 (left) and appropriate TC-interpolants are visualized in Fig. 5.3. Let us notice that the smallest interval is $\overline{\mathcal{I}} = (0, f_4^{-1}(5\pi/3))$. It means if we choose arbitrary value from this interval we can always create two TC-interpolants for such data.

   (1b) Let us discuss the remaining area for the angle $\overline{\theta}_3$ which is expressed as

   $$\overline{\theta}_3 \in \overline{\mathcal{I}}_3 = \begin{cases} (8\pi/3 - \theta_0, 2\pi), & \theta_0 \in (-\pi, -5\pi/6) \\ (f_3^{-1}(\theta_0), 2\pi), & \theta_0 \in \langle -5\pi/6, -2\pi/3). \end{cases}$$

   Knowing the value of the angle $\overline{\theta}_3 > \pi$, we do not compute θ_3^s for second TC-interpolant and we are able to compute $\overline{\theta}_0$ directly. We arrive at

   $$\overline{\theta}_0 \in \overline{\mathcal{I}}_0 = (-4\pi/3 + \theta_3, 0)$$

   and consequently to the angle $\overline{\theta}$

   $$\overline{\theta} = -\overline{\mathcal{I}}_0 \cap \overline{\mathcal{I}}_3^s = \overline{\mathcal{I}}_3^s,$$

where $\overline{\mathcal{I}}_3^s$ is

$$\overline{\mathcal{I}}_3^s = \begin{cases} (0, 2\pi/3 - \theta_0), & \theta_0 \in (-\pi, -5\pi/6) \\ (0, 2\pi - f_3^{-1}(\theta_0)), & \theta_0 \in \langle -5\pi/6, -2\pi/3). \end{cases}$$

The described construction is shown in Fig. 5.1 (right) and one pair of possible TC-interpolants are shown in Fig. 5.3.
Similarly, we know the smallest interval $\overline{\mathcal{I}}_3^s = (0, 2\pi - f_3^{-1}(-\pi/3))$, from which we can always choose the value $\overline{\theta}$.

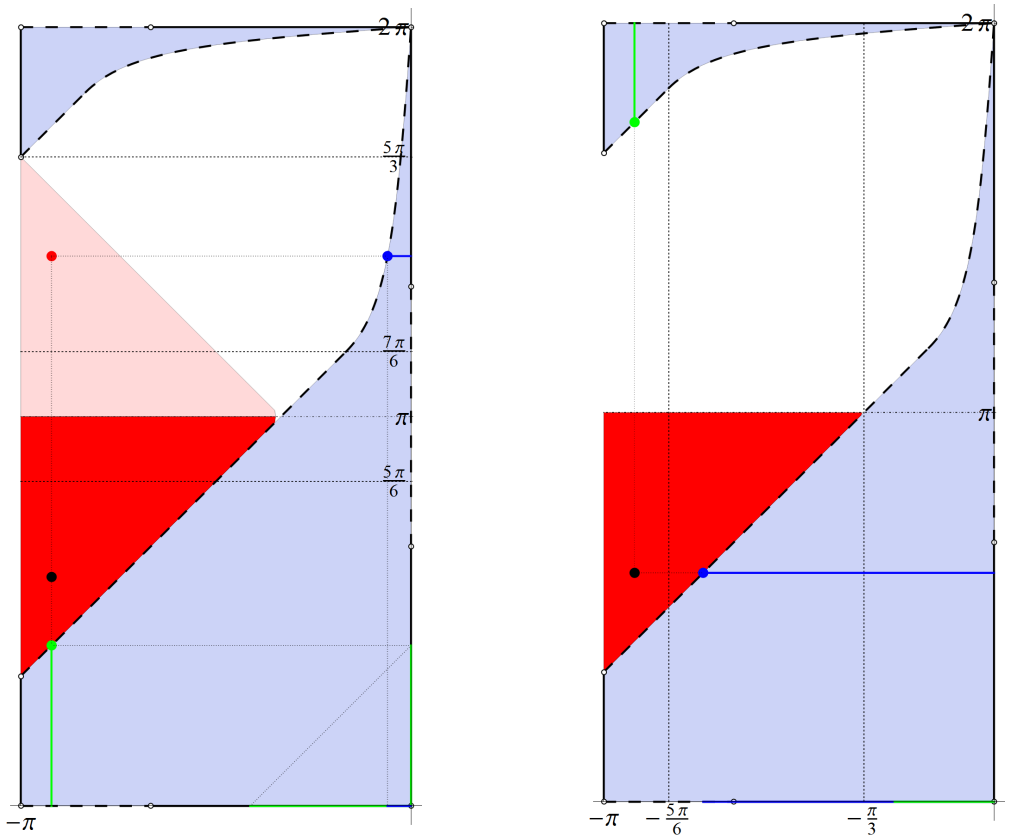


Figure 5.1: A point (θ_0, θ_3) (black) and point (θ_0, θ_3^s) (red). Green line segment is an interval $\overline{\mathcal{I}}_3$ and blue line segment describes an interval $\overline{\mathcal{I}}_0$. Left: Situation in item (1a); Right: Situation in item (1b).

2. $(\theta_0, \theta_3) \notin \Gamma^{\mathrm{PH}} \quad \wedge \quad \theta_3 > \pi$
   This case is very similar to the first case of the construction and we also subdivide the ideas into two subitems:

   (2a) The angle range $\overline{\mathcal{I}}_3$ can be written as

$$\overline{\theta}_3 \in \overline{\mathcal{I}}_3 = \begin{cases} (0, 4\pi/3 + \theta_0), & \theta_0 \in (-\pi, -\pi/6\rangle \\ (0, f_4^{-1}(\theta_0)), & \theta_0 \in (-\pi/6, 0). \end{cases}$$

Since the angle $\theta_3 > \pi$ and according to the orientation of the system we have to compute the angle $\theta_3^s = 2\pi - \theta_3 < \pi$. Using this value we are able to compute $\overline{\theta}_0$ as

$$\overline{\theta}_0 \in \overline{\mathcal{I}}_0 = \begin{cases} (-4\pi/3 + \theta_3^s, 0), & \theta_3^s \in (\pi/3, \pi) \\ (-\pi, 0), & \theta_3^s \in (0, \pi/3). \end{cases}$$

Finally, the interval for angle $\overline{\theta}$, i.e. $\overline{\mathcal{I}} = -\overline{\mathcal{I}}_0 \cap \overline{\mathcal{I}}_3$ is not empty. The situation is shown in Fig. 5.2 (left) and one pair of TC-interpolants is visualized in Fig. 5.4.

(2b) Now, let us use the remaining part of the domain of definition of TC-interpolant. The angle $\overline{\theta}_3$ is in the interval

$$\overline{\theta}_3 \in \overline{\mathcal{I}}_3 = \begin{cases} (8\pi/3 - \theta_0, 2\pi), & \theta_0 \in (-\pi, -5\pi/6\rangle \\ (f_3(\theta_0), 2\pi), & \theta_0 \in (-5\pi/6, 0). \end{cases}$$

According to the orientation of the system, we can compute an interval of the angle $\overline{\theta}_0$ as

$$\overline{\theta}_0 \in \overline{\mathcal{I}}_0 \begin{cases} \langle -\pi, -f_3^{-1}(\theta_3)) \cup (-f_4^{-1}(\theta_3), 0), & \theta_3 \in (11\pi/6, 2\pi), \\ \langle -\pi, -8\pi/3 + \theta_3) \cup (-f_4^{-1}(\theta_3), 0), & \theta_3 \in (5\pi/3, 11\pi/6), \\ (-f_4^{-1}(\theta_3), 0), & \theta_3 \in (7\pi/6, 5\pi/3), \\ (-4\pi/3 - \theta_3, 0), & \theta_3 \in (\pi, 7\pi/6). \end{cases}$$

Analogously, we arrive at the final interval range of the angle $\overline{\theta}$, which is written as an intersection of $\overline{\mathcal{I}}_3^s$ and $-\overline{\mathcal{I}}_0$ and is always nonempty. The construction is shown in Fig. 5.2 (right) and TC-interpolants in Fig. 5.4.

In every item we have showed that an interval $\overline{\mathcal{I}}$ is not empty and therefore there exist at most two TC-interpolants matching given G^1 Hermite data, which do not belong to the domain of definition of TC-interpolant.

Example 4. Let us consider G^1 Hermite data not lying in Γ^{PH}, e.g.,

$$\mathbf{P}_0 = (0,0)^\top, \mathbf{P}_3 = (6,1)^\top, \quad \theta_0 = -\frac{11}{12}\pi, \theta_3 = \frac{3}{5}\pi.$$

On the line segment $\mathbf{P}_0\mathbf{P}_3$ we suppose the point $\overline{\mathbf{P}}$ with associated angles $\overline{\theta}_3, \overline{\theta}_0$. We seek such two TC-interpolants, where the first one is determined by the points $\mathbf{P}_0, \overline{\mathbf{P}}$ and by the angles $\theta_0, \overline{\theta}_3$ and the second one is given by the points $\overline{\mathbf{P}}, \mathbf{P}_3$ and by the angles $\overline{\theta}_0, \theta_3$.

Since $\theta_3 < \pi$ we obtain the interval

$$\overline{\mathcal{I}}_3 = (0\,, \underbrace{\frac{4}{3}\pi - \frac{11}{12}\pi}_{\frac{5}{12}\pi}\,),$$

from which we can pick up the value of the angle $\overline{\theta}_3$ and therefore it determines all possible TC-interpolants matching the points $\mathbf{P}_0, \overline{\mathbf{P}}$.

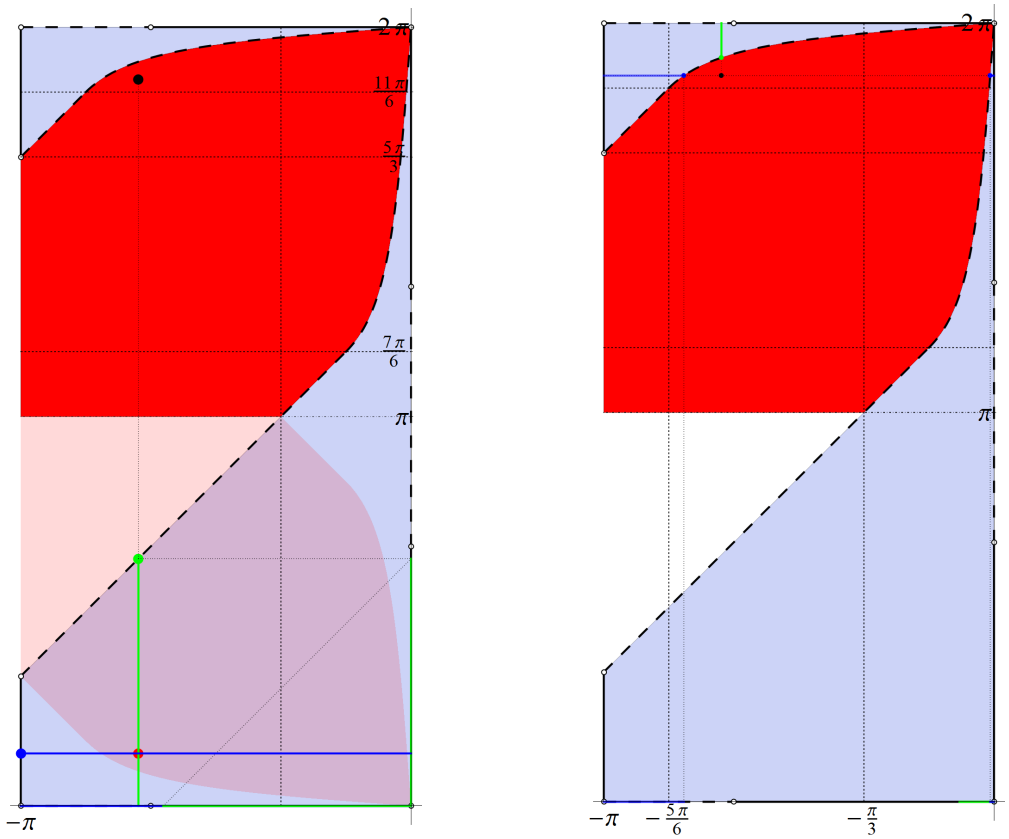


Figure 5.2: A point (θ_0, θ_3) (black) and a point (θ_0, θ_3^s) (red). Green line segment is an interval $\overline{\mathcal{I}}_3$ and blue line segment describes interval $\overline{\mathcal{I}}_0$. Left: Situation in item (2a); Right: Situation in item (2b).

From the first TC-interpolant we get all tangent vectors $\bar{\mathbf{t}}$ which have the same orientation like $\mathbf{t}_3$. Knowing this fact we have to consider $\theta_3^s = 2\pi - \theta_3 = 7\pi/5$ [1] and consequently we arrive to the interval determining angle $\overline{\theta}_0$

$$\overline{\mathcal{I}}_0 = (\underbrace{-f_4^{-1}\left(\frac{7}{5}\pi\right)}_{\approx -\frac{\pi}{18}}, 0).$$

Finally we have the interval $\overline{\mathcal{I}} = -\overline{\mathcal{I}}_0 \cap \overline{\mathcal{I}}_3$,[2] which can be rewritten as

$$\overline{\mathcal{I}} = \left(0, \frac{\pi}{18}\right).$$

Choosing the value $\overline{\theta} = \pi/19$ we get two TC-interpolants, see Fig. 5.3.

Moreover, the first TC-interpolant is defined for the $\overline{\theta}_3$ lying in the interval

$$\overline{\mathcal{I}}_3 = (\underbrace{\frac{8}{3}\pi - \frac{11}{3}\pi}_{\frac{7}{4}\pi}, 2\pi)$$

[1] The upper left index **s** means symmetrical value with respect to the line $\theta_3 = \pi$. We will do the similar with the interval, i.e. for the interval $\mathcal{I} = (a, b)$ we obtain $\mathcal{I}^s = (2\pi - b, 2\pi - a)$.

[2] By the interval $-\mathcal{I}$ we mean the symmetrical interval to the interval $\mathcal{I} = (a, b)$, i.e. $-\mathcal{I} = (-b, -a)$.

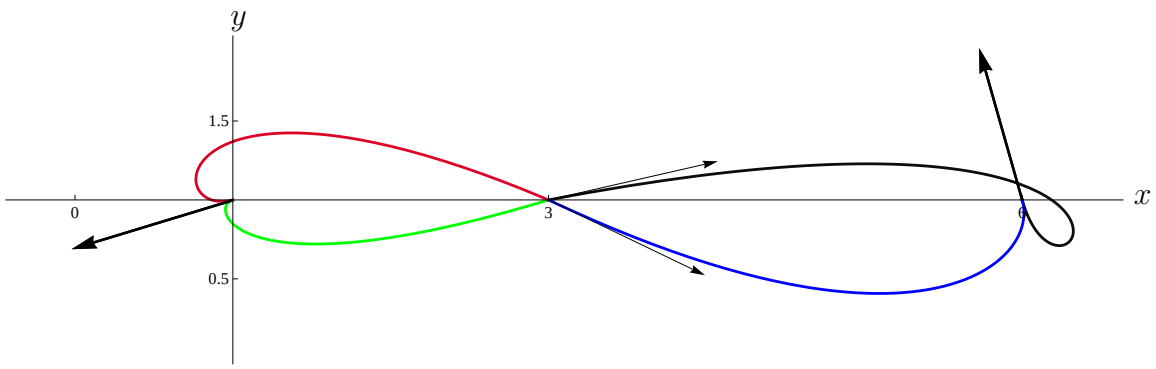


Figure 5.3: Two pairs of TC-interpolants. Green-black curve constructed using the item (1a) and red-blue curve is done according to the item (1b).

which implies the interval $\overline{\mathcal{I}}_0$ as follows

$$\overline{\mathcal{I}}_0 = (\underbrace{-\frac{4}{3}\pi + \frac{3}{5}\pi}_{-\frac{11}{15}\pi}, 0).$$

Because we keep the same orientation we have to transform the range of the interval $\overline{\mathcal{I}}_3$ to $\overline{\mathcal{I}}_3^s = (0, \pi/4)$ and the final interval $\overline{\mathcal{I}} = -\overline{\mathcal{I}}_0 \cap \overline{\mathcal{I}}_3^s = (0, \pi/4)$. Using the value $\overline{\theta} = \pi/7$ we get two TC-interpolants, see Fig. 5.3.

It remains to show the construction for G^1 Hermite data having $\theta_3 > \pi$, for example

$$\mathbf{P}_0 = (0,0)^\top, \mathbf{P}_3 = (6,1)^\top, \quad \theta_0 = -\frac{7}{10}\pi, \theta_3 = \frac{28}{15}\pi.$$

We apply similar approach as for previous data, except the change of orientation. Therefore, we arrive at the intervals

$$\overline{\mathcal{I}}_3 = (0, \underbrace{\frac{4}{3}\pi - \frac{7}{10}\pi}_{\frac{19}{10}\pi}), \ \overline{\mathcal{I}}_0 = (-\pi, 0),$$

which give us $\overline{\mathcal{I}} = (0, 19\pi/30)$ and two TC-interpolants for specific value $\overline{\theta} = \pi/7$ are shown in Fig. 5.4. The first TC-interpolants are also defined for the angle $\overline{\theta}_3$ in the interval

$$\overline{\mathcal{I}}_3 = (\underbrace{f_3^{-1}\left(-\frac{7}{10}\pi\right)}_{\approx\frac{44}{23}\pi}, 2\pi),$$

which figures out the interval

$$\overline{\mathcal{I}}_0 = \langle -\pi, \underbrace{-f_3^{-1}\left(\frac{28}{15}\pi\right)}_{\approx-\frac{23}{29}\pi}) \cup \underbrace{(-f_4^{-1}\left(\frac{28}{15}\pi\right)}_{\approx-\frac{\pi}{80}}, 0\rangle$$

and the final interval is $\overline{\mathcal{I}} = -\overline{\mathcal{I}}_0 \cap \overline{\mathcal{I}}_3^s = (0, \pi/80)$. For the random value $\overline{\theta} = \pi/85$ we get a pair of appropriate TC-interpolants, see Fig. 5.4. ♦

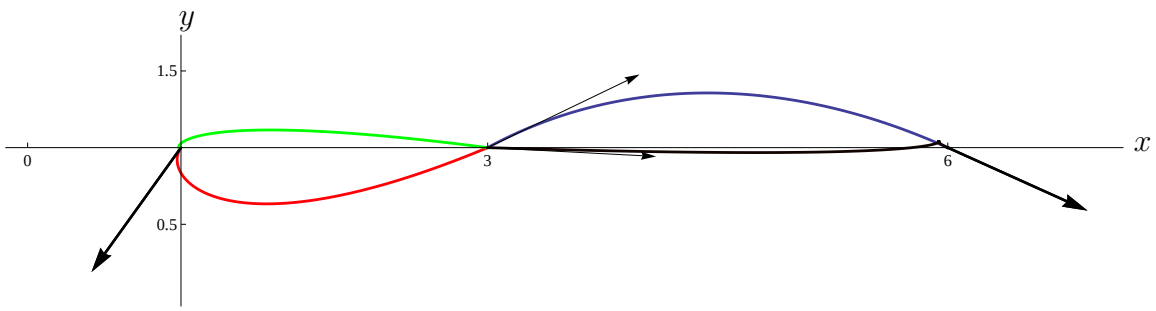


Figure 5.4: Two pairs of TC-interpolants. Green-black curve constructed using the item (2a) and red-blue curve is done according to the item (2b).

5.2 C^1 interpolation

In this section we describe C^1 Hermite interpolation by TC-interpolants. We prove that at most two TC-interpolants are always necessary to match this data.

Theorem 15. *Let us consider two TC-interpolants* $\mathbf{r}(t)$ *and* $\overline{\mathbf{r}}(t), t \in [0,1]$ *which are determined by the control points* $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ *and* $\overline{\mathbf{P}}_0, \overline{\mathbf{P}}_1, \overline{\mathbf{P}}_2, \overline{\mathbf{P}}_3$*. Such two TC-interpolants are* C^1 *connected if and only if*

$$u_1 = \overline{u}_0 \ \wedge \ v_1 = \overline{v}_0 \ \wedge \ \mathbf{P}_3 = \overline{\mathbf{P}}_0. \tag{5.1}$$

Proof. We want to prove that the control points $\mathbf{P}_2, \mathbf{P}_3$ $(= \overline{\mathbf{P}}_0)$ and $\overline{\mathbf{P}}_1$ are collinear and that the distances $|\mathbf{P}_2 - \mathbf{P}_3|$ and $|\overline{\mathbf{P}}_1 - \overline{\mathbf{P}}_0|$ hold.

According to the expression of the control points of TC-interpolant, see equation (4.9), we can rewrite the collinearity condition as

$$\begin{aligned} \mathbf{P}_3 - \mathbf{P}_2 &= \alpha(\overline{\mathbf{P}}_1 - \overline{\mathbf{P}}_0), \ \alpha \neq 0, \\ \mathbf{P}_2 + \tfrac{1}{3}(u_1^2 - v_1^2, 2u_1v_1)^\top - \mathbf{P}_2 &= \alpha(\overline{\mathbf{P}}_0 + \tfrac{1}{3}(u_1^2 - v_1^2, 2u_1v_1)^\top - \overline{\mathbf{P}}_0), \ \alpha \neq 0, \\ \tfrac{1}{3}(u_1^2 - v_1^2, 2u_1v_1)^\top &= \alpha\tfrac{1}{3}(\overline{u}_0^2 - \overline{v}_0^2, 2\overline{u}_0\overline{v}_0)^\top, \ \alpha \neq 0. \end{aligned}$$

If $\alpha = 1$, the statement of the theorem follows.

□

Corollary 16. *If two TC-interpolants are* C^1 *connected then their preimages are* C^0 *connected, but opposite statement does not hold, see Fig. 5.5.*

In Chapter 4 we have found out for which G^1 Hermite data there exists one TC-interpolant. Now, we want to derive for which C^1 Hermite data there exist two TC-interpolants C^1 connected. We can formulate the following theorem.

Theorem 17. *For any* C^1 *Hermite data there always exist four pairs of* C^1 *joined TC-interpolants.*

Proof. For the sake of simplicity we can consider

$$\mathbf{A} = (0,0)^\top \quad \text{and} \quad \mathbf{B} = (b,0)^\top \tag{5.2}$$

with their tangent vectors $\mathbf{t_A}$ and $\mathbf{t_B}$. Further, the tangent vectors determine the control points of the control polygon of TC-interpolants, i.e., $\mathbf{t_A}$ gives $\mathbf{P}_1 = (p_{1x}, p_{1y})^\top$ and $\mathbf{t_B}$ determines $\overline{\mathbf{P}}_2 = (p_{2x}, p_{2y})^\top$.

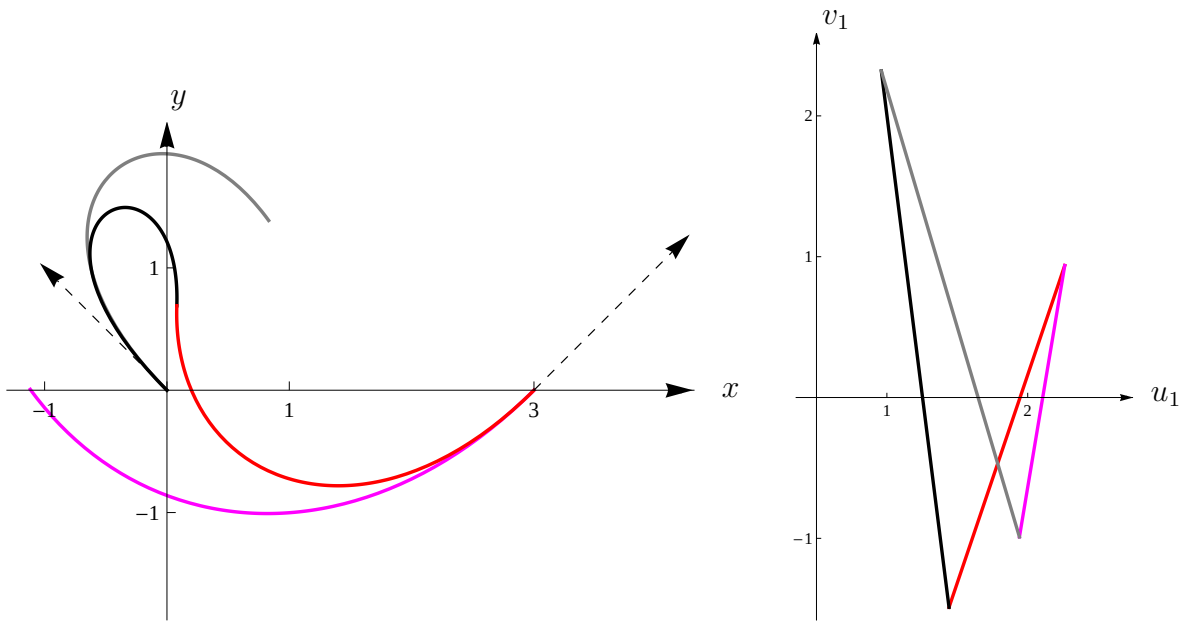


Figure 5.5: Situation describing the statement in Corollary 16, where C^0 continuity of preimages of TC-interpolants do not give C^1 continuity of TC-interpolants.

