Università
degli Studi
di Padova

Sede amministrativa: Università degli Studi di Padova
Sede consorziata: Alma Mater Studiorum · Università di Bologna

Dipartimento di Matematica

Scuola di Dottorato in Scienze matematiche
Indirizzo: Matematica computazionale
Ciclo XXVII

New strategies for curve and arbitrary-topology surface constructions for design

Direttore della Scuola:
Chiar.mo Prof. Pierpaolo Soravia

Coordinatore di indirizzo:
Chiar.ma Prof.ssa Michela Redivo Zaglia

Supervisore:
Chiar.mo Prof. Giulio Casciola

Candidato:
Michele Antonelli

New strategies for curve and arbitrary-topology surface constructions for design

Michele Antonelli

antonelli.miki@gmail.com

January 2015

Abstract

This dissertation presents some novel constructions for curves and surfaces with arbitrary topology in the context of geometric modeling. In particular, it deals mainly with three intimately connected topics that are of interest in both theoretical and applied research: subdivision surfaces, non-uniform local interpolation (in both univariate and bivariate cases), and spaces of generalized splines.

Specifically, we describe a strategy for the integration of subdivision surfaces in computer-aided design systems and provide examples to show the effectiveness of its implementation.
Moreover, we present a construction of locally supported, non-uniform, piecewise polynomial univariate interpolants of minimum degree with respect to other prescribed design parameters (such as support width, order of continuity and order of approximation).
Still in the setting of non-uniform local interpolation, but in the case of surfaces, we devise a novel parameterization strategy that, together with a suitable patching technique, allows us to define composite surfaces that interpolate given arbitrary-topology meshes or curve networks and satisfy both requirements of regularity and aesthetic shape quality usually needed in the CAD modeling framework.
Finally, in the context of generalized splines, we propose an approach for the construction of the optimal normalized totally positive (B-spline) basis, acknowledged as the best basis of representation for design purposes, as well as a numerical procedure for checking the existence of such a basis in a given generalized spline space.

All the constructions presented here have been devised keeping in mind also the importance of application and implementation, and of the related requirements that numerical procedures must satisfy, in particular in the CAD context.

Keywords: computer-aided design, subdivision surfaces, local non-uniform spline interpolation, interpolating surfaces, arbitrary topology, generalized spline spaces, B-spline basis.

Nuove strategie per la costruzione di curve e superfici a topologia arbitraria per la progettazione

Michele Antonelli

antonelli.miki@gmail.com

Gennaio 2015

Sommario

Questa tesi presenta alcune nuove costruzioni per curve e superfici a topologia arbitraria nel contesto della modellazione geometrica. In particolare, riguarda principalmente tre argomenti strettamente collegati tra loro che sono di interesse sia nella ricerca teorica sia in quella applicata: le superfici di suddivisione, l'interpolazione locale non-uniforme (nei casi univariato e bivariato), e gli spazi di spline generalizzate.

Nello specifico, descriviamo una strategia per l'integrazione di superfici di suddivisione in sistemi di progettazione assistita dal calcolatore e forniamo degli esempi per mostrare l'efficacia della sua implementazione.

Inoltre, presentiamo un metodo per la costruzione di interpolanti univariati polinomiali a tratti, non-uniformi, a supporto locale e che hanno grado minimo rispetto agli altri parametri di progettazione prescritti (come l'ampiezza del supporto, l'ordine di continuità e l'ordine di approssimazione).

Sempre nel contesto dell'interpolazione locale non-uniforme, ma nel caso di superfici, introduciamo una nuova strategia di parametrizzazione che, insieme a una opportuna tecnica di patching, ci permette di definire superfici composite che interpolano mesh o network di curve a topologia arbitraria e che soddisfano i requisiti di regolarità e di qualità estetica di forma solitamente richiesti nell'ambito della modellazione CAD.

Infine, nel contesto delle spline generalizzate, proponiamo un approccio per la costruzione della base (B-spline) ottimale, normalizzata, totalmente positiva, riconosciuta come la miglior base di rappresentazione ai fini della progettazione. In aggiunta, forniamo una procedura numerica per controllare l'esistenza di una tale base in un dato spazio di spline generalizzate.

Tutte le costruzioni qui presentate sono state ideate tenendo in considerazione anche l'importanza delle applicazioni e dell'implementazione, e dei relativi requisiti che le procedure numeriche devono soddisfare, in particolare nel contesto CAD.

Parole chiave: progettazione assistita dal calcolatore, superfici di suddivisione, interpolazione spline locale non-uniforme, superfici di interpolazione, topologia arbitraria, spazi di spline generalizzate, base B-spline.

Contents

List of figures

List of tables

Acknowledgements

I would like to express my gratitude to my advisor Prof. Giulio Casciola, for the inspiration and the encouragement he has provided throughout my doctoral studies. I wish to thank also Dr. Carolina Vittoria Beccari, for her expertise and the patient guidance she has given to me during these years, and Prof. Lucia Romani. It has been a pleasure working with such talented researchers.

Furthermore, I want to thank Prof. Mike Neamtu for reviewing this work.

Finally, my deepest gratitude goes to my parents Mara and Giovanni and to my brother Stefano, for their unconditional support.

Chapter 1

Introduction

The research activity that resulted in this dissertation falls within the framework of the research in Numerical Analysis concerning the theory of approximation, and, more specifically, numerical methods for multivariate data approximation and interpolation, and within the context of Computer-Aided Geometric Design. It deals mainly with three intimately connected topics that are of interest in both theoretical and applied research: subdivision schemes, non-uniform local integration into the system was achieved by introducing a new type of geometric description that exploits a heterogeneous boundary representation of a solid CAD model. In this way, the new type of representation automatically inherits any pre-existing CAD tool, and it can interact in a natural way with the other geometric descriptions supported by the system. Moreover, the critical behavior of subdivision surfaces in the neighborhood of extraordinary points was overcome by exploiting a local modification of the limit surface so as to tune the analytic properties without affecting its geometric shape. This method was inspired by the polynomial blending approach proposed independently by [Lev06b] and [Zor06], which we extended in some aspects and generalized to multipatch surfaces evaluable at constant time at arbitrary parameter values.

Among the interpolatory subdivision schemes, only the version of the Catmull-Clark algorithm by [HKD93], together with Stam's technique, is capable of generating exactly evaluable limit surfaces interpolating the initial control polyhedron (or mesh). The main drawback of this method is that it requires the solution of a global linear system to determine the control points of the surface and a post-processing optimization step to fair its shape. Some variations of the Catmull-Clark interpolatory scheme (e.g., [MMN07]) propose to improve computational cost and surface quality exploiting the so-called *progressive iterative approximation* or *progressive interpolation* technique, which consist in constructing the interpolating surface by a sequential approximation of its polygon control. However, these techniques are still inherently global, in that, as soon as a vertex changes its location, the entire control polygon must be updated in order to keep the interpolation property.

In a work by [BCR13b], non-tensor-product subdivision schemes interpolating regular grids of control points are defined by exploiting a class of univariate C^1 interpolating four-point (hence local) subdivision schemes, in order to generate C^1 limit surfaces with a better behavior than the well-established tensor-product subdivision and spline surfaces. In particular, the method considers a regular grid of points (i.e., with four edges incident at each vertex), where each polygonal section is parameterized independently of the others in a non-uniform way. From this structure, a family of nested curve networks is constructed by an iterative refinement process, in which each successively generated network is denser and contains the previous one. In the limit, the process converges to a C^1 surface, and the curves of the initial network are immersed in this surface and retain their original parameterization. The approach generates surfaces of higher quality compared to the classical non-uniform tensor-product splines or subdivision schemes. The downside is that, because the limit surface is generated by a recursive process, it cannot be described in parametric form or exactly evaluated at arbitrary points.

1.2 Interpolatory methods

Interpolatory methods for curve and surface modeling have the property that the designed curve or surface interpolates a given set of data points. This feature is relevant and often desirable in many contexts: for example, in free-form and interactive design it provides an intuitive outline of the final shape of the object, and in engineering applications it allows to avoid the "shrinkage" effect that is typical of approximating methods and makes it easy to enforce interpolation constraints. The main drawback is the possible presence of artifacts like unpredictable ondulations in the curve/surface which may compromise the overall quality of the shape. In this regard, it has been recently shown ([Lee89, KY06, Flo08, YSK11, BCR11a, BCR11b]) that in spline interpolation these undesirable effects may disappear completely by adopting a suitable non-uniform parameterization (such as the centripetal or chordal), resulting in interpolants of much better quality than their uniform counterparts. In addition, we emphasize the fact that an interpolant should have a piecewise (spline) nature and be featured by the property of locality, i.e., built exploiting compactly supported fundamental functions, so that each piece or patch depends only on a limited (and possibly low) number of data.

Concerning non-uniform interpolation, we started by addressing the design of interpolating curves. In particular, we considered the constructive framework for designing families of local non-uniform spline interpolants featured by given continuity order, polynomial reproduction degree, and support width, which was presented and analyzed by [BCR13a]. The outcome of extending that framework and exploiting it in order to obtain local polynomial spline interpolants of non-uniformly spaced data which are characterized by having minimum degree is detailed in [ABC14a] and presented here in Chapter 3.

With regard to the construction of free-form objects from arbitrary 3D meshes, the non-uniform parameterization principle has been a subject of interest especially in the area of approximation ([CADS09, Cas10, Cas11, KSD13, MRF06, MFR$^+$10, SZSS98]), with a view to establishing high-quality models compatible and comparable with the non-uniform B-spline standard. However, even in the context of interpolation, its potential usefulness is easily understood. In fact, it is clear that a high-quality surface must have high-quality section curves. Therefore, intuitively, we can think of generating a good-quality surface by interpolating a set of good-quality section curves. The benefits of this approach are outlined in [BCR13b].

Motivated by these results, our objective was to provide a method for generating an interpolatory parametric surface of high quality and sufficient regularity from an arbitrary quadrilateral mesh, which can be evaluated exactly at arbitrary points in the parametric domain.

In particular, in Chapter 4 we devise an approach for the representation of a free-form object having complex shape by exploiting surfaces that interpolate the vertices of a given control mesh composed of quadrilateral faces and whose vertices can have arbitrary connectivity. Moreover, by means of a suitable extension, our technique allows for interpolating data that consist of a network of curves, in which, similarly as before, an arbitrary number of curves can meet at each vertex. In both cases, the basic requirement is to generate models that satisfy both criteria of analytical nature (regularity of the surface) and aesthetic concepts (shape quality).

Specifically, we propose a two-step construction that, given a quadrilateral mesh with vertices of arbitrary valence (i.e., a vertex can be shared by an arbitrary number of edges), firstly generates a network of high-quality curves interpolating the mesh vertices (and satisfying G^1 or G^2 compatibility conditions at the intersections), and then builds a composite surface that interpolates the curves of the network with a 1-1 association between mesh faces and surface patches. In addition, if a suitable network of parametric curves is given, a direct generalization of our approach allows generating a surface that interpolates the network, regardless of the nature of the curves (polynomial, trigonometric, etc.). Now, let us outline the salient aspects of this technique.

During the construction of the curve network, each curve is independently parameterized by the centripetal or chordal parameterization, in such a way to optimize its quality. The curves are then represented by piecewise-polynomial functions and are defined in a completely local way exploiting a class of compactly supported fundamental functions having the desired continuity (for the construction, see [BCR13a]), which turns out to be particularly convenient due to its property of a very limited support width. In this way, the portion of curve between two mesh vertices depends only on few adjacent points belonging to a polygonal section of the mesh (which does not necessarily need to be planar). In the neighborhood of valence-4 vertices, the network does not have to meet any particular requirement for compatibility. In the neighborhood of *extraordinary vertices* (i.e., of valence other than 4), we construct the network in order to satisfy compatibility constraints of G^1 and G^2 type.

In the second stage, the curve network serves as a skeleton from which we build a smooth and aesthetically pleasing surface by suitably blending the curves. To this aim, two innovative ingredients are introduced: the first is to use as blending functions a class of non-uniform, locally supported fundamental functions (see again [BCR13a]); the second consists in a suitable strategy of local parameterization. In this way the curves become the boundaries of the patches of a composite surface, where each parametric patch corresponds to a face of the mesh and depends on only one or two rings of vertices surrounding the face. Moreover, inspired by the technique by [BCR13b], we introduce a novel parameterization strategy, called *augmented parameterization*, which does not require that opposite boundaries of a face have the same parameterization. As a result, the curves of the network are embedded in the surface, without being approximated or altered in any way, and retain their original parameterization and consequently their good shape quality. This is a major difference with respect to the classical approaches based on tensor-product or Gordon surfaces, where averaging the parameterization of parallel section curves is necessary. In particular, it is worth pointing out that the classical tensor-product parameterization, albeit non-uniform, in many cases is not flexible enough to generate surfaces of good quality.

Moreover, if a mesh includes isolated non-quadrilateral faces, the local nature of our method makes it possible to patch all the quadrilateral faces. After this operation, non-quadrilateral faces can be handled via hole-filling techniques, extrapolating the necessary information from the adjacent patches. In this context, [VRS11, VSR12] propose a method for n-sided holes, which takes into account the individual lengths of the sides of the hole, a reference to the concept of non-uniform parameterization that is central to our approach. Therefore we suggest that this technique could effectively complement our method, to patch possible isolated n-sided faces.

Another possible approach for handling non-quadrilateral faces consists in refining the initial mesh, either globally or adaptively, through an interpolatory subdivision scheme, so as to split each n-sided face into n new quadrilateral faces. In this regard, as already mentioned, interpolating subdivision schemes do not admit an analytical representation of the limit surface or an exact evaluation algorithm. However, recently much research has been devoted to the study of non-uniform subdivision schemes derived from spline interpolants ([BCR11a, BCR11b, KP13]) and their bivariate generalization ([BCR13b]). By applying our patching approach starting from the same classes of splines related to a subdivision scheme, we could therefore approximate the limit surface, so as to provide a parametric representation. The construction presented in Chapter 4 lays the foundation for the study of this relevant topic, which is among our future research objectives.

1.3 Generalized splines

A further topic that we investigated concerns the theory of generalized splines, whose aim is to extend the theory of polynomial splines in order to include also non-polynomial spaces (e.g., spaces of trigonometric, hyperbolic, or mixed-type functions).

In CAGD the standard representation of curves and surfaces consists in expressing them as linear combinations of basis functions, in which the coefficients are called control points. To be useful as a tool for design, the basis functions must form an *optimal normalized totally positive basis* (ONTP), which means that they must possess some important properties: nonnegativity, partition of unity (i.e., they sum up to 1), convex-hull property, affine invariance, endpoint interpolation, variation diminishing property, total positivity, and the existence of knot insertion algorithms. In fact, such properties ensure that the designer is provided with an intuitive control on the shape of the object during interactive modeling.

In Chapter 5 we show how, considering spline spaces built upon Quasi Extended Chebyshev spaces (QEC-spaces for short) that contain constants and such that the spaces obtained by differentiation are QEC-spaces as well, we can formulate a novel and general approach to construct the ONTP basis, if the spline space admits one. To this aim, we recall and formalize the idea of *transition function* that was introduced in [BCR13a] and that, in a sense, can be dated back to the basis functions initially used by Bézier for curve design (see, e.g., [Rab02]). We exploit the notion of transition function to explicitly construct a basis that coincides with the ONTP one, if it exists.

Moreover, we show that a system of weight functions can be computed in terms of transition functions and, investigating the connection between the existence of weight functions and the existence of the ONTP basis, we provide a criterion for determining whether an arbitrary generalized spline space admits the ONTP basis.

In addition, we provide some relevant algorithms for the computation and modeling in spaces of generalized splines that are of interest from the point of view of applications.

Finally, Chapter 6 collects some concluding remarks.

We remind that Chapters 2 and 3 reflect work that has already been published (in [ABC$^+$13] and [ABC14a], respectively), while Chapters 4 and 5 contain original, unpublished material.

Chapter 2

Subdivision surfaces integrated in a CAD system

This chapter discusses the problem of the integration of subdivision surfaces in computer-aided design (CAD) systems for modeling, engineering, and manufacturing purposes, and presents an approach that has proven to be effective in achieving such an integration.

In particular, the main roadblock that has limited the usage of subdivision surfaces in CAD systems is the lack of quality and precision that a geometric model must achieve for being suitable for the engineering and manufacturing phases of the production process. The second roadblock concerns the integration into the modeling workflow, which, for engineering purposes, means providing a precise and controlled way of defining and editing models possibly composed of different geometric representations. This chapter, together with the recent paper [ABC$^+$13], documents the experience in the context of a European project whose goal was the integration of subdivision surfaces in a commercial CAD software. To this aim, a new CAD system paradigm with an extensible geometric kernel is introduced, where any new shape description can be integrated through the two successive steps of parameterization and evaluation, and a hybrid boundary representation is used to easily model different kinds of shapes. In this way, the newly introduced geometric description automatically inherits any pre-existing CAD functionality and tools, and it can interact in a natural way with the other geometric representations supported by the CAD system.

To overcome the irregular behavior of subdivision surfaces in the neighborhood of the so-called extraordinary points, the limit surface of the subdivision scheme is locally modified so as to tune the analytic properties without affecting its geometric shape. Such a correction is inspired by a polynomial blending approach [Lev06b, Zor06], which generates multipatch surfaces evaluable at constant time at arbitrary parameter values.

Some modeling examples will demonstrate the benefits of the achieved integration, and some tests will confirm the effectiveness of the proposed local correction method.

2.1 Introduction

Subdivision surfaces have a long history, being studied for more than 30 years now. Since the first time they were used in Pixar's movie "Geri's game", they had a big success in entertainment industry and they are now widely supported in nearly all modern modeling programs for graphic applications. Their flexibility and the fact that some subdivision surfaces represent a superset of the standard Non-Uniform Rational B-Spline (NURBS) representation (see, e.g., [CADS09, SZSS98]) easily suggests that they can be the future description form of all geometric data. However, after all these years, their use in CAD systems is still negligible. This is partly due to the fact that subdivision surfaces do not have a closed-form representation, which means that most subdivision schemes can not be parameterized and evaluated exactly. As pointed out in [Ma05], the lack of closed-form evaluation restricts the class of subdivision methods suitable for CAD to schemes that generalize the spline representation to arbitrary topology, namely those of Catmull-Clark [CC78], Doo-Sabin [DS78], and Loop [Loo87]. Key aspects of these schemes, such as the smoothness of the limit surface [DS78, Rei95, Pra98], the curvature behavior [SDHI03, PR04], and the nature and kinds of surface artifacts [KPR04, SB03], have been widely studied in the literature. The state-of-the-art also provides several proposals for algorithms, mainly based on Catmull-Clark surfaces, that may be useful in various CAD-related applications, such as surface fitting [MZ02], reverse engineering [MZ00, BFL$^+$10] and curve lofting [Nas01, SWZ04]. However, despite such a large amount of effort, the quality achieved by current subdivision schemes is still far from the requirements that a geometric model must satisfy for being suitable for the engineering and manufacturing phases of design. In particular, subdivision surfaces manifest a critical behavior in the neighborhood of the so-called extraordinary vertices, which is not a punctual deficiency; actually, it involves a whole region around each extraordinary vertex, making it difficult to apply most of the CAD tools. This has confined their application in CAD merely to the phase of conceptual design, where the precision and quality of the shape is not extremely important.

Subdivision surfaces have already been tested in some CAD systems (CATIA, Creo), while other softwares (Rhinoceros, SolidWorks) have introduced plug-in modules based on T-splines [SZBN03], that show the same lack of precision and quality of subdivision methods around extraordinary vertices. None of the existing systems treats subdivision surfaces (nor T-splines) as a truly integrated geometric primitive. This means that a subdivision model needs to be converted to NURBS to interact with the rest of the CAD system.

This chapter documents part of the work that has been undertaken within the European Eurostars Project NIIT4CAD under the coordination of think3 company. One of the aims of the project was to overcome the traditional approach to surface and solid modeling of current 3D CAD systems by introducing new methodologies based on subdivision surfaces which, overcoming the mentioned roadblocks, could exploit their potentialities to obtain high-quality geometrical model suited for engineering and manufacturing purposes. A prototype of software module for subdivision surfaces for CAD has been integrated in the geometric kernel of the ThinkDesign commercial CAD system, together with a module for fast design and editing of meshes for subdivisions.

The contribution of the work presented in this chapter is twofold. We describe the new framework which guarantees a full integration of subdivision surfaces in a CAD system, and we introduce a correction method to improve the quality issues in the subdivision representation, so as to achieve the accuracy required by CAD applications. The overall objective is to provide a mathematical representation based on subdivision surfaces that achieves the desired accuracy and is integrated with the other representations present in the CAD system, e.g., NURBS and analytical shapes. This lays the groundwork for a new CAD paradigm for hybrid modeling with NURBS/subdivision/analytical geometries, in which all the geometric representations may coexist and interact, and all the functionalities of the CAD system are automatically inherited.

The realized integration is focused on Catmull-Clark subdivision surfaces [CC78], which is the most popular choice of all modern CAD systems. This is, in a way, a natural choice both because Catmull-Clark surfaces represent a generalization of bicubic B-splines, and because they provide an explicit parametric expression for the limit surface [Ma05]. However, the gen-

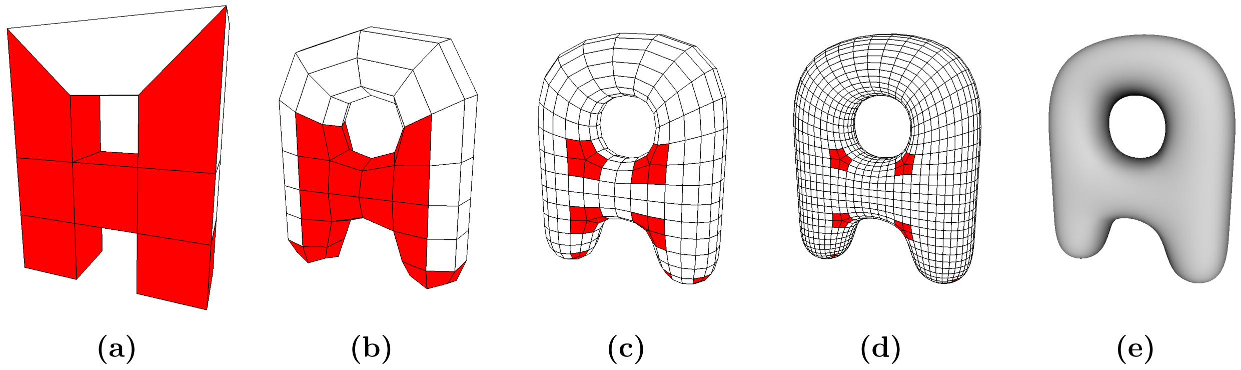


Figure 2.1: (a) Control mesh, (b)–(d) some steps of the Catmull-Clark subdivision algorithm, and (e) related limit surface. In (a)–(d) faces containing at least one extraordinary vertex and thus corresponding to non-spline patches are highlighted in red color.

erality of the approach makes it easily extendable to all subdivision schemes based on splines. To achieve the quality required in CAD, where necessary, we have introduced a local correction inspired by the idea of polynomial blending proposed in [Lev06b, Zor06]. The approach consists in blending, in the neighborhood of extraordinary points, the limit surface with a proper polynomial approximant. This allows tuning the analytic properties of the surface, without affecting its geometric shape. In [Lev06b, Zor06] the authors assume to work on a discretization of the surface, obtained by applying a finite number of refinement steps to the mesh of the initial control points. As a consequence, after the correction, the limit surface is evaluated only at the discrete set of points generated by the process of iterative subdivision. To suit our needs, the original blending idea is here further developed and generalized, assuming to work with a continuous representation of the surface which is evaluable at arbitrary points of the domain.

The remainder of the chapter is organized as follows. Section 2.1.1 briefly presents some basic notions on subdivision surfaces. Section 2.2 describes the CAD system paradigm on which our integration approach relies. Section 2.3 presents our approach for the integration of subdivision surface primitives into a CAD system, which is based on the B-rep representation and a suitable parameterization method, described in Section 2.3.1. Section 2.4 is devoted to the local correction of subdivision surfaces around extraordinary vertices. Section 2.5 discusses the most relevant computational issues to deal with when implementing exact evaluation of subdivision surfaces in a CAD context. Section 2.6 illustrates some modeling examples aimed at demonstrating the potentialities of the proposed integration, and some tests which confirm the effectiveness of the proposed local correction patching method. Concluding remarks can be found in Section 2.7.

2.1.1 Subdivision basics

A subdivision scheme generates a smooth surface as the limit of a sequence of successive refinements of an initial configuration of vertices, edges and faces with connectivity relations, called *control mesh*. To illustrate the process, Figures 2.1(b)–2.1(d) show some refinement steps of the Catmull-Clark scheme applied to the control mesh of Figure 2.1(a), while in Figure 2.1(e) the associated limit surface is shown.

In the control mesh, an arbitrary number of edges can be incident at each vertex (called the *valence* of the vertex), as well as an arbitrary number of edges can define a face, called an n-sided face. In particular, a vertex is called *extraordinary* if its valence is different from the

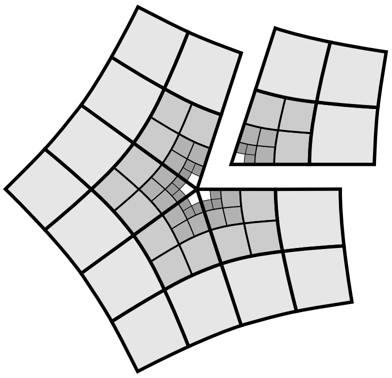


Figure 2.2: Limit surface structure around an extraordinary vertex of valence $n = 5$. Each of the nested spline rings is composed by n sectors and each sector in turn of three spline patches in an L-shaped configuration.

regular configuration, which for Catmull-Clark is 4, and it is called *regular* otherwise.

The potential of subdivision surfaces with respect to other forms of representation (e.g., NURBS) is the possibility of handling objects of arbitrary topological genus. Essentially, like NURBS, subdivision surfaces describe a geometric model by means of control points. However, while in a NURBS the control points form a regular rectangular grid, subdivision handles a generic mesh with arbitrary topology. The benefit is that, when moving the points of the control mesh, the geometric object automatically stays connected and the faces remain tangent- and curvature-continuous to each other.

The successive refinement procedure, which characterizes the subdivision surface, provides a representation that is inherently discrete. However, a parametric representation is admitted by a special family of subdivision schemes that, away from extraordinary vertices (i.e., in regular regions), reproduce a known analytic representation (which hereinafter can be assumed in spline form). In this setting, the limit surface can be represented at a given parametric point (u, v) belonging to regular or extraordinary regions as a finite linear combination

$$\boldsymbol{S}(u, v) = \sum_i \boldsymbol{p}_i b_i(u, v), \tag{2.1}$$

where $\boldsymbol{p}_i \in \mathbb{R}^3$ are the vertices of the input control mesh and $b_i(u, v)$ are suitable compactly supported basis functions.

If the scheme has an analytic representation underneath, then away from extraordinary vertices (2.1) represents a tensor-product surface patch. In this case, for Catmull-Clark scheme the basis functions are the bicubic tensor-product B-splines [PT95].

In proximity of extraordinary vertices the number of basis functions depends on the valence. For Catmull-Clark scheme these functions can be evaluated by well-known algorithms, as described in [Sta98, Yam01, LC06], and by their several variants to non-closed surface evaluation in the vicinity of the boundary [BLZ00, SES04, LB07]. These methods are capable of evaluating the surface at arbitrary parameters in the vicinity of one extraordinary vertex based on the idea illustrated in Figure 2.2. Each patch is parameterized by a collection of quadrilateral domains arranged in nested L-shaped groups which progressively shrink towards the extraordinary vertex; on each of these domains the surface is represented as a bicubic tensor-product B-spline.

2.2 The CAD system paradigm

In the following we discuss the fundamental features required for a CAD system to be able to integrate the subdivision surface representation into its geometric kernel. In particular, the CAD system must satisfy the following requirements:

- "parameterization and evaluation" paradigm;
- B-rep representation;
- hybrid geometric descriptions.

A major challenge for the integration of a new geometric primitive within a CAD system is represented by the problem of including the primitive in the geometric engine of the system, the so-called *geometric kernel*.

In many of current commercial CAD systems the structure of the geometric kernel is quite similar: it provides a set of geometric representations and a set of tools which operate on them. Curves and surfaces are represented in parametric form, while for solids the boundary representation (B-rep) is mostly used. In the B-rep a geometric model is described through its boundary surfaces, which, in case of a solid model, represent the boundary between solid and non-solid. Such a boundary is constituted by one or more connected components, called *shells*. In turn, each shell is represented as a collection of surface patches, called B-rep faces. Faces are chosen so that the geometric form of each face can be represented as a single parametric function. To describe each shell, a B-rep stores two types of information: topological information, namely the connectivity relations between faces, and the geometric representation of the surface patch corresponding to each face.

The various available CAD tools include intersections, projections, Boolean operations, offsets, fillets, and many others. Some geometric kernels rely on the NURBS representation only, while more advanced ones also allow different representations, such as planes, cylinders and quadrics, which provide greater precision at lower storage cost. Since the CAD system must support different geometric representations, the B-rep modeling system underneath should adopt a hybrid description of the geometry, allowing the coexistence of faces with different forms of geometric representation.

The proposed approach is based on a CAD system paradigm with an extensible geometric kernel, in which any geometric description can be integrated by simply implementing the related evaluation algorithm that acts as an interface with the entire system. This paradigm of extensibility requires that any geometric representation be parameterizable and there must be a known evaluation method that associates with each point in the parametric domain a surface value and derivatives up to second order. This makes all the functionalities of the CAD system automatically inherited by any newly introduced type of representation, which is therefore indistinguishable from others already enclosed in the system. In this way, pre-existing operators and tools can be easily and directly applied to any hybrid representation. This defines our concept of full integration of the geometric representations in the CAD system, although, obviously, a suitable tuning of some algorithms may be required to optimize the output quality.

This paradigm reflects the structure of the geometric kernel of the ThinkDesign system (developed by think3), used for testing the proposed approach. Moreover, ThinkDesign supports hybrid modeling where the B-rep has been extended to allow special, non-solid model types called non-manifold models, which are used to represent thin-plate objects and integrate surface modeling into a solid modeling environment. This makes the B-rep a more flexible descriptive model and provides the CAD system with a much richer set of operations.

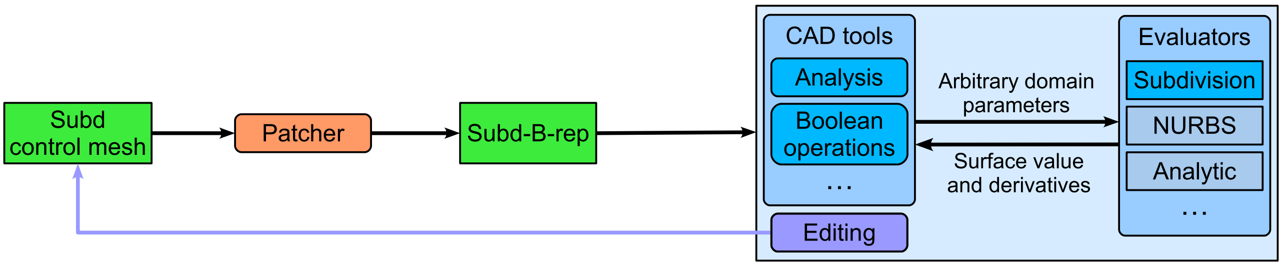


Figure 2.3: Workflow for the creation and editing of a Subd-B-rep.

2.3 Subdivision B-rep

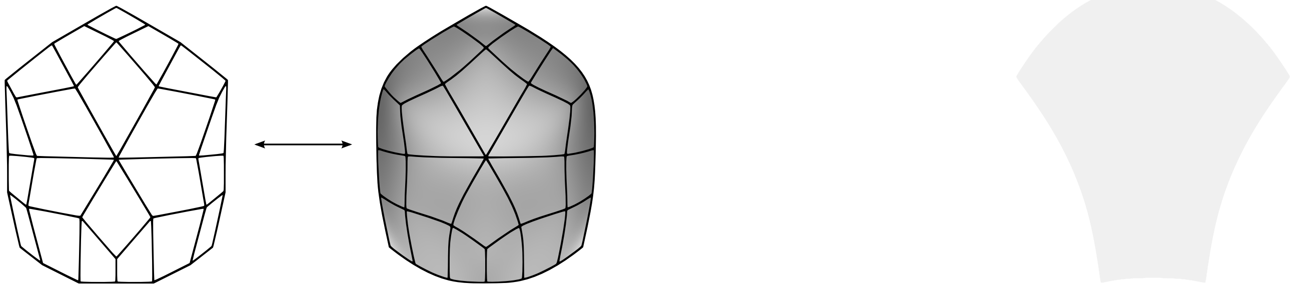
This section describes how a subdivision entity has been integrated in the previously described CAD system. A subdivision surface is a single entity describing an arbitrary topology geometric model by means of its control mesh. However, a subdivision surface can be exactly evaluated only by associating with each quadrilateral face of the control mesh a parametric domain. Thus the most natural way to handle this structure is to store the information on the geometric model in a B-rep, where the control mesh is used to derive the B-rep topological information. To this end, we introduce the following B-rep characterization for subdivision surfaces.

Definition 2.1 (Subd-B-rep)**.** A *Subd-B-rep* represents a B-rep geometric model in which each B-rep face is a patch of a subdivision surface associated with a rectangular parametric domain and the B-rep topology is defined by the subdivision control mesh.

Figure 2.3 shows the workflow for the creation and editing of a subdivision entity. The faces which form a Subd-B-rep are extracted from the control mesh of the subdivision surface through a patcher function. A suitable evaluator for subdivision entities is used by the various CAD algorithms to interrogate the Subd-B-rep and compute surface value and derivatives of its faces at arbitrary domain parameters.
Exploiting the classical handling of B-rep, a Subd-B-rep can interact naturally with other entities in the system. For example, the result of operations of solid composition is a B-rep whose faces can have hybrid geometric nature, e.g., they can be either NURBS, analytical, or subdivision surfaces patches. Such B-rep will be editable while maintaining the hybrid nature of its faces, and any CAD tool will operate on the hybrid form. In this case the workflow in Figure 2.3 applies to those faces of the B-rep model that are of subdivision type.

In the following we illustrate how the topological and geometric information that characterizes a Subd-B-rep is derived by the patcher function starting from an initial control mesh. The leading idea is that the Subd-B-rep should maintain an intuitive association (possibly 1-1) between the control mesh faces and the B-rep faces. Since the topological information is strictly related to the specific subdivision scheme considered, we will specialize the following description to the particular Catmull-Clark subdivision method. In this case, if the faces of the control mesh are quadrilateral faces, the 1-1 correspondence is obtained in a natural way as illustrated in Figure 2.4(a).

Topology. The topology is described by adjacency relations between topological entities. In order to generate the Subd-B-rep, such information can be inferred from the control mesh of the subdivision surface. The control mesh is a (consistent) collection of vertices, edges and faces with connectivity relations in which each face can be formed by an arbitrary number of edges, and each edge is adjacent to exactly two faces. In our B-rep construction, a quadrilateral face



(a) Mesh with all quadrilateral faces.

(b) Mesh with some non-quadrilateral faces; loop of 6 edges highlighted in the corresponding Subd-B-rep.

Figure 2.4: Control mesh of a subdivision surface and related Subd-B-rep.

of the control mesh defines a single face of the Subd-B-rep (see, e.g., Figure 2.4(a)). Conversely, an n-sided face in the control mesh (with $n \neq 4$) cannot be represented as a unique parametric surface patch on a quadrilateral domain. Thus we apply a refinement step of the Catmull-Clark scheme which splits the face into n quadrilateral faces, each of which becomes an independent face in the construction of the Subd-B-rep (see Figure 2.4(b)).

Besides adjacency relations, the B-rep typically stores the loops, i.e., circular sequences of edges that form the boundaries of a B-rep face. In particular, in the Subd-B-rep each face contains one single loop which can be either regular when it consists of 4 edges (Figure 2.4(a)), or extraordinary (more than 4 edges) in case of quadrilateral mesh faces adjacent to non-quadrilateral ones. In the example shown in Figure 2.4(b), an extraordinary loop of 6 edges that defines a quadrilateral face is highlighted in red color. This determines a T-shaped configuration of B-rep edges, which is easily handled by any standard B-rep data structure. In the worst case, corresponding to a quadrilateral mesh face adjacent to 4 non-quadrilateral faces, the loop is formed by 8 edges.

We remark that the consistency of the control mesh on which the B-rep is built automatically leads to the consistency of the resulting Subd-B-rep.

Geometry. We describe the geometry associated with each face of the Subd-B-rep as a parametric surface patch over the domain $Q \coloneqq [0,1]^2$, called *base domain.* This patch is the limit surface of the subdivision scheme which is parameterized and evaluated as described in the following subsection.

The notation used from this point on is summarized in Table 2.1.

2.3.1 Subd-B-rep parameterization

In the proposed approach, the limit surface $\boldsymbol{S}$ of the subdivision scheme, associated with a given control mesh, is represented as a parametric multipatch surface in which each patch $\boldsymbol{S}_i$ is associated either with a quadrilateral face of the control mesh or with a quadrilateral face generated by refining an n-sided face. Each patch $\boldsymbol{S}_i$ is parameterized on the base domain Q through a parameterization function

$$\psi_{\boldsymbol{S}_i} : Q \longrightarrow \boldsymbol{S}_i.$$

In the following, we denote by $\boldsymbol{q}_\ell, \ell = 0, \ldots, 3$, the vertices of Q indexed in counterclockwise order from $(0,0)$. The function ψ is the parameterization in which the limit surface is evaluated

σ_h	uniform scaling by a factor h
ρ_a	counterclockwise rotation by an angle a about the origin
$\tau_{\boldsymbol{v}}$	translation by a vector $\boldsymbol{v}$
$\psi_{\boldsymbol{Z}}$	parameterization of a Catmull-Clark surface $\boldsymbol{Z}$
$\boldsymbol{Z}_i$	ith patch of the parametric multipatch surface $\boldsymbol{Z}$
Q	base domain $[0,1]^2$
$\boldsymbol{q}_\ell$	vertices of Q, in counterclockwise order starting from $(0,0)$
Q_ℓ	sub-square of Q that cointains $\boldsymbol{q}_\ell$
Q^h	rectangular domain $[0,h]^2$
(u,v)	parameters of a surface patch, in Q
$\boldsymbol{S}$	Catmull-Clark limit parametric multipatch surface
$\boldsymbol{P}$	polynomial surface
$\boldsymbol{S}^*$	blended surface
w	blending function
$\boldsymbol{K}_0^{[n]}$	sector of the characteristic map of valence n
K_n	local domain parameterizing the neighborhood of an extraordinary vertex
(s,t)	parameters in K_n
D_n	blending region $\subset K_n$
$\lambda^{[n]}$	subdominant eigenvalue of the local subdivision matrix for valence n
$\boldsymbol{p}_k$	approximation points for the construction of $\boldsymbol{P}$

Table 2.1: The main symbols used in this chapter.

and, as recalled in Section 2.1.1, it is known in two situations, namely for regular faces and faces that contain only one extraordinary vertex.

To fit the definition of Subd-B-rep, it is necessary that each B-rep face define a unique surface patch parameterized over the domain Q. Thus, faces of the control mesh that contain more than one extraordinary vertex and non-quadrilateral faces must be suitably treated so as to satisfy this requirement. To this aim, we distinguish three different types of extraordinary patches, each of which corresponds to a different parameterization strategy, described in the following and illustrated in Figure 2.5.

(a) *Quadrilateral face with only one extraordinary vertex* (Figure 2.5(a)): it is mapped to a surface patch parameterized over Q in such a way that

$$\psi_{\boldsymbol{S}_i}(\boldsymbol{q}_0) = \psi_{\boldsymbol{S}_i}(0,0) = \boldsymbol{p}_{\mathrm{ev}},$$

where $\boldsymbol{p}_{\mathrm{ev}}$ is the limit point corresponding to the extraordinary vertex. Patches of type (a) can be evaluated exploiting one of the algorithms cited in Section 2.1.1.

(b) *Quadrilateral face with more than one extraordinary vertex* (Figure 2.5(b)): we define a reparameterization map over Q as follows. The patch $\boldsymbol{S}_i$ is partitioned into the union of 4 sub-patches $\boldsymbol{S}_{i,\ell}$, $\ell = 0,\dots,3$, such that each of them contains at most one extraordinary vertex. Each $\boldsymbol{S}_{i,\ell}$ is associated with a sub-square Q_ℓ of the base domain Q, as illustrated in Figure 2.5(b). To evaluate each sub-patch as in (a) it is necessary to apply beforehand the reparameterization

$$\phi_\ell : Q_\ell \longrightarrow Q$$

defined by

$$\phi_\ell := \sigma_2 \circ \rho^{-1}_{\ell\frac{\pi}{2}} \circ \tau_{\boldsymbol{q}_\ell - \boldsymbol{q}_0}, \tag{2.2}$$

where σ_h is a scaling by a factor h, ρ_a^{-1} is a clockwise rotation around the origin by an angle a, and $\tau_{\boldsymbol{v}}$ is a translation of a vector $\boldsymbol{v}$ (note that all these functions are applied to vectors).

(c) *Quadrilateral face generated by the splitting of an n-sided face*: the patcher function associates with an n-sided face of the control mesh n quadrilateral patches, by refining the face through the Catmull-Clark scheme. Each of them contains at least one n-valent extraordinary vertex generated by the splitting of the face (Figure 2.5(c.1)), and possibly two (Figure 2.5(c.2)). The two situations give rise to the following cases:

(c.1) if the patch contains only one extraordinary vertex, then it is treated as in (a);

(c.2) if the patch contains two extraordinary vertices, then it is treated as in (b).

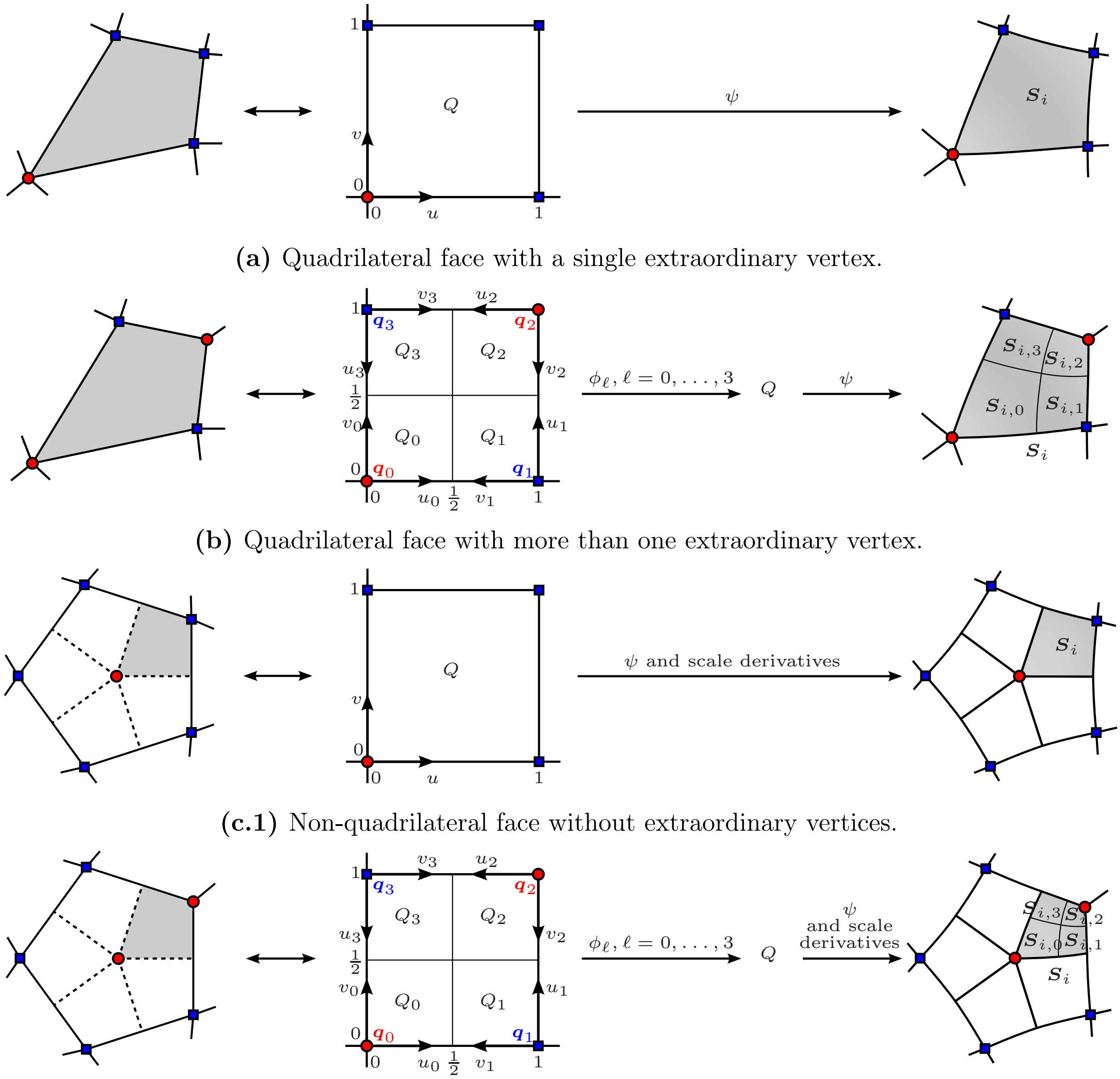
In both sub-cases, since the n quadrilateral patches derive from the refinement of an n-sided face of the control mesh, an additional scaling of their first and second derivatives is achieved respectively by the transformations σ_h and σ_{h^2} with $h = 2$.

We observe that, after the subdivision of an n-sided face, cases (c.1) and (c.2) differ from cases (a) and (b) only for an additional scaling of the parametric domain, which takes into account the fact that faces of type (c.1) and (c.2) have been subdivided one more time with respect to faces of type (a) and (b). So, actually, only the three cases (a), (b) and (c) are strictly necessary, with the two sub-cases of (c) falling into (a) or (b). However, considering all the distinct four cases will simplify the comprehension of the method of local correction described in Section 2.4.

The described parameterization technique allows us to isolate extraordinary points, in such a way that each point $\boldsymbol{p}_{\text{ev}}$ belongs to a rectangular patch $\boldsymbol{S}_i$ or a sub-patch $\boldsymbol{S}_{i,\ell}$, whose remaining vertices are regular.

2.4 Surface tuning around extraordinary vertices

It is well known that Catmull-Clark surfaces exhibit a critical behavior in the neighborhood of extraordinary points. Depending on the variant of the scheme that we consider, the curvature at those points is either undefined [DS78] or zero [PU98]. Although some modifications were proposed to overcome such issue [Sab91, Loo02, GU07, ADS06, SCAD07], available methods guarantee bounded but not necessarily convergent curvature. This phenomenon is common to the entire family of schemes whose limit surface can be represented as a collection of nested spline rings around an extraordinary vertex, namely all parameterizable schemes. In fact, intuitively, since these rings are essentially a deformation of a regular grid, using them to fill in the n-sided hole around an n-valent vertex generates a singularity.
The main reason that prevents the use of subdivision surfaces in CAD applications is that the erratic behavior of curvature is not confined to an isolated point; conversely, it affects a whole area around each extraordinary vertex (see, e.g., [KPR04, SB03, PR08]). This causes many fundamental geometric operations to produce unpredictable results or completely fail.
Another problem concerns the parameterization of the surface. Approaching an extraordinary point, the absolute value of derivatives progressively increases (when valence $n \geqslant 5$) or decreases (when $n = 3$) without bound. Such an undesirable situation is very far from the ideal, in which the parameter follows the distance on the surface, and it causes the instability of many geometric algorithms (e.g., intersection, projection) that rely on approximation of the surface



(a) Quadrilateral face with a single extraordinary vertex.

(b) Quadrilateral face with more than one extraordinary vertex.

(c.1) Non-quadrilateral face without extraordinary vertices.

(c.2) Non-quadrilateral face with at least one extraordinary vertex.

Figure 2.5: Correspondence between the different types of faces of the control mesh and the parametric patches that form the limit surface. Left: face of the initial mesh; center: parametric base domain Q associated with the shaded face; right: patch $\boldsymbol{S}_i$ of the limit surface. Extraordinary vertices are marked by circles, while regular vertices by squares.

by local Taylor expansion.

To address these issues, a possible approach relies on *patching* methods, based on replacing all the subdivision surface patches containing an extraordinary point by another form of representation. Most of these techniques result in a C^1 construction [Pet00] while, to our knowledge, only [LS08] achieves G^2 continuity. However, a drawback of the patching approach is that the modification affects a wide region of the surface, and this partly disqualifies it with respect to our aims. In fact, since the shape of Catmull-Clark surfaces is already satisfactory, we are interested in maintaining their appearance and B-spline nature in the widest possible area, while improving their analytic properties in the smallest neighborhood of the extraordinary points. To this aim, we pursue a *blending* approach [Lev06b, Zor06], which consists in blending the Catmull-Clark and another suitable surface in a small circular neighborhood of the extraordinary point, called *blending region*. This results in C^2 continuity without significantly altering the shape and the almost-everywhere spline nature of the Catmull-Clark method. The novelty of our technique is its capability of generating a parametric surface, evaluable at arbitrary points of the domain.
In Section 2.4.1 we overview the basic blending approach, while in the subsequent sections we present the original aspects of our proposal. We remark that, although here we focus on Catmull-Clark surfaces, the approach is very general and straightforwardly applies to any exactly evaluable subdivision scheme.

2.4.1 Local correction through polynomial blending

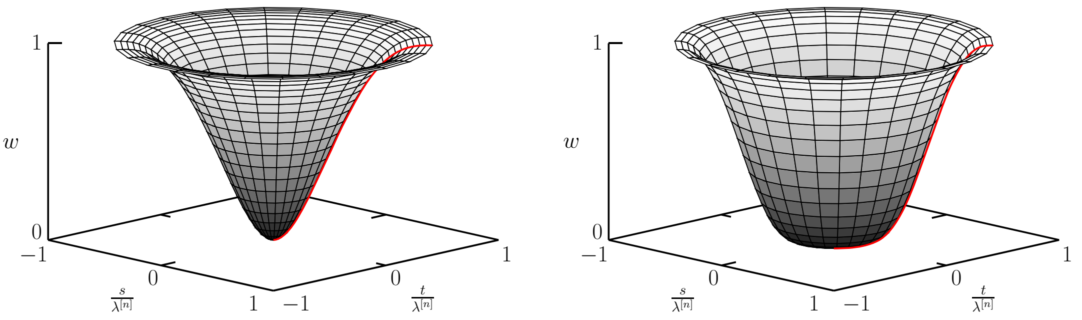
Let $\boldsymbol{p}_{\text{ev}}$ be an extraordinary point of valence n and $\boldsymbol{S}_i$, $i = 0, \dots, n-1$, the n surface patches around it. Then, for each $\boldsymbol{S}_i$, we define on Q a new geometric representation, named *blended surface*, as

$$\boldsymbol{S}_i^* := \begin{cases} w\boldsymbol{S}_i + (1-w)\boldsymbol{P}, & \text{inside the blending region,} \\ \boldsymbol{S}_i, & \text{elsewhere,} \end{cases} \tag{2.3}$$

where w is a bivariate blending function and $\boldsymbol{P}$ is a polynomial surface that approximates an area of the Catmull-Clark surface centered at $\boldsymbol{p}_{\text{ev}}$. The function w must be chosen so as to guarantee that the transition between $\boldsymbol{S}_i^*$ and $\boldsymbol{S}_i$ is C^2-continuous. Figure 2.6 illustrates two blending functions that can be conveniently exploited in our context: the function in Figure 2.6(a) was proposed in [Lev06b], while Figure 2.6(b) shows a function which is zero in a circular neighborhood $\mathcal{I}$ of the origin and whose derivatives vanish up to the second order on both internal and external boundaries of the blending region. It can be easily observed that the latter blending function generates a purely polynomial surface in a neighborhood of the extraordinary vertex, and the blending region reduces to an annulus: in fact, on $\mathcal{I}$, $w \equiv 0$ and thus, from (2.3), $\boldsymbol{S}_i^*$ coincides with $\boldsymbol{P}$.
The blended surface defined in (2.3) requires that locally the patches $\boldsymbol{S}_i$, $i = 0, \dots, n-1$, the polynomial $\boldsymbol{P}$, and the function w be parameterized over a common domain, which is introduced in the following section.

2.4.2 A star-shaped parameterization domain

The proposed construction is based on the *characteristic map* that defines the natural parameterization of the surface generated by the subdivision scheme in the neighborhood of an extraordinary vertex (see, e.g., [Rei95]). This map is defined as the limit surface of the subdivision scheme applied to the 2D mesh whose vertices are the entries of the two subdominant



(a) **(b)**

Figure 2.6: Blending function w.

eigenvectors of the local subdivision matrix[1]. The characteristic map itself can be regarded as a parametric multipatch surface, consisting of n patches $\boldsymbol{K}_i^{[n]}$ of type (a) that, up to proper normalization[2], have rotational symmetry with respect to the extraordinary vertex. For our purposes, without loss of generality, we consider one of these sectors, namely $\boldsymbol{K}_0^{[n]}$, which is represented over the square domain Q by $\psi_{\boldsymbol{K}_0^{[n]}} : Q \longrightarrow \boldsymbol{K}_0^{[n]}$.

In the following we describe how, starting from $\boldsymbol{K}_0^{[n]}$, it is possible to define a star-shaped domain to parameterize the n surface patches $\boldsymbol{S}_i$, $i = 0, \dots, n-1$, in a suitable neighborhood of $\boldsymbol{p}_{\mathrm{ev}}$ that contains the blending region. To this aim we introduce the *star-shaped transformation*

$$\kappa_i := \rho_{i\frac{2\pi}{n}} \circ \psi_{\boldsymbol{K}_0^{[n]}} \circ \sigma_h \circ \phi,$$

where $\psi_{\boldsymbol{K}_0^{[n]}}$ is applied to $Q^h\big|_Q$, with $Q^h := [0, h]^2$, σ_h is a scaling with factor h given by

patch type	(a)	(b)	(c.1)	(c.2)
h	4	2	2	1

, (2.4)

and

$$\phi(u,v) := \begin{cases} (u,v), & \text{if the } i\text{th patch is of type (a) or (c.1)}, \\ \phi_\ell(u,v), & \ell = 0, \dots, 3,\ (u,v) \in Q_\ell,\ \phi_\ell \text{ in (2.2)}, \\ & \text{if the } i\text{th patch is of type (b) or (c.2)}. \end{cases}$$

For a patch of type (a), the star-shaped transformation is illustrated in Figure 2.7. Note that the transformation $\psi_{\boldsymbol{K}_0^{[n]}} \circ \sigma_h \circ \phi$ maps the parametric domain of a patch to $\boldsymbol{K}_0^{[n]}$, which is a fixed sector of the characteristic map. Finally, the rotation $\rho_{i\frac{2\pi}{n}}$ maps $\boldsymbol{K}_0^{[n]}$ to the ith sector of K_n, the global parameterization domain around the extraordinary point defined as follows.

Definition 2.2. For a given extraordinary vertex of valence n we define the *star-shaped parameterization domain* as

$$K_n := \bigcup_{i=0}^{n-1} \left\{ \kappa_i\,(u,v) \;\middle|\; (u,v) \in Q \text{ and } \sigma_h(\phi(u,v)) \in Q \right\}.$$

[1] The local subdivision matrix M_n is such that $P^{(k+1)} = M_n P^{(k)}$, where $P^{(k)}$ are the vertices of the k-times refined mesh.

[2] The normalization is such that $\boldsymbol{K}_0^{[n]}$ is located in the positive quadrant of the real plane and with one edge having endpoints in $(0,0)$ and $(1,0)$.

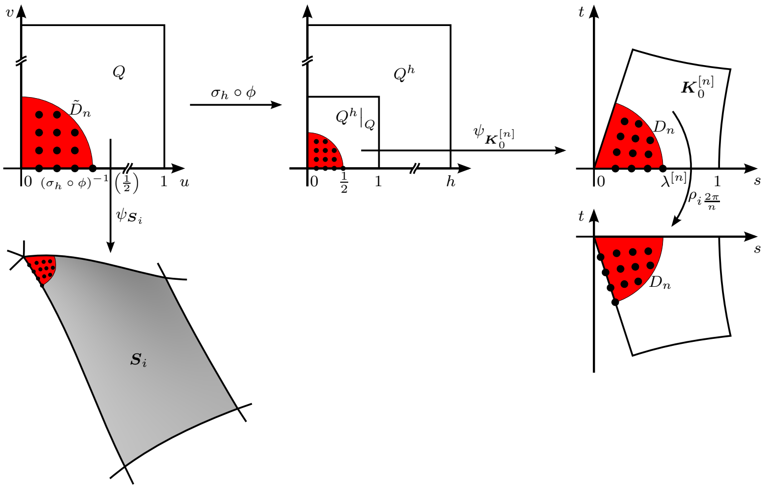


Figure 2.7: A surface value $\boldsymbol{S}_i(u,v)$ is in 1-1 correspondence with a couple of parameters $(s,t) \in \boldsymbol{K}_0^{[n]}$, where (s,t) is obtained by applying to $(u,v) \in Q$ the transformation $\kappa_i = \rho_{i\frac{2\pi}{n}} \circ \psi_{\boldsymbol{K}_0^{[n]}} \circ \sigma_h \circ \phi$. The figure refers to a surface patch of type (a), hence $h = 4$. Dots represent the least squares approximation points.

Remark 2.1. The point $(0,0)$ of the domain K_n is mapped to the extraordinary point shared by the n surface patches $\boldsymbol{S}_i$.

Assuming $(s,t) := \kappa_i\,(u,v)$, inside the domain K_n we define the circular *blending region*

$$D_n := \left\{ (s,t) \in K_n \,\middle|\, \|(s,t)\|_2 \leqslant \lambda^{[n]} \right\},$$

centered at the origin and having radius equal to the subdominant eigenvalue $\lambda^{[n]}$ of the subdivision matrix.

In this way, we can now characterize the blended surface in (2.3) as follows:

$$\boldsymbol{S}_i^*(u,v) = \begin{cases} w(s,t)\boldsymbol{S}_i(u,v) + (1-w(s,t))\boldsymbol{P}(s,t), & (s,t) \in D_n, \\ \boldsymbol{S}_i(u,v), & \text{elsewhere}. \end{cases} \tag{2.5}$$

In the base domain Q of each patch $\boldsymbol{S}_i$ the blending region is represented by

$$\tilde{D}_{n,i} := \left\{ (u,v) \in Q \,\middle|\, \sigma_h(\phi(u,v)) \in Q \text{ and } \|\kappa_i(u,v)\|_2 \leqslant \lambda^{[n]} \right\}.$$

In Figure 2.8 the blending regions around adjacent extraordinary vertices are highlighted in red color. We observe that the radius of the blending region is chosen in such a way that blending regions surrounding extraordinary vertices of the same face are well separated from each other. This guarantees that the spline nature of the limit surface of the Catmull-Clark scheme is altered in quite restricted areas. In particular, as a consequence of the scalings by σ_h and possibly ϕ_ℓ, $\tilde{D}_{n,i}$ is contained inside $Q^{\frac{1}{8}}$ for patch types (a), (b) and $Q^{\frac{1}{4}}$ for (c.1), (c.2).