First, we have to solve the system of equations

$$\begin{aligned} p_{1x} &= \tfrac{1}{3}(u_0^2 - v_0^2), \\ p_{1y} &= \tfrac{2}{3}u_0 v_0, \end{aligned} \tag{5.3}$$

for unknowns u_0, v_0. We get four solutions

$$\begin{aligned} u_0 &= (-1)^m \frac{\sqrt{\frac{3}{2}}\, p_{1y}}{\sqrt{(-1)^n \sqrt{p_{1x}^2 + p_{1y}^2} - p_{1x}}}, \\ v_0 &= (-1)^m \sqrt{\tfrac{3}{2}} \sqrt{(-1)^n \sqrt{p_{1x}^2 + p_{1y}^2} - p_{1x}}, \quad \{m, n\} \in \{1, 2\}, \end{aligned} \tag{5.4}$$

where two pairs are real and two pairs are complex solution (see Fig. 5.8, where the equations (5.3) can be seen as an intersection of two hyperbolas). It is important to emphasize that if we set $p_{1y} = 0$ we get zero denominator in the expression of u_0. Such situation we have to solve in a different way. Since we set $p_{1y} = 0$ then the equation $\frac{2}{3}u_0 v_0 = 0$ describes two lines $u_0 = 0$ and $v_0 = 0$ and the solution, i.e., the intersection of the hyperbola with its axes, is in the form

$$u_0 = \pm\sqrt{3p_{1x}},\ v_0 = 0, \quad \text{and} \quad u_0 = \pm i\sqrt{3p_{1x}},\ v_0 = 0, \tag{5.5}$$

where the real ones are exactly the vertices of the hyperbola. In further text we assume $p_{1y} \neq 0$. If we substitute (5.4) to (4.9) we get points $\mathbf{P}_2, \mathbf{P}_3$ dependent only on parameters u_1, v_1.

Next, we have to solve the system of equations

$$\begin{aligned} b - p_{2x} &= \tfrac{1}{3}(\overline{u}_1^2 - \overline{v}_1^2), \\ p_{2y} &= \tfrac{2}{3}\overline{u}_1 \overline{v}_1. \end{aligned} \tag{5.6}$$

We solve this system for unknowns $\overline{u}_1, \overline{v}_1$, and we obtain the solution

$$\begin{aligned} \overline{u}_1 &= (-1)^m \frac{\sqrt{\frac{3}{2}}\, p_{2y}}{\sqrt{(-1)^n \sqrt{(p_{2x}-b)^2+p_{2y}^2}+p_{2x}-b}}, \\ \overline{v}_1 &= (-1)^m \sqrt{\frac{3}{2}} \sqrt{(-1)^n \sqrt{(p_{2x}-b)^2+p_{2y}^2}+p_{2x}-b}, \quad \{m,n\} \in \{1,2\}, \end{aligned} \tag{5.7}$$

where as in the equation (5.4) we have two real and two complex solutions and we also have to pay attention what happens if we set $p_{2y} = 0$. Analogous to the setting $p_{1y} = 0$ we arrive to

$$\overline{u}_1 = \pm\sqrt{3(b-p_{2x})},\ v_0 = 0, \quad \text{and} \quad u_0 = \pm i\sqrt{3(b-p_{2x})},\ v_0 = 0. \tag{5.8}$$

Using the expression of the control points of the TC-interpolant we rewrite the point $\overline{\mathbf{P}}_0$ as

$$\overline{\mathbf{P}}_0 = \overline{\mathbf{P}}_2 - \frac{1}{3}(\overline{u}_0^2 - \overline{v}_0^2 + \overline{u}_0\overline{u}_1 - \overline{v}_0\overline{v}_1, 2\overline{u}_0\overline{v}_0 + \overline{u}_0\overline{v}_1 + \overline{u}_1\overline{v}_0)^\top.$$

If we substitute for $\overline{u}_1, \overline{v}_1$ from the equation (5.7) we get a point $\overline{\mathbf{P}}_0$ dependent on the variables $\overline{u}_0, \overline{v}_0$ and it can be written as $\overline{\mathbf{P}}_0 = (g_x, g_y)^\top$, where

$$g_x(\overline{u}_0, \overline{v}_0) = \underbrace{c_{11}}_{-\frac{1}{3}}\overline{u}_0^2 + \underbrace{c_{22}}_{\frac{1}{3}}\overline{v}_0^2 + c_{13}\overline{u}_0 + c_{23}\overline{v}_0 + c_{33}, \quad c_{ij} \in \mathbb{R} \tag{5.9}$$

and

$$g_y(\overline{u}_0, \overline{v}_0) = \underbrace{q_{12}}_{-\frac{2}{3}}\overline{u}_0\overline{v}_0 + q_{13}\overline{u}_0 + q_{23}\overline{v}_0 + q_{33}, \quad q_{ij} \in \mathbb{R}. \tag{5.10}$$

Let us emphasize that the point $\mathbf{P}_3$ depends on the parameters u_1, v_1, i.e.,

$$\mathbf{P}_3 = \mathbf{P}_1 + \frac{1}{3}(u_1^2 - v_1^2 + u_0u_1 - v_0v_1, 2u_1v_1 + u_0v_1 + u_1v_0)^\top,$$

where u_0 and v_0 are obtained from equation (5.4). Consequently, the final point can be written as $\mathbf{P}_3 = (f_x, f_y)^\top$, where f_x, f_y have the forms

$$f_x(u_1, v_1) = \underbrace{h_{11}}_{\frac{1}{3}}u_1^2 + \underbrace{h_{22}}_{-\frac{1}{3}}v_1^2 + h_{13}u_1 + h_{23}v_1 + h_{33}, \tag{5.11}$$

$$f_y(u_1, v_1) = \underbrace{k_{12}}_{\frac{2}{3}}u_1v_1 + k_{13}u_1 + k_{23}v_1 + k_{33}. \tag{5.12}$$

Using condition from the equation (5.1) we arrive at the expression

$$\mathbf{P}_3 = \overline{\mathbf{P}}_0 \ \Rightarrow\ (g_x, g_y)^\top = (f_x, f_y)^\top \ \Rightarrow (\underbrace{g_x - f_x}_{w_x}, \underbrace{g_y - f_y}_{w_y})^\top = (0,0)^\top.$$

If we substitute $u_1 = \overline{u}_0, v_1 = \overline{v}_0$ we obtain two quadratic surfaces

$$\begin{aligned} w_x(u_1, v_1) &= \underbrace{m_{11}}_{\pm\frac{2}{3}}u_1^2 + \underbrace{m_{22}}_{\mp\frac{2}{3}}v_1^2 + m_{13}u_1 + m_{23}v_1 + m_{33}, \\ w_y(u_1, v_1) &= \underbrace{n_{12}}_{\pm\frac{2}{3}}u_1v_1 + n_{13}u_1 + n_{23}v_1 + n_{33}, \end{aligned} \tag{5.13}$$

with the coefficients $m_{ij} = c_{ij} - h_{ij}, n_{ij} = q_{ij} - k_{ij}, m_{ij}, n_{ij} \in \mathbb{R}$. Looking for the zero set, we obtain two conic sections

$$w_x(u_1, v_1) = 0 \quad \text{and} \quad w_y(u_1, v_1) = 0$$

with the same center[3] $\mathbf{S} = (s_x, s_y)^\top$. If we want to decide about the type of conic section, we have to focus on two invariants[4]

$$\Delta(a) = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{vmatrix}, \ \delta(a) = \begin{vmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{vmatrix} \tag{5.14}$$

of the arbitrary conic section

$$a(x, y) = a_{11}x^2 + 2a_{12}xy + a_{22}x^2 + 2a_{13}x + 2a_{23}y + a_{33} = 0.$$

By the computing of the invariants (5.14) of the conic sections w_x, w_y we get

$$\delta(w_x) = \begin{vmatrix} \pm\frac{2}{3} & 0 \\ 0 & \mp\frac{2}{3} \end{vmatrix} = -\frac{4}{9} < 0, \ \delta(w_y) = \begin{vmatrix} 0 & \pm\frac{2}{3} \\ \pm\frac{2}{3} & 0 \end{vmatrix} = -\frac{4}{9} < 0$$

and, moreover,

$$\Delta(w_x) \neq 0, \ \Delta(w_y) \neq 0,$$

which imply that the conic sections w_x and w_y are hyperbolas. Only for the value $v_0 = 0$ or $\overline{v}_1 = 0$ (see equation (5.5), (5.8)) we obtain two intersecting lines (singular conic section).

Let us solve the system of equation

$$w_x(u_1, v_1) = 0 \ \wedge \ w_y(u_1, v_1) = 0. \tag{5.15}$$

Knowing that both function w_x, w_y are of degree 2 and using Bézout theorem we obtain four solutions. We are interested only in real solutions which do not have to exist for two arbitrary hyperbolas. Now, we have to use the properties of the axes and asymptotes of these two conic sections, i.e., the conic section w_x has axes $u_1 = s_x$ and $v_1 = s_y$ and conic section w_y has asymptotes $u_1 = s_x$ and $v_1 = s_y$, which implies that these two curves have always two real intersections, see Fig. 5.8.

We have found the intersection of two conic section which gives the value of joining point. This point always exists and therefore we can interpolate arbitrary Hermite data with C^1 smoothness. □

5.2.1 A loop on TC-interpolatns

In this section we discuss the quality of TC-interpolant matching C^1 Hermite data, i.e., whether it contains a loop or not. Let us assume that PH cubic has parametric expression $\mathbf{r}(t)$ which is dependent on parameters u_0, v_0, u_1, v_1, see Equation (4.9). If we want to determine a selfintersection we have to solve the following system of equations

$$\mathbf{r}(t^+) = \mathbf{r}(t^-) \ \wedge \ t^+ \neq t^-,$$

[3] Using the expression of the center of the hyperbola we can check that the hyperbolas described above have the same values of the center.

[4] In general, there are two invariants, the first one is determinant of the conic section and the second one is determinant of quadratic terms of conic section. The notation of these two invariants can be different depending on a source.

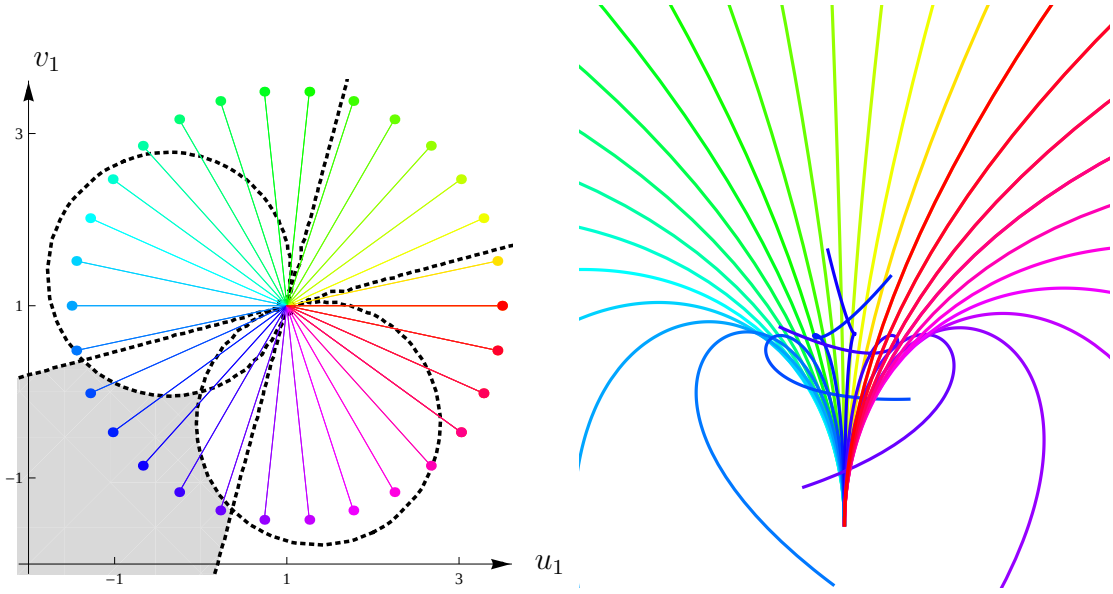


Figure 5.6: Left: Behavior of the point $(u_1, v_1)^\top$ where we fix $(u_0, v_0)^\top$; Right: Corresponding TC-interpolants.

for unknowns t^+, t^-. We obtain following expressions

$$t^+ = \frac{u_0^2 + v_0^2 - u_0u_1 - v_0v_1 + \sqrt{3}(u_1v_0 - u_0v_1)}{(u_0 - u_1)^2 + (v_0 - v_1)^2}, \tag{5.16}$$

$$t^- = \frac{u_0^2 + v_0^2 - u_0u_1 - v_0v_1 - \sqrt{3}(u_1v_0 - u_0v_1)}{(u_0 - u_1)^2 + (v_0 - v_1)^2}. \tag{5.17}$$

The PH cubic as a Bézier curve is given for the parameter in the domain $[0, 1]$, and therefore we can easily check the domain, where the self intersection is, i.e.,

$$\Upsilon = \{0 \le t^+ \le 1 \wedge 0 \le t^- \le 1\}.$$

This domain is bounded by the curves obtained from the previous equations, which can be rewritten as

$$\begin{aligned} &u_0^2 + v_0^2 - u_0u_1 - v_0v_1 \pm \sqrt{3}(u_1v_0 - u_0v_1) = 0, \\ &u_0^2 + v_0^2 - u_0u_1 - v_0v_1 \pm \sqrt{3}(u_1v_0 - u_0v_1) - ((u_0 - u_1)^2 + (v_0 - v_1)^2) = 0. \end{aligned} \tag{5.18}$$

For each TC-interpolant we fix either u_0, v_0 or u_1, v_1 and then the equations (5.18) describe two lines and two circles which bound the domain, either $\Upsilon(u_1, v_1) = \Upsilon_1$ or $\Upsilon(\overline{u}_0, \overline{v}_0) = \Upsilon_0$, where a loop appears, see Fig. 5.6.

To decide if a computed pair of TC-interpolants contains a loop, we only evaluate Equations (5.16) and (5.17), because we know all the values u_0, v_0, u_1, v_1.

5.2.2 Rational curves on Blaschke cylinder

We describe some properties of PH curves in dual space. This concept was first introduced in works of *W. Blaschke*. Further, Pottmann and Peternell inspired by Laguerre geometry set several statement about the PH curves, c.f. [89], [96], [97], [98]. Let us mention the theorem describing PH curve on Blaschke cylinder.

Theorem 18. *The image of any plane PH curve on Blaschke cylinder is a spatial rational curve.*

Proof. Can be found in [89]. □

The aim is to describe the behavior of two TC-interpolants connected in G^1 or C^1 smoothness. We can formulate the following theorem.

Theorem 19. *Images of any two TC-interpolants joined with C^1 or G^1 continuity are G^1 connected on Blaschke cylinder.*

Proof. The TC-interpolant given by the control points

$$\mathbf{P}_0 = (a_x, a_y)^\top,\ \mathbf{P}_1 = (b_x, b_y)^\top, \mathbf{P}_2 = (c_x, c_y)^\top, \mathbf{P}_3 = (d_x, d_y)^\top \tag{5.19}$$

has a parametric expression

$$\mathbf{r}(t) = (a_x(1-t)^3+3b_x(1-t)^2t+3c_x(1-t)t^2+d_xt^3, a_y(1-t)^3+3b_y(1-t)^2t+3c_y(1-t)t^2+d_yt^3)^\top,$$

where $t \in [0, 1]$. Then the first derivative has an expression

$$\begin{aligned}\mathbf{r}'(t) = \ & (3(-a_x + b_x + 2(a_x - 2b_x + c_x)t + (-a_x + 3b_x - 3c_x + d_x)t^2),\\ & 3(-a_y + b_y + 2(a_y - 2b_y + c_y)t + (-a_y + 3b_y - 3c_y + d_y)t^2))^\top,\end{aligned}$$

with PH property, i.e. $\langle \mathbf{r}'(t), \mathbf{r}'(t)\rangle = \sigma(t)^2, \sigma \in \mathbb{R}[t]$. The tangent line is written as

$$\langle \mathbf{n}(t), \mathbf{x}\rangle + x_3 = 0, \quad \mathbf{x} = (x_1, x_2)^\top,$$

where $\mathbf{n}(t) = (n_x(t), n_y(t))^\top$ is unit normal vector and $x_3 = -\langle \mathbf{n}(t), \mathbf{r}(t)\rangle$. Since the PH cubic is of degree 3, its curve on Blaschke cylinder is of degree 5, because the third coordinate x_3 is a dot product of $\mathbf{n}$ (degree 2) and $\mathbf{r}(t)$ (degree 3). The curve $\mathbf{c}(t)$ on Blaschke cylinder is written as

$$\mathbf{c}(t) = (n_x(t), n_y(t), x_3(t))^\top,$$

where

$$\begin{aligned}x_3(t) = \ & (-a_y(t-1)^2(b_x + 2(c_x + b_x)t + (b_x - 2c_x + d_x)t^2)+\\ & a_x(t-1)^2(b_y + 2(c_y - b_y)t + (b_y - 2c_y + d_y)t^2)+\\ & t^2((c_xd_y + c_yd_x)t^2 - b_y(t-1)(3c_x(t-1) - 2d_xt)+\\ & b_x(t-1)(3c_y(t-1) - 2d_yt)))/\sigma(t).\end{aligned}$$

Computing the first derivative of the curve $\mathbf{c}(t)$ and evaluating it for the values $t = 0, t = 1$ we arrive at

$$\mathbf{c}'(0) = \alpha_0(a_x - b_x, a_y - b_y, a_x(b_x - a_x) + a_y(b_y - a_y))^\top,$$

where

$$\alpha_0 = 2\frac{b_xc_y - b_yc_x + a_y(c_x + b_x) + a_x(b_y - c_y)}{((a_x - b_x)^2 + (a_y - b_y)^2)^{3/2}}$$

and

$$\mathbf{c}'(1) = \alpha_1(c_x - d_x, c_y - d_y, d_x(d_x - c_x) + d_y(d_y - c_y))^\top, \tag{5.20}$$

where

$$\alpha_1 = 2\frac{c_x d_y - c_y d_x + b_y(d_x - c_x) + b_x(c_y - d_y)}{((c_x - d_x)^2 + (c_y - d_y)^2)^{3/2}}.$$

Let us focus on two TC-interpolants and their connection. We have two curves $\mathbf{c}(t)$ and $\overline{\mathbf{c}}(t)$ on Blaschke cylinder, the first TC-interpolant is determined by the control points (5.19) and the second TC-interpolant is given by the points

$$\overline{\mathbf{P}}_0 = (\overline{a}_x, \overline{a}_y)^\top,\ \overline{\mathbf{P}}_1 = (\overline{b}_x, \overline{b}_y)^\top,\ \overline{\mathbf{P}}_2 = (\overline{c}_x, \overline{c}_y)^\top\ \overline{\mathbf{P}}_3 = (\overline{d}_x, \overline{d}_y)^\top.$$

By the investigation of the first derivatives at the end points, we get $\mathbf{c}'(1)$, see equation (5.20) and the first derivative

$$\overline{\mathbf{c}}'(0) = \overline{\alpha}_0(\overline{a}_x - \overline{b}_x, \overline{a}_y - \overline{b}_y, \overline{a}_x(\overline{b}_x - \overline{a}_x) + \overline{a}_y(\overline{b}_y - \overline{a}_y))^\top.$$

If two TC-interpolants are G^1 or C^1 connected, they have common point $\mathbf{P}_3 = \overline{\mathbf{P}}_0$ and the corresponding tangent vectors have to be collinear, or similarly

$$|\ \underbrace{\mathbf{P}_3 - \mathbf{P}_2}_{(d_x - c_x, d_y - c_y)^\top}\ | = \lambda |\ \underbrace{\overline{\mathbf{P}}_1 - \overline{\mathbf{P}}_0}_{(\overline{b}_x - \overline{a}_x, \overline{b}_y - \overline{a}_y)^\top}\ |,\ \lambda \in \mathbb{R},$$

where $\lambda = 1$ yields C^1 continuity.

Let us focus on the vectors $\mathbf{c}'(1)$ and $\overline{\mathbf{c}}'(0)$ and we want to decide whether they are also collinear. It is obvious that the first two parts are collinear and therefore we investigate only the third part, where we substitute $\overline{\mathbf{P}}_0 = \mathbf{P}_3$

$$d_x(\underbrace{d_x - c_x}_{t_x}) + d_y(\underbrace{d_y - c_y}_{t_y}) \stackrel{?}{=} d_x(\underbrace{\overline{b}_x - d_x}_{\beta t_x}) + d_y(\underbrace{\overline{b}_y - d_y}_{\beta t_y}),\ \beta \in \mathbb{R},$$

where the equality holds, because the points $\mathbf{P}_2, \mathbf{P}_3 (= \overline{\mathbf{P}}_0)$ and $\overline{\mathbf{P}}_1$ are collinear. The remaining part is to find whether the first two parts are also multiplied by β, i.e.,

$$(\underbrace{c_x - d_x}_{-t_x}, \underbrace{c_y - d_y}_{-t_y})^\top = (\underbrace{d_x - \overline{b}_x}_{-\beta t_x}, \underbrace{d_y - \overline{b}_y}_{-\beta t_y})^\top,$$

which gives the desired continuity on Blaschke cylinder. □

Corollary 20. *It is obvious that the previous theorem holds not only for PH cubics but also for* ***arbitrary*** *Bézier cubics joined in G^1 or C^1 continuity, although they do not possess rational curve on Blaschke cylinder.*

Moreover from the Blaschke cylinder we cannot distinguish whether the curves are C^1 or G^1 connected.

In Fig. 5.7 TC-interpolants G^1 and C^1 connected are shown and there is visualized the correspondence on Blaschke cylinder, where appropriate TC-interpolants are shown.

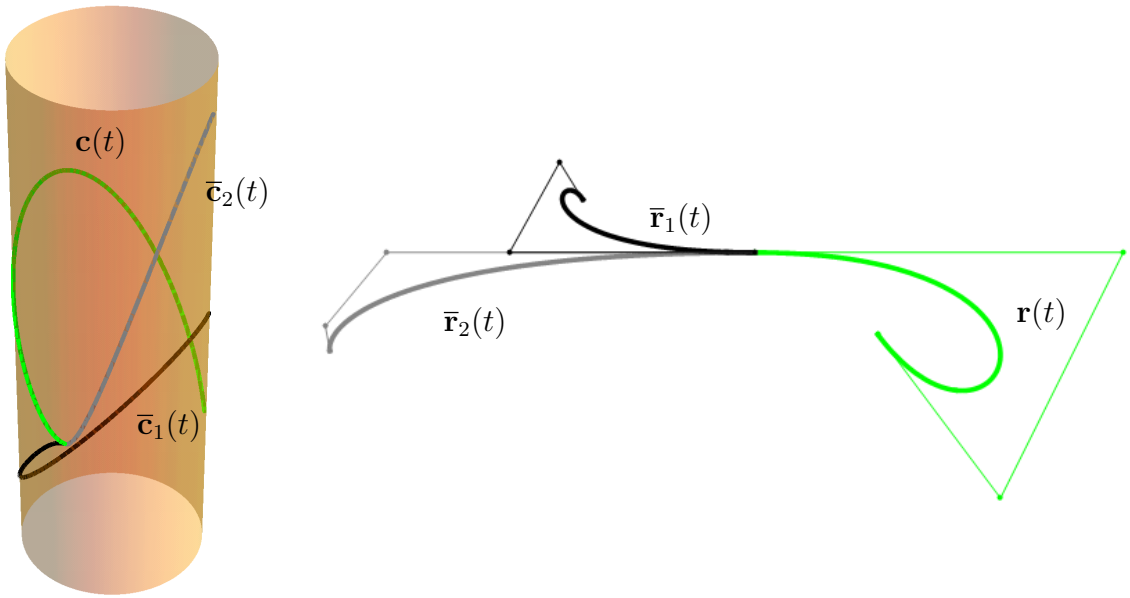


Figure 5.7: Left: Rational curves on Blaschke cylinder which are G^1 connected. Right: G^1 connected (green and black) and C^1 connected (green and gray) TC-interpolants drawn with their control polygons.

5.3 Example

In this section we show C^1 Hermite interpolation on one example.

Example 5. For C^1 Hermite data

$$\mathbf{A} = (0,0)^\top,\ \mathbf{B} = (5,0)^\top,\ \mathbf{t_A} = (6,6)^\top,\ \mathbf{t_B} = (6,3)^\top$$

find all TC-interpolants and decide whether they contain a loop.

Knowing the tangent vector we can directly express the control points $\mathbf{P}_1 = (2,2)^\top$ and $\overline{\mathbf{P}}_2 = (3,-1)^\top$ and obviously $\mathbf{P}_0 = (0,0), \overline{\mathbf{P}}_3 = (5,0)^\top$.

By solving the system of equations (5.3) for unknown u_1, v_1 we obtain four solutions which are described by (5.4). We get

$$\begin{aligned}(u_0^{\mathsf{a}}, v_0^{\mathsf{a}})^\top &= (-2.691, -1.114)^\top,\quad (u_0^{\mathsf{b}}, v_0^{\mathsf{b}})^\top = (2.691, 1.114)^\top\\ (u_0^{\mathsf{c}}, v_0^{\mathsf{c}})^\top &= (0 + 1.114i, 0 - 2.691i)^\top,\quad (u_0^{\mathsf{d}}, v_0^{\mathsf{d}})^\top = (0 - 1.114i, 0 + 2.691i)^\top\end{aligned}$$

and we pick up only the real solutions $(u_0^{\mathsf{a}}, v_0^{\mathsf{a}})^\top, (u_0^{\mathsf{b}}, v_0^{\mathsf{b}})^\top$, which are moreover symmetrical with respect to the origin $(0,0)^\top$. The solution of the system of equations (5.3) can be also seen as to find the intersection of two hyperbolas, see Fig. 5.8.

If we substitute $(u_0^{\mathsf{a}}, v_0^{\mathsf{a}})^\top, (u_0^{\mathsf{b}}, v_0^{\mathsf{b}})^\top$ to the expression of the control point $\mathbf{P}_3$ we obtain $\mathbf{P}_3 = (f_x(u_1, v_1), f_y(u_1, v_1))^\top$, where f_x, f_y are gained from the equation (5.11) and (5.12). These functions have formulas

$$\begin{aligned}f_x^{\mathsf{a}}:&\ 2 + \tfrac{1}{3}(u_1^2 - v_1^2 - 2.691u_1 + 1.114v_1),\\ f_y^{\mathsf{a}}:&\ 2 + \tfrac{1}{3}(2u_1v_1 - 1.114u_1 - 2.691v_1),\quad \text{where}\quad (u_1, v_1) \in \mathbb{R}^2\end{aligned}$$

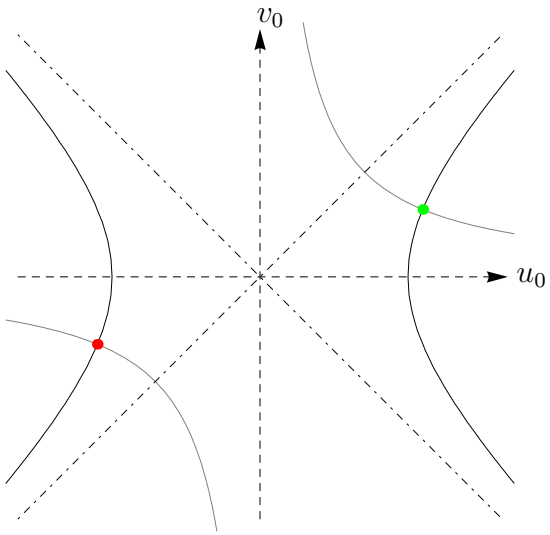


Figure 5.8: The solution of the system of equation (5.3) described as an intersection of two hyperbolas. Gray hyperbola is $2u_0v_0/3 - 2 = 0$ and black hyperbola is $(u_0^2 + v_0^2)/3 - 2 = 0$. The points $(u_0^{\mathsf{a}}, v_0^{\mathsf{a}})^\top, (u_0^{\mathsf{b}}, v_0^{\mathsf{b}})^\top$ are drawn in red and green color, respectively.

and

$$\begin{aligned} f_x^{\mathsf{b}} &: \ 2 + \tfrac{1}{3}(u_1^2 - v_1^2 + 2.691u_1 - 1.114v_1), \\ f_y^{\mathsf{b}} &: \ 2 + \tfrac{1}{3}(2u_1v_1 + 1.114u_1 + 2.691v_1), \quad \text{where} \quad (u_1, v_1) \in \mathbb{R}^2, \end{aligned}$$

which are shown in Fig. 5.9.

Now, let us focus on the equations (5.6). According to the equations (5.7), (5.9) and (5.10) we arrive at four solutions, where first two are real and other two are complex, i.e.,

$$\begin{aligned} &(\overline{u}_1^{\mathsf{a}}, \overline{v}_1^{\mathsf{a}})^\top = (-2.52, -0.595)^\top, \quad (\overline{u}_1^{\mathsf{b}}, \overline{v}_1^{\mathsf{b}})^\top = (2.52, 0.595)^\top, \\ &(\overline{u}_1^{\mathsf{c}}, \overline{v}_1^{\mathsf{c}})^\top = (0.595i, -2.52i)^\top, \quad (\overline{u}_1^{\mathsf{d}}, \overline{v}_1^{\mathsf{d}})^\top = (-0.595i, 2.52i)^\top. \end{aligned}$$

By substituting the real values $(\overline{u}_1^{\mathsf{a}}, \overline{v}_1^{\mathsf{a}}), (\overline{u}_1^{\mathsf{b}}, \overline{v}_1^{\mathsf{b}})$ to the expression of the point $\overline{\mathbf{P}}_0$ we obtain

$$\begin{aligned} g_x^{\mathsf{a}} &: \ 3 - \tfrac{1}{3}(\overline{u}_0^2 - \overline{v}_0^2 - 2.52\overline{u}_0 + 0.595\overline{v}_0), \\ g_y^{\mathsf{a}} &: \ -1 - \tfrac{1}{3}(2\overline{u}_0\overline{v}_0 - 0.595\overline{u}_0 - 2.52\overline{v}_0), \end{aligned}$$

$$\begin{aligned} g_x^{\mathsf{b}} &: \ 3 - \tfrac{1}{3}(\overline{u}_0^2 - \overline{v}_0^2 + 2.52\overline{u}_0 - 0.595\overline{v}_0), \\ g_y^{\mathsf{b}} &: \ -1 - \tfrac{1}{3}(2\overline{u}_0\overline{v}_0 + 0.595\overline{u}_0 + 2.52\overline{v}_0), \end{aligned}$$

where $(\overline{u}_0, \overline{v}_0) \in \mathbb{R}^2$.

The next step is to compute the functions w_x, w_y from the equation (5.13). In general, we have four values for w_x and four values for w_y in the form

$$\begin{aligned} w_x^{j,k} &= g_x^j - f_x^k \\ w_y^{j,k} &= g_y^j - f_y^k, \quad j, k \in \{\mathsf{a}, \mathsf{b}\}, \end{aligned}$$

namely

$$\begin{aligned} w_x^{j,j} &= -1 \pm 1.737u_1 + \tfrac{2}{3}u_1^2 \mp 0.569v_1 - \tfrac{2}{3}v_1^2, \ j \in \{\mathsf{a}, \mathsf{b}\}, \\ w_x^{j,k} &= -1 \pm 0.056u_1 + \tfrac{2}{3}u_1^2 \mp 0.173v_1 - \tfrac{2}{3}v_1^2, \ j, k \in \{\mathsf{a}, \mathsf{b}\} \wedge j \neq k, \\ w_y^{j,j} &= 3 \pm 1.737v_1 \mp 0.569u_1 + 2\tfrac{2}{3}u_1v_1, \ j \in \{\mathsf{a}, \mathsf{b}\}, \\ w_y^{j,k} &= 3 \pm 0.056v_1 \mp 0.173u_1 + 2\tfrac{2}{3}u_1v_1, \ j, k \in \{\mathsf{a}, \mathsf{b}\} \wedge j \neq k. \end{aligned} \tag{5.21}$$

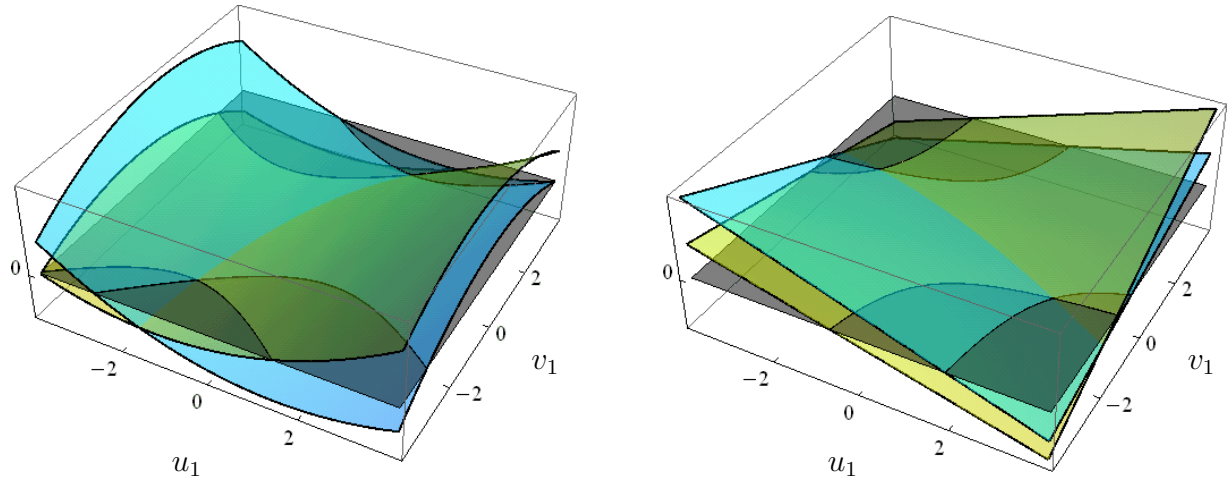


Figure 5.9: Left: Functions f_x^{a} (cyan) and f_x^{b} (yellow) with their zero sets. Right: Functions f_y^{a} (cyan) and f_y^{b} (yellow) with their zero sets.

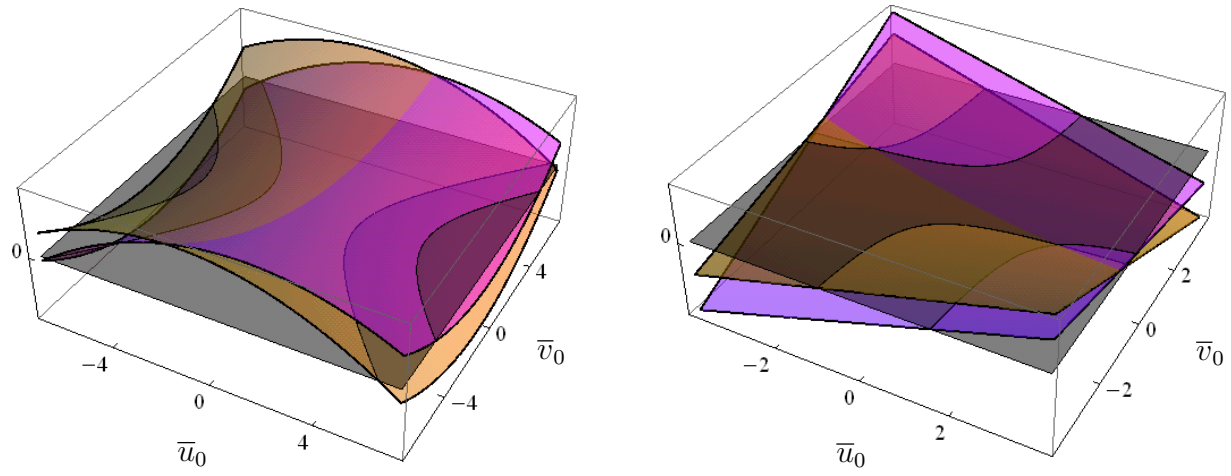


Figure 5.10: Left: Functions g_x^{a} (magenta) and g_x^{a} (orange) with their zero sets. Right: Functions g_y^{b} (magenta) and g_y^{b} (orange) with their zero sets.

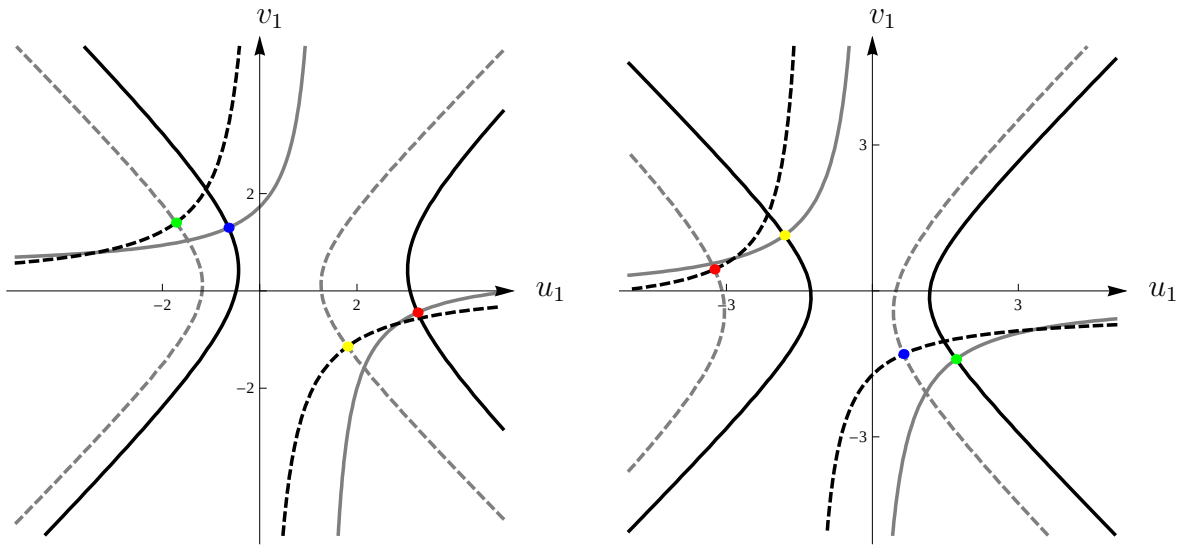


Figure 5.11: Zero sets of function w_x, w_y and their intersection (colored points). Left: $w_x^{\mathrm{a,a}} = 0$ (gray), $w_x^{\mathrm{a,b}} = 0$ (black dashed) and $w_y^{\mathrm{a,a}} = 0$ (black), $w_y^{\mathrm{a,b}} = 0$ (gray dashed). Right: $w_x^{\mathrm{b,a}} = 0$ (gray), $w_x^{\mathrm{b,b}} = 0$ (black dashed) and $w_y^{\mathrm{b,a}} = 0$ (black), $w_y^{\mathrm{b,b}} = 0$ (gray dashed).

Their zero sets, i.e., conic sections, are plotted in Fig. 5.11.

By finding the intersection points of conic sections we obtain the values u_1, v_1, which we get as a solution of the system of equations (5.15), which are shown as colored points in Fig. 5.11. We arrive at eight points (we know $(u_{1,m}^{j,k}, v_{1,m}^{j,k}) = (\overline{u}_{0,m}^{j,k}, \overline{v}_{0,m}^{j,k})$)

$$\begin{aligned}
&(u_{1,1}^{\mathrm{a,a}}, v_{1,1}^{\mathrm{a,a}})^\top = (3.245, -0.443)^\top,\ (u_{1,1}^{\mathrm{b,b}}, v_{1,1}^{\mathrm{b,b}})^\top = (-3.245, 0.443)^\top,\\
&(u_{1,2}^{\mathrm{a,a}}, v_{1,2}^{\mathrm{a,a}})^\top = (-0.639, 1.298)^\top,\ (u_{1,2}^{\mathrm{b,b}}, v_{1,2}^{\mathrm{b,b}})^\top = (0.639, -1.298)^\top,\\
&(u_{1,1}^{\mathrm{a,b}}, v_{1,1}^{\mathrm{a,b}})^\top = (-1.719, 1.403)^\top,\ (u_{1,1}^{\mathrm{b,a}}, v_{1,1}^{\mathrm{b,a}})^\top = (1.719, -1.403)^\top,\\
&(u_{1,2}^{\mathrm{a,b}}, v_{1,2}^{\mathrm{a,b}})^\top = (1.805, -1.143)^\top,\ (u_{1,2}^{\mathrm{b,a}}, v_{1,2}^{\mathrm{b,a}})^\top = (-1.805, 1.143)^\top,
\end{aligned} \tag{5.22}$$

where for values $(u_{1,j}^{k,l}, v_{1,j}^{k,l})^\top$ hold

$$(u_{1,j}^{k,l}, v_{1,j}^{k,l})^\top = -(u_{1,j}^{l,k}, v_{1,j}^{l,k})^\top, \quad \text{for} \quad j \in \{1,2\}, k, l \in \{\mathrm{a}, \mathrm{b}\},$$

$$(u_{1,j}^{k,k}, v_{1,j}^{k,k})^\top = -(u_{1,j}^{l,l}, v_{1,j}^{l,l})^\top, \quad \text{for} \quad j \in \{1,2\}, k, l \in \{\mathrm{a}, \mathrm{b}\},$$

i.e., they are symmetrical with respect to the origin $(0,0)^\top$, see Fig. 5.12.

If we substitute (5.22) to the corresponding formula (5.21), we get only four distinct points $\mathbf{P}_3 = \overline{\mathbf{P}}_0$. For the coefficients m_{13}, m_{23} and n_{13}, n_{23} of the formula w_x and w_y, see equation (5.13), hold

$$m_{j3}^{k,l} = -m_{j3}^{l,k},\ m_{j3}^{k,k} = -m_{j3}^{l,l}, \qquad n_{j3}^{k,l} = -n_{j3}^{l,k},\ n_{j3}^{k,k} = -n_{j3}^{l,l}, j \in \{1,2\},\ k, l \in \{\mathrm{a}, \mathrm{b}\}.$$

Finally, we arrive at the four pairs of TC-interpolants (all TC-interpolants have the same control points $\mathbf{P}_0, \mathbf{P}_1, \overline{\mathbf{P}}_2$ and $\overline{\mathbf{P}}_3$).

1. The first two TC-interpolants are determined by the control points

$$\mathbf{P}_{2,1}^{\mathrm{a,a}} = (-1.076, 1.192)^\top, \mathbf{P}_{3,1}^{\mathrm{a,a}} = (2.369, 0.231)^\top = \overline{\mathbf{P}}_{0,1}^{\mathrm{a,a}}, \overline{\mathbf{P}}_{1,1}^{\mathrm{a,a}} = (5.815, -0.729)^\top$$

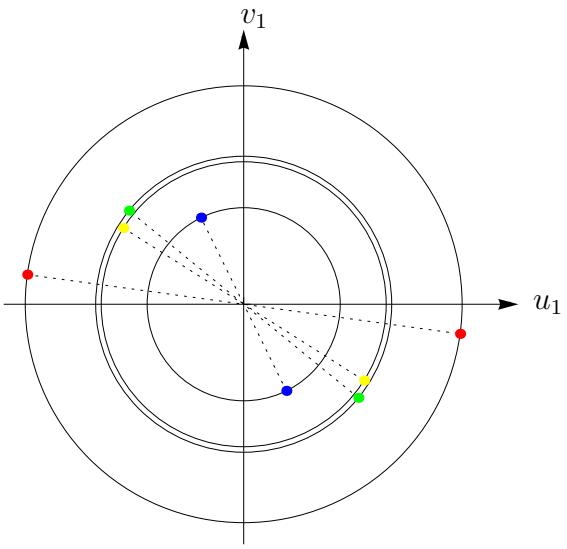


Figure 5.12: All solution of the system of equation (5.15). The same colored points are symmetrical with respect to the origin. Symmetrical points $(u_{1,1}^{\mathrm{a,a}}, v_{1,1}^{\mathrm{a,a}})^\top$ (blue), $(u_{1,2}^{\mathrm{a,a}}, v_{1,2}^{\mathrm{a,a}})^\top$ (red), $(u_{1,1}^{\mathrm{a,b}}, v_{1,1}^{\mathrm{a,b}})^\top$ (green), $(u_{1,2}^{\mathrm{a,b}}, v_{1,2}^{\mathrm{a,b}})^\top$ (yellow).

and parametric expressions have the following formulae

$$\mathbf{r}_1^{\mathrm{a,a}}(t) = (6t - 15.23t^2 + 11.59t^3, 6t - 8.423t^2 + 2.654t^3)^\top,$$
$$\overline{\mathbf{r}}_1^{\mathrm{a,a}}(t) = (2.369 + 10.338t - 18.784t^2 + 11.076t^3, 0.231 - 2.882t + 2.069t^2 + 0.581t^3)^\top,$$

where $t \in [0, 1]$. The TC-interpolants are shown in Fig. 5.13 (top left).

2. The second TC-interpolants have the control points

$$\mathbf{P}_{2,2}^{\mathrm{a,a}} = (3.056, 1.072)^\top, \mathbf{P}_{3,2}^{\mathrm{a,a}} = (2.63, 0.518)^\top = \overline{\mathbf{P}}_{0,2}^{\mathrm{a,a}}, \overline{\mathbf{P}}_{1,2}^{\mathrm{a,a}} = (2.204, -0.035)^\top$$

and parametric expressions

$$\mathbf{r}_2^{\mathrm{a,a}}(t) = (6t - 2.83t^2 - 0.539t^3, 6t - 8.782t^2 + 3.301t^3)^\top$$
$$\overline{\mathbf{r}}_2^{\mathrm{a,a}}(t) = (2.631 - 1.277t + 3.663t^2 - 0.016t^3, 0.518 - 1.662t - 1.231t^2 + 2.374t^3)^\top,$$

see Fig. 5.13 (top right).

3. The third TC-interpolants have the control points

$$\mathbf{P}_{2,1}^{\mathrm{a,b}} = (4.064, 1.38)^\top, \mathbf{P}_{3,1}^{\mathrm{a,b}} = (4.393, -0.228)^\top = \overline{\mathbf{P}}_{0,1}^{\mathrm{a,b}}, \overline{\mathbf{P}}_{1,1}^{\mathrm{a,b}} = (4.723, -1.838)^\top$$

and parametric expressions

$$\mathbf{r}_1^{\mathrm{a,b}}(t) = (6t + 0.193t^2 - 1.799t^3, 6t - 7.859t^2 + 1.63t^3)^\top,$$
$$\overline{\mathbf{r}}_1^{\mathrm{a,b}}(t) = (4.393 + 0.988t - 6.159t^2 + 5.776t^3, -0.228 - 4.827t + 7.341t^2 - 2.285t^3)^\top,$$

see Fig. 5.13 (bottom left).

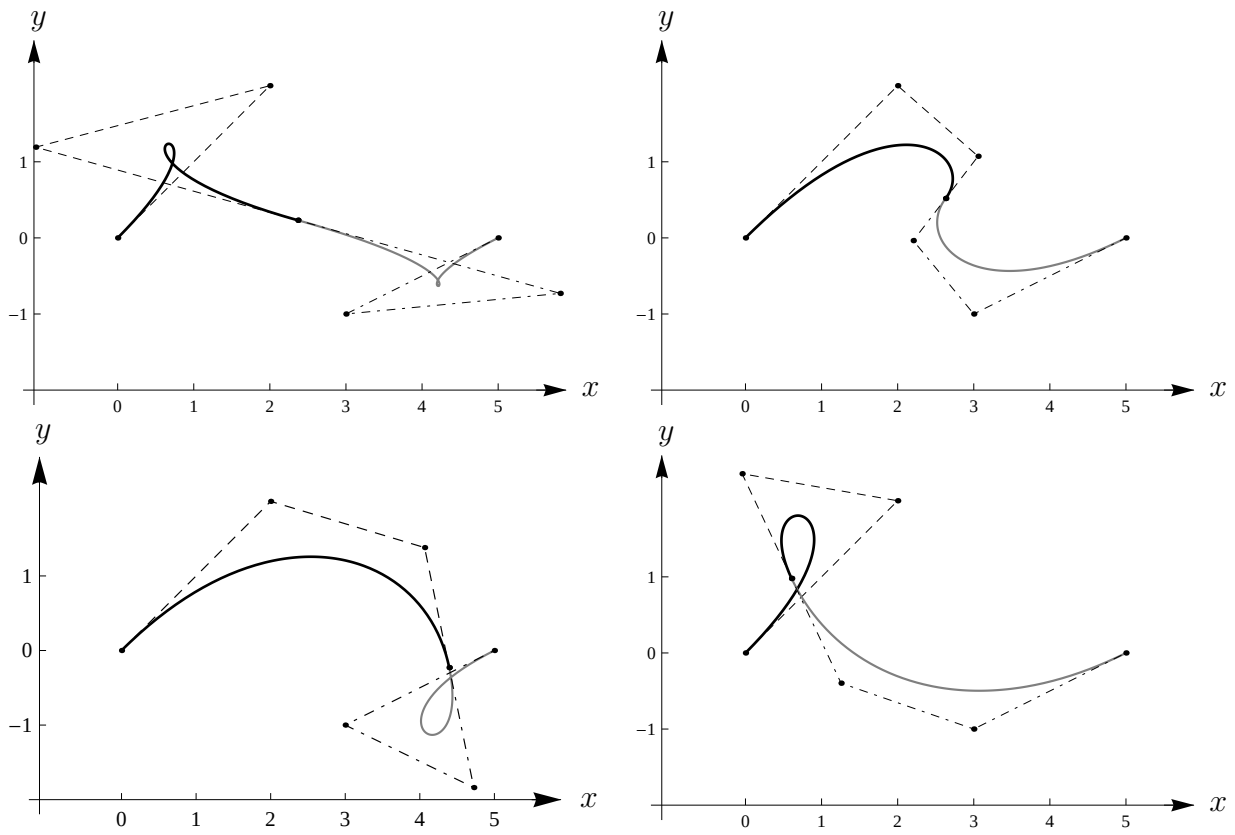


Figure 5.13: The four pairs of TC-interpolants matching given C^1 Hermite data

4. The fourth TC-interpolants have the control points

$$\mathbf{P}_{2,2}^{\mathrm{a,b}} = (-0.044, 2.355)^\top, \mathbf{P}_{3,2}^{\mathrm{a,b}} = (0.606, 0.978)^\top = \overline{\mathbf{P}}_{0,2}^{\mathrm{a,b}}, \overline{\mathbf{P}}_{1,2}^{\mathrm{a,b}} = (1.256, -0.397)^\top$$

and parametric expressions

$$\mathbf{r}_2^{\mathrm{a,b}}(t) = (6t - 12.132t^2 + 6.738t^3, 6t - 4.934t^2 - 0.086t^3)^\top$$
$$\overline{\mathbf{r}}_2^{\mathrm{a,b}}(t) = (0.606 + 1.951t + 3.279t^2 - 0.836t^3, 0.978 - 4.128t + 2.32t^2 + 0.829t^3)^\top,$$

see Fig. 5.13 (bottom right).