The first and second partial derivatives of $w\,(s(u,v),t(u,v))$ and of each component P_χ of $\boldsymbol{P}\,(s(u,v),t(u,v))$ with respect to u, v are computed by applying the chain rule. Let f be w or P_χ functions, then:

$$\begin{pmatrix} \partial_u f \\ \partial_v f \end{pmatrix} = J \begin{pmatrix} \partial_s f \\ \partial_t f \end{pmatrix}, \tag{2.6}$$

where

$$J := \begin{pmatrix} \partial_u s & \partial_v s \\ \partial_u t & \partial_v t \end{pmatrix}$$

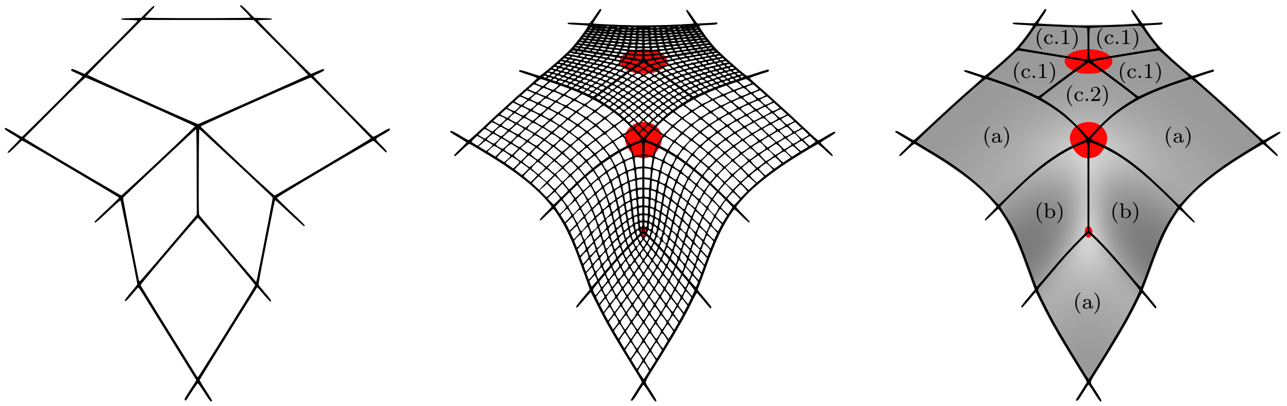


Figure 2.8: Blending regions around adjacent extraordinary vertices. From left to right: control mesh, isoparametric curves corresponding to the same parameter grid for each patch, limit surface with patches of different type (according to the classification in Section 2.3.1). Blending regions are highlighted in red color; in the grey area the surface has a bicubic spline representation.

is the Jacobian matrix of the map $(u, v) \mapsto (s, t)$ defined by the star-shaped transformation κ_i, and

$$\begin{pmatrix} \partial_{uu} f & \partial_{uv} f \\ \partial_{vu} f & \partial_{vv} f \end{pmatrix} = J \begin{pmatrix} \partial_{ss} f & \partial_{st} f \\ \partial_{ts} f & \partial_{tt} f \end{pmatrix} J^T + \begin{pmatrix} \begin{pmatrix} \partial_s f \\ \partial_t f \end{pmatrix}^T \begin{pmatrix} \partial_{uu} s \\ \partial_{uu} t \end{pmatrix} & \begin{pmatrix} \partial_s f \\ \partial_t f \end{pmatrix}^T \begin{pmatrix} \partial_{uv} s \\ \partial_{uv} t \end{pmatrix} \\ \begin{pmatrix} \partial_s f \\ \partial_t f \end{pmatrix}^T \begin{pmatrix} \partial_{vu} s \\ \partial_{vu} t \end{pmatrix} & \begin{pmatrix} \partial_s f \\ \partial_t f \end{pmatrix}^T \begin{pmatrix} \partial_{vv} s \\ \partial_{vv} t \end{pmatrix} \end{pmatrix}, \qquad (2.7)$$

where $\partial_{vu} f = \partial_{uv} f$ and $\partial_{ts} f = \partial_{st} f$ since we are dealing with C^2 functions.

2.4.3 Computation of the approximating polynomial

In principle, the polynomial $\boldsymbol{P}$ can be computed by any suitable approximation method. As pursued in [Lev06b, Zor06], in the following we exploit a least squares approach. Alternatively, an energy minimization approach was explored in [Lev06a] and could be adapted to our setting as well.

Since $\boldsymbol{P}$ is required to interpolate the extraordinary vertex, we represent it in bivariate form of degree d:

$$\boldsymbol{P}(s, t) = \boldsymbol{p}_{\mathrm{ev}} + C\boldsymbol{m}(s, t),$$

where $C = (c_{\chi,j}) \in \mathbb{R}^{3\times n_{\mathrm{c}}}$, with $n_{\mathrm{c}} := \frac{1}{2}(d+1)(d+2) - 1$, is the matrix having as rows the polynomial coefficients of coordinates χ, and the entries of vector $\boldsymbol{m}(s, t) \in \mathbb{R}^{n_{\mathrm{c}}\times 1}$ represent the n_{c} bivariate monomials $s, t, s^2, st, t^2, \ldots$ up to degree at most d.

We assume $d = 3$ for extraordinary vertices of valence $n \geqslant 5$, and $d = 2$ in the case $n = 3$.

To complete the definition of the polynomial, we compute the coefficients $c_{\chi,j}$ by least squares fitting exploiting $n_{\mathrm{p}} := 12n$ points $\boldsymbol{p}_k$ obtained by evaluating the Catmull-Clark limit surface at parameters $(u_k, v_k) \in \tilde{D}_{n,i}$. In particular, for each patch $\boldsymbol{S}_i$ of type (a) or (b) containing $\boldsymbol{p}_{\mathrm{ev}}$, these parameters correspond to the following points in the base domain:

$$\begin{array}{llll} \left(\frac{1}{32}, 0\right), & \left(\frac{1}{16}, 0\right), & \left(\frac{3}{32}, 0\right), & \left(\frac{1}{8}, 0\right), \\ \left(\frac{1}{32}, \frac{1}{32}\right), & \left(\frac{1}{16}, \frac{1}{32}\right), & \left(\frac{3}{32}, \frac{1}{32}\right), & \\ \left(\frac{1}{32}, \frac{1}{16}\right), & \left(\frac{1}{16}, \frac{1}{16}\right), & \left(\frac{3}{32}, \frac{1}{16}\right), & \\ \left(\frac{1}{32}, \frac{3}{32}\right), & \left(\frac{1}{16}, \frac{3}{32}\right). & & \end{array} \qquad (2.8)$$

In Figure 2.7 these points and their image through the star-shaped transformation κ_i are represented by solid black dots. For patches of type (c.1) or (c.2) the parameters associated with the approximation points are those in (2.8), but additionally scaled by a factor 2, as these types of patches derive from the subdivision of a non-quadrilateral face. The parameters in (2.8) have been obtained by evaluating the patch on a grid of equally spaced points in the base domain and selecting the domain coordinates whose image in K_n falls inside the blending region D_n. The same values are used for each valence n.

In the star-shaped domain K_n, the n_{p} points (u_k, v_k) correspond to the points $(s_k, t_k) := \kappa_i\,(u_k, v_k)$, $i = 0, \ldots, n-1$, that determine the coefficients of the Vandermonde matrix $V \in \mathbb{R}^{n_{\mathrm{p}} \times n_{\mathrm{c}}}$ having as kth row the vector

$$\boldsymbol{m}(s_k, t_k) = \left(s_k, t_k, s_k^2, s_k t_k, t_k^2, \ldots\right).$$

By slightly abusing the notation, we can assume to work componentwise and reduce the least squares fitting problem to the linear system

$$V^T V \boldsymbol{c} = V^T \left(\boldsymbol{p} - \boldsymbol{p}_{\mathrm{ev}}\right),$$

where the approximation points $\boldsymbol{p}_k$ are arranged in a vector $\boldsymbol{p}$. Since the coefficients $\boldsymbol{c}$ depend only on the approximation points, in a preprocessing stage the inverse of the matrix $V^T V$ can be computed and the entries of $\left(V^T V\right)^{-1} V^T$ can be stored for each valence n. Referring to the workflow in Figure 2.3, the patcher function computes the solution of the least squares problem as a matrix-vector product, stores the polynomial coefficients and, if necessary, updates them after editing of the subdivision control mesh.

2.4.4 Extraordinary vertices on the boundary

The local correction method presented in the previous section can also be extended to extraordinary vertices on the boundary of non-closed surfaces. Let $\hat{\psi}$ be the mapping between the base domain Q and one Catmull-Clark limit surface patch incident in a boundary extraordinary vertex of valence $n \geqslant 4$. Thus the function $\hat{\psi}$ is the evaluator for a Catmull-Clark surface with boundary, e.g., as defined in [BLZ00, SES04, LB07]. We consider an open configuration formed by $n-1$ adjacent sectors of the mesh whose vertices are the entries of the subdominant eigenvectors of the local subdivision matrix relative to valence $2(n-1)$. If we apply $\hat{\psi}$ to this configuration, we get $n-1$ patches $\hat{\boldsymbol{K}}_i^{[n]}$, $i = 0, \ldots, n-2$, which can be used to define the fan-shaped domain:

$$\hat{K}_n := \bigcup_{i=0}^{n-2} \left\{ \hat{\psi}_{\hat{\boldsymbol{K}}_i^{[n]}}\left(\sigma_h\left(\phi(u,v)\right)\right) \,\middle|\, (u,v) \in Q \text{ and } \sigma_h(\phi(u,v)) \in Q \right\},$$

with h given by (2.4). $\hat{K}_n$ plays the same role as the star-shaped domain K_n in Section 2.4.2, i.e., it is the local domain in which both the approximating polynomial and the Catmull-Clark surface are represented in a neighborhood of the boundary extraordinary vertex. On this domain, a semicircular blending region $\hat{D}_n$ can be defined similarly to D_n as

$$\hat{D}_n := \left\{ (s,t) \in \hat{K}_n \,\middle|\, \|(s,t)\|_2 \leqslant \lambda^{[2(n-1)]} \right\},$$

and the approximating polynomial can be computed as described in Section 2.4.3, thus by sampling the Catmull-Clark limit surface at the parametric values (2.8) for each of the $n-1$ patches incident in the vertex.

2.4.5 Exact evaluation of the blended surface

This section presents the algorithm to compute the value, first and second derivatives of a patch $\boldsymbol{S}_i^*$ defined as in (2.5), containing at least one extraordinary vertex. Given the couple of parameters $(\bar{u}, \bar{v})$ in the base domain Q of the patch, we check whether $(\bar{u}, \bar{v})$ lies inside the preimage $\tilde{D}_{n,i}$ of a blending region. If $\boldsymbol{S}_i^*$ contains more than one extraordinary vertex, then either the image of $(\bar{u}, \bar{v})$ lies outside all the related blending regions, or it belongs to exactly one of them, since by construction the blending regions do not overlap. Thus, if the image of $(\bar{u}, \bar{v})$ in the star-shaped domain K_n through the reparameterization κ_i does not lie inside the blending region D_n of any extraordinary vertex, the surface is directly evaluated in the parameterization $\psi_{\boldsymbol{S}_i}$ of the original Catmull-Clark patch $\boldsymbol{S}_i$. Alternatively, the value and the derivatives of the weight function w and the polynomial $\boldsymbol{P}$ are computed in K_n; then the value of the blended surface $\boldsymbol{S}_i^*$ is computed through (2.5), and its derivatives w.r.t. u, v are obtained by applying the inverse mapping κ_i^{-1} to the derivatives of w and $\boldsymbol{P}$ w.r.t. s, t as in (2.6), (2.7), and by composition.

The algorithm below summarizes the described evaluation procedure.

```
Algorithm. Evaluation of the blended surface.
Input:  $\boldsymbol{S}_i$ ($i$ local index of the patch around the e.v.),
        vertex valence $n$,
        $w$, $\boldsymbol{P}$,
        evaluation parameters $(\bar{u}, \bar{v}) \in Q$.
Output: $\boldsymbol{S}_i^*, \partial_u \boldsymbol{S}_i^*, \partial_v \boldsymbol{S}_i^*, \partial_{uu} \boldsymbol{S}_i^*, \partial_{vv} \boldsymbol{S}_i^*, \partial_{uv} \boldsymbol{S}_i^*$ at $(\bar{u}, \bar{v})$.

Evaluate $\boldsymbol{S}_i$, $\partial_u \boldsymbol{S}_i, \partial_v \boldsymbol{S}_i, \partial_{uu} \boldsymbol{S}_i, \partial_{vv} \boldsymbol{S}_i, \partial_{uv} \boldsymbol{S}_i$ at $(\bar{u}, \bar{v})$;
Compute $(\bar{s}, \bar{t}) = \kappa_i(\bar{u}, \bar{v})$;
if $(\bar{s}, \bar{t}) \in D_n$ then
    Compute the value and the derivatives of $w$ and $\boldsymbol{P}$ w.r.t. $s, t$;
    $\boldsymbol{S}_i^*(\bar{u}, \bar{v}) = w(\bar{s}, \bar{t}) \boldsymbol{S}_i(\bar{u}, \bar{v}) + (1 - w(\bar{s}, \bar{t})) \boldsymbol{P}(\bar{s}, \bar{t})$;
    Compute the derivatives of $w$ and $\boldsymbol{P}$ w.r.t. $u, v$ by using (2.6) and (2.7);
    Evaluate $\partial_u \boldsymbol{S}_i^*, \partial_v \boldsymbol{S}_i^*, \partial_{uu} \boldsymbol{S}_i^*, \partial_{vv} \boldsymbol{S}_i^*, \partial_{uv} \boldsymbol{S}_i^*$ at $(\bar{u}, \bar{v})$;
else
    $\boldsymbol{S}_i^*(\bar{u}, \bar{v}) = \boldsymbol{S}_i(\bar{u}, \bar{v})$ (and derivatives);
end if
```

2.5 Computational issues

The integration of subdivision surfaces in a CAD system and the application of the described local correction method require that the limit surface be exactly evaluable at arbitrary parametric coordinates. Although, as mentioned, there are various state-of-the-art evaluation techniques, the implementation of these methods raises some general issues, that do not depend on a particular algorithm, but on the specific requirement for application in CAD. This section discusses such issues and how they were practically addressed in our implementation to guarantee compatibility within a CAD environment.

Parameterization gap around an extraordinary point. The parameterization of the limit surface through nested spline rings, which is at the basis of known evaluation methods, covers the neighborhood of an extraordinary vertex only ad infinitum (see Figure 2.2). In practice, the need to truncate this infinite sequence generates a parameterization gap around the vertex and the consequent impossibility of evaluating the surface in this area. This situation also occurs when performing the local correction introduced in the previous section, and in particular when evaluating the sector of characteristic map $\boldsymbol{K}_0^{[n]}$.

In practice the problem can be fixed as described in the following. At the extraordinary point it is possible to compute exactly both the limit surface value and the normal by well-known formulas (see [HKD93, Appendix A]). Moreover, the first and second derivatives with respect to the base domain are extrapolated (and additionally the first derivatives are projected on the tangent plane defined by the normal) from the values they assume along the boundary of the last ring of the truncated spline sequence, i.e., at a predefined distance δ from the extraordinary point. Finally, linear interpolation between surface and derivative values on the boundary of the last ring and those at the extraordinary point has been exploited to bridge the gap in the intermediate region. Clearly, this procedure results in a small local perturbation of the regularity of the surface in a δ-wide neighborhood of each extraordinary point. However, this is not a major issue for application, provided that the distance δ is estimated in such a way that the numerical algorithms available in the system are not sensitive to this perturbation. Empirical observations suggest a suitable value $\delta = 10^{-10}$.

Evaluation outside the parametric domain of a patch. Some tools (such as Boolean operations, fitting, and offset) may require to evaluate a surface patch outside its parametric domain, which is in our case the base domain Q. The structure of a subdivision surface, made of spline (thus polynomial) rings, suggests that this operation could be accomplished by extrapolating a suitable spline ring. However, this approach can not generate satisfactory results when the extrapolated value is close to an extraordinary point, since the size of the rings reduces exponentially while approaching the point. This issue has been overcome by extrapolating the values external to Q and at a distance smaller than 10^{-10} from a tangent ribbon (i.e., a ruled surface defined by the first derivatives in the cross-boundary direction).

Corner vertices. A corner corresponds to a vertex of valence 2 of an open control mesh, and therefore it belongs to one limit surface patch only. To evaluate this patch, we extrapolate linearly one ring of mesh faces across the boundary. In the fictitious extrapolated mesh, the corner vertex has valence 4, so it can be regarded as regular and the corner patch can be evaluated as a spline one.

2.6 Modeling examples

This section presents some modeling examples provided by think3 and realized with the ThinkDesign system.

Figure 2.9 illustrates a modeling and editing sequence of a B-rep solid with hybrid faces. In the first row, from the left, the sequence shows a Subd-B-rep of a tap model. A blind hole is created through Boolean intersection with a cylinder and afterwards a fillet is inserted to smooth down the sharp joint between the two surfaces. Note that the Boolean operations already available in ThinkDesign were used, i.e., no ad hoc implementation for subdivision surfaces was necessary. The three rightmost figures show mean curvature, isophotes, and rendering of the model. In the second row, the hybrid nature of the faces of the B-rep is highlighted: subdivision faces are colored in yellow, a cylindrical face in green and a NURBS face in red. The two central figures show the editing of the subdivision control mesh, which results in a still hybrid B-rep represented in the last figure of the second row. In particular, after editing the control mesh,

the hole and the fillet were automatically updated.
For this and the following examples, we recall that the radius of the blending region is chosen as described in Section 2.4.2. Accordingly, the extension of the blending region depends on the type of patch as illustrated in Figure 2.8.
The modeling sequence proposed in Figure 2.10 emphasizes the possibility of working with surfaces and solids at the same time in a hybrid modeling CAD environment. An iron model is designed by exploiting subdivision surfaces and thus represented as a Subd-B-rep with thickness. Successively, a blind hole is created giving rise to a hybrid B-rep, which is further edited by intersection with planar and quadric surfaces. The model is completed by two NURBS surfaces representing the knob and the cable.
Figure 2.11 shows a magnification of curvature and isophotes visualization[3] around an extraordinary point of the two models in Figures 2.9 and 2.10, comparing them with the Catmull-Clark limit surface without the local correction through blending: the improved numerical behavior around extraordinary points is even visibly noticeable.

The design and development of the subdivision software prototype integrated in ThinkDesign went in a feedback loop with the validation of some end-users. The Italian-based firm of household appliances Alessi has validated the software from the computer-aided styling (CAS) point of view. One of the major features required by Alessi and then embedded in the subdivision prototype was the integration between the conceptual design and the engineering phases which can be only achieved by a perfect integration of the new geometric representation in the geometric kernel. This allows us to approach subdivision as any other CAD feature in a history-based CAD system where a model description is stored and successively redefined interactively. Thus, a geometric model based on subdivision can be modified shortly after some features, such as shells, fillets, chamfers, Boolean operations, parting lines, etc., have been applied to it, and these features are automatically updated.
Figures 2.12 and 2.13 illustrate the modeling of a product manufactured by Alessi exploiting the subdivision surface capability integrated in ThinkDesign. The illustrated model was developed from a Subd-B-rep obtained from the conversion of a solid cube. The sequence of divisions, extrusions, and editing of faces and vertices has allowed the user to model the desired shape in a fast and easy way (Figure 2.12). The final model, shown in Figure 2.13(a), now needs to be engineered in order to be produced. This engineering design phase explains why it is necessary that the subdivision module be intimately tied around the CAD system. Assuming the model is to be produced in plastic, we need to add thickness to it, that is to apply a shell feature, as illustrated in Figure 2.13(b). Successively, for manufacturing reasons due to the processing of molding machines, it is convenient to separate the object into several parts: within an integrated system, this task is easily performed by existing Boolean operators or through the cutting surfaces feature, exploiting the peculiarity of a hybrid system (Figure 2.13(c)). If we want to make major changes in the style and/or in the dimension of the object, the integrated module allows us to just remodel and automatically update. Conversely, if the subdivision module were not integrated, the designer should restart working on the initial model, redoing all the steps involving thickness and cutting features. The feedback from Alessi on the benefit introduced by the integration of subdivision surfaces in the CAD system is very positive. Alessi emphasizes the visual quality of the obtained surfaces, the ease of use, and especially the complete integration with the rest of the CAD environment, and concludes that this makes a real progress with respect to the standard CAD approach.

[3] For a description of the use of isophotes in surface interrogation and quality assessment see, e.g., [KK88, The01].

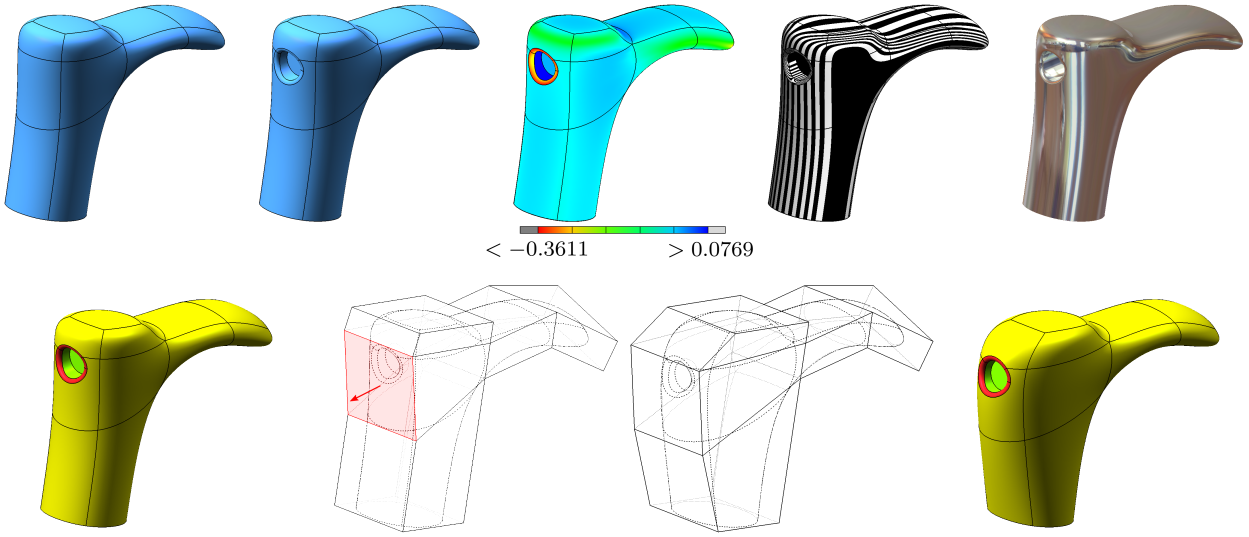


Figure 2.9: Example of a tap model. First row, from left to right: B-rep solid, creation of a filleted blind hole, visualization of mean curvature, isophotes, and final rendering of the model. Second row: hybrid B-rep of the model, before and after the editing performed on the vertices of a face of the control mesh. The different colors highlight faces having different geometric representation.

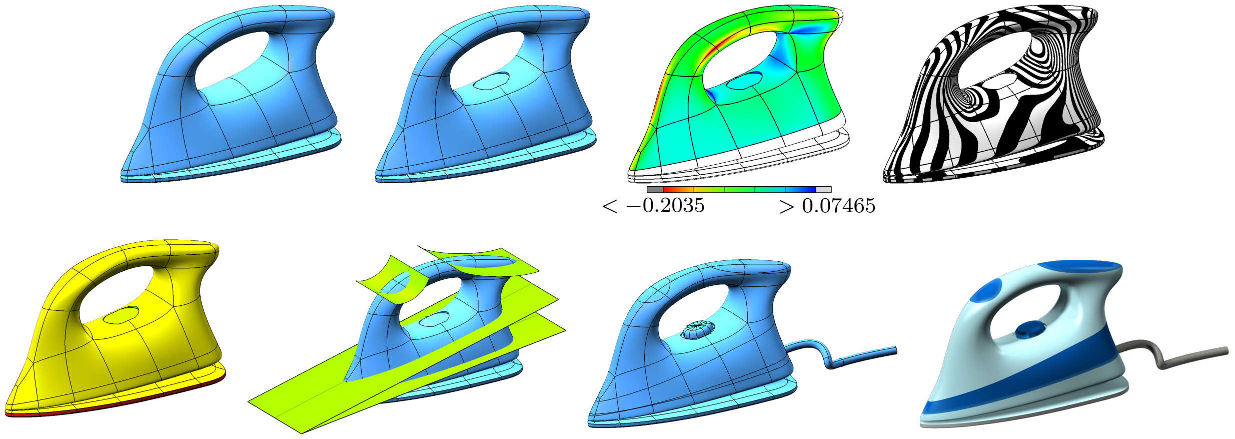


Figure 2.10: Example of an iron model. First row, from left to right: B-rep solid, creation of a blind hole, visualization of mean curvature and isophotes. Second row: hybrid B-rep of the model (faces with different geometric representations are shown in different colors), hybrid modeling exploiting surface/solid intersection tools, creation of knob and cable, and final rendering of the assembled model.

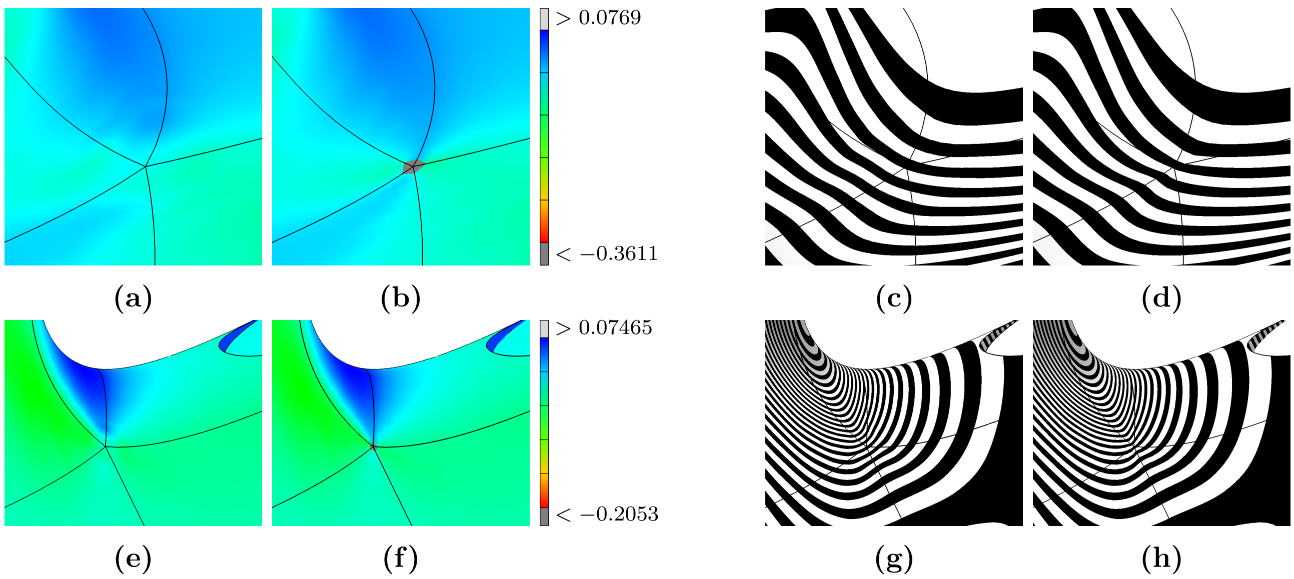
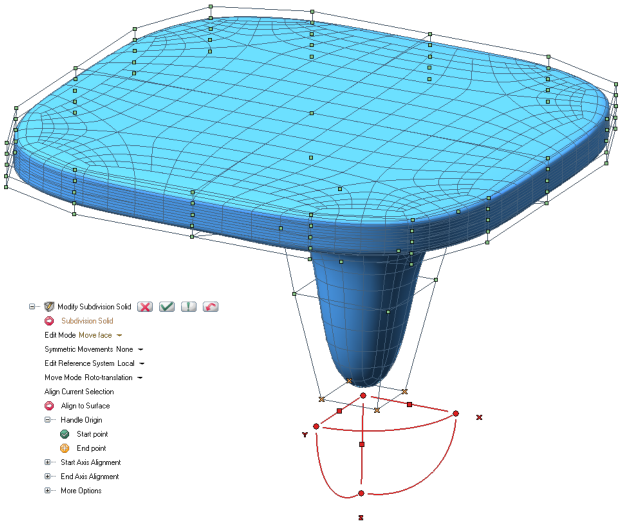


Figure 2.11: (a), (c), (e), (g): magnification of curvature visualization and isophotes in Figures 2.9 (first row) and 2.10 (second row) around an extraordinary point with local correction; (b), (d), (f), (h): same area of the model without local correction.



Modify Subdivision Solid ☒ ☑ ! ↶
Subdivision Solid
Edit Mode Move face
Symmetric Movements None
Edit Reference System Local
Move Mode Roto-translation
Align Current Selection
Align to Surface
Handle Origin
Start point
End point
Start Axis Alignment
End Axis Alignment
More Options

Figure 2.12: Editing of the subdivision control mesh.

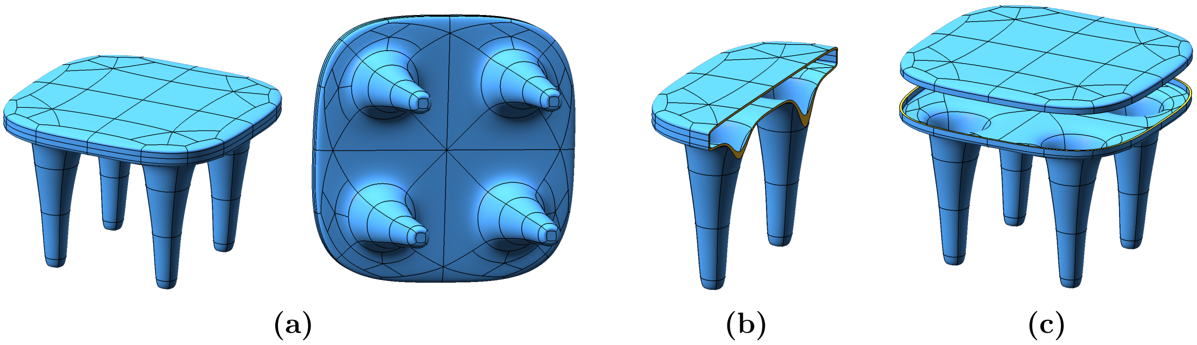


Figure 2.13: Example of a stool designed by Alessi. (a) Subd-B-rep solid, (b) model with added thickness, (c) division of the object in two parts.

2.7 Conclusions

In this chapter we discussed the theoretical and practical aspects involved in the integration of subdivision surfaces in a CAD system, exploiting a suitable parameterization method and exact evaluation. We have proposed a solution to the regularity issues that usually prevent the adoption of subdivision in CAD applications, by providing a local correction method that guarantees C^2 continuity also in the neighborhood of extraordinary vertices. To this end, we have pursued a blending approach, whose effectiveness, although limited to the discrete setting, was already emphasized in [Lev06b, Zor06]. In particular, this method provides a quite regular curvature behavior against a negligible drawback, which is a slight violation of the convex-hull property in situations of minor practical relevance, as discussed in [Lev06b]. The good quality of the obtained surfaces was confirmed by the testing carried out by our industrial partners.
Using the proposed local correction, commonly used CAD tools like offset, local approximation in Taylor expansion, intersections, and any other are performed successfully. Conversely, they produce unpredictable results when applied to the (standard) Catmull-Clark surfaces. This is due to the fact that these algorithms make extensive use of iterated interrogations of the surface and its first and second derivatives at successively generated arbitrary parameter values and, in this process, they need or assume a regular curvature behavior everywhere.
Subdivision surfaces have been fully integrated in the ThinkDesign CAD system since version 2011.1, in the sense that any existing CAD tool can be applied to them and they can interact with other geometric entities, like NURBS, making it possible to perform Boolean operations between hybrid representations. This allows a seamless integration of hybrid NURBS/subdivision/analytical geometric models in a hybrid CAD system, which was never attained before, either theoretically or in practical applications.

2.8 Acknowledgments

The research that resulted in the paper [ABC$^+$13], which is at the base of this chapter, and in the integration of subdivision surfaces in the ThinkDesign CAD system was supported by the European Eurostars Project NIIT4CAD under the coordination of think3. The author of this dissertation and the other authors of [ABC$^+$13] particularly thank, among the project partners, Cristiano Colosio, Product Modeling Engineer at Alessi, for the collaboration in the assessment of the software prototype and for providing the example shown in Figures 2.12 and 2.13.

Chapter 3

A general framework for the construction of piecewise-polynomial local interpolants of minimum degree

This chapter is devoted to the problem of designing piecewise polynomial local interpolants of non-uniformly spaced data. Replicating the content of the recent paper [ABC14a], we present a constructive approach that, for any assigned degree of polynomial reproduction, continuity order, and support width, allows us to generate the fundamental spline functions of minimum degree having the desired properties. Moreover, the proposed construction is extended to handle open sets of data and to the case of multiple knots.

3.1 Introduction

In this chapter we present a general framework for the construction of locally supported fundamental functions for spline interpolation defined on non-uniform knot partitions. As opposed to the traditional approach, in which one imposes the interpolation constraint and thereafter builds on it, we address here the situation in which the user may want to choose the relevant characteristics of the interpolant depending on the context of application. A similar idea is also at the basis of the work in [BTU03], which, however, is limited to considering the uniform case only. In that paper, it is observed that any piecewise-polynomial interpolation kernel can be characterized by four main *design parameters*: degree, support, regularity, and order of approximation. Obviously, such a classification can be adopted also in the non-uniform case and possibly extended to other kinds of splines (e.g., exponential or rational). Inspired by this observation, we consider the general problem of designing polynomial spline interpolants of given degree, support, regularity, and order of approximation on non-uniform knot partitions. In this respect, although many results are known about the uniform setting, at present the non-uniform case is addressed by the work in [BCR13a] only. In that paper, families of splines with different design parameters are defined through a suitable combination of polynomial interpolants with either polynomial or rational, compactly supported blending functions. In this way, polynomial reproduction is guaranteed by construction, whereas the order of continuity must be ensured by a proper choice of blending functions. To some extent, the approach can be seen as a generalization of the well-known construction of Catmull-Rom splines [CR74].

In approximation theory the best quality, within a fixed support, corresponds to the highest possible approximation order and continuity. Based on this idea, the work in [DGM88] proves the existence and uniqueness of some families of *optimally local interpolating splines*, defined as those splines having degree h, approximation order $h+1$, and order of continuity $h-1$. Thus, optimally local interpolating splines have minimum degree and highest possible continuity. However, this feature is not a requirement for most applications: for instance, in image processing one usually requires high approximation order and small support, regardless of continuity; conversely, in computer-aided design a reasonable balance between all the four design parameters may be more effective in order to provide an interpolant with good shape quality. Clearly, regardless of the context, the spline degree should be the lowest possible. In particular, the minimum degree achievable in local spline interpolation is either equal to the polynomial reproduction degree or equal to the order of continuity plus one. In this respect, it must be observed that the aforementioned technique [BCR13a] does not allow designing optimally local interpolating splines nor, more generally, splines having minimum degree.

The objective of this chapter (as well as of the work [ABC14a]) is to provide a complete parameterization of the class of non-uniform, minimum-degree, locally supported, interpolating splines with respect to the considered design parameters: more precisely, we present a constructive approach that, for any arbitrary choice of continuity order, polynomial reproduction degree, and support width, allows us to design the corresponding fundamental spline functions of minimum degree. To this aim, we extend and generalize the approach in [BCR13a] relying on the observation that, in the design of locally supported fundamental splines, increased flexibility can be obtained interpolating by splines with knots $\boldsymbol{t} := \{t_j\}$ where the interpolation nodes $\boldsymbol{x} := \{x_j\}$ are strictly contained in $\boldsymbol{t}$. We can date back this idea to the well-known Powell-Sabin interpolant [PS77], where the additional degrees of freedom are used in order to do Hermite interpolation at the interpolation points. Successively, the paper [Woo87] introduces the term *B2-splines* to describe a local representation for cubic spline interpolants where $t_{2j} = x_j$ and $t_{2j+1} \in (x_j, x_{j+1})$. A similar construction is also proposed in [DGM88, KFK94], and adapted to B3-splines in [Chu90]. More generally, in this chapter we construct *Br-splines*, namely splines made of r polynomial pieces between t_{rj} and t_{rj+r}. Also, unlike the previous works, the proposed approach allows us to handle not only the case $x_j = t_{rj}$, but also the case $x_j \in (t_{rj-r}, t_{rj})$.

We will see that the proposed method provides an effective procedure for the computation of optimally local interpolating splines. In addition, it also allows us to generate other families of non-uniform interpolating splines, parameterized by the considered design parameters, that, to our knowledge, have never appeared before. Besides this, the framework includes the families of splines investigated in [BCR13a], regarded as B1-splines.

The remainder of the chapter is organized as follows. In Section 3.2 we review the approach in [BCR13a] and the relevant properties of the generated splines. In Section 3.3 we generalize the interpolation framework to design Br-splines and illustrate the constructive method by means of some significant examples. In Section 3.4 we study the family of minimum-degree splines and show that the proposed method always allows us to design a minimum-degree spline with assigned other design parameters. Section 3.5 extends the framework to open sets of data. Section 3.6 discusses the role of the non-uniform knot partition, with particular emphasis on the interesting situation in which the knots are set to coincident locations. Some final remarks and general observations on interpolation by Br-splines are collected in Section 3.7.

3.2 A blending-based approach to the design of non-uniform local interpolating splines

In this section we review the approach presented in [BCR13a] for the construction of non-uniform piecewise-polynomial local interpolants with given smoothness and approximation order.
We denote by $\{(x_j, y_j)\}$ the given interpolation data. In particular, we assume that $\boldsymbol{x} := \{x_j\}$, called *nodes*, form an increasing sequence. We also denote by $p_{i,m} \in \Pi_m$ the degree-m polynomial such that $p_{i,m}(x_j) = y_j$, $j = i, \ldots, i+m$.
We consider the family of splines represented by

$$F(x) = \sum_{i=\ell-n}^{\ell} p_{i-\sigma,m}(x) B_{i,n+1}(x), \qquad x \in I_\ell := [\tau_\ell, \tau_{\ell+1}). \tag{3.1}$$

In the above expression, $B_{i,n+1}$ are *blending functions* defined on a strictly increasing sequence of values $\boldsymbol{\tau} := \{\tau_j\}$ and satisfying the general requirements of having

(i) *Compact support*: $B_{i,n+1}(x) > 0, x \in (\tau_i, \tau_{i+n+1})$, and $B_{i,n+1}(x) = 0, x \notin (\tau_i, \tau_{i+n+1})$;

(ii) *Partition of unity property*: $\sum_i B_{i,n+1}(x) = 1, \forall x \in \mathbb{R}$.

The reciprocal configuration of $\boldsymbol{x}$ and $\boldsymbol{\tau}$ determines a first distinction among all the splines represented by (3.1). In this respect, two relevant situations are worth considering and correspond to

$$\boldsymbol{\tau} \equiv \boldsymbol{x} \qquad (\textit{even configuration}); \tag{3.2a}$$

$$\boldsymbol{\tau} \neq \boldsymbol{x} \text{ and } x_j < \tau_j < x_{j+1}, \forall j \qquad (\textit{odd configuration}). \tag{3.2b}$$

From formula (3.1) we can understand the leading idea of the proposed approach, that is, the sought degree of polynomial reproduction (and the related approximation order) is automatically guaranteed by construction, whereas the order of continuity must be ensured by a proper choice of blending functions. Although, generally speaking, the blending functions in (3.1) may be of various types (e.g., polynomial, rational, trigonometric, hyperbolic), in this chapter we focus on the cases in which they are either polynomial or rational. In the following, we enumerate the relevant properties of the considered family of splines, referring to [BCR13a] for their proof.

1) *Interpolation.* Since we are interested in the construction of splines that are interpolating, we need to characterize when this is the case for the functions defined by (3.1). In this regard, it can be proved that (3.1) represents a family of interpolating splines for fixed degree m of the polynomials and support width $n+1$ of the blending functions if and only if the integer shift σ belongs to Σ, where either $\Sigma = \{-1, \ldots, m-n\}$ and $\sigma \neq -1$ if $n = 0$ in case of even configuration, or $\Sigma = \{-1, \ldots, m-n-1\}$ in case of odd configuration.

   Moreover, the requirement that F interpolates the data provides useful necessary conditions relating m and n. More precisely, F is interpolating only if

   ◦ $m + 1 \geqslant n$, in case of even configuration;

   ◦ $m + 1 \geqslant n + 1$, in case of odd configuration.

2) *Support width.* We call *fundamental functions* associated with a class of local interpolating splines and a given set of nodes $\boldsymbol{x} = \{x_j\}$ the functions ψ_i that belong to the considered class and satisfy $\psi_i(x_j) = \delta_{i,j}$. The support width w of a fundamental function associated with a family (3.1) is
$$w = m + n + 1. \tag{3.3}$$

3) *Smoothness.* By construction, the order of continuity of F is at least equal to the order of continuity of the blending functions that appear in (3.1).

   In the even configuration, an improvement on this estimate is obtained in two cases:

   3.a) The $n+2$ polynomials associated with the non-vanishing blending functions on the interval $[\tau_{\ell-1}, \tau_{\ell+1})$ are all interpolating the data (x_ℓ, y_ℓ), which corresponds to having $m+1 > n$ and $\sigma \neq -1, m-n$. In this case, if the blending functions are C^k at $\boldsymbol{\tau}$ and at least C^{k+1} everywhere else, then F is C^{k+1}.

   3.b) Conversely, if $\sigma = -1, m-n$, F is C^{k+1} if the blending functions are C^k at $\boldsymbol{\tau}$, C^{k+1} at the endpoints of their support, and at least C^{k+1} everywhere else. Note that this is always the case when $m+1 = n$ because $\Sigma = \{-1\}$.

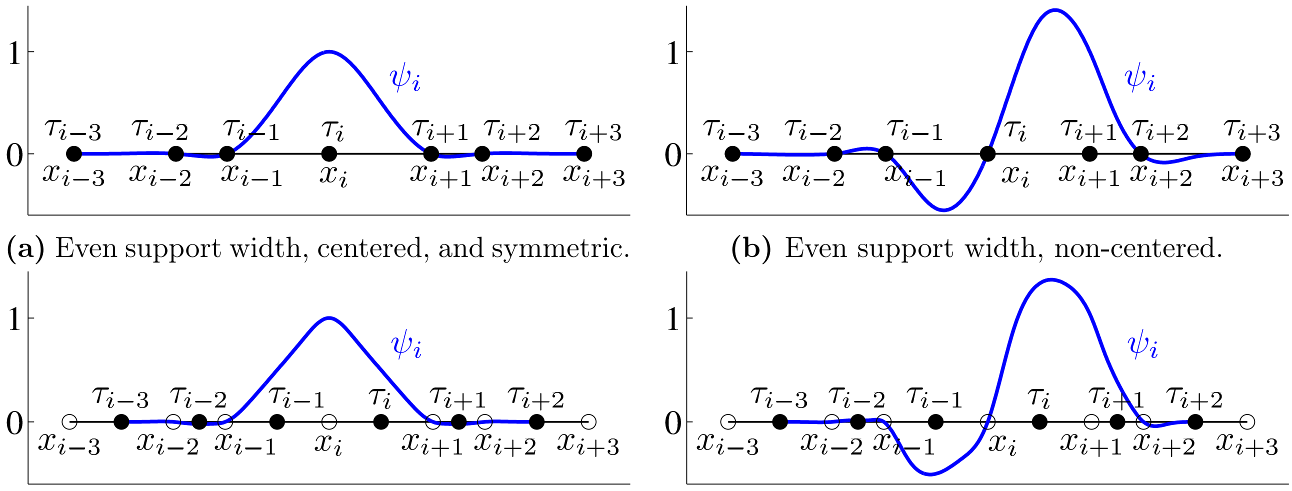
4) *Polynomial reproduction and approximation order.* By construction, F reproduces polynomials of degree at least m. From this and the interpolation property of F, exploiting standard estimates for polynomial approximation, it can be proved that, for any $f \in C^{m+1}(\mathbb{R})$ and interpolation data $(x_j, f(x_j))$,
$$\|F - f\|_{I_\ell} \leqslant \gamma \left(x_{\ell+\lceil \frac{w}{2} \rceil} - x_{\ell-\lfloor \frac{w}{2} \rfloor+1}\right)^{m+1} \left\|f^{(m+1)}\right\|_{\left[x_{\ell-\lfloor \frac{w}{2} \rfloor+1}, x_{\ell+\lceil \frac{w}{2} \rceil}\right]},$$
   where γ is independent on f, w is the support width, and $\|f\|_I := \max\limits_{x \in I} |f(x)|$. Thus F has approximation order $m+1$.

5) *Symmetry.* We say that the fundamental functions are *centered* if they assume the value 1 at the central node of their support in case of even support width and at the midpoint of the central interval of their support in case of odd support width. The property of being centered is a necessary condition for the symmetry of F and holds when
$$\sigma = \left\lfloor \frac{m-n-1}{2} \right\rfloor, \tag{3.4}$$
   and $w = m + n + 1$ is even/odd respectively in the even/odd configuration.

   Some examples of centered and non-centered fundamental functions are shown in Figure 3.1.

By the partition of unity property of the blending functions, we can reformulate (3.1) as[1]
$$\begin{aligned} F(x) &= p_{\ell-n-\sigma,m}(x) + \sum_{i=\ell-n+1}^{\ell} \left(p_{i-\sigma,m}(x) - p_{i-\sigma-1,m}(x)\right) \sum_{j=i}^{\ell} B_{j,n+1}(x) \\ &= p_{\ell-n-\sigma,m}(x) + \sum_{i=\ell-n+1}^{\ell} c_{i-\sigma}\, f_i(x) \prod_{j=i-\sigma}^{i-\sigma+m-1} (x - x_j), \qquad x \in I_\ell, \end{aligned} \tag{3.5}$$

[1] We refer to [BCR13a] for a detailed derivation of expression (3.5).



(a) Even support width, centered, and symmetric.

(b) Even support width, non-centered.

(c) Odd support width, centered, and symmetric.

(d) Odd support width, non-centered.

Figure 3.1: Centered and non-centered fundamental functions in even (top row) and odd (bottom row) configuration.

where (denoting by $[\cdot]$ the standard divided differences)

$$c_j \coloneqq [y_j, \dots, y_{j+m-1}, y_{j+m}] - [y_j, \dots, y_{j+m-1}, y_{j-1}],$$

and $f_i(x) = \sum_{j=i}^{\ell} B_{j,n+1}(x)$, for $x \in I_\ell$. Note that we can extend the definition of f_i to $\mathbb{R}$ as

$$f_i(x) \coloneqq \sum_{j \geqslant i} B_{j,n+1}(x) = \begin{cases} 0, & x < \tau_i, \\ \displaystyle\sum_{j=i}^{i+n-1} B_{j,n+1}(x), & x \in [\tau_i, \tau_{i+n}), \\ 1, & x \geqslant \tau_{i+n}. \end{cases}$$

The functions f_i are called *transition functions*. By the above formula and recalling that $\operatorname{supp}(B_{i,n+1}) = [\tau_i, \tau_{i+n+1}]$, we have $B_{i,n+1} = f_i - f_{i+1}$.

In the remainder of the chapter we focus only on spline interpolants that preserve possible symmetries of the data. For this to happen, two conditions must hold true. First, the corresponding fundamental functions must be centered, which, according to Property 5, requires σ as in (3.4) and the even/odd configuration in correspondence of even/odd support width. Second, the blending functions must be symmetric (when so are the related values in $\boldsymbol{\tau}$), which means that the transition functions f_i must be antisymmetric with respect to the midpoint of $[\tau_i, \tau_{i+n}]$.

3.3 Generalization of the interpolation framework

The formulation (3.5) readily suggests a method for minimizing the degree of F within the considered framework. The idea is to design the transition functions f_i, $\forall i = \ell - n + 1, \dots, \ell$, as piecewise rational functions whose denominator simplifies as many factors as possible among $(x - x_j), j = i - \sigma, \dots, i - \sigma + m - 1$. In the remainder of this section we will present a suitable constructive method to achieve this goal.

In the following we refer to the elements of $\boldsymbol{\tau}$ as *breakpoints*. We introduce an increasing sequence $\boldsymbol{t} := \{t_i\}$, which is a refinement of $\boldsymbol{\tau}$ such that $t_{rj} = \tau_j$, and $t_{rj+h} \in [\tau_j, \tau_{j+1}]$, $h = 1, \ldots, r-1$. In this way, each interval I_ℓ is subdivided into r subintervals $I_{\ell,h} := [t_{r\ell+h-1}, t_{r\ell+h})$, $h = 1, \ldots, r$. Hence, we define each f_i as a piecewise rational function on $\boldsymbol{t}$ of the form

$$f_i(x) = \begin{cases} 0, & x < t_{ri}, \\ \dfrac{q_{i,j}(x)}{\pi_{i,j}(x)}, & x \in [t_{ri+j-1}, t_{ri+j}), \quad j = 1, \ldots, rn, \\ 1, & x \geqslant t_{ri+rn}, \end{cases} \tag{3.6}$$

where $q_{i,j}$ and $\pi_{i,j}$ are suitable polynomials. On each interval $I_{\ell,h}$, $\ell = i, \ldots, i+n-1$, the denominator of f_i is the product of all the factors among $(x - x_j), j = i-\sigma, \ldots, i-\sigma+m-1$ that do not vanish in $I_{\ell,h}$, while the numerator is determined by imposing that the resulting spline interpolant F has continuity C^{k+1}. The number of continuity conditions at each point of $\boldsymbol{t}$ is inferred from Property 3 in Section 3.2. In particular, in the even configuration we require that f_i be C^k at $\boldsymbol{\tau}$ and C^{k+1} at $\boldsymbol{t}\backslash\boldsymbol{\tau}$ (see case 3.a); moreover, if $m+1 = n$, then f_i must be C^{k+1} also at t_{ri} and $t_{r(i+n)}$ (see case 3.b). In the odd configuration f_i is required to be C^{k+1} at any point of $\boldsymbol{t}$. Let us denote by $k_{i,h}$ the order of continuity of f_i at t_{ri+h}, $h = 0, \ldots, rn$, and set $f_{i,j} := \frac{q_{i,j}}{\pi_{i,j}}$, $j = 1, \ldots, rn$. Then f_i is obtained by solving the linear system

$$\begin{aligned} f_{i,1}^{(s)}(t_{ri}^+) &= 0, & s &= 0, \ldots, k_{i,0}, \\ f_{i,j}^{(s)}(t_{ri+j}^-) &= f_{i,j+1}^{(s)}(t_{ri+j}^+), & s &= 0, \ldots, k_{i,j}, \quad j = 1, \ldots, rn-1, \\ f_{i,rn}^{(s)}(t_{ri+rn}^-) &= \delta_{s,0}, & s &= 0, \ldots, k_{i,rn}. \end{aligned}$$

If we express the polynomials $q_{i,j}$, $j = 1, \ldots, rn$, which have degree $d_{i,j} := \deg(q_{i,j})$ respectively, in a basis $u_0, \ldots, u_{d_{i,j}}$ with coefficients $\boldsymbol{b}_{i,j}$, then the above system can be written in the form

$$M\boldsymbol{b} = \boldsymbol{c},$$

with

$$M := \begin{pmatrix} M_1(t_{ri}) & & & & & & \\ M_1(t_{ri+1}) & -M_2(t_{ri+1}) & & & & & \\ & M_2(t_{ri+2}) & -M_3(t_{ri+2}) & & & & \\ & & \ddots & \ddots & & & \\ & & & M_{rn-2}(t_{ri+rn-2}) & -M_{rn-1}(t_{ri+rn-2}) & \\ & & & & M_{rn-1}(t_{ri+rn-1}) & -M_{rn}(t_{ri+rn-1}) \\ & & & & & M_{rn}(t_{ri+rn}) \end{pmatrix},$$

$$\boldsymbol{b} := (\boldsymbol{b}_{i,1}, \boldsymbol{b}_{i,2}, \ldots, \boldsymbol{b}_{i,rn-1}, \boldsymbol{b}_{i,rn})^T, \qquad \text{and} \qquad \boldsymbol{c} := (0, \ldots, 0, 1, 0, \ldots, 0)^T,$$

and where $M_j(t_{ri+h})$ is a matrix of size $(k_{i,h}+1) \times (d_{i,j}+1)$ whose sth row, for $s = 1, \ldots, k_{i,h}+1$, is of the form

$$\left(D^{(s-1)} \frac{u_0(x)}{\pi_{i,j}(x)}\bigg|_{x=t_{ri+h}}, \ldots, D^{(s-1)} \frac{u_{d_{i,j}}(x)}{\pi_{i,j}(x)}\bigg|_{x=t_{ri+h}} \right).$$

We observe that each set of functions

$$\left\{ \frac{u_h}{\pi_{i,j}}, h = 0, \ldots, d_{i,j} \right\}$$

is an extended Chebyshev system on $[t_{ri+j-1}, t_{ri+j}]$ (see Theorem 2.33 in [Sch07]). Consequently, the matrix $M_j(t_{ri+h})$ has full rank, namely equal to the number of its rows $k_{i,h}+1$. From this formulation we can also see that, when the degrees of the numerators $q_{i,j}$ are chosen so that the total number of unknown coefficients is equal to the total number of continuity conditions, namely $\sum_{h=0}^{rn}(k_{i,h}+1)$, then M is square and nonsingular. This choice turns out to be appropriate in most cases (see, e.g., the examples below). The exception is represented by those cases in which the requirement, stated at the end of Section 3.2, that the transition function be antisymmetric (which means that the degrees of the numerators $q_{i,j}$ must be set accordingly) may result in having one degree of freedom exceeding the number of continuity conditions. However, also these cases can be handled by exploiting an additional condition of symmetry, in such a way that f_i is always uniquely determined (see, e.g., Example 3.2 below, in the case $r = 2$).

To characterize the spline interpolants provided by the proposed constructive approach, we introduce the following definition.

Definition 3.1 (Br-spline)**.** An interpolating spline relative to breakpoints $\boldsymbol{\tau}$ is called *Br-spline* if it is made of r polynomial pieces inside each interval $[\tau_\ell, \tau_{\ell+1})$.

The proposed interpolation framework allows us to design several families of splines, where each family is identified by four relevant features, namely degree, continuity, polynomial reproduction, and support width, that, hereinafter, are referred to as *design parameters*. For brevity, we denote a certain family using the acronym Br $D^gC^{\bar{k}}P^mS^w$, where $g, \bar{k}, m, w$ are integers representing respectively degree (g), smoothness ($\bar{k}$), degree of polynomial reproduction (m), and support width (w) of the fundamental function.

Based on the proposed construction, we illustrate the design of polynomial spline interpolants of minimum degree by considering four relevant situations that may occur in the design of f_i:

- Example 3.1 represents a basic situation and also outlines the procedure for constructing the family of optimal splines B2 $D^hC^{h-1}P^hS^{2h}$;
- Example 3.2 shows how we can design a spline of minimum support and degree with given polynomial reproduction and smoothness;
- Example 3.3 illustrates the situation in which, to obtain a spline of assigned continuity, the transition functions must have different orders of continuity at the knots.
- Example 3.4 illustrates the construction of a family of optimal splines with odd support width.

In the first three examples, we have chosen to focus on the even configuration, being this setting the most common in applications. However, the last example provides a glimpse of a case with the odd configuration.

Example 3.1 (B2 $D^3C^2P^3S^6$)**.** Our aim is designing a local B2-spline interpolant of the general form (3.1) having degree $g = h$, continuity order $\bar{k} = h-1$, polynomial reproduction $m = h$, and support width $w = 2h$. We address the particular case $h = 3$, although the same method applies in general. In the considered setting, from (3.3) and (3.4) we deduce that the blending functions have support width $n+1 = 3$ and that $\sigma = 0$. Figure 3.2(a) illustrates the reciprocal configuration between the non-vanishing blending functions in $I_\ell := [\tau_\ell, \tau_{\ell+1})$ and the associated interpolating polynomials. Note that, since the support width is even, also $\boldsymbol{x}$ and $\boldsymbol{\tau}$ are assumed

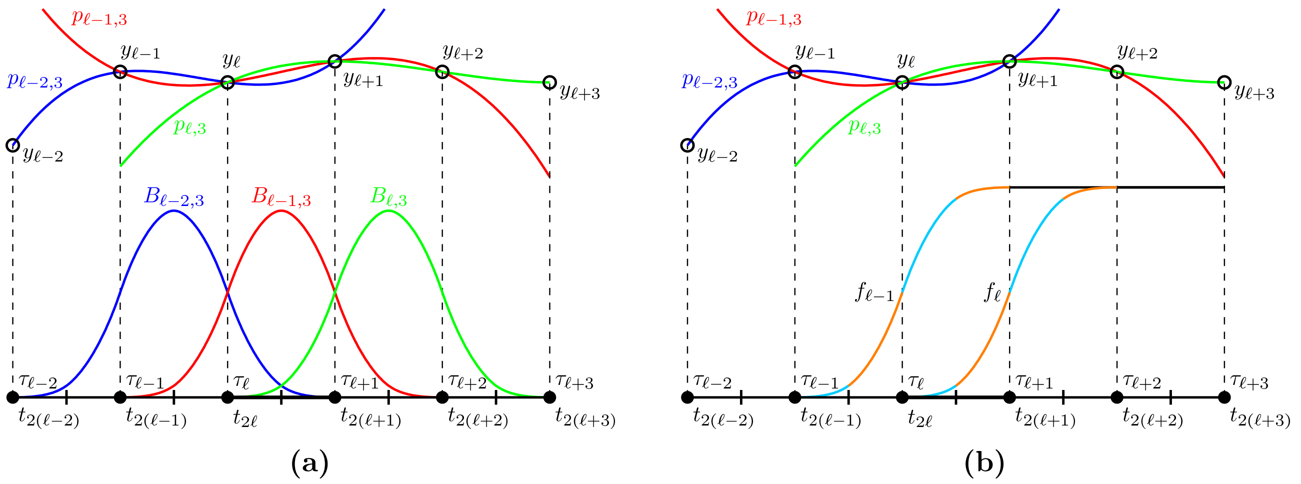


Figure 3.2: Interpolating polynomials and associated (a) blending and (b) transition functions involved in the construction of the spline interpolant in Example 3.1 on the interval $I_\ell = [\tau_\ell, \tau_{\ell+1})$. Being the configuration even, $\boldsymbol{\tau} \equiv \boldsymbol{x}$.

to be in even configuration. Following the procedure illustrated in the previous section, we can write equation (3.5) on the single interval I_ℓ as

$$F(x) = p_{\ell-2,3}(x) + f_{\ell-1}(x)\underbrace{\big(p_{\ell-1,3}(x) - p_{\ell-2,3}(x)\big)}_{c_{\ell-1}(x-x_{\ell-1})(x-x_\ell)(x-x_{\ell+1})} + f_\ell(x)\underbrace{\big(p_{\ell,3}(x) - p_{\ell-1,3}(x)\big)}_{c_\ell(x-x_\ell)(x-x_{\ell+1})(x-x_{\ell+2})}, \qquad x \in I_\ell.$$

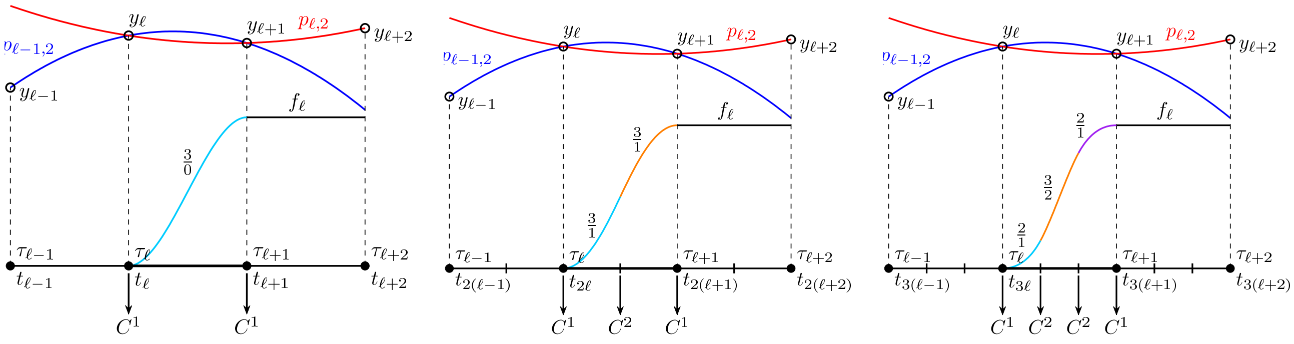
On the refined sequence $\boldsymbol{t}$ (such that $t_{2j} = \tau_j$, $\forall j$), any transition function, say f_ℓ, is made of four rational pieces between $t_{2\ell} = \tau_\ell$ and $t_{2(\ell+2)} = \tau_{\ell+2}$ (see Figure 3.2(b)). To determine its expression we need to write f_ℓ in the form (3.6) and require that F be C^2-continuous. Any other transition function can be simply determined by index shifting. Based on Property 3.a in Section 3.2, F is C^2 if f_ℓ is C^1 at $t_{2\ell}, t_{2(\ell+1)}, t_{2(\ell+2)}$ and C^2 at $t_{2\ell+1}, t_{2\ell+3}$. Overall, this gives us 12 continuity conditions, which can be satisfied supposing that the numerator $q_{\ell,j}$ of each piece $j = 1, \ldots, 4$ of f_ℓ on $\boldsymbol{t}$ is a quadratic polynomial. Since our aim is also minimizing the degree of F, the denominator $\pi_{\ell,j}$ is the product of all factors among $(x - x_\ell), (x - x_{\ell+1}), (x - x_{\ell+2})$ that do not vanish in the corresponding interval $[t_{2\ell+j-1}, t_{2\ell+j})$. The detailed construction of a transition function f_ℓ and the expression for the numerators and denominators of its pieces are given in Section 3.3.1.