The remaining step is to decide about the quality of the TC-interpolants, i.e., whether they contain a loop or not. Knowing all the values u_0, v_0, u_1, v_1 and $\overline{u}_0, \overline{v}_0, \overline{u}_1, \overline{v}_1$, which describe control points of TC-interpolants we obtain the equations from the expression (5.18), which bound the domain where a loop arises.

1. Substituting the values either $(u_0^{\mathrm{a}}, v_0^{\mathrm{a}})^\top$ or $(\overline{u}_1^{\mathrm{a}}, \overline{v}_1^{\mathrm{a}})^\top$ to the equations (5.18) we get two lines and two circles, see Fig. 5.14 (left).

   To decide about the quality of TC-interpolants (containing a loop) we have to discuss where the points $(u_{1,1}^{\mathrm{a,a}}, v_{1,1}^{\mathrm{a,a}})^\top = (\overline{u}_{0,1}^{\mathrm{a,a}}, \overline{v}_{0,1}^{\mathrm{a,a}})^\top$ and $(u_{1,2}^{\mathrm{a,a}}, v_{1,2}^{\mathrm{a,a}})^\top = (\overline{u}_{0,2}^{\mathrm{a,a}}, \overline{v}_{0,2}^{\mathrm{a,a}})^\top$ lie, or, from a computational point of view, we have to check the value of t^+ and t^-, see equations (5.16), (5.17).

The point $(u_{1,1}^{a,a}, v_{1,1}^{a,a})^\top = (\overline{u}_{0,1}^{a,a}, \overline{v}_{0,1}^{a,a})^\top = (3.245, -0.443)^\top$ lies in both domains, i.e., in $\Upsilon_0 \cap \Upsilon_1$ which implies that both TC-interpolants contain a loop, see Fig. 5.14 (left), or by the computing the parameter values we get for the first TC-interpolant

$$t^+ = 0.235, t^- = 0.702 \;\Rightarrow\; t^+, t^- \in [0,1]$$

and for the second TC-interpolant

$$t^+ = 0.719, t^- = 0.401 \;\Rightarrow\; t^+, t^- \in [0,1].$$

The point $(u_{1,2}^{a,a}, v_{1,2}^{a,a})^\top = (\overline{u}_{0,2}^{a,a}, \overline{v}_{0,2}^{a,a})^\top = (-0.639, 1.298)^\top$ does not lie either in Υ_1 or in Υ_2 and therefore TC-interpolants are without a loop, see Fig. 5.14 (left) or analogously we arrive to

$$t^+ = 0.092, t^- = 1.545 \;\Rightarrow\; t^+ \in [0,1], t^- \notin [0,1]$$

for the first TC-interpolant and

$$t^+ = -0.711, t^- = 1.064 \;\Rightarrow\; t^+, t^- \notin [0,1]$$

for the second TC-interpolant.

2. In the second case we use the value $(u_0^a, v_0^a)^\top$ again, but instead $(\overline{u}_1^a, \overline{v}_1^a)^\top$ we use $(\overline{u}_1^b, \overline{v}_1^b)^\top$. This value create new domain Υ_1 bounded by lines and circles see Fig. 5.14 (right).

   Checking where the points $(u_{1,1}^{a,b}, v_{1,1}^{a,b})^\top = (\overline{u}_{0,1}^{a,b}, \overline{v}_{0,1}^{a,b})^\top$ and $(u_{1,2}^{a,b}, v_{1,2}^{a,b})^\top = (\overline{u}_{0,2}^{a,b}, \overline{v}_{0,2}^{a,b})^\top$ lie, we decide about a loop on TC-interpolants.

   The point $(u_{1,1}^{a,b}, v_{1,1}^{a,b})^\top = (\overline{u}_{0,1}^{a,b}, \overline{v}_{0,1}^{a,b})^\top = (-1.719, 1.403)^\top$ lies only in the domain Υ_0 and therefore second TC-interpolant has a loop. Or by the substitution to the equations (5.16) and (5.17) we obtain parameter values of the first TC-interpolant

   $$t^+ = 2.098, t^- = -0.609 \;\Rightarrow\; t^+, t^- \notin [0,1]$$

   and values

   $$t^+ = 0.876, t^- = 0.028 \;\Rightarrow\; t^+, t^- \in [0,1],$$

   which belong to the second TC-interpolant.

   Finally, the point $(u_{1,2}^{a,b}, v_{1,2}^{a,b})^\top = (\overline{u}_{0,2}^{a,b}, \overline{v}_{0,2}^{a,b})^\top = (1.805, -1.143)^\top$ does not lie neither in domains Υ_1 or Υ_0 which causes no loop on both TC-interpolants. This we can verify by checking the parameter values of the first TC-interpolant

   $$t^+ = 0.161, t^- = 1.033 \;\Rightarrow\; t^+ \in [0,1], t^- \notin [0,1]$$

   and the second TC-interpolant

   $$t^+ = -1.742, t^- = 2.136 \;\Rightarrow\; t^+, t^- \notin [0,1].$$ ♦

Remark 21. *It is obvious that we can discuss only a local self-intersection, i.e. loop. Otherwise for global self-intersection we have to use some techniques, see [2], [107], where several approaches are introduced.*

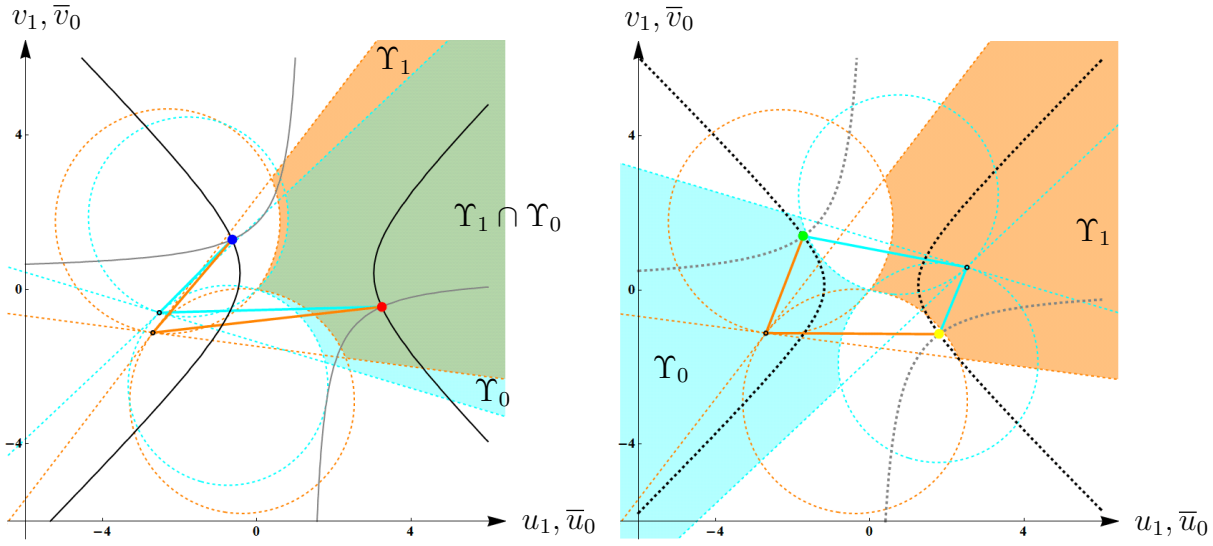


Figure 5.14: The domains Υ_1 (orange) and Υ_0 (cyan) and the preimages of the first (orange line segment) and the second (cyan line segment) TC-interpolants. Left: Points $(u_{1,1}^{\mathrm{a,a}}, v_{1,1}^{\mathrm{a,a}})^\top$ (blue point) and $(u_{1,2}^{\mathrm{a,a}}, v_{1,2}^{\mathrm{a,a}})^\top$ (red point); Right: $(u_{1,1}^{\mathrm{a,b}}, v_{1,1}^{\mathrm{a,b}})^\top$ (green point) and $(u_{1,2}^{\mathrm{a,b}}, v_{1,2}^{\mathrm{a,b}})^\top$ (yellow point).

5.4 C^1 TC-spline

In this section, we conclude Chapter 5 by setting a criteria on initial data to obtain a "nice" interpolation or, equivalently, to achieve good approximation order. First, we deal only with Hermite data, where we show the criteria, how to get good TC-interpolants. Further, we extend the observation for any data and create C^1 TC-spline.

Conjecture 22. *Let us suppose C^1 Hermite data $\mathbf{A}, \mathbf{B}$ with associated tangent vectors $\mathbf{t_A}, \mathbf{t_B}$. If the data fulfill the condition*

$$|\mathbf{t_A}| \leq \frac{\sqrt{3}}{2}|\mathbf{B}-\mathbf{A}| \ \wedge \ |\mathbf{t_B}| \leq \frac{\sqrt{3}}{2}|\mathbf{B}-\mathbf{A}| \tag{5.23}$$

we obtain at least one pair of TC-interpolants without a local and global self-intersections.

The condition (5.23) is estimated from several observation which have been done and we mention only the lower bound for which good TC-interpolants undoubtedly exist. The criterium (5.23) depends on the initial data, especially on the orientation of associated tangent vectors, see Fig. 5.15.

Let us point out the difficulties of the 'proof' of Conjecture 22. In Section 5.2 the construction of C^1 TC-interpolants was done, but we do not discuss which data possess a local (or a global) self-intersection. From the equation (5.16), (5.17) we obtain a subdomain where a point $(u_1, v_1)^\top = (\overline{u}_0, \overline{v}_0)^\top$ has to lie to get TC-interpolants without local self-intersections. The problem occur when we want to describe the point $(u_1, v_1)^\top = (\overline{u}_0, \overline{v}_0)^\top$, which is the intersection of two hyperbolas. Expressing the formula of intersection of two hyperbolas described in Equation (5.13) is a computational problem. It is impossible to decide exactly which input data offer TC-interpolants without loops, since we do not know almost anything about the intersection of hyperbolas.

Let us suppose a sequence of points $\mathbf{P}_0, \ldots, \mathbf{P}_n, n \in \mathbb{N}$, and tangent vectors at the first and last points, i.e., $\mathbf{t}_0, \mathbf{t}_n$. Using previous observation (5.23) we derive two important statements how to

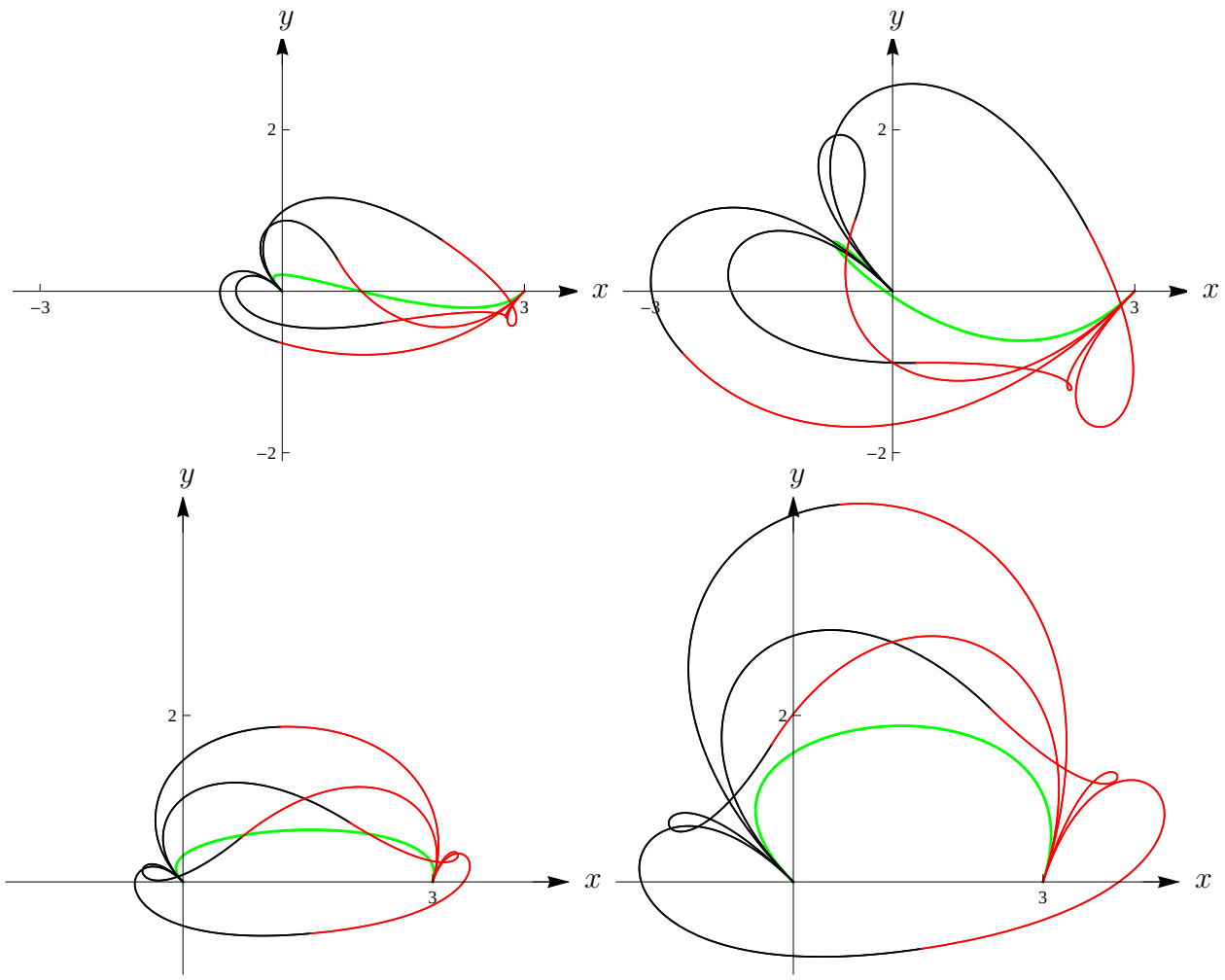


Figure 5.15: All TC-interpolants for the same data $\mathbf{P}_i = (0,0)^\top, \mathbf{P}_{i+1} = (3,0)^\top$ where only tangent vectors $\mathbf{t}_i, \mathbf{t}_{i+1}$ are changed. Also plotted Ferguson cubic (green) for given data. Top left: $\mathbf{t}_i = (-3,3)/\sqrt{2}, \mathbf{t}_{i+1} = (3,3)/\sqrt{2}$; Top right: $\mathbf{t}_i = (-9,9)/\sqrt{2}, \mathbf{t}_{i+1} = (9,9)/\sqrt{2}$; Bottom left: $\mathbf{t}_i = (-3,3)/\sqrt{2}, \mathbf{t}_{i+1} = (-2,-4)/\sqrt{2}$; Bottom right: $\mathbf{t}_i = (-5,5)/\sqrt{2}, \mathbf{t}_{i+1} = (-4,-6)/\sqrt{2}$

obtain a good interpolation by TC-interpolants. The first one is to compute the directions of tangent vectors $\mathbf{t}_i, i = 1, \ldots, n-1$ at corresponding points $\mathbf{P}_i$ as follows

$$\mathbf{t}_i = \frac{\mathbf{P}_i - \mathbf{P}_{i-1}}{||\mathbf{P}_i - \mathbf{P}_{i-1}||} + \frac{\mathbf{P}_{i+1} - \mathbf{P}_i}{||\mathbf{P}_{i+1} - \mathbf{P}_i||}. \tag{5.24}$$

The second one is to control a length of $\mathbf{t}_i$

$$|\mathbf{t}_i| = \min\left\{\frac{\sqrt{3}}{2}|\mathbf{P}_i - \mathbf{P}_{i+1}|, \frac{\sqrt{3}}{2}|\mathbf{P}_i - \mathbf{P}_{i-1}|\right\}. \tag{5.25}$$

Remark 23. *To find only the direction of tangent vectors $\mathbf{t}_i$ we can use different approaches, e.g. to use cubic spline with its tangent vectors. Then we have to use condition* (5.25) *for a length, which ensure a good pair of TC-interpolant.*

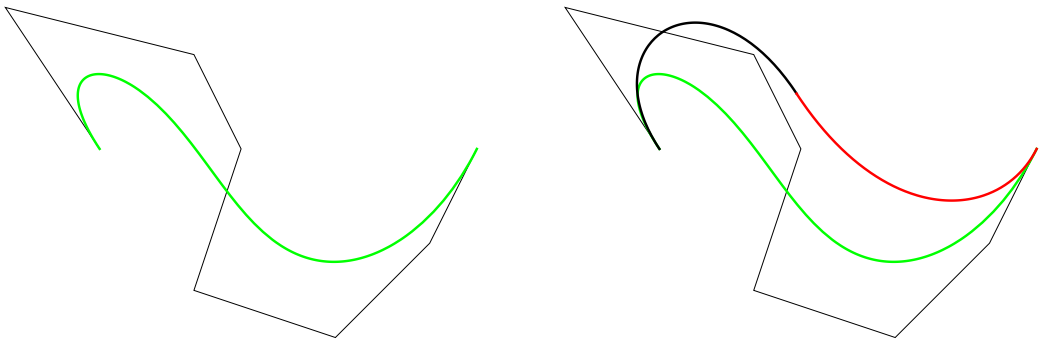


Figure 5.16: Left: Bézier curve of degree 7. Right: Approximation by C^1 TC-interpolants using only end points with associated tangent vectors.

There are two branches, where the construction of TC-interpolants suits. The first one was described in previous section, i.e., C^1 Hermite interpolation. The second one is to approximate given curve.

In general, when we want to approximate given curve by C^1 TC-interpolants then we have to subdivide[5] a curve to several segments, which we approximate separately. Each segment is determined by points (which are computed from given curve) and tangent vectors, which direction and length should be suitable. The direction of tangent vectors can be computed either according to the formula (5.24) or taking the same direction of tangent vector of the given curve. Let us emphasize that for small number of segments the direction of tangent vector differs significantly, but by the increasing of the number of segments both approaches gives quite similar directions. Finally, using the statement in equation (5.25) we obtain acceptable tangent vectors.

To compute approximation order we recall Hausdorff distance, which is defined as

$$d_H(X,Y) = \max\{\sup_{x\in X}\inf_{y\in Y} d(x,y), \sup_{y\in Y}\inf_{x\in X} d(x,y)\},$$

where X, Y are subsets of a metric space and $d(\cdot,\cdot)$ is classical Euclidean distance. The distance can be estimated via sampling points on the curves and evaluating their distances, which is not easy in general, since the distance depends on the number of sampling point. In Figures 5.16, 5.17, 5.18 are shown up to sixteen approximating pairs.

[5]We can subdivide the curve differently, but we use the number of parts from the formula $2^n, n \in \mathbb{N}$.

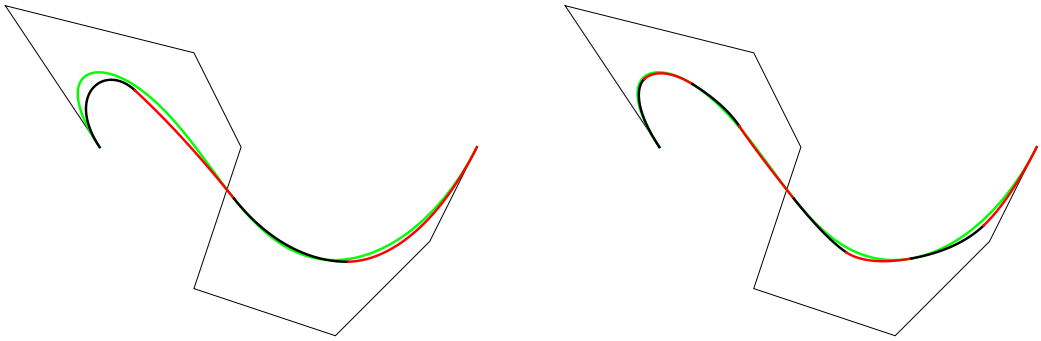


Figure 5.17: Approximation of given Bézier curve (green) by C^1 TC-interpolants. Left: Two pairs of C^1 TC-interpolants; Right: Four pairs of C^1 TC-interpolants.

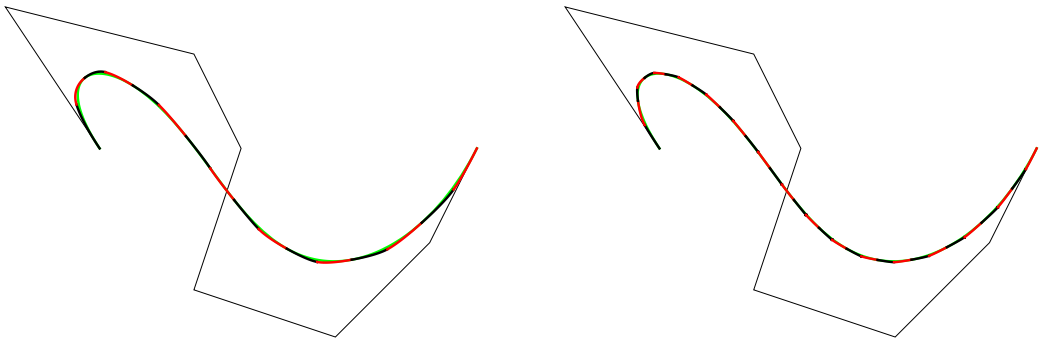


Figure 5.18: Approximation of given Bézier curve (green) by C^1 TC-interpolants. Left: Eight pairs of C^1 TC-interpolants; Right: Sixteen pairs of C^1 TC-interpolants.

Part III

SURFACES

6

Interpolating Bubble patches on quadrilateral meshes

The idea of this chapter is to introduce brand new technique for data interpolation. We consider quadrilateral mesh with unit normals at each vertex. Such data are usually called PN (**P**oint and **N**ormal) data.

6.1 Motivation

The construction of smooth surfaces from given data, such as triangular or quadrilateral meshes, is an important problem in Computer Aided Geometric Design (cf. [27, 54]) and it is of substantial interest in the design process of geometric objects in industrial applications. In the literature, a large number of different methods for generating parametric representations of smooth surfaces can be found. Two main approaches for constructing G^n-surfaces are the manifold construction and the multi-patch scheme. Many related references for these two design methods can be found in [47, 120].

The fundamental idea of the manifold approach is to define the surface with the help of overlapping charts and transition functions, which possess the same order of smoothness as the desired surface. A first constructive manifold construction was given by Grimm and Hughes in [45]. In [120], Ying and Zorin presented a method for creating manifold surfaces from quadrilateral meshes, which was based on the manifold approach from [45]. Further examples of manifold-type constructions can be found in [17, 20, 21, 43, 44, 121].

In the multi-patch approach, surfaces with the appropriate geometric continuity are built by joining several polynomial or rational surface patches together. A survey of this concept can be found in [92]. In general, multi-patch schemes generate smooth surfaces of relatively low degree, but typically require some additional consideration at extraordinary points. For instance, a construction method for curvature continuous free-form surfaces of degree $(3, 5)$ is explained in [91]. It can be generalized to a G^n-construction of degree $(n + 1, d + 2n - 2)$, where d is a flexibility parameter at extraordinary points. In [47], a G^1 interpolation scheme for quadrilateral meshes with vertices of arbitrary valency is presented, where the resulting surfaces are piecewise bicubic. The same technique is used for triangular meshes in [46]. Further examples of multi-patch methods can be found in [60, 74, 76, 101, 102, 103].

Adapted from the multi-patch approach, several methods for constructing smooth surfaces, interpolating a given rectangular curve mesh, have been developed (cf. [49, 73, 93, 119]). In [73, 93], there are described different design schemes for constructing interpolating G^1-surfaces. In [49] and [119], local construction methods for generating G^2-surfaces are explained, which interpolate a mesh of given quintic curves. The construction in [49] is based on Gregory patches, whereas the method in [119] uses standard polynomial surfaces patches which have to fulfill certain compatibility conditions. In general, networks of curves have to satisfy specific algebraic conditions to be suitable for G^1 and G^2-interpolation (cf. [50, 51]).

We describe a new multi-patch scheme for generating a G^n-surface, interpolating the vertices and normals of a quadrilateral mesh. For representing the single surface patches, we introduce a new concept, called bubble patches. In addition, our local construction scheme is based on Gordon-Coons interpolation (cf. [27, 54]) and each surface patch is constructed in such a way that the several patches are connected with G^n-continuity. In the case of G^0, G^1 and G^2-surfaces our algorithm is explained in detail.

The advantages of our method are numerous. Our construction also works for irregular quadrilateral meshes, i.e. meshes with vertices of arbitrary valency, and provides a uniform approach to all valencies. The resulting patches are rational surfaces with arbitrary smoothness, which are joined together with G^n-continuity. We can use low degree polynomial approximations to reduce the obtained degree of the rational surfaces. Furthermore, the construction is local and simple. We have only to solve small system of linear equations.

6.2 Bubble patches

We consider a quadrilateral mesh $\mathcal{M}$ given by vertices $\mathbf{V} \in \mathcal{V}$ and edges $\mathbf{e} = (\mathbf{V}, \mathbf{W}) \in \mathcal{E}$ with $\mathbf{V}, \mathbf{W} \in \mathcal{V}$, where $\mathcal{V}$ is the vertex set and $\mathcal{E}$ is the edge set of the mesh. For a vertex $\mathbf{V} \in \mathcal{V}$ we have an associated implicitly defined algebraic surface

$$F_{\mathbf{V}} = \{\mathbf{Z} \in \mathbb{R}^3 : f_{\mathbf{V}}(\mathbf{Z}) = 0\}$$

of degree m, given by its truncated Taylor expansion

$$f_{\mathbf{V}}(\mathbf{Z}) = \mathbf{g}(\mathbf{V})^\top \cdot (\mathbf{Z} - \mathbf{V}) + \frac{1}{2}(\mathbf{Z} - \mathbf{V})^\top \cdot \mathbb{H}(\mathbf{V}) \cdot (\mathbf{Z} - \mathbf{V}) + \dots \tag{6.1}$$

where $\mathbf{g}(\mathbf{V})$ is the gradient and $\mathbb{H}(\mathbf{V})$ is the Hessian matrix of the function $f_{\mathbf{V}}$ at the vertex $\mathbf{V}$. Conceptually, we consider $f_{\mathbf{V}}$ to be the Taylor expansion of a globally implicitly defined algebraic surface about the vertex $\mathbf{V}$. Therefore we can refer to $\mathbf{g}(\mathbf{V})$ and $\mathbb{H}(\mathbf{V})$ as local gradient and local Hessian matrix at $\mathbf{V}$, respectively.

Remark 24. *A typical input is a mesh, where each vertex $\mathbf{V}$ has an associated normal $\mathbf{n}_{\mathbf{V}}$. In this situation, the function $f_{\mathbf{V}}$ can be estimated as follows. We generate for $f_{\mathbf{V}}$ a function of degree m*

$$f_{\mathbf{V}}(x, y, z) = \sum_{r,s,t \in \mathbb{N}_0:\ r+s+t \le m} c_{r,s,t} x^r y^s z^t, \quad c_{r,s,t} \in \mathbb{R}$$

satisfying

$$f_{\mathbf{V}}(\mathbf{V}) = 0 \tag{6.2}$$

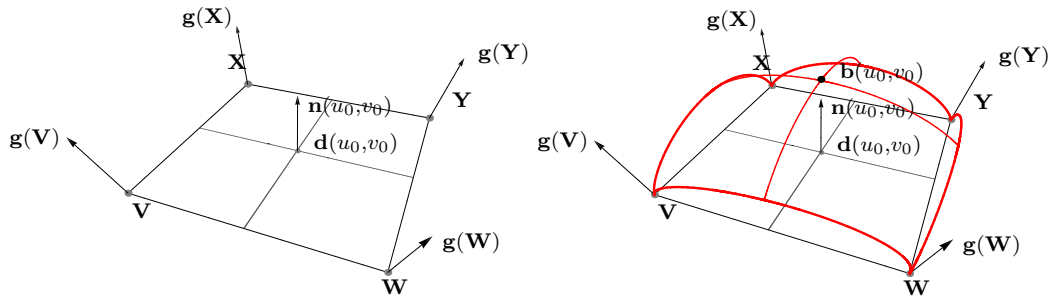


Figure 6.1: Left: A quadrilateral with given vertices and normals with the point $\mathbf{d}(u_0, v_0)$ and the normal $\mathbf{n}(u_0, v_0)$; Right: The boundary curves and two additional parameter lines of a bubble patch on a quadrilateral with the points $\mathbf{d}(u_0, v_0)$ and $\mathbf{b}(u_0, v_0)$ and the normal $\mathbf{n}(u_0, v_0)$.

and

$$(\bigtriangledown f_{\mathbf{V}})(\mathbf{V}) = \mathbf{g}(\mathbf{V}) = \mathbf{n}_{\mathbf{V}}. \tag{6.3}$$

Now we compute the unknown coefficients $c_{r,s,t}$ by solving the minimization problem

$$\min_{c_{r,s,t}} \sum_{\mathbf{W}\in\Omega^i(\mathbf{V})} \omega_{\mathbf{W}}(f_{\mathbf{V}}(\mathbf{W})^2 + ||(\bigtriangledown f_{\mathbf{V}})(\mathbf{W}) - \mathbf{n}_{\mathbf{W}}||^2) \tag{6.4}$$

subject to the constraints (6.2) and (6.3), where $\Omega^i(\mathbf{V})$ is the i-ring neighborhood of vertices of $\mathbf{V}$ and $\omega_{\mathbf{W}}$ is the weight for the vertex $\mathbf{W}$ in the i-ring neighborhood $\Omega^i(\mathbf{W})$.

Definition 25. *Let $\mathcal{Q}$ be a quadrilateral of $\mathcal{M}$ with the vertices $\mathbf{V}, \mathbf{W}, \mathbf{X}, \mathbf{Y} \in \mathcal{V}$, connected by the edges $(\mathbf{V}, \mathbf{W}), (\mathbf{X}, \mathbf{Y}), (\mathbf{V}, \mathbf{X}), (\mathbf{W}, \mathbf{Y}) \in \mathcal{E}$, and the corresponding normals $\mathbf{g}(\mathbf{V})$, $\mathbf{g}(\mathbf{W})$, $\mathbf{g}(\mathbf{X})$, $\mathbf{g}(\mathbf{Y})$, see Fig. 6.1 (left). On the quadrilateral $\mathcal{Q}$, we define a surface $\mathbf{b} : [0,1]^2 \to \mathbb{R}^3$ as follows*

$$\mathbf{b}(u, v) = \mathbf{d}(u, v) + h(u, v)\mathbf{n}(u, v), \quad (u, v) \in [0, 1]^2, \tag{6.5}$$

where $\mathbf{d}$ and $\mathbf{n}$ are bilinear interpolants, given by

$$\mathbf{d}(u, v) = (1 - v)((1 - u)\mathbf{V} + u\mathbf{W}) + v((1 - u)\mathbf{X} + u\mathbf{Y})$$

and

$$\mathbf{n}(u, v) = (1 - v)((1 - u)\mathbf{g}(\mathbf{V}) + u\mathbf{g}(\mathbf{W})) + v((1 - u)\mathbf{g}(\mathbf{X}) + u\mathbf{g}(\mathbf{Y})),$$

and h is a scalar function. Moreover, we require that $\mathbf{b}(0, 0) = \mathbf{V}$, $\mathbf{b}(1, 0) = \mathbf{W}$, $\mathbf{b}(0, 1) = \mathbf{X}$ and $\mathbf{b}(1, 1) = \mathbf{Y}$ which implies that $h(0, 0) = h(1, 0) = h(0, 1) = h(1, 1) = 0$. The function $\mathbf{b}$ is referred to as a bubble patch *and the function h is called the* bubble function.

An example of a bubble patch on a quadrilateral is presented in Figure 6.1 (right).

For later reference, let $n \in \mathbb{N}_0$, $k \in \{0, 1\}$ and $j \in \{0, \dots n\}$, then we denote by $H_{k,j}^{2n+1}$ the classical Hermite polynomials of degree $2n + 1$, i.e.

$$H_{k,j}^{2n+1} : [0, 1] \to \mathbb{R}$$

with

$$\frac{\partial^i}{\partial t^i}(H_{k,j}^{2n+1}(t)) \Big|_{t=l} = \delta_{i,j}\delta_{k,l}$$

for $j \in \{0, \dots, n\}$ and $l \in \{0, 1\}$.

6.3 Construction of G^n-surfaces

We describe a method for constructing a G^n-surface, consisting of bubble patches, that interpolates the vertices and normals of a quadrilateral mesh with normals. For each quadrilateral we generate a surface, which is described by the bubble patch introduced in (6.5), in such a way that the surface patches are pieced together with G^n-continuity. The main idea is to construct the bubble function h with the help of Gordon-Coons interpolation by using boundary data for h which provide the desired continuity between the patches. Our construction consists of the following two steps.

Step 1 We make a compatible mesh of G^n-surface strips between the vertices of the quadrilateral mesh $\mathcal{M}$ which are connected by edges. A G^n-surface strip can be seen as an equivalence class of all surfaces through a curve between two vertices having contact of order n along this curve. The G^n-surface strips for $n \leq 2$ can be described as follows.

- A G^0-surface strip is simply the boundary curve between two vertices which are connected by an edge. The corresponding boundary curves have to meet at the vertices.
- A G^1-surface strip is the boundary curve with the associated tangent planes between two vertices which are connected by an edge. It can be seen as an equivalence class of all surfaces through the boundary curve which have the same tangent planes along the boundary curve.
- A G^2-surface strip is the boundary curve with the associated tangent planes and normal curvatures between two vertices which are connected by an edge. It can be seen as an equivalence class of all surfaces through the boundary curve which have the same tangent planes and the same normal curvatures along the boundary curve.

In Figure 6.2, the G^0, G^1 and G^2-surface strips between two vertices of a quadrilateral are visualized. The construction of the G^n-surface strip for $n \in \{0, 1, 2\}$ is explained in detail in Section 6.4.

Step 2 We generate the bubble functions by using Gordon-Coons interpolation. Given a quadrilateral $\mathcal{Q}$ on the mesh $\mathcal{M}$ with the associated G^n surface strip, we evaluate the boundary values and the cross boundary derivatives of the bubble function. For this we use the fact that the desired bubble patch need to have a contact of order n with the G^n-surface strip along the boundary curve. Furthermore the obtained boundary data of the bubble function has to satisfy the so-called twist compatibility condition (6.25) at the vertices of the patch, which is fulfilled by our choice of the G^n-surface strip. Then we can use Gordon-Coons interpolation to construct a suitable bubble function. This step is explained in more detail in Section 6.5.

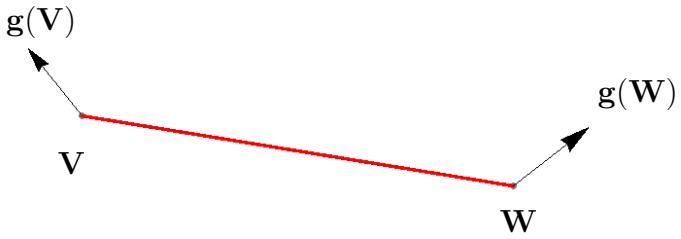
6.4 G^n-surface strips

We explain the construction of a G^n-surface strip. For this we consider two vertices $\mathbf{V}, \mathbf{W} \in \mathcal{V}$ which are connected by an edge. At first we construct a boundary curve $\mathbf{p} : [0, 1] \to \mathbb{R}^3$, given by

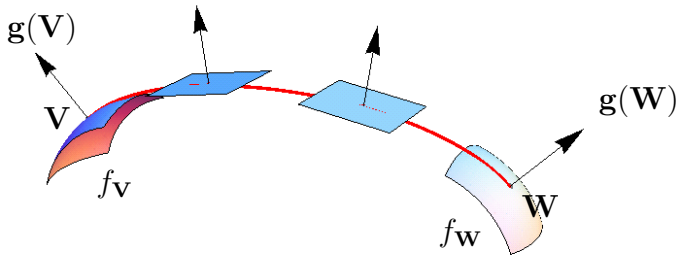
$$\mathbf{p}(t) = (1-t)\mathbf{V} + t\mathbf{W} + \hat{h}(t)((1-t)\mathbf{g}(\mathbf{V}) + t\mathbf{g}(\mathbf{W})), \tag{6.6}$$

where $\hat{h}$ is the restriction of the bubble function h on the boundary between the vertices $\mathbf{V}$ and $\mathbf{W}$. Then we generate a family of implicitly defined algebraic surfaces

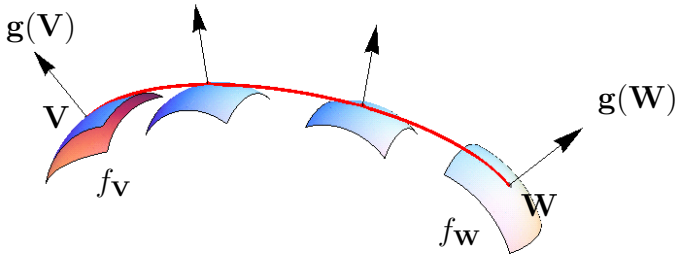
$$F_{\mathbf{Q}} = \{\mathbf{Z} \in \mathbb{R}^3 : f_{\mathbf{Q}}(\mathbf{Z}) = 0\}$$



G^0-surface strip



G^1-surface strip



G^2-surface strip

Figure 6.2: The G^0, G^1 and G^2-surface strips between two vertices of a quadrilateral.

along the boundary curve $\mathbf{p}$. For a point $\mathbf{Q}$ on the boundary curve, the function $f_{\mathbf{Q}}$ is given by

$$f_{\mathbf{Q}}(\mathbf{Z}) = \underbrace{\mathbf{g}(\mathbf{Q})^\top \cdot (\mathbf{Z}-\mathbf{Q})}_{\text{only for } n\geq 1} + \underbrace{\frac{1}{2}(\mathbf{Z}-\mathbf{Q})^\top \cdot \mathbb{H}(\mathbf{Q}) \cdot (\mathbf{Z}-\mathbf{Q})}_{\text{only for } n\geq 2} + \ldots, \tag{6.7}$$

where $\mathbf{g}(\mathbf{Q})$ is the local gradient and $\mathbb{H}(\mathbf{Q})$ is the local Hessian matrix of the function $f_{\mathbf{Q}}$ at the point $\mathbf{Q}$. At the vertices on the boundary curve $\mathbf{p}$, the implicitly defined algebraic surfaces introduced in (6.1) and (6.7) agree up to the term of degree n. The family of implicitly defined algebraic surfaces $F_{\mathbf{q}}$ along the boundary curve $\mathbf{p}$ describes the compatible G^n-surface strip between the vertices $\mathbf{V}$ and $\mathbf{W}$. In the following subsections we explain the construction of the boundary curves and of the implicitly defined algebraic surfaces for different values of the smoothness parameter n. The G^n-surface strips have to be generated in such a way that the boundary values and the cross boundary derivatives of the bubble function h, evaluated in Section 6.5, fulfill the so-called twist compatibility condition (6.25) at the vertices of the patch. This is necessary for applying Gordon-Coons interpolation to obtain the bubble function h, see Section 6.5, and will be achieved by having a higher order contact of the G^n-surface strips with the implicitly defined algebraic surfaces at the vertices.

6.4.1 G^0-surface strip

This case is trivial. $\hat{h}$ is any function with $\hat{h}(0) = \hat{h}(1) = 0$ and we simply choose $\hat{h} = 0$. The family of implicitly defined algebraic surfaces is not required (formally $f_{\mathbf{Q}} = 0$).

6.4.2 G^1-surface strip

We explain the generation of a G^1-surface strip which will be used in Section 6.5 to construct bubble patches which are joined together with G^1-continuity. This will be achieved by generating bubble patches with a first order contact with the G^1-surface strip along the boundary. In this subsection we explain the construction of a compatible G^1-surface strip which is given by the functions $f_{\mathbf{Q}}$, defined up to linear terms, along the boundary curve $\mathbf{p}$.

The first step is the construction of the function $\hat{h}$, specifying the boundary curve $\mathbf{p}$. It is chosen as a quintic polynomial in Bernstein-Bézier representation, i.e.

$$\hat{h}(t) = \sum_{i=0}^{5} c_i B_i^5(t), \tag{6.8}$$

where B_i^5 are the Bernstein polynomials of degree 5 and $c_i \in \mathbb{R}$. To get the function $\hat{h}$, we compute a boundary curve $\mathbf{p}$ which has a contact of second order with the implicitly defined algebraic surfaces $F_{\mathbf{V}}$ and $F_{\mathbf{W}}$ at the vertices $\mathbf{V}$ and $\mathbf{W}$, respectively. That means the boundary curve $\mathbf{p}$ has to satisfy the following conditions

$$\left(\frac{\partial^i}{\partial t^i} f_{\mathbf{V}}(\mathbf{p}(t))\right)\Big|_{t=0} = 0 \text{ and } \left(\frac{\partial^i}{\partial t^i} f_{\mathbf{W}}(\mathbf{p}(t))\right)\Big|_{t=1} = 0 \tag{6.9}$$

for $i \in \{0,1,2\}$. These conditions are a system of linear equations for the coefficients c_i of the function $\hat{h}$. The second order contact of the boundary curve with the implicitly defined algebraic surfaces at the vertices ensures that the twist compatibility condition (6.25) is fulfilled, see Lemma 29 below.

The second step is the generation of a family of implicitly defined algebraic surfaces $F_{\mathbf{p}}$ along the boundary curve $\mathbf{p}$, for which the functions $f_{\mathbf{p}}$ are needed to be defined up to linear terms. For this we have to construct local gradients $\mathbf{g}(\mathbf{p})$ which are compatible with the boundary curve. This compatibility condition is described in the following lemma.

Lemma 26. *Let $\mathbf{p}$ be the boundary curve in (6.8) and $f_{\mathbf{p}}$ be the functions in (6.7), defined up to linear terms. Then the local gradients $\mathbf{g}(\mathbf{p})$ are compatible with the boundary curve $\mathbf{p}$ if*

$$\mathbf{g}(\mathbf{p}(t))^{\top} \cdot \frac{\partial}{\partial t}\mathbf{p}(t) = 0. \tag{6.10}$$

Proof. To get a well-defined G^1-surface strip, the boundary curve $\mathbf{p}$ needs to have a first order contact with the functions $f_{\mathbf{p}}$ along the boundary, i.e.

$$(\frac{\partial}{\partial s} f_{\mathbf{p}(t)}(\mathbf{p}(s))) \,\big|_{s=t} = 0$$

for $t \in [0,1]$, which is equivalent to equation (6.10) and provides us the compatibility condition for $\mathbf{g}(\mathbf{p})$. □

We construct the local gradients $\mathbf{g}(\mathbf{p})$ as follows. At first we consider the vector fields $\hat{\mathbf{g}}(\mathbf{p})$ along the boundary curve $\mathbf{p}$ which are obtained from

$$\hat{\mathbf{g}}(\mathbf{p}(t)) = \sum_{j=0}^{1} H^3_{0,j}\hat{\mathbf{g}}^j_{\mathbf{V}} + H^3_{1,j}\hat{\mathbf{g}}^j_{\mathbf{W}}, \tag{6.11}$$

where

$$\hat{\mathbf{g}}^j_{\mathbf{V}} = \frac{\partial^j}{\partial t^j}(\bigtriangledown f_{\mathbf{V}})(\mathbf{p}(t)) \,\Big|_{t=0} \text{ and } \hat{\mathbf{g}}^j_{\mathbf{W}} = \frac{\partial^j}{\partial t^j}(\bigtriangledown f_{\mathbf{W}})(\mathbf{p}(t)) \,\Big|_{t=1},$$

for $j \in \{0,1\}$. In particular we have

$$\hat{\mathbf{g}}^0_{\mathbf{V}} = \mathbf{g}(\mathbf{V}) \text{ and } \hat{\mathbf{g}}^0_{\mathbf{W}} = \mathbf{g}(\mathbf{W}).$$

The resulting vector fields $\hat{\mathbf{g}}(\mathbf{p})$ ensure that the twist compatibility condition (6.25) is satisfied, see Lemma 29.

In general, the vector fields $\hat{\mathbf{g}}(\mathbf{p})$ do not satisfy Lemma 26. Therefore we construct suitable gradients $\mathbf{g}(\mathbf{p})$ by solving the minimization problem

$$\mathbf{g}(\mathbf{p}(t)) = \arg\min_{\bar{\mathbf{g}}} ||\bar{\mathbf{g}} - \hat{\mathbf{g}}(\mathbf{p}(t))||^2 \tag{6.12}$$

subject to the constraint (6.10). Its solution is given by

$$\mathbf{g}(\mathbf{p}(t)) = \hat{\mathbf{g}}(\mathbf{p}(t)) - \frac{\hat{\mathbf{g}}(\mathbf{p}(t))^{\top} \cdot \frac{\partial}{\partial t}\mathbf{p}(t)}{\frac{\partial}{\partial t}(\mathbf{p}(t))^{\top} \cdot \frac{\partial}{\partial t}\mathbf{p}(t)}(\frac{\partial}{\partial t}(\mathbf{p}(t))). \tag{6.13}$$

This possesses a simple geometric interpretation. The gradients $\mathbf{g}(\mathbf{p})$ are obtained as the projections of the gradients $\hat{\mathbf{g}}(\mathbf{p})$ into the normal plane of the curve $\mathbf{p}$. In addition, the projection preserves the twist compatibility condition (6.25).

6.4.3 G^2-surface strip

We explain the construction of a G^2-surface strip, which is given by the functions $f_{\mathbf{Q}}$, defined up to quadratic terms, along the boundary curve $\mathbf{p}$. In Section 6.5, we will use this strip to generate bubble patches which have a second order contact with the strip to get G^2-continuity between the neighboring patches.

We start with the construction of the function $\hat{h}$, for which we choose a polynomial in Bernstein-Bézier representation of degree 9, i.e.

$$\hat{h}(t) = \sum_{i=0}^{9} c_i B_i^9(t),$$

where B_i^9 are the Bernstein polynomials of degree 9 and $c_i \in \mathbb{R}$. The coefficients are obtained by computing a boundary curve $\mathbf{p}$ which possesses a contact of order 4 with the implicitly defined algebraic surface $F_{\mathbf{V}}$ and $F_{\mathbf{W}}$ at the vertices $\mathbf{V}$ and $\mathbf{W}$,respectively, i.e.

$$(\frac{\partial^i}{\partial t^i} f_{\mathbf{V}}(\mathbf{p}(t)) \,\big|_{t=0} = 0 \text{ and } (\frac{\partial^i}{\partial t^i} f_{\mathbf{W}}(\mathbf{p}(t)) \,\big|_{t=1} = 0 \tag{6.14}$$

for $i \in \{0, \ldots, 4\}$. These conditions lead to a system of linear equations for the coefficients c_i. The contact of order 4 of the boundary curve with the implicitly defined algebraic surfaces at the vertices guarantees that the twist compatibility condition (6.25) is fulfilled, see Lemma 30 below.

Next we generate a family of implicitly defined algebraic surfaces $F_{\mathbf{p}}$ along the boundary curve $\mathbf{p}$, for which the functions $f_{\mathbf{p}}$ are needed to be defined up to quadratic terms. That means we construct local gradients $\mathbf{g}(\mathbf{p})$ and local Hessian matrices $\mathbb{H}(\mathbf{p})$ which have to be compatible with the boundary curve $\mathbf{p}$. The following lemma describes this compatibility condition.