Finally, a local cubic spline interpolant is obtained from (3.5) by simplifying the denominator of the transition functions with the factors inside the product symbol. The fundamental function associated with the class B2 $D^3C^2P^3S^6$ is shown in Figure 3.1(a).

Example 3.2 (B1 $D^5C^2P^2S^4$, B2 $D^4C^2P^2S^4$, B3 $D^3C^2P^2S^4$)**.** In this example, we fix the continuity and polynomial reproduction degree to be $\bar{k} = 2$ and $m = 2$ respectively. From relation (3.3), it can be easily seen that $w = 4$ is the smallest support width that allows (in the even configuration) obtaining centered fundamental functions that reproduce quadratic polynomials. From (3.3) and (3.4) we deduce that $n = 1$ and $\sigma = 0$. The resulting spline interpolant (3.5) has the expression

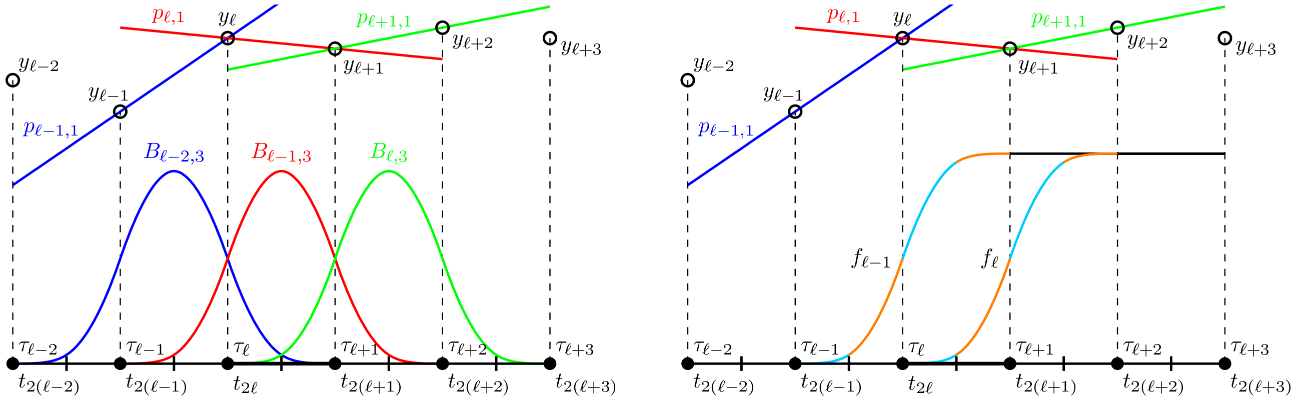
$$F(x) = p_{\ell-1,2}(x) + c_\ell f_\ell(x)(x - x_\ell)(x - x_{\ell+1}), \qquad x \in I_\ell. \tag{3.7}$$

We design the spline interpolant F for different refinement coefficients $r = 1, 2, 3$, deriving the transition function f_ℓ in the three cases (for the sake of completeness, the corresponding expressions are provided in Section 3.3.1). Figure 3.3 represents schematically the continuity



(a) B1 $D^5C^2P^2S^4$ **(b)** B2 $D^4C^2P^2S^4$ **(c)** B3 $D^3C^2P^2S^4$

Figure 3.3: Illustration of the interpolating polynomials and transition functions involved in the design of F on $I_\ell = [\tau_\ell, \tau_{\ell+1})$ in Example 3.2. The figures indicate the required continuity at each knot and the degree of the numerators and denominators of the pieces of f_ℓ (in the form $\deg(q_{\ell,j})/\deg(\pi_{\ell,j})$, $j = 1, \ldots, r$). Being the configuration even, $\boldsymbol{\tau} \equiv \boldsymbol{x}$.



(a) **(b)**

Figure 3.4: Interpolating polynomials and associated (a) blending and (b) transition functions involved in the construction of the spline interpolant in Example 3.3 on the interval $I_\ell = [\tau_\ell, \tau_{\ell+1})$. Being the configuration even, $\boldsymbol{\tau} \equiv \boldsymbol{x}$.

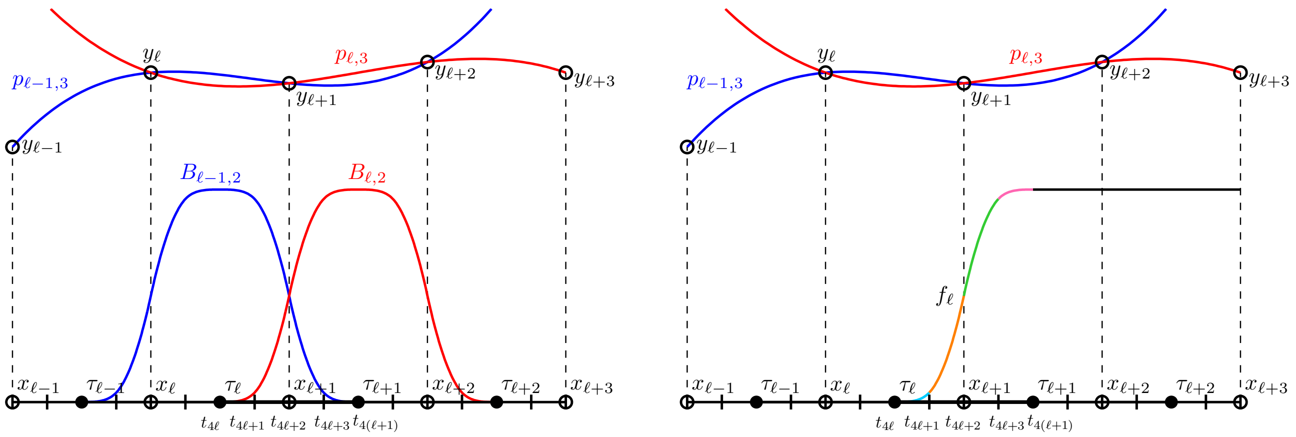
conditions at each knot of $\boldsymbol{t}$ and the degree of the numerators and denominators in (3.6) that allow satisfying them. Using this information, from the formulation (3.5) we can compute the degree of F, which turns out to be equal to $5, 4, 3$ respectively for $r = 1, 2, 3$. This example emphasizes the remarkable fact that, by increasing r, it is possible to get a progressive reduction of degree, down to the minimum allowed for a local spline interpolant that is C^2-continuous.

Example 3.3 (B2 $D^3C^2P^1S^4$)**.** This class illustrates the situation in which, in order for F to be globally C^2, a different order of continuity of f_i must be required at some points of $\boldsymbol{\tau}$, according to Property 3.b in Section 3.2.

In the considered example, $w = 4$, $m = 1$ and, from (3.3) and (3.4), $n = 2$ and $\sigma = -1$. The interpolating spline F has the expression

$$F(x) = \sum_{i=\ell-2}^{\ell} p_{i+1,1}(x) B_{i,3}(x) = p_{\ell-1,1}(x) + \sum_{i=\ell-1}^{\ell} c_{i+1} f_i(x)(x - x_{i+1}), \qquad x \in I_\ell,$$

and the reciprocal configuration of interpolating polynomials and related blending/transition functions is illustrated in Figure 3.4. Thus we observe that, for any i, the data lying in the support of $B_{i,3}$ are (x_j, y_j), $j = i, \ldots, i+3$, but the polynomial $p_{i+1,1}$ only interpolates (x_j, y_j),



(a) **(b)**

Figure 3.5: Interpolating polynomials and associated (a) blending and (b) transition functions needed for the evaluation of the spline interpolant in Example 3.4 on the interval $I_\ell = [\tau_\ell, \tau_{\ell+1})$. Being the configuration odd, $\boldsymbol{\tau} \neq \boldsymbol{x}$.

$j = i+1, i+2$. This implies that (see Property 3.b in Section 3.2), in addition to continuity order C^1 at $\boldsymbol{\tau}$ and C^2 at $\boldsymbol{t}\backslash\boldsymbol{\tau}$, we also have to require C^2 continuity at the extremal points of the support of $B_{i,3}$, namely f_i must be C^2 at t_{2i} and $t_{2(i+2)}$. The complete expression of the corresponding transition functions is given in Section 3.3.1.

Example 3.4 (B4 $D^3C^2P^3S^5$)**.** Our aim is designing an optimal spline having odd support width. Thus, from the assumption that $\sigma = \sigma_c$ and Property 5 in Section 3.2, it follows that we need to work in the odd configuration (3.2b), that is $x_j \in (t_{4j-4}, t_{4j}) = (\tau_{j-1}, \tau_j)$, $\forall j$. From relations (3.3) and (3.4), we derive the values $n = 1$ and $\sigma = 0$, and accordingly the representation of F in the form (3.5):

$$F(x) = p_{\ell-1,3}(x) + f_\ell(x)\, \underbrace{(p_{\ell,3}(x) - p_{\ell-1,3}(x))}_{c_\ell(x-x_\ell)(x-x_{\ell+1})(x-x_{\ell+2})}, \qquad x \in I_\ell.$$

Figure 3.5 illustrates the polynomials and blending/transition functions involved in our construction on the interval I_ℓ.

Next, we set the proper continuity conditions needed to determine a transition function f_ℓ. At this point, it is convenient to make the particular assumption that $t_{4\ell+2} = x_{\ell+1}$, $\forall \ell$. In this way, in order for F to be C^2-continuous, we need to require that f_ℓ be C^2 at any point of $\boldsymbol{t}$ in $[t_{4\ell}, t_{4\ell+4}]$, but just C^1 at $t_{4\ell+2}$[2]. This gives us 14 continuity conditions against 14 unknown coefficients corresponding to proper degrees of the numerators in (3.6). For a detailed computation of f_ℓ we refer again to Section 3.3.1.

The four above examples suggest a general formulation for the considered class of Br-splines. In fact, if in equation (3.5) we substitute the expression (3.6) and simplify the denominator of the transition functions, the spline interpolant F becomes a piecewise polynomial of the form

$$F(x) = p_{\ell-n-\sigma,m}(x) + \sum_{i=\ell-n+1}^{\ell} c_{i-\sigma}\, K_{i,r(\ell-i)+h}(x)\, q_{i,r(\ell-i)+h}(x), \qquad x \in I_{\ell,h},\ h = 1,\ldots,r, \tag{3.8}$$

[2] In the odd configuration, it can be proved that, whenever, for any j, one of the internal knots t_{rj-h}, $h = -r+1,\ldots,1$ coincides with x_j, then F gains one order of continuity with respect to the continuity of the transition functions at the knot. The result can be obtained straightforwardly, verifying that the left and right derivatives of F at such knot are equal up to the order of continuity of the transition functions plus one.

where $q_{i,j}$ is the numerator of the jth piece of f_i. The term $K_{i,j}(x)$ contains the factors under the product symbol in (3.5) that are not simplified by the denominator of the transition function. Recalling that by construction the denominator of f_i is the product of all the factors among $(x - x_j), j = i - \sigma, \ldots, i - \sigma + m - 1$, that do not vanish in $I_{\ell,h}$, it turns out that $K_{i,j}(x)$ can be of the following types:
if $r = 1$,

$$K_{i,j}(x) = \kappa_{i,j} \begin{cases} 1 \\ (x - x_\ell) \\ (x - x_\ell)(x - x_{\ell+1}) \\ (x - x_{\ell+1}) \end{cases} \quad \text{in even configuration,}$$

or

$$K_{i,j}(x) = \kappa_{i,j} \begin{cases} 1 \\ (x - x_{\ell+1}) \end{cases} \quad \text{in odd configuration;}$$

otherwise (if $r \geqslant 2$),

$$K_{i,j}(x) = \kappa_{i,j} \begin{cases} 1 \\ (x - x_\ell) \\ (x - x_{\ell+1}) \end{cases} \quad \text{in even configuration,}$$

or

$$K_{i,j}(x) = \kappa_{i,j} \begin{cases} 1 \\ (x - x_{\ell+1}) \end{cases} \quad \text{in odd configuration.}$$

The term $\kappa_{i,j}$ is a multiplicative constant that depends on how we represent the denominator of the transition functions. Using the Bernstein basis, an example for the explicit form of $\kappa_{i,j}$ is given in Section 3.3.1, at the end of Example 3.1.

Expression (3.8) provides a useful and concise evaluation algorithm for the interpolants in the family (3.1). It also allows a direct computation of their derivatives. In particular, differentiating (3.8) s times and using Leibniz's formula, we get

$$F^{(s)}(x) = p^{(s)}_{\ell-n-\sigma,m}(x) + \sum_{i=\ell-n+1}^{\ell} c_{i-\sigma} \sum_{j=0}^{s} \binom{s}{j} K^{(j)}_{i,r(\ell-1)+h}(x)\, q^{(s-j)}_{i,r(\ell-1)+h}(x), \qquad x \in I_{\ell,h}.$$

The above formula can be further simplified in the particular case $n + 1 = 2$, as pointed out in the following remark.

Remark 3.1. If $n+1 = 2$ and the blending functions (or, equivalently, the transition functions) are C^k, then $F^{(s)}(\tau_\ell) = p^{(s)}_{\ell-1-\sigma,m}(\tau_\ell)$, $s = 0, \ldots, k$.

The assertion follows from (3.5), recalling that the transition functions are such that $f_\ell(\tau_\ell) = 0$ and $f_\ell(\tau_{\ell+1}) = 1$ (only the blending function $B_{\ell-1,2}$ is non-zero at τ_ℓ, and by partition of unity its value is 1), and the derivatives of f_ℓ (equivalently, of $B_{\ell,2}$) up to order k vanish at τ_ℓ and $\tau_{\ell+1}$.

Equation (3.8) also suggests some interesting observations about the degree of F. Denoted by $\bar{d} \coloneqq \max_{j=1,\ldots,rn} d_{i,j}$ the maximum degree of the numerators of f_i, it can be seen that, if $r = 1$, the lowest possible degree for F is

$$g = \begin{cases} \max(m, \bar{d} + 2), & \text{in even configuration,} \\ \max(m, \bar{d} + 1), & \text{in odd configuration;} \end{cases}$$

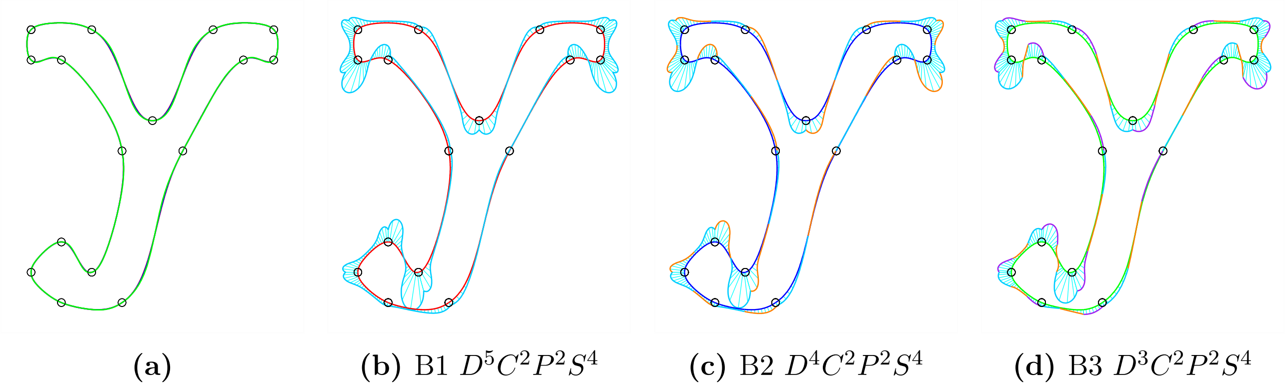


Figure 3.6: Comparison between B1-, B2-, and B3-spline parametric interpolants of planar data: (a) all interpolants superimposed, (b)–(d) interpolant and related curvature comb.

otherwise, namely if $r \geqslant 2$, the lowest degree turns out to be

$$g = \max(m, \bar{d} + 1).$$

Thus, Br-splines always allow us to get interpolants with lower degree with respect to the classical (i.e., B1) setting. Indeed we will show that *any* interpolating spline of minimum degree (i.e., corresponding to arbitrarily chosen smoothness, polynomial reproduction degree, and support width) can be obtained as a Br-spline for a suitable r.

Figure 3.6 illustrates three parametric interpolants (see Example 3.2), corresponding respectively to a B1-, B2-, and B3-spline, which provide a progressive reduction of degree. The figure emphasizes that curves featured by the same order of continuity, polynomial reproduction degree, and support width are very similar. Moreover, they all generate a faithful approximation of the original data, due to a proper choice of non-uniform parameterization (the centripetal parameterization is used in this example). In the curvature comb the different polynomial pieces between two breakpoints are highlighted in different colors: since the curve is C^2-continuous, the curvature is only C^0 at internal knots $\boldsymbol{t} \setminus \boldsymbol{\tau}$. In general we observe that at internal knots $\boldsymbol{t} \setminus \boldsymbol{\tau}$, a B$r$-spline is $C^{\bar{k}}$, while a B1-spline is C^{∞}-continuous.

A last consideration about the computational cost for the evaluation of a Br-spline is in order. There are two contributions in such an estimate: the cost of determining the subinterval $I_{\ell,h}$ of the refined sequence $\boldsymbol{t}$ and the evaluation of expression (3.8). However, since the degree of (3.8) is generally lower in the Br setting with $r \geqslant 2$, the overall computational cost may decrease with respect to the B1 case.

3.3.1 Expressions for the transition functions designed in Section 3.3

We give here the explicit expressions for the transition functions designed in Examples 3.1–3.4 of the previous section.

In the following, we express each numerator $q_{i,j}$ in (3.6) as a degree-$d_{i,j}$ polynomial in the Bernstein basis using the notation

$$q_{i,j}(x) = q(x|a_0^{i,j}, \dots, a_{d_{i,j}}^{i,j}) := \sum_{k=0}^{d_{i,j}} a_k^{i,j} \binom{d_{i,j}}{k} R_{i+j-1}^{d_{i,j}-k}(x) L_{i+j-1}^{k}(x), \qquad x \in [t_{i+j-1}, t_{i+j}),$$

with

$$L_h(x) := \frac{x - t_h}{t_{h+1} - t_h} \qquad \text{and} \qquad R_h(x) := \frac{t_{h+1} - x}{t_{h+1} - t_h}.$$

Example 3.1 (B2 $D^3C^2P^3S^6$)**.** Since $r = 2$ and $n = 2$, the transition functions f_i are, for any i, piecewise rational of the form (3.6) made of $rn = 4$ pieces between t_{2i} and $t_{2(i+2)}$. The denominator of each piece is chosen so as to simplify as many factors as possible among $(x - x_j), j = i, \dots, i + 2$. Thus, recalling that we are in the even configuration $\boldsymbol{x} \equiv \boldsymbol{\tau} = \{t_{2j}\}$, we can write:

$$f_i(x) = \begin{cases} 0, & x < t_{2i}, \\ \dfrac{q(x|a_0^{i,1}, a_1^{i,1}, a_2^{i,1})}{R_{2i+1}(x)R_{2i+3}(x)}, & x \in [t_{2i}, t_{2i+1}), \\ \dfrac{q(x|a_0^{i,2}, a_1^{i,2}, a_2^{i,2})}{L_{2i}(x)R_{2i+3}(x)}, & x \in [t_{2i+1}, t_{2(i+1)}), \\ \dfrac{q(x|a_0^{i,3}, a_1^{i,3}, a_2^{i,3})}{L_{2i}(x)R_{2i+3}(x)}, & x \in [t_{2(i+1)}, t_{2i+3}), \\ \dfrac{q(x|a_0^{i,4}, a_1^{i,4}, a_2^{i,4})}{L_{2i}(x)L_{2i+2}(x)}, & x \in [t_{2i+3}, t_{2(i+2)}), \\ 1, & x \geqslant t_{2(i+2)}. \end{cases}$$

The coefficients $a_h^{i,j}$ of the numerators are determined by solving the 12×12 linear system obtained by imposing C^1 smoothness at $\boldsymbol{\tau}$ and C^2 smoothness at $\boldsymbol{t} \backslash \boldsymbol{\tau}$, which has the unique solution

$$\begin{aligned}
a_0^{i,1} &= a_1^{i,1} = 0, \qquad a_2^{i,1} = a_0^{i,2} = -\frac{(t_{2i} - t_{2i+1})(t_{2i+1} - t_{2i+2})\eta_{2i}}{3(t_{2i} - t_{2i+2})\gamma_{2i}}, \\
a_1^{i,2} &= -\frac{(t_{2i+1} - t_{2i+2})\eta_{2i}}{2\gamma_{2i}}, \qquad a_2^{i,2} = a_0^{i,3} = \frac{(t_{2i} - t_{2i+2})(t_{2i+2} - t_{2i+1})\eta_{2i}}{(t_{2i} - t_{2i+1})\gamma_{2i}}, \\
a_1^{i,3} &= \frac{(t_{2i}\alpha_{2i} + \beta_{2i})\eta_{2i}}{2(t_{2i} - t_{2i+1})\gamma_{2i}}, \\
a_2^{i,3} &= a_0^{i,4} = \frac{t_{2i+4} - t_{2i+3}}{3(t_{2i} - t_{2i+1})(t_{2i+2} - t_{2i+4})\gamma_{2i}} \big[\big(-3t_{2i+2}^3 - (t_{2i+2} + 2t_{2i+3})t_{2i+4}^2 + \zeta_{2i}t_{2i+4} \\
&\quad +t_{2i+1}\left(6t_{2i+2}^2 - 4\left(t_{2i+3} + 2t_{2i+4}\right)t_{2i+2} + t_{2i+3}^2 + 3t_{2i+4}^2 + 2t_{2i+3}t_{2i+4}\right)\big)\, t_{2i}^2 \\
&\quad + \big(-\left(t_{2i+2}^2 - 3t_{2i+3}^2 + 2\left(3t_{2i+1} - 2t_{2i+2}\right)t_{2i+3}\right)t_{2i+4}^2 + \big((t_{2i+1} + 3t_{2i+2})t_{2i+2}^2 \\
&\quad +14(t_{2i+1} - t_{2i+2})t_{2i+3}t_{2i+2} - \left(3t_{2i+1} + t_{2i+2}\right)t_{2i+3}^2\big)\, t_{2i+4} + t_{2i+2}\big(6t_{2i+3}t_{2i+2}^2 \\
&\quad -t_{2i+1}\zeta_{2i}\big)\big)\, t_{2i} - \left(t_{2i+2}^3 - 4t_{2i+3}t_{2i+2}^2 - 3\left(t_{2i+1} - 2t_{2i+2}\right)t_{2i+3}^2\right)t_{2i+4}^2 \\
&\quad +t_{2i+2}^2t_{2i+3}(t_{2i+1}(2t_{2i+2} + t_{2i+3}) - 3t_{2i+2}t_{2i+3}) + t_{2i+2}\big(t_{2i+1}t_{2i+2}^2 - 2\big(2t_{2i+1} \\
&\quad +t_{2i+2}\big)\, t_{2i+3}t_{2i+2} + (8t_{2i+2} - 3t_{2i+1})t_{2i+3}^2\big)\, t_{2i+4}\big], \\
a_1^{i,4} &= \frac{-t_{2i}\alpha_{2i+1} - \beta_{2i+1}}{2(t_{2i} - t_{2i+1})(t_{2i+2} - t_{2i+3})}, \qquad a_2^{i,4} = \frac{(t_{2i} - t_{2i+4})(t_{2i+2} - t_{2i+4})}{(t_{2i} - t_{2i+1})(t_{2i+2} - t_{2i+3})},
\end{aligned}$$

where

$$\begin{aligned}
\alpha_i &:= -2t_{i+1} + t_{i+2} + t_{i+3}, \\
\beta_i &:= -2t_{i+2}t_{i+3} + t_{i+1}(t_{i+2} + t_{i+3}), \\
\gamma_i &:= (t_{i+3} - t_{i+4})\left((t_{i+1} - t_{i+3})t_{i+2}^2 - \left(t_{i+2}^2 + (t_{i+1} - 2t_{i+2})t_{i+3}\right)t_{i+4}\right. \\
&\qquad \left. +t_i\left(t_{i+2}^2 - t_{i+3}t_{i+4} + t_{i+1}\alpha_{i+1}\right)\right), \\
\eta_i &:= (t_i - t_{i+3})(t_{i+2} - t_{i+4})^2, \\
\zeta_i &:= 3t_{i+2}^2 + 4t_{i+3}t_{i+2} - t_{i+3}^2.
\end{aligned}$$

For completeness, we provide the expression of the terms $K_{i,j}$ needed to write F in the form (3.8) on the interval I_ℓ:

$$K_{\ell-1,3}(x) = \kappa_{\ell-1,3}(x - x_\ell), \qquad K_{\ell-1,4}(x) = \kappa_{\ell-1,4}(x - x_{\ell+1}),$$
$$K_{\ell,1}(x) = \kappa_{\ell,1}(x - x_\ell), \qquad K_{\ell,2}(x) = \kappa_{\ell,2}(x - x_{\ell+1}),$$

with constants

$$\kappa_{\ell-1,3} = -(t_{2\ell-1} - t_{2\ell-2})(t_{2\ell+2} - t_{2\ell+1}), \qquad \kappa_{\ell-1,4} = (t_{2\ell-1} - t_{2\ell-2})(t_{2\ell+1} - t_{2\ell}),$$
$$\kappa_{\ell,1} = (t_{2\ell+2} - t_{2\ell+1})(t_{2\ell+4} - t_{2\ell+3}), \qquad \kappa_{\ell,2} = -(t_{2\ell+1} - t_{2\ell})(t_{2\ell+4} - t_{2\ell+3}).$$

Example 3.2 (B1 $D^5C^2P^2S^4$, B2 $D^4C^2P^2S^4$, B3 $D^3C^2P^2S^4$)**.** We derive the expressions of a generic transition function f_i in (3.7), in the cases Br, $r = 1, 2, 3$. We refer to Figure 3.3 for the count of degrees of the numerators and continuity conditions which is exploited in the three cases below.

$r = 1$: f_i is defined on $\boldsymbol{t} := \{t_j\} \equiv \boldsymbol{\tau}$. Based on (3.6), we write

$$f_i(x) = \begin{cases} 0, & x < t_i, \\ q(x|a_0^{i,1}, a_1^{i,1}, a_2^{i,1}, a_3^{i,1}), & x \in [t_i, t_{i+1}), \\ 1, & x \geqslant t_{i+1}. \end{cases}$$

Note that, for any i, f_i is polynomial in $[t_i, t_{i+1})$: in fact, none of the terms $(x - x_i)$, $(x - x_{i+1})$ (which we may want to simplify, see (3.5)) can be set as a denominator without generating a singularity.

We determine the coefficients $a_h^{i,1}$, $h = 0, \dots, 3$, by solving the linear system arising from the C^1 continuity constraint at t_i and t_{i+1}. In this way, we get

$$a_0^{i,1} = a_1^{i,1} = 0, \qquad a_2^{i,1} = a_3^{i,1} = 1.$$

$r = 2$: f_i is piecewise rational on the sequence $\boldsymbol{t} := \{t_j\}$, such that $t_{2j} = \tau_j$, $\forall j$. In fact, according to (3.6), we can express f_i as

$$f_i(x) = \begin{cases} 0, & x < t_{2i}, \\ \dfrac{q(x|a_0^{i,1}, a_1^{i,1}, a_2^{i,1}, a_3^{i,1})}{R_{2i+1}(x)}, & x \in [t_{2i}, t_{2i+1}), \\ \dfrac{q(x|a_0^{i,2}, a_1^{i,2}, a_2^{i,2}, a_3^{i,2})}{L_{2i}(x)}, & x \in [t_{2i+1}, t_{2(i+1)}), \\ 1, & x \geqslant t_{2(i+1)}. \end{cases}$$

The denominator of f_i is a multiple of $(x - x_i)$ in $[t_{2i}, t_{2i+1})$ and of $(x - x_{i+1})$ in $[t_{2i+1}, t_{2(i+1)})$, in such a way that f_i is never singular on $\mathbb{R}$. To determine the coefficients of the numerators in the above formula, we require C^1 continuity at $t_{2i}, t_{2(i+1)}$ and C^2 at t_{2i+1}. This provides 7 constraints, against 8 unknowns. As additional constraint, we require that f_i be antisymmetric with respect to t_{2i+1}, namely $f_i(t_{2i+1}) = \frac{t_{2i+1} - t_{2i}}{t_{2(i+1)} - t_{2i}}$. This choice results in the coefficients

$$a_0^{i,1} = a_1^{i,1} = 0, \qquad a_2^{i,1} = \frac{1}{9}(t_{2i+1} - t_{2i}) \left(\frac{3}{t_{2i+2} - t_{2i+1}} + \frac{1}{t_{2i+2} - t_{2i}} \right),$$

$$a_3^{i,1} = a_0^{i,2} = \frac{t_{2i} - t_{2i+1}}{t_{2i} - t_{2i+2}}, \qquad a_1^{i,2} = \frac{1}{9}\left(\frac{t_{2i} - t_{2i+1}}{t_{2i} - t_{2i+2}} + 8\right),$$
$$a_2^{i,2} = -\frac{-3t_{2i} + t_{2i+1} + 2t_{2i+2}}{3(t_{2i} - t_{2i+1})}, \qquad a_3^{i,2} = \frac{t_{2i} - t_{2i+2}}{t_{2i} - t_{2i+1}}.$$

$r = 3$: f_i is piecewise rational on the sequence $\boldsymbol{t} := \{t_j\}$, such that $t_{3j} = \tau_j$, $\forall j$. In this case, from (3.6), we have

$$f_i(x) = \begin{cases} 0, & x < t_{3i}, \\ \dfrac{q(x|a_0^{i,1}, a_1^{i,1}, a_2^{i,1})}{R_{3i+2}(x)}, & x \in [t_{3i}, t_{3i+1}), \\ \dfrac{q(x|a_0^{i,2}, a_1^{i,2}, a_2^{i,2}, a_3^{i,2})}{L_{3i}(x)R_{3i+2}(x)}, & x \in [t_{3i+1}, t_{3i+2}), \\ \dfrac{q(x|a_0^{i,3}, a_1^{i,3}, a_2^{i,3})}{L_{3i}(x)}, & x \in [t_{3i+2}, t_{3(i+1)}), \\ 1, & x \geqslant t_{3(i+1)}. \end{cases}$$

Note that, in $[t_{3i+1}, t_{3i+2})$, the denominator of f_i contains both factors $(x - x_i)$ and $(x - x_{i+1})$, as none of them vanishes in that interval.

Requiring C^1 continuity at $t_{3i}, t_{3(i+1)}$ and C^2 at t_{3i+1}, t_{3i+2}, we get

$$a_0^{i,1} = a_1^{i,1} = 0, \qquad a_2^{i,1} = a_0^{i,2} = \frac{(t_{3i} - t_{3i+1})\alpha_{3i}}{3(t_{3i} - t_{3i+2})\beta_{3i}},$$
$$a_1^{i,2} = \frac{1}{3}\left(1 - \frac{(t_{3i} - t_{3i+2})(t_{3i+3} - t_{3i+1})}{\beta_{3i}}\right), \qquad a_2^{i,2} = \frac{(t_{3i} - t_{3i+2})\alpha_{3i}}{3(t_{3i} - t_{3i+1})\beta_{3i}},$$
$$a_3^{i,2} = a_0^{i,3} = \frac{1}{3}\left(\frac{(t_{3i} - t_{3i+2})^2}{(t_{3i} - t_{3i+1})(t_{3i} - t_{3i+3})} + \frac{t_{3i} + t_{3i+1} - 2t_{3i+2}}{t_{3i} - t_{3i+1}} + \frac{t_{3i+1} - t_{3i+2}}{t_{3i+1} - t_{3i+3}}\right),$$
$$a_1^{i,3} = -\frac{-2t_{3i} + t_{3i+2} + t_{3i+3}}{2(t_{3i} - t_{3i+1})}, \qquad a_2^{i,3} = \frac{t_{3i} - t_{3i+3}}{t_{3i} - t_{3i+1}},$$

where

$$\alpha_i := t_{i+3}^2 - t_{i+1}t_{i+2} + t_i(t_{i+1} + t_{i+2} - 2t_{i+3}), \qquad \beta_i := (t_i - t_{i+3})(t_{i+2} - t_{i+3}).$$

Example 3.3 (B2 $D^3C^2P^1S^4$)**.** Since $r = 2$ and $n = 2$, the transition functions f_i are piecewise rational made of $rn = 4$ pieces between t_{2i} and $t_{2(i+2)}$. In order to minimize the degree of F, the denominator of each piece is chosen so as to simplify the factor $(x - x_{i+1})$, if possible. Pursuing this principle, (3.6) can be rewritten as

$$f_i(x) = \begin{cases} 0, & x < t_{2i}, \\ \dfrac{q(x|a_0^{i,1}, a_1^{i,1}, a_2^{i,1}, a_3^{i,1})}{R_{2i+1}(x)}, & x \in [t_{2i}, t_{2i+1}), \\ q(x|a_0^{i,2}, a_1^{i,2}, a_2^{i,2}), & x \in [t_{2i+1}, t_{2(i+1)}), \\ q(x|a_0^{i,3}, a_1^{i,3}, a_2^{i,3}), & x \in [t_{2(i+1)}, t_{2i+3}), \\ \dfrac{q(x|a_0^{i,4}, a_1^{i,4}, a_2^{i,4}, a_3^{i,4})}{L_{2i+2}(x)}, & x \in [t_{2i+3}, t_{2(i+2)}), \\ 1, & x \geqslant t_{2(i+2)}. \end{cases}$$

The coefficients $a_h^{i,j}$ are determined by solving the 14×14 linear system obtained by imposing C^1 continuity at $t_{2i}, t_{2(i+1)}, t_{2(i+2)}$, and C^2 continuity at t_{2i+1}, t_{2i+3}, and, recalling Property 3.b in Section 3.2, at $t_{2i}, t_{2(i+2)}$ as well. Their computation provides

$$\begin{aligned}
a_0^{i,1} &= a_1^{i,1} = a_2^{i,1} = 0, \qquad a_3^{i,1} = a_0^{i,2} = -\frac{(t_{2i} - t_{2i+1})^2 (t_{2i+1} - t_{2i+2})\alpha_{2i}}{3(t_{2i} - t_{2i+2})\beta_{2i}}, \\
a_1^{i,2} &= -\frac{(t_{2i} - t_{2i+1})(t_{2i+1} - t_{2i+2})\alpha_{2i}}{2\beta_{2i}}, \qquad a_2^{i,2} = a_0^{i,3} = \frac{(t_{2i} - t_{2i+2})(t_{2i+2} - t_{2i+1})\alpha_{2i}}{\beta_{2i}}, \\
a_1^{i,3} &= -\frac{\left((t_{2i+1} - t_{2i+2})(t_{2i} - t_{2i+3}) + (t_{2i+1} - t_{2i+3})(t_{2i} - t_{2i+2})\right)\alpha_{2i}}{2\beta_{2i}}, \\
a_2^{i,3} &= a_0^{i,4} = -\frac{1}{3(t_{2i+2} - t_{2i+4})\beta_{2i}} \left[t_{2i+2}\left(-(t_{2i+2} - 4t_{2i+3})t_{2i+4}^2\right.\right. \\
&\quad \left. + (t_{2i+2} - 4t_{2i+3})(3t_{2i+2} - t_{2i+3})t_{2i+4} + t_{2i+3}\left(3t_{2i+2}^2 + 2t_{2i+3}t_{2i+2} - 2t_{2i+3}^2\right)\right) \\
&\quad \left. + t_{2i+1}\gamma_{2i} + t_{2i}\left(\gamma_{2i} + 3t_{2i+1}(t_{2i+2} - t_{2i+4})\alpha_{2i}\right)\right], \\
a_1^{i,4} &= -\frac{-3t_{2i+2} + 2t_{2i+3} + t_{2i+4}}{3(t_{2i+2} - t_{2i+3})}, \quad a_2^{i,4} = -\frac{-3t_{2i+2} + t_{2i+3} + 2t_{2i+4}}{3(t_{2i+2} - t_{2i+3})}, \quad a_3^{i,4} = \frac{t_{2i+2} - t_{2i+4}}{t_{2i+2} - t_{2i+3}},
\end{aligned}$$

where

$$\begin{aligned}
\alpha_i &:= 2t_{i+2} - t_{i+3} - t_{i+4}, \\
\beta_i &:= (t_{i+1} - t_{i+3})t_{i+2}^2 - \left(t_{i+2}^2 + (t_{i+1} - 2t_{i+2})t_{i+3}\right) t_{i+4} + t_i \left(t_{i+2}^2 - t_{i+3}t_{i+4} - t_{i+1}\alpha_i\right), \\
\gamma_i &:= -3t_{i+2}^3 - t_{i+3}^2 t_{i+2} + t_{i+3}^3 - (t_{i+2} + 2t_{i+3})t_{i+4}^2 + (3t_{i+2} - t_{i+3})(t_{i+2} + 2t_{i+3})t_{i+4}.
\end{aligned}$$

Example 3.4 (B4 $D^3C^2P^3S^5$)**.** In the case of odd configuration, in addition to the previous definitions of $L_h(x)$ and $R_h(x)$, we set

$$\tilde{L}_h(x) := x - x_h \qquad \text{and} \qquad \tilde{R}_h(x) := x_{h+1} - x.$$

Since $r = 4$ and $n = 1$, the transition functions f_i are piecewise rational made of $rn = 4$ pieces between t_{4i} and t_{4i+4}. Recalling the assumption that $t_{4i+2} = x_{i+1}$, the expression (3.6) for f_i has the form

$$f_i(x) = \begin{cases}
0, & x < t_{4i}, \\
\dfrac{q(x|a_0^{i,1}, a_1^{i,1}, a_2^{i,1}, a_3^{i,1})}{\tilde{L}_i(x)\tilde{R}_i(x)\tilde{R}_{i+1}(x)}, & x \in [t_{4i}, t_{4i+1}), \\
\dfrac{q(x|a_0^{i,2}, a_1^{i,2}, a_2^{i,2})}{\tilde{L}_i(x)\tilde{R}_{i+1}(x)}, & x \in [t_{4i+1}, t_{4i+2}), \\
\dfrac{q(x|a_0^{i,3}, a_1^{i,3}, a_2^{i,3})}{\tilde{L}_i(x)\tilde{R}_{i+1}(x)}, & x \in [t_{4i+2}, t_{4i+3}), \\
\dfrac{q(x|a_0^{i,4}, a_1^{i,4}, a_2^{i,4}, a_3^{i,4})}{\tilde{L}_i(x)\tilde{L}_{i+1}(x)\tilde{R}_{i+1}(x)}, & x \in [t_{4i+3}, t_{4(i+1)}), \\
1, & x \geqslant t_{4(i+1)}.
\end{cases}$$

The formulation above provides all the degrees of freedom necessary to satisfy the constraints for C^1 continuity at t_{4i+2} and C^2 continuity at the other points of $\boldsymbol{t}$. The coefficients $a_h^{i,j}$ found as solution of the resulting 14×14 linear system are

$$a_0^{i,1} = a_1^{i,1} = a_2^{i,1} = 0, \quad a_3^{i,1} = -\frac{\alpha_i \gamma_i^2}{3(t_{4i} - x_{i+1})\beta_i},$$

$$
\begin{aligned}
a_0^{i,2} &= \frac{(t_{4i}-t_{4i+1})\alpha_i\gamma_i}{3(t_{4i}-x_{i+1})\beta_i}, \quad a_1^{i,2} = \frac{\alpha_i\gamma_i}{2\beta_i}, \quad a_2^{i,2} = a_0^{i,3} = \frac{\alpha_i(t_{4i}-x_{i+1})(t_{4i+1}-x_{i+1})}{\beta_i}, \\
a_1^{i,3} &= \frac{(t_{4i}(2t_{4i+1}-t_{4i+3}-x_{i+1}) + 2t_{4i+3}x_{i+1} - t_{4i+1}(t_{4i+3}+x_{i+1}))\alpha_i}{2\beta_i}, \quad a_2^{i,3} = \frac{\omega_i}{\eta_i}, \\
a_0^{i,4} &= (t_{4i+3}-x_{i+1})\frac{\omega_i}{\eta_i}, \quad a_1^{i,4} = \frac{1}{3}\left(t_{4i+3}^2(\xi_i - 3t_{4i+4}) + \vartheta_i - \rho_i(2t_{4i+3}+t_{4i+4})\right), \\
a_2^{i,4} &= \frac{1}{3}\left(t_{4i+4}^2(\xi_i - 3t_{4i+3}) + \vartheta_i - \rho_i(2t_{4i+4}+t_{4i+3})\right), \\
a_3^{i,4} &= -(t_{4i+4}-x_i)(t_{4i+4}-x_{i+1})(t_{4i+4}-x_{i+2}),
\end{aligned}
$$

where

$$
\begin{aligned}
\alpha_i &\coloneqq x_{i+1}^2(t_{4i+4}-x_i) - x_{i+2}(t_{4i+4}x_i + x_{i+1}(x_{i+1}-2x_i)) \\
&\quad + t_{4i+3}\left(t_{4i+4}(x_i - 2x_{i+1} + x_{i+2}) + x_{i+1}^2 - x_i x_{i+2}\right), \\
\beta_i &\coloneqq x_{i+1}(2t_{4i+3}t_{4i+4} - (t_{4i+3}+t_{4i+4})x_{i+1}) + t_{4i+1}\left(x_{i+1}^2 - t_{4i+3}t_{4i+4}\right) \\
&\quad + t_{4i}\left(t_{4i+1}(t_{4i+3}+t_{4i+4}-2x_{i+1}) - t_{4i+3}t_{4i+4} + x_{i+1}^2\right), \\
\gamma_i &\coloneqq (t_{4i}-t_{4i+1})(t_{4i+1}-x_{i+1}), \\
\eta_i &\coloneqq 3(t_{4i+4}-x_{i+1})\big((-t_{4i+1}+t_{4i+3}+t_{4i+4})x_{i+1}^2 - 2t_{4i+3}t_{4i+4}x_{i+1} + t_{4i+1}t_{4i+3}t_{4i+4} \\
&\quad -t_{4i}\left(x_{i+1}^2 - t_{4i+3}t_{4i+4} + t_{4i+1}(t_{4i+3}+t_{4i+4}-2x_{i+1})\right)\big), \\
\zeta_i &\coloneqq \left(3t_{4i+4}^2 - 3x_{i+1}t_{4i+4} + x_{i+1}^2 - x_i x_{i+2}\right)t_{4i+3}^3 + \big(2x_{i+1}^3 + x_i x_{i+2}x_{i+1} - 3t_{4i+4}^2(x_i+x_{i+2}) \\
&\quad +t_{4i+4}\left((3x_i - 5x_{i+1})x_{i+1} + (2x_i + 3x_{i+1})x_{i+2}\right)\big)t_{4i+3}^2 + \big(-3(x_i+x_{i+2})x_{i+1}^3 \\
&\quad +t_{4i+4}\left(2x_{i+1}^2 + 3x_i x_{i+1} + 3x_{i+2}x_{i+1} - 5x_i x_{i+2}\right)x_{i+1} + t_{4i+4}^2\left(x_{i+1}^2 + 2x_i x_{i+2}\right)\big)t_{4i+3} \\
&\quad - t_{4i+4}^2 x_{i+1}^3 + x_i x_{i+1}\left(t_{4i+4}^2 - 3x_{i+1}t_{4i+4} + 3x_{i+1}^2\right)x_{i+2}, \\
\xi_i &\coloneqq x_i + x_{i+1} + x_{i+2}, \\
\vartheta_i &\coloneqq 2t_{4i+3}t_{4i+4}\xi_i + 3x_i x_{i+1}x_{i+2}, \\
\rho_i &\coloneqq x_i x_{i+1} + (x_i + x_{i+1})x_{i+2}, \\
\omega_i &\coloneqq -t_{4i+1}\zeta_i + x_{i+1}\big(\big(6t_{4i+4}^2 - 9x_{i+1}t_{4i+4} + 3x_{i+1}^2 + x_i x_{i+1} + (x_{i+1}-2x_i)x_{i+2}\big)t_{4i+3}^3 \\
&\quad + \big(-3(2x_i + x_{i+1} + 2x_{i+2})t_{4i+4}^2 + (x_{i+1}(7x_i + 3x_{i+1}) + (4x_i + 7x_{i+1})x_{i+2})t_{4i+4} \\
&\quad +2x_{i+1}((x_i - 2x_{i+1})x_{i+2} - 2x_i x_{i+1})\big)t_{4i+3}^2 + (4t_{4i+4} - x_{i+1})\big(t_{4i+4}\rho_i \\
&\quad -3x_i x_{i+1}x_{i+2}\big)t_{4i+3} + t_{4i+4}x_{i+1}(3x_i x_{i+1}x_{i+2} - t_{4i+4}\rho_i)\big) + t_{4i}\big(-\zeta_i + t_{4i+1}\big(\big(3t_{4i+4} - x_i \\
&\quad -x_{i+1} - x_{i+2}\big)t_{4i+3}^3 + (t_{4i+4} - 4x_{i+1})(3t_{4i+4} - x_i - x_{i+1} - x_{i+2})t_{4i+3}^2 \\
&\quad + \big(2(-2x_i + x_{i+1} - 2x_{i+2})t_{4i+4}^2 + (x_{i+1}(7x_i + 4x_{i+1}) + (3x_i + 7x_{i+1})x_{i+2}\big)t_{4i+4} \\
&\quad -3x_{i+1}\left(2x_i x_{i+1} + (x_i + 2x_{i+1})x_{i+2}\right)\big)t_{4i+3} + 6x_i x_{i+1}^2 x_{i+2} - 9t_{4i+4}x_i x_{i+1}x_{i+2} \\
&\quad +t_{4i+4}^2\left((x_i - 2x_{i+1})x_{i+1} + (3x_i + x_{i+1})x_{i+2}\right)\big)\big),
\end{aligned}
$$

3.4 Minimum-degree Br-spline interpolants

Based on the previous discussion and on Properties 1–5 in Section 3.2, it can be seen that the formulation (3.8) allows designing splines featured by any *feasible* quadruple of design parameters $(g, \bar{k}, m, w)$. More precisely, $\bar{k}$ can assume any arbitrary value, while the values of m and w must satisfy a reciprocal relation given by the necessary interpolation condition expressed by Property 1, which determines the bounds $\lceil \frac{w-2}{2} \rceil \leqslant m \leqslant w-1$ in case of even configuration and $\lceil \frac{w-1}{2} \rceil \leqslant m \leqslant w-1$ in odd configuration. The degree is bounded below by

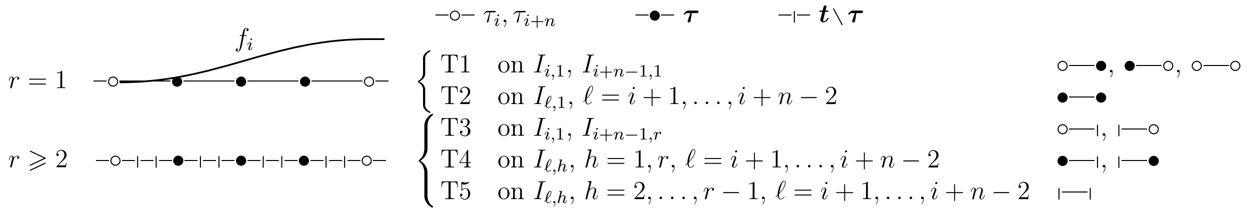

τ_i, τ_{i+n}
$\boldsymbol{\tau}$
$\boldsymbol{t} \setminus \boldsymbol{\tau}$
f_i
$r = 1$
$r \geqslant 2$
T1 on $I_{i,1}$, $I_{i+n-1,1}$
T2 on $I_{\ell,1}$, $\ell = i+1, \dots, i+n-2$
T3 on $I_{i,1}$, $I_{i+n-1,r}$
T4 on $I_{\ell,h}$, $h = 1, r$, $\ell = i+1, \dots, i+n-2$
T5 on $I_{\ell,h}$, $h = 2, \dots, r-1$, $\ell = i+1, \dots, i+n-2$

Figure 3.7: Classification of the types of pieces of each transition function based on the configuration of the endpoints of the related interval of definition.

$g \geqslant \max\bigl(m, \bar{k}+1\bigr)$. In fact, intuitively, a polynomial spline interpolant has minimum degree when there could not exist another spline with same continuity, polynomial reproduction, and support, but lower degree. Thus its degree must be equal either to the polynomial reproduction degree or to the order of continuity plus one (as, e.g., in the case of global spline interpolation). More formally, we introduce the definition:

Definition 3.2 (Minimum-degree spline)**.** We say that a polynomial spline interpolant $D^g C^{\bar{k}} P^m S^w$ has *minimum degree* if $g = \max\bigl(m, \bar{k}+1\bigr)$.

In this section we show that, for any feasible choice of degree, order of continuity, degree of polynomial reproduction, and support width, our framework always allows us to construct a local Br-spline interpolant having the prescribed parameters. Successively, we exploit this result to investigate in more detail the class of minimum-degree splines that belong to the considered framework.

For the sake of conciseness we address in detail the even setting (3.2a) only, while the case of odd configuration (3.2b) can be analyzed following the same outline. Hereinafter we suppose $n \geqslant 1$; the case $n = 0$ is addressed separately at the end of this section. We introduce a preliminary lemma and successively we prove the main result.

Lemma 3.1. *Let F be a Br-spline interpolant of the form* (3.8) *belonging to the class $D^g C^{\bar{k}} P^m S^w$ and assume the even configuration $\boldsymbol{\tau} \equiv \boldsymbol{x}$. Then the numerators $q_{i,j}$ of the related transition functions f_i satisfy*

$$\sum_{j=1}^{rn} \deg(q_{i,j}) \leqslant \begin{cases} n(rg-2), & \textit{if } m+1 > n, \\ n(rg-2)+2, & \textit{if } m+1 = n. \end{cases} \tag{3.9}$$

Proof. We focus on a single transition function f_i in the form (3.6). Equation (3.8) and the fact that F has degree g imply that

$$\deg(q_{i,j}) \leqslant g - \deg(K_{i,j}), \qquad j = 1, \dots, rn. \tag{3.10}$$

Since $\pi_{i,j}$ is assumed to be the product of all factors $(x - x_h)$, $h = i-\sigma, \dots, i-\sigma+m-1$, that do not vanish on $[t_{ri+j-1}, t_{ri+j}]$, the degree of $K_{i,j}$ represents how many factors of the product in (3.5) remain after simplification by $\pi_{i,j}$.

To compute $\deg(K_{i,j})$ we distinguish 5 types of pieces, according to the configuration of the endpoints of the interval of definition; they are schematically represented in Figure 3.7. A simple count allows us to determine $\deg(K_{i,j})$ and thus evaluate the right-hand side of condition (3.10). The related upper bounds for $\deg(q_{i,j})$ are listed in the third and fourth columns of Table 3.1, according to the type of piece. Note that the cases $m+1 > n$ and $m+1 = n$ must be treated separately. This depends on the fact that, in the latter case, by Property 1 in Section 3.2, we always have $\sigma = -1$ and, as a consequence, it is always possible to simplify one more factor $(x - x_h)$ with respect to the former case (see also Example 3.3 in Section 3.3).

refinement coefficient	type of piece	upper bound for $\deg(q_{i,j})$		number of pieces in f_i
		$m+1>n$	$m+1=n$	
$r=1$	T1	$g-2$	$g-1$	2 if $n \geqslant 2$ 1 if $n=1$
	T2	$g-2$	$g-2$	$n-2$ if $n \geqslant 2$ 0 if $n=1$
$r \geqslant 2$	T3	$g-1$	g	2
	T4	$g-1$	$g-1$	$2n-2$
	T5	g	g	$n(r-2)$

Table 3.1: Upper bounds for $\deg(q_{i,j})$ and number of different types of pieces for the transition functions f_i, based on the classification in Figure 3.7.

The refinement coefficient r indicates how many pieces of each type appear in f_i (see again Figure 3.7 and the rightmost column of Table 3.1). Now, taking into account the maximum degree for $q_{i,j}$ allowed per piece and the total number of pieces of each type, relation (3.9) can be obtained straightforwardly. □

The left-hand side of (3.9) determines how many degrees of freedom are provided by the numerators of the pieces of the transition functions in (3.6) and which can be spent to ensure that any f_i has the desired continuity. Thus, for a given degree g, the maximum number of degrees of freedom (#DoF) is obtained from (3.9) with the equality sign and gives

$$\#\text{DoF} = \begin{cases} n(r(g+1)-2), & \text{if} \quad m+1>n, \\ n(r(g+1)-2)+2, & \text{if} \quad m+1=n. \end{cases} \tag{3.11}$$

Since this estimate depends on r, we now determine which is the smallest value r such that #DoF is greater than or equal to the total number of continuity conditions at the points of $\boldsymbol{t}$ involved in the definition of f_i.

Proposition 3.2. *Let F be a spline interpolant of the form* (3.8)*, in even configuration $\boldsymbol{\tau} \equiv \boldsymbol{x}$, with continuity $\bar{k}$, polynomial reproduction m, and support w. The smallest value of r guaranteeing that F has given degree $g \geqslant \max(m, \bar{k}+1)$ is*

$$r = \left\lceil \frac{\bar{k}+w-m-1}{(w-m-1)(g-\bar{k})} \right\rceil . \tag{3.12}$$

Proof. In order to determine r we need to require that, for any transition function f_i, the total number of continuity conditions #CC at the points of $\boldsymbol{t}$ contained in $[\tau_i, \tau_{i+n}]$ be smaller than or equal to the total number of degrees of freedom #DoF inferred from (3.6) (represented by the coefficients of the numerators $q_{i,j}$).

We distinguish the two cases $m+1>n$ and $m+1=n$.

- If $m+1>n$, then $\#\text{CC} = \bar{k}(n+1)+(\bar{k}+1)(r-1)n$. Thus, by imposing $\#\text{CC} \leqslant \#\text{DoF}$, with #DoF given by (3.11), and recalling that $n = w-m-1$, we get (3.12).
- If $m+1=n$, then $\#\text{CC} = \bar{k}(n+1)+(\bar{k}+1)(r-1)n+2$ and (3.12) is obtained by proceeding analogously to the previous case.

□

Following the outline of the proofs of Lemma 3.1 and Proposition 3.2, we can derive analogous results for the odd configuration (3.2b). In this setting, the counterpart of (3.9) turns out to be

$$\sum_{j=1}^{rn} \deg(q_{i,j}) \leqslant \begin{cases} n(rg-2), & \text{if } r \text{ is even,} \\ n(rg-1), & \text{if } r \text{ is odd,} \end{cases}$$

and (3.12) is replaced by

$$r = \left\lceil \frac{\bar{k} + w - m}{(w-m-1)(g-\bar{k})} \right\rceil . \tag{3.13}$$

Now, since the above results hold for any degree g, by setting $g = \max(m, \bar{k}+1)$ and substituting it in (3.12) or in (3.13) (according to the reciprocal configuration of $\boldsymbol{x}$ and $\boldsymbol{\tau}$), we can straightforwardly compute the suitable refinement coefficient r that is necessary for obtaining a minimum-degree spline with fixed continuity, polynomial reproduction, and support. In particular, in the case of optimal splines corresponding to $D^hC^{h-1}P^h$, (3.12) and (3.13) reduce respectively to the simpler expressions $r = \lceil \frac{w-2}{w-h-1} \rceil$ and $r = \lceil \frac{w-1}{w-h-1} \rceil$.

Tables 3.3(a), 3.3(b), and 3.3(c) show some centered, minimum-degree Br-spline interpolants with even configuration and support width 4, 6, 8 respectively that belong to the considered framework.

In particular, the continuity progressively increases moving across columns, up to any arbitrary order. At the same time, moving across rows, the polynomial reproduction degree m increases and n decreases according to the relation $w = m + n + 1$.

We observe that optimal splines, featured by $D^hC^{h-1}P^h$, are located in a diagonal fashion, in which respect the splines in the corresponding upper and lower diagonals are featured respectively by sub-optimal reproducibility ($D^hC^{h-1}P^{h-1}$) and sub-optimal continuity ($D^hC^{h-2}P^h$). Moreover, the upper and lower triangular parts of each table contain respectively minimum-degree splines with degree equal to $\bar{k}+1$ and m.

From the tables we notice that some splines share the same design parameters but the support width and have the same value of r. In this case, it is common understanding that one is likely to choose the class having smaller support. Thus the cell of the equivalent counterpart with wider support, which is outperformed, is left blank. It is also worth observing that when two of the listed splines have same continuity, polynomial reproduction, and degree, the one having the smallest support width exhibits the highest r (see, e.g., B3 $D^3C^2P^2S^4$ vs. B2 $D^3C^2P^2S^6$).

In the case $n = 0$ the spline interpolant F coincides on each interval $[\tau_\ell, \tau_{\ell+1})$ with the polynomial $p_{\ell-\sigma,m}$. Therefore, F can be directly designed as a B1-spline and trivially has minimum degree m. We also observe that, when $w = 2$ this family provides the only optimal spline which can be obtained with $r = 1$, namely the class B1 $D^1C^0P^1S^2$, which represents the piecewise linear interpolant of the given data.

To conclude this discussion, we point out that, to our knowledge, all the listed families of non-uniform sub-optimal splines have never appeared in the existing literature. Instead, as previously recalled, optimal splines were originally proposed in [DGM88], although only in the even configuration. That work investigates the two classes B2 $D^hC^{h-1}P^hS^{2h}$ and Bh $D^hC^{h-1}P^hS^{h+2}$ (see Theorems 3.2.1 and 4.2.1 respectively) and proves their uniqueness. Moreover, the corresponding Bh case for open data sets is explicitly calculated and analyzed in [DV12, Sections 10.5 and 10.6]. The same classes of [DGM88] can be also generated by our approach, which thus provides an effective method for their design and computation (see, e.g., Example 3.1 and Figure 3.1(a)).

Moreover, the class Bh $D^hC^{h-1}P^hS^{h+2}$ deserves a more detailed discussions. In fact, when h

$m, n+1$ \ $\bar{k}$	0	1	2	3	4
$1, 3$		B2 $D^2C^1P^1$ $\ast$	B2 $D^3C^2P^1$	B3 $D^4C^3P^1$	B3 $D^5C^4P^1$
$2, 2$	B1 $D^2C^0P^2$ $\diamond$	B2 $D^2C^1P^2$ $+$	B3 $D^3C^2P^2$ $\ast$	B4 $D^4C^3P^2$	B5 $D^5C^4P^2$
$3, 1$	B1 $D^3C^0P^3$	-	-	-	-

(a) $w = 4$

$m, n+1$ \ $\bar{k}$	0	1	2	3	4
$2, 4$			B2 $D^3C^2P^2$ $\ast$	B2 $D^4C^3P^2$	B3 $D^5C^4P^2$
$3, 3$		B1 $D^3C^1P^3$ $\diamond$	B2 $D^3C^2P^3$ $+$	B3 $D^4C^3P^3$ $\ast$	B3 $D^5C^4P^3$
$4, 2$	B1 $D^4C^0P^4$	B1 $D^4C^1P^4$	B2 $D^4C^2P^4$ $\diamond$	B4 $D^4C^3P^4$ $+$	B5 $D^5C^4P^4$ $\ast$
$5, 1$	B1 $D^5C^0P^5$	-	-	-	-

(b) $w = 6$

$m, n+1$ \ $\bar{k}$	0	1	2	3	4	5
$3, 5$				B2 $D^4C^3P^3$ $\ast$	B2 $D^5C^4P^3$	B3 $D^6C^5P^3$
$4, 4$			B1 $D^4C^2P^4$ $\diamond$	B2 $D^4C^3P^4$ $+$	B3 $D^5C^4P^4$ $\ast$	B3 $D^6C^5P^4$
$5, 3$		B1 $D^5C^1P^5$	B1 $D^5C^2P^5$	B2 $D^5C^3P^5$ $\diamond$	B3 $D^5C^4P^5$ $+$	B4 $D^6C^5P^5$ $\ast$
$6, 2$	B1 $D^6C^0P^6$	B1 $D^6C^1P^6$	B1 $D^6C^2P^6$	B2 $D^6C^3P^6$	B3 $D^6C^4P^6$ $\diamond$	B6 $D^6C^5P^6$ $+$
$7, 1$	B1 $D^7C^0P^7$	-	-	-	-	-

(c) $w = 8$

Table 3.2: Centered, local, interpolating Br-splines of support width (a) 4, (b) 6, (c) 8, and minimum degree. Optimal, sub-optimal reproducibility, sub-optimal continuity splines are highlighted respectively in red, green, and cyan, and labeled with $+$, $\ast$, $\diamond$. Blank cells correspond to classes that are outperformed by counterparts having the same value of r and the same design parameters but a smaller support width. (For interpretation of the references to color, the reader is referred to the web version of this thesis.)

is odd, the even configuration assumed in [DGM88] necessarily leads to fundamental functions that are non-centered (and thus non-symmetric, according to Property 5 in Section 3.2). Although the same functions could be constructed by our method as well, our approach can do even better. In fact, it also allows us to generate optimal fundamental functions of odd support that are centered and thus preserve possible symmetry of the data. Based on Property 5 in Section 3.2, these can be obtained in the odd configuration as B$(h+1)$ $D^hC^{h-1}P^hS^{h+2}$ splines (see Example 3.4). For instance, Figure 3.1(c) illustrates the fundamental function related to this class for $h = 3$.

In addition to this, from the tables we observe that, also in the even configuration, other new classes of optimal splines can be found in our framework, as, e.g., the class B3 $D^5C^4P^5S^8$ in Table 3.3(c).

3.5 Open spline interpolants with Hermite end conditions

In the following we extend the interpolation framework in order to design spline interpolants of an open set of data $\{(x_j, y_j) \in \mathbb{R}^2, j = 0, \ldots, N\}$. The following approach applies both to the Br-splines addressed here and to the B1 family of locally-supported interpolants in [BCR13a]. For this reason, differently with respect to the rest of this chapter, we address here the more general context in which $\sigma \in \Sigma$ (see Property 1 in Section 3.2) and not only the centered case (3.4).

Since F must be defined in the whole interval $[x_0, x_N]$, we assume to know a sequence of suitable length $\boldsymbol{\tau} = \{\tau_j, j = -\nu, \dots, N+n\}$ where $\nu = n$ in the even configuration and $\nu = n+1$ in the odd configuration, in such a way that all the blending functions $B_{i,n+1}$ that do not vanish in $[x_0, x_N]$ be well-defined.

The leading idea is that the definition of the spline interpolant F in the vicinity of the boundary be formally identical to (3.1) in such a way that

$$F(x) = \sum_{i=-\nu}^{-\nu+2n-1} p_{i-\sigma,m}(x) B_{i,n+1}(x), \qquad x \in \tilde{I}_{\mathrm{L}}, \tag{3.14a}$$

$$F(x) = \sum_{i=N-2n}^{N-1} p_{i-\sigma,m}(x) B_{i,n+1}(x), \qquad x \in \tilde{I}_{\mathrm{R}}, \tag{3.14b}$$

where $\tilde{I}_{\mathrm{L}}$ is either $[\tau_0, \tau_n]$ or $[\tau_{-1}, \tau_{n-1}]$ in the even or odd configuration respectively, and $\tilde{I}_{\mathrm{R}} := [\tau_{N-n}, \tau_N]$.

To define the polynomials $p_{i-\sigma,m}$ that appear in the above formulas, we extrapolate from the given set of data $\{(x_j, y_j), j = 0, \dots, N\}$ a suitable number of fictitious data to the left of (x_0, y_0) and to the right of (x_N, y_N). If the nodes $\boldsymbol{x}$ and the breakpoints $\boldsymbol{\tau}$ are in the even configuration $\boldsymbol{\tau} \equiv \boldsymbol{x}$, the additional nodes x_j, $j < 0, j > N$, regardless of their number, can be spaced with the same spacing of the corresponding breakpoints. Otherwise, i.e., in the odd configuration, the additional nodes can be chosen arbitrarily, provided that $x_j < \tau_j < x_{j+1}$. As a consequence, it only remains to establish how many fictitious data (x_j, y_j) need to be added on each side of the boundary and how the corresponding y_j, for $j < 0$ and for $j > N$, may be suitably chosen. The former issue is addressed by the following proposition.

Proposition 3.3. *For any open set of data, the spline interpolant F is completely defined*

(a) in the interval $\tilde{I}_{\mathrm{L}}$ by adding $\nu + \sigma$ interpolation data to the left of (x_0, y_0);

(b) in the interval $\tilde{I}_{\mathrm{R}}$ by adding $m - 1 - \sigma$ data to the right of (x_N, y_N).

Proof.

(a) It is sufficient to work out how many fictitious data are necessary in order to define the polynomial $p_{-\nu-\sigma,m}$ associated in (3.14a) with the leftmost blending function $B_{-\nu,n+1}$ that does not vanish in $\tilde{I}_{\mathrm{L}}$. By definition, $p_{j-\sigma,m}$ is the polynomial of degree m that interpolates the data $(x_{j-\sigma}, y_{j-\sigma}), \dots, (x_{j+m-\sigma}, y_{j+m-\sigma})$, thus we can conclude that $p_{-\nu-\sigma,m}$ is uniquely determined by adding the $\nu + \sigma$ fictitious data $(x_{-\nu-\sigma}, y_{-\nu-\sigma}), \dots, (x_{-1}, y_{-1})$ to the left of (x_0, y_0). Obviously these are all the supplementary data that we may need in order to determine also the remaining polynomials $p_{-\nu+j-\sigma,m}$, $j = 1, \dots, n$, associated with the other non-vanishing blending functions in $\tilde{I}_{\mathrm{L}}$.

(b) In the interval $\tilde{I}_{\mathrm{R}}$, the rightmost non-vanishing blending function $B_{N-1,n+1}$ is associated in (3.14b) with the polynomial $p_{N-1-\sigma,m}$ interpolating the data $(x_{N-1-\sigma}, y_{N-1-\sigma}), \dots,$ $(x_{N-1+m-\sigma}, y_{N-1+m-\sigma})$. As a consequence, the set of interpolation data needs to be extended by appending the $m-1-\sigma$ fictitious data $(x_{N+1}, y_{N+1}), \dots, (x_{N-1+m-\sigma}, y_{N-1+m-\sigma})$.

□

Corollary 3.4. *The total number of fictitious data that need to be added is* $m+\nu-1$*, namely* $w-2$ *in the even configuration and* $w-1$ *in the odd configuration.*

Corollary 3.5. *In the centered case,* $\sigma = \left\lfloor \frac{m-n-1}{2} \right\rfloor$ *(see* (3.4)*), and we need to append the same number* $\left\lceil \frac{m+n-1}{2} \right\rceil$ *of fictitious data to both ends.*

In the following we discuss a strategy to derive the needed fictitious values, which is based on imposing that F has prescribed derivatives at the endpoints τ_{L} and τ_{R} of its interval of definition. We recall that $\tau_{\mathrm{L}} = \tau_0$ in the even configuration and $\tau_{\mathrm{L}} = \tau_{-1}$ in the odd configuration, and $\tau_{\mathrm{R}} = \tau_N$ in both cases. The following preliminary result guarantees the feasibility of this strategy.

Proposition 3.6. *The degree of the interpolating spline* F *is greater than or equal to the number of fictitious data added to each end.*

Proof. From (3.8) and the subsequent formulas for $K_{i,j}$, the degree of the interpolating spline F is

$$g = \begin{cases} \max(m, \bar{d}+2), & \text{for B1-splines in even configuration,} \\ \max(m, \bar{d}+1), & \text{otherwise,} \end{cases}$$

with $\bar{d} := \max_{j=1,\ldots,rn} \deg(q_{i,j})$, where $q_{i,j}$ is the numerator of the jth piece of the ith transition function. Moreover, by Proposition 3.3, the maximum number of fictitious data that need to be added to any end is $\max(\nu+\sigma, m-1-\sigma)$.
We denote $\sigma_{\mathrm{c}} := \left\lfloor \frac{m-n-1}{2} \right\rfloor$ and distinguish three cases according to the value of the integer shift $\sigma \in \Sigma = \{-1, \ldots, m-\nu\}$.

- $-1 \leqslant \sigma \leqslant \sigma_{\mathrm{c}} - 1$. Then, $\max(\nu+\sigma, m-1-\sigma) = m-1-\sigma$ and $m-1-\sigma \leqslant m \leqslant g$.

- $\sigma = \sigma_{\mathrm{c}}$. Then, $\max(\nu+\sigma, m-1-\sigma) = \left\lceil \frac{m+n-1}{2} \right\rceil$. Recalling the necessary condition for interpolation (Property 1 in Section 3.2), that is $m+1 \geqslant n$ in case of even configuration and $m+1 \geqslant n+1$ in case of odd configuration, we obtain the bounds

$$\left\lceil \frac{m+n-1}{2} \right\rceil = \begin{cases} \dfrac{m+n-1}{2} \leqslant \dfrac{m+m+1-1}{2} = m, & \text{for even configuration,} \\ \dfrac{m+n}{2} \leqslant \dfrac{m+m}{2} = m, & \text{for odd configuration.} \end{cases}$$

  Thus, $\left\lceil \frac{m+n-1}{2} \right\rceil \leqslant m \leqslant g$ for both types of configuration.

- $\sigma_{\mathrm{c}} + 1 \leqslant \sigma \leqslant m - \nu$. Then, $\max(\nu+\sigma, m-1-\sigma) = \nu+\sigma$ and $\nu+\sigma \leqslant m \leqslant g$.

□

Based on the above proposition, denoted by $D_{\mathrm{L},s}, D_{\mathrm{R},s}$ the given derivatives of order s at $\tau_{\mathrm{L}}, \tau_{\mathrm{R}}$ respectively, we determine the $\nu+\sigma$ fictitious values y_j, $j<0$, and the $m-1-\sigma$ fictitious values y_j, $j>N$, respectively as the solutions of the linear systems

$$F^{(s)}(\tau_{\mathrm{L}}) = D_{\mathrm{L},s}, \qquad s = 1, \ldots, \nu+\sigma, \tag{3.15a}$$

$$F^{(s)}(\tau_{\mathrm{R}}) = D_{\mathrm{R},s}, \qquad s = 1, \ldots, m-1-\sigma, \tag{3.15b}$$

which can be written in the form $A\boldsymbol{z} = \boldsymbol{b}$ with the following general expressions.

Let us recall that $\nu = n$ in the case of even configuration and $\nu = n+1$ in the case of odd configuration.

For simplicity, we denote the unknowns $y_{-\nu-\sigma+s-1}$ as z_s, $s=1,\dots,\nu+\sigma$. In this way, at τ_{L}, the system (3.15a) can be written in the general form $A\boldsymbol{z}=\boldsymbol{b}$, where

$$\begin{aligned}
A_{s,h} &= \ell^{(s)}_{-\nu-\sigma+h-1,m}(\tau_{\mathrm{L}}) + \sum_{k=1}^{n} C_{-\nu-\sigma+h-1,-\nu-\sigma+k}\, Q^{(s)}_{-\nu+k}(\tau_{\mathrm{L}}), \qquad s,h=1,\dots,\nu+\sigma,\\
b_s &= D_{\mathrm{L},s} - \sum_{k=0}^{-\nu-\sigma+m} y_k \ell^{(s)}_{k,m}(\tau_{\mathrm{L}}) - \sum_{k=1}^{n} T_{-\nu-\sigma+k}\, Q^{(s)}_{-\nu+k}(\tau_{\mathrm{L}}),
\end{aligned}$$

with

$$\begin{aligned}
Q_i(x) &= K_i(x)\, q_{i,-ri+1}(x), \qquad\qquad i=-\nu+1,\dots,1,\\
C_{i,j} &= \begin{cases} \dfrac{x_{j+m}-x_{j-1}}{\prod\limits_{\substack{k=j-1\\k\neq i}}^{j+m}(x_i-x_k)}, & \text{if } i \geqslant j,\\[2ex] -\dfrac{1}{\prod\limits_{k=j}^{j+m-1}(x_i-x_k)}, & \text{if } i=j-1,\\[2ex] 0, & \text{if } i\leqslant j-2, \end{cases}\\
T_j &= \frac{y_{j+m}}{\prod\limits_{k=j}^{j+m-1}(x_{j+m}-x_k)} + (x_{j+m}-x_{j-1}) \sum_{k=0}^{j+m-1} \frac{y_k}{\prod\limits_{\substack{\ell=j-1\\ \ell\neq k}}^{j+m}(x_k-x_\ell)},
\end{aligned}$$

and $\ell_{k,m}(x)$ the Lagrange basis polynomial of degree m relative to $x_{-\nu-\sigma},\dots,x_{-\nu-\sigma+m}$, namely

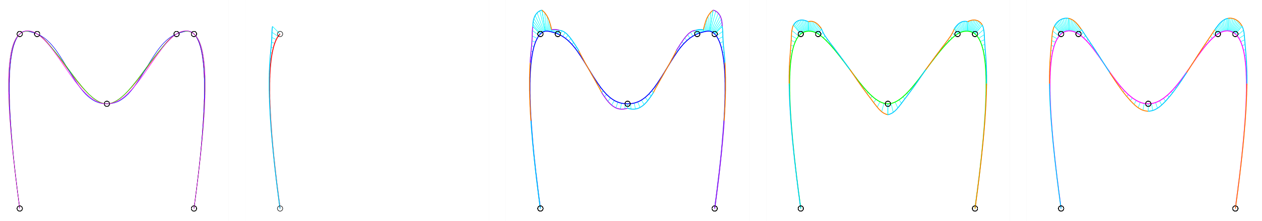
$$\ell_{k,m}(x) = \prod_{\substack{j=-\nu-\sigma\\ j\neq k}}^{-\nu-\sigma+m} \frac{x-x_j}{x_k-x_j}.$$

Analogously, denoting y_{N+s} as $\bar{z}_s$, $s=1,\dots,m-1-\sigma$, the linear system (3.15b) at τ_{R} can be written as $\bar{A}\bar{\boldsymbol{z}}=\bar{\boldsymbol{b}}$, where

$$\begin{aligned}
\bar{A}_{s,h} &= \begin{cases} \bar{\ell}^{(s)}_{N+h,m}(\tau_{\mathrm{R}}) + \sum\limits_{k=1}^{n} \bar{C}_{N+h,N-1-n-\sigma+k}\, \bar{Q}^{(s)}_{N-1-n+k}(\tau_{\mathrm{R}}), & \text{if } h\leqslant m-1-\sigma-n,\\ \sum\limits_{k=1}^{n} \bar{C}_{N+h,N-1-n-\sigma+k}\, \bar{Q}^{(s)}_{N-1-n+k}(\tau_{\mathrm{R}}), & \text{if } h> m-1-\sigma-n, \end{cases}\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad s,h=1,\dots,m-1-\sigma,\\
\bar{b}_s &= D_{\mathrm{R},s} - \sum_{k=N-1-n-\sigma}^{\min(N,N-1-n-\sigma+m)} y_k \bar{\ell}^{(s)}_{k,m}(\tau_{\mathrm{R}}) - \sum_{k=1}^{n} \bar{T}_{N-1-n-\sigma+k}\, \bar{Q}^{(s)}_{N-1-n+k}(\tau_{\mathrm{R}}),
\end{aligned}$$

with

$$\bar{Q}_i(x) = K_i(x)\, q_{i,r(N-i)}(x), \qquad\qquad i=N-n,\dots,N-1,$$



(a) **(b)** B1 $D^3C^1P^2S^4$ **(c)** B3 $D^3C^2P^2S^4$ **(d)** B2 $D^3C^2P^3S^6$ **(e)** B2 $D^4C^3P^4S^8$

Figure 3.8: Some parametric Br-splines interpolating open sets of unevenly spaced planar data: (a) all interpolants superimposed, (b)–(e) interpolant and related curvature comb.

$$\bar{C}_{i,j} = \begin{cases} \dfrac{x_{j+m} - x_{j-1}}{\displaystyle\prod_{\substack{k=j-1\\k\neq i}}^{j+m} (x_i - x_k)}, & \text{if } i \leqslant j+m-1, \\ \dfrac{1}{\displaystyle\prod_{k=j}^{j+m-1} (x_i - x_k)}, & \text{if } i = j+m, \\ 0, & \text{if } i \geqslant j+m+1, \end{cases}$$

$$\bar{T}_j = -\frac{y_{j-1}}{\displaystyle\prod_{k=j}^{j+m-1} (x_{j-1} - x_k)} + (x_{j+m} - x_{j-1}) \sum_{k=j}^{N} \frac{y_k}{\displaystyle\prod_{\substack{\ell=j-1\\ \ell\neq k}}^{j+m} (x_k - x_\ell)},$$

and $\bar{\ell}_{k,m}(x)$ the Lagrange basis polynomial of degree m relative to $x_{N-1-n-\sigma}, \dots, x_{N-1-n-\sigma+m}$, namely

$$\bar{\ell}_{k,m}(x) = \prod_{\substack{j=N-1-n-\sigma\\ j\neq k}}^{N-1-n-\sigma+m} \frac{x - x_j}{x_k - x_j}.$$

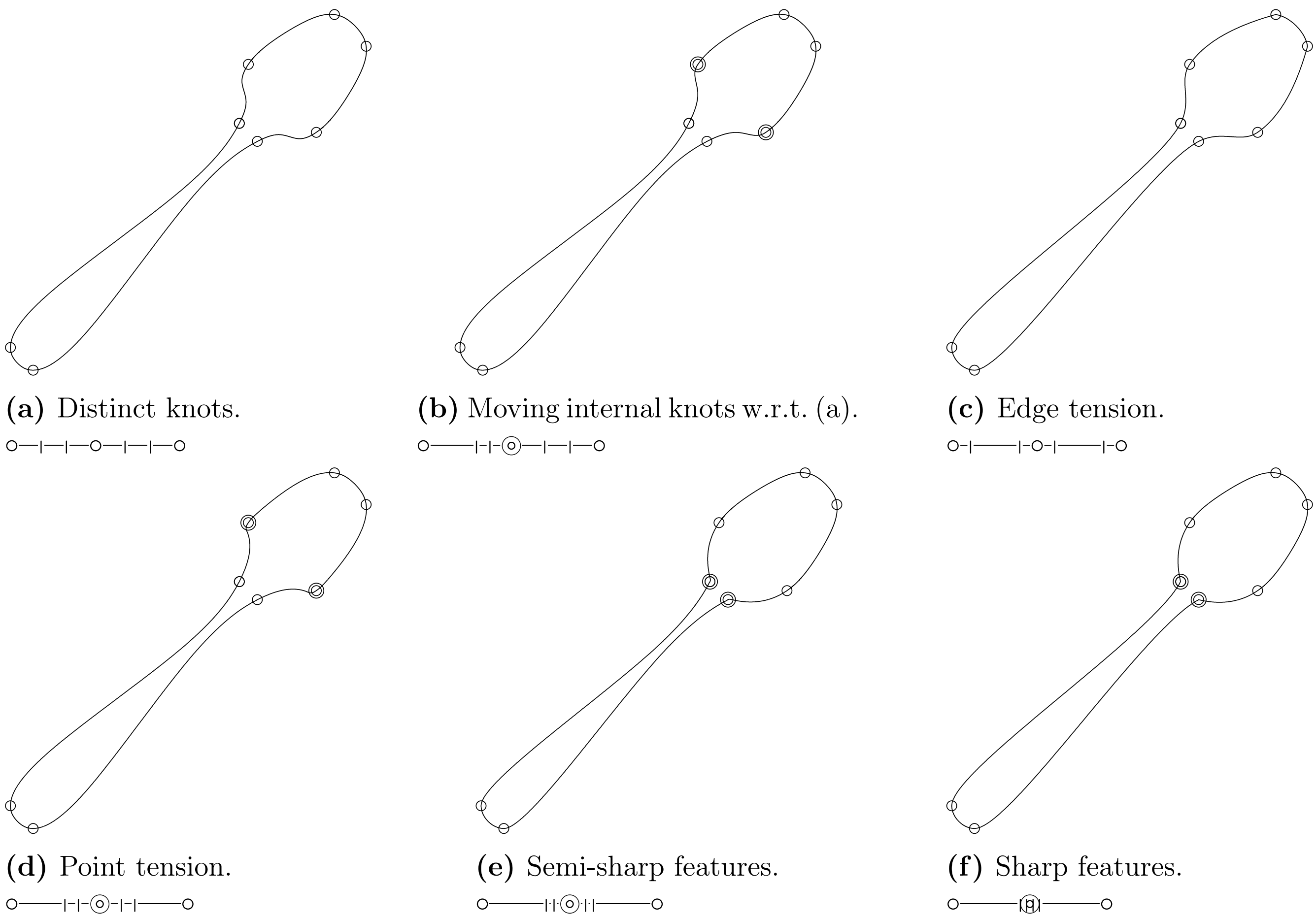
We conclude this section by showing in Figure 3.8 some examples of open spline interpolants.

3.6 The role of internal knots in Br-spline interpolation

The illustrated construction of Br-splines provides a polynomial interpolant of the form (3.8), whose peculiarity is being defined on the rth refinement $\boldsymbol{t}$ of the initial sequence $\boldsymbol{\tau}$. The breakpoints $t_{ri} := \tau_i$ are constrained by the relation between $\boldsymbol{\tau}$ and $\boldsymbol{x}$ (see (3.2a) and (3.2b)) and in particular are required to be distinct. Conversely, the values in $\boldsymbol{t}\setminus\boldsymbol{\tau}$ (hereinafter referred to as *internal knots*) can be chosen arbitrarily, within each interval in $\boldsymbol{\tau}$. In this respect, two situations are worth considering and are investigated in this section: the former corresponds to the presence of highly unevenly spaced internal knots, and the latter to the case of internal knots coinciding one another or with the endpoints of the related interval in $\boldsymbol{\tau}$.

3.6.1 Internal knots as shape parameters

In the former setting, it is interesting to study how the location of internal knots influences the shape of the spline. We illustrate these effects on parametric interpolants (see Figure 3.9),



(a) Distinct knots.

(b) Moving internal knots w.r.t. (a).

(c) Edge tension.

(d) Point tension.

(e) Semi-sharp features.

(f) Sharp features.

Figure 3.9: Internal knots as shape parameters for the class B3 $D^3C^2P^2S^4$. The captions show the configurations of knots related to the various kinds of shape effects.

so that they can be more evident. The considered interpolation data are highly non-uniform (and thus difficult to interpolate) and the nodes $\boldsymbol{x} \equiv \boldsymbol{\tau}$ are computed through the centripetal parameterization (thus the set $\boldsymbol{x}$ is the same for all the following examples).

In general, when we move an internal knot $t_{r\ell+h}$, $h = 1, \ldots, r-1$, towards one of the endpoints of the corresponding macro-interval $I_\ell := [\tau_\ell, \tau_{\ell+1}) = [t_{r\ell}, t_{r(\ell+1)})$, the spline becomes tighter to the data polygon in the region surrounding the related interpolation point. An accumulation of internal knots around the endpoint of a macro-interval boosts the effect, as illustrated in Figure 3.9(b). These observations suggest that internal knots can be regarded as shape parameters. Their most effective usage consists in the generation of *edge* or *point tension effects*. To generate tension along an edge (Figure 3.9(c)) it is sufficient to place half of the internal knots close to one endpoint and the other half close to the other endpoint of the macro-interval that parameterizes the edge. Tension at an interpolation point (Figure 3.9(d)) is achieved by moving the internal knots located to the left and the right of the corresponding node towards the node itself. In the limit this generates *semi-sharp* and *sharp features* (Figures 3.9(e) and 3.9(f) respectively). In particular, semi-sharp features are of great interest in computer-aided design since they represent the "effective" shape of corners of real objects, nevertheless they are hardly achievable by most spline- and non-spline-based methods (see, e.g., [KSD13]).

For all the examples in Figure 3.9, it is worth pointing out that the location of internal knots does not influence the overall quality of the interpolants, which is good due to the suitable choice of the nodes through the centripetal parameterization.

3.6.2 Multiple knots and continuity reduction

Until now we have assumed for simplicity that the values in $\boldsymbol{t} \setminus \boldsymbol{\tau}$ form a strictly increasing sequence, however it turns out that this restriction is not necessary at all. In fact, the requirement that the nodes $\boldsymbol{x}$ be distinct implies, for any r, ℓ, that $\tau_\ell = t_{r\ell} < t_{r(\ell+1)} = \tau_{\ell+1}$, whereas the internal knots $t_{r\ell+h}$, $h = 1, \dots, r-1$, can be set to any location inside the closed interval $[\tau_\ell, \tau_{\ell+1}]$. So, once the expression of F in (3.8) has been derived (assuming that all points in $\boldsymbol{t}$ are distinct), we can move one or more of the internal knots to coincident locations and, accordingly, one or more polynomial pieces of F disappear. The following discussion establishes which is the continuity of an interpolant of the form (3.8) in correspondence of multiple knots.

We start by supposing that the knots $\boldsymbol{t}$ are distinct. In this setting, any interpolant (3.8) with design parameters $D^g C^{\bar{k}} P^m S^w$ is a piecewise polynomial, with breaks at points $\boldsymbol{t}$, of degree g and continuity $\bar{k}$. As such, it can be represented in the B-spline basis of order $g+1$ over a suitable knot partition (see, e.g., [dB02])

$$\bar{\boldsymbol{t}} := \{\dots, \underbrace{t_{i-1}, \dots, t_{i-1}}_{g-\bar{k}\text{ terms}}, \underbrace{t_i, \dots, t_i}_{g-\bar{k}\text{ terms}}, \underbrace{t_{i+1}, \dots, t_{i+1}}_{g-\bar{k}\text{ terms}}, \dots\},$$

where each knot has the same multiplicity $g - \bar{k}$. Figure 3.10(a) illustrates the fundamental function (converted to B-spline form) corresponding to a class of local interpolating splines of the form (3.8) and only the B-spline basis functions associated with non-zero coefficients in the spline representation. Notice that the compact support of the fundamental function implies that any B-spline basis function whose support does not entirely lie inside the support of the fundamental function has related zero coefficient.

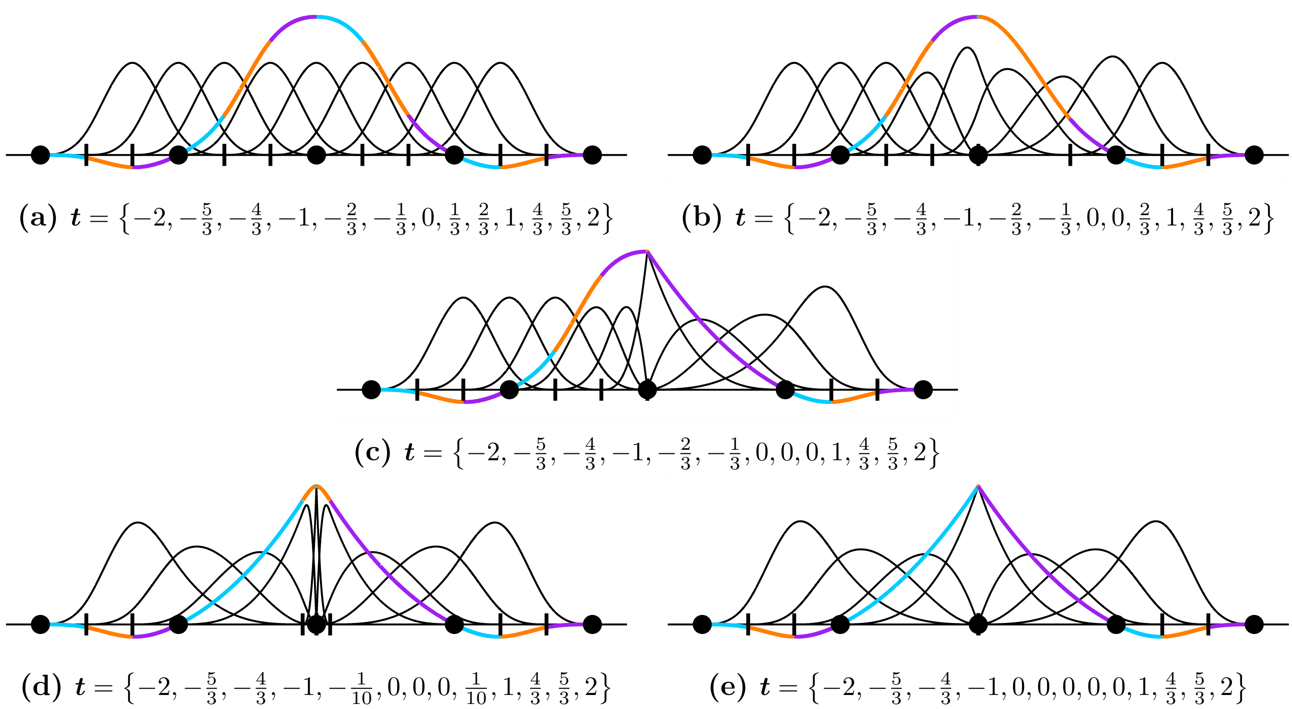
We are now in the position to make some knots in $\boldsymbol{t}$ coalesce and estimate the continuity of F by reasoning on the behavior of the B-spline basis functions on $\bar{\boldsymbol{t}}$. When μ knots in $\boldsymbol{t}$ are set to coincident locations, the corresponding blocks of knots in $\bar{\boldsymbol{t}}$ must be set accordingly. Therefore, the corresponding B-spline basis functions have continuity equal to $g - \mu(g - \bar{k})$. Now, recalling that the interpolants (3.8) are always at least C^0-continuous by construction, we can conclude that the least continuity of F is

$$\max\left(C^0, C^{\bar{k}+1-\mu(g-\bar{k})}\right). \tag{3.16}$$

We remark that for a Br-spline the maximum multiplicity at a knot of $\boldsymbol{t}$ is $2r-1$. This bound derives from the condition that the breakpoints need to be distinct and is obtained when we move all the internal knots from the left and the right to the location of a breakpoint. Figures 3.10(b) and 3.10(c) illustrate the reduction of continuity for a fundamental function of the class B3 $D^3C^2P^2S^4$, respectively when one and two internal knots per macro-interval coincide with one of its endpoints.

Usually, (3.16) is the exact continuity of the interpolant F, but, quite obviously, in some cases the continuity order may also be higher than the least.

This is the case for all F of the form (3.8) whose related blending functions have support $n+1 = 2$. We illustrate why by means of an example. Let us recall the general relation $B_{i,n+1} = f_i - f_{i+1}$ and that the continuity of F depends on the continuity of the blending functions according to Property 3 in Section 3.2. Figure 3.11 shows the transition functions and the related blending functions of the class B2 $D^4C^2P^4S^6$ in the cases of (a) distinct knots, (b) $t_{2i} = t_{2i+1}$, and (c) $t_{2i-1} = t_{2i} = t_{2i+1}$. In the case (b), the continuity of f_i (and hence of $B_{i,2}$ and $B_{i-1,2}$) decreases to C^0 at $t_{2i} = t_{2i+1}$ (reasoning on the B-spline representation of F, we have a knot of multiplicity 4 in $\bar{\boldsymbol{t}}$). In the case (c), also f_{i-1} becomes C^0 at $t_{2i-1} = t_{2i} = t_{2i+1}$: this affects the continuity of $B_{i-2,2}$, but, overall, all the non-vanishing blending functions remain


(a) $\boldsymbol{t} = \{-2, -\frac{5}{3}, -\frac{4}{3}, -1, -\frac{2}{3}, -\frac{1}{3}, 0, \frac{1}{3}, \frac{2}{3}, 1, \frac{4}{3}, \frac{5}{3}, 2\}$

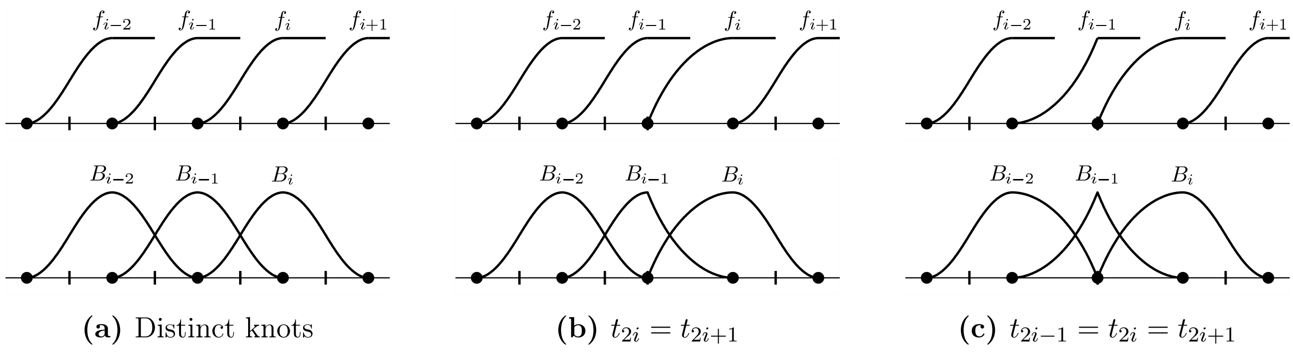
(b) $\boldsymbol{t} = \{-2, -\frac{5}{3}, -\frac{4}{3}, -1, -\frac{2}{3}, -\frac{1}{3}, 0, 0, \frac{2}{3}, 1, \frac{4}{3}, \frac{5}{3}, 2\}$

(c) $\boldsymbol{t} = \{-2, -\frac{5}{3}, -\frac{4}{3}, -1, -\frac{2}{3}, -\frac{1}{3}, 0, 0, 0, 1, \frac{4}{3}, \frac{5}{3}, 2\}$

(d) $\boldsymbol{t} = \{-2, -\frac{5}{3}, -\frac{4}{3}, -1, -\frac{1}{10}, 0, 0, 0, \frac{1}{10}, 1, \frac{4}{3}, \frac{5}{3}, 2\}$

(e) $\boldsymbol{t} = \{-2, -\frac{5}{3}, -\frac{4}{3}, -1, 0, 0, 0, 0, 0, 1, \frac{4}{3}, \frac{5}{3}, 2\}$

Figure 3.10: Fundamental functions of class B3 $D^3C^2P^2S^4$ with different multiplicities of knots and having continuity (a) C^2, (b) C^1, (c) C^0, (d) C^1, (e) C^0. The scale of the axes is 1:1.



(a) Distinct knots

(b) $t_{2i} = t_{2i+1}$

(c) $t_{2i-1} = t_{2i} = t_{2i+1}$

Figure 3.11: Transition and blending functions for the class B2 $D^4C^2P^4S^6$.

C^0 at t_{2i} and thus the continuity of F cannot decrease when passing from double to triple knots in $\boldsymbol{t}$. Clearly, the example illustrates a behavior that is common to all classes such that $n+1=2$: if we define the left (respectively right) multiplicity of t_{ri} as the number μ_L (respectively μ_R) such that $t_{ri-\mu_\mathrm{L}+1} = t_{ri}$ (respectively $t_{ri} = t_{ri+\mu_\mathrm{R}-1}$), then the continuity of F at t_{ri} is given by (3.16) with $\mu = \max(\mu_\mathrm{L}, \mu_\mathrm{R})$. This peculiarity is a consequence of the constructive approach behind the derivation of the general expression (3.8) which characterizes the considered family of splines. Figures 3.10(d) and 3.10(e) show two fundamental functions of the class B3 $D^3C^2P^2S^4$ with $\mu_\mathrm{L} = \mu_\mathrm{R} = 2$ and $\mu_\mathrm{L} = \mu_\mathrm{R} = 3$ that, based on the above discussion, are C^1 and C^0 respectively.

3.7 Final remarks

The original idea of Br-splines, that is to interpolate with splines whose knot partition strictly contains the interpolation nodes, came out in the late Eighties. In the seminal paper [DGM88] it is proved that, theoretically, the Br approach leads to local fundamental spline functions of degree h with the optimal order of continuity and degree of polynomial reproduction, respectively $h-1$ and h. Only for the order-$3, 4, 5$ members of the sub-class of B2-splines in [DGM88] an explicit expression was found in terms of the corresponding B-spline basis functions and used for the construction of compactly supported wavelets (see [CDV96]).

In about the same years, practical techniques to design B2- and B3-spline interpolants with applications in computer-aided geometric design were given in [Woo87, Chu90, KFK94], which apply to uniform knot partitions only. The resulting splines, though, showed a very poor shape quality and noticeable interpolation artifacts. Since this erratic behavior was interpreted as an intrinsic feature of the Br construction itself, it disqualified the approach, which for long time was considered of minor interest and mostly forgotten.

Nowadays, it has become common understanding that most of interpolation artifacts do arise due to a wrong choice of parameterization and do not inherently depend on a specific interpolation method. To fix the problem, a proper parameterization is usually computed by some automatic approach (i.e., centripetal or chordal) and results in a non-uniform distribution of parameter values. Conversely, the uniform parameterization is known to be unsuitable, unless the data are evenly spaced. In our view, this motivates the renewed interest in investigating the Br approach, especially in the non-uniform setting.

Besides this, non-uniform locally supported splines of B1 and B2 type have recently shown their importance in connection with the study of subdivision schemes that arise by sampling fundamental spline bases [BCR11a, BCR11b, KP13], because the continuity and polynomial reproduction properties of the schemes are related to the corresponding properties of the bases. In the bivariate setting, these subdivision schemes can be generalized to define non-uniform non-tensor-product interpolation methods [BCR13b], which have already proven to be effective in some applications [BFL$^+$10] and have significant potential to gain even more interest, due to the limited computational cost and the high quality of interpolation.

Overall, the main contribution of the work presented in this chapter is to provide a constructive method for the design of Br-spline interpolants over arbitrary non-uniform knot partitions. In this respect, we are not aware of any alternative approach (if we except the few and individual classes of splines considered in the aforementioned work [CDV96]). Moreover, the proposed method is comprehensive, because it comprises a large family of interpolants having among its members all the existing uniform and non-uniform B1- and Br-splines. Other two points of novelty of this approach are the possibility of handling in full generality multiple knots, which were not considered in previous Br constructions, and open sets of data.
As concerns the potential for future research, we believe that the generalized blending-based approach pursued in this chapter could be successfully exploited to tackle other relevant problems in interpolation (like the design of Hermite, shape-preserving, or variable-degree interpolants) and effectively applied to the multivariate setting as well.

In the next chapter, the families of non-uniform, local, univariate interpolating splines introduced in [BCR13a, ABC14a] (and reviewed in this chapter) will serve as one building block in the construction of analytical interpolating surfaces of high quality and regularity starting from input meshes of arbitrary topology.

Moreover, the idea of using transition functions to define blending functions with desired continuity properties will be exploited also in Chapter 5 to propose a general approach for the construction of B-spline bases for generalized splines spaces, whose elements are piecewise functions with pieces belonging to suitable, possibly non-polynomial, function spaces.

Chapter 4

High-quality local interpolation of arbitrary-topology meshes and curve networks by composite parametric surfaces

In computer-aided geometric design, generating a surface that interpolates a given arbitrary quadrilateral mesh or curve network is an important and challenging task. The basic requirement is to satisfy both criteria of analytical nature (regularity of the surface) and aesthetic concepts (shape quality). With regard to the aesthetic quality, it is well known that an interpolatory surface may show an excessively oscillating behavior in the intermediate areas in between the locations of the constraints. In the context of univariate interpolation, it is now commonly accepted that the use of a non-uniform parameterization can solve, or at least alleviate, the problem. Recently, a non-uniform non-tensor-product parameterization has been also successfully exploited to generate interpolants of regular 3D meshes by means of subdivision schemes [BCR13b]. In this context, the basic principle is to design a regular network of high-quality curves interpolating the mesh polylines, where each curve is parameterized independently of the others, and to construct a subdivision surface that interpolates the curves.
The work presented in this chapter originated from the same idea, but in order to design analytical surfaces of high quality and regularity starting from input meshes of arbitrary topology. In particular, these surfaces are built upon a family of non-uniform, local, univariate interpolating splines [BCR13a] by exploiting an innovative parameterization technique which allows the parameters to smoothly vary between one mesh face and another.
As an additional benefit, we will show that the proposed approach can be easily generalized in order to interpolate a given arbitrary network of curves.

The construction detailed in this chapter is also the topic of a recently submitted paper [ABC14b].

4.1 Introduction

A central topic in computer-aided geometric design is the creation of free-form objects of arbitrary topology by interpolating an initial control polyhedron (or mesh). While triangular meshes prevail in the context of reconstructing three-dimensional models from acquired data, for example by 3D scanning, when modeling free-form objects the control mesh is usually composed of quadrilateral faces and, if the model has to represent a complex shape, its vertices

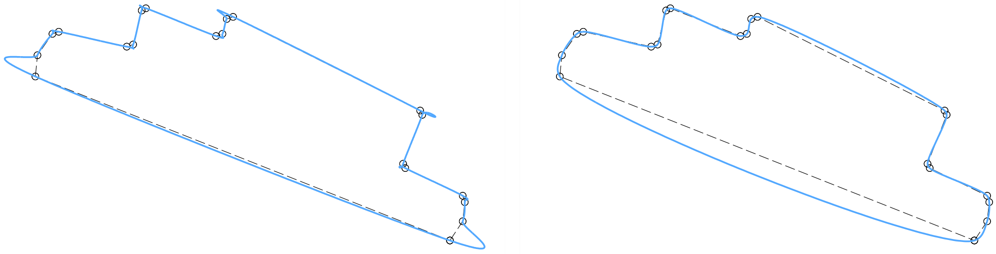


Figure 4.1: A cubic spline interpolant of very unevenly-spaced data. The uniform parameterization (left) fails compared to the non-uniform (here centripetal) one (right).

may have arbitrary connectivity. A second scenario of interest for applications is one where the interpolation data consist of a network of curves in which, similarly as before, an arbitrary number of curves can meet at each vertex. Although this chapter is focused on the first context, namely the interpolation of meshes, we will devise an approach that allows us to address the second situation basically at the same time, by means of a suitable extension. In both cases, the basic requirement is to generate models that satisfy both criteria of analytical nature (regularity of the surface) and aesthetic concepts (shape quality). With regard to the aesthetic quality, it is well known that when a surface has to satisfy a complex set of interpolation constraints it may show an oscillating behavior in the intermediate areas in between the locations of the constraints, quite often giving rise to noticeable interpolation artifacts.

When the data to be interpolated are of discrete nature, the usual approaches rely on patching techniques, such as Coons-Gregory patches [Far02], or subdivision schemes [WW01].
The first family of methods generates a composite surface of smoothness class G^1 or G^2, where each face of the mesh becomes a polynomial patch (usually of bidegree three or five). The approach relies on blending suitable values and derivatives at the corners [Her96], so that the construction is extremely local. Theoretically, the initially assigned derivative values could be used to optimize the shape of the surface. In practice, however, since these values can be estimated only heuristically, they can be hardly exploited to ensure a good aesthetic quality of the interpolant. Indeed, despite the wide literature on how to prescribe initial data for shape-preserving univariate interpolation, the surface setting is rarely addressed. Moreover, in the latter case no known estimate for the derivatives guarantees by itself a good quality of the resulting surfaces.
On the other hand, subdivision schemes recursively refine the control polyhedron by means of an iterative process that in the limit generates a smooth surface. However, if we confine ourselves to considering interpolatory schemes, only a version of the famous Catmull-Clark algorithm is capable of generating exactly evaluable limit surfaces [HKD93, Sta98]. The main drawback of this method is that it requires the solution of a global linear system to determine the control points of the surface and a post-processing optimization step to fair its shape. Some variations of the Catmull-Clark interpolatory scheme [MMN07, LBW05, Lin10] propose to improve computational cost and surface quality by exploiting the so-called *progressive iterative approximation* or *progressive interpolation* technique, which consists in constructing the interpolating surface by sequential approximation of its control polyhedron. These techniques are still inherently global because, as soon as a vertex changes its location, the entire control polyhedron must be updated in order to keep the interpolation property.

In the context of interpolation of 2D points, a recent trend is the study of approaches based on a non-uniform parameterization [DFH09, BCR11b, BCR11a, BCR13a]. The interest is motivated by the observation that, while the uniform parameterization may give rise to unacceptable artifacts, these undesirable effects may disappear completely by adopting a suitable parameterization (such as the centripetal or the chordal ones) [YSK11, KY06, Lee89, Flo08]. As a demonstration, see Figure 4.1.

With regard to the construction of free-form objects from arbitrary 3D meshes, the non-uniform parameterization principle has been a subject of interest especially in the area of approximation [CADS09, KSD13, MFR$^+$10, MRF06, SZSS98, Cas10, Cas11], with a view to establishing high-quality models compatible with the non-uniform B-spline standard. However, even in the context of interpolation, its potential usefulness is easily understood. In fact, it is clear that a high-quality surface must have high-quality section curves. Therefore, intuitively, we can think of generating a good-quality surface by interpolating a set of good-quality section curves. The benefits of this approach are outlined in [BCR13b]. That work considers a regular grid of points (i.e., with four edges incident at each vertex), where each polygonal section is parameterized independently of the others in a non-uniform way. From this structure, a family of nested curve networks is constructed by an iterative refinement process, in which each successively generated network is denser and contains the previous one. In the limit, the process converges to a C^1 surface. The curves of the initial network are immersed in this surface and retain their original parameterization. The approach generates surfaces of higher quality compared to the classical non-uniform tensor-product splines or subdivision schemes. The downside is that, because the limit surface is generated by a recursive process, it cannot be described in analytic form or exactly evaluated at arbitrary points. Besides this, the approach [BCR13b] does not allow handling meshes with extraordinary vertices.

Motivated by these results, our objective is to provide a method for generating an interpolatory surface of high quality and sufficient regularity from a quadrilateral mesh whose vertices may have an arbitrary number of incident edges (for short, an *arbitrary mesh*). In general, we are not given a suitable network of curves to interpolate, where suitable means of good quality and satisfying G^1- or G^2-compatibility conditions at the intersections. For this reason, we propose a two-step approach that, given an arbitrary quadrilateral mesh:

1. generates a network of high-quality curves interpolating the mesh vertices;
2. builds a composite surface that interpolates the curves of the network with a 1-1 association between mesh faces and surface patches.

In addition, if a suitable network of parametric curves is given, a direct generalization of our approach allows us to generate a surface that interpolates the network, regardless of the nature of the curves (polynomial, trigonometric, etc.).

We can outline the two steps of our technique as follows.

First stage — During the construction of the curve network, each curve is independently parameterized by the centripetal or chordal parameterization, in such a way to optimize its quality. The curves are then represented by piecewise-polynomial functions and are defined in a completely local way exploiting a class of compactly supported fundamental functions [BCR13a], which turns out to be particularly convenient due to its property of a very limited support width. In this way, the portion of curve between two mesh vertices depends only on few adjacent points belonging to a polygonal section of the mesh (which does not necessarily need to be planar). In the neighborhood of valence-4 vertices, the network does not have to meet any particular requirement for compatibility. In the neighborhood of *extraordinary vertices* (i.e., vertices of valence other than 4), we construct the network in order to satisfy compatibility constraints of G^1 and G^2 type, according to the desired smoothness for the final surface.

Second stage — The curve network serves as a skeleton from which we build a smooth surface by suitably blending the curves. At this stage two innovative ingredients are introduced: the first is to use as blending functions a class of non-uniform, locally supported fundamental functions [BCR13a]; the second consists in a suitable strategy of local parameterization. In this way, the curves become the boundaries of the patches of a composite surface, where each patch, parameterized over the domain $[0,1]^2$, corresponds to a face of the mesh and depends on only one or two rings of vertices surrounding the face. Moreover, we introduce a novel parameterization strategy, called *augmented parameterization*, which does not require that opposite boundaries of a face have the same parameterization. As a result, the curves of the network are embedded in the surface, without being approximated or altered in any way, and retain their original parameterization and consequently their good shape quality. This is a major difference with respect to the classical approaches based on tensor-product or Coons-Gordon surfaces, where averaging the parameterization of parallel section curves is necessary.

As a historical note, it is interesting to remark the similarities between our augmented parameterization and a technique proposed by Geoff Hayes in 1974 [Hay74] in the context of fitting samples of a bivariate function through bicubic splines. Hayes observed that a better interpolation can be obtained if, instead of using the usual rectangular grid of knot intervals, we use what he called *curved knot-lines*, where the knot intervals are not forced to form a grid, but their lengths can be selected in a more flexible way. Thus, Hayes' technique and ours share the common idea that the classical tensor-product parameterization, albeit non-uniform, is not flexible enough to generate surfaces of good quality.

The remainder of the chapter is organized as follows. In Section 4.2 we outline the family of local non-uniform spline interpolants which is at the basis of the curve network construction and of the local patching scheme. We also formulate these splines in such a way to better exploit their local dependence on the parameterization. In Section 4.3 we describe the first stage of our approach, which consists in the construction of a network of curves with suitable compatibility conditions at the vertices that will serve as a skeleton for the patching scheme. Starting from this structure, we describe in Section 4.4 how a local surface patch with augmented parameterization can be worked out from every quadrilateral face of the mesh that does not contain extraordinary vertices. In Section 4.5 we propose a construction which applies to faces with an arbitrary number of extraordinary vertices. We address in particular the definition of G^1 and G^2 surface patches that guarantee the interpolation of the curve network and provide a smooth connection with the surrounding surface patches, exploiting again the augmented parameterization. In Section 4.6 we briefly discuss the case in which the data consist in a network of curves instead of a mesh, thus the first stage of our approach is skipped. In this situation, a straightforward application of the proposed augmented parameterization gives rise to an augmented Gordon-Coons scheme with the property of locality. In Section 4.7 we provide some examples of surfaces obtained with the described method starting from regular meshes or from meshes containing extraordinary vertices, and we conclude with some final remarks in Section 4.8.

4.2 Locally supported, non-uniform, fundamental spline interpolants

In the remainder of this chapter we extensively exploit some families of non-uniform local interpolatory splines. Given a set of points $\{\boldsymbol{p}_j \in \mathbb{R}^n\}$, associated with a knot sequence $\{x_j \in \mathbb{R}\}$, we focus on local spline interpolants expressed by the formula

$$\boldsymbol{F}(x) = \sum_i \boldsymbol{p}_i \, \psi_i(x), \tag{4.1}$$

where the functions ψ_i, called *fundamental functions*, satisfy the relation $\psi_i(x_j) = \delta_{i,j}$. We assume that the knots $\{x_j\}$ are computed by a proper automatic data-dependent method, such as the chordal or centripetal parameterization [ANW67, Lee89, FS06]. In particular,

$$x_{i+1} = x_i + \|\boldsymbol{p}_{i+1} - \boldsymbol{p}_i\|_2^\alpha \,, \tag{4.2}$$

where $\alpha = \frac{1}{2}$ or $\alpha = 1$ respectively for the centripetal or chordal parameterization ($\|\cdot\|_2$ denotes the Euclidean norm).

If the support of each function ψ_i is finite, then the interpolant (4.1) is local, i.e., its value at a certain location depends on only a limited number of points among the $\boldsymbol{p}_j$'s. In particular, we say that the function $\psi_i(x)$ has support width equal to w if it is identically zero for any x outside a compact interval of the form $[x_j, x_{j+w}]$ for some j.

A construction for several families of polynomial fundamental functions, defined on non-uniform knot sequences, is presented in [BCR13a] and extended in Chapter 3. Each such family is characterized by its polynomial degree, order of continuity, approximation order (or, equivalently, maximum degree of polynomials that are reproduced), and support width. The constructive framework allows choosing, within some reasonable restrictions, the most convenient balance between these four features according to the needs of a specific application. For brevity, we indicate a class of splines having Degree g, Continuity order k, Polynomial reproduction degree m and Support width w by the shorthand $D^gC^kP^mS^w$. So, for example, if we fix the support width to 4, we can find the classes $D^3C^0P^3S^4$, $D^3C^1P^2S^4$, $D^4C^2P^1S^4$, $D^5C^2P^2S^4$, ..., and, similarly, many more classes exist having support width 6 (see Tables 1 and 2 in [BCR13a], and for the generalization to Br-splines of minimum degree see Table 3.2 in the previous chapter). Conversely, if we are interested in a certain continuity or order of approximation, the approach in [BCR13a] always allows us to construct a class of locally supported fundamental functions, of suitable support width, having the desired property.

We will focus on fundamental splines that have even support width w and that assume the value 1 at the central knot of the support. In fact, such condition identifies the most relevant classes of splines, namely those which preserve symmetries in the initial data (recall Property 5 in Chapter 3). This restriction is not really necessary in what we will see in the following sections, but we prefer slightly sacrificing generality in favor of a more streamlined notation. In this way, it holds

$$\psi_i(x) = 0, \qquad x \notin [x_{i-\frac{w}{2}}, x_{i+\frac{w}{2}}],$$

and if we set

$$d_i \coloneqq x_{i+1} - x_i, \tag{4.3}$$

we can observe that each fundamental function of support width w is a piecewise polynomial made of w pieces, one per each interval $[x_j, x_{j+1}]$, $j = i - \frac{w}{2}, \dots, i + \frac{w}{2} - 1$, and it depends on (at most) the w corresponding knot intervals d_j, $j = i-\frac{w}{2}, \dots, i+\frac{w}{2}-1$ (see Figure 4.2(a)). However, since in practice we evaluate each polynomial piece in (4.1) at a time, it is more convenient to reason on a single interval. Focusing on $[x_\ell, x_{\ell+1}]$, we observe that w different kinds of pieces of the fundamental functions ψ_j, $j = \ell - \frac{w}{2} + 1, \dots, \ell + \frac{w}{2}$, contribute to the evaluation of (4.1) (see Figure 4.2(b)) and, overall, the knot intervals involved are d_j, $j = \ell - \frac{w}{2} + 1, \dots, \ell + \frac{w}{2} - 1$. This means that not all the knot intervals which constitute the support of a single fundamental function are involved in the evaluation of (4.1) on an interval. Based on this consideration, to emphasize the local nature of the spline interpolant, it is convenient to reformulate the piece of the spline interpolant (4.1) in $[x_\ell, x_{\ell+1}]$ as

$$\boldsymbol{F}(x)\big|_{[x_\ell, x_{\ell+1}]} = \sum_{i=1}^{w} \bar{\boldsymbol{p}}_i \, \psi_{\ell-\frac{w}{2}+i}(x; \bar{\boldsymbol{d}}), \tag{4.4}$$

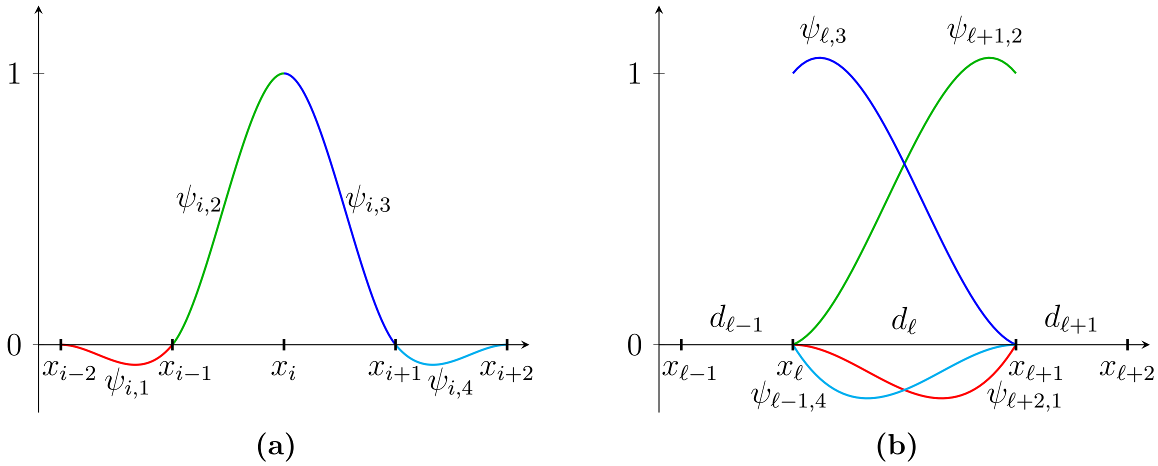


Figure 4.2: (a) Fundamental function of the class $D^3C^1P^2S^4$; (b) the four pieces of the fundamental functions of this class on a single interval $[x_\ell, x_{\ell+1}]$, and related parameters d_j in equation (4.5).

where

$$\bar{\boldsymbol{p}} := \left(\boldsymbol{p}_{\ell-\frac{w}{2}+1}, \ldots, \boldsymbol{p}_{\ell+\frac{w}{2}}\right),$$

and

$$\bar{\boldsymbol{d}} := \left(d_{\ell-\frac{w}{2}+1}, \ldots, d_{\ell+\frac{w}{2}-1}\right).$$

It is useful to point out that three classes of splines will be exploited as running examples in this chapter: besides the so-called Catmull-Rom splines [CR74] (hereinafter indicated by $D^3C^1P^2S^4$), we will consider other two "unnamed" classes, $D^5C^2P^2S^4$ and $D^4C^2P^3S^6$, developed in [BCR13a][1]. The motivation for our choice is that they represent a good tradeoff between computational cost (represented by support width and degree) and analytical properties (order of continuity and approximation order). To make our proposal easily reproducible, in Section 4.2.1 we provide the expressions related to the two classes with support width 4 (we do not report the expressions for the class with support width 6 because of their excessive length). Moreover, we remark that in the context of 3D points interpolation it may be interesting to consider also classes of splines featuring C^3 continuity. In fact, such splines can achieve torsion continuity and we expect that this will result in fairer space curves.

Usually, the data sequence $\{\boldsymbol{p}_j\}$ is meant to be periodic, but, for later use, it is useful to discuss as of now the case in which it is open. In this situation, the interpolant (4.1) can be defined in the boundary knot interval $[x_0, x_1]$ and on the subsequent ones (and, equivalently, in the last intervals) by extrapolating $\frac{w}{2} - 1$ points across the boundary. This is equivalent to prescribing the derivatives of $\boldsymbol{F}$ at x_0 up to order $\frac{w}{2} - 1$, as detailed more formally in Section 3.5. The resulting spline interpolates at x_0 the prescribed derivatives, to which we will thus refer as *Hermite end conditions*.

[1] According to the terminology introduced in Chapter 3 (precisely, in Definition 3.1), the spline interpolants associated with these classes are B1-splines.

4.2.1 Fundamental functions associated with the considered classes of spline interpolants

We provide here the expressions of the fundamental functions associated with the two classes having support width 4 and used as examples in this chapter.
We remark that in the following expressions $\psi_{i,j}$ denotes the jth piece of the fundamental function ψ_i.