Lemma 27. *Let $\mathbf{p}$ be the boundary curve in (6.8) and $f_{\mathbf{p}}$ be the functions in (6.7), defined up to quadratic terms. Then the local gradients $\mathbf{g}(\mathbf{p})$ and the local Hessian matrices $\mathbb{H}(\mathbf{p})$ are compatible with the boundary curve $\mathbf{p}$ if condition (6.10) and the condition*

$$\frac{\partial}{\partial t}\mathbf{g}(\mathbf{p}(t)) = \mathbb{H}(\mathbf{p}(t)) \cdot \frac{\partial}{\partial t}\mathbf{p}(t) \tag{6.15}$$

are satisfied.

Proof. The boundary curve $\mathbf{p}$ needs to have a first and second order contact with the functions $f_{\mathbf{p}}$ along the boundary, i.e.

$$(\frac{\partial}{\partial s} f_{\mathbf{p}(t)}(\mathbf{p}(s))) \,\big|_{s=t} = 0 \tag{6.16}$$

and

$$(\frac{\partial^2}{\partial s^2} f_{\mathbf{p}(t)}(\mathbf{p}(s))) \,\big|_{s=t} = 0 \tag{6.17}$$

for $t \in [0, 1]$, respectively. In the proof of Lemma 26 we have seen that the conditions (6.10) and (6.16) are equivalent. In addition we obtain by differentiating condition (6.10) the equation

$$\mathbf{g}(\mathbf{p}(t))^\top \cdot \frac{\partial^2}{\partial t^2}\mathbf{p}(t) + \frac{\partial}{\partial t}\mathbf{g}(\mathbf{p}(t))^\top \cdot \frac{\partial}{\partial t}\mathbf{p}(t) = 0.$$

Since condition (6.17) is equivalent to

$$\mathbf{g}(\mathbf{p}(t))^{\top} \cdot \frac{\partial^2}{\partial t^2}\mathbf{p}(t) + \frac{\partial}{\partial t}\mathbf{p}(t) \cdot \mathbb{H}(\mathbf{p}(t)) \cdot \frac{\partial}{\partial t}\mathbf{p}(t) = 0,$$

we obtain the compatibility condition (6.15). □

The construction of the local gradients $\mathbf{g}(\mathbf{p})$ is similar to the case of G^1-surface strips. The only difference is that we use interpolants of degree 7 for the vector fields $\hat{\mathbf{g}}(\mathbf{p})$, instead of cubic interpolants, i.e.

$$\hat{\mathbf{g}}(\mathbf{p}(t)) = \sum_{j=0}^{3} H_{0,j}^7 \hat{\mathbf{g}}_{\mathbf{V}}^j + H_{1,j}^7 \hat{\mathbf{g}}_{\mathbf{W}}^j,$$

where

$$\hat{\mathbf{g}}_{\mathbf{V}}^j = \frac{\partial^j}{\partial t^j}(\bigtriangledown f_{\mathbf{V}})(p(t)) \Big|_{t=0} \text{ and } \hat{\mathbf{g}}_{\mathbf{W}}^j = \frac{\partial^j}{\partial t^j}(\bigtriangledown f_{\mathbf{W}})(p(t)) \Big|_{t=1},$$

for $j \in \{0, \ldots, 3\}$. In particular we obtain

$$\hat{\mathbf{g}}_{\mathbf{V}}^0 = \mathbf{g}(\mathbf{V}) \text{ and } \hat{\mathbf{g}}_{\mathbf{W}}^0 = \mathbf{g}(\mathbf{W}).$$

These interpolants of higher degree ensure that the twist compatibility condition (6.25) is satisfied for G^2-surfaces, which will be presented in Lemma 30.

The local gradients $\mathbf{g}(\mathbf{p})$ are again the solution (6.13) of the minimization problem (6.12) subject to the constraint (6.10) and still preserve the twist compatibility condition (6.25).

The next step is the construction of the local Hessian matrices $\mathbb{H}(\mathbf{p})$. For this we start with the computation of matrices $\hat{\mathbb{H}}(\mathbf{p})$ which are quintic interpolants of the Hessian matrices $\mathbb{H}(\mathbf{v})$ and $\mathbb{H}(\mathbf{W})$, i.e.

$$\hat{\mathbb{H}}(\mathbf{p}(t)) = \sum_{j=0}^{2} H_{0,j}^5 \hat{\mathbb{H}}_{\mathbf{V}}^j + H_{1,j}^5 \hat{\mathbb{H}}_{\mathbf{W}}^j, \tag{6.18}$$

where

$$\hat{\mathbb{H}}_{\mathbf{V}}^j = \frac{\partial^j}{\partial t^j}\text{Hess}(f_{\mathbf{V}})(\mathbf{p}(t)) \Big|_{t=0} \text{ and } \hat{\mathbb{H}}_{\mathbf{W}}^j = \frac{\partial^j}{\partial t^j}\text{Hess}(f_{\mathbf{W}})(\mathbf{p}(t)) \Big|_{t=1},$$

for $j \in \{0, 1, 2\}$ and where $\text{Hess}(\cdot)$ is the Hessian matrix. In particular we have

$$\hat{\mathbb{H}}_{\mathbf{V}}^0 = \mathbb{H}(\mathbf{V}) \text{ and } \hat{\mathbb{H}}_{\mathbf{W}}^0 = \mathbb{H}(\mathbf{W}).$$

The matrices $\hat{\mathbb{H}}(\mathbf{p})$ ensure that the twist compatibilty condition (6.25) is satisfied, see Lemma 30.

But in general, condition (6.15) in Lemma 27 is not fulfilled. Therefore we generate suitable Hessian matrices $\mathbb{H}(\mathbf{p})$ by solving the minimization problem

$$\mathbb{H}(\mathbf{p}(t)) = \arg\min_{\bar{\mathbb{H}}} ||\bar{\mathbb{H}} - \hat{\mathbb{H}}(\mathbf{p}(t))||^2 \tag{6.19}$$

subject to the constraint (6.15).

Lemma 28. *The construction of the Hessian matrices* $\mathbb{H}(\mathbf{p})$ *is invariant with respect to the choice of a coordinate system.*

Proof. Let $\mathbf{p}$ be the boundary curve in (6.8) and $f_{\mathbf{p}}$ be the functions in (6.7), defined up to quadratic terms. Let $\mathbb{U} \in \mathrm{SO}(3)$, then we denote by $\tilde{\mathbf{p}}$ the curve

$$\tilde{\mathbf{p}} = \mathbb{U}^\top \cdot \mathbf{p}$$

and by $\tilde{f}_{\mathbf{p}}$ the function

$$\tilde{f}_{\tilde{\mathbf{p}}}(\mathbf{Z}) = f_{\mathbf{p}}(\mathbb{U} \cdot \mathbf{Z}).$$

The gradient $\tilde{\mathbf{g}}(\tilde{\mathbf{p}})$ and the Hessian matrix $\tilde{\mathbb{H}}(\tilde{\mathbf{p}})$ of $\tilde{f}_{\tilde{\mathbf{p}}}$ are given by

$$\tilde{\mathbf{g}}(\tilde{\mathbf{p}}) = \mathbb{U} \cdot \mathbf{g}(\mathbf{p}) \text{ and } \tilde{\mathbb{H}}(\tilde{\mathbf{p}}) = \mathbb{U} \cdot \mathbb{H}(\mathbf{p}) \cdot \mathbb{U}^\top,$$

where $\mathbf{g}(\mathbf{p})$ and $\mathbb{H}(\mathbf{p})$ are the gradient and the Hessian matrix of $f_{\mathbf{p}}$, respectively. In addition, let $\mathbb{H}(\mathbf{p})$ be the solution of the minimization problem (6.19) subject to the constraint (6.15) and let

$$\tilde{\mathbb{H}}_0(\tilde{\mathbf{p}}) = \mathbb{U} \cdot \hat{\mathbb{H}}(\mathbf{p}) \cdot \mathbb{U}^\top.$$

Now we have to show that $\tilde{\mathbb{H}}(\tilde{\mathbf{p}})$ is the solution of the minimization problem

$$\arg \min_{\tilde{\mathbb{H}}(\tilde{\mathbf{p}}(t))} ||\tilde{\mathbb{H}}(\tilde{\mathbf{p}}(t)) - \tilde{\mathbb{H}}_0(\tilde{\mathbf{p}}(t))||^2$$

subject to the constraint

$$\frac{\partial}{\partial t}\tilde{\mathbf{g}}(\tilde{\mathbf{p}}(t)) = \tilde{\mathbb{H}}(\tilde{\mathbf{p}}(t)) \cdot \frac{\partial}{\partial t}\tilde{\mathbf{p}}(t).$$

Let $||\cdot||$ be the Frobenius norm and $\mathrm{tr}(\cdot)$ be the trace of a matrix. We have

$$\begin{aligned}
&||\tilde{\mathbb{H}}(\tilde{\mathbf{p}}) - \tilde{\mathbb{H}}_0(\tilde{\mathbf{p}})||^2 = \\
&\mathrm{tr}((\mathbb{U} \cdot (\mathbb{H}(\mathbf{p}) - \hat{\mathbb{H}}(\mathbf{p})) \cdot \mathbb{U}^\top) \cdot (\mathbb{U} \cdot (\mathbb{H}(\mathbf{p}) - \hat{\mathbb{H}}(\mathbf{p})) \cdot \mathbb{U}^\top)^\top) = \\
&\mathrm{tr}(\mathbb{U} \cdot (\mathbb{H}(\mathbf{p}) - \hat{\mathbb{H}}(\mathbf{p})) \cdot (\mathbb{H}(\mathbf{p}) - \hat{\mathbb{H}}(\mathbf{p})) \cdot \mathbb{U}^\top) = \\
&\mathrm{tr}((\mathbb{H}(\mathbf{p}) - \hat{\mathbb{H}}(\mathbf{p})) \cdot (\mathbb{H}(\mathbf{p}) - \hat{\mathbb{H}}(\mathbf{p}))) = \\
&||\mathbb{H}(\mathbf{p}) - \hat{\mathbb{H}}(\mathbf{p})||^2.
\end{aligned}$$

In addition, it is easy to show that

$$\frac{\partial}{\partial t}\tilde{\mathbf{g}}(\tilde{\mathbf{p}}(t)) = \tilde{\mathbb{H}}(\tilde{\mathbf{p}}(t)) \cdot \frac{\partial}{\partial t}\tilde{\mathbf{p}}(t)$$

is equivalent to

$$\frac{\partial}{\partial t}\mathbf{g}(\mathbf{p}(t)) = \mathbb{H}(\mathbf{p}(t)) \cdot \frac{\partial}{\partial t}\mathbf{p}(t).$$

□

6.4.4 G^n-surface strip

Finally we give a short overview of the general construction of a G^n-surface strip. At first we construct a function $\hat{h}$ which is given as a polynomial in Bernstein-Bézier representation of degree $4n+1$. The coefficients of the function $\hat{h}$ are obtained by computing a boundary curve $\mathbf{p}$ with a contact of order $2n$ with the implicitly defined algebraic surfaces $F_{\mathbf{V}}$ and $F_{\mathbf{W}}$ at the vertices $\mathbf{V}$ and $\mathbf{W}$, respectively.

Then we generate a family of implicitly defined algebraic surfaces $F_{\mathbf{p}}$, defined up to terms of degree n, such that the functions $f_{\mathbf{p}}$ are compatible with the boundary function $\mathbf{p}$. For this purpose, compatibility conditions can be formulated, which are generalizations of Lemma 27 with additional conditions.

6.5 Construction of bubble functions

We explain the construction of the bubble function h for a bubble patch (6.5) by using Gordon-Coons interpolation. This method is a well-known tool for constructing a bivariate function, in our case h, which interpolates given boundary data (cf. [27, 54]). At first we describe the generation of this boundary data, which will be used for Gordon-Coons interpolation and has to be chosen in such a way that G^n-continuity between the surface patches is guaranteed. For this purpose we construct a bubble patch which has a contact of order n with the G^n-surface strips along the corresponding boundary curves to obtain the desired boundary data. Then we use Gordon-Coons interpolation to construct the bubble function h.

6.5.1 Evaluation of boundary values and cross boundary derivatives

We consider the bubble patch $\mathbf{b}$ in (6.5) with the vertices $\mathbf{V}, \mathbf{W}, \mathbf{X}, \mathbf{Y}$ and the corresponding G^n-surface strip with the boundary curve $\mathbf{p}$ for the edge $(\mathbf{V}, \mathbf{W})$, constructed in Section 6.4. Now we use this surface strip to generate the boundary function $h(u,0)$ and the cross boundary derivatives

$$(\frac{\partial^i}{\partial v^i} h(u,v)) \mid_{v=0} \tag{6.20}$$

for $i \in \{1, \ldots, n\}$, which are needed for Gordon-Coons interpolation in the following subsection. We explain the generation of these functions in detail for the case $n \leq 2$. The idea can be generalized to any $n \geq 3$.

For the generation of the boundary data, we use the fact that the bubble patch $\mathbf{b}$ needs to have a contact of order n with the G^n-surface strip along the boundary curve $\mathbf{p}$. Having this contact of order n along the boundary curve $\mathbf{p}$, we guarantee that the bubble patch $\mathbf{b}$ and its beighboring patch are connected with G^n-continuity. Since the bubble patch $\mathbf{b}$ has a parametric representation and the G^n-surface strip is given implicit, the contact of order n can be described by the simple contact order

$$(\frac{\partial^s}{\partial v^s} f_{\mathbf{p}(u)}(\mathbf{b}(u,v))) \mid_{v=0} = 0 \tag{6.21}$$

for $s \in \{0, \ldots, n\}$. Depending on the order of contact, the conditions (6.21) are used to get the several functions as follows.

- Contact of order 0: We have

  $$f_{\mathbf{p}(u)}(\mathbf{b}(u,0)) = 0,$$

  which implies

  $$\mathbf{b}(u,0) = \mathbf{p}(u).$$

  Therefore we get

  $$h(u,0) = \hat{h}(u)$$

  where $\hat{h}$ is the function in (6.6).

- Contact of order 1: The condition

  $$(\frac{\partial}{\partial v} f_{\mathbf{p}(u)}(\mathbf{b}(u,v))) \mid_{v=0} = 0$$

  has to be fulfilled, which is equivalent to

  $$\mathbf{g}(\mathbf{p}(u))^\top \cdot \mathbf{b}_v(u,0) = 0. \tag{6.22}$$

  Since the first partial derivative $\mathbf{b}_v(u,0)$ is given by

  $$\mathbf{b}_v(u,0) = \mathbf{d}_v(u,0) + h_v(u,0)\mathbf{n}(u,0) + h(u,0)\mathbf{n}_v(u,0),$$

  we obtain the first partial derivative

  $$h_v(u,0) = -\frac{\mathbf{g}(\mathbf{p}(u))^\top \cdot \mathbf{d}_v(u,0) + h(u,0)\mathbf{g}(\mathbf{p}(u))^\top \cdot \mathbf{n}_v(u,0)}{\mathbf{g}(\mathbf{p}(u))^\top \cdot \mathbf{n}(u,0)}.$$

- Contact of order 2: We have to satisfy the following condition

  $$(\frac{\partial^2}{\partial v^2} f_{\mathbf{p}(u)}(\mathbf{b}(u,v))) \mid_{v=0} = 0$$

  which is equivalent to

  $$\mathbf{g}(\mathbf{p}(u))^\top \cdot \mathbf{b}_{vv}(u,0) + \mathbf{b}_v(u,0)^\top \cdot \mathbb{H}(\mathbf{p}(u)) \cdot \mathbf{b}_v(u,0) = 0.$$

  Since the second partial derivative $\mathbf{b}_{vv}(u,0)$ is given by

  $$\mathbf{b}_{vv}(u,0) = h_{vv}(u,0)\mathbf{n}(u,0) + 2h_v(u,0)\mathbf{n}_v(u,0),$$

  we get the second partial derivative $h_{vv}(u,0)$ as follows:

  $$h_{vv}(u,0) = -\frac{\mathbf{b}_v(u,0)^\top \cdot \mathbb{H}(\mathbf{p}(u)) \cdot \mathbf{b}_v(u,0) + 2h_v(u,0)\mathbf{g}(\mathbf{p}(u))^\top \cdot \mathbf{n}_v(u,0)}{\mathbf{g}(\mathbf{p}(u))^\top \cdot \mathbf{n}(u,0)}.$$

- Contact of order $s \geq 3$: We construct the partial derivative $(\frac{\partial^s}{\partial v^s} h(u,v)) \mid_{v=0}$ with the help of the condition

$$(\frac{\partial^s}{\partial v^s} f_{\mathbf{p}(u)}(\mathbf{b}(u,v))) \mid_{v=0} = 0$$

and the already computed functions $(\frac{\partial^i}{\partial v^i} h(u,v)) \mid_{v=0}$ for $i \in \{0,\ldots,s-1\}$.

Analogously, we can compute for $i \in \{0,\ldots,n\}$ the functions

$$(\frac{\partial^i}{\partial v^i} h(u,v)) \mid_{v=1}, (\frac{\partial^i}{\partial u^i} h(u,v)) \mid_{u=0} \text{ and } (\frac{\partial^i}{\partial u^i} h(u,v)) \mid_{u=1} \tag{6.23}$$

with the help of the G^n-surface strips for the remaining edges $(\mathbf{X},\mathbf{Y})$, $(\mathbf{V},\mathbf{X})$ and $(\mathbf{W},\mathbf{Y})$, respectively.

For applying Gordon-Coons interpolation we need the following additional values

$$(\frac{\partial^{i+j}}{\partial u^i \partial v^j} h(u,v)) \mid_{(u,v)=(k,l)} \tag{6.24}$$

for $k,l \in \{0,1\}$ and $i,j \in \{1,\ldots,n\}$. These values, which are called *twist values*, are determined by the cross boundary derivatives (6.20) and (6.23) and are obtained by differentiating the corresponding derivatives. Another possibility for computing the values (6.24) is to solve the linear equations

$$(\frac{\partial^{i+j}}{\partial u^i \partial v^j} f_{\mathbf{b}(k,l)}(\mathbf{b}(u,v))) \mid_{(u,v)=(k,l)}$$

for $k,l \in \{0,1\}$ and $i,j \in \{1,\ldots,n\}$, where the values (6.24) are the unknowns. Because of our choice of the G^n-surface strips and hence of the cross boundary derivatives (6.20) and (6.23), we garante that

$$(\frac{\partial^{i+j}}{\partial u^i \partial v^j} h(u,v)) \mid_{(u,v)=(k,l)} = (\frac{\partial^{i+j}}{\partial v^j \partial u^i} h(u,v)) \mid_{(u,v)=(k,l)}, \tag{6.25}$$

for $k,l \in \{0,1\}$ and $i,j \in \{1,\ldots,n\}$. The fulfillment of condition (6.25), which is called *twist compatibility condition*, is described for G^1-surfaces ($n=1$) and for G^2-surfaces in Lemma 29 and Lemma 30, respectively.

Lemma 29. *Our construction of the boundary functions*

$$h(u,0), h(u,1), h(0,v), h(1,v) \tag{6.26}$$

and the first cross boundary derivatives

$$h_v(u,0), h_v(u,1), h_u(0,v), h_u(1,v). \tag{6.27}$$

ensures that the twist compatibility condition

$$h_{uv}(k,l) = h_{vu}(k,l) \tag{6.28}$$

for $k,l \in \{0,1\}$ is fulfilled.

Proof. We will only prove the twist compatibility condition at the vertex $\mathbf{V}$, i.e.

$$h_{uv}(0,0) = h_{vu}(0,0),$$

because at the other vertices the proof can be done analogously. For this purpose we first show that

$$\mathbf{b}_{uv}(0,0) = \mathbf{b}_{uv}(0,0).$$

Since the first cross boundary derivatives $\mathbf{b}_v(u,0)$ and $\mathbf{b}_u(0,v)$ satisfy the corresponding condition (6.22), i.e.

$$\mathbf{g}(\mathbf{b}(u,0))^\top \cdot \mathbf{b}_v(u,0) = 0, \tag{6.29}$$
$$\mathbf{g}(\mathbf{b}(0,v))^\top \cdot \mathbf{b}_u(0,v) = 0, \tag{6.30}$$

we get by differentiating equation (6.29) with respect to u and by differentiating equation (6.30) with respect to v the following two equations for the vertex $\mathbf{V}$:

$$(\frac{\partial}{\partial u}\mathbf{g}(\mathbf{b}(u,0))^\top \big|_{u=0}) \cdot \mathbf{b}_v(0,0) + \mathbf{g}(\mathbf{b}(0,0))^\top \cdot \mathbf{b}_{vu}(0,0) = 0, \tag{6.31}$$
$$(\frac{\partial}{\partial v}\mathbf{g}(\mathbf{b}(0,v))^\top \big|_{v=0}) \cdot \mathbf{b}_u(0,0) + \mathbf{g}(\mathbf{b}(0,0))^\top \cdot \mathbf{b}_{uv}(0,0) = 0. \tag{6.32}$$

With the help of the fact that

$$\frac{\partial}{\partial u}\mathbf{g}(\mathbf{b}(u,0)) \big|_{u=0} = \frac{\partial}{\partial u}(\bigtriangledown f_{\mathbf{V}})(\mathbf{b}(u,0)) \big|_{u=0} \tag{6.33}$$

and

$$\frac{\partial}{\partial v}\mathbf{g}(\mathbf{b}(0,v)) \big|_{v=0} = \frac{\partial}{\partial v}(\bigtriangledown f_{\mathbf{V}})(\mathbf{b}(0,v)) \big|_{v=0}, \tag{6.34}$$

which is a consequence of (6.11), it is easy to verify that

$$(\frac{\partial}{\partial u}\mathbf{g}(\mathbf{b}(u,0))^\top \big|_{u=0}) \cdot \mathbf{b}_v(0,0) = (\frac{\partial}{\partial v}\mathbf{g}(\mathbf{b}(0,v))^\top \big|_{v=0}) \cdot \mathbf{b}_u(0,0).$$

Therefore we have that

$$\mathbf{g}(\mathbf{b}(0,0))^\top \cdot \mathbf{b}_{vu}(0,0) = \mathbf{g}(\mathbf{b}(0,0))^\top \cdot \mathbf{b}_{uv}(0,0).$$

Since

$$\begin{aligned}\mathbf{b}_{vu}(0,0) = {} & \mathbf{d}_{vu}(0,0) + h_{vu}(0,0)\mathbf{n}(0,0) + h_v(0,0)\mathbf{n}_u(0,0) + \\ & h_u(0,0)\mathbf{n}_v(0,0) + h(0,0)\mathbf{n}_{vu}(0,0)\end{aligned}$$

and

$$\begin{aligned}\mathbf{b}_{uv}(0,0) = {} & \mathbf{d}_{uv}(0,0) + h_{uv}(0,0)\mathbf{n}(0,0) + h_u(0,0)\mathbf{n}_v(0,0) + \\ & h_v(0,0)\mathbf{n}_u(0,0) + h(0,0)\mathbf{n}_{uv}(0,0),\end{aligned}$$

we get

$$h_{vu}(0,0)\mathbf{g}(0,0)\mathbf{n}(0,0) = h_{uv}(0,0)\mathbf{g}(0,0)\mathbf{n}(0,0),$$

which implies that

$$h_{vu}(0,0) = h_{uv}(0,0).$$

It remains to show that the derivatives (6.33) and (6.34) of the gradient are compatible with the boundary curves $\mathbf{b}(u,0)$ and $\mathbf{b}(0,v)$ at the vertex $\mathbf{V}$, i.e.

$$\frac{\partial}{\partial u}\mathbf{g}(\mathbf{b}(u,0)) \,\big|_{u=0} = (\text{Hess}(f_{\mathbf{V}})(\mathbf{b}(0,0))) \cdot \mathbf{b}_u(0,0)$$

and

$$\frac{\partial}{\partial v}\mathbf{g}(\mathbf{b}(0,v)) \,\big|_{v=0} = (\text{Hess}(f_{\mathbf{V}})(\mathbf{b}(0,0))) \cdot \mathbf{b}_v(0,0),$$

respectively. But this is a consequence of the second order contact of the two boundary curves $\mathbf{b}(u,0)$ and $\mathbf{b}(0,v)$ with the function $f_{\mathbf{V}}$ at the vertex $\mathbf{V}$, given in (6.9). □

Lemma 30. *Our construction of the boundary functions (6.26), the first cross boundary derivatives (6.27) and the second cross boundary derivatives*

$$h_{vv}(u,0), h_{vv}(u,1), h_{uu}(0,v), h_{uu}(1,v) \tag{6.35}$$

ensures that the twist compatibility conditions (6.28) and

$$h_{uuv}(k,l) = h_{vuu}(k,l) \tag{6.36}$$
$$h_{uvv}(k,l) = h_{vvu}(k,l), \tag{6.37}$$
$$h_{uuvv}(k,l) = h_{vvuu}(k,l), \tag{6.38}$$

for $k,l \in \{0,1\}$ are fulfilled.