4.2.1.1 Class $D^3C^1P^2S^4$

The general expression of the fundamental function of the class $D^3C^1P^2S^4$ (Catmull-Rom splines [CR74]) on its support $[-d_{i-2}-d_{i-1}, d_i+d_{i+1}]$, which is illustrated in Figure 4.2(a), is:

$$\psi_i(x)=\begin{cases} \frac{(d_{i-1}+x)(d_{i-2}+d_{i-1}+x)^2}{d_{i-2}d_{i-1}(d_{i-2}+d_{i-1})}, & -d_{i-2}-d_{i-1}\leqslant x<-d_{i-1},\\ -\frac{(d_{i-1}+x)\left(d_i\left(x^2-d_{i-1}(d_{i-2}+d_{i-1})\right)+x(d_{i-2}+d_{i-1})(d_{i-1}+x)\right)}{d_{i-1}^2(d_{i-2}+d_{i-1})d_i}, & -d_{i-1}\leqslant x<0,\\ \frac{(d_i-x)\left(x^2(-(d_{i-1}+d_i+d_{i+1}))+xd_i(d_i+d_{i+1})+d_{i-1}d_i(d_i+d_{i+1})\right)}{d_{i-1}d_i^2(d_i+d_{i+1})}, & 0\leqslant x<d_i,\\ \frac{(d_i-x)(d_i+d_{i+1}-x)^2}{d_id_{i+1}(d_i+d_{i+1})}, & d_i\leqslant x\leqslant d_i+d_{i+1}. \end{cases}$$

Additionally, the expressions of the four pieces of the fundamental functions of the same class on $[x_\ell, x_{\ell+1}]$ are:

$$\begin{aligned} \psi_{\ell-1,4}(x)&=-\frac{x\left(x-d_\ell\right)^2}{d_{\ell-1}d_\ell\left(d_{\ell-1}+d_\ell\right)},\\ \psi_{\ell,3}(x)&=\frac{1}{d_\ell^2}\left(x-d_\ell\right)\left(\frac{x^2}{d_\ell+d_{\ell+1}}+\frac{x\left(x-d_\ell\right)}{d_{\ell-1}}-d_\ell\right),\\ \psi_{\ell+1,2}(x)&=\frac{1}{d_\ell^2}x\left(\frac{d_\ell\left(d_{\ell-1}+2x\right)-x^2}{d_{\ell-1}+d_\ell}+\frac{x\left(d_\ell-x\right)}{d_{\ell+1}}\right),\\ \psi_{\ell+2,1}(x)&=\frac{x^2\left(x-d_\ell\right)}{d_\ell d_{\ell+1}\left(d_\ell+d_{\ell+1}\right)}, \end{aligned} \tag{4.5}$$

These pieces are shown in Figure 4.2(b).

4.2.1.2 Class $D^5C^2P^2S^4$

The general expression of the fundamental function of the class $D^5C^2P^2S^4$ on its support $[-d_{i-2}-d_{i-1}, d_i+d_{i+1}]$ is:

$$\psi_i(x)=\begin{cases} -\frac{(d_{i-1}+x)(d_{i-2}+d_{i-1}+x)^3(-d_{i-2}+2d_{i-1}+2x)}{d_{i-2}^3d_{i-1}(d_{i-2}+d_{i-1})}, & -d_{i-2}-d_{i-1}\leqslant x<-d_{i-1},\\ \frac{(d_{i-1}+x)\left((d_{i-2}+d_{i-1}+d_i)\left(3x^3d_{i-1}+2x^4\right)+d_{i-1}^3(d_{i-2}+d_{i-1})(d_i-x)\right)}{d_{i-1}^4(d_{i-2}+d_{i-1})d_i}, & -d_{i-1}\leqslant x<0,\\ \frac{(d_i-x)\left((2x^4-3x^3d_i)(d_{i-1}+d_i+d_{i+1})+d_i^3(d_i+d_{i+1})(x+d_{i-1})\right)}{d_{i-1}d_i^4(d_i+d_{i+1})}, & 0\leqslant x<d_i,\\ -\frac{(x-d_i)(-2d_i+d_{i+1}+2x)(d_i+d_{i+1}-x)^3}{d_id_{i+1}^3(d_i+d_{i+1})}, & d_i\leqslant x\leqslant d_i+d_{i+1}. \end{cases}$$

Additionally, the expressions of the four pieces of the fundamental functions of the same class on $[x_\ell, x_{\ell+1}]$ are:

$$\begin{aligned}
\psi_{\ell-1,4}(x) &= \frac{x\left(x-d_\ell\right)^3\left(d_\ell+2x\right)}{d_{\ell-1}d_\ell^3\left(d_{\ell-1}+d_\ell\right)},\\
\psi_{\ell,3}(x) &= \frac{\left(d_\ell-x\right)\left(d_{\ell-1}\left(-3x^3d_\ell+d_\ell^4+d_\ell^3d_{\ell+1}+2x^4\right)+x\left(d_\ell+d_{\ell+1}\right)\left(d_\ell+2x\right)\left(x-d_\ell\right)^2\right)}{d_{\ell-1}d_\ell^4\left(d_\ell+d_{\ell+1}\right)},\\
\psi_{\ell+1,2}(x) &= \frac{1}{d_\ell^4}x\left(\frac{x^2\left(2x-3d_\ell\right)\left(x-d_\ell\right)}{d_{\ell+1}}+\frac{-5x^3d_\ell+3x^2d_\ell^2+d_\ell^3\left(d_{\ell-1}+x\right)+2x^4}{d_{\ell-1}+d_\ell}\right),\\
\psi_{\ell+2,1}(x) &= -\frac{x^3\left(2x-3d_\ell\right)\left(x-d_\ell\right)}{d_\ell^3d_{\ell+1}\left(d_\ell+d_{\ell+1}\right)}.
\end{aligned}$$

4.3 From an arbitrary mesh to a curve network compatible for continuity

In this section we exploit the considered local spline interpolants to work out a network of curves, starting from an arbitrary input mesh. As we will see, the local nature of the interpolants makes the proposed procedure easy and effective. We recall that a network of curves is called G^k*-compatible* if, in a neighborhood of each node (i.e., any point where two or more curves of the network intersect), it can be embedded into a C^k surface.

Hereinafter we implicitly assume that the starting mesh is manifold (it has no boundary and each edge is shared by exactly two faces), while the case of non-closed meshes will be briefly addressed in Section 4.3.3.4. We also call an *extraordinary vertex* a vertex with valence other than 4. Moreover, we call *regular region* a portion of the mesh in which the vertices form a quadrilateral grid with all vertices having valence 4. To provide a more readable outline, we split the section into three parts corresponding to the three following stages: first we associate a non-uniform parameterization to the input mesh (Section 4.3.1), then we compute the curve segments corresponding to regular regions of the mesh (Section 4.3.2), and finally we compute the curve segments emanating from the extraordinary vertices (Section 4.3.3).

4.3.1 Non-uniform parameterization of the mesh

A preliminary step towards the construction of the curve network is the computation of a proper parameterization for its curves. To this aim, we associate with each edge e of the mesh a knot interval d_e. Denoted by $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ the two endpoints of e, then

$$d_e := \left\|\boldsymbol{p}_2 - \boldsymbol{p}_1\right\|_2^\alpha, \tag{4.6}$$

where $\alpha = \frac{1}{2}$ or $\alpha = 1$ respectively for the centripetal or chordal parameterization. Note that this definition is equivalent to (4.2)–(4.3).

4.3.2 Regular regions of the mesh

Since a regular region of the mesh has a grid-like structure, we can conveniently denote its vertices by $\boldsymbol{p}_{i,j}$. In such a region, the mesh can be seen as a collection of two families of *section polylines*, $\mathfrak{P}_s = \{\boldsymbol{p}_{i,j}\}_i$ and $\mathfrak{P}_t = \{\boldsymbol{p}_{i,j}\}_j$, associated with two independent directions s and t.

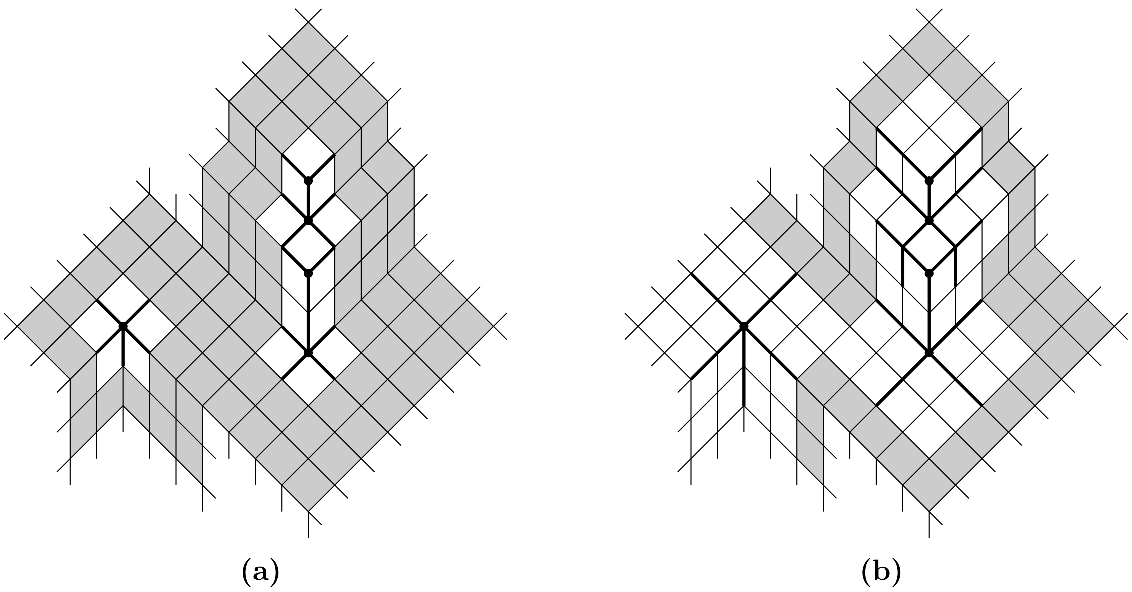


Figure 4.3: Mesh containing several extraordinary vertices in various configurations: isolated, close, on the same face, adjacent. The regular regions are highlighted in gray, for spline interpolants having support width (a) 4 and (b) 6. Mesh edges where the regular construction is not possible are thicker.

We construct the curve network in a piecewise manner, by associating with each edge $\overline{\boldsymbol{p}_{i,j}, \boldsymbol{p}_{i+1,j}}$ of the mesh (and likewise for edges $\overline{\boldsymbol{p}_{i,j}, \boldsymbol{p}_{i,j+1}}$ in the other direction) a curve segment of an interpolating spline $D^g C^k P^m S^w$ defined locally by equation (4.4). The local sequences of points $\bar{\boldsymbol{p}}$ and knot intervals $\bar{\boldsymbol{d}}$ involved in (4.4) are formed by the vertices and knot intervals of a suitable number (depending on the support width w of the class) of edges of the section polyline which contains $\overline{\boldsymbol{p}_{i,j}, \boldsymbol{p}_{i+1,j}}$, located on both sides of the edge. Note that, as a result, each section curve in either direction is parameterized independently of the others. Moreover, if the class of splines has continuity order k, then the resulting network can be embedded in a C^k surface. In fact, at each node, the derivatives up to order k are continuous in the two independent directions s, t.

4.3.3 Mesh regions with extraordinary vertices and G^1/G^2-compatibility conditions

Due to its local nature, the construction proposed in the previous section can be applied in all the areas of a mesh that are sufficiently far from extraordinary vertices. Thus in this section we will only discuss how to define the curve segments whose endpoints belong to the neighborhood of an extraordinary vertex. As will become clear in the following, the width of the neighborhood where such a construction is necessary depends on the support of the underlying fundamental functions. Therefore we proceed as follows: first we illustrate some notions common to all cases, then we detail the construction for our three running examples, namely the classes $D^3C^1P^2S^4$, $D^5C^2P^2S^4$, and $D^4C^2P^3S^6$.

Let $\boldsymbol{V}$ be a vertex of valence n. We denote by $e_i, i = 1, \ldots, n$, the edges emanating from $\boldsymbol{V}$ and by $\boldsymbol{p}_i$ the other endpoint of e_i (see Figure 4.4). Next, we associate with $\boldsymbol{V}$ and with each edge e_i a vector $\boldsymbol{\Delta}_{\boldsymbol{V},e_i}$ as follows.
Let

$$\boldsymbol{f}_i := \frac{\boldsymbol{p}_i - \boldsymbol{V}}{d_{e_i}},$$

where $d_{e_i} \coloneqq \|\boldsymbol{p}_i - \boldsymbol{V}\|_2^\alpha$ is the knot interval of edge e_i. Then we define

$$\boldsymbol{\Delta}_{\boldsymbol{V},e_i} = \alpha_i \boldsymbol{f}_i + (1 - \alpha_i)\bar{\boldsymbol{f}}, \tag{4.7}$$

where

$$\alpha_i = \frac{\bar{d}}{d_{e_i} + \bar{d}}, \qquad \bar{d} = -\sum_{\substack{j=1 \\ j \neq i}}^{n} \cos\left(\frac{2\pi(j-i)}{n}\right) d_{e_j}, \qquad \bar{\boldsymbol{f}} = \sum_{\substack{j=1 \\ j \neq i}}^{n} \cos\left(\frac{2\pi(j-i)}{n}\right) \boldsymbol{f}_j.$$

When $n = 4$, equation (4.7) reduces to the well-known Bessel estimate for computing an approximation to the first derivative of a parametric curve [Far02, Sections 9.5 and 9.8]. As a consequence, for general valence $n \neq 4$, we can say that $\boldsymbol{\Delta}_{\boldsymbol{V},e_i}$ represents a heuristic estimate of the first derivative at $\boldsymbol{V}$ of the curve segment between $\boldsymbol{V}$ and $\boldsymbol{p}_i$. In particular, we can observe that, when n is even and the points have rotational symmetry with respect to $\boldsymbol{V}$, then $\boldsymbol{\Delta}_{\boldsymbol{V},e_i}$ estimated by (4.7) corresponds to the Bessel formula applied to the three points $\boldsymbol{p}_{i+\frac{n}{2}}, \boldsymbol{V}, \boldsymbol{p}_i$, which are intuitively associated with a curve passing through $\boldsymbol{V}$.
Other reasonable estimates for $\boldsymbol{\Delta}_{\boldsymbol{V},e_i}$ could be used. For example, in case of even valence and a rather symmetric configuration of the vertices, we may regard half of the curves that contain the extraordinary vertex as the continuation of the other half. In case of odd valence, we may directly complete the information needed for the definition of the curve segment by considering one (or more) fictitious point located on the other side of the extraordinary vertex. This point can be simply the centroid of the opposite mesh face or computed as a suitable weighted average of neighboring vertices.

We exploit the computed vectors $\boldsymbol{\Delta}_{\boldsymbol{V},e_i}$, $i = 1, \ldots, n$, to generate the first and the second derivatives, respectively $\boldsymbol{\delta}_{\boldsymbol{V},e_i}$ and $\boldsymbol{\delta}^{(2)}_{\boldsymbol{V},e_i}$, of the curve segment associated with the edge e_i. These information is generated to guarantee that the network satisfies the G^1- or G^2-compatibility conditions at $\boldsymbol{V}$. To this aim, our strategy is to construct a polynomial $\boldsymbol{P}$ that interpolates $\boldsymbol{V}$ and approximates in a least-squares sense a suitable set of points $\boldsymbol{q}_j, j = 1, \ldots, 2n$ around $\boldsymbol{V}$ and set $\boldsymbol{\delta}_{\boldsymbol{V},e_i}$ and $\boldsymbol{\delta}^{(2)}_{\boldsymbol{V},e_i}$ as the derivatives of such polynomial along some proper directions. The approximation points $\boldsymbol{q}_j$ are chosen so that the polynomial will have a reasonable shape (w.r.t. the shape suggested by the input mesh) in a small neighborhood of $\boldsymbol{V}$. In particular, for each $i = 1, \ldots, n$, $\boldsymbol{q}_i$ and $\boldsymbol{q}_{n+i}$ are respectively the values at parameters $\frac{d_i}{4}$ and $\frac{d_i}{2}$ of the cubic polynomial $\boldsymbol{\rho}$ such that $\boldsymbol{\rho}(0) = \boldsymbol{V}$, $\boldsymbol{\rho}'(0) = \boldsymbol{\Delta}_{\boldsymbol{V},e_i}$, $\boldsymbol{\rho}(d_i) = \boldsymbol{p}_i$, $\boldsymbol{\rho}'(d_i) = \boldsymbol{\Delta}_{\boldsymbol{p}_i,e_i}$. Their expressions are given explicitly by

$$\begin{aligned}
\boldsymbol{q}_i &= \boldsymbol{\rho}\left(\frac{d_i}{4}\right) = \frac{1}{64}\left(54\boldsymbol{V} + 10\boldsymbol{p}_i + d_i\left(9\boldsymbol{\Delta}_{\boldsymbol{V},e_i} - 3\boldsymbol{\Delta}_{\boldsymbol{p}_i,e_i}\right)\right), \\
\boldsymbol{q}_{n+i} &= \boldsymbol{\rho}\left(\frac{d_i}{2}\right) = \frac{1}{8}\left(4\boldsymbol{V} + 4\boldsymbol{p}_i + d_i\left(\boldsymbol{\Delta}_{\boldsymbol{V},e_i} - \boldsymbol{\Delta}_{\boldsymbol{p}_i,e_i}\right)\right).
\end{aligned}$$

We use bivariate polynomials of total degree 3 or 2 respectively in the case $n \geqslant 5$ or $n = 3$. The coefficients of $\boldsymbol{P}$ are determined componentwise by minimizing the expression

$$\sum_{j=1}^{2n} \left(\boldsymbol{P}(s_j, t_j) - \boldsymbol{q}_j\right)^2.$$

In the above equation, the parametric coordinates (s_j, t_j) associated with the point $\boldsymbol{q}_j$ are given by

$$(s_j, t_j) = r_j\,(\cos\eta_i, \sin\eta_i),$$

where

$$r_j = \|\boldsymbol{q}_j - \boldsymbol{V}\|_2^{\alpha},$$

(the value of α is the same as chosen in (4.6)) and the angles $\eta_i, i = 1, \dots, n$, are obtained by mapping onto the (s,t)-plane the spatial configuration formed by the angles $\zeta_i := \widehat{\boldsymbol{\Delta}_{\boldsymbol{V},e_i}, \boldsymbol{\Delta}_{\boldsymbol{V},e_{i+1}}}$, $i = 1, \dots, n$, namely

$$\eta_1 = 0, \qquad \eta_i = \eta_{i-1} + \zeta_i \frac{2\pi}{\sum_{j=1}^{n} \zeta_j}, \quad i = 2, \dots, n.$$

Once computed the polynomial $\boldsymbol{P}$, we can set $\boldsymbol{\delta}_{\boldsymbol{V},e_i}$ and $\boldsymbol{\delta}^{(2)}_{\boldsymbol{V},e_i}$, $i = 1, \dots, n$, as the first and the second derivatives of $\boldsymbol{P}$ at $\boldsymbol{V}$ in the direction determined by η_i, i.e.

$$\begin{aligned}
\boldsymbol{\delta}_{\boldsymbol{V},e_i} &= \frac{\partial \boldsymbol{P}}{\partial s}(0,0)\cos\eta_i + \frac{\partial \boldsymbol{P}}{\partial t}(0,0)\sin\eta_i,\\
\boldsymbol{\delta}^{(2)}_{\boldsymbol{V},e_i} &= \frac{\partial^2 \boldsymbol{P}}{\partial s^2}(0,0)\cos^2\eta_i + 2\frac{\partial^2 \boldsymbol{P}}{\partial s \partial t}(0,0)\cos\eta_i \sin\eta_i + \frac{\partial^2 \boldsymbol{P}}{\partial t^2}(0,0)\sin^2\eta_i.
\end{aligned} \tag{4.8}$$

Remark 4.1. If a vertex $\boldsymbol{V}$ has valence 3, and there are three independent tangent lines meeting at $\boldsymbol{V}$, then any approximation of the second directional derivatives of the mesh will be compatible with G^2 continuity, as any three independent curves determine a unique curvature value at their meeting point. However, also in this case, computing the least-squares polynomial $\boldsymbol{P}$ may be useful for providing a reasonable estimate of the sought second directional derivatives.

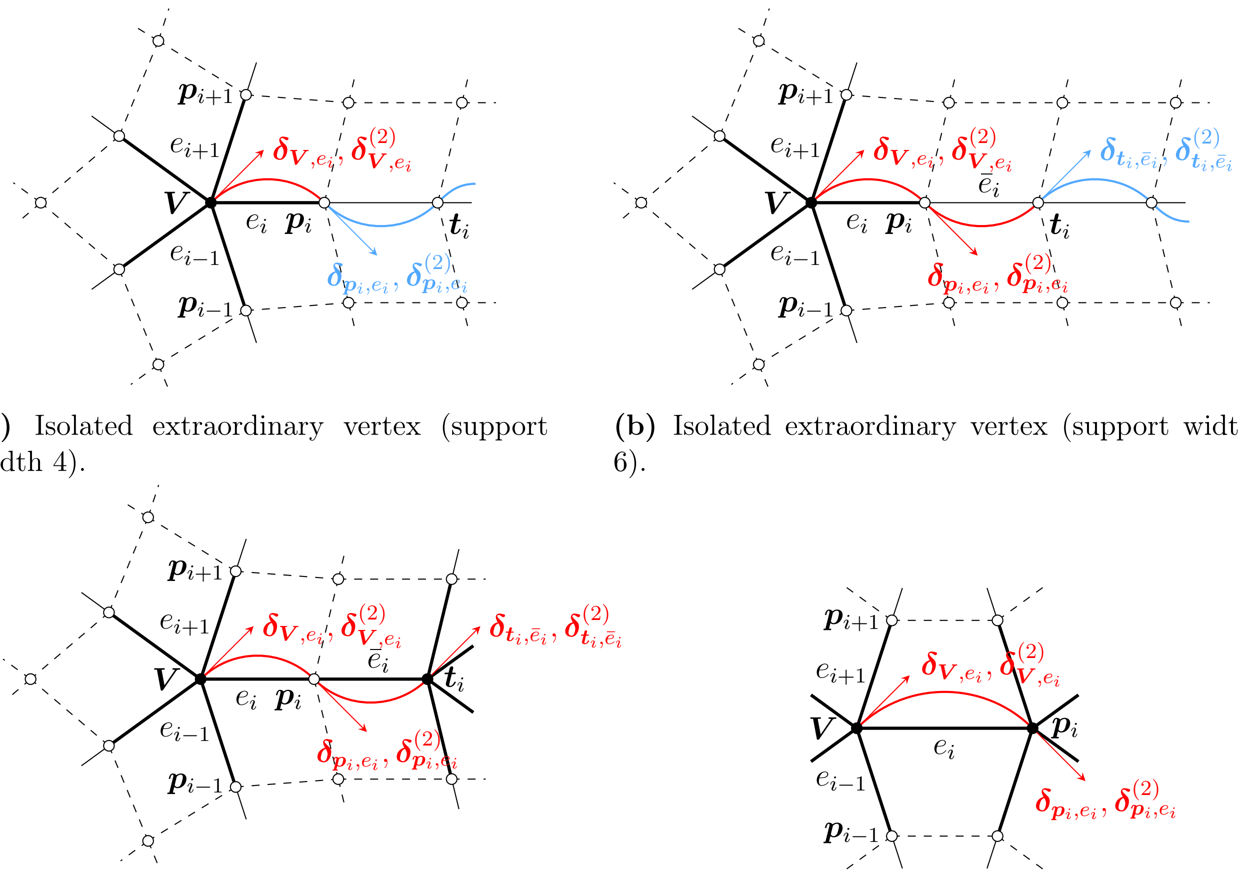
The derivatives obtained with the above steps serve as a starting point for the definition of the curve segments of the network emanating from an extraordinary vertex, in the sense that they provide the end conditions. We illustrate the construction of such curve pieces for our three running examples (classes $D^3C^1P^2S^4$, $D^5C^2P^2S^4$, $D^4C^2P^3S^6$). Figure 4.4 is a graphical reference for what we will see in the following: it shows the configuration around isolated or multiple extraordinary vertices on the same face for classes with support width 4 or 6. In this figure, the edges which cannot be processed as regular are emphasized in bold.

4.3.3.1 Class $D^3C^1P^2S^4$

This class has support width 4 and each curve piece is defined by four successive points and three parameters (see Equation (4.5)). Thus, the curve network is defined by the regular strategy, described in Section 4.3.2, everywhere except for the first ring around each extraordinary vertex $\boldsymbol{V}$ (see Figure 4.3(a)). Note that, being the fundamental functions in the considered class C^1 continuous, it would not make much sense to require G^2-compatibility at $\boldsymbol{V}$, thus we limit ourselves to requiring G^1-compatibility. We focus on constructing the piece of curve between $\boldsymbol{V}$ and $\boldsymbol{p}_i$, as the same reasoning holds for any $i = 1, \dots, n$.

We start by considering the situation in which both $\boldsymbol{p}_i$ and $\boldsymbol{t}_i$ are regular, depicted in Figure 4.4(a). The piece of curve bounded by $\boldsymbol{p}_i$ and $\boldsymbol{t}_i$ can be computed by (4.5) and belongs to the class $D^3C^1P^2S^4$. We can thus define the curve segment between $\boldsymbol{V}$ and $\boldsymbol{p}_i$ as the unique cubic polynomial $\boldsymbol{\rho}(s)$, $s \in [0, d_{e_i}]$, such that $\boldsymbol{\rho}(0) = \boldsymbol{V}$, $\boldsymbol{\rho}'(0) = \boldsymbol{\delta}_{\boldsymbol{V},e_i}$, $\boldsymbol{\rho}(d_{e_i}) = \boldsymbol{p}_i$, $\boldsymbol{\rho}'(d_{e_i}) = \boldsymbol{\delta}_{\boldsymbol{p}_i,e_i}$, where $\boldsymbol{\delta}_{\boldsymbol{V},e_i}$ is given by (4.8) and $\boldsymbol{\delta}_{\boldsymbol{p}_i,e_i}$ is the first derivative at $\boldsymbol{p}_i$ of the curve segment between $\boldsymbol{p}_i$ and $\boldsymbol{t}_i$. We remark that this technique makes so that the whole curve of the network emanating from $\boldsymbol{V}$ belongs to the considered class.

If either $\boldsymbol{p}_i$ or $\boldsymbol{t}_i$ is extraordinary (Figures 4.4(d) and 4.4(c) respectively), then the piece of curve between $\boldsymbol{V}$ and $\boldsymbol{p}_i$ is defined analogously, except that $\boldsymbol{\delta}_{\boldsymbol{p}_i,e_i}$ is given either by (4.7) with $n = 4$ (i.e., it is set to be equal to the classical Bessel estimate) if $\boldsymbol{t}_i$ is extraordinary and $\boldsymbol{p}_i$ is regular, or by (4.8) if also $\boldsymbol{p}_i$ is extraordinary (where now the least-squares polynomial to be considered is the one related to $\boldsymbol{p}_i$).



$\boldsymbol{p}_{i+1}$
e_{i+1}
$\boldsymbol{V}$
e_i
$\boldsymbol{p}_i$
e_{i-1}
$\boldsymbol{p}_{i-1}$
$\boldsymbol{t}_i$
$\bar{e}_i$
$\boldsymbol{\delta}_{\boldsymbol{V},e_i}, \boldsymbol{\delta}^{(2)}_{\boldsymbol{V},e_i}$
$\boldsymbol{\delta}_{\boldsymbol{p}_i,e_i}, \boldsymbol{\delta}^{(2)}_{\boldsymbol{p}_i,e_i}$
$\boldsymbol{\delta}_{\boldsymbol{t}_i,\bar{e}_i}, \boldsymbol{\delta}^{(2)}_{\boldsymbol{t}_i,\bar{e}_i}$

(a) Isolated extraordinary vertex (support width 4).

(b) Isolated extraordinary vertex (support width 6).

(c) Extraordinary vertices separated by a regular vertex.

(d) Adjacent extraordinary vertices.

Figure 4.4: Underlying mesh (dashed line), polylines emanating from an extraordinary vertex (solid line), mesh edges where the regular construction is not possible (thick line) and the data (values and derivatives) involved in the definition of the curve piece (in red) containing the extraordinary vertex, for classes of spline interpolants having support width 4 and 6. Note that the second derivatives $\boldsymbol{\delta}^{(2)}_{\cdot,\cdot}$ should be computed only if necessary.

4.3.3.2 Class $D^5C^2P^2S^4$

This class has support width 4 and continuity C^2, thus we aim to use it in order to construct a G^2 curve network. The reasoning is analogous to the previous case, except that in this setting we will use also information regarding the second derivatives at the endpoints of each curve segment. Recalling that the functions in the class have degree 5, we construct each piece of curve between $\boldsymbol{V}$ and $\boldsymbol{p}_i$ as the quintic polynomial $\boldsymbol{\rho}(s)$, $s \in [0, d_{e_i}]$, such that $\boldsymbol{\rho}(0) = \boldsymbol{V}$, $\boldsymbol{\rho}'(0) = \boldsymbol{\delta}_{\boldsymbol{V},e_i}$, $\boldsymbol{\rho}''(0) = \boldsymbol{\delta}^{(2)}_{\boldsymbol{V},e_i}$, $\boldsymbol{\rho}(d_{e_i}) = \boldsymbol{p}_i$, $\boldsymbol{\rho}'(d_{e_i}) = \boldsymbol{\delta}_{\boldsymbol{p}_i,e_i}$, $\boldsymbol{\rho}''(d_{e_i}) = \boldsymbol{\delta}^{(2)}_{\boldsymbol{p}_i,e_i}$. Same as for the previous class, the first and the second derivatives at $\boldsymbol{p}_i$ may be estimated from the adjacent curve piece if it exists (see again Figure 4.4(a) for a graphical outline), or, in alternative, they may be set according to (4.8).

4.3.3.3 Class $D^4C^2P^3S^6$

In this case, the fundamental functions of the class have support width 6. This means that around each extraordinary vertex there will be two rings of vertices (or, equivalently, edges) where the regular construction does not apply, as illustrated in Figures 4.3(b) and 4.4(b). We thus define the curve segment between $\boldsymbol{V}$ and $\boldsymbol{p}_i$ as the quintic polynomial $\boldsymbol{\rho}(s)$, $s \in [0, d_{e_i}]$, interpolating $\boldsymbol{\rho}(0) = \boldsymbol{V}$, $\boldsymbol{\rho}'(0) = \boldsymbol{\delta}_{\boldsymbol{V},e_i}$, $\boldsymbol{\rho}''(0) = \boldsymbol{\delta}^{(2)}_{\boldsymbol{V},e_i}$, $\boldsymbol{\rho}(d_{e_i}) = \boldsymbol{p}_i$, $\boldsymbol{\rho}'(d_{e_i}) = \boldsymbol{\delta}_{\boldsymbol{p}_i,e_i}$,

$\boldsymbol{\rho}''(d_{e_i}) = \boldsymbol{\delta}^{(2)}_{\boldsymbol{p}_i,e_i}$. Moreover, denoted by $\bar{e}_i$ the edge $\overline{\boldsymbol{p}_i,\boldsymbol{t}_i}$ and by $d_{\bar{e}_i}$ its knot interval, we define the curve segment between $\boldsymbol{p}_i$ and $\boldsymbol{t}_i$ as the quintic polynomial $\boldsymbol{\rho}(s)$, $s \in [0, d_{\bar{e}_i}]$, interpolating $\boldsymbol{\rho}(0) = \boldsymbol{p}_i$, $\boldsymbol{\rho}'(0) = \boldsymbol{\delta}_{\boldsymbol{p}_i,e_i}$, $\boldsymbol{\rho}''(0) = \boldsymbol{\delta}^{(2)}_{\boldsymbol{p}_i,e_i}$, $\boldsymbol{\rho}(d_{\bar{e}_i}) = \boldsymbol{t}_i$, $\boldsymbol{\rho}'(d_{\bar{e}_i}) = \boldsymbol{\delta}_{\boldsymbol{t}_i,\bar{e}_i}$, $\boldsymbol{\rho}''(d_{\bar{e}_i}) = \boldsymbol{\delta}^{(2)}_{\boldsymbol{t}_i,\bar{e}_i}$. Same as for the previous classes, if allowed by the configuration of the neighboring vertices, $\boldsymbol{\delta}_{\boldsymbol{t}_i,\bar{e}_i}$ and $\boldsymbol{\delta}^{(2)}_{\boldsymbol{t}_i,\bar{e}_i}$ are taken as the derivatives of a piece of curve belonging to the class $D^4C^2P^3S^6$. Otherwise, they are estimated through the construction of a least-squares polynomial around $\boldsymbol{t}_i$.

4.3.3.4 Non-closed meshes

The case of non-closed meshes (i.e., meshes with boundary) is addressed in a way similar to the extraordinary configuration described above. In fact, “regular” vertices located on the boundary of the mesh (having valence $n = 3$) or that are also corners ($n = 2$) are treated like extraordinary ones, and the curve segments that contain them are constructed by estimating the derivatives across the mesh boundary and imposing such end conditions in an interpolation problem.

4.4 Patching regular faces with augmented parameterization

Recalling the terminology introduced at the beginning of Section 4.3, we say that a mesh face (or a surface patch) is *regular* if it belongs to a region where the mesh (or the composite surface) has a grid-like structure, whereas it is called *extraordinary* otherwise. Obviously, a regular face contains no extraordinary vertices. In Figure 4.3 the regular faces are shaded in gray, and rings of extraordinary faces clearly surround extraordinary vertices, where the number of rings depends on the support width of the considered class of local interpolants.

We start by discussing the parameterization principle which lies underneath our construction focusing on regular faces. As a premise, useful to motivate our approach, we recall the well-known tensor-product parameterization, which is usually adopted when designing a composite spline surface on a regular mesh. It consists in associating with each mesh edge a knot interval in such a way that the overall parameter configuration is the cartesian product of a (possibly non-uniform) sequence of knot intervals $\boldsymbol{d} = \{d_h\}$ in the s-direction of the domain and an analogous sequence $\boldsymbol{e} = \{e_h\}$ in the t-direction. Starting from any class of fundamental functions $D^gC^kP^mS^w$ of the form considered in Section 4.2, we can straightforwardly define a C^k-continuous interpolating surface generalizing (4.4) as a collection of tensor-product patches. By virtue of the local support of the underlying class of univariate fundamental functions, each patch will depend only on a limited number of vertices and knot intervals.

Unfortunately, the tensor-product parameterization implies that the curves of a network interpolating the vertices of the mesh share the same parameterization in each direction. Thus it is not well suited for designing a surface which interpolates a general network of curves, where each curve may be parameterized independently of the others. On the other hand, since the quality of each curve depends on its particular parameterization, it is intuitive that changing the parameterization of the curves compatibly with the tensor product, e.g., assigning to each curve an average parameterization, would mean losing or altering information that is crucial for the quality of the interpolant. Indeed, the fact that the strict structure of the tensor product may compromise the quality of an interpolation method is well known (see, e.g., [FHK02, Section 7.5.1]).

In the following we will design a surface which interpolates a curve network without altering

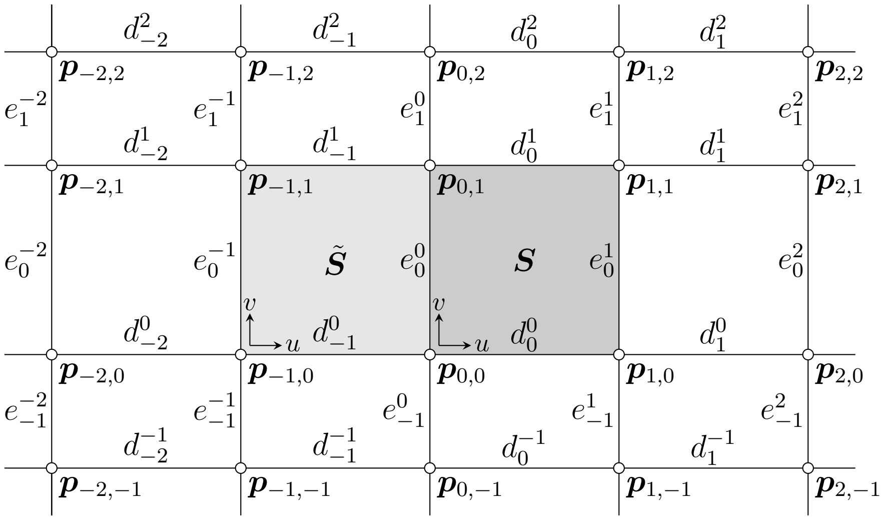


Figure 4.5: Configuration and labeling of vertices and knot intervals around a face (in dark gray) surrounded by a sufficient number of regular vertices. The related surface patch $\boldsymbol{S}$ with augmented parameterization is defined by (4.9).

the parameterization of the given curves.

We construct a composite surface, where each mesh face gives rise to one surface patch $\boldsymbol{S}$ parameterized over the domain $[0,1]^2$, whose parameters are denoted by (u,v). Figure 4.5 illustrates the configuration and the labeling of vertices and knot intervals around a face surrounded by a sufficient number of regular vertices. For any $(u,v) \in [0,1]^2$, we compute two local sequences of knot intervals $\boldsymbol{d}$ and $\boldsymbol{e}$, each of them being made of $w-1$ elements, of the form

$$\boldsymbol{d} = \boldsymbol{d}(v) = \Big(d_h(v),\ h = -\frac{w}{2}+1,\ldots,\frac{w}{2}-1\Big),$$

and

$$\boldsymbol{e} = \boldsymbol{e}(u) = \Big(e_h(u),\ h = -\frac{w}{2}+1,\ldots,\frac{w}{2}-1\Big),$$

and we derive a couple of coordinates (s,t) defined as

$$s = u\,d_0(v), \qquad t = v\,e_0(u),$$

such that $s \in [0, d_0(v)]$ and $t \in [0, e_0(u)]$.

We set the value at (u,v) of the patch $\boldsymbol{S}$ which interpolates the vertices of the considered face to be equal to the value at (s,t) of a tensor-product patch $\boldsymbol{T}$ parameterized over $\boldsymbol{d}$ and $\boldsymbol{e}$ as follows:

$$\boldsymbol{S}(u,v) := \boldsymbol{T}(s,t) = \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{i,j} \Psi_{i,j}(s,t;\boldsymbol{d},\boldsymbol{e}), \tag{4.9}$$

where the *control points* $\boldsymbol{p}_{i,j}$ represent a local grid of mesh vertices, and

$$\Psi_{i,j}(s,t;\boldsymbol{d},\boldsymbol{e}) := \psi_i(s;\boldsymbol{d})\,\psi_j(t;\boldsymbol{e}).$$

In particular, from the above expression, the bivariate fundamental function $\Psi_{i,j}$ associated with $\boldsymbol{p}_{i,j}$ is defined as the product of two fundamental functions of the form (4.4), respectively parameterized over the knot intervals sequences $\boldsymbol{d}$ and $\boldsymbol{e}$, which are different for each couple (u,v). We aim to choose the functions $\boldsymbol{d}(v)$ and $\boldsymbol{e}(u)$ so that:

i) the boundary of the patch, which is described by (4.9) with either u or v equal to 0 or 1, coincides with segments of curves of the network;

ii) C^k continuity of the surface holds.

To prescribe the functions $\boldsymbol{d}(v)$ and $\boldsymbol{e}(u)$ we have pursued the maximum locality of the construction, in such a way that the surface is not affected by data that may not be representative of the local configuration of interpolation points. In particular, condition i) can be easily satisfied by interpolating the parameters associated with opposing edges. We will thus set $d_h(v), e_h(u) \in \mathcal{P}_{2k+2}$ (the space of polynomials of order $2k+2$, i.e., of degree at most $2k+1$) such that

$$\begin{aligned} d_h(0) &= d_h^0, & d_h(1) &= d_h^1, & d_h^{(r)}(0) &= d_h^{(r)}(1) = 0, & r &= 1, \dots, k, \\ e_h(0) &= e_h^0, & e_h(1) &= e_h^1, & e_h^{(r)}(0) &= e_h^{(r)}(1) = 0, & r &= 1, \dots, k. \end{aligned} \tag{4.10}$$

For each surface patch, we may note that the knot intervals associated with its boundary need no longer form a rectangle. This parameterization technique, which we call *augmented parameterization*, allows the parameters to vary smoothly in between one curve and another, while interpolating the parameterization of the curves of the network. In this way, the boundary of each patch coincides exactly with a piece of curve of the network, which means that the curve network is embedded in the interpolating surface, as proven below.

Proposition 4.1. *The composite surface with augmented parameterization interpolates the curves of the network.*

Proof. We consider the boundary of the patch, which is described by (4.9) with either u or v equal to 0 or 1, and prove that it coincides with segments of curves of the network.

In particular, let us suppose $u = 0$. Then $s = 0$ and, observing that $\psi_i(0; \boldsymbol{d}(v)) = 1$ if $i = 0$ and it is zero otherwise, equation (4.9) becomes

$$\begin{aligned} \boldsymbol{S}(0, v) = \boldsymbol{T}(0, t) &= \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{i,j}\, \psi_i(0; \boldsymbol{d}(v))\, \psi_j(t; \boldsymbol{e}(0)) \\ &= \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{0,j}\, \psi_j \left(t; e^0_{-\frac{w}{2}+1}, \dots, e^0_{\frac{w}{2}-1} \right), \end{aligned}$$

which is exactly the curve of the network interpolating the sequence of points $\boldsymbol{p}_{0,j}$.

Similarly, when $u = 1$, then $s = d_0(v)$ and $\psi_i(d_0(v); \boldsymbol{d}(v)) = 1$ if $i = 1$ and it is zero otherwise; thus, from (4.9),

$$\begin{aligned} \boldsymbol{S}(1, v) = \boldsymbol{T}(d_0(v), t) &= \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{i,j}\, \psi_i(d_0(v); \boldsymbol{d}(v))\, \psi_j(t; \boldsymbol{e}(1)) \\ &= \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{1,j}\, \psi_j \left(t; e^1_{-\frac{w}{2}+1}, \dots, e^1_{\frac{w}{2}-1} \right), \end{aligned}$$

which is exactly the curve of the network interpolating the sequence of points $\boldsymbol{p}_{1,j}$.

An analogous reasoning holds for the other patch boundaries $v = 0, 1$. □

Proposition 4.2. *The composite surface with augmented parameterization built upon a class of C^k-continuous fundamental functions is G^k-continuous.*

Proof. We consider two adjacent patches $\boldsymbol{S}$ and $\tilde{\boldsymbol{S}}$ (see Figure 4.5). Without loss of generality, we can assume that their common boundary corresponds to $u = 0$ for $\boldsymbol{S}$ and $u = 1$ for $\tilde{\boldsymbol{S}}$, thus $\boldsymbol{S}$ and $\tilde{\boldsymbol{S}}$ share the same knot intervals sequence on the boundary $\boldsymbol{e}(0) = \tilde{\boldsymbol{e}}(1) = (e^0_{-\frac{w}{2}+1}, \ldots, e^0_{\frac{w}{2}-1})$, while their knot intervals sequences in the transversal direction are respectively $\boldsymbol{d}(v) = (d_{-\frac{w}{2}+1}(v), \ldots, d_{\frac{w}{2}-1}(v))$ and $\tilde{\boldsymbol{d}}(v) = (d_{-\frac{w}{2}}(v), \ldots, d_{\frac{w}{2}-2}(v))$.
By Proposition 4.1, it is trivially verified that C^k continuity holds along the common boundary. In the cross-boundary direction u, the rth-order derivative of $\boldsymbol{S}$ is given by

$$\frac{\partial^r}{\partial u^r}\boldsymbol{S}(u,v) = \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{i,j} \left[\frac{\partial^r}{\partial u^r}\psi_i\left(ud_0(v); \boldsymbol{d}(v)\right)\, \psi_j\left(ve_0(u); \boldsymbol{e}(u)\right) \right.$$
$$\left. + \psi_i\left(ud_0(v); \boldsymbol{d}(v)\right)\, \frac{\partial^r}{\partial u^r}\psi_j\left(ve_0(u); \boldsymbol{e}(u)\right) \right].$$

We observe that

$$\frac{\partial}{\partial u}\psi_j\left(ve_0(u); \boldsymbol{e}(u)\right) = \sum_{h=-\frac{w}{2}+1}^{\frac{w}{2}-1} \frac{\partial \psi_j}{\partial e_h}\frac{\partial e_h}{\partial u},$$

and thus, recalling that $e_0(0) = e^0_0$,

$$\left.\frac{\partial}{\partial u}\psi_j\left(ve_0(u); \boldsymbol{e}(u)\right)\right|_{u=0} = 0,$$

since $e'_h(0) = 0$ by construction (see (4.10)). Again from (4.10), if we iterate the differentiation process (or, more concisely, use Faà di Bruno's formula [FdB57, Joh02]), it can be easily verified that the higher-order derivatives of ψ_j vanish at $u = 0$ up to order k. Thus, observing that $\frac{\partial^r}{\partial u^r}\psi_{\frac{w}{2}}(ud_0(v); \boldsymbol{d}(v))\big|_{u=0} = 0$, for $r = 1, \ldots, k$,

$$\left.\frac{\partial^r}{\partial u^r}\boldsymbol{S}(u,v)\right|_{u=0} = \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}-1} \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{i,j}\, \left.\frac{\partial^r}{\partial u^r}\psi_i\left(ud_0(v); \boldsymbol{d}(v)\right)\right|_{u=0} \psi_j\left(ve^0_0; \boldsymbol{e}(0)\right). \tag{4.11a}$$

Analogously, considering the other patch $\tilde{\boldsymbol{S}}$, we have

$$\left.\frac{\partial^r}{\partial u^r}\tilde{\boldsymbol{S}}(u,v)\right|_{u=1} = \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}-1} \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{i,j}\, \left.\frac{\partial^r}{\partial u^r}\psi_i\left(ud_{-1}(v); \tilde{\boldsymbol{d}}(v)\right)\right|_{u=1} \psi_j\left(ve^0_0; \boldsymbol{e}(0)\right), \tag{4.11b}$$

since $\tilde{\boldsymbol{e}}(1) = \boldsymbol{e}(0)$ and $\frac{\partial^r}{\partial u^r}\psi_{-\frac{w}{2}}(ud_{-1}(v); \tilde{\boldsymbol{d}}(v))\big|_{u=1} = 0$, for $r = 1, \ldots, k$.

The derivative in the u-direction can be related to the derivative with respect to s through

$$\frac{\partial^r}{\partial u^r} = \frac{\partial^r}{\partial s^r}\left(\frac{\partial s}{\partial u}\right)^r,$$

where $s = ud_0(v)$ for $\boldsymbol{S}$ and $s = ud_{-1}(v)$ for $\tilde{\boldsymbol{S}}$.

We observe that, by construction, the knot intervals in $\boldsymbol{d}$ and $\tilde{\boldsymbol{d}}$ are equal, except for the first and the last element respectively of $\boldsymbol{d}$ and of $\tilde{\boldsymbol{d}}$, which nevertheless do not influence the value of (4.11a) and (4.11b). Thus, being the family of fundamental functions ψ_i C^k-continuous with respect to s, it holds

$$\left.\frac{\partial^r}{\partial s^r}\psi_i(s;\boldsymbol{d}(v))\right|_{s=0} = \left.\frac{\partial^r}{\partial s^r}\psi_i(s;\tilde{\boldsymbol{d}}(v))\right|_{s=d_{-1}(v)},$$

which results in C^k continuity w.r.t. the variable s across the considered boundary between patches $\boldsymbol{S}$ and $\tilde{\boldsymbol{S}}$. Moreover,

$$\left.\frac{\partial^r}{\partial u^r}\boldsymbol{S}(u,v)\right|_{u=0} = \left(\frac{d_0(v)}{d_{-1}(v)}\right)^r \left.\frac{\partial^r}{\partial u^r}\tilde{\boldsymbol{S}}(u,v)\right|_{u=1},$$

which means that G^k continuity holds across the boundary w.r.t. the parameters in the (u,v)-domain. □

Remark 4.2. A surface patch defined by (4.9) is C^∞ in its interior. In fact, it is obtained by the composition of pieces of ψ_i (or ψ_j), which are C^∞, with the C^∞ functions d_h (or e_h).

4.5 Patching of faces containing extraordinary vertices

In this section we address the definition of the surface patches that are associated with faces containing extraordinary vertices, for which the regular construction described above is not applicable. In particular, we focus on the definition of G^1 and G^2 surface patches that, exploiting again the augmented parameterization, guarantee the interpolation of the curve network, preserving the original parameterization of each curve, and provide a smooth connection with the surrounding surface patches.

We point out that the proposed construction applies to faces with an arbitrary number of extraordinary vertices.

4.5.1 Augmented bicubically blended Coons patch with Gregory correction

We consider a modified form of the classical bicubically blended Coons patch (see, e.g., [Far02, Chapter 22]) that we will call *augmented Coons-Gregory patch.*

Figure 4.6 shows the quantities needed for the definition of such a patch. For any $(u,v) \in [0,1]^2$, we determine $d_0(v)$ and $e_0(u)$ as described in Section 4.4 (more precisely, as in equation (4.10)). Moreover, we consider as blending functions the cubic Hermite basis functions on $[0,1]$, arranged in the vector

$$\boldsymbol{H}(u) := \left(-1, 2u^3-3u^2+1, -2u^3+3u^2, u^3-2u^2+u, u^3-u^2\right)^T.$$

We compute the value of the augmented patch $\boldsymbol{X}$ according to

$$\boldsymbol{X}(u,v) = -\boldsymbol{H}(u)^T M(u,v)\boldsymbol{H}(v),$$

where the patch matrix is

$$M(u,v) := \begin{pmatrix} 0 & \boldsymbol{\gamma}_0(s_0) & \boldsymbol{\gamma}_2(s_2) & e_0(u)\boldsymbol{\chi}_0(s_0) & e_0(u)\boldsymbol{\chi}_2(s_2) \\ \boldsymbol{\gamma}_3(t_3) & \boldsymbol{p}_0 & \boldsymbol{p}_3 & e_0^0\boldsymbol{\gamma}_3'(0) & e_0^0\boldsymbol{\gamma}_3'(e_0^0) \\ \boldsymbol{\gamma}_1(t_1) & \boldsymbol{p}_1 & \boldsymbol{p}_2 & e_0^1\boldsymbol{\gamma}_1'(0) & e_0^1\boldsymbol{\gamma}_1'(e_0^1) \\ d_0(v)\boldsymbol{\chi}_3(t_3) & d_0^0\boldsymbol{\gamma}_0'(0) & d_0^1\boldsymbol{\gamma}_2'(0) & \multicolumn{2}{c}{\multirow{2}{*}{$\Omega_{1,1}$}} \\ d_0(v)\boldsymbol{\chi}_1(t_1) & d_0^0\boldsymbol{\gamma}_0'(d_0^0) & d_0^1\boldsymbol{\gamma}_2'(d_0^1) & & \end{pmatrix}, \tag{4.12}$$

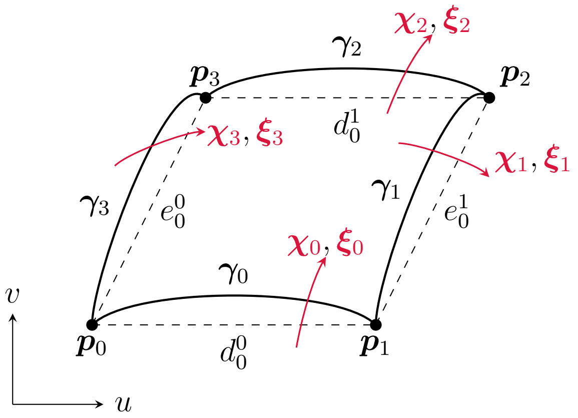


Figure 4.6: Schematic interpretation of the quantities needed for the definition of an augmented Coons-Gregory patch (the second-order cross-boundary fields $\boldsymbol{\xi}_i$ are needed only for the biquintically blended patch).

whose entries are specified as follows:

- $\boldsymbol{p}_0, \boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3$ are the patch corners, i.e., the four vertices of the mesh face;
- $\boldsymbol{\gamma}_0(s_0), \boldsymbol{\gamma}_1(t_1), \boldsymbol{\gamma}_2(s_2), \boldsymbol{\gamma}_3(t_3)$ are the boundary curves, which are pieces of the curves of the curve network, with "local" coordinates
$$s_0 := ud_0^0, \qquad t_1 := ve_0^1, \qquad s_2 := ud_0^1, \qquad t_3 := ve_0^0;$$
- $\boldsymbol{\gamma}_0'(0), \boldsymbol{\gamma}_3'(0), \boldsymbol{\gamma}_0'(d_0^0), \boldsymbol{\gamma}_1'(0), \boldsymbol{\gamma}_1'(e_0^1), \boldsymbol{\gamma}_2'(d_0^1), \boldsymbol{\gamma}_2'(0), \boldsymbol{\gamma}_3'(e_0^0)$ are the first derivatives of the boundary curves at the corners;
- $\boldsymbol{\chi}_0(s_0), \boldsymbol{\chi}_1(t_1), \boldsymbol{\chi}_2(s_2), \boldsymbol{\chi}_3(t_3)$ are the cross-boundary first derivatives;
- $\Omega_{1,1}$ is the twist vectors matrix, possibly with *Gregory correction for twist-incompatibility*, i.e.:

$$\Omega_{1,1} := \left(\begin{array}{cc} d_0^0 e_0^0 \dfrac{u\boldsymbol{\chi}_3'(0) + v\boldsymbol{\chi}_0'(0)}{u+v} & d_0^1 e_0^0 \dfrac{u\boldsymbol{\chi}_3'(e_0^0) + (1-v)\boldsymbol{\chi}_2'(0)}{u+(1-v)} \\ d_0^0 e_0^1 \dfrac{(1-u)\boldsymbol{\chi}_1'(0) + v\boldsymbol{\chi}_0'(d_0^0)}{(1-u)+v} & d_0^1 e_0^1 \dfrac{(1-u)\boldsymbol{\chi}_1'(e_0^1) + (1-v)\boldsymbol{\chi}_2'(d_0^1)}{(1-u)+(1-v)} \end{array} \right). \tag{4.13}$$

To ensure that all the boundary curves and the prescribed derivatives are interpolated, a suitable scaling factor is associated with the corresponding entries of the patch matrix M. We observe that such a factor is different at each corner, but constant w.r.t. u, v for the derivatives along the boundary and for the twist vectors. Conversely, the scaling factor associated with the cross-boundary derivatives changes for each (u, v). As a consequence, in analogy to the regular case described in Section 4.4, the definition of the augmented patch changes pointwise.

Since the curve network constructed in Section 4.3 is G^1-compatible, the Coons patch defined above ensures G^1 continuity at the corners, along and across the boundaries, provided that the cross-boundary first derivatives are chosen in a proper way, as we detail in the following.

4.5.2 Construction of the cross-boundary tangent fields

We address the problem of suitably prescribing the tangent field $\boldsymbol{\chi}_i$ across $\boldsymbol{\gamma}_i$. Denoted by $\boldsymbol{S}_i$ the surface patch that is adjacent to $\boldsymbol{X}$ and shares with it the curve $\boldsymbol{\gamma}_i$, we distinguish the two situations in which $\boldsymbol{S}_i$ is regular or not. If $\boldsymbol{S}_i$ is regular, the tangent field $\boldsymbol{\chi}_i$ for $\boldsymbol{X}$ is the cross-boundary derivative of $\boldsymbol{S}_i$. If $\boldsymbol{S}_i$ is not regular, let us suppose that $t \in [0, \ell]$ is the variable which describes the boundary curve $\boldsymbol{\gamma}_i$. Then we define the cross-boundary tangent field as

$$\boldsymbol{\chi}_i(t) = a(t)\boldsymbol{\gamma}_i'(t) + b(t)\boldsymbol{r}(t), \tag{4.14}$$

with suitable $a(t), b(t), \boldsymbol{r}(t)$ that generate a G^1 surface. We observe that, if the parameterization is uniform, in the regular case, the tangent field $\boldsymbol{\chi}_i$ has the same degree g as the chosen class of fundamental functions. Similarly, also in the extraordinary case we construct $\boldsymbol{\chi}_i$ as a polynomial of degree at most g. Since $\boldsymbol{\gamma}_i'$ has degree $g-1$, then $a(t)$ has to be at most linear and, in analogy, we also choose $b(t)$ of degree 1. As a consequence, $\boldsymbol{r}(t)$ has degree at most $g-1$. In particular, our numerical tests have shown that satisfactory results can be obtained defining $\boldsymbol{r}$ by either linear or quadratic interpolation. For the linear case, we interpolate the vectors $\boldsymbol{r}_0 = \boldsymbol{\gamma}_i'(0) \times \boldsymbol{n}_0$ and $\boldsymbol{r}_1 = \boldsymbol{\gamma}_i'(\ell) \times \boldsymbol{n}_1$, where $\boldsymbol{n}_0$ and $\boldsymbol{n}_1$ are the normal vectors to the curve network at the endpoints of $\boldsymbol{\gamma}_i$. In the quadratic case, an additional interpolation vector is estimated as $\boldsymbol{r}_{\frac{1}{2}} = \boldsymbol{\gamma}_i'\left(\frac{\ell}{2}\right) \times \boldsymbol{n}_{\frac{1}{2}}$, where $\boldsymbol{n}_{\frac{1}{2}}$ is the average of the normals to the two faces sharing the edge associated with $\boldsymbol{\gamma}_i$. To derive a and b we proceed as follows. The values of a and b at the endpoints of $\boldsymbol{\gamma}_i$ are fixed by the requirement that $\boldsymbol{\chi}_i(0)$ and $\boldsymbol{\chi}_i(\ell)$ be equal to the first derivatives of $\boldsymbol{\gamma}_{i-1}$ and $\boldsymbol{\gamma}_{i+1}$ at $\boldsymbol{\gamma}_i(0)$, $\boldsymbol{\gamma}_i(\ell)$. Thus a and b are uniquely determined by linear interpolation of their values at the endpoints.

4.5.3 Augmented biquintically blended Coons patch with Gregory correction

If G^2 smoothness is sought, the degree of the blending functions used for the definition of a Coons patch must be increased. In this case, we exploit as blending functions the quintic Hermite basis functions, arranged in the vector

$$\begin{aligned}\boldsymbol{H}(u) := \Big(&-1, -6u^5+15u^4-10u^3+1, 6u^5-15u^4+10u^3, -3u^5+8u^4-6u^3+u,\\ &-3u^5+7u^4-4u^3, -\frac{1}{2}u^5+\frac{3}{2}u^4-\frac{3}{2}u^3+\frac{1}{2}u^2, \frac{1}{2}u^5-u^4+\frac{1}{2}u^3\Big)^T.\end{aligned}$$

We compute the value of the augmented patch $\boldsymbol{X}$ according to

$$\boldsymbol{X}(u,v) = -\boldsymbol{H}(u)^T\tilde{M}(u,v)\boldsymbol{H}(v).$$

The patch matrix is now

$$\tilde{M}(u,v) := \left(\begin{array}{cccc|cc}
 & & & & (e_0(u))^2\boldsymbol{\xi}_0(s_0) & (e_0(u))^2\boldsymbol{\xi}_2(s_2)\\
 & & M(u,v) & & (e_0^0)^2\boldsymbol{\gamma}_3''(0) & (e_0^0)^2\boldsymbol{\gamma}_3''(e_0^0)\\
 & & & & (e_0^1)^2\boldsymbol{\gamma}_1''(0) & (e_0^1)^2\boldsymbol{\gamma}_1''(e_0^1)\\
 & & & & \multicolumn{2}{c}{\Omega_{1,2}}\\
\cline{1-4}
(d_0(v))^2\boldsymbol{\xi}_3(t_3) & (d_0^0)^2\boldsymbol{\gamma}_0''(0) & (d_0^1)^2\boldsymbol{\gamma}_2''(0) & \Omega_{2,1} & \multicolumn{2}{c}{\Omega_{2,2}}\\
(d_0(v))^2\boldsymbol{\xi}_1(t_1) & (d_0^0)^2\boldsymbol{\gamma}_0''(d_0^0) & (d_0^1)^2\boldsymbol{\gamma}_2''(d_0^1) & & &
\end{array}\right),$$

where M is given by (4.12) but using the definition for $\Omega_{1,1}$ provided below, instead of (4.13), and the remaining entries are specified as follows (refer to Figure 4.6 for a graphical interpretation):

- $\boldsymbol{\gamma}_0''(0), \boldsymbol{\gamma}_3''(0), \boldsymbol{\gamma}_0''(d_0^0), \boldsymbol{\gamma}_1''(0), \boldsymbol{\gamma}_1''(e_0^1), \boldsymbol{\gamma}_2''(d_0^1), \boldsymbol{\gamma}_2''(0), \boldsymbol{\gamma}_3''(e_0^0)$ are the second derivatives of the boundary curves at the corners;

- $\boldsymbol{\xi}_0(s_0), \boldsymbol{\xi}_1(t_1), \boldsymbol{\xi}_2(s_2), \boldsymbol{\xi}_3(t_3)$ are the cross-boundary second derivatives;
- the mixed derivatives matrices, possibly with *Gregory correction for twist-incompatibility*, are defined as follows:

$$\Omega_{1,1} := \begin{pmatrix} d_0^0 e_0^0 \dfrac{u^2\boldsymbol{\chi}_3'(0) + v^2\boldsymbol{\chi}_0'(0)}{u^2+v^2} & d_0^1 e_0^0 \dfrac{u^2\boldsymbol{\chi}_3'(e_0^0) + (1-v)^2\boldsymbol{\chi}_2'(0)}{u^2+(1-v)^2} \\ d_0^0 e_0^1 \dfrac{(1-u)^2\boldsymbol{\chi}_1'(0) + v^2\boldsymbol{\chi}_0'(d_0^0)}{(1-u)^2+v^2} & d_0^1 e_0^1 \dfrac{(1-u)^2\boldsymbol{\chi}_1'(e_0^1) + (1-v)^2\boldsymbol{\chi}_2'(d_0^1)}{(1-u)^2+(1-v)^2} \end{pmatrix},$$

$$\Omega_{1,2} := \begin{pmatrix} d_0^0 (e_0^0)^2 \dfrac{u^2\boldsymbol{\chi}_3''(0) + v^2\boldsymbol{\xi}_0'(0)}{u^2+v^2} & d_0^1 (e_0^0)^2 \dfrac{u^2\boldsymbol{\chi}_3''(e_0^0) + (1-v)^2\boldsymbol{\xi}_2'(0)}{u^2+(1-v)^2} \\ d_0^0 (e_0^1)^2 \dfrac{(1-u)^2\boldsymbol{\chi}_1''(0) + v^2\boldsymbol{\xi}_0'(d_0^0)}{(1-u)^2+v^2} & d_0^1 (e_0^1)^2 \dfrac{(1-u)^2\boldsymbol{\chi}_1''(e_0^1) + (1-v)^2\boldsymbol{\xi}_2'(d_0^1)}{(1-u)^2+(1-v)^2} \end{pmatrix},$$

$$\Omega_{2,1} := \begin{pmatrix} (d_0^0)^2 e_0^0 \dfrac{u^2\boldsymbol{\xi}_3'(0) + v^2\boldsymbol{\chi}_0''(0)}{u^2+v^2} & (d_0^1)^2 e_0^0 \dfrac{u^2\boldsymbol{\xi}_3'(e_0^0) + (1-v)^2\boldsymbol{\chi}_2''(0)}{u^2+(1-v)^2} \\ (d_0^0)^2 e_0^1 \dfrac{(1-u)^2\boldsymbol{\xi}_1'(0) + v^2\boldsymbol{\chi}_0''(d_0^0)}{(1-u)^2+v^2} & (d_0^1)^2 e_0^1 \dfrac{(1-u)^2\boldsymbol{\xi}_1'(e_0^1) + (1-v)^2\boldsymbol{\chi}_2''(d_0^1)}{(1-u)^2+(1-v)^2} \end{pmatrix},$$

$$\Omega_{2,2} := \begin{pmatrix} (d_0^0 e_0^0)^2 \dfrac{u^2\boldsymbol{\xi}_3''(0) + v^2\boldsymbol{\xi}_0''(0)}{u^2+v^2} & (d_0^1 e_0^0)^2 \dfrac{u^2\boldsymbol{\xi}_3''(e_0^0) + (1-v)^2\boldsymbol{\xi}_2''(0)}{u^2+(1-v)^2} \\ (d_0^0 e_0^1)^2 \dfrac{(1-u)^2\boldsymbol{\xi}_1''(0) + v^2\boldsymbol{\xi}_0''(d_0^0)}{(1-u)^2+v^2} & (d_0^1 e_0^1)^2 \dfrac{(1-u)^2\boldsymbol{\xi}_1''(e_0^1) + (1-v)^2\boldsymbol{\xi}_2''(d_0^1)}{(1-u)^2+(1-v)^2} \end{pmatrix}.$$

4.5.4 Construction of the cross-boundary second-order derivative fields

Now we have to address the problem of suitably prescribing the second-order field $\boldsymbol{\xi}_i$ across $\boldsymbol{\gamma}_i$. Using the same notation as above, if $\boldsymbol{S}_i$ is regular, the tangent field $\boldsymbol{\xi}_i$ for $\boldsymbol{X}$ is the cross-boundary derivative of $\boldsymbol{S}_i$. If $\boldsymbol{S}_i$ is not regular, assuming again $t \in [0, \ell]$ as the variable which describes the boundary curve $\boldsymbol{\gamma}_i$, we define the cross-boundary second-order derivative field as

$$\boldsymbol{\xi}_i(t) = a^2(t)\boldsymbol{\gamma}_i''(t) + c(t)\boldsymbol{\gamma}_i'(t) + d(t)\boldsymbol{r}(t) + 2a(t)b(t)\boldsymbol{r}'(t) + b^2(t)\boldsymbol{w}(t),$$

with $a(t), b(t), \boldsymbol{r}(t)$ being the same as in (4.14) and $c(t), d(t), \boldsymbol{w}(t)$ suitably determined as described below. This condition ensures G^2 continuity across $\boldsymbol{\gamma}_i$ (see [Her96, Theorem C]). We observe that, if the parameterization is uniform, in the regular case, the second-order derivative field $\boldsymbol{\xi}_i$ has the same degree g as the chosen class of fundamental functions. Thus, since $\boldsymbol{\gamma}_i'$ has degree $g-1$, then $c(t)$ has to be at most linear. In analogy, we also choose $d(t)$ of degree 1. To keep the number of degrees of freedom as low as possible, we set the degree of $\boldsymbol{w}$ equal to 1. Intuitively, $\boldsymbol{w}$ represents how much the surface deviates from its tangent plane. Thus, to complete its definition, we specify the value of $\boldsymbol{w}$ at the endpoints of $\boldsymbol{\gamma}_i$ by using the approach suggested in [Her96, Section 8], i.e., we set

$$\boldsymbol{w}(0) = (\mu^2\kappa_1 + \nu^2\kappa_2)\boldsymbol{n},$$
$$\boldsymbol{w}(\ell) = (\bar{\mu}^2\bar{\kappa}_1 + \bar{\nu}^2\bar{\kappa}_2)\bar{\boldsymbol{n}},$$

where $\boldsymbol{n}, \kappa_1, \kappa_2$ are the surface normal and the principal curvatures at $\boldsymbol{\gamma}_i(0)$, and $\bar{\boldsymbol{n}}, \bar{\kappa}_1, \bar{\kappa}_2$ are the analogous quantities at $\boldsymbol{\gamma}_i(\ell)$. The values for (μ, ν) and $(\bar{\mu}, \bar{\nu})$ are the coordinates of $\boldsymbol{r}(0)$ and $\boldsymbol{r}(\ell)$ expressed in the local coordinate systems of the principal directions $(\boldsymbol{k}_1, \boldsymbol{k}_2)$ and $(\bar{\boldsymbol{k}}_1, \bar{\boldsymbol{k}}_2)$ in the tangent planes, i.e.,

$$\boldsymbol{r}(0) = \mu\boldsymbol{k}_1 + \nu\boldsymbol{k}_2,$$

$$\boldsymbol{r}(\ell) = \bar{\mu}\bar{\boldsymbol{k}}_1 + \bar{\nu}\bar{\boldsymbol{k}}_2.$$

To derive c and d we proceed as follows. The values of c and d at the endpoints of $\boldsymbol{\gamma}_i$ are fixed by the requirement that $\boldsymbol{\xi}_i(0)$ and $\boldsymbol{\xi}_i(\ell)$ be equal to the second derivatives of $\boldsymbol{\gamma}_{i-1}$ and $\boldsymbol{\gamma}_{i+1}$ at $\boldsymbol{\gamma}_i(0)$, $\boldsymbol{\gamma}_i(\ell)$. Thus c and d are uniquely determined by linear interpolation of their values at the endpoints.

4.6 Gordon surfaces with augmented parameterization

Gordon surfaces (see, e.g., [Far02, Section 22.7]) are a generalization of Coons patches, in the sense that they interpolate a prescribed network of curves and not just four boundary curves. Depending on the univariate interpolation method that is exploited in a Gordon scheme, the construction has either global or local nature. In fact, if we use blending functions without local support, typically in the case of Lagrange polynomial or spline interpolation, the scheme is global because all the given curves influence the surface at any point. On the contrary, if we use locally supported blending functions, like the fundamental functions introduced in Section 4.2, we can achieve locality in the solution of the network-interpolation problem. Moreover, also in this case the augmented parameterization strategy can be exploited to construct high-quality Gordon surfaces, as we briefly describe below.

Let us start by considering the case in which the initial data consist in a regular curve network. In this situation, two families of space curves $\boldsymbol{f}_j(s_j), \boldsymbol{g}_i(t_i)$ are given and each curve has its own parameterization (as we have stressed here by explicitly indicating the related parameter). We denote by $\boldsymbol{p}_{i,j} := \boldsymbol{f}_j \cap \boldsymbol{g}_i$ the vertices of the network, i.e., the intersections between curves belonging to different families.
We use the same approach described in Section 4.4 and combine the curves and their intersections with local fundamental functions, giving rise to an augmented Gordon scheme with the property of locality. In particular, following the notation introduced there, we define each patch of the composite augmented Gordon surface as (cf. (4.9))

$$\begin{aligned}\boldsymbol{S}(u,v) = &\sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{f}_j(s_j)\,\psi_j(ve_0(u);\boldsymbol{e}(u)) + \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{g}_i(t_i)\,\psi_i(ud_0(v);\boldsymbol{d}(v)) \\ &- \sum_{i=-\frac{w}{2}+1}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}+1}^{\frac{w}{2}} \boldsymbol{p}_{i,j}\,\psi_i(ud_0(v);\boldsymbol{d}(v))\,\psi_j(ve_0(u);\boldsymbol{e}(u)).\end{aligned}$$

If extraordinary vertices are present, the above formulation applies to patches in the regular regions of the network, while regions surrounding extraordinary vertices are patched by means of augmented Coons-Gregory patches defined as in Section 4.5.

4.7 Examples

In this section we show some examples of interpolating surfaces obtained with our approach based on the augmented parameterization, in both regular and extraordinary configurations of mesh vertices.

Given a mesh with a region of non-evenly spaced vertices to be interpolated, Figures 4.8 and 4.9 compare the surfaces obtained by a classical tensor-product approach and our method. In Figure 4.8, interpolation artifacts are evident for the tensor product case in the critical region

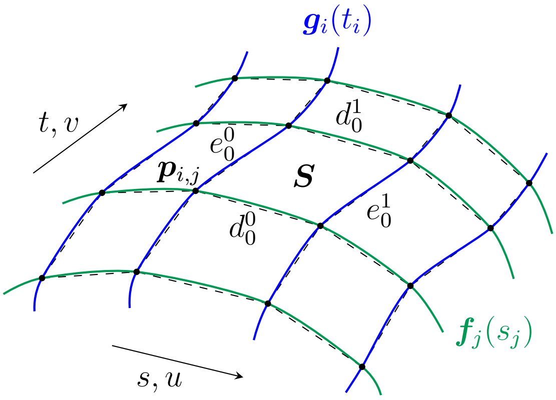


Figure 4.7: Schematic interpretation of a regular network-interpolation problem, where two families of curves are given. Each curve has its own parameterization, which will be retained in the final surface thanks to the use of the augmented parameterization strategy.

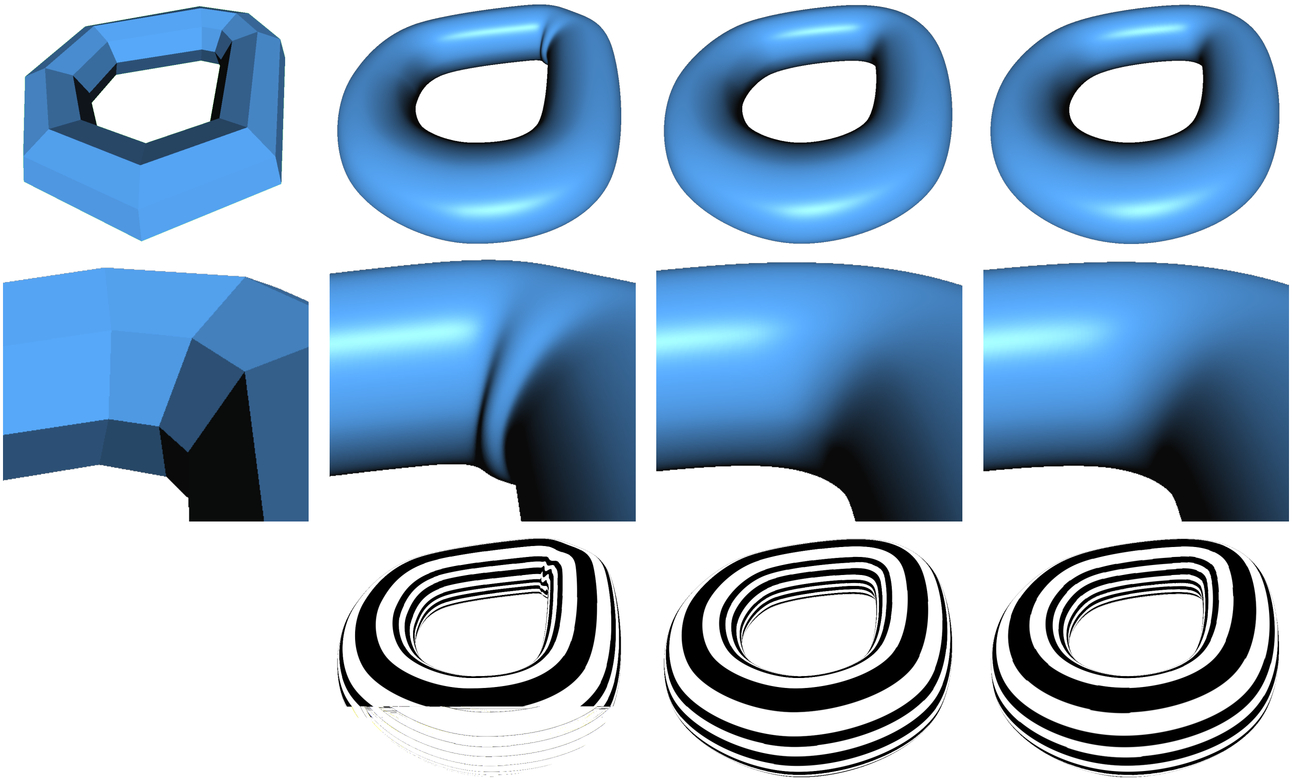
of the mesh, where the horizontal section polylines have edges of quite different lengths. On the contrary, they are not present when the augment parameterization strategy is exploited. In particular, the examples in the last two columns build upon two classes of local interpolants that have support width 4 and different orders of continuity (C^1 and C^2 respectively), thus giving rise to tangent-continuous and curvature-continuous surfaces. Figure 4.9 features a mesh that is a modification of the previous one, obtained by shortening some edges also in the other direction, so that the region becomes even more critical for the classical tensor-product approach. Yet, the augmented surfaces still behave quite well.

Figure 4.10 shows another batch of examples of composite surfaces with augmented parameterization, where classes of local spline interpolants with support width 4 and 6 are used. In the center column, the curvature of the patch boundaries is visualized by means of the so-called curvature comb. From that, in the bottom two rows we can actually see that the boundary curves are univariate curvature-continuous interpolants, and in the right-hand column the surface curvature plots confirm that these augmented surfaces are indeed G^2. As a side note, it is not surprising that the surface obtained by using interpolants with support width 6 exhibits a fairer distribution of curvature values.

In case of interpolation of arbitrary-topology meshes, we recall that the regular construction described in Section 4.4 is exploited where possible, while in the regions around extraordinary vertices the patching technique detailed in Section 4.5 is applied. Examples of this kind can be found in Figures 4.11 and 4.12, where tangent-continuous and curvature-continuous interpolating surfaces are build, respectively. In particular, for the examples in Figure 4.11, Catmull-Rom (C^1) splines are used as univariate local interpolants, and augmented bicubically blended Coons-Gregory patches as hole-filling method around each extraordinary vertex. For the example in Figure 4.12, we resort to a class of local interpolants with at least C^2 continuity and to augmented biquintically blended Coons-Gregory patches in order to achieve a higher order of smoothness. In all the cases, the resulting surface quality is good.

4.8 Meshes with non-quadrilateral faces and final remarks

The proposed approach generates high-quality interpolants from meshes with quadrilateral faces and arbitrary connectivity or analogous curve networks. Discussing the case in which the mesh includes non-quadrilateral (*extraordinary*) faces, is beyond the scope of this work. However, it is worth noting that, if a mesh includes isolated extraordinary faces, the local nature of

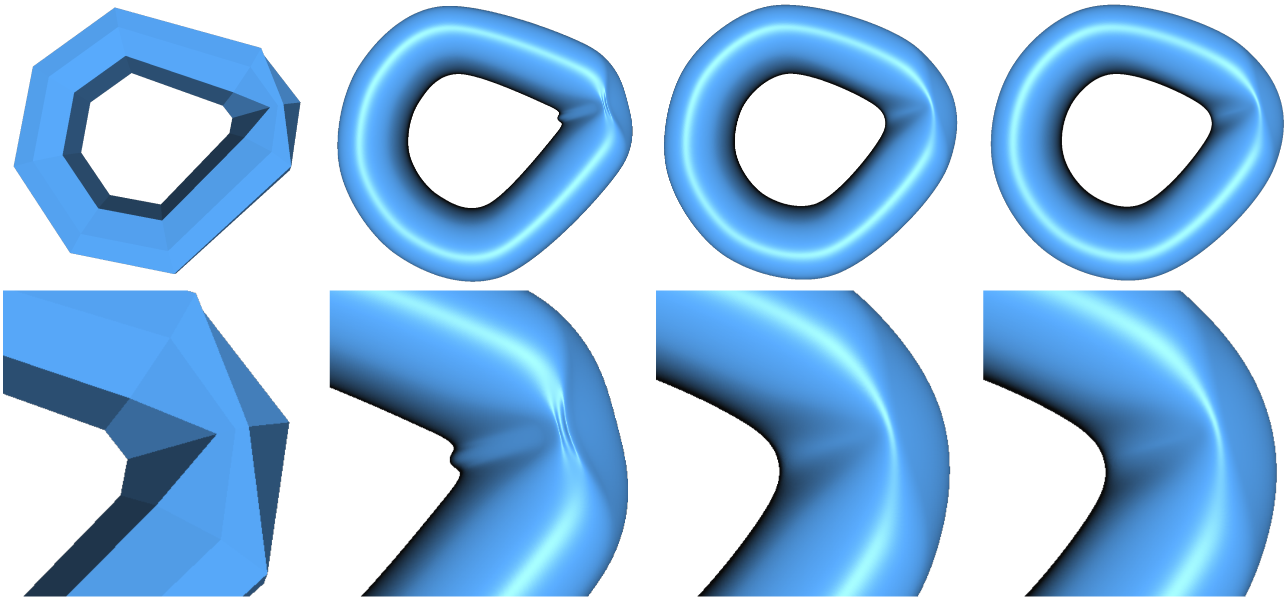


(a) Mesh. **(b)** Tensor-product surface. **(c)** Augmented G^1 surface (class $D^3C^1P^2S^4$). **(d)** Augmented G^2 surface (class $D^5C^2P^2S^4$).

Figure 4.8: Comparison of the surfaces obtained by tensor-product interpolation and our method, on a mesh with a critical region where the horizontal section polylines have edges of quite different lengths. It is evident that, in both cases of G^1 and G^2 smoothness, our construction does not exhibit any artifact. The top row shows a global view of the mesh and the surfaces, while the middle row provides a magnification of the critical region. In the bottom row, isophotes visualization highlights the interpolation artifacts that affect the tensor-product surface.

our method makes it possible to patch all the quadrilateral faces. After this operation, extraordinary faces can be handled via hole-filling techniques, extrapolating the necessary information from the adjacent patches.
In this context, the works [VRS11, VSR12] propose a method for patching n-sided holes, which takes into account the individual lengths of the sides of the hole, a reference to the concept of non-uniform parameterization that is central to our approach. Therefore we suggest that this technique could effectively complement our method, to patch possible isolated n-sided faces.

Another possible approach for handling extraordinary faces consists in refining the initial mesh, either globally or adaptively, through an interpolatory subdivision scheme, so as to split each n-sided face into n new quadrilateral faces. In this regard, as already mentioned, interpolating subdivision schemes do not admit an analytical representation of the limit surface or an exact evaluation algorithm. However, recently much research has been devoted to the study of non-uniform subdivision schemes derived from spline interpolants [BCR11a, BCR11b, KP13] and their bivariate generalization [BCR13b]. By applying our patching approach starting from the same classes of splines related to a subdivision scheme, we could therefore approximate the limit surface, so as to provide a parametric representation. This work lays the foundation for the study of this relevant topic, which is among our future research objectives.



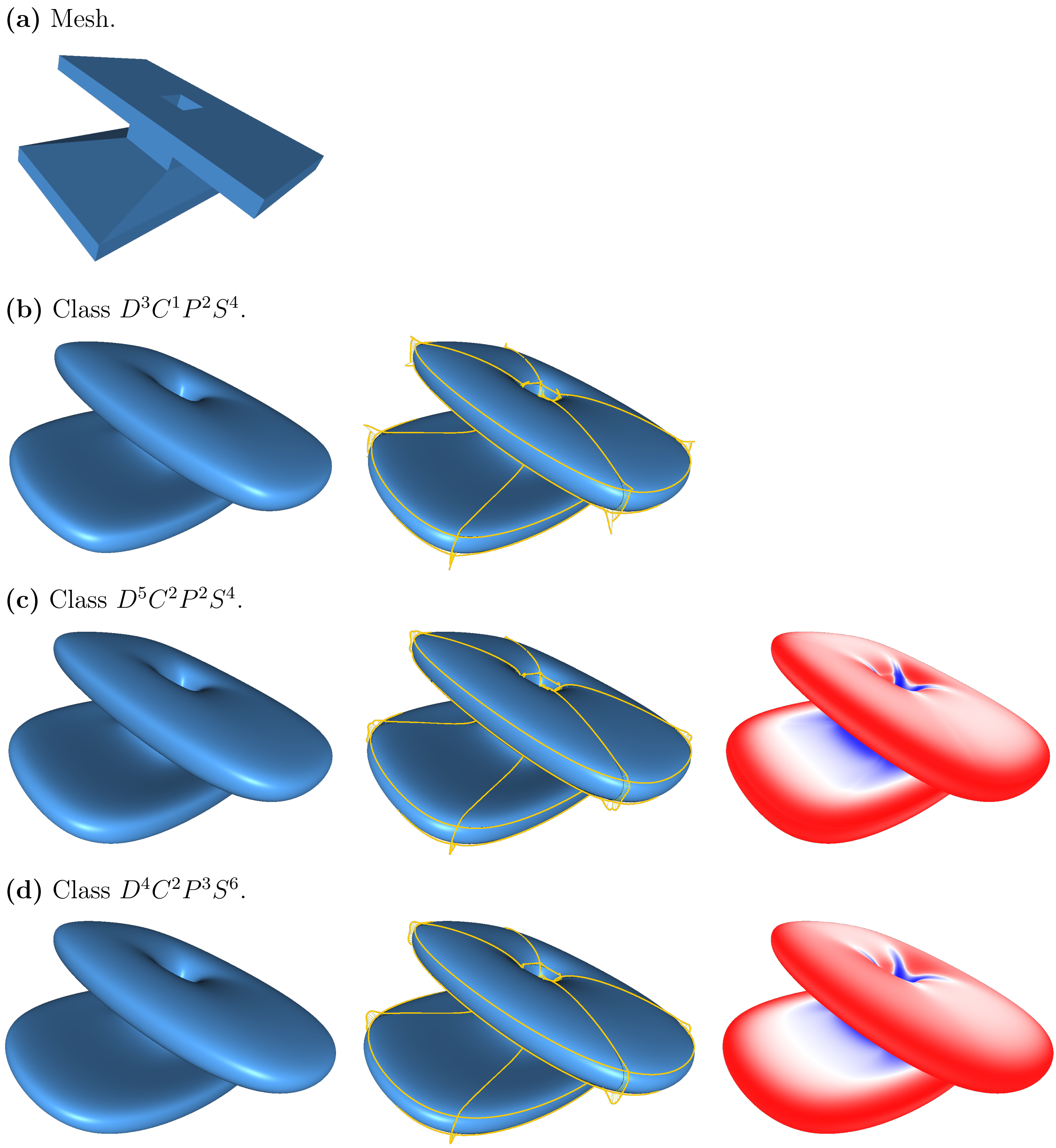
(a) Mesh.

(b) Tensor-product surface.

(c) Augmented G^1 surface (class $D^3C^1P^2S^4$).

(d) Augmented G^2 surface (class $D^5C^2P^2S^4$).

Figure 4.9: Additional comparison of the surfaces obtained by tensor-product interpolation and our method, on a mesh with a more critical configuration of vertices. Also in this case our construction produces surfaces with a better shape.



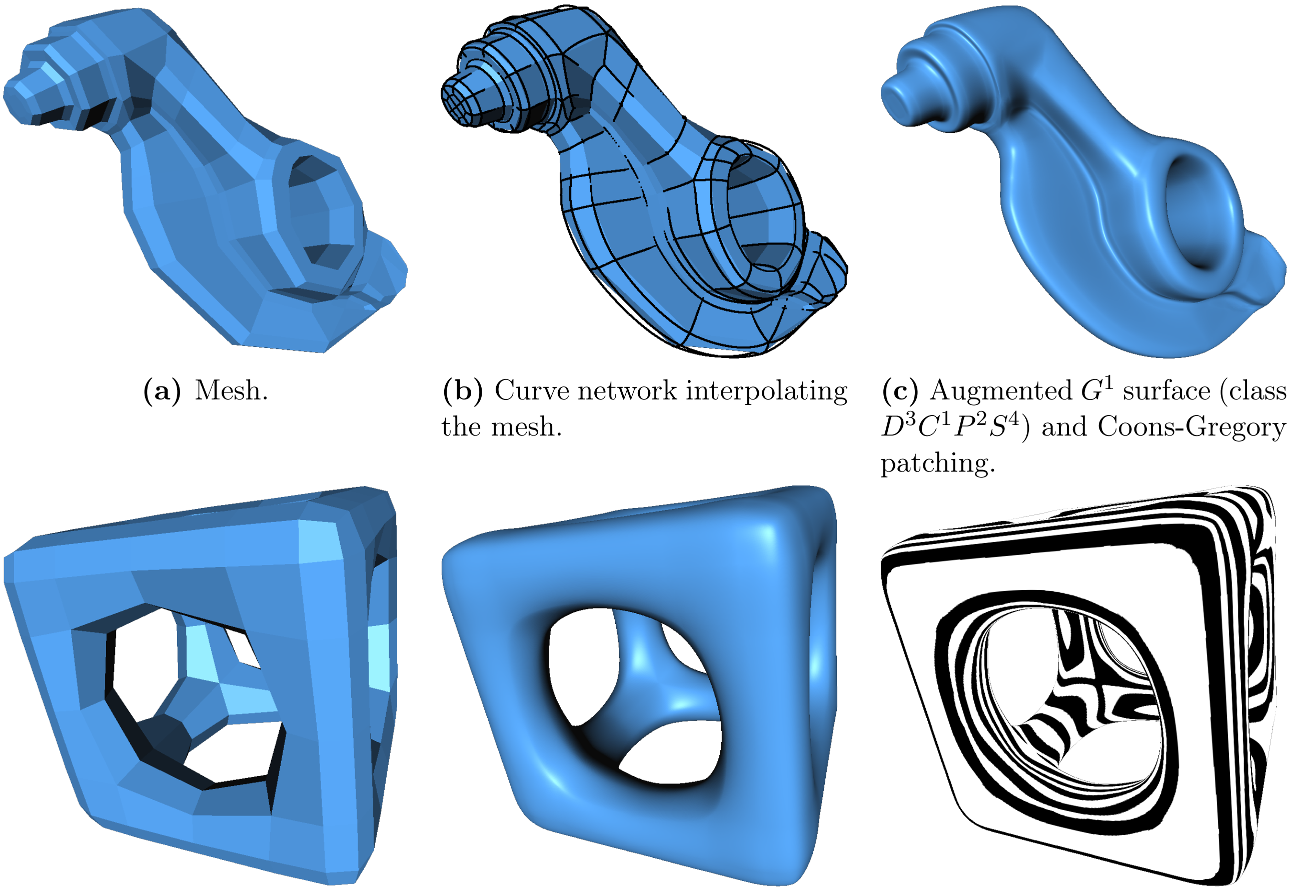
(a) Mesh.

(b) Class $D^3C^1P^2S^4$.

(c) Class $D^5C^2P^2S^4$.

(d) Class $D^4C^2P^3S^6$.

Figure 4.10: Surfaces with augmented parameterization obtained by using classes of local spline interpolants with order of continuity 1 and 2, and support width 4 and 6. Center column: comb graph of the curvature of the patch boundary curves; right-hand column: visualization of mean curvature values.



(a) Mesh.

(b) Curve network interpolating the mesh.

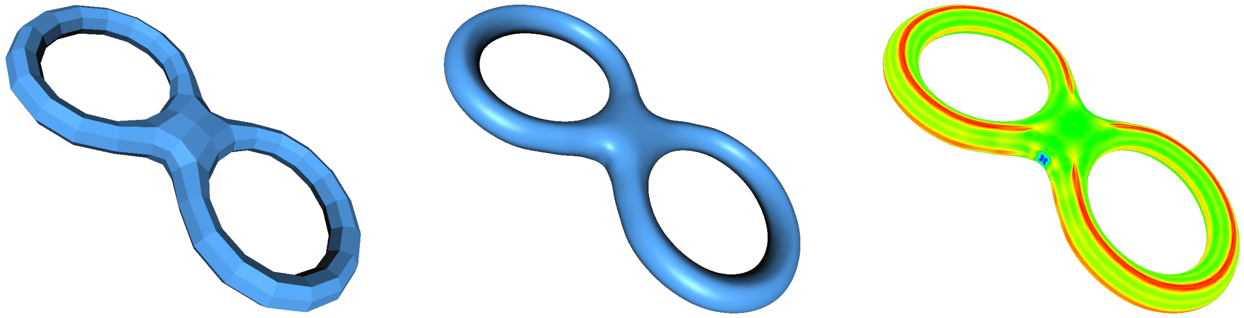
(c) Augmented G^1 surface (class $D^3C^1P^2S^4$) and Coons-Gregory patching.

(d) Mesh.

(e) Augmented G^1 surface (class $D^3C^1P^2S^4$) and Coons-Gregory patching.

(f) Isophotes.

Figure 4.11: Interpolation of meshes with arbitrary topology representing a rocker arm and a hollow cube. In both cases, Catmull-Rom splines are used as local interpolants, and the augmented bicubically blended Coons-Gregory patches around extraordinary vertices provides overall G^1 surfaces with augmented parameterization.



(a) Mesh.

(b) Augmented G^2 surface (class $D^5C^2P^2S^4$) and Coons-Gregory patching.

(c) Mean curvature.

Figure 4.12: Interpolation of a mesh with arbitrary topology representing a double torus. Local interpolating splines belonging to the class $D^5C^2P^2S^4$ are used, and the augmented biquintically blended Coons-Gregory patches around extraordinary vertices provides an overall G^2 surface with augmented parameterization.

Chapter 5

A constructive approach to generalized splines

Quasi Extended Chebyshev (QEC) spaces containing constants have been recently acknowledged to be the largest class of spaces suitable for design [Maz08]. In this chapter we consider spline spaces whose sections may belong to any of these QEC-spaces, including the case in which different sections may belong to QEC-spaces of different type (i.e., generated by functions of polynomial, trigonometric, hyperbolic, or mixed type), and where adjacent spline pieces are connected via parametric continuity. We propose a general approach to construct a basis for all such spline spaces which is based on the notion of transition function (already recalled in previous chapters). If a spline space admits an optimal normalized totally positive (ONTP) basis, then this is precisely the basis yield by our construction. Moreover, we show that a system of weight functions can be computed in terms of transition functions and, investigating the connection between the existence of weight functions and the existence of the ONTP basis, we provide a criterion for determining whether an arbitrary generalized spline space admits the ONTP basis.

In addition, exploiting the notion of transition function, which is the key tool of the aforementioned constructive approach, we formulate a set of algorithms for efficient computation in spaces of generalized splines. We also show how the basic operations of knot insertion and order elevation can be conveniently performed and expressed in terms of the transition functions. Finally, we provide some remarkable examples to illustrate the use of generalized splines in applications, with particular focus on geometric modeling.

The content of this chapter is also the subject of two recently submitted papers [ABCR14b, ABCR14a].

5.1 Introduction

5.1.1 Current interests and recent trends

Generalized splines are a class of piecewisely defined functions that generalize the well-established piecewise polynomial splines in order to contain not only polynomial sections, but also sections of trigonometric, hyperbolic, or exponential type, as well as of more outlandish types. Generalized splines are conveniently applied in modeling free-form curves and surfaces since they both inherit all the desirable properties of polynomial splines and they allow for representing much more various shapes and reproducing fundamental geometric primitives, such as conic sections and some remarkable transcendental curves. Moreover, generalized splines are essential in obtaining shape-preserving approximations, since they offer the possibility of modifying the tension of a curve in order to follow the shape of the underlying data as close as possible. Finally, in recent years, the potential of generalized splines has become apparent also in the context of isogeometric analysis. This is due to the fact that, on the one side, they possess the capability of exactly representing conic sections, like NURBS do, and, on the other side, they not only fulfill the appealing features of polynomial B-splines such as knot insertion and order elevation, but also behave in the same way with respect to integration and differentiation.

5.1.2 Historical perspectives and state of the art

B-spline bases for generalized spline spaces were first constructed by Schoenberg in [Sch64], where a space of trigonometric splines was considered. His work motivated the subsequent study of the more general *Chebyshevian splines* and of their local bases conducted by Karlin [Kar68], Jerome and Schumaker [JS71, JS76], Lyche, Winter and Koch [LW79, KL89] in the Seventies and Eighties. Schumaker showed that a generalized spline space, where each piece is spanned by an Extended Complete Chebyshev system of functions, admits a locally supported normalized basis that enjoys the property of total positivity. As a consequence, the resulting generalized spline curves can be represented as a linear combination of B-spline basis functions and the variation diminishing property holds. There follows that the shape of a parametrically defined curve mimics the shape of its control polygon and such a B-spline basis can be considered suitable for design purposes.

An important extension of Chebyshevian splines, *piecewise Chebyshevian splines*, can be obtained by considering spline spaces having sections in different Extended Chebyshev spaces (EC-spaces for short). In this case, the continuity conditions between adjacent spline pieces are expressed in terms of connection matrices, linking the appropriate number of left and right generalized derivatives. Depending on the choice of the generalized derivatives and on the form of the connection matrices, piecewise Chebyshevian splines can be either geometrically or parametrically continuous. In the latter case, generalized derivatives are simply ordinary derivatives and the connection matrices are the identity matrix. The works by Barry [Bar96] and Buchwald and Mühlbach [BM03] showed that also for these spaces it is possible to construct the B-spline basis respectively when geometric and parametric continuity conditions are used at the knots.

In the last two decades, Carnicer and Peña [CP94] proved that, among all normalized totally positive (NTP) bases of a generalized space, there exists a unique optimal normalized totally positive (ONTP) one, which they called *B-basis*, which is the one associated with the control polygon that resembles with the highest fidelity the shape of the curve, among all the control polygons of the same curve corresponding to NTP bases. A space that has an ONTP basis can be considered suitable for design purposes, because any space obtained from it by knot insertion has an ONTP basis as well. In the case of polynomials and polynomial splines, the well-known Bernstein and B-spline bases are both ONTP bases. However, in the context of generalized spline spaces, there may exist bases that enjoy some properties commonly ascribed to the B-spline basis but that in fact are not ONTP bases. Therefore, in this chapter we prefer clearly distinguishing the two situations, and thus we will lean on two different definitions for (generalized) B-spline bases and ONTP bases. More precisely, since an ONTP basis is trivially a B-spline basis, henceforth by ONTP basis we mean a B-spline basis which is also ONTP.

At the state of the art, much work has been devoted to the study of necessary and sufficient conditions to guarantee that an EC-space admits the Bernstein basis [CMP03, BM12] and therefore can be considered suitable for the design purposes. In particular, in this context the notion of *critical length for design* has been introduced to characterize the largest possible interval on which the existence of the Bernstein basis for the considered space is guaranteed. For spline spaces, similar conditions are hardly derivable and indeed have been established only for particular families of spaces [MP10, BM14]. These results in the spline context are derived from the well-known theory of *systems of (piecewise) weight functions* and corresponding generalized derivatives.

Besides these studies, another avenue of research [Cos00, CM06, KS99] showed the existence of other interesting generalized spline spaces whose segments do not belong to EC-spaces and that result computationally more efficient in handling shape-preserving approximations.

To include also these families of splines in a comprehensive framework, the class of spaces of sufficiently regular functions which can be considered suitable for design shall be extended to a superset of EC-spaces, called Quasi Extended Chebyshev spaces (QEC-spaces for short), consisting of all spaces that permit Hermite interpolation, Taylor interpolation excepted. In particular, Mazure [Maz08] identified the largest class of spaces suitable for design as the class of all spaces containing constants and such that the spaces obtained by differentiation are QEC-spaces. By means of the blossoming formalism, she proved that this is the class of all spaces that always admit the Bernstein basis.

This novelty has attracted interest in the study of *quasi Chebyshevian splines* and *piecewise quasi Chebyshevian splines*. The former name refers to splines having all sections in the same QEC-space; the latter one to splines whose sections belong to QEC-spaces that may differ for the number and/or for the type of their generators.

In analogy to the aforementioned characterization of QEC-spaces, we can say that any piecewise quasi Chebyshevian spline space is suitable for design when it contains constants and has the ONTP basis.

In particular, it is also significant to consider spaces featured by non-uniform knot partitions, including multiple knots, and spline spaces where different segments may belong to different QEC-spaces containing shape parameters.

In this regard, a recent work by Mazure [Maz11b] has characterized the existence of the ONTP basis for spline spaces where each segment is spanned by a different QEC-system of the type $\{1, x, \ldots, x^{m-3}, u_i(x), v_i(x)\}$. In that paper, for the special case of QEC-spaces where the functions $u_i(x)$ and $v_i(x)$ are of polynomial, trigonometric or hyperbolic type, explicit necessary and sufficient conditions for the existence of the ONTP basis are provided in terms of the maximum allowable width of a certain number of consecutive knot intervals.

If we now turn to consider the constructive aspects of the ONTP basis for spaces of parametrically continuous piecewise (quasi) Chebyshevian splines, the first available approach of general kind is based on a suitable generalization of the notion of divided differences [BM03, Müh06]. A computationally more efficient strategy, which is adopted in many recent papers [WF08, MP10, SW10, Maz11a], is based on a generalization of the well-known integral recurrence formula used to compute the B-spline basis of polynomial spline spaces [BP97]. The method starts from a given ONTP basis, which must therefore be known a priori, and by means of successive integrations it allows for generating an analogous basis for a space that contains higher-order polynomials. This approach can be generalized by introducing sequences of (non-constant) weight functions and exploiting them in the integral procedure, in order to obtain spaces that contain functions of types other than polynomial. However, this way is hardly feasible in practice, because it requires that a proper sequence of weight functions be available so that the integral recurrence formula produces the ONTP basis of the desired space. Besides the aforementioned more general constructions, there are also methods tailored to specific classes of spaces, such as the geometric approach proposed in [Cos00, CM06] for generalized cubic splines from QEC-spaces of the form $\{1, x, u_i(x), v_i(x)\}$, or the construction of the so-called GB-splines

[KS99].

5.1.3 Motivations and contributions of this work

As previously mentioned, all spaces useful for design can be characterized as QEC-spaces containing constants and whose derivative space is also a QEC-space. Therefore, spaces of generalized splines with sections belonging to any of these QEC-spaces are of great interest for design provided that they admit an ONTP basis. Despite the wealth of proposals for constructing ONTP bases for spaces of generalized polynomials (i.e., Bernstein bases) the problem of deriving analogous bases for spline spaces has been addressed only in particular cases, namely for specific classes of spaces and/or uniform knot partitions.

The main result of the present work consists in providing a general and effective approach to construct the ONTP basis for any arbitrary space of parametrically continuous piecewise quasi Chebyshevian splines, without limitations on the type of the underlying QEC-spaces or on the location of knots. Moreover, we show that the same construction yields a set of weight functions that can be conveniently exploited to assess the existence of the ONTP basis, if this is not known a priori.

The proposed construction is based on a generalization of the notion of *transition function*, which has long been known in the classical literature, even if never used in the context of the present work.

Oversimplifying, a transition function is a piecewise function having proper continuity and that takes respectively the values 0 and 1 at the endpoints of a certain real interval. To the best of our knowledge, this terminology was introduced by Schumaker [Sch07, p. 141], although a similar concept was already used by Bézier, who, precisely in terms of these types of functions, described the curves that now bear his name (see [Rab02] or the more recent monograph on Bernstein polynomials [Far12]). As recalled in Chapter 3, transition functions have been recently used for the construction of local polynomial interpolants from unevenly spaced interpolation nodes [BCR13a, ABC14a].

Given a generalized spline space, we show that there exists a unique set of transition functions forming a basis for the space and that each of these functions can be simply determined as the solution of a suitable system of linear equations. We also show that if a space admits an ONTP basis, then the elements of this basis are nothing but a linear combination of the transition functions associated to the space. Moreover, we illustrate how to construct a set of piecewise weight functions, which can be simply formalized in terms of transition functions. Investigating the relationship between such weight functions and the existence of the ONTP basis, we are able to provide a criterion for determining whether a given spline space admits the ONTP basis, and we show how this result can be translated into a general methodology for the verification of the existence of such a basis.

For its generality and its conceptual simplicity, the approach has both a theoretical value and a direct computational interest. The construction proposed for the transition functions applies with full generality to spline spaces in which the QEC-spaces on each interval may be different and contain many functions of non-polynomial type and shape parameters, and to the case where the knot partition is non-uniform and includes multiple knots.

Moreover, since we do not construct the ONTP basis by means of an integral recurrence formula, the presented method does not require the knowledge of a system of weight functions and a basis to be used as a starting point for the integral formulation. In addition, the proposed approach readily translates into numerical procedures, which is the inevitable way of proceeding in the very general setting that allows non-uniform knot partitions and the presence of spaces that differ in type and parameters. On the contrary, since in the literature only particular situations have been addressed so far, there is the reasonable doubt that the spline bases determined in such cases have been computed symbolically and thus do not straightforwardly lend themselves to more general settings.

Besides illustrating the theoretical aspects of the framework, namely those related to the construction and study of the existence of the B-spline basis, as of now we would like to emphasize that the intention of this work is also to provide all the tools needed to improve computation in generalized spline spaces. In fact, the transition functions turn out to be extremely convenient to formulate computational algorithms as well as to perform knot insertion, order elevation and, more generally, any other essential operation in the contexts of modeling and design.

Hereinafter it is assumed that different spline segments are spanned by QEC-spaces having the same dimension and join with parametric continuity. However, we would like to point out that these restrictions are merely dictated by the attempt to maintain the theoretical framework as streamlined as possible. A slight generalization of the approach allows for handling the cases where the QEC-spaces may differ in dimension and geometric continuity conditions are required at the knots.

The remainder of the chapter is organized as follows. After recalling some preliminary notions in Section 5.2, in Section 5.3 we introduce the definition of transition function and we devise a general method to construct a basis for any arbitrary space of piecewise quasi Chebyshevian splines. Successively, in Section 5.4 we exploit the transition functions to derive a criterion for establishing whether a given space possesses the ONTP basis. In particular, this criterion translates into a numerical test, as detailed in Section 5.4.2. Moreover, Section 5.4.3 briefly focuses on the particular case of an empty knot partition, in which the ONTP basis is the Bernstein basis. Additionally, some numerical examples on the existence of the ONTP basis are presented in Section 5.5. Section 5.6 deals with computational and modeling aspects that may be of interest from the point of view of applications. In fact, knot insertion and order elevation algorithms for generalized splines are provided in Sections 5.6.1 and 5.6.2 respectively, and the explicit expressions of the ONTP bases for some spaces and stable corner cutting evaluation algorithms are collected in Section 5.6.3. Section 5.6.4 discusses some connections between knot insertion and subdivision schemes in generalized spline spaces. Then, in Section 5.6.5 we summarize in an algorithmic form the main computational tools that can be drawn from the theoretical results in the preceding sections, and finally we illustrate some application examples in Section 5.6.6.

5.2 QEC-spaces and piecewise Chebyshevian spline spaces

Before delving into the details of our construction, we outline some basic notions on QEC-spaces and generalized spline spaces.

5.2.1 Preliminary notions on QEC-spaces

Let us recall that an m-dimensional function space $\mathcal{U}_m$ defined on a closed interval $I \subset \mathbb{R}$ is a subspace of $C^k(I)$ if every function in $\mathcal{U}_m$ is k-times continuously differentiable. Then we can introduce the following definitions.

Definition 5.1 (Chebyshev space)**.** An m-dimensional space $\mathcal{U}_m$ contained in $C^0(I)$ is a *Chebyshev space* (C-space for short) on I if any Lagrange interpolation problem in m data has a unique solution in $\mathcal{U}_m$. In terms of zeros, this means that any nonzero element of $\mathcal{U}_m$ vanishes at most $m-1$ times in I, not including possible multiplicities.

Definition 5.2 (Extended Chebyshev space)**.** An m-dimensional space $\mathcal{U}_m$ contained in $C^{m-1}(I)$ is an *Extended Chebyshev space* (EC-space for short) on I if any nonzero element of $\mathcal{U}_m$ vanishes at most $m-1$ times in I, counting multiplicities as far as possible for C^{m-1} functions (that is, up to m), or, equivalently, if any Hermite interpolation problem in m data in I has a unique solution in $\mathcal{U}_m$.

A superset of EC-spaces, originally introduced by Mazure [Maz08, Maz11c], is defined as follows.

Definition 5.3 (Quasi Extended Chebyshev space)**.** An m-dimensional space $\mathcal{U}_m$ contained in $C^{m-2}(I)$, $m \geqslant 2$, is a *Quasi Extended Chebyshev space* (QEC-space for short) on I if any Hermite interpolation problem in m data in I, with at least two distinct points, has a unique solution in $\mathcal{U}_m$. Equivalently, for $m > 2$, $\mathcal{U}_m$ is a QEC-space if any nonzero element of $\mathcal{U}_m$ with at least two distinct zeros vanishes at most $m-1$ times in I counting multiplicities.

From the two above definitions it follows that QEC-spaces differ from EC-spaces in that they do not permit Taylor interpolation. Moreover, if $\mathcal{U}_m$ is a QEC-space on I, then it is a QEC-space on any subinterval $[a,b] \subseteq I$, with $a < b$. Hereinafter we refer to the basis functions spanning a QEC-space as a *QEC-system.* It is well known that spaces obtained by integrating the functions of a QEC-system or by multiplying them by positive functions are in turn QEC-spaces. In contrast, there is no guarantee that by differentiating a QEC-system we will get a QEC-space. However, we recall from [Maz11c, third item of Remark 2.1] that a 2-dimensional space $\mathcal{U}_2 \subset C^0(I)$ is a QEC-space on I if and only if it is a C-space on I, or, equivalently, if and only if for any subinterval $[a,b] \subseteq I$, with $a < b$, $\mathcal{U}_2$ possesses a basis of Lagrange type.

Definition 5.4 (Bernstein basis)**.** Let $I \subset \mathbb{R}$ be a closed bounded interval. Given $a, b \in I$, with $a < b$, and $B_{0,m}, \ldots, B_{m-1,m} \in C^{m-2}(I)$, we say that $\{B_{0,m}, \ldots, B_{m-1,m}\}$ is a *Bernstein basis* relative to $[a,b]$ if it meets the following requirements:

i) *zero property*:

  - $B_{0,m}, \ldots, B_{m-1,m}$ are nonvanishing in (a,b);
  - $B_{0,m}(a) \neq 0$ and $B_{0,m}$ vanishes $m-1$ times at b;
  - $B_{m-1,m}(b) \neq 0$ and $B_{m-1,m}$ vanishes $m-1$ times at a;
  - for $1 \leqslant i \leqslant m-2$, $B_{i,m}$ vanishes exactly i times at a and exactly $m-1-i$ times at b;

ii) *positivity property*: for $0 \leqslant i \leqslant m-1$, $B_{i,m}$ is positive in (a,b);

iii) *normalization property*: $\{B_{0,m}, \ldots, B_{m-1,m}\}$ is normalized, i.e., $\sum_{i=0}^{m-1} B_{i,m}(x) = 1$, $\forall x \in [a,b]$.

Definition 5.5 (Bernstein-like basis)**.** If requirement iii) in Definition 5.4 is not fulfilled, then $\{B_{0,m}, \ldots, B_{m-1,m}\}$ is called a *Bernstein-like basis.*

Remark 5.1. In the particular case in which the space $\mathcal{U}_m$ is an EC-space, then requirement i) in Definition 5.4 can be replaced by:

i') *zero property*: for $0 \leqslant i \leqslant m-1$, $B_{i,m}$ vanishes exactly i times at a and exactly $m-1-i$ times at b, and nowhere else in (a,b).

In this work we are interested in spaces of piecewise quasi Chebyshevian splines where each section belongs to an m-dimensional QEC-space suitable for design and where adjacent spline pieces are connected via the standard parametric continuity. For such spaces, a basis will be derived in Section 5.3.

5.2.2 General notions on splines

Let $[a,b]$ be a bounded and closed interval, and $\boldsymbol{\Delta} := \{x_i, i = 1,\dots,k\}$ a set of points such that $a \equiv x_0 < x_1 < \dots < x_k < x_{k+1} \equiv b$. Let us consider the partition of $[a,b]$ induced by the set $\boldsymbol{\Delta}$ into the subintervals $I_i := [x_i, x_{i+1})$, $i = 0,\dots,k-1$, and $I_k := [x_k, x_{k+1}]$.
Moreover, let m be a positive integer, and $\boldsymbol{M} := (m_1,\dots,m_k)$ a vector of positive integers such that $1 \leqslant m_i \leqslant m$ for every $i = 1,\dots,k$. We denote by $\boldsymbol{\mathcal{U}}_m := \{\mathcal{U}_{0,m},\dots,\mathcal{U}_{k,m}\}$ an ordered set of spaces of dimension m such that every $\mathcal{U}_{i,m}$ is a QEC-space on the interval I_i for $i = 0,\dots,k$.

Definition 5.6 (Generalized splines)**.** We define the set of *generalized splines* of order m with knots $x_1,\dots,x_k$ of multiplicities $m_1,\dots,m_k$ as

$$\begin{aligned} S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) := \big\{ s \,\big|\, & \text{there exist } s_i \in \mathcal{U}_{i,m}, i = 0,\dots,k, \text{ such that:} \\ & \text{i) } s(x) = s_i(x) \text{ for } x \in I_i, i = 0,\dots,k; \\ & \text{ii) } D^r s_{i-1}(x_i) = D^r s_i(x_i) \text{ for } r = 0,\dots,m-m_i-1,\ i = 1,\dots,k \,\big\}. \end{aligned}$$

For conciseness, in the following we only prove results that cannot be found as a generalization of classical spline theory. As for the others, which can be obtained by following the same outline of their classical counterpart, we confine ourselves to providing a convenient reference.

Proposition 5.1. *The set $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ of generalized splines with multiple knots is a function space of dimension $m + K$ with*

$$K := \sum_{i=1}^{k} m_i.$$

Proof. The assertion can be proved following the same outline as for [Sch07, Theorem 4.4]. In particular, for a spline in $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$, it can be observed that the block diagonal matrix that represents the continuity conditions at the knots has full rank, because each $\mathcal{U}_{i,m}$ is a QEC-space on I_i. □

Definition 5.7 (Extended partition)**.** The set $\boldsymbol{\Delta}^* := \{t_i, i = 1,\dots,2m+K\}$, with $K = \sum_{i=1}^{k} m_i$, is called an *extended partition* associated with $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ if and only if:

i) $t_1 \leqslant t_2 \leqslant \dots \leqslant t_{2m+K}$;

ii) $t_m \equiv a$ and $t_{m+K+1} \equiv b$;

iii) $\{t_{m+1},\dots,t_{m+K}\} \equiv \{\underbrace{x_1,\dots,x_1}_{m_1 \text{ times}},\dots,\underbrace{x_k,\dots,x_k}_{m_k \text{ times}}\}$.

Let us denote by x_i, with $i \leqslant 0$ and $i \geqslant k+1$, the distinct external knots, and by m_i their multiplicities, i.e.

$$\{\dots,\underbrace{x_{-1},\dots,x_{-1}}_{m_{-1} \text{ times}},\underbrace{x_0,\dots,x_0}_{m_0 \text{ times}}\} \equiv \{t_1,\dots,t_m\},$$

$$\{\underbrace{x_{k+1},\dots,x_{k+1}}_{m_{k+1} \text{ times}},\underbrace{x_{k+2},\dots,x_{k+2}}_{m_{k+2} \text{ times}},\dots\} \equiv \{t_{m+K+1},\dots,t_{2m+K}\}.$$

For every external interval $[x_i, x_{i+1}]$, with $i \leqslant -1$ and $i \geqslant k+1$, let us assign an additional QEC-space $\mathcal{U}_{i,m}$ of dimension m.

Definition 5.8 (B-spline basis)**.** A sequence $\{N_{i,m}, i = 1, \ldots, m+K\}$ of elements of $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ is said to be a *B-spline basis* for $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ if it meets the following requirements:

i) *support property*: for each $i \in \{1, \ldots, m+K\}$, $N_{i,m}(x) = 0$ for $x \notin [t_i, t_{i+m}]$;

ii) *positivity property*: for each $i \in \{1, \ldots, m+K\}$, $N_{i,m}(x) > 0$ for $x \in (t_i, t_{i+m})$;

iii) *endpoint property*: for each $i \in \{1, \ldots, m+K\}$,

$$D_+^r N_{i,m}(t_i) = 0 \;\text{ for }\; 0 \leqslant r \leqslant m-2-m_i^{\mathrm{R}}, \;\text{ and }\; D_+^{m-1-m_i^{\mathrm{R}}} N_{i,m}(t_i) \neq 0,$$
$$D_-^r N_{i,m}(t_{i+m}) = 0 \;\text{ for }\; 0 \leqslant r \leqslant m-2-m_{i+m}^{\mathrm{L}}, \;\text{ and }\; D_-^{m-1-m_{i+m}^{\mathrm{L}}} N_{i,m}(t_{i+m}) \neq 0,$$

where $m_i^{\mathrm{L}} := \max\{p \geqslant 0 \text{ such that } t_{i-p} = t_i\}$ and $m_i^{\mathrm{R}} := \max\{p \geqslant 0 \text{ such that } t_{i+p} = t_i\}$;

iv) *normalization property*: $\sum_i N_{i,m}(x) = 1$, $\forall x \in [a, b]$.

Definition 5.9. If requirement iv) in Definition 5.8 is not fulfilled, then $\{N_{i,m}, i = 1, \ldots, m+K\}$ is called a *B-spline-like basis* for $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$.

The following result on ONTP bases as been proved by Mazure (see, e.g., [Maz12, Theorem 2.4]).

Proposition 5.2. *We say that a spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ has the ONTP basis when it has a B-spline basis and so does any space obtained from $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ by knot insertion.*

5.3 Basis functions for generalized spline spaces

In this section we illustrate a general method, based on the concept of transition function, to construct a basis for any arbitrary space of piecewise quasi Chebyshevian spline functions. The proposed approach applies to spaces with non-uniform knot partition, possibly multiple knots, and where each spline piece may belong to a different m-dimensional QEC-space. In Section 5.4 we will prove that, if the generalized spline space admits the ONTP basis, then it is precisely the basis yield by the proposed approach.

Working in the generalized spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$, the following definition restates in a more general way the notion of transition function already introduced in Chapter 3.

Definition 5.10 (Transition functions)**.** Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a generalized spline space of dimension $m+K$ with $\boldsymbol{\Delta} = \{x_i, i = 1, \ldots, k\}$ a partition of $[a, b]$ and $\boldsymbol{\Delta}^* = \{t_i, i = 1, \ldots, 2m+K\}$ the associated extended partition, such that each QEC-space $\mathcal{U}_{i,m}$ of $\boldsymbol{\mathcal{U}}_m$ contains constants. We call *transition functions* the piecewise functions $f_{i,m}$, $i = 1, \ldots, m+K$, having the following properties:

a)

$$f_{i,m}(x) := \begin{cases} 0, & x < t_i, \\ u(x), & x \geqslant t_{i+m-1}, \end{cases}$$

where $u(x)$ is a positive function belonging to $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$;

b) if $t_i < t_{i+m-1}$, denoted by $\{x_{i_j}, j = 1, \ldots, p\}$ the knots of $\boldsymbol{\Delta}$ such that

$$\{\underbrace{x_{i_1}, \ldots, x_{i_1}}_{m_{i,i_1} \text{ times}}, \ldots, \underbrace{x_{i_p}, \ldots, x_{i_p}}_{m_{i,i_p} \text{ times}}\} \equiv \{t_i, \ldots, t_{i+m-1}\}$$

and by $m_{i,i_1}, \ldots, m_{i,i_p}$ their multiplicities, it holds

$$\begin{aligned}
&D^r f_{i,m}^{[1]}(x_{i_1}) = 0, && r = 0, \ldots, k_{i,i_1}, \\
&D^r f_{i,m}^{[j-1]}(x_{i_j}) = D^r f_{i,m}^{[j]}(x_{i_j}), && r = 0, \ldots, k_{i,i_j}, \quad j = 2, \ldots, p-1, \\
&D^r f_{i,m}^{[p-1]}(x_{i_p}) = D^r u(x_{i_p}), && r = 0, \ldots, k_{i,i_p},
\end{aligned} \tag{5.1}$$

where $k_{i,i_j} := m - m_{i,i_j} - 1$ and $f_{i,m}^{[j]}$, $j = 1, \ldots, p-1$, are the different pieces of $f_{i,m}$ on the intervals $[x_{i_j}, x_{i_{j+1}})$.

Remark 5.2. It is easy to check that, when restricted to $[a, b]$, the transition functions $\{f_{i,m}\}$ in Definition 5.10 are piecewise functions belonging to $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$.

For any spline space identified by Definition 5.6, the following proposition guarantees that there always exists a unique set of transition functions satisfying the properties in Definition 5.10. The proof is constructive and also illustrates how the transition functions can be simply computed as the solution of a proper system of linear equations.