Proof. The proof works similar like in Lemma 29. For the sake of brevity we only give a short overview. At first we show step by step that

$$\mathbf{b}_{uv}(0,0) = \mathbf{b}_{vu}(0,0), \mathbf{b}_{uuv}(0,0) = \mathbf{b}_{vuu}(0,0), \mathbf{b}_{uvv}(0,0) = \mathbf{b}_{vvu}(0,0)$$

and

$$\mathbf{b}_{uvv}(0,0) = \mathbf{b}_{vvu}(0,0),$$

which provides us the twist compatibility conditions (6.28) and (6.36) - (6.38). Finally we show the compatibility of the used derivatives of the gradients and of the used derivatives of the Hessian matrices with the boundary curves $\mathbf{b}(u,0)$ and $\mathbf{b}(0,v)$ at the vertex $\mathbf{V}$. □

6.5.2 Applying Gordon-Coons interpolation

The bubble function h is obtained by applying Gordon-Coons interpolation to the Hermite boundary data, which has been computed in the last subsection. For getting the desired G^n-continuity between the surface patches we have to choose the Gordon-Coons interpolation of degree $2n+1$, which requires Hermite boundary data given by the cross boundary derivatives (6.20) and (6.23). In addition these cross boundary derivatives have to fulfill the twist compatibility condition (6.25), which is shown in Lemma 29 for $n = 1$ and in Lemma 30 for $n = 2$. For more detail of Gordon-Coons interpolation we refer to [27, 54].

6.6 Implementation details and examples

In this section we describe the implementation details of our method and present it on several examples for constructing G^0, G^1 and G^2-surfaces. We demonstrate the smoothness by using reflection lines which is a well-known tool for verifying the resulting geometric continuity (cf. [114]).

6.6.1 Implementation details

We have implemented our algorithms for generating G^n-surfaces for $n \leq 2$ in Mathematica and used Povray for visualizing the surfaces with reflection lines. Since the resulting G^1 and G^2-surfaces are rational surfaces of relatively high degree, we use simple low degree approximations to represent the single surface patches. In detail, we approximate the first and second cross boundary derivatives (6.20) and (6.23), which are responsible for the rationality of the resulting surfaces, by polynomial functions of degree 9. This leads to surface patches of degree $(13, 13)$ for G^1 and of degree $(15, 15)$ for G^2.

For some examples (see Fig. 6.3-6.6), we generate additionally an auxiliary mesh from the original one. This new mesh has the same number of patches as the original one and is generated in the following way. We apply two steps of the standard Catmull-Clark subdivision algorithm (cf. [94]) to the original mesh with normals to obtain a finer mesh with new vertices and normals. Now we do not keep all new obtained vertices and normals, only those, which are related to the old ones. These vertices and normals provide the auxiliary mesh which is used instead of the original mesh. By using the new mesh we generate an approximating surface of the original mesh which possesses a better shape than an interpolating one.

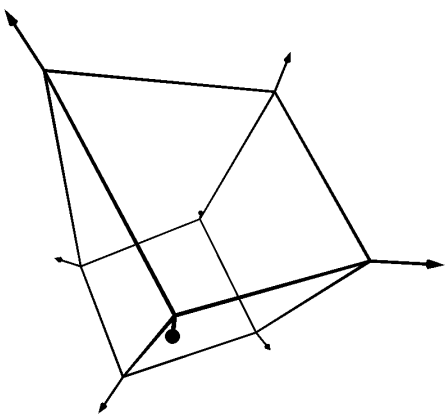
6.6.2 Examples

We show different surfaces that were generated by our method.

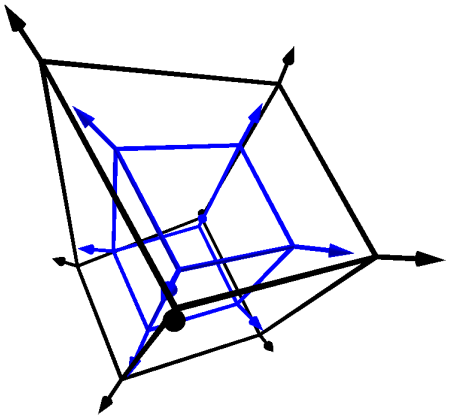
Example 6. In Fig. 6.3-6.6, several meshes and the corresponding auxiliary meshes with associated normals of different objects are visualized. By applying our method to the original or auxiliary meshes we construct G^1 and G^2-surfaces which are interpolating or approximating the vertices and normals of the original meshes, respectively. The reflection lines verify that we have generated G^1 and G^2-surfaces. In the case of G^1 and G^2, all reflection lines of the resulting surfaces are at least G^0 and G^1, respectively. In most of the examples we can also observe that the reflection lines of the G^1-surfaces are only G^0 and not G^1. In most of the examples we can also observe that the reflection lines of the G^1-surfaces are only G^0 and not G^1. ⧫

Example 7. In Fig. 6.6, we have shown the mesh with associated normals of a bottle, given by 72 quadrilaterals. We have used the mesh to generate G^0, G^1 and G^2-surfaces. The resulting surfaces are presented with reflection lines to demonstrate the corresponding smoothness. ⧫

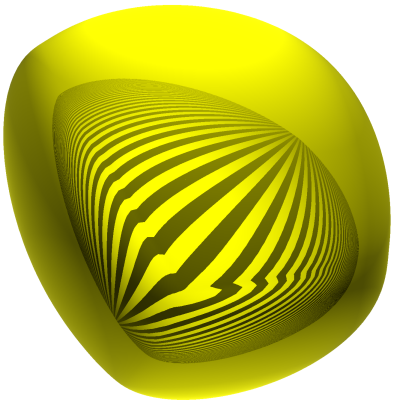
Example 8. In Figure 6.7, we have presented the mesh of the implicitly defined algebraic surface $F = \{(x, y, z)^\top | z = x^3 - 3xy^2\}$ (monkey saddle) with a vertex of valency 6 in the saddle point $(0, 0, 0)^\top$ in two different views (side and top view). The mesh consists of 22 patches with vertices of different valencies (3, 4, 5 and 6) and associated normals. For better visibility of the mesh in Figure 6.7, we have not displayed the associated normals (obtained from the implicitly defined algebraic surface F in the vertices). We have used the mesh to construct G^1 and G^2-surfaces by applying our design method. The resulting smoothness of the surfaces is again verified by using reflection lines. ⧫



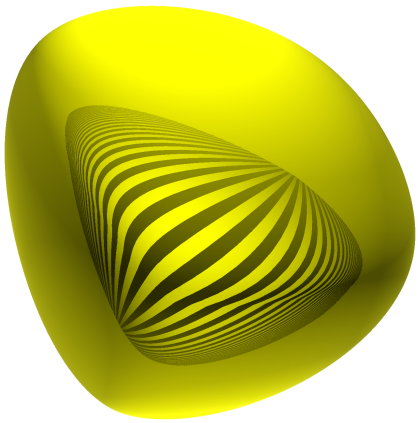
Mesh of a simple object



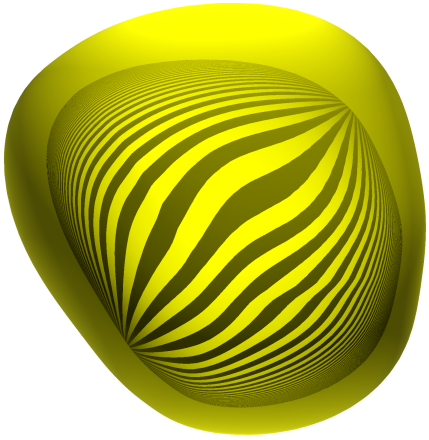
Mesh with auxiliary mesh



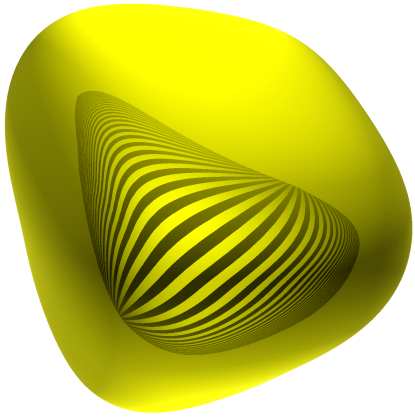
interpolating G^1-surface



approximating G^1-surface

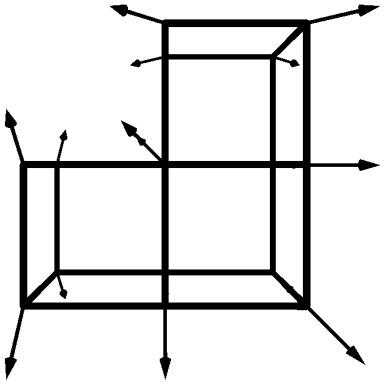


interpolating G^2-surface

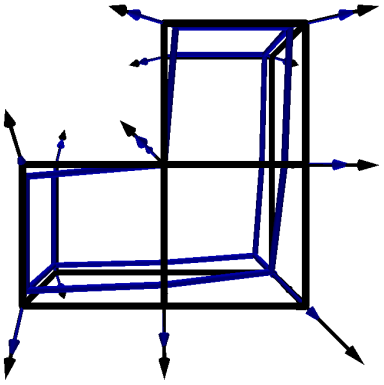


approximating G^2-surface

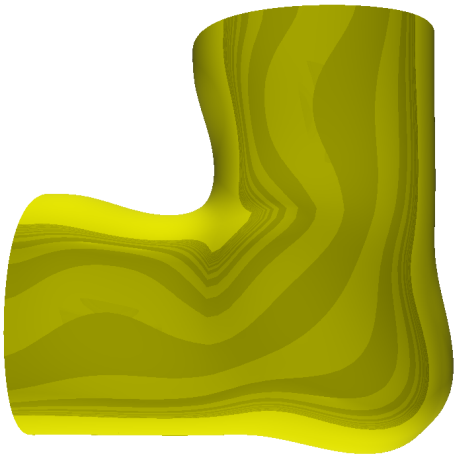
Figure 6.3: The mesh and the auxiliary mesh (blue) with associated normals of a simple object (cube-like object), described by 6 quadrilaterals, and the resulting interpolating and approximating G^1 and G^2-surfaces with reflection lines.



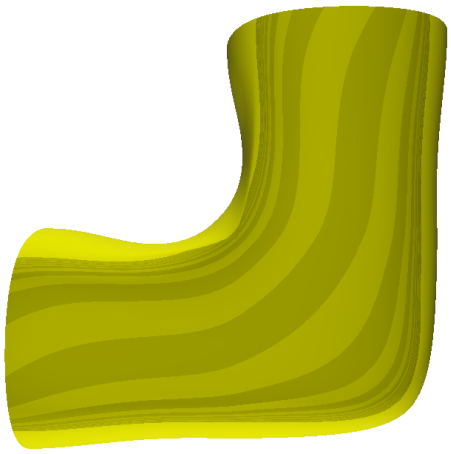
Mesh of a tube



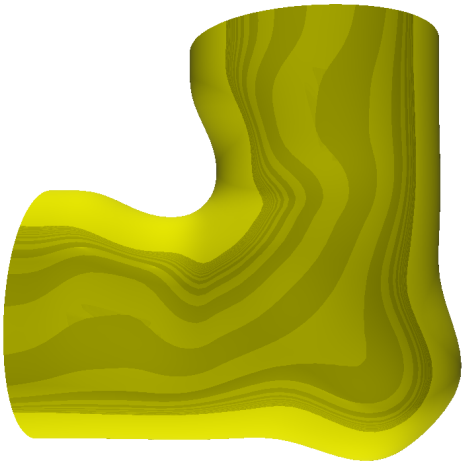
Mesh with auxiliary mesh



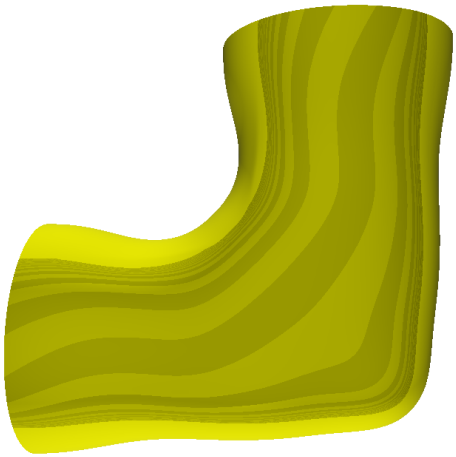
interpolating G^1-surface



approximating G^1-surface

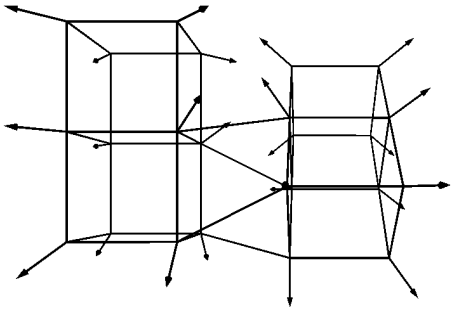


interpolating G^2-surface

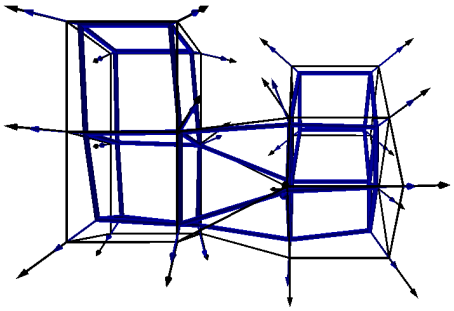


approximating G^2-surface

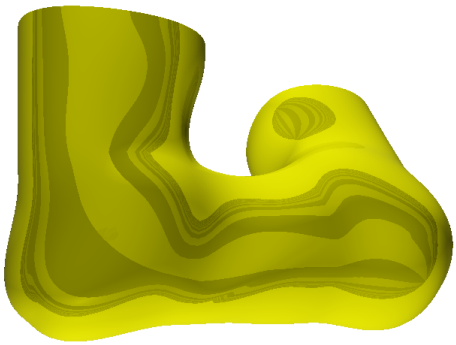
Figure 6.4: The mesh and the auxiliary mesh (blue) with associated normals of a tube, described by 12 quadrilaterals, and the resulting interpolating and approximating G^1 and G^2-surfaces with reflection lines.



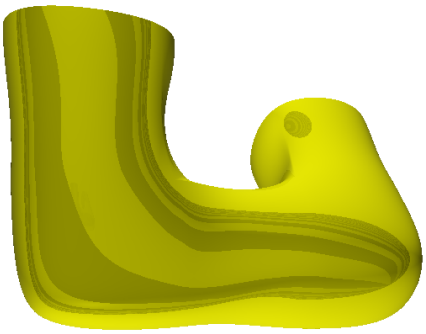
Mesh of a rotated horse shoe



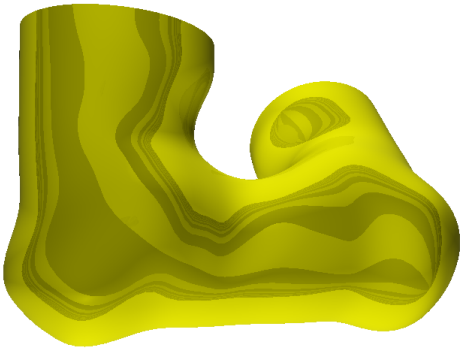
Mesh with auxiliary mesh



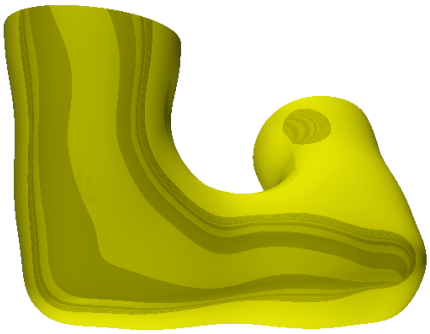
interpolating G^1-surface



approximating G^1-surface



interpolating G^2-surface



approximating G^2-surface

Figure 6.5: The mesh and the auxiliary mesh (blue) with associated normals of a rotated horse shoe, described by 20 quadrilaterals, and the resulting interpolating and approximating G^1 and G^2-surfaces with reflection lines.

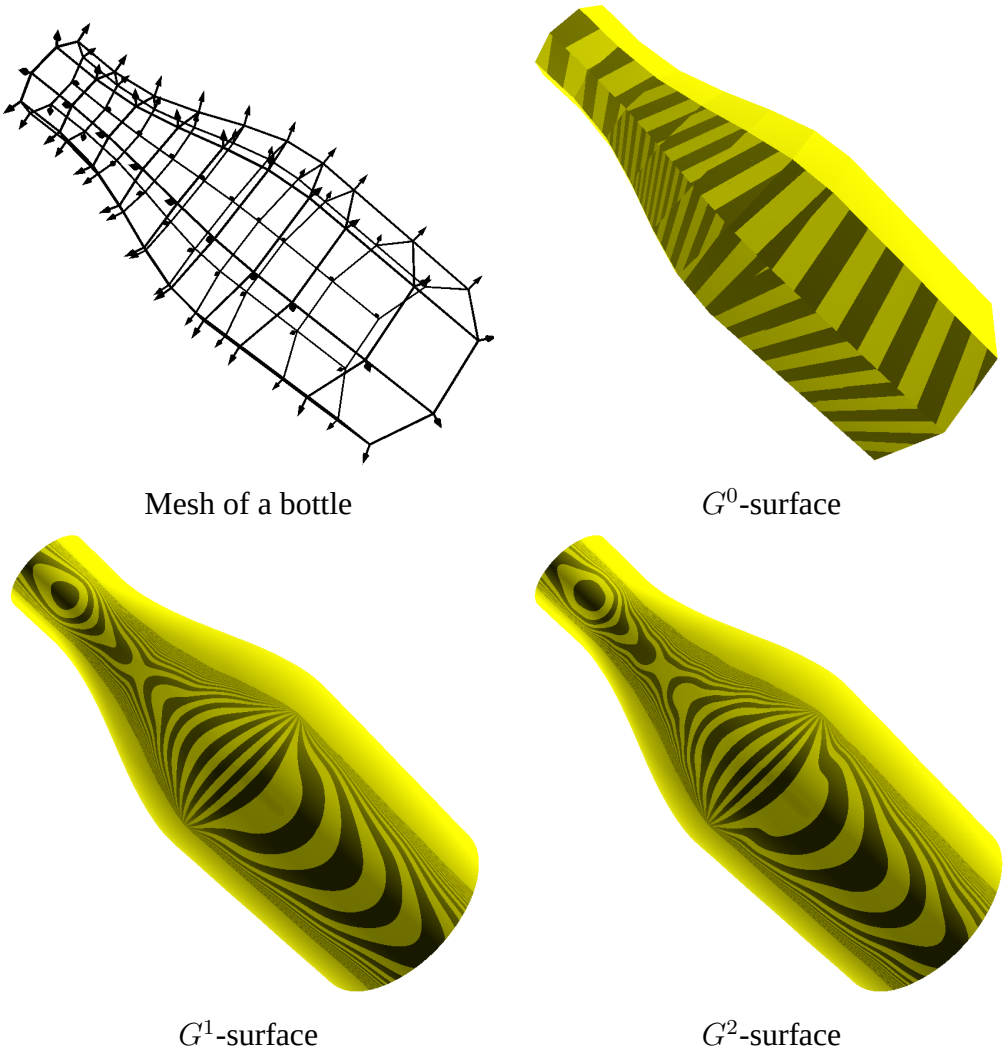


Figure 6.6: The mesh with associated normals of a bottle, described by 72 quadrilaterals, and the resulting G^0, G^1 and G^2-surfaces with reflection lines.

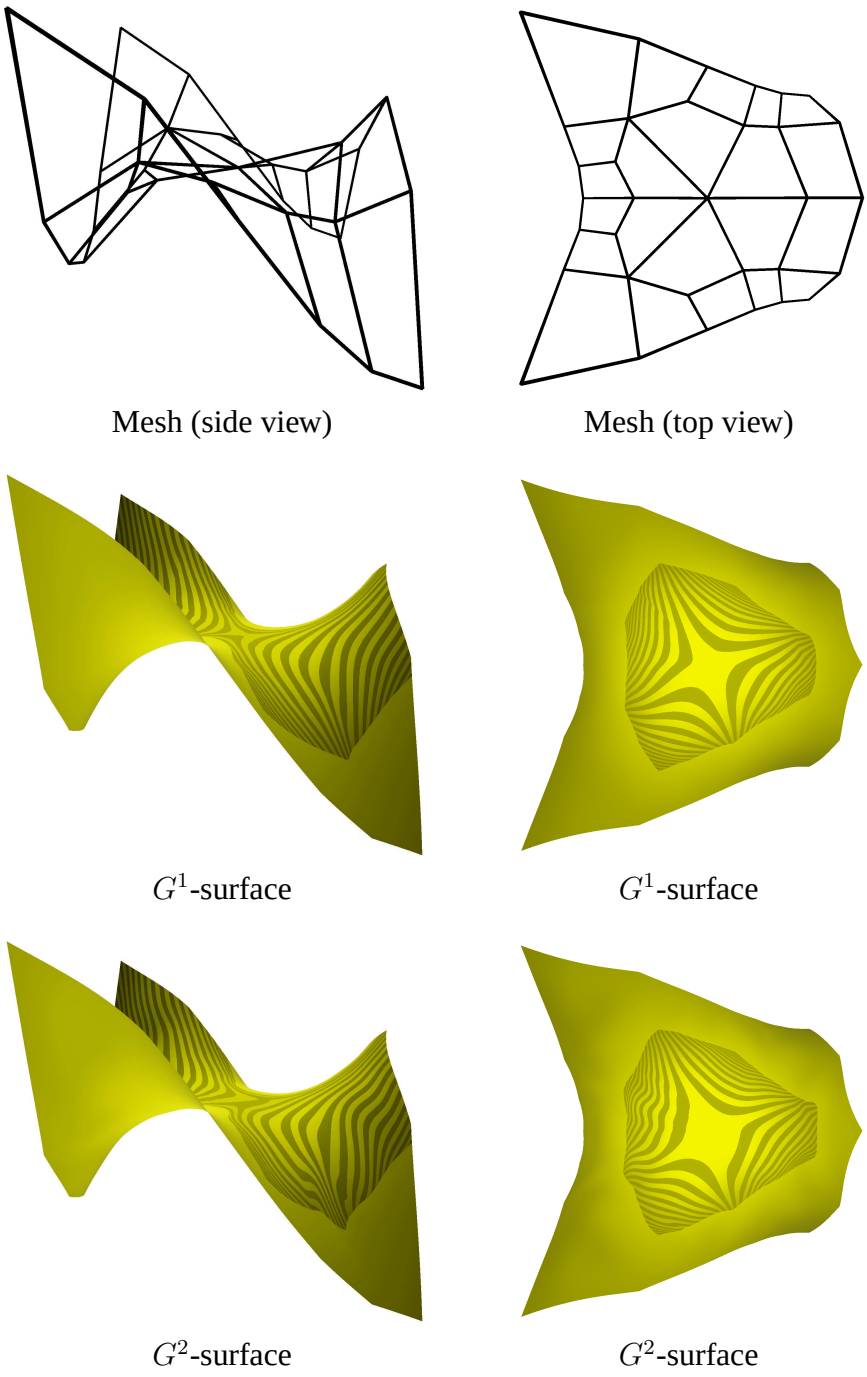


Figure 6.7: The mesh of the implicitly defined algebraic surface $F = \{(x, y, z)^\top | z = x^3 - 3xy^2\}$ (monkey saddle) with a vertex of valency 6 in the saddle point $(0, 0, 0)^\top$ in two different views and the resulting G^1 and G^2-surfaces with reflection lines.

Part IV

SUMMARY

7 Conclusion

This doctoral thesis dealt with data interpolation, which is still active research area in CAGD. In the first part of the thesis, we served historical overview of CAGD and we focused on a state of art of Pythagorean hodograph curves and related topics.

Chapters 4 and 5 are devoted to the study of PH curves. In Chapter 4, we investigated G^1 Hermite interpolation by PH cubic, where we described for which input data an interpolating PH cubic arc exists. These results are extension of the results in [80], which were moreover not precise. Further, we discussed the quality and a number of PH cubic interpolant for input data. We described for which data only one, two qualitatively (if it contain a loop or not) different or two qualitatively same interpolants exist. These results have been already published in the journal Computer Aided Geometric Design, see [9].

Since interpolating PH cubic does not exist for arbitrary G^1 Hermite data, in Chapter 5 we analyzed how many PH cubics are needed to interpolate any G^1 Hermite data. Moreover, we demonstrated that PH cubic is also suitable for C^1 Hermite interpolation, although it has not so much flexibility like standard cubic. We proved that any C^1 Hermite data can be fitted by a pair of PH cubic joined in C^1 continuity. In the thesis is also presented a technique how to gain all four solutions, which match C^1 Hermite interpolation. In the Section 5.4, we constructed C^1 TC-spline and used it for a curve approximation. Since the lengths of tangent vectors influence a lot the shape of TC-spline curve, and it is not desirable to obtain a curve with selfintersections, we formulated a conjecture about the length of tangent vectors, which yield good TC-splines. Further research in this topic could be to analyze the direction of tangent vectors in curve approximation. We computed the direction from a curve, which was approximated and there is still a question whether a better choice exists.