Proposition 5.3 (Existence and uniqueness of the transition functions)**.** *The transition functions of Definition 5.10 exist and are uniquely determined.*

Proof. If we express each piece $f_{i,m}^{[j]}$, $j = 1, \ldots, p-1$, on I_{i_j} in the associated QEC-system

$$\{u_{i_j,1}, u_{i_j,2}, \ldots, u_{i_j,m}\}$$

with coefficients $b_{i,j}$, $j = 1, \ldots, m(p-1)$, then the linear system of equations arising from the continuity conditions (5.1) can be written in the form

$$A\boldsymbol{b} = \boldsymbol{c}, \tag{5.2}$$

with

$$A := \begin{pmatrix}
A_1(x_{i_1}) & & & & \\
A_1(x_{i_2}) & -A_2(x_{i_2}) & & & \\
 & A_2(x_{i_3}) & -A_3(x_{i_3}) & & \\
 & & \ddots & \ddots & \\
 & & & A_{p-2}(x_{i_{p-1}}) & -A_{p-1}(x_{i_{p-1}}) \\
 & & & & A_{p-1}(x_{i_p})
\end{pmatrix},$$

$$\boldsymbol{b} := (b_{i,1}, \ldots, b_{i,m(p-1)})^T,$$
$$\boldsymbol{c} := (0, \ldots, 0, 1, \underbrace{0, \ldots, 0}_{k_{i,i_p} \text{ times}})^T,$$

where $A_j(x_{i_h})$, $h = j, j+1$, is a matrix of dimensions $(k_{i,i_h} + 1) \times m$ whose rth row, $r = 1, \ldots, k_{i,i_h} + 1$, has the form

$$\left(D^{r-1} u_{i_h,1}(x)\big|_{x=x_{i_h}}, \ldots, D^{r-1} u_{i_h,m}(x)\big|_{x=x_{i_h}}\right).$$

Therefore, the matrix $A_j(x_{i_h})$ has full rank, and precisely equal to $k_{i,i_h}+1$. In fact, since $k_{i,i_h}+1<m$, the matrix $A_j(x_{i_h})$ is a submatrix of the $m \times m$ Hermite interpolation matrix relative to the two points x_{i_j} and $x_{i_{j+1}}$, which is always nonsingular in a QEC-space. From this formulation, we can infer that the number of degrees of freedom for the definition of the pieces $f_{i,m}^{[j]}$, that is $m(p-1)$, equals the number of the required continuity conditions given by $\sum_{j=1}^{p}(k_{i,i_j}+1)$. In fact, recalling that by construction $\sum_{j=1}^{p} m_{i,i_j}=m$, we have $\sum_{j=1}^{p}(k_{i,i_j}+1)=\sum_{j=1}^{p}(m-m_{i,i_j})=mp-m=m(p-1)$. Hence, A is square and nonsingular and thus the transition functions $\{f_{i,m}\}$ are uniquely determined. □

Exploiting the properties of QEC-spaces, it is also easy to show that a spline in $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ cannot be nonzero on $m-1$ (or less) consecutive knot intervals and zero everywhere else.

Proposition 5.4. *Let $s \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ and $\boldsymbol{\Delta}^* = \{t_i, i=1,\dots,2m+K\}$ be the extended partition associated to the spline space. If for any $i \in \{1,\dots,m+K\}$, $r \in \{1,\dots,m-1\}$,*

$$s(x)=0, \qquad x \in (-\infty, t_i] \cup [t_{i+r}, +\infty),$$

then $s \equiv 0$.

Proof. Suppose that we want to determine a function in the spline space by requiring that it equals 0 together with its derivatives at t_i and also at t_{i+r}, $r \in \{1,\dots,m-1\}$. For $r \in \{1,\dots,m-2\}$, this necessarily leads to the zero constant function, because for a nonzero spline the total number of continuity conditions to be imposed at the knots would exceed the number of degrees of freedom. Now, let $r=m-1$ and suppose that there exists a spline which does not vanish in $(t_i, t_{i+m-1})=(x_{i_1}, x_{i_p})$. The $m(p-1)$ coefficients of the spline can be determined by solving a homogeneous linear system whose coefficients matrix is the matrix A exploited in the proof of Proposition 5.3. Being such a matrix nonsingular, the unique solution of the system is the zero vector. □

Since our aim is to provide an approach for the construction of a B-spline basis and, ultimately, the ONTP basis for a spline space (if it admits one), we observe that a necessary condition for the existence of such a basis is that each space $\mathcal{U}_{i,m}$ contains constants, so that the normalization property iv) in Definition 5.8 is satisfied. This entails that the function u that appears in the definition of the transition functions is chosen to be $u \equiv 1$. Moreover, recalling that each $\mathcal{U}_{i,m}$ is spanned by the functions $\{u_{i,1},\dots,u_{i,m}\}$, it is convenient to set $u_{i,1} \equiv 1$ for all i.

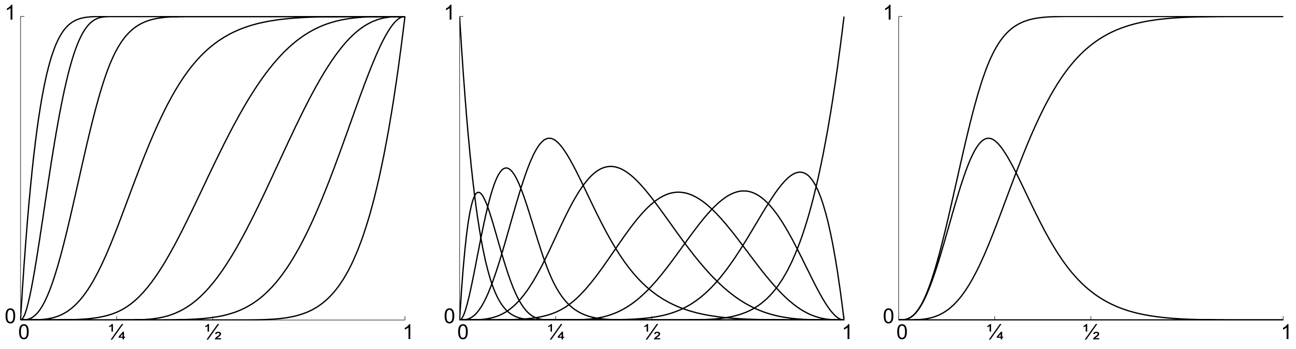
In view of this, the transition functions introduced in Definition 5.10 can be exploited to define a set of functions $\{N_{i,m}, i=1,\dots,m+K\}$ as follows:

$$\begin{aligned} N_{i,m} &:= f_{i,m} - f_{i+1,m}, \qquad i=1,\dots,m+K-1, \\ N_{m+K,m} &:= f_{m+K,m}. \end{aligned} \tag{5.3}$$

Proposition 5.5. *The functions $\{N_{i,m}, i=1,\dots,m+K\}$ defined as above enjoy by construction the properties i), iii), and iv) of Definition 5.8.*

Proof. These properties readily follow from the definition of transition functions.

i) By construction, $f_{i,m}$ is zero for $x<t_i$ and is equal to $u \equiv 1$ for $x>t_{i+m-1}$. Analogously, $f_{i+1,m}$ is 0 for $x<t_{i+1}$ and 1 for $x>t_{i+m}$. Therefore, $N_{i,m}=f_{i,m}-f_{i+1,m}$ is zero for $x<t_i$ and for $x>t_{i+m}$, proving that the support property holds true.



(a) **(b)** **(c)**

Figure 5.1: Transition functions (left) and B-spline basis (center) for the spline space $S(\boldsymbol{\mathcal{U}}_6, \boldsymbol{M}, \boldsymbol{\Delta})$ with extended partition $\boldsymbol{\Delta}^* = \{0, 0, 0, 0, 0, 0, \frac{1}{4}, \frac{1}{4}, \frac{1}{2}, 1, 1, 1, 1, 1, 1\}$ and $\mathcal{U}_{i,6} = \text{span}\{1, x, \cos x, \sin x, \cosh x, \sinh x\}$ for all i. The figure on the right shows a single B-spline basis function and the two transition functions involved in its definition.

iii) By construction, $f_{i,m}$ satisfies conditions (5.1), where now $u \equiv 1$ implies $D^r u(x_{i_p}) = \delta_{r,0}$, $r = 0, \ldots, k_{i,i_p}$. Since $N_{i,m} = f_{i,m} - f_{i+1,m}$ has at t_i a zero of the same multiplicity as $f_{i,m}$ and at t_{i+m} a zero of the same multiplicity as $f_{i+1,m}$, it fulfills the endpoint property.

iv) We have to verify that the normalization property is satisfied, i.e., that $\sum_{i=1}^{m+K} N_{i,m} = 1$. Substituting each $N_{i,m}$ with the difference of the related transition functions, we get

$$\sum_{i=1}^{m+K} (f_{i,m} - f_{i+1,m}) = f_{1,m} - f_{2,m} + f_{2,m} - f_{3,m} + \ldots - f_{m+K,m} + f_{m+K,m} = 1,$$

since $f_{1,m} \equiv 1$ and subsequent terms cancel each other in the sum.

□

Proposition 5.6. *Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a spline space that has the B-spline basis. Then the transition functions $\{f_{i,m}, i = 1, \ldots, m+K\}$ for such space, which satisfy Definition 5.10 with $u \equiv 1$, form a basis for $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ and $f_{i,m}(x) \in [0, 1]$, for all $x \in [a, b]$.*

Proof. Let $\bar{f}_{i,m} := \sum_{j=i}^{m+K} N_{j,m}$, where $\{N_{i,m}, i = 1, \ldots, m+K\}$ is the B-spline basis of $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$. It can be easily checked that $\bar{f}_{i,m}$ satisfies Definition 5.10 with $u \equiv 1$. From the uniqueness of the transition functions (see Proposition 5.3) follows that $\bar{f}_{i,m} = f_{i,m}$ for all $i = 1, \ldots, m+K$. Moreover, from the linear independence, nonnegativity, and partition of unity properties of the B-spline basis we obtain that the functions $\{f_{i,m}, i = 1, \ldots, m+K\}$ are linearly independent and that $0 \leqslant f_{i,m}(x) \leqslant 1$, for all $i = 1, \ldots, m+K$, $x \in [a, b]$. □

Corollary 5.7. *If $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits a B-spline basis in the sense of Definition 5.8, then such a basis is precisely given by the functions defined in* (5.3).

As an example, Figure 5.1 shows the transition functions and the related B-spline basis in the case of a generalized spline space defined on mixed polynomial/trigonometric/hyperbolic QEC-spaces $\mathcal{U}_{i,6} = \text{span}\{1, x, \cos x, \sin x, \cosh x, \sinh x\}$.

5.3.1 Generalized spline spaces that admit the ONTP basis

In this section, we investigate some properties that the transition functions constructed by the proposed approach enjoy if the generalized spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits the ONTP basis.

Preliminarily, we point out that Mazure [Maz08, Maz11c] has recently characterized the largest class of spaces of sufficient regularity which are *suitable for design* as the one of QEC-spaces containing constants and such that the associated space of derivatives is also a QEC-space. In fact, she proved that under these assumptions a space admits the Bernstein basis, as we recall below together with the related definition of space of derivatives.

Definition 5.11 (Space of derivatives)**.** Given an m-dimensional function space $\mathcal{U}_m$ containing constants and spanned by the functions $\{1, u_2, \dots, u_m\}$, we define $D\mathcal{U}_m$ as the $(m-1)$-dimensional space spanned by the functions $\{Du_2, \dots, Du_m\}$.

Proposition 5.8. *Let $\mathcal{U}_m \subset C^{m-2}(I)$ be an m-dimensional space. Then the following properties are equivalent:*

i) $\mathcal{U}_m$ is a QEC-space on I;

ii) $\mathcal{U}_m$ possesses a Bernstein-like basis relative to any $[a, b] \subseteq I$, $a < b$.

Proposition 5.9. *Let $\mathcal{U}_m \subset C^{m-2}(I)$ be an m-dimensional space that contains constants. Then the following properties are equivalent:*

i) the space $D\mathcal{U}_m$ is an $(m-1)$-dimensional QEC-space on I;

ii) $\mathcal{U}_m$ possesses a Bernstein basis relative to any $[a, b] \subseteq I$, $a < b$.

In view of the above results, we remark that a necessary condition for the existence of the ONTP basis for a generalized spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ is that each space $D\mathcal{U}_{i,m}$ is a QEC-space. In fact, if $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ has the ONTP basis, then also any space obtained from $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ by knot insertion has the ONTP basis. This includes the space obtained when the multiplicity of each knot is equal to m, where each piece is represented in the Bernstein basis. For this reason, we assume hereinafter to work with QEC-spaces $\{\mathcal{U}_{i,m}\}$ that contain constants and such that also $\{D\mathcal{U}_{i,m}\}$ are QEC-spaces on the related knot interval. This means that on the interval I_i the space $\mathcal{U}_{i,m}$ possesses the Bernstein basis.

Before proceeding to discuss some relevant properties enjoyed by the transition functions in the case of existence of the ONTP basis, we introduce the spline space whose sections are spanned by the derivatives of the QEC-systems associated to the spaces $\mathcal{U}_{i,m}$.

Definition 5.12 (Spline space of derivatives)**.** Let each $D\mathcal{U}_{i,m}$ be a QEC-space on the related interval I_i, $i = 0, \dots, k$, and denote $D\boldsymbol{\mathcal{U}}_m := \{D\mathcal{U}_{0,m}, \dots, D\mathcal{U}_{k,m}\}$. Then the spline space of derivatives is the spline space $DS(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) = S(D\boldsymbol{\mathcal{U}}_m, \bar{\boldsymbol{M}}, \boldsymbol{\Delta})$, where the vector of multiplicities $\bar{\boldsymbol{M}} := (\bar{m}_1, \dots, \bar{m}_k)$ is such that $\bar{m}_i = \min(m_i, m-1)$, $i = 1, \dots, k$.

Proposition 5.10. *Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a generalized spline space that has an ONTP basis. Then, each transition function $f_{i,m}$ is positive and monotonically increasing in $(x_{i_1}, x_{i_p}) = (t_i, t_{i+m-1})$ and the set of functions $\{Df_{i,m}, i = 2, \dots, m+K\}$ is a B-spline-like basis for $DS(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$.*

Proof. Let us denote by $\{N_{i,m}, i = 1, \ldots, m+K\}$ the ONTP basis of $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$. By (5.3) and the proof of Proposition 5.6, we get that in this basis the transition functions have the expression

$$f_{i,m}(x) = \sum_{j=i}^{m+K} N_{j,m}(x), \qquad \forall x \in [a,b], \tag{5.4}$$

for any $i = 1, \ldots, m+K$. Since by equation (5.4) the coefficients of $f_{i,m}$ w.r.t. the ONTP basis $\{N_{i,m}\}$ form a non-decreasing nonnegative sequence, namely $0, \ldots, 0, 1, \ldots, 1$, from the variation diminishing property of the ONTP basis it follows that $f_{i,m}$ is positive and monotonically non-decreasing, in particular monotonically increasing, in (t_i, t_{i+m-1}).
As a consequence, each $Df_{i,m}$ is nonnegative, locally supported, and enjoys the endpoint property of Definition 5.8. Hence, $\{Df_{i,m}\}$ form a B-spline-like basis for $DS(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$. □

Proposition 5.11. *Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a generalized spline space that as a B-spline basis, and let $\{f_{i,m}, i = 1, \ldots, m+K\}$ be the transition functions. Then, $f_{i,m}$ has at x_{i_1} a zero of multiplicity exactly $m - m_{i_1} - 1$ if $m_{i_1} > 1$ and at least $m - m_{i_1} - 1$ if $m_{i_1} = 1$.*

Proof. Since the spline space admits a B-spline basis, from Proposition 5.6 it follows that the transition functions are linearly independent. In particular, the restrictions of the m transition functions $f_{j,m}$, $j = i-m+1, \ldots, i$, to the knot interval $[t_i, t_{i+1}] = [x_\ell, x_{\ell+1}]$ are linearly independent.
Recalling that the knot x_ℓ has multiplicity m_ℓ, then the transition functions $f_{i-m_\ell+1,m}, \ldots, f_{i,m}$ are constructed by imposing that at x_ℓ they have a zero of multiplicity $m - m_\ell - 1, \ldots, m-2$ respectively. We have to prove that $f_{j,m}$, $j = i-m_\ell+1, \ldots, i-1$, has at x_ℓ a zero of multiplicity not greater than the imposed one.
Under our assumptions, i.e., being $\mathcal{U}_{\ell,m}$ a QEC-space on $[x_\ell, x_{\ell+1}]$, we know that there exists a canonical basis with origin at x_ℓ, which we denote by $(u_1, \ldots, u_m)$. As a canonical basis, it is such that u_j has at x_ℓ a zero of multiplicity exactly $j-2$ for $j = 1, \ldots, m-1$, while u_m has there a zero of multiplicity at least $m-2$.
Proceeding by reductio ad absurdum, let us suppose that one of the functions f_{i-k}, with $i - m_\ell + 1 \leqslant i - k \leqslant i-1$, has at x_ℓ a zero of multiplicity greater than the one imposed by construction. In this way, now the functions f_{i-j}, $j = k, \ldots, 0$, all have at x_ℓ a zero of multiplicity greater than $m-2-k$. In terms of canonical basis functions, the functions f_{i-j} for $j = k, \ldots, 0$ must be represented only by the u_j with $j = m-k, \ldots, m-1$, because they are the only ones having at x_ℓ a zero of multiplicity greater than $m-2-k$. Now, the number of u_j functions involved in such a representation is k, while the transition functions to be represented are $k+1$. From this follows that f_{i-j}, $j = k, \ldots, 0$, cannot be linearly independent, which conflicts with what stated above. □

Proposition 5.12. *Let $\boldsymbol{\Delta}^* = \{t_i, i = 1, \ldots, 2m+K\}$ be an extended partition. By inserting a new knot $\hat{t}$ in $\boldsymbol{\Delta}^*$, $t_\ell \leqslant \hat{t} < t_{\ell+1}$, we obtain a new knot partition $\hat{\boldsymbol{\Delta}}^* = \{\hat{t}_i, i = 1, \ldots, 2m+K+1\}$. If the associated spline spaces are such that $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) \subset \hat{S}(\hat{\boldsymbol{\mathcal{U}}}_m, \hat{\boldsymbol{M}}, \hat{\boldsymbol{\Delta}})$, and $\{N_{i,m}\}$ and $\{\hat{N}_{i,m}\}$ are the B-spline functions on the knot partitions $\boldsymbol{\Delta}^*$ and $\hat{\boldsymbol{\Delta}}^*$ respectively, there exist coefficients $0 \leqslant \alpha_i \leqslant 1$, $i = 1, \ldots, m+K$, such that*

$$f_{i,m} = \alpha_{i,m}\, \hat{f}_{i,m} + (1 - \alpha_{i,m})\, \hat{f}_{i+1,m}, \qquad i = 1, \ldots, m+K. \tag{5.5}$$

Proof. In general, it holds

$$f_{i,m} = \alpha_{i,m}\hat{f}_{i,m} + \beta_{i,m}\hat{f}_{i+1,m}, \qquad i = 1, \ldots, m+K, \tag{5.6}$$

i.e., $f_{i,m}$ is a combination of the two functions $\hat{f}_{i,m}$ and $\hat{f}_{i+1,m}$ only. This follows from expressing $f_{i,m}$ as a combination of the new basis functions $\{\hat{f}_{i,m}\}$ and differentiating: since $\{Df_{i,m}\}$ and are $\{D\hat{f}_{i,m}\}$ B-spline-like functions (see Proposition 5.10) and thus have compact support, the contribution in the combination can come only from $\hat{f}_{i,m}$ and $\hat{f}_{i+1,m}$.

Under the assumptions of this proposition, the transition functions are positive and take values from 0 to 1. Thus, each $f_{i,m}$ is monotonically increasing in a right neighborhood of x_{i_1} and in a left neighborhood of x_{i_p}, and coefficients $\alpha_{i,m}$ and $\beta_{i,m}$ are positive since they are given by the following expressions:

$$\alpha_{i,m} = \begin{cases} 1, & i \leqslant \ell - m + 1, \\ \dfrac{D_+^{k^*_{i,i_1}+1} f_{i,m}(x_{i_1})}{D_+^{k^*_{i,i_1}+1} \hat{f}_{i,m}(x_{i_1})}, & \ell - m + 2 \leqslant i \leqslant \ell - r + 1, \\ 0, & i \geqslant \ell - r + 2, \end{cases} \tag{5.7}$$

and

$$\beta_{i,m} = \begin{cases} 0, & i \leqslant \ell - m + 1, \\ \dfrac{D_-^{k^*_{i,i_p}+1} f_{i,m}(x_{i_p})}{D_-^{k^*_{i,i_p}+1} \hat{f}_{i+1,m}(x_{i_p})}, & \ell - m + 2 \leqslant i \leqslant \ell - r + 1, \\ 1, & i \geqslant \ell - r + 2, \end{cases}$$

where k^*_{i,i_j} is the smallest integer such that $D_{+/-}^{k^*_{i,i_j}+1} f_{i,m}(x_{i_j}) > 0$, and $1 \leqslant r \leqslant m$ is the multiplicity of $\hat{t}$ in $\hat{\boldsymbol{\Delta}}^*$.

Now we want to show that $\beta_{i,m} = 1 - \alpha_{i,m}$. Evaluating (5.6) at $\bar{t} > t_{i+m-1}$, we get

$$1 = f_{i,m} = \hat{f}_{i,m} = \hat{f}_{i+1,m},$$

and it follows that $1 = \alpha_{i,m} + \beta_{i,m}$.

As a consequence of the fact that $\alpha_{i,m}$ and $\beta_{i,m}$ are positive and sum to 1, we get $0 \leqslant \alpha_{i,m} \leqslant 1$. □

Corollary 5.13. *Under the assumptions of the above proposition, by considering the B-spline functions* $\{N_{i,m}\}$ *defined as in* (5.3)*, we can obtain the well-known knot-insertion relation with nonnegative coefficients*

$$N_{i,m} = \alpha_{i,m}\hat{N}_{i,m} + (1 - \alpha_{i+1,m})\hat{N}_{i+1,m}.$$

Proof. In fact,

$$\begin{aligned} N_{i,m} = f_{i,m} - f_{i+1,m} &= \alpha_{i,m}\hat{f}_{i,m} + (1 - \alpha_{i,m})\hat{f}_{i+1,m} - \alpha_{i+1,m}\hat{f}_{i+1,m} - (1 - \alpha_{i+1,m})\hat{f}_{i+2,m} \\ &= \alpha_{i,m}(\hat{f}_{i,m} - \hat{f}_{i+1,m}) + \hat{f}_{i+1,m} - \hat{f}_{i+2,m} - \alpha_{i+1,m}(\hat{f}_{i+1,m} - \hat{f}_{i+2,m}) \\ &= \alpha_{i,m}\hat{N}_{i,m} + (1 - \alpha_{i+1,m})\hat{N}_{i+1,m}. \end{aligned}$$

□

5.4 On the existence of the ONTP basis

It is well known that not all spaces of piecewise quasi Chebyshevian splines possess an ONTP basis and thus not all such spaces are suitable for design. In particular, as we shall recall in the following, such a problem is closely related to the existence of a system of (piecewise) weight functions that satisfy the properties of being positive and having prescribed continuity at the knots.

In this section we exploit the transition functions in order to derive a system of weight functions for any spline space that admits an ONTP basis. The main relevance of this result lies in the fact that we do not need to know a priori whether the ONTP basis exists or not. In particular, we show that, as soon as one of the computed functions fails to be positive, we can conclude that there does not exist any proper system of weight functions for the considered spline space, or, in other words, that the space is not suitable for design. This provides a criterion to establish whether a given space possesses the ONTP basis.

Besides this, it is interesting to remark that, prior to the present work, knowing the weight functions was the only viable way for computing the B-spline basis of an arbitrary generalized spline space, that is, for applying the generalized integral recurrence relation [Maz11a]. In this view, the lack of a general method for deriving the weight functions associated with the target space made the integral approach a merely theoretical tool. Thus the weight functions constructed in the following sections also allow us to exploit the integral recurrence relation for practical computation of the B-spline basis. In this respect, however, it should be noted that transition functions are easier to handle and computationally simpler than the integral approach.

In the remainder of this section, we first recall some basic definitions and results from the theory of weight functions and spline spaces. Successively, we illustrate the construction of a system of weight functions for a generalized spline space by exploiting the transition functions (Section 5.4.1), and then in Section 5.4.2 we propose and justify a numerical test to establish whether a generalized spline space has the ONTP basis. Finally, in Section 5.4.3 we focus on the particular case of an empty knot partition $\boldsymbol{\Delta} = \emptyset$, where a single QEC-space is considered on the whole domain and the ONTP basis is the Bernstein one.

We start with proving that all generalized spline spaces based on 2-dimensional QEC-spaces $\{\mathcal{U}_{i,2}\}$ containing constants have the ONTP basis.

Proposition 5.14. *All generalized spline spaces $S(\boldsymbol{\mathcal{U}}_2, \boldsymbol{M}, \boldsymbol{\Delta})$ (i.e., with $m = 2$) such that each $\mathcal{U}_{i,2}$ contains constants possess the ONTP basis.*

Proof. The idea is to exploit the transition functions to construct a B-spline basis and prove that it actually is the ONTP basis, which corresponds to a piecewise Langrage-type basis.
If $t_i \neq t_{i+1}$, the transition function $f_{i,2}$ has a unique zero at t_i, takes values from 0 to 1, and it cannot vanish in (t_i, t_{i+1}) since $\mathcal{U}_{i,2}$ is a QEC-space. Therefore, $\{N_{i,2} = f_{i,2} - f_{i+1,2}\}$ form a basis that fulfills all the properties i)–iv) in Definition 5.8. Moreover, for any other knot partition $\hat{\boldsymbol{\Delta}}$ obtained from $\boldsymbol{\Delta}$ by knot insertion, we can perform an analogous construction that results in a B-spline basis. Then, in view of Proposition 5.2, $S(\boldsymbol{\mathcal{U}}_2, \boldsymbol{M}, \boldsymbol{\Delta})$ admits the ONTP basis. □

Definition 5.13 (Piecewise weight functions)**.** A sequence of piecewise functions $\{w_0, \ldots, w_{m-2}\}$ is a system of *piecewise weight functions* on a partition $\boldsymbol{\Delta} = \{x_i, i = 1, \ldots, k\}$ of $[a, b]$ if, for any $j = 0, \ldots, m-2$:

i) w_j is positive on $[a, b]$;

ii) for all $i = 0, \ldots, k$, w_j is $C^{m-j-m_i^j-1}$ at x_i, where $m_i^j := \min(m_i, m-j)$.

Given a system of piecewise weight functions, we may define the associated piecewise differential operators (or generalized derivatives) on $\boldsymbol{\Delta}$ as

$$L_0 f := \frac{f}{w_0}, \qquad L_j f := \frac{1}{w_j} D L_{j-1} f, \quad j = 1, \ldots, m-2,$$

where $f \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ and D denotes ordinary differentiation.
Accordingly, we denote by $L_j S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) = S(L_j \boldsymbol{\mathcal{U}}_m, \boldsymbol{M}_j, \boldsymbol{\Delta})$ the generalized spline space having sections in the spaces $L_j|_{I_i} \mathcal{U}_{i,m}$ obtained by generalized differentiation through the weight functions $w_j|_{I_i}$, $i = 0, \ldots, k$, and where $\boldsymbol{M}_j := \left(m_1^j, \ldots, m_k^j\right)$.

Remark 5.3. From the above definitions, it straightforwardly follows that $w_j \in D L_{j-1} S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$.

Remark 5.4. Note that if $\dim \mathcal{U}_{i,m} = m$, then $\dim L_j|_{I_i} \mathcal{U}_{i,m} = m - j$.

Proposition 5.15. *Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a generalized spline space containing constants. The space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits the ONTP basis if and only if it is possible to find a system of piecewise weight functions on $\boldsymbol{\Delta}$ and a spline space $T := S(L_{m-2} \boldsymbol{\mathcal{U}}_m, \boldsymbol{M}_{m-2}, \boldsymbol{\Delta})$ of dimension $2 + K_{m-2}$, with $K_{m-2} := \sum_{i=1}^{k} m_i^{m-2}$ and m_i^j as in Definition 5.13, that contains constants and such that*

$$S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) = \{ s \in C^{m - m_i^0 - 1} \ at\ x_i \,\big|\, L_{m-2} s \in T \}.$$

Proof. If $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits the ONTP basis, which means a B-spline basis and knot insertion, the proof consists in showing that the construction of a system of piecewise weight functions is always feasible, and that the generalized spline spaces $L_j S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ obtained by general differentiation are normalized and still admit the ONTP basis, up to the space $L_{m-2} S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ which has the ONTP basis by Proposition 5.14. We will provide a proof of this kind later in Section 5.4.1, based on the explicit construction of a sequence of functions $\{w_j\}$ that will be described there.
Conversely, if a spline space T of dimension $2 + K_{m-2}$ and a system of piecewise weight functions are given, then starting from the ONTP basis of T (which exists, as proven in Proposition 5.14) the integration procedure generates the ONTP basis for the space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$. □

Corollary 5.16. *Let a generalized spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ containing constants be given, together with the sequence of piecewise weight functions $\{w_j, j = 0, \ldots, m-2\}$, with $w_0 = 1$. $L_j S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$, with $j \in \{0, \ldots, m-2\}$, has the ONTP basis if and only if $L_i S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$, for all $i = 0, \ldots, m-2$, have the ONTP basis.*

5.4.1 Construction of weight functions for proving the existence of the ONTP basis

Given a generalized spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ and the related transition functions, in this section we intend to show that they can be exploited to define a particular sequence of functions $\{w_j\}$ that is meaningful for two main reasons:

- if $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits the ONTP basis, then $w_0 = 1$ and w_j, $j = 1, \ldots, m-2$, represent a system of piecewise weight functions for that spline space;
- if there is at least one $r \in \{1, \ldots, m-2\}$ such that w_r is not positive, then we can conclude that $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ has no ONTP basis.

The two above facts are proved respectively in Proposition 5.15 and Proposition 5.17, based on the procedure of generation of the functions $\{w_j\}$ given below.

Remark 5.5. Note that, unlike the approach that is usually undertaken in the literature, the piecewise functions $\{w_j\}$ that we will define in the following will trivially met the condition on the continuity at knots in Definition 5.13 because they will belong to spline spaces $DL_jS(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$. Hence, we will only have to check their positivity in order to label them as a system of piecewise weight functions.

In Algorithm 1, in the subsequent description, and in the proof of Proposition 5.15, we adopt the shorthand notation S in place of $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ to denote the initial spline space, and recalling Definition 5.13 we set $K_j := \sum_{i=1}^{k} m_i^j$.

Algorithm 1 (Construction of the sequence $\{w_j, j = 0, \dots, m-2\}$)**.** Let $w_0 = 1$.

- Step 0:
  - (N) In the space $L_0S = \frac{S}{w_0}$, compute $f_{i,m}$, $i = 1, \dots, m+K$, by solving (5.2);
  - (D) In the space DL_0S, set $\tilde{N}_{i,m-1} = Df_{i+1,m}$, $i = 1, \dots, m+K-1$[1];
  - (S) Still in DL_0S, compute $\tilde{f}_{i,m-1} = \sum_{\ell=i}^{m+K_1-1} \tilde{N}_{\ell,m-1}$, $i = 1, \dots, m+K_1-1$, and set $w_1 := \tilde{f}_{1,m-1}$;
- Step j, $j = 1, \dots, m-3$:
  - (N) In the space $L_jS = \frac{1}{w_j} DL_{j-1}S$, compute $f_{i,m-j} = \frac{\tilde{f}_{i,m-j}}{w_j}$, $i = 1, \dots, m-j+K_j$;
  - (D) In the space DL_jS, set $\tilde{N}_{i,m-j-1} = Df_{i+1,m-j}$, $i = 1, \dots, m-j+K_j-1$[1];
  - (S) Still in DL_jS, compute $\tilde{f}_{i,m-j-1} = \sum_{\ell=i}^{m-j+K_{j+1}-1} \tilde{N}_{\ell,m-j-1}$, $i = 1, \dots, m-j+K_{j+1}-1$, and set $w_{j+1} := \tilde{f}_{1,m-j-1}$.

Remark 5.6. Note that, if $\boldsymbol{\Delta}$ includes knots with multiplicity greater than 1, C^0 functions occur in the above construction. In this case, differentiation has to be intended as for piecewise functions, i.e., working separately on the knot intervals.

Each step of the above procedure can be split in three successive stages, referred to as *normalization* (N), *differentiation* (D) and *summation* (S) stage. At step 0, having taken $w_0 = 1$, the normalization stage gives back the spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ and for such space the related transition functions $\{f_{i,m}\}$ can be computed as the solution of the linear system (5.1). Note that, since all the m-dimensional QEC-spaces $\mathcal{U}_{i,m}$ contain constants, the spline space DL_0S obtained by ordinary differentiation is spanned by $(m-1)$-dimensional QEC-spaces. Moreover, since $f_{1,m} = 1$, the differentiation stage consists in computing $m+K-1$ functions, namely $\tilde{N}_{i,m-1} = Df_{i+1,m}$, $i = 1, \dots, m+K-1$. Finally, during the summation stage, we compute a new sequence $\tilde{f}_{i,m-1} = \sum_{\ell=i}^{m+K_1-1} \tilde{N}_{\ell,m-1}$, $i = 1, \dots, m+K_1-1$. Note that $\{\tilde{f}_{i,m-1}\}$ matches the properties of Definition 5.10, with $u = \tilde{f}_{1,m-1}$. Based on this observation, we shall choose $w_1 = \tilde{f}_{1,m-1}$ to proceed to the successive step.

For any subsequent step $j = 1, \dots, m-3$, during the normalization stage we generate the space L_jS by dividing the computed functions $\tilde{f}_{i,m-j}$ by the current function w_j, which is the one computed at the summation stage of step $j-1$. In this way, we obtain a set of transition functions for L_jS by a simple update of the previous ones. Because w_j is nothing but $\tilde{f}_{1,m-j}$, there follows that $f_{1,m-j} = 1$ and the space L_jS contains the unit function. The

[1] For any $j = 0, \dots, m-3$, taking into account that some of the functions $Df_{i+1,m-j}$, $i = 1, \dots, m-j+K_j-1$, may be identically zero, the sequence $\{\tilde{N}_{i,m-j-1}\}$ can be re-indexed as $\tilde{N}_{i,m-j-1}$, $i = 1, \dots, m-j+K_{j+1}-1$, where $m-j+K_{j+1}-1$ is precisely the dimension of the spline space of derivatives DL_jS.

latter property ensures that, proceeding to the differentiation stage, the space DL_jS generated by the derivatives of such transition functions has segments belonging to $(m-j-1)$-dimensional QEC-spaces and accordingly the space DL_j has dimension $m-j+K_{j+1}-1$. Finally, during the summation stage, we generate a new sequence $\{\tilde{f}_{i,m-j-1}\}$, whose first element is the function $w_{j+1}=\tilde{f}_{1,m-j-1}$, which we shall exploit at the successive iteration.

By means of transition functions and the construction of the sequence $\{w_j\}$ described above, it is possible to verify that the spaces $L_jS(\boldsymbol{\mathcal{U}}_m,\boldsymbol{M},\boldsymbol{\Delta})$ have the ONTP basis. In fact, we are now able to provide a constructive proof of the implication in Proposition 5.15 postponed earlier. It consists in showing that the functions $\{w_j\}$ obtained through the procedure outlined in Algorithm 1 are indeed piecewise weight functions.

Proof of Proposition 5.15. As already observed in Remark 5.5, the functions $\{w_j\}$ trivially satisfy the continuity requirements of Definition 5.13 at the knots by construction, therefore we shall prove that they are positive. To this aim, we proceed by induction on j.

At the 0th step the statement holds, since $w_0=1$ and we are assuming that $S:=S(\boldsymbol{\mathcal{U}}_m,\boldsymbol{M},\boldsymbol{\Delta})$ contains constants and has the ONTP basis.

For the inductive step, let the statement hold true at the $(j-1)$th step, which means that w_j is positive and the space L_jS admits a B-spline basis and knot insertion. We have to prove that w_{j+1} is positive and that also the space $L_{j+1}S$ admits a B-spline basis and knot insertion. According to the construction in Algorithm 1, recalling that K_j denotes the dimension of the space L_jS, at step j we get

$$w_{j+1}=\sum_{\ell=1}^{m-j+K_j-1} Df_{\ell+1,m-j}, \tag{5.8}$$

which is a positive function because, by Proposition 5.10, the transition functions are monotonically increasing and thus their derivatives are positive.
From Proposition 5.12 it follows that, in order to verify that the space $L_{j+1}S$ admits knot insertion, it is sufficient to verify that the spaces $L_{j+1}S$ and $L_{j+1}\hat{S}$, obtained by inserting a node $\hat{t}\in[t_\ell,t_{\ell+1})$, both possess B-spline bases, respectively $N_{i,m-j-1}$ and $\hat{N}_{i,m-j-1}$. To this aim, exploiting the relation (5.5) between functions $\{f_{i,m-j}\}$ and functions $\{\hat{f}_{i,m-j}\}$, formula (5.7), and (5.8), we can write:

$$\begin{aligned} w_{j+1}&=\sum_{i=1}^{\ell-m+j} D\hat{f}_{i,m-j}+\sum_{i=\ell-m+j+1}^{\ell-r+1}\alpha_{i,m-j}D\hat{f}_{i,m-j}+(1-\alpha_{i,m-j})D\hat{f}_{i+1,m-j}+\sum_{i=\ell-r+2}^{m-j+K_j} D\hat{f}_{i+1,m-j}\\ &=\sum_{i=1}^{m-j+K_j+1} a_{i,m-j}D\hat{f}_{i,m-j}, \end{aligned}$$

with

$$a_{i,m-j}=\begin{cases} 1, & i=1,\dots,l-m+j+1,\\ \alpha_{i,m-j}, & i=\ell-m+j+2,\\ \alpha_{i,m-j}+(1-\alpha_{i-1,m-j}), & i=\ell-m+j+3,\dots,\ell-r+1,\\ 1-\alpha_{i,m-j}, & i=\ell-r+2,\\ 1, & i=\ell-r+3,\dots,m-j+K_j+1. \end{cases}$$

From the above expression and Proposition 5.12, the coefficients $a_{i,m-j}$ are positive.
This allows us to define a B-spline-like basis for $DL_j\hat{S}$ by scaling the functions $\{D\hat{f}_{i,m-j}\}$ by

coefficients $\{a_{i,m-j}\}$, more precisely:

$$\tilde{\hat{N}}_{i,m-j-1} := a_{i+1,m-j} D\hat{f}_{i+1,m-j}, \qquad i = 1, \ldots, m-j+K_j.$$

In this way, at the summation stage of step j it results that $\hat{w}_{j+1} \equiv w_{j+1}$, and at the successive normalization stage of step $j+1$ the two spaces $L_{j+1}\hat{S}$ and $L_{j+1}S$ turn out to be related by the insertion of the knot $\hat{t}$. In fact, the division by w_{j+1} just provides the normalization, thus proving that both spaces possess a B-spline basis.

At the last step we get the spline space $L_{m-2}S$, which by construction has dimension $2+K_{m-2}$ and has a B-spline basis. By Proposition 5.14, it admits the ONTP basis. □

Proposition 5.17. *If the construction outlined in Algorithm 1 does not provide a complete sequence of functions $\{w_j\}$ that are all positive, then there is no sequence of piecewise weight functions that verify Proposition 5.15. Hence, in this case the considered generalized spline space does not admit an ONTP basis.*

Proof. The proof is by reductio ad absurdum. Let us suppose that the ONTP basis $\{N_{i,m}\}$ for $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ exists, despite one function provided by our construction is nonpositive, say w_r. Then, we can define the transition functions $f_{i,m} := \sum_{\ell=i}^{m+K} N_{\ell,m}$ and proceed with steps $j = 0, \ldots, r-1$, up to generating the function w_r. But from the existence of the ONTP basis and in view of Corollary 5.16, it would follow that this function is positive, contradicting what supposed. □

5.4.2 A numerical approach for checking the existence of the ONTP basis

In this section, we present a practical numerical test to establish whether the ONTP basis for a given generalized spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ containing constants exists. As we will see, this procedure exploits the construction provided in Algorithm 1 and requires verifying the positivity of each function w_j generated at step j. In particular, it checks the existence of a B-spline-like basis in spaces $DL_jS(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ by inspecting the positivity and the monotonicity of the transition functions.

Before detailing the numerical approach, we present some results that will be used to justify our way of proceeding. Specifically, we show that a monotonically increasing spline function must have a monotonically increasing sequence of coefficients when represented in the B-spline basis, but also in any other basis obtained from it by knot insertion, and in particular in the piecewise Bernstein basis.

Proposition 5.18. *Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a generalized spline space with ONTP basis. $s \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ has non-decreasing and nonnegative coefficients in the B-spline basis $\{N_{j,m}\}$ if and only if Ds has nonnegative coefficients in the B-spline-like basis $\{\tilde{N}_{j,m-1}\}$.*

Proof. Let $s \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ such that $s = \sum_{j=1}^{m+K} c_j N_{j,m}$ with $c_j \geqslant 0$ and $c_{j+1} - c_j \geqslant 0$. Then, recalling (5.3),

$$\begin{aligned} Ds = D\sum_{j=1}^{m+K} c_j N_{j,m} &= D\sum_{j=1}^{m+K} c_j (f_{j,m} - f_{j+1,m}) = \sum_{j=1}^{m+K} c_j (Df_{j,m} - Df_{j+1,m}) \\ &= \sum_{j=1}^{m+K} c_j Df_{j,m} - \sum_{j=1}^{m+K} c_j Df_{j+1,m} = \sum_{j=2}^{m+K} c_j \tilde{N}_{j-1,m-1} - \sum_{j=1}^{m+K-1} c_j \tilde{N}_{j,m-1} \end{aligned}$$

$$= \sum_{j=1}^{m+K-1} (c_{j+1} - c_j)\tilde{N}_{j,m-1} = \sum_{j=1}^{m+K-1} d_j \tilde{N}_{j,m-1},$$

where $d_j \coloneqq c_{j+1} - c_j$. Hence, $\{c_j\}$ are non-decreasing and nonnegative if and only if $\{d_j\}$ are nonnegative. □

Proposition 5.19. *Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a generalized spline space with ONTP basis. $s \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ has non-decreasing coefficients in the B-spline basis $\{N_{j,m}\}$ if and only if s has nonnegative coefficients in any other basis of a space $\hat{S}(\hat{\boldsymbol{\mathcal{U}}}_m, \hat{\boldsymbol{M}}, \hat{\boldsymbol{\Delta}})$ obtained from $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ by knot insertion.*

Proof. Let $s \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) \subset \hat{S}(\hat{\boldsymbol{\mathcal{U}}}_m, \hat{\boldsymbol{M}}, \hat{\boldsymbol{\Delta}})$. Since $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ possesses the ONTP basis, then it admits a B-spline basis and knot insertion with nonnegative coefficients. For this reason, if s has non-decreasing coefficients in the B-spline basis of $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$, by knot insertion the new coefficients are still non-decreasing because they are a convex combination of the old ones. □

Corollary 5.20. *In particular, s must have non-decreasing coefficients in the B-spline basis $\hat{N}_{j,m}$ obtained by inserting each node of $\boldsymbol{\Delta}$ up to multiplicity $m-1$. In such a basis, s has on any knot interval the same coefficients that it would have in the associated Bernstein basis.*

Proof. Let $s \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be expressed in the B-spline basis as $s = \sum_{j=1}^{m+K} c_j N_{j,m}$ with $c_j \geqslant 0$ and $c_{j+1} - c_j \geqslant 0$. Then on any knot interval, s, expressed in the Bernstein basis, has nonnegative and non-decreasing coefficients:

$$s|_{I_i} = \sum_{\ell=1}^{m} c_\ell^{[i]} B_{\ell,m}^{[i]} \quad \text{with} \quad c_\ell^{[i]} \geqslant 0 \quad \text{and} \quad c_{\ell+1}^{[i]} - c_\ell^{[i]} \geqslant 0.$$

□

Now, we describe the numerical approach for checking whether a given spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits the ONTP basis, under the assumptions that the space satisfies the necessary conditions previously mentioned in Section 5.3, namely that each $\mathcal{U}_{i,m}$ is a QEC-space containing constants and $D\mathcal{U}_{i,m}$ is a QEC-space as well.

The proposed approach consists in performing the steps of the construction outlined in Algorithm 1, starting from the transition functions determined by solving (5.2). Then, at each step $j = 0, \dots, m-4$, it must be checked that the function w_{j+1} is positive or, equivalently, that the space $DL_j S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits a B-spline-like basis. Finally, by Proposition 5.14 the procedure reaches its completion at the normalization stage of step $m-2$, by verifying the existence of the B-spline basis for the space $L_{m-2} S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$.

In particular, at each step the transition functions are computed using as QEC-systems on the knot intervals the related generalized Bernstein bases. Basically, this means that the transition functions are represented on a Bernstein basis on any interval.

As we have seen in Proposition 5.18 under the assumptions that the ONTP basis exists, the transition functions must have non-decreasing and nonnegative coefficients when expressed in the B-spline basis, and so they must do when expressed in any other basis obtained from it by knot insertion (Proposition 5.19 and Corollary 5.20). Since transition functions are represented in Bernstein bases for this numerical approach, their positivity and monotonicity can be checked by performing a simple test on the coefficients in any interval.

Ultimately, the numerical test consists in checking whether, at each step $j = 0, \ldots, m-2$, the coefficients c_r, $r = 1, \ldots, m-j$, in the Bernstein basis representation of each transition function $f_{i,m-j}$ in each knot interval I_ℓ form a non-decreasing sequence with values in $[0,1]$, namely

$$c_{r+1} - c_r \geqslant 0 \qquad \text{and} \qquad 0 \leqslant c_r \leqslant 1. \tag{5.9}$$

Remark 5.7. If the above test on the coefficients passes at each step of the procedure, then it means that the transition functions are monotonically increasing, their derivatives are positive, and their sums, i.e., the functions $\{w_j\}$, are also positive and thus they form a system of piecewise weight functions. By Proposition 5.15, we can conclude that the spline space admits the ONTP basis.
Conversely, if the test fails at any step, that is, there exists at least one coefficient that does not satisfy conditions (5.9), then one of the functions $\{w_j\}$ is not positive, and by Proposition 5.17 we can assert that the ONTP basis does not exist.

In the case of generalized spline spaces with $m = 3$, if the requirements for the application of the numerical procedure are fulfilled, namely if $\{U_{i,m}\}$ and $\{DU_{i,m}\}$ are QEC-spaces and $\{U_{i,m}\}$ contain constants, then the following proposition guarantees that the spline space admits the ONTP basis and no checking is needed.

Proposition 5.21. *All generalized spline spaces $S(\boldsymbol{\mathcal{U}}_3, \boldsymbol{M}, \boldsymbol{\Delta})$ (i.e., with $m = 3$), containing constants and such that $\{D\mathcal{U}_{i,3}\}$ are QEC-spaces, admit ONTP bases.*

Proof. Each transition function $f_{i,3}$ has in general a nontrivial expression on two knot intervals. Being all the $\{D\mathcal{U}_{j,3}\}$ QEC-spaces by assumptions, each $Df_{i,3}$ cannot have any zero in the interior of its support. Hence, $f_{i,3}$ is either monotonically increasing or monotonically decreasing. Since by construction $f_{i,3}$ takes values from 0 to 1, it is increasing and $Df_{i,3}$ is positive in the interior of its support.

Now, let us consider $N_{i,3} := f_{i,3} - f_{i+1,3}$, for any $i = 1, \ldots, m+K$. By construction, $\{N_{i,3}\}$ satisfy properties i), iii),iv) of Definition 5.8. We prove that any $N_{i,3}$ does not vanish in the interior of its support, and that in particular it is positive, so that it enjoys also property ii). To this aim, we show that $N_{i,3}$ can have only one extremum in its support $[t_i, t_{i+3}]$, or equivalently, that $DN_{i,3}$ has only one zero in (t_i, t_{i+3}). In fact, if $DN_{i,3}$ is nonzero on three consecutive knot intervals, it has a zero at t_i, another zero at t_{i+3}, and, since all $DU_{i,3}$ are QEC-spaces, it can have at most one zero in (t_{i+1}, t_{i+2}). If $DN_{i,3}$ is nonzero on one or two knot intervals, by analogous arguments it follows that it can have at most one zero in the interior of its support. Thus $N_{i,3}$ has only one maximum or minimum.

Positivity of $N_{i,3}$ can be proved by analyzing its behavior at the left endpoint of its support, t_i. Both transition functions involved in the definition of $N_{i,3} = f_{i,3} - f_{i+1,3}$ are monotonically increasing and have at t_i zeros of different multiplicities. For these reasons, their difference is positive in a right neighborhood of t_i, and therefore we can conclude that $N_{i,3} > 0$ in (t_i, t_{i+3}). □

5.4.3 The particular case of an empty knot partition

In the previous sections, while addressing the problem of determining whether a given generalized spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ admits the ONTP basis, we assumed that each space $\mathcal{U}_{i,m}$ contains constants and $D\mathcal{U}_{i,m}$ is a QEC-space. For known results already recalled, this means that each $\mathcal{U}_{i,m}$ has the Bernstein (ONTP) basis on the related interval, and the numerical approach described in the previous section heavily relies on this fact. Hence, knowing the largest interval on which a QEC-space possesses the Bernstein basis is a crucial information, and as

we have mentioned in the historical overview at the beginning of this chapter, much work has been devoted to the study of necessary and sufficient conditions to guarantee the existence of the Bernstein basis.
In particular, in this context the following notions of *critical length* and *critical length for design* have been introduced. We conveniently precede them with the definition of a W-space.

Definition 5.14 (W-space)**.** An m-dimensional space $\mathcal{U}_m$ contained in $C^{m-1}(I)$ is a *W-space* on I if the Wronskian[2] of any basis of $\mathcal{U}_m$ does not vanish on I.

Remark 5.8. Any EC-space on I is a W-space on I.

Definition 5.15 (Critical length)**.** The *critical length* of a given W-space $\mathcal{U}_m$ on $\mathbb{R}$ is the supremum $\ell \in [0, +\infty]$ of all positive h such that $\mathcal{U}_m$ is a QEC-space on any interval $[a, b]$, $a < b$, such that $b - a < h$.

Definition 5.16 (Critical length for design)**.** The *critical length for design* of a given W-space $\mathcal{U}_m$ on $\mathbb{R}$ which contain constants is the supremum $\ell' \in [0, +\infty]$ of all positive h such that $\mathcal{U}_m$ admits the Bernstein basis on any interval $[a, b]$, $a < b$, such that $b - a < h$.

Corollary 5.22 ([CMP03, Corollary 4.1])**.** *Let $\mathcal{U}_m$ be an m-dimensional subset of $C^{m-2}([a, b])$ with constants which is invariant under translations. Let ℓ and ℓ' be the critical lengths of $\mathcal{U}_m$ and $D\mathcal{U}_m$, respectively. Then:*

i) $\ell' \leqslant \ell$;

ii) the space $\mathcal{U}_m$ is QEC with a normalized TP basis on $[a, b]$ if and only if $b - a < \ell'$.

To our knowledge, the value of the critical length for design has been investigated, exactly determined or estimated, for generalized polynomial spaces of the form span$\{1, x, \ldots, x^{m-3}, u(x), v(x)\}$ with suitable conditions on $u(x)$ and $v(x)$, and also span$\{1, x, \ldots, x^{m-5}, u_1(x), u_2(x), u_3(x), u_4(x)\}$ [CMP03, BM12]. These results provides us with the initial information required to use and justify our approach. However, in all the other cases, the problem of determining the critical length for design is still an open issue. For this reason, a numerical method to estimate its value may be useful. To this aim, our approach based on the construction of transition functions can be also exploited as a method for estimating the value of the critical length for design of a given space, if it is not known. This particular application is motivated by the following observation, based on the results recalled above and known in the literature.

Remark 5.9. Let $\mathcal{U}_m$ be an m-dimensional QEC-space on $[a, b]$ containing constants, and let us consider a generic Bernstein-like basis $\{V_{i,m}, i = 1, \ldots, m\}$ (which exists by Proposition 5.8). Moreover, let $\{\tilde{V}_{i,m}\}$ be the functions obtained via scaling the basis functions $\{V_{i,m}\}$ by coefficients $\{c_i\}$ such that $\sum_{i=1}^{m} c_i V_{i,m} = 1$. We call such $\{\tilde{V}_{i,m}\}$ a *signed normalized Bernstein-like basis.*
If we consider the family of all bases of this type obtained as the length of the interval $[a, b]$ varies, we can observe that, on intervals for which the Bernstein basis exists, the $\{\tilde{V}_{i,m}\}$ are all positive. Instead, on intervals for which the Bernstein basis does not exist, some of the $\{\tilde{V}_{i,m}\}$ are positive and others are negative. Since the functions of this family vary with continuity as the length of $[a, b]$ varies, for at least one index i we have that $\tilde{V}_{i,m}$ passes from being positive to being negative, and $\tilde{V}_{i,m} = 0$ represents the critical situation corresponding to the critical length for design ℓ'. In this case, we remark that a signed normalized Bernstein-like basis does not exist.

[2] The *Wronskian* of m functions $u_1, \ldots, u_m \in C^{m-1}(I)$ is a function on I defined as the determinant of the matrix constructed by placing the $(r-1)$th derivative of the functions in the rth row, $r = 1, \ldots, m$.

In view of the above observation, a possible approach for computing the critical length for design ℓ' of a given space relies on determining the critical value for which our constructive method does not provide a basis for the space, since at least one of the generated functions is zero. This criterion can be reformulated in terms of the transition functions, based on the following result.

Proposition 5.23. *Let $\mathcal{U}_m$ be an m-dimensional QEC-space on $[a,b]$. $\mathcal{U}_m$ has the Bernstein basis $\{B_{i,m}\}$ if and only if all transition functions $f_{i,m}$, $i=1,\dots,m$, are monotonically non-decreasing in $[a,b]$.*

Proof. Let us express the transition functions in terms of the Bernstein basis functions as $f_{i,m} = \sum_{j=i}^{m} B_{j,m}$, for $i=1,\dots,m$. Trivially, $\{f_{i,m}\}$ take values in $[0,1]$ and $f_{i,m} > f_{i+1,m}$. From the variation diminishing property of the Bernstein representation, each $f_{i,m}$ is monotonically non-decreasing because its coefficients in the Bernstein basis form a monotonically non-decreasing sequence $0,\dots,0,1,\dots,1$.
Conversely, let the transition functions $\{f_{i,m}\}$ be determined, and define the functions $B_{i,m} \coloneqq f_{i,m} - f_{i+1,m}$, for $i=1,\dots,m-1$, and $B_{m,m} = f_{m,m}$. Since each $f_{i,m}$ is monotonically non-decreasing and takes values from 0 to 1, then $\{Df_{i,m}\}$ form a Bernstein-like basis of $D\mathcal{U}_m$. Hence, by Proposition 5.8, $D\mathcal{U}_m$ is a QEC-space and, by Proposition 5.9, $\{B_{i,m}\}$ is the Bernstein basis. □

Remark 5.10. According to the above results, we can distinguish the following situation depending on the length of the parametric interval.

- On $[0,\ell')$, $f_{i,m} > f_{i+1,m}$ for $i=1,\dots,m$, and the transition functions are monotonically increasing;
- on $[0,\ell']$, at least two transition functions coincide;
- on $[0,\ell)$, there exists an index i such that $f_{i,m} < f_{i+1,m}$;
- on $[0,\ell]$, the linear system (5.2) arising from conditions (5.1) is singular.

Ultimately, the numerical approach for estimating the value of the critical length for design of a given m-dimensional QEC-space $\mathcal{U}_m$ on $[a,b]$ that contains constants consists in the following steps:

1. Determine the transition functions $\{f_{i,m}\}$ by imposing conditions (5.1);
2. For all $i=1,\dots,m-1$, check whether $f_{i,m} > f_{i+1,m}$:
   if two consecutive transition functions coincide, then the related signed normalized Bernstein-like (basis) function is zero, and the length of the interval $[a,b]$ gives the value of the critical length for design ℓ'.

Remark 5.11. This test must be performed on each knot interval also during the numerical procedure proposed in Section 5.4.2 for checking the existence of the ONTP basis in generalized spline spaces, in order to verify the existence of the Bernstein basis on each knot interval.

Remark 5.12. If $\mathcal{U}_m$ is invariant under translations and reflections, the transition functions are symmetric with respect to the line through $(a,0)$ and $(b,1)$. For this reason, it is sufficient to check only $f_{i,m}$ and $f_{i+1,m}$ with $i > \frac{m}{2}$.

Besides the results on critical lengths, also many explicit formulations of Bernstein bases for specific generalized spaces have been proposed during the years. We can exploit this fact to improve the performance of the proposed approach for the construction of the ONTP basis of a generalized spline space. In fact, if the explicit expression of a Bernstein basis is already available for a certain space, we can choose such a basis as the QEC-system for the computation of transition functions. This entails an improvement of the numerical stability because of the well-known properties of the Bernstein basis.

As we have recalled above, in the case of an empty knot partition, the problem of determining the largest interval on which a given space admits the Bernstein basis has been addressed by introducing the notion of critical length for design and devising methods for the computation of its exact or approximated value. In this context, the approach based on the theory of weight functions is of little use, unlike what happens in the spline case.
However, it may be interesting to see an application of the construction of weight functions outlined in Algorithm 1 in the case of generalized polynomials, in order to explicitly derive the expression of the canonical basis of an EC-space.

Remark 5.13. (Canonical basis) The system of weight functions constructed as in Algorithm 1 can be exploited to provide the expression of the canonical basis of a given EC-space on $[a, b]$ in terms of transition functions:

$$\begin{aligned}
\psi_0(x) &= w_0(x) = 1,\\
\psi_1(x) &= w_0(x) \int_a^x w_1(\xi_1) d\xi_1 = \sum_{i=2}^{m} f_{i,m}(x),\\
&\vdots\\
\psi_r(x) &= w_0(x) \int_a^x w_1(\xi_1) \int_a^{\xi_1} \dots \int_a^{\xi_{r-1}} w_r(\xi_r) d\xi_r \dots d\xi_2 d\xi_1\\
&= \sum_{i=r+1}^{m} \binom{i-2}{r-1} f_{i,m}(x), \qquad r = 2, \dots, m-1,
\end{aligned}$$

where in particular $\psi_{m-1} = f_{m,m}$.

5.5 Numerical examples

This section presents some examples in which the existence of the ONTP basis is tested through the procedure described in Section 5.4.2. We remark that, although for completeness the figures will show also the transition functions together with the generated bases, in general the existence of the ONTP basis cannot be proved by a simple inspection of the transition functions. In fact, in order to produce an answer regarding the existence of the ONTP basis, the above procedure must be performed. However, in the case in which the transition functions are not monotonically increasing we can readily conclude that the considered space does not admit an ONTP basis.

Example 5.1. Let us consider the generalized spline space $S(\boldsymbol{\mathcal{U}}_5, \boldsymbol{M}, \boldsymbol{\Delta})$ on a uniform knot partition with intervals of length h, and where the EC-spaces have dimension $m = 5$ and are $\mathcal{U}_{i,5} = \text{span}\{1, \cos(\theta_i x) \cosh(\phi_i x), \cos(\theta_i x) \sinh(\phi_i x), \sin(\theta_i x) \cosh(\phi_i x), \sin(\theta_i x) \sinh(\phi_i x)\}$, with $\theta_i = 1$ and $\phi_i = \phi > 0$ for all i.

Letting h vary, the procedure presented earlier allows us to verify numerically that such a spline space admits the ONTP basis if $h < \pi$, as illustrated in Figure 5.2. This result agrees with the conditions presented in [BM14], where the considered space is labeled as "Case 9".

If we consider the more general setting of a non-uniform knot partition and also allow the parameters θ_i, ϕ_i to be possibly different on each interval, no closed-form conditions for the existence of the ONTP basis of the spline space are available. This lack of theoretical results reinforce the importance of the proposed numerical test as a way for checking whether the ONTP basis exists for a given spline space with specific sets of parameters and interval lengths. Figure 5.3 shows the transition functions in two cases that differ only in the lengths of two knot intervals. In one case, the test affirms that the ONTP basis exists, while in the other the transition functions are not monotonically increasing, the test fails to pass and allows us to conclude that the ONTP does not exist.

Example 5.2. In this example we consider on each knot interval I_i the space $\mathcal{U}_{i,6} = \text{span}\{1, t, (1-t)^n, t^n, \cos(\theta_i x), \sin(\theta_i x)\}$, with $x \in [x_i, x_{i+1}]$, $t := \frac{x-x_i}{x_{i+1}-x_i} \in [0,1]$, and $n \geqslant 3$, which is a QEC-space proper.
In the case of a uniform knot partition, fixing $\theta_i = 1$ for all i, the pass/fail machinery of the numerical test allows us to estimate the largest width for the intervals that, for any value of the exponent n, guarantees the existence of the ONTP basis of the spline space. In particular, the length of each interval must be smaller than π (see Figure 5.4).

As in the previous example, in the more general cases in which different interval lengths and/or parameters θ_i are allowed, the test provides us with either a positive or a negative answer regarding the existence of the ONTP basis in each specific situation.

Example 5.3. Recalling Definition 5.8 and Proposition 5.2, this example exhibits a peculiar case showing that the requirements for the existence of an ONTP basis are indeed stronger than the ones for having a B-spline basis.
In particular, in Figure 5.5 we consider a spline space that does not admit the ONTP basis, as the numerical procedure correctly determine, although the basis constructed by our approach (showed in Figure 5.5(a)) satisfies properties i)–iv) in Definition 5.8. On a closer inspection, one of the functions $\{w_j\}$ constructed as in Algorithm 1 is in fact found to be nonpositive, as we can see from Figure 5.5(h), thus the ONTP basis does not exist.

An illuminating way to explain this behavior is to consider knot insertion. In particular, Figure 5.5(i) shows the situation after the insertion of a knot in one of the intervals. It is now evident that the basis obtained in this case is no longer a B-spline basis, thus the initial basis does not meet all the conditions of Proposition 5.2 and is not ONTP.

Example 5.4. We consider the space $\mathcal{U}_3 = \text{span}\{1, \cos x, \sin x\}$, $x \in [0, h]$. It is a QEC-space on $[0, 2\pi)$, and the space obtained by differentiating its generators, namely $D\mathcal{U}_3 = \text{span}\{\cos x, \sin x\}$, is a QEC-space on $[0, \pi)$. Therefore, by Proposition 5.9, $\mathcal{U}_3$ possesses the Bernstein basis on any interval $[0, h]$ with $h < \pi$, and, recalling the definitions in Section 5.4.3, the values 2π and π are respectively the critical length and the critical length for design for this space.
As we have seen, the Bernstein basis can be constructed by exploiting the proposed approach, and it is showed in Figures 5.6(a) and 5.6(b). Moreover, if $h \in (\pi, 2\pi)$, the space does not admit the Bernstein basis and the proposed construction produces instead a Bernstein-like basis, up to a sign inversion of the functions that are negative (Figure 5.6(c)). For $h > 2\pi$, $\mathcal{U}_3$ is not a QEC-space and the generated set of functions is not even a Bernstein-like basis (Figure 5.6(d)).

In the spline setting, illustrated in Figure 5.7, on a uniform partition with intervals of length $h < \pi$ we obtain the ONTP basis, as expected. On the contrary, for larger intervals the spline space does not possess the ONTP basis.

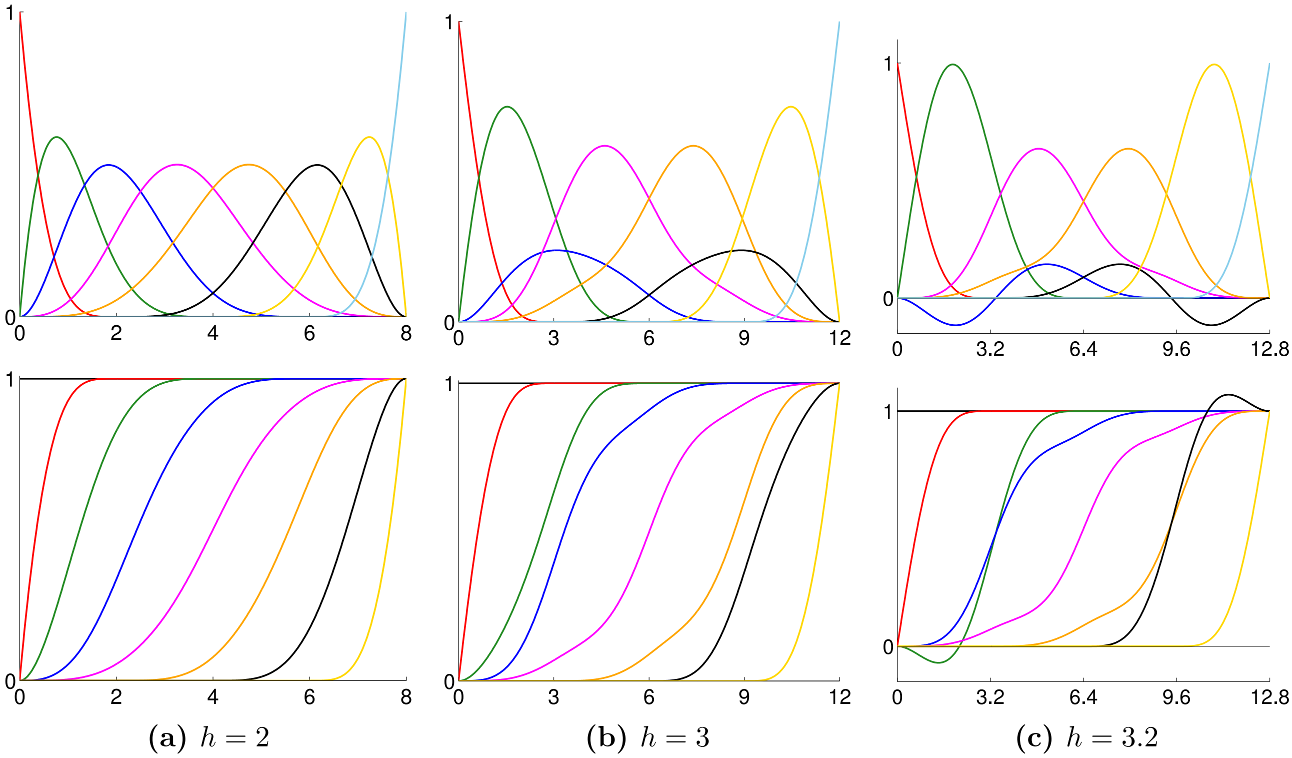


Figure 5.2: Top row: bases generated by our approach for the spline space $S(\boldsymbol{\mathcal{U}}_5, \boldsymbol{M}, \boldsymbol{\Delta})$ on a uniform partition with interval length h, where $\mathcal{U}_{i,5} = \text{span}\{1, \cos(\theta_i x)\cosh(\phi_i x), \cos(\theta_i x)\sinh(\phi_i x), \sin(\theta_i x)\cosh(\phi_i x), \sin(\theta_i x)\sinh(\phi_i x)\}$ with $\theta_i = 1$, $\phi_i = 0.5$ for all i. In particular, (a), (b) for $h < \pi$ the spline space admits the ONTP basis, while (c) for larger values of h the set of generated functions has no meaningful properties. Bottom row: related transition functions.

5.6 Computation and modeling in spaces of generalized splines

In this section we discuss some computational and modeling aspects that may be of interest from the point of view of applications. In particular, we make some remarks about knot insertion, we provide order elevation algorithms for generalized splines and corner cutting evaluation algorithms for generalized polynomials. Moreover, we summarize in an algorithmic form the main computational tools that can be drawn from the constructive approach, and finally we illustrate some application examples.

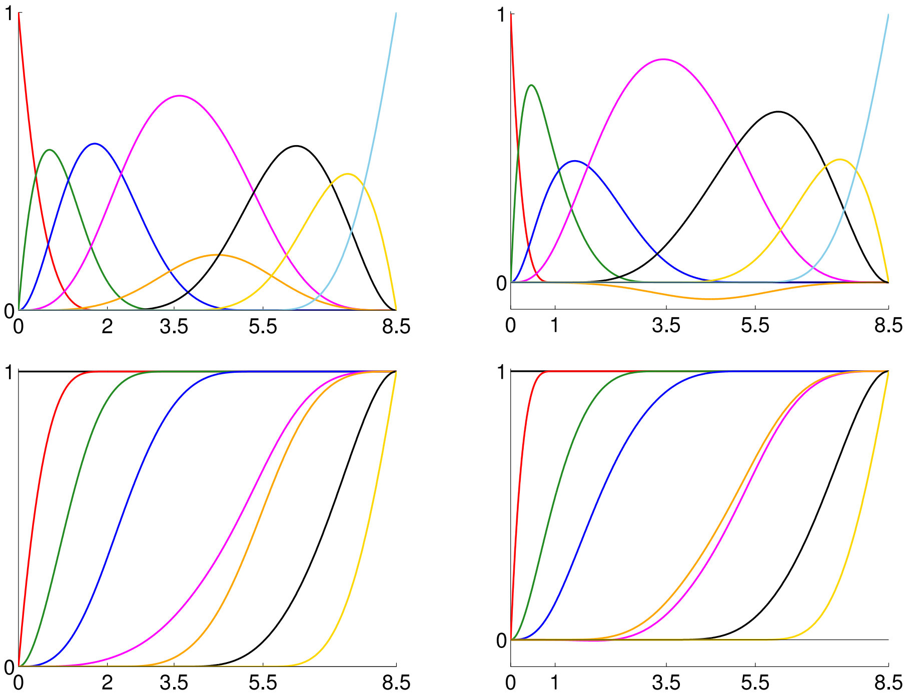
In the following, we will refer to the representation of a spline $s \in S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ in the B-spline basis $\{N_{i,m}, i = 1, \dots, m+K\}$ of $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$, namely

$$s(x) = \sum_{i=1}^{m+K} c_i \, N_{i,m}(x), \qquad x \in [a, b]. \tag{5.10}$$

5.6.1 Knot insertion in generalized spline spaces

In this section we give further insights concerning knot insertion in spaces of generalized splines. In particular, after some remarks on the knot insertion formula (5.7) presented in Section 5.3.1, we discuss some aspects to take into account when performing knot insertion in generalized spline spaces, and we illustrate them with the help of some examples.

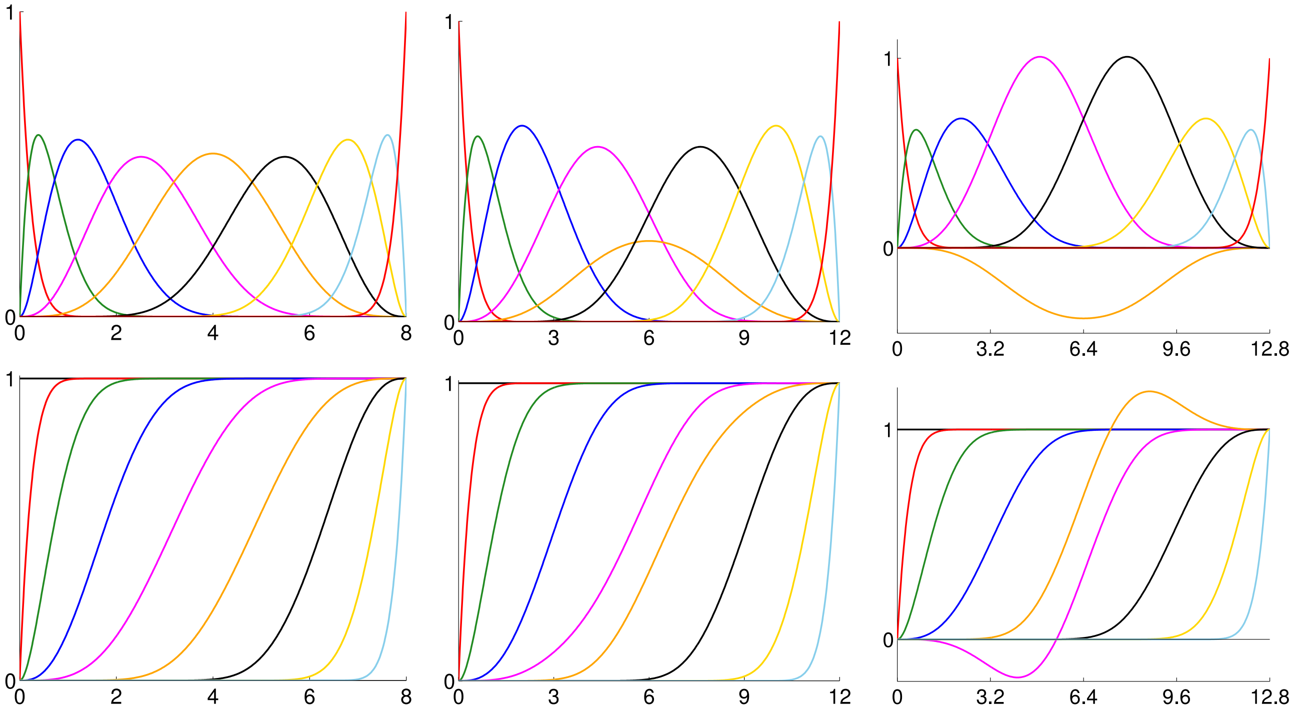
Remark 5.14. In (5.7) $\alpha_{i,m}$ is determined by computing the ratio of the derivatives of the



(a) Knot intervals of lengths $2, 1.5, 2, 3$.

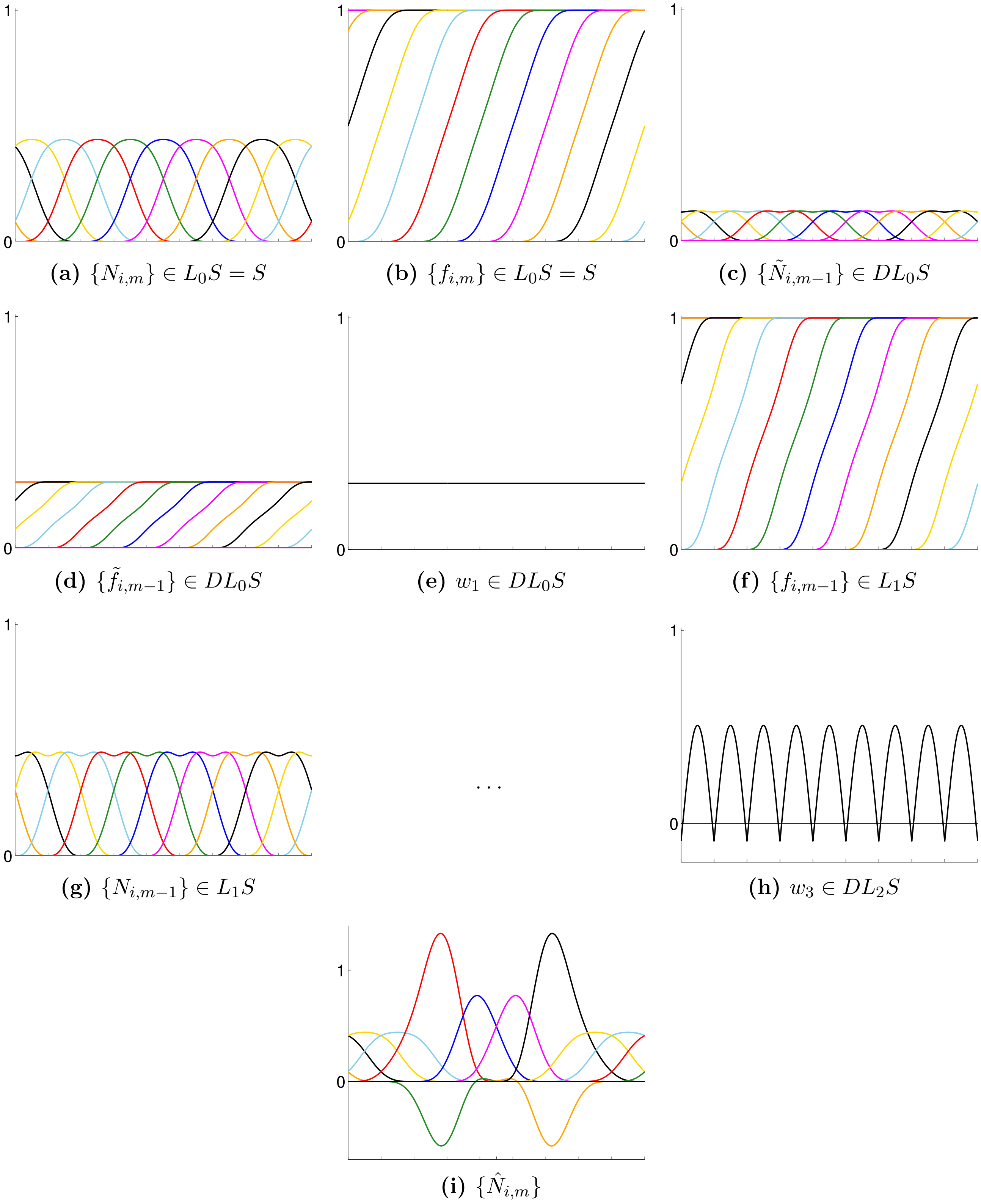
(b) Knot intervals of lengths $1, 2.5, 2, 3$.

Figure 5.3: Top row: bases generated by our approach for the spline space $S(\boldsymbol{\mathcal{U}}_5, \boldsymbol{M}, \boldsymbol{\Delta})$ on non-uniform knot partitions (a) $\boldsymbol{\Delta}^* = \{0, 0, 0, 0, 0, 2, 3.5, 5.5, 8.5, 8.5, 8.5, 8.5, 8.5\}$ and (b) $\boldsymbol{\Delta}^* = \{0, 0, 0, 0, 0, 1, , 3.5, 5.5, 8.5, 8.5, 8.5, 8.5, 8.5\}$, where for all i $\mathcal{U}_{i,5}$ is as for Figure 5.2. In case (a) the spline space admits the ONTP basis and it is the one showed. Conversely, in case (b) such a basis does not exist. In particular, in this case the basis obtained by the proposed construction is a B-spline-like basis, provided that the sign of the negative function is inverted. Bottom row: related transition functions.



(a) $h = 2$ **(b)** $h = 3$ **(c)** $h = 3.2$

Figure 5.4: Top row: bases generated by our approach for the spline space $S(\boldsymbol{\mathcal{U}}_6, \boldsymbol{M}, \boldsymbol{\Delta})$ on a uniform partition with interval length h, where $\mathcal{U}_{i,6} = \text{span}\{1, t, (1-t)^n, t^n, \cos(\theta_i x), \sin(\theta_i x)\}$, with $t := \frac{x - x_i}{x_{i+1} - x_i} \in [0, 1]$, $n = 8$, and $\theta_i = 1$ for all i. In particular, (a), (b) for $h < \pi$ the spline space admits the ONTP basis, while (c) for larger values of h the set of generated functions either is a B-spline-like basis (up to signs) or has no meaningful properties. Bottom row: related transition functions.



(a) $\{N_{i,m}\} \in L_0 S = S$

(b) $\{f_{i,m}\} \in L_0 S = S$

(c) $\{\tilde{N}_{i,m-1}\} \in DL_0 S$

(d) $\{\tilde{f}_{i,m-1}\} \in DL_0 S$

(e) $w_1 \in DL_0 S$

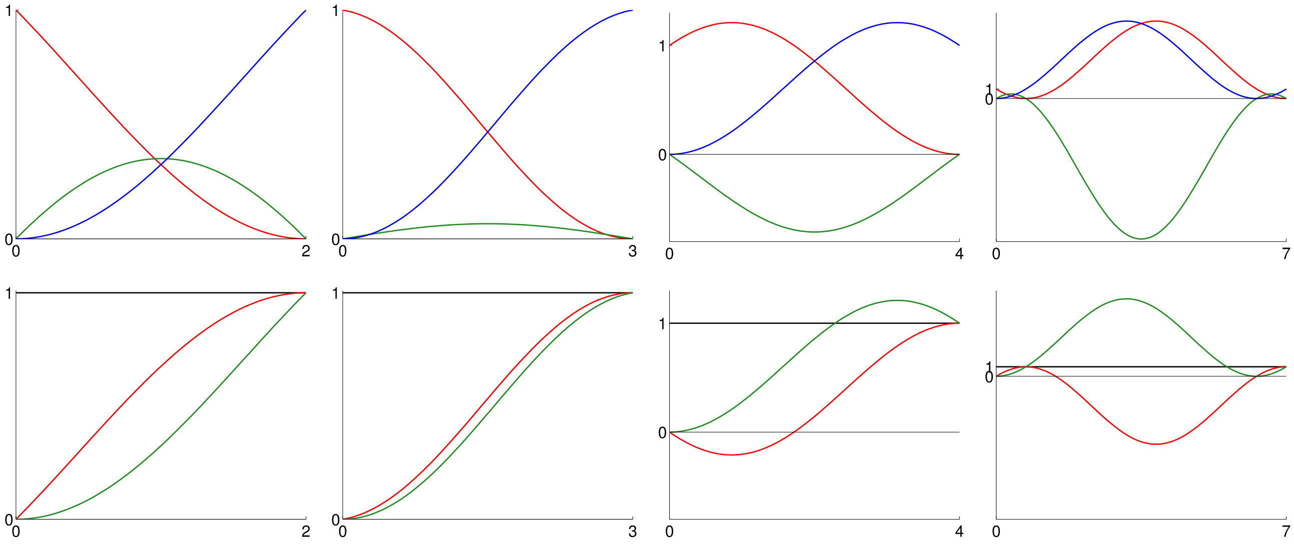
(f) $\{f_{i,m-1}\} \in L_1 S$

(g) $\{N_{i,m-1}\} \in L_1 S$

(h) $w_3 \in DL_2 S$

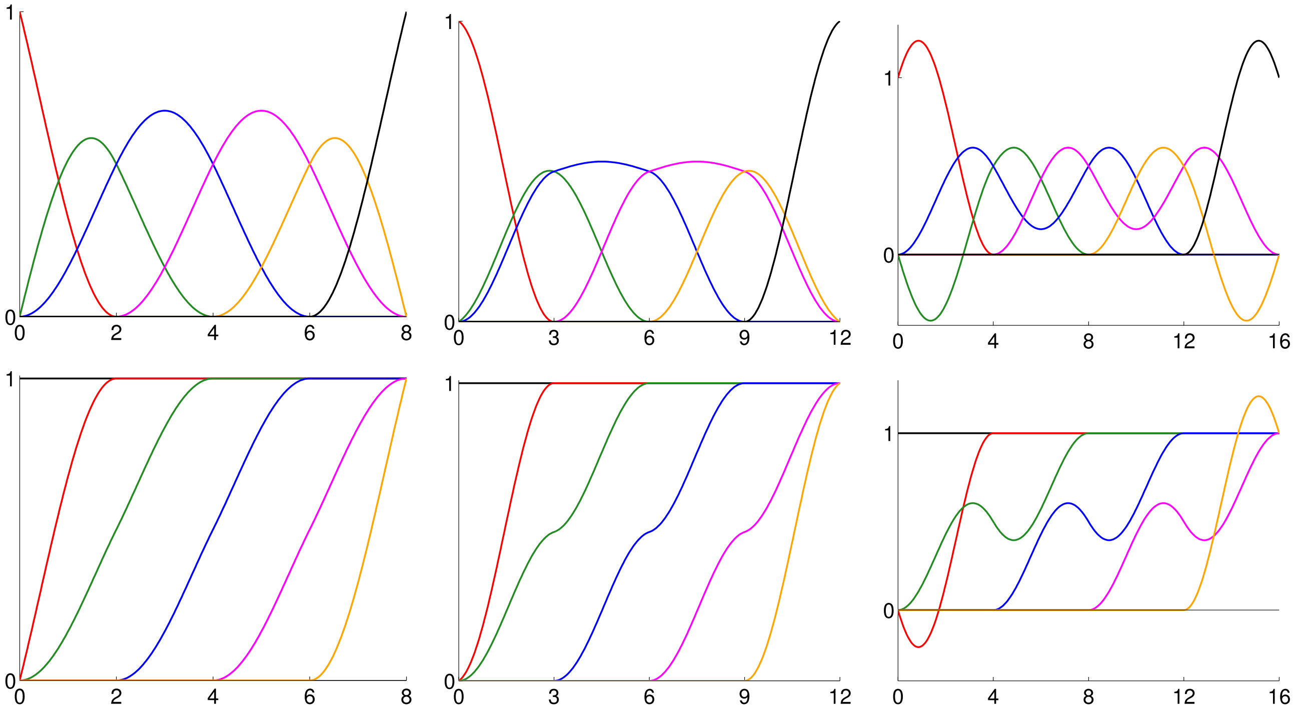
(i) $\{\hat{N}_{i,m}\}$

Figure 5.5: Example of a B-spline basis that is not ONTP. We consider the spline space $S := S(\boldsymbol{\mathcal{U}}_5, \boldsymbol{M}, \boldsymbol{\Delta})$ on a periodic uniform knot partition with intervals of length 3.5, and $\mathcal{U}_{i,5} = \text{span}\{1, x, x^2, \cos x, \sin x\}$ for all i. Referring to Algorithm 1 for the notation, (a)–(g) illustrate the first steps of the procedure for the construction of the sequence of functions $\{w_j\}$, exploited to check the existence of the ONTP basis in the given spline space. Since one of these functions, namely w_3, is found to be nonpositive (see (h)), we can conclude that the space does not admit an ONTP basis. Thus, the basis generated by our construction and showed in (a) is not ONTP, but only a B-spline basis. The fact that the space does not possess an ONTP basis can be inferred also by performing knot insertion in one interval (e.g., the central one) and observing that the basis generated on the finer partition is not a B-spline basis (see (i)), thus violating the requirements in Proposition 5.2.



(a) $h = 2$ **(b)** $h = 3$ **(c)** $h = 4$ **(d)** $h = 7$

Figure 5.6: Top row: bases generated by our approach for the space $\mathcal{U}_3 = \text{span}\{1, \cos x, \sin x\}$, with $x \in [0, h]$, in correspondence of different values of h. In particular, (a), (b) for $h < \pi$ the construction produces the Bernstein basis, while (c) for $h \in (\pi, 2\pi)$ it generates a Bernstein-like basis, provided that the sign of the negative function is inverted. (d) For $h > 2\pi$, the set of functions has no meaningful properties. Bottom row: related transition functions.



(a) $h = 2$ **(b)** $h = 3$ **(c)** $h = 4$

Figure 5.7: Top row: bases generated by our approach for the spline space $S(\boldsymbol{\mathcal{U}}_3, \boldsymbol{M}, \boldsymbol{\Delta})$, with $\mathcal{U}_{i,3} = \text{span}\{1, \cos x, \sin x\}$ for all i, and uniform knot partition, in correspondence of different values of the length h of the intervals. In particular, (a), (b) for $h < \pi$ the spline space admits the ONTP basis, while (c) for larger values of h the set of generated functions has no meaningful properties. Bottom row: related transition functions.

transition functions of the appropriate order $k^*_{i,i_1}+1$ such that they are not zero at x_{i_1}. Note that $f_{i,m}$ and $\hat{f}_{i,m}$ have at x_{i_1} the same order of continuity. The need for the right-hand derivative is explained by the fact that the considered order of differentiation is greater than the order of continuity of the transition functions.

Remark 5.15. Formula (5.7) provides a way for computing the coefficients $\alpha_{i,m}$ for all i. Note that, in the case of an EC-space, k^*_{i,i_1} is equal to k_{i,i_1}, so it can be determined easily from the multiplicity of x_{i_1}. On the contrary, in the case of a QEC-space proper, $k^*_{i,i_1} \geqslant k_{i,i_1}$ and its exact value depends on the specific space that is used.

Proposition 5.24. *Let*

$$s(x) = \sum_{i=1}^{m+K} c_i\, N_{i,m}(x) = \sum_{i=1}^{m+K+1} \hat{c}_i\, \hat{N}_{i,m}(x)$$

be a spline in $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) \subset \hat{S}(\hat{\boldsymbol{\mathcal{U}}}_m, \hat{\boldsymbol{M}}, \hat{\boldsymbol{\Delta}})$, *being the setting the same as in Proposition 5.12. Then, from Corollary 5.13 we easily obtain*

$$\hat{c}_i = \begin{cases} c_i, & i \leqslant \ell - m + 1, \\ \alpha_{i,m}\, c_i + (1-\alpha_{i,m})\, c_{i-1}, & \ell - m + 2 \leqslant i \leqslant \ell - r + 1, \\ c_{i-1}, & i \geqslant \ell - r + 2, \end{cases} \tag{5.11}$$

with $\alpha_{i,m}$, $i = \ell - m + 2, \ldots, \ell - r + 1$, *as in* (5.7), *and where* r *is the multiplicity of the inserted knot in* $\hat{\boldsymbol{\Delta}}^*$.

The given definition of a generalized spline space allows for a different QEC-space $\mathcal{U}_{i,m}$ to be present in each interval of the knot partition. Inserting a knot in a certain interval I_i entails redefining the QEC-systems that span each of the subintervals into which the original interval is split in such a way that the new spline space with the additional knot contains the initial spline space. One way to guarantee this inclusion for any kind of space consists in choosing the new QEC-systems to be equal to the parent one (i.e., the generators will be still defined on the coarser interval).

However, we observe that there exist interesting generalized spline spaces where this choice is not the only viable strategy for inserting a knot. This is the case of some splines in which the underlying QEC-spaces differ for a parameter only, such as, for example, the trigonometric splines [MP10], where the QEC-space associated with an interval $[x_i, x_{i+1}]$ is spanned by $\{1, \cos(\theta_i t), \sin(\theta_i t)\}$, with $t = x - x_i$ and $\theta_i \in \left(0, \frac{\pi}{x_{i+1}-x_i}\right)$. In such cases, depending on whether the functions in the QEC-space are invariant under scaling, two possibilities exist for choosing the QEC-spaces on the two subintervals, while guaranteeing that the initial spline space is contained in the new one:

1. we still use the QEC-space defined on the original interval, simply restricting it to either subinterval;

2. we define two new QEC-spaces and related QEC-systems, each of which is associated with a subinterval and such that they can reproduce the functions in the original QEC-system, thus guaranteeing the inclusion between the spline spaces.

The difference between the two ways of proceeding is substantial. In fact, in the first way the QEC-systems on the refined intervals will be equal to those of the parent coarse interval.

On the contrary, in the second way the new QEC-systems will be peculiar to the refined knot intervals. We remark that the latter way of proceeding is not always feasible, but when it is, it generates the same system of transition functions obtained through the former one.

Our distinction between the two kinds of knot insertion is motivated by the usual association of the notion of knot insertion with refinable spline spaces only. Nevertheless, this association is not mandatory, since there exist generalized spline spaces that, although not refinable, admit the knot insertion according to the first above strategy. Some examples of such spaces are $\text{span}\{1, t, (1-t)^{n_1}, t^{n_2}\}$ with $n_1, n_2 \geqslant 3$, and $\text{span}\left\{1, t, \frac{(1-t)^3}{1+(\mu-3)(1-t)t}, \frac{t^3}{1+(\nu-3)(1-t)t}\right\}$ with $\mu, \nu \geqslant 3$, where $t = \frac{x-x_i}{x_{i+1}-x_i} \in [0,1]$. Conversely, examples of generalized spline spaces where refinability holds are all those of mixed algebraic/trigonometric/hyperbolic splines.

The following example illustrates how to choose suitable QEC-spaces to perform knot insertion according to equation (5.7).

Example 5.5. Given a spline space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$, we want to insert a knot $\hat{t}$, $x_h \leqslant \hat{t} < x_{h+1}$, to generate the space $S(\hat{\boldsymbol{\mathcal{U}}}_m, \hat{\boldsymbol{M}}, \hat{\boldsymbol{\Delta}})$. According to Proposition 5.12, we need that the space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be included in $S(\hat{\boldsymbol{\mathcal{U}}}_m, \hat{\boldsymbol{M}}, \hat{\boldsymbol{\Delta}})$. This can be accomplished by suitably choosing the spaces $\hat{\mathcal{U}}_{h,m}$ and $\hat{\mathcal{U}}_{h+1,m}$, and setting $\hat{\mathcal{U}}_{i,m} = \mathcal{U}_{i,m}$ for any $i < h$ and $\hat{\mathcal{U}}_{i,m} = \mathcal{U}_{i-1,m}$ for any $i > h+1$.

a) We start by analyzing the case where $\mathcal{U}_{h,m} = \text{span}\{1, t, (1-t)^{n_{1,h}}, t^{n_{2,h}}\}$, with $n_{1,h}, n_{2,h} > 3$, $t = \frac{x-x_h}{x_{h+1}-x_h}$ and $x \in [x_h, x_{h+1})$. The only feasible way to perform knot insertion in this situation requires to set:

   - $\hat{\mathcal{U}}_{h,m} = \text{span}\{1, t, (1-t)^{n_{1,h}}, t^{n_{2,h}}\}$, with $x \in [x_h, \hat{t})$, $t = \frac{x-x_h}{x_{h+1}-x_h}$,
   - $\hat{\mathcal{U}}_{h+1,m} = \text{span}\{1, t, (1-t)^{n_{1,h}}, t^{n_{2,h}}\}$, with $x \in [\hat{t}, x_{h+1})$, $t = \frac{x-x_h}{x_{h+1}-x_h}$,

   and thus $n_{1,h}, n_{2,h}$ are the same as for $\mathcal{U}_{h,m}$. In fact this is the only choice that guarantees that the generators of $\mathcal{U}_{h,m}$ can be represented by the generators of $\hat{\mathcal{U}}_{h,m}$ and $\hat{\mathcal{U}}_{h+1,m}$ on the two intervals $[x_h, \hat{t})$ and $[\hat{t}, x_{h+1})$ respectively.

b) Consider now the case where $\mathcal{U}_{h,m} = \text{span}\{1, t, \cos(\theta_h t), \sin(\theta_h t)\}$, with $\theta_h \in (0, \frac{\pi}{x_{h+1}-x_h})$, $t = x - x_h$ and $x \in [x_h, x_{h+1})$. For this space we could proceed analogously as above, but since in this case the space is invariant under translation and scaling, we can also choose:

   - $\hat{\mathcal{U}}_{h,m} = \text{span}\{1, t, \cos(\theta_h t), \sin(\theta_h t)\}$, with $x \in [x_h, \hat{t})$, $t = x - x_h$,
   - $\hat{\mathcal{U}}_{h+1,m} = \text{span}\{1, t, \cos(\theta_h t), \sin(\theta_h t)\}$, with $x \in [\hat{t}, x_{h+1})$, $t = x - \hat{t}$.

   In particular, for a space where refinability holds, the two kinds of knot insertion lead to the same result.

5.6.2 Order elevation in generalized spline spaces

Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ and $S(\boldsymbol{\mathcal{U}}_{m+n}, \tilde{\boldsymbol{M}}, \boldsymbol{\Delta})$, $n \geqslant 1$, be two spline spaces such that $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta}) \subset S(\boldsymbol{\mathcal{U}}_{m+n}, \tilde{\boldsymbol{M}}, \boldsymbol{\Delta})$. Then we will show that there exists an order elevation algorithm in the sense that the B-spline basis $\{N_{i,m}\}$ of $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ can be expressed as a linear combination of the elements of the B-spline basis $\{N_{i,m+n}\}$ of $S(\boldsymbol{\mathcal{U}}_{m+n}, \tilde{\boldsymbol{M}}, \tilde{\boldsymbol{\Delta}})$, where $\tilde{\boldsymbol{M}}$ and $\tilde{\boldsymbol{\Delta}}$ are obtained from $\boldsymbol{M}$ and $\boldsymbol{\Delta}$ by simply increasing by n the multiplicity of each knot. In particular, the basic steps of the order elevation algorithm are:

1. subdivide the spline into pieces expressed in terms of the Bernstein basis;

2. apply the order elevation to each piece in the Bernstein basis;
3. remove the redundant knots.

The above steps can be applied repeatedly starting from a space of dimension m up to a space of dimension $m + n$. As regards the implementation of step 1, we only need to recall that, for any generalized spline, the Bernstein-type representation can be achieved by performing a suitable number of knot insertions to obtain a partition with knots of multiplicity m. Thus, in the following, we will focus on the second step of the algorithm, which requires providing a relation between the Bernstein bases of order m and order $m+n$. In this regard, $n = 1, 2$ are the essential cases to consider. In particular, several reasons motivate the need for a direct formula to accomplish elevation by two orders at once. In fact, not only there exist Bernstein bases of QEC-spaces of dimension m that cannot be expressed via convex combinations of elements of Bernstein bases for QEC-spaces of dimension $m + 1$ [CMSR98], but also 2-order elevation turns out to be useful when working with spaces containing trigonometric (and/or hyperbolic) functions, because such functions are coupled as elements of the null space of differential operators. In the latter spaces, besides being computationally less efficient, proceeding through 1-order elevations could also raise some issues, as illustrated by the following example.

Example 5.6. We want to order-elevate the EC-space $\mathrm{span}\{1, t\}$ to $\mathrm{span}\{1, t, \cos(\theta t), \sin(\theta t)\}$, where $t \in [0, 1]$ and θ has a fixed value in $(0, \pi)$. We shall observe that, for values of $\theta \geqslant \frac{\pi}{2}$, the strategy of performing two successive elevations by one order is not feasible if we start by elevating $\mathrm{span}\{1, t\}$ to $\mathrm{span}\{1, t, \sin(\theta t)\}$, because the latter space is an EC-space in the range $\theta \in (0, \frac{\pi}{2})$ only. Conversely, the strategy of elevating to $\mathrm{span}\{1, t, \cos(\theta t)\}$ first, and successively to the target space $\mathrm{span}\{1, t, \cos(\theta t), \sin(\theta t)\}$ is feasible. This example illustrates that proceeding through successive elevations by one order is potentially troublesome when the target space contains an additional couple of trigonometric functions (or hyperbolic likewise).

The following result holds in the case $n = 1$, where we denote by $\mathcal{U}_m$ an m-dimensional QEC-space.

Proposition 5.25. *Let $\{B_{i,m}, i = 1, \dots, m\}$ and $\{B_{i,m+1}, i = 1, \dots, m + 1\}$ be the Bernstein bases in $[a, b]$ for the generalized spaces $\mathcal{U}_m$ and $\mathcal{U}_{m+1}$ with $\mathcal{U}_m \subset \mathcal{U}_{m+1}$. Then there exist the coefficients $\gamma_1, \dots, \gamma_{m+1}$, with $0 \leqslant \gamma_i \leqslant 1$, $i = 1, \dots, m + 1$, such that*

$$B_{i,m}(x) = \gamma_i\, B_{i,m+1}(x) + (1 - \gamma_{i+1})\, B_{i+1,m+1}(x), \qquad i = 1, \dots, m, \quad x \in [a, b]. \tag{5.12}$$

In particular,

$$\gamma_i = \begin{cases} 1, & i = 1, \\ \dfrac{D^{i-1} f_{i,m}(a)}{D^{i-1} f_{i,m+1}(a)}, & i = 2, \dots, m, \\ 0, & i = m + 1, \end{cases} \tag{5.13}$$

where $f_{i,m}$ and $f_{i,m+1}$ are the transition functions that define respectively the $\{B_{i,m}\}$ and the $\{B_{i,m+1}\}$.

Proof. From the conditions (5.1) that define the transition functions, the generalized Bernstein basis functions $B_{i,m}$, $i = 1, \dots, m$, have zeros at the endpoints a and b with multiplicities $i - 1$ and $m - i$ respectively (as is well known in the polynomial case). Hence, the relation between the bases of orders m and $m + 1$ must have the form

$$B_{i,m}(x) = \gamma_i\, B_{i,m+1}(x) + \delta_{i+1}\, B_{i+1,m+1}(x), \qquad x \in [a, b].$$

In fact, if $B_{i,m}$ were a combination of additional basis functions, it could not have a and b as zeros of the above-mentioned multiplicities.

By construction we know that at a $B_{1,m}$ and $B_{1,m+1}$ are 1 while $B_{2,m+1}$ is 0, thus $\gamma_1 = 1$. A similar reasoning at b leads to $\delta_{m+1} = 1$.

In addition, we know that the bases are positive for all $x \in (a,b)$ and that $D^{i-1}B_{i,m}(a) > 0$ for $i = 2, \dots, m$, therefore $\gamma_i, \delta_{i+1} \geqslant 0$. Since

$$1 \equiv \sum_{i=1}^{m} B_{i,m}(x) = \sum_{i=1}^{m} (\gamma_i \, B_{i,m+1}(x) + \delta_{i+1} \, B_{i+1,m+1}(x)) = \sum_{i=1}^{m+1} B_{i,m+1}(x),$$

for all $x \in [a,b]$, we have $\gamma_1 = \delta_{m+1} = 1$ and $\gamma_i + \delta_i = 1$ for $i = 2, \dots, m$, proving that

$$\gamma_i = \begin{cases} 1, & i = 1, \\ \dfrac{D^{i-1}B_{i,m}(a)}{D^{i-1}B_{i,m+1}(a)}, & i = 2, \dots, m, \\ 0, & i = m+1, \end{cases}$$

from which (5.13) immediately follows. □

Corollary 5.26. *Let* $p(x) = \sum_{i=1}^{m} c_i \, B_{i,m}(x) = \sum_{i=1}^{m+1} \tilde{c}_i \, B_{i,m+1}(x)$, $x \in [a,b]$. *Then from equation* (5.12) *we easily obtain*

$$\tilde{c}_i = \begin{cases} c_1, & i = 1, \\ \gamma_i \, c_i + (1 - \gamma_i) \, c_{i-1}, & i = 2, \dots, m, \\ c_m, & i = m+1, \end{cases} \tag{5.14}$$

with γ_i, $i = 2, \dots, m$, *as in* (5.13).

Elevation by two orders can be performed as stated in the following proposition.

Proposition 5.27. *Let* $\{B_{i,m}, i = 1, \dots, m\}$ *and* $\{B_{i,m+2}, i = 1, \dots, m+2\}$ *be the Bernstein bases in* $[a,b]$ *for the generalized spaces* $\mathcal{U}_m$ *and* $\mathcal{U}_{m+2}$ *with* $\mathcal{U}_m \subset \mathcal{U}_{m+2}$. *Then there exist the coefficients* $\gamma_1, \dots, \gamma_{m+2}$ *and* $\delta_2, \dots, \delta_{m+2}$, *with* $0 \leqslant \gamma_i \leqslant 1$, $i = 1, \dots, m+2$, $0 \leqslant \delta_i \leqslant 1$, $i = 2, \dots, m+2$, *such that for all* $x \in [a,b]$

$$B_{i,m}(x) = \gamma_i \, B_{i,m+2}(x) + \delta_{i+1} \, B_{i+1,m+2}(x) + (1 - \gamma_{i+2} - \delta_{i+2}) \, B_{i+2,m+2}(x), \quad i = 1, \dots, m. \tag{5.15}$$

In particular,

$$\gamma_i = \begin{cases} 1, & i = 1, \\ \dfrac{D^{i-1}f_{i,m}(a)}{D^{i-1}f_{i,m+2}(a)}, & i = 2, \dots, m, \\ 0, & i = m+1, m+2, \end{cases} \tag{5.16a}$$

$$\delta_{i+1} = \begin{cases} 1 - \gamma_2, & i = 1, \\ \gamma_i - \gamma_{i+1} + \dfrac{D^i f_{i,m}(a) - \gamma_i D^i f_{i,m+2}(a)}{D^i f_{i+1,m+2}(a)}, & i = 2, \dots, m, \\ 0, & i = m+1, \end{cases} \tag{5.16b}$$

where $f_{i,m}$ *and* $f_{i,m+2}$ *are the transition functions that define respectively the* $\{B_{i,m}\}$ *and the* $\{B_{i,m+2}\}$.

Proof. From the conditions that define the transition functions, the generalized Bernstein basis functions $B_{i,m}$, $i = 1, \ldots, m$, are zero at the endpoints a and b with multiplicities $i-1$ and $m-i$ respectively (as is well known in the polynomial case). Hence, the relation between the bases of orders m and $m+2$ must have the form

$$B_{i,m}(x) = \gamma_i \, B_{i,m+2}(x) + \delta_{i+1} \, B_{i+1,m+2}(x) + \epsilon_{i+2} \, B_{i+2,m+2}(x), \qquad x \in [a,b].$$

In fact, if $B_{i,m}$ were a combination of additional basis functions, it could not have a and b as zeros of the above-mentioned multiplicities.

By construction we know that at a $B_{1,m}$ and $B_{1,m+2}$ are 1 while $B_{2,m+2}$ and $B_{3,m+2}$ are 0, thus $\gamma_1 = 1$. A similar reasoning at b leads to $\epsilon_{m+2} = 1$.

In addition, we know that the bases are positive for all $x \in (a,b)$ and that $D^{i-1}B_{i,m}(a) > 0$ for $i = 2, \ldots, m$, therefore $\gamma_i, \delta_{i+1}, \epsilon_{i+2} \geqslant 0$. Since

$$1 \equiv \sum_{i=1}^{m} B_{i,m}(x) = \sum_{i=1}^{m} (\gamma_i \, B_{i,m+2}(x) + \delta_{i+1} \, B_{i+1,m+2}(x) + \epsilon_{i+2} \, B_{i+2,m+2}(x)) = \sum_{i=1}^{m+2} B_{i,m+2}(x),$$

for all $x \in [a,b]$, we have $\gamma_1 = \epsilon_{m+2} = 1$, $\gamma_2 + \delta_2 = 1$, $\delta_{m+1} + \epsilon_{m+1} = 1$, and $\gamma_i + \delta_i + \epsilon_i = 1$, for $i = 3, \ldots, m$, proving that

$$\gamma_i = \begin{cases} 1, & i = 1, \\ \dfrac{D^{i-1}B_{i,m}(a)}{D^{i-1}B_{i,m+2}(a)}, & i = 2, \ldots, m, \\ 0, & i = m+1, m+2, \end{cases}$$

$$\delta_{i+1} = \begin{cases} 1 - \gamma_2, & i = 1, \\ \dfrac{D^{i}B_{i,m}(a) - \gamma_i D^{i}B_{i,m+2}(a)}{D^{i}B_{i+1,m+2}(a)}, & i = 2, \ldots, m, \\ 0, & i = m+1, \end{cases}$$

from which (5.16a) and (5.16b) follow. □

Corollary 5.28. *Let* $p(x) = \sum_{i=1}^{m} c_i \, B_{i,m}(x) = \sum_{i=1}^{m+2} \tilde{c}_i \, B_{i,m+2}(x)$, $x \in [a,b]$. *Then from equation* (5.15) *we easily obtain*

$$\tilde{c}_i = \begin{cases} c_1, & i = 1, \\ \gamma_2 \, c_2 + (1 - \gamma_2) \, c_1, & i = 2, \\ \gamma_i \, c_i + \delta_i \, c_{i-1} + (1 - \gamma_i - \delta_i) \, c_{i-2}, & i = 3, \ldots, m, \\ \delta_{m+1} \, c_m + (1 - \delta_{m+1}) \, c_{m-1}, & i = m+1, \\ c_m, & i = m+2, \end{cases} \tag{5.18}$$

with γ_i, $i = 2, \ldots, m$, *as in* (5.16a) *and* δ_i, $i = 3, \ldots, m+1$, *as in* (5.16b).

5.6.3 Explicit formulae of basis functions and corner cutting algorithms

As we have already anticipated, in Section 5.6.5 we will see that our approach can be simply translated into numerical procedures for the computation in spaces of generalized splines. As an alternative, it is interesting to exploit the proposed approach to work out through symbolic computation the explicit expressions of the basis functions and corner cutting algorithms for stable evaluation and splitting of generalized spline curves. Although our discussion is confined here to some relevant examples, we believe that this aspect of our approach is of general interest, because it provides a starting point for both studying many other particular cases and tackling the wider problem of devising general formulae for whole classes of spaces.

In this section, by direct application of the constructive method presented in Section 5.3, we will easily derive the symbolic expression of the Bernstein and B-spline (ONTP) bases for spaces that are among the most useful for design and widely considered in the literature. Moreover, we will investigate the existence of corner cutting algorithms via repeated knot insertion. Such algorithms perform a finite number of iterations, where each step consists of a convex combinations of control points. This feature guarantees stability of the algorithm and also offers an interesting geometric interpretation. In fact, while evaluating a curve at a given parametric value, these algorithms generate as a by-product the splitting of the curve at that parameter, by providing the control polygons of the two parts. The most famous examples of corner cutting algorithms are those by de Casteljau and the de Boor, which respectively provide a stable evaluation procedure for polynomials expressed in the Bernstein basis and polynomial B-splines. We will thus refer also to the analogous methods for generalized splines as de Casteljau and de Boor algorithms.

5.6.3.1 Bernstein bases

Proposition 5.29. *Let us consider the space* $\mathcal{U}_3 = \operatorname{span}\{1, \cos(\theta t), \sin(\theta t)\}$ *with* $t \in [0,1]$ *and* $\theta \in (0,\pi)$. *The Bernstein basis functions for this space are:*

$$\begin{aligned}
B_{1,3}(t) &= B_{3,3}(1-t) = \frac{1-\cos(\theta(1-t))}{1-\cos\theta} = \frac{\sin^2\left(\frac{\theta}{2}(1-t)\right)}{\sin^2\left(\frac{\theta}{2}\right)},\\
B_{2,3}(t) &= 2\frac{\cos\left(\frac{\theta}{2}\right)}{\sin^2\left(\frac{\theta}{2}\right)} \sin\left(\frac{\theta}{2}(1-t)\right) \sin\left(\frac{\theta}{2}t\right),\\
B_{3,3}(t) &= \frac{1-\cos(\theta t)}{1-\cos\theta} = \frac{\sin^2\left(\frac{\theta}{2}t\right)}{\sin^2\left(\frac{\theta}{2}\right)},
\end{aligned}$$

or, expressed in an alternative, more compact form:

$$\begin{pmatrix} B_{1,3}(t) \\ B_{2,3}(t) \\ B_{3,3}(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ B_{3,3}(1-t) \\ B_{3,3}(t) \end{pmatrix}.$$

Proposition 5.30. *Let us consider the following spaces of the form* $\mathcal{U}_4 = \operatorname{span}\{1, t, u(t), v(t)\}$ *with* $t \in [0,1]$*:*

A) $u(t) = \cos(\theta t)$ *and* $v(t) = \sin(\theta t)$*, with* $\theta \in (0, 2\pi)$*;*

B) $u(t) = (1-t)^n$ *and* $v(t) = t^n$*, with* $n \geqslant 3$*;*

C) $u(t) = \frac{(1-t)^3}{1+(h-3)(1-t)t}$ *and* $v(t) = \frac{t^3}{1+(h-3)(1-t)t}$*, with* $h \geqslant 3$.

The Bernstein basis functions for the considered spaces are respectively:

A)

$$\begin{aligned}
B_{1,4}(t) &= B_{4,4}(1-t) = \frac{\theta(1-t)-\sin(\theta(1-t))}{\theta-\sin\theta} \\
&= \frac{\theta-\theta t-\sin\theta\cos(\theta t)+\cos\theta\sin(\theta t)}{\theta-\sin\theta}, \\
B_{2,4}(t) &= B_{3,4}(1-t) \\
&= \frac{1}{F(t)}(1-\cos\theta)\left(\theta\cos\theta-\sin\theta+\theta(1-\cos\theta)t+(\sin\theta-\theta\cos\theta)\cos(\theta t)\right. \\
&\quad \left.+(1-\cos\theta-\theta\sin\theta)\sin(\theta t)\right), \\
B_{3,4}(t) &= \frac{1}{F(t)}(1-\cos\theta)\left(\theta-\sin\theta-\theta(1-\cos\theta)t-(\theta-\sin\theta)\cos(\theta t)\right. \\
&\quad \left.+(1-\cos\theta)\sin(\theta t)\right) \\
&= \frac{1}{F(t)}(1-\cos\theta)\left((\theta-\sin\theta)(1-\cos(\theta t))-(1-\cos\theta)(\theta t-\sin(\theta t))\right), \\
B_{4,4}(t) &= \frac{\theta t-\sin(\theta t)}{\theta-\sin\theta},
\end{aligned}$$

with $F(t) := (\theta-\sin\theta)(2(1-\cos\theta)-\theta\sin\theta)$*;*

B)

$$\begin{aligned}
B_{1,4}(t) &= B_{4,4}(1-t) = (1-t)^n, \\
B_{2,4}(t) &= B_{3,4}(1-t) = \frac{1}{n-2}\left(-1+n(1-t)-(n-1)(1-t)^n+t^n\right) \\
&= \frac{1}{n-2}\left((n-1)(1-t)-t-(n-1)(1-t)^n+t^n\right), \\
B_{3,4}(t) &= \frac{1}{n-2}\left(-1+nt+(1-t)^n-(n-1)t^n\right) \\
&= \frac{1}{n-2}\left(-(1-t)+(n-1)t+(1-t)^n-(n-1)t^n\right), \\
B_{4,4}(t) &= t^n;
\end{aligned}$$

C)

$$\begin{aligned}
B_{1,4}(t) &= B_{4,4}(1-t) = \frac{(1-t)^3}{1+(h-3)(1-t)t}, \\
B_{2,4}(t) &= B_{3,4}(1-t) = \frac{h(1-t)^2t}{1+(h-3)(1-t)t}, \\
B_{3,4}(t) &= \frac{h(1-t)t^2}{1+(h-3)(1-t)t}, \\
B_{4,4}(t) &= \frac{t^3}{1+(h-3)(1-t)t}.
\end{aligned}$$

Alternatively, these Bernstein basis functions can be expressed in the following compact form:

$$\begin{pmatrix} B_{1,4}(t) \\ B_{2,4}(t) \\ B_{3,4}(t) \\ B_{4,4}(t) \end{pmatrix} = \frac{1}{\sigma+\tau}\begin{pmatrix} 0 & 0 & \sigma+\tau & 0 \\ \sigma & \tau & -\sigma & -\tau \\ \tau & \sigma & -\tau & -\sigma \\ 0 & 0 & 0 & \sigma+\tau \end{pmatrix}\begin{pmatrix} 1-t \\ t \\ B_{4,4}(1-t) \\ B_{4,4}(t) \end{pmatrix},$$

where σ, τ, and $B_{4,4}$ are given by:

A) $\sigma = \theta\cos\theta - \sin\theta$, $\tau = \theta - \sin\theta$, *and* $B_{4,4}(t) = \frac{\theta t - \sin(\theta t)}{\theta - \sin\theta}$;

B) $\sigma = n-1$, $\tau = -1$, *and* $B_{4,4}(t) = t^n$;

C) $\sigma = h-1$, $\tau = -1$, *and* $B_{4,4}(t) = \frac{t^3}{1+(h-3)(1-t)t}$.

Proposition 5.31. *Let us consider the space* $\mathcal{U}_5 = \mathrm{span}\,\{1, t, t^2, \cos(\theta t), \sin(\theta t)\}$ *with* $t \in [0,1]$ *and* $\theta \in (0, 2\pi)$. *The Bernstein basis functions for this space are:*

$$\begin{aligned}
B_{1,5}(t) &= B_{5,5}(1-t),\\
B_{2,5}(t) &= B_{4,5}(1-t),\\
B_{3,5}(t) &= \frac{\theta\left(-\sin\theta - \theta(1-t)t(1-\cos\theta) + \sin(\theta t) + \sin(\theta(1-t))\right)}{\left(\theta\cos\left(\frac{\theta}{2}\right) - 2\sin\left(\frac{\theta}{2}\right)\right)^2},\\
B_{4,5}(t) &= \frac{(1-\cos\theta)\left(2-\theta^2t^2\right) + 2(\theta(\theta t - \sin(\theta t) - t\sin\theta) + \cos(\theta(1-t)) - \cos(\theta t))}{2\left(\theta\cos\left(\frac{\theta}{2}\right) - 2\sin\left(\frac{\theta}{2}\right)\right)^2} + B_{5,5}(t),
\end{aligned}$$

$$B_{5,5}(t) = \frac{2(1-\cos(\theta t)) - \theta^2t^2}{2(1-\cos\theta) - \theta^2},$$

or, expressed in a more compact form:

$$\begin{pmatrix} B_{1,5}(t)\\ B_{2,5}(t)\\ B_{3,5}(t)\\ B_{4,5}(t)\\ B_{5,5}(t) \end{pmatrix} = C \begin{pmatrix} (1-t)^2\\ 2(1-t)t\\ t^2\\ B_{5,5}(1-t)\\ B_{5,5}(t) \end{pmatrix},$$

where, setting $\lambda := \frac{\theta(\theta-\sin\theta)}{2(\cos\theta-1)+\theta\sin\theta}$, $\sigma := \frac{(\theta-\sin\theta)\left(\left(2-\theta^2\right)\cos\theta+2\theta\sin\theta-2\right)}{2\sin\theta\left(\theta\cos\left(\frac{\theta}{2}\right)-2\sin\left(\frac{\theta}{2}\right)\right)^2}$, $\tau := \sigma + \lambda$, *and* $\upsilon := \frac{\tan\frac{\theta}{2}}{\theta}\lambda$, *the matrix C is*

$$C := \begin{pmatrix} 0 & 0 & 0 & 1 & 0\\ \sigma & \upsilon & \tau & -\sigma & -\tau\\ \tau\left(\frac{1}{\upsilon}-2\right) & 1-2\upsilon & \tau\left(\frac{1}{\upsilon}-2\right) & -\tau\left(\frac{1}{\upsilon}-2\right) & -\tau\left(\frac{1}{\upsilon}-2\right)\\ \tau & \upsilon & \sigma & -\tau & -\sigma\\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

5.6.3.2 De Casteljau algorithms

For any given Bernstein basis of an m-dimensional QEC-space $\mathcal{U}_m$ we can provide a de Casteljau algorithm for the stable evaluation of the associated Bézier representation. This algorithm can be sketched in the following form, where c_i, $i = 1, \ldots, m$, denote the coefficients of the Bézier representation in the Bernstein basis of $\mathcal{U}_m$, and $\bar{t} \in [0,1]$ is the evaluation parameter. The additional quantities ω_j^ℓ, $\ell = 1, \ldots, m-1$, $j = 1, \ldots, m-\ell$, that appear in the expressions depend on the value $\bar{t}$ and are obtained by considering suitable ratios of transition functions defined on the different knot partitions that underlie each step of the algorithm:

$$\omega_j^\ell = \frac{D_+^{k_{j,j_1}^*+1} f_{j,m}(x_{j_1})}{D_+^{k_{j,j_1}^*+1} \hat{f}_{j,m}(x_{j_1})},$$

where $\{\hat{f}_{j,m}\}$ are the transition functions defined on the extended partition obtained after the ℓth insertion of $\bar{t}$, and k^*_{i,i_1} is determined as for (5.7).

```
**Algorithm 2** (de Casteljau algorithm).
Input: $\bar{t} \in [0,1]$;
        $c_i^0 := c_i,\ i = 1,\dots,m$;
        $\omega_j^\ell,\ \ell = 1,\dots,m-1,\ j = 1,\dots,m-\ell$;
for $\ell = 1,\dots,m-1$
  for $j = 1,\dots,m-\ell$
    $c_j^\ell = (1-\omega_j^\ell)\, c_j^{\ell-1} + \omega_j^\ell\, c_{j+1}^{\ell-1}$;
Output: $c_1^{m-1}$, the value of the Bézier representation at $\bar{t}$.
```

Let us observe that the above algorithm can be compactly expressed in the following matrix form:

$$\Omega_{m-1}\,\Omega_{m-2}\cdots\Omega_1 \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{pmatrix},$$

where

$$\Omega_\ell := \begin{pmatrix} 1-\omega_1^\ell & \omega_1^\ell & & & \\ & 1-\omega_2^\ell & \omega_2^\ell & & \\ & & \ddots & \ddots & \\ & & & 1-\omega_{m-\ell}^\ell & \omega_{m-\ell}^\ell \end{pmatrix} \in \mathbb{R}^{(m-\ell)\times(m-\ell+1)}.$$

In the following, we provide the explicit expressions of the coefficients ω_j^ℓ for some QEC-spaces:

- $\operatorname{span}\{1,\cos(\theta t),\sin(\theta t)\}$ with $\theta \in (0,\pi)$;
- $\operatorname{span}\{1,t,u(t),v(t)\}$, where either $u(t) = \cos(\theta t), v(t) = \sin(\theta t)$ with $\theta \in (0,2\pi)$, or u, v as in [Cos00] or as in [DG85].

In all cases, we will denote by $\{B_{i,m}\}$ the Bernstein basis of the considered space and deal with functions $p(t) = \sum_{i=1}^m c_i\, B_{i,m}(t)$ expressed in such a basis with coefficients $\{c_i, i = 1,\dots,m\}$.

Figure 5.8 shows some examples of the application of de Casteljau algorithms for the evaluation of generalized Bézier curves built on the above-mentioned spaces and in space $\operatorname{span}\{1,t,t^2,\cos(\theta t),\sin(\theta t)\}$ with $\theta \in (0,2\pi)$.

Proposition 5.32. *Let us consider the space* $\operatorname