In Chapter 6 we developed a new technique for interpolation of quadrilateral meshes with associated normals. The approach is called Bubble patch and is based on Gordon-Coons interpolation. Our construction possessed a rational parametrization and moreover the computation of arbitrary G^n continuity between neighboring patches was presented. In Section 6.4 we showed how to construct G^0, G^1 and G^2-surfaces in detail. Further, we presented the advantages of our method. One of them was computing compatible twist values at the vertices, which we got as a solution of linear system of equations. Another one is that the computation was not limited by the valency of quadrilateral mesh. The reason is that the approach was based on local properties of the mesh. At the end of Chapter 6

we verified the desired smoothness by the reflection lines on several examples.

Although the Bubble patch method was constructed for quadrilateral mesh with associated normal vectors, there is a challenge to generalize it for triangular meshes with normal vectors (which are not needed, since they can be computed from the behavior of the given mesh). The advantage of the generalization from quadrilateral mesh to triangular could be significant because a lot of solid object is represented by triangular meshes.

Bibliography

[1] Bakenov, A.: *T-splines: Tensor product B-spline Surfaces with T-Junction*. Master's thesis, Brigham Young University, 2001.

[2] Bartoň, M., Eldber, G.: *Spiral fat arcs – Bounding regions with cubic convergence*. Graphical Models, Vol 73, pp. 49-56, Elsevier, 2010.

[3] Bastl, B., Jüttler, B., Kosinka, J., Lávička, M: *Volumes with piecewise quadratic medial surface transforms: Computation of boundaries and trimmed offsets*. Computer Aided Design, Vol. 42, pp. 671-679, 2010.

[4] Beltran, J.V., Monterde, J.: *A characterization of quintic helice*. J. Comput. Appl. Math., Vol. 26, pp. 116-121, 2007.

[5] Bishop, R. L.: *There is More than One Way to Frame a Curve*. The Americal Mathematical Monthly, Vol. 82, No. 3, pp. 246-251, 1975.

[6] Blaschke, W.: *Differentialgeometrie*. 1923.

[7] Blum, H.: *A transformation for extracting new descriptors of shape*. In W. Wathen-Dunn, editor, Models for the perception of speech and visual form, MIT Press, pp. 362–380, 1967.

[8] Brechner, E.L.: *General offset curves and surfaces*. R. Barnhill, Geometry Processing for Design and Manufacturing, Philadelphia: SIAM, pp. 101-121, 1992.

[9] Byrtus, M., Bastl, B.: G^1 *Hermite interpolation by PH cubics revisited*. Computer Aided Geometric Design, Vol. 27, pp. 622-630, 2010.

[10] Catmull, E.E., Clark, J.H.: *Recursively generated B-spline surfaces on arbitrary topological meshes*. Computer Aided Geometric Design, Vol. 10, pp. 350-355, 1978.

[11] Casteljau, P. F.: *Courbes et surface à pôles*. Technical report, Citroën, Paris, 1963.

[12] Chen, Y. J., Ravani, B.: *Offset surface generation and contouring in Computer Aided Desing*. ASME Journal of Mechanisms, Transmissions and Automation in Design, Vol. 109, pp. 133-142, 1987.

[13] Choi, H. I., Farouki, R. T., Kwon, S. H., Moon, H. P.: *Topological criterion for selection of quintic Pythagorean–hodograph Hermite interpolants*. Computer Aided Geometric Design, Vol. 25, pp. 411-433, 2008.

[14] Choi, H. I., Han, C. Y.: *Euler-Rodrigues frames on spatial Pythagorean-hodograph curves*. Computer Aided Geometric Design, Vol. 19, pp. 603-620, 2002.

[15] Choi, H. I., Han, C. Y., Moon, H. P., Wee, N. S.: *New algorithm for medial axis transform of plane domain*. Graph. Models Image Proc. 59, pp. 463-483, 1997.

[16] Choi, H. I., Lee, D. S., Moon, H. P.: *Clifford algebra, spin representation and rational parametrization of curves and surfaces*. Advances in Computational Mathematics, Vol. 17, pp. 5-48, 2002.

[17] Cotrina-Navau, J., Garcia, N. P.: *Modeling surfaces from meshes of arbitrary topology*. Compuer Aided Geometric Design, Vol. 17, pp. 643-671, 2000.

[18] Černohorská, E., Šír, Z.: *Support Function of Pythagorean Hodograph Cubics and G^1 Hermite Interpolation*. In Geometric Modeling and Processing 2010, Lecture Notes in Computer Science 6130, pp 29–42. Springer, 2010.

[19] de Boor, C: *Bicubic Spline inerpolation*. Journal of Mathematics and Physics, Vol. 41, pp. 212-218, 1962.

[20] Della Vecchia, G., Jüttler, B.: *Piecewise Rational Manifold Surfaces with Sharp Features*. The Mathematics of Surfaces XIII, Lecture Notes in Computer Science, pp. 90-105, 2009.

[21] Della Vecchia, G., Jüttler, B., Kim, M.S.: *A construction of rational manifold surfaces of arbitrary topology and smoothness from triangular meshes*. Computer Aided Geometric Design, Vol. 25, pp. 801-815, 2008.

[22] Dietz, R., Hoschek, J., Jüttler, B.: *An algebraic approach to curves and surfaces on the sphere and on other quadrics*. Computer Aided Geometric Design, Vol. 10, pp. 211-229, 1993.

[23] Doo, D., Sabin, M.: *Behaviour of recursive division surfaces near extraordinary points*. Computer Aided Geometric Design, Vol. 10, pp. 356-360, 1978.

[24] Duhamel du Monceau, H.L.: *Eléments de l'Architecture Navale ou Traité Pratique de la Construction des Vaissaux*. Paris, 1752.

[25] Eisenhart, L. P.: *A Treatise on the Differential Geometry of Curves and Surfaces*. Ginn and Co., Boston, 1909.

[26] Fakuda, K., Weibel, C.: *Computing All Faces of the Minkowski Sum of $\mathcal{V}$-Polytopes*. Proceedings of the 17^{th} Canadian Conference on Computational Geometry, 2005.

[27] Farin, G. : *Curves and surfaces for computer-aided geometric design*. Academic Press, 1997.

[28] Farouki, R. T.: *Pythagorean-Hodograph Curves: Algebra and Geometry Inseparable*. Series: Geometry and Computing. Springer, Berlin, 2008.

[29] Farouki, R. T.: *The conformal map $z \rightarrow z^2$ of the hodograph plane*. Computer Aided Geometric Design, Vol. 11, pp. 363-390, 1994.

[30] Farouki, R. T.: *The approximation of non-degenerate offset surface*. Computer Aided Design, Vol. 3, pp. 15-43, 1986.

[31] Farouki, R. T., Giannelli, C., Manni, C., Sestini, A.: *Quintic space curves with ratinal-minimizing frames*. Computer Aided Geometric Design, Vol 26, pp. 580-592, 2009.

[32] Farouki, R. T., Giannelli, C., Sestini, A.: *Helical polynomial curves and double Pythagorean hodographs I. Enumeration of low-degree curves*. Computer Aided Geometric Design, Vol. 44, pp. 307-332, 2009.

[33] Farouki, R. T., Giannelli, C., Sestini, A.: *Helical polynomial curves and double Pythagorean hodographs I. Quaternion and Hopf map representations*. Computer Aided Geometric Design, Vol. 44, pp. 161-179, 2009.

[34] Farouki, R. T., Han, C. Y.: *Algorithms for spatial Pythagorean-hodograph curves*. Geometric Properties for Incomplete Data, Springer, pp. 43-58, 2006.

[35] Farouki, R. T., Han, C. Y.: *Rational approximation schemes for rotation-minimizing frames on Pythagorean-hodograph curves*. Computer Aided Geometric Design, Vol. 20, pp. 435-454, 2003.

[36] Farouki, R. T., Neff, C. A.: *Hermite interpolation by Pythagorean-hodograph quintics*. Mathematics of Computation, Vol. 64, Issue 212, pp. 1589-1609. American Mathematical Society, 1995.

[37] Farouki, R. T., Sakkalis, T.: *Pythagorean hodograps*. IBM Journal of Research and Development, Vol. 34, No. 5, pp. 736-753. IBM, 1990.

[38] Farouki, R. T., Sakkalis, T.: *Pythagorean hodograph space curves*. Advances in Computational Mathematics, Vol. 2, No. 1, pp. 41-66. Springer, 1994.

[39] Farouki, R. T., Sakkalis, T.: *Rational rotation-minimizing frames on polynomial space curves of arbitrary degree*. Journal of Symbolic computation, Vol. 45, pp. 844-856, 2010.

[40] Farouki, R. T., Sederberg, T. W.: *Analysis of the offset to a parabola*. Computer Aided Geometric Design, Vol. 12, pp. 639-645, 1995.

[41] Farouki, R. T., Šír, Z.: *Rational Pythagorean-hodograph curves*. Computer Aided Geometric Design, Vol. 28, pp. 75-88, 2011.

[42] Forrest, A.R.: *Interactive interpolation and approximation by Bézier polynomials*. The Computer Journal, Vol. 15, pp. 71-79, 1973.

[43] Grimm, C.M.: *Parameterization using manifolds*. International Journal of Shape Modeling, Vol. 10, pp. 51-80, 2004.

[44] Grimm, C.M.: *Simple manifolds for surface modeling and parameterization*. Shape Modeling International, pp. 237-245, 2002.

[45] Grimm, C.M., Hughes, J.F.: *Modeling surfaces of arbitrary topology*. Siggraph'95, pp. 77-87, 1995.

[46] Hahmann, St., Bonneau, G.P.: *Triangular G^1 interpolation by 4-splitting domain triangles*. Computer Aided Geometric Design, Vol. 17, pp. 731-757, 2000.

[47] Hahmann, St., Bonneau, G.P., Caramiaux, B.: *Bicubic G^1 interpolation of irregular quad meshes using a 4-split*. Advances in geometric modeling and processing, Lecture Notes in Comput. Sci., pp. 17-32, 2008.

[48] Han, C. Y.: *Nonexistence of rational rotation-minimizing frame on cubic curves*. Compuer Aided Geometric Design, Vol. 25, pp. 298-304, 2008.

[49] Hermann, T.: *G^2 interpolation of free form curve networks by biquintic Gregory patches*. Computer Aided Geometric Design, Vol. 13, pp. 873-893, 1996.

[50] Hermann, T., Peters, J., Strotman, T.: *A geometric criterion for smooth interpolation of curve networks*. SIAM/ACM Joint Conferences on Geometric and Physical Modeling, Keyser, J., ACM, New York, pp. 169-173, 2009.

[51] Hermann, T., Peters, J., Strotman, T.: *Constraints on curve networks suitable for G^2 interpolation*. Lecture Notes in Computer Science, pp. 77-87, 2010.

[52] Hoschek, J.: *Offset curves in the plane*. Computer Aided Design, Vol. 17, pp. 77-82, 1985.

[53] Hoschek, J.: *Spline approximation of offset curves*. Computer Aided Geometric Design, Vol. 20, pp. 475-483, 1988.

[54] Hoschek, J. and Lasser, D. : *Fundamentals of computer aided geometric design*. A K Peters Ltd., Wellesley, MA, 1993.

[55] Jaklič, G., Kozak, J., Krajnc, M., Vitrih, V., Žagar, E.: *Geometric Lagrange interpolation by planar cubic Pythagorean-hodograph curves*. Computer Aided Geometric Design, Vol. 25, pp. 720-728, 2008.

[56] Jaklič, G., Kozak, J., Krajnc, M., Vitrih, V., Žagar, E.: *On interpolation by planar cubic G^2 pythagorean-hodograph spline curves*. Mathematics of Computation, Vol. 79, Issue 269, pp. 305-326. American Mathematical Society, 2010.

[57] Jüttler, B., Mäurer, C.: *Cubic Pythagorean hodograph spline curves and applications to sweep surface modeling*. Computer Aided Design, Vol. 31, pp. 73-83, 1999.

[58] Jüttler, B.: *Hermite interpolation by Pythagorean hodograph curves of degree seven*. Mathematics of Computation, Vol. 70, pp. 1089-1111. American Mathematical Society, 2001.

[59] Jüttler, B., Wagner, M. G.: *Rational motion-based surface generation*. Computer Aided Design, Vol. 3, pp 217-229, 1999.

[60] Karčiauskas, K., Peters, J.: *Guided spline surfaces*. Computer Aided Geometric Design, Vol. 26, pp. 105-116, 2009.

[61] Klok, F.:*Two moving coordinate frames for sweeping along a 3D trajectory*. Computer Aided Geometric Design, Vol. 3, pp. 217-229, 1986.

[62] Kosinka, J., Jüttler, B.: C^1 *Hermite interpolation by Pythagorean hodograph quintics in Minkowski space*. Advances in Computational Mathematics, Vol. 30, pp. 123-140, 2009.

[63] Kosinka, J., Jüttler, B.: G^1 *Hermite interpolation by Minkowski Pythagorean hodograph cubic*. Computer Aided Geometric Design, Vol. 23, pp. 401-418, 2006.

[64] Kosinka, J., Jüttler, B.: *MOS surfaces: Medial surface transforms with rational domain boundaries*. The Mathematics of Surfaces XII, R. Martin, M. Sabin, J. Winkler (eds.), Lecture Notes in Computer Science, Springer, pp. 245-262, 2007.

[65] Kosinka, J., Lávička, M.: *On rational Minkowski Pythagoreal hodograph curves*. Computer Aided Geometric Design, Vol. 27, pp. 514-524, 2010.

[66] Kosinka, J., Šír, Z.: C^2 *Hermite interpolation by Minkowski Pythagorean hodograph curves and medial axis transform approximation*. Computer Aided Geometric Design, Vol. 27, pp. 631-643, 2010.

[67] Krasauskas, R.: *PN-surfaces for blending applications*. In: Electronic Proceedings of Effective Methods in Algebraic Geometry (MEGA) 2007, [cit. 2007/08/11], `http://www.ricam.oeaw.ac.at/mega2007/electronic/44.pdf`.

[68] Krasauskas, R.: *Branching blend of natural quadrics based on surfaces with rational offsets*. Computer Aided Geometric Design, Vol. 25, pp. 332–341, 2008.

[69] Kubota, K. K.: *Pythagorean Triples in Unique Factorization Domains*, Amer. Math. Monthly 79, pp. 503-505, 1973.

[70] Lávička, M., Bastl, B.: *PN surfaces and their convolutions with rational surfaces*. Computer Aided Geometric Design, Vol. 25, pp. 763-774, 2008.

[71] Lávička, M., Bastl, B.: *Rational hypersurfaces with rational convolutions*. Computer Aided Geometric Design, Vol. 24, pp. 410–426, 2007.

[72] Lee, I. K., Kim, M. S., Eldber, G.: *The Minkowski Sum of 2D Curved Objects*. Proceedings of Israel-Korea Bi-National Conference on New Themes in Computerized Geometrical Modeling, Tel Aviv Univ., pp. 155-164, 1998.

[73] Liu, Q., Sun, T.C.: G^1 *interpolation of mesh curves*. Computer Aided Geometric Design, Vol. 26, pp. 259-267, 1994.

[74] Loop, C.T.: *Smooth spline surfaces over irregular meshes*. Proc. SIGGRAPH, pp. 303-310, 1994.

[75] Loop, C.T.: *Smooht subdivision surfaces based on triangles*. M.S. Thesis, Department of Mathematics, University of Utah, 1987.

[76] Loop, C.T., DeRose, T.: *Generalized B-spline Surfaces of Arbitrary Topology*. Computer Graphics, Vol. 24, pp. 346-357, 1990.

[77] Lü, W., Pottmann, H.: *Rational parameterization of quadrics and their offsets*. Computing, Vol. 57, pp. 135-147. Springer, 1996.

[78] Lü, W., Pottmann, H.: *Pipe surfaces with rational spine curve are rational*. Computer Aided Geometric Design, Vol. 13, pp. 621-628, 1996.

[79] Mäurer, C., Jüttler, B.: *Rational approximation of rotation minimizing frames using Pythagorena–hodograph curves*. Journal for Geometry and Graphics, Vol. 3, No. 2, pp. 141-159, 1999.

[80] Meek, D. S., Walton, D. J.: *Geometric Hermite interpolation with Tschirnhausen cubics*. Journal of Computational and Applied Mathematics, Vol. 81, pp. 299-309, 1997.

[81] Meek, D. S., Walton, D. J.: *Hermite interpolation with Tschirnhausen cubic spirals*. Computer Aided Geometric Design, Vol. 14, pp. 619-635, 1997.

[82] Moon, H. P.: *Minkowski Pythagorean Hodographs*. Computer Aided Geometric Design, Vol. 16, pp. 739-753, 1999.

[83] Moon, H. P., Farouki, R. T., Choi, H. I.: *Construction and shape analysis of PH quintic Hermite interpolants*. Computer Aided Geometric Design, Vol. 18, pp. 93-115, 2001.

[84] Pelosi, F., Farouki, R. T., Manni, C., Sestini, A.: *Geometric Hermite interpolation by spatial Pythagorean-hodograph cubics*. Advances in Computational Mathematics, Vol. 22, No. 4, pp. 325-352. Springer, 2005.

[85] Pelosi, F., Sampoli, M. L., Farouki, R. T., Manni, C.: *A control polygon scheme for design of planar C^2 PH quintic spline curves*. Computer Aided Geometric Design, Vol. 24, pp. 28-52, 2007.

[86] Peternell, M.: *Rational two-parameter families of spheres and rational offset surfaces*. Journal of Symbolic Computation, Vol. 45, pp. 1-18, 2010.

[87] Peternell, M., Odehnal, B., Sampoli, M.L.: *On quadratic two-parameter families of spheres and their envelopes*. computer Aided Geometric Design, Vol. 25, pp. 342-355, 2008.

[88] Peternell, M., Odehnal, B.: *Convolution surfaces of quadratic triangular Bézier surfaces*. Computer Aided Geometric Design, Vol.25, pp. 116–129, 2008.

[89] Peternell, M., Pottmann, H.: *A Laguerre geometric approach to rational offsets*. Computer Aided Geometric Design, Vol. 15, pp. 223-249, 1998.

[90] Peternell, M., Pottmann, H.: *Computing Rational Parametrizatin of Canal Surfaces*. Journal of Symbolic Computation, Vol. 23, pp. 255-266, 1997.

[91] Peters, J.: *C^2 free-form surfaces of degree $(3, 5)$*. Computer Aided Geometric Design, Vol. 19, pp. 113-126, 2002.

[92] Peters, J.: *Geometric continuity*. Handbook of computer aided geometric design, North-Holland, Amsterdam, pp. 193-227, 2002.

[93] Peters, J.: *Smooth interpolation of a mesh of curves*. Constructive Approximation. An International Journal for Approximations and Expansions, Vol. 7, pp. 221-246, 1991.

[94] Peters, J., Reif, U.: *Subdivision surfaces.* Vol. 3 of Geometry and Computing. Springer-Verlag, Berlin, 2008.

[95] Piegl, L., Tiller, W.: *The NURBS Book*. Monographs in Visual Communications. Springer, Berlin, 1997.

[96] Pottmann, H.: *Applications of the dual Bézier representation of rational curves and surfaces*. Curves and Surfaces in Geometric Design, A K Peters, Wellesly, MA, pp. 377-384, 1994.

[97] Pottmann, H.: *Curve design with rational Pythagorean-hodograph curves*. Advances in Comp. Mathematics, Vol. 3, pp. 147-170, 1995.

[98] Pottmann, H.: *Rational curves and surfaces with rational offsets*. Computer Aided Geometric Design, Vol. 12, pp. 175-192,1995.

[99] Pottmann, H., Peternell, M.: *Application of Laguerre geometry in CAGD*. Computer Aided Geometric Design, Vol. 15, pp. 165-186. Elsevier, 1998.

[100] Powell,M. D. J., Sabin, M.:*Piecewice quadratic approximation on triangles*. ACM Trans. Math. Software, Vol. 3, pp. 316-325, 1977.

[101] Prautzsch, H.: *Freeform splines*. Computer Aided Geometric Design, Vol. 14, pp. 201-206, 1997.

[102] Reif, U.: *Biquadratic G-spline surfaces*. Computer Aided Geomteric Design, Vol. 12, pp. 193-205, 1995.

[103] Reif, U.: *TURBS—topologically unrestricted rational B-splines*. Constructive Approximation. An International Journal for Approximations and Expansions, Vol. 14, pp. 57-77, 1998.

[104] Sabin, M.: *Offset parametric surfaces*. Technical Report VTO/MS/149, British Aircraft Corporation, 1968.

[105] Sampoli, M.L., Peternell, M., Jüttler, B.: *Rational surfaces with linear normals and their convolutions with rational surfaces*. Computer Aided Geometric Design, Vol. 23, pp. 179–192, 2006.

[106] Shoenberg, I.J.: *Contributions to the problem of approximation of equidistant data by analytic functions*. Quart. Appl. Math., Vol. 4, pp. 45-99, 112-141, 1946.

[107] Sederberg, T. W., Nishita, T.: *Curve intersection using Bézier clipping*. Computer Aided Design, Vol. 22, pp538-550, 1990.

[108] Sederberg, T.W., Zheng, J., Bakenov, A., Nasri, A.: *T-splines and T-NURCCs*, CM Transactions on Graphics, Vol. 22, pp. 477-484, 2003.

[109] Sederberg, T.W., Zheng, J., Sewell, D., Sabin, M.: *Non-uniform Subdivision Surfaces*. Computer Graphics Annual Conference Series, pp. 387-394, 1998.

[110] Stone, M. C., DeRose, T. D.: *A Geometric Characterization of Parametric Cubic Curves*. ACM Transactions on Graphics, ACM, Vol. 8, No. 3, pp. 147-163, 1989.

[111] Šír, Z., Bastl, B., Lávička, M.: *Hermite interpolation by hypocycloids and epicycloids with rational offsets*. Computer Aided Geometric Design, Vol. 27, pp. 405-417, 2010.

[112] Šír, Z., Feichtinger, R., Jüttler, B.: *Approximating curves and their offsets using biarcs and Pythagorean hodograph quintics*. Computer Aided Design, Vol. 38, pp. 608-618. Elsevier, 2006.

[113] Šír, Z. and Jüttler, B.: C^2 *Hermite interpolation by Pythagorean hodograph space curves*. Mathematics of Computation, Vol. 76, pp. 1373-1391, 2007.

[114] Theisel, H.: *Are isophotes and reflection lines the same?* Special issue Pierre Bézier, Computer Aided Geometric Design, Vol. 18, pp. 711-722, 2001.

[115] `www.tspline.com`

[116] Wang, G., Fang, L.: *On control polygons of quartic Pythagorean-hodograph curves*. Computer Aided Geometric Design, Vol. 26, pp 1006-1015, 2009.

[117] Wang, W., Jüttler, B., Zheng, D. , Liu, Y.: *Computation of Rotation minimizing frames*. ACM Transaction on Graphics, Vol. 27, No. 1, Article 2, 2008.

[118] Wang, W., Joe, B. :*Robust computation of the rotation minimizing frame for sweep surface modeling*. Computer Aided Geometric Design, Vol. 29, No. 5, pp. 379-391, 1997.

[119] Ye, X.: *Curvature continuous interpolation of curve meshes*. Computer Aided Geometric Design, Vol. 14, pp. 169-190, 1997.

[120] Ying, L., Zorin, D.: *A simple manifold-based construction of surfaces of arbitrary smoothness*. Proc.Siggraph, ACM Transactions on Graphics, Vol. 23, pp. 271-275, 2004.

[121] Yoon, S.H.: *A surface displaced from a manifold*. Geometric Modeling and Processing, Lecture Notes in Computer Science, pp. 677-686, 2006.

List of authors' publications

- *G^1 Hermite interpolation by PH cubics revisited.* Computer Aided Geometric Design, Vol. 27, pp. 622-630, 2010. (co-autor B. Bastl)
- *Interpolating Bubble Patches on Quadrilateral Meshes.* submitted to journal Computer Aided Geometric Design. (co-autors M. Kapl, B. Jüttler)
- *Existence of PH cubic interpolant for G^1 Hermite interpolation problem.* Proceedings of the 29th Conference on Geometry and Graphics, ISBN: 80-86195-61-9, pp. 95-104, 2009.
- *PH cubics fitting.* Proceedings of 28th Conference on Geometry and Graphics, ISBN 978-80-7375-249-1, 2008.
- *Solving undercut problem using offset theory and differential geometry.* Proceedings of 7th international Conference Aplimat, ISBN 978-80-89313-03-7, pp. 595-604, 2008.
- *Solving undercut problem using offset theory and differential geometry.* Journal of Applied Mathematics, Vol. 1, No. 2, pp. 21-30, 2008.
- *Offset and Differential Geometry in Mathematica.* Proceedings of Wolfram Technology Conference, 2007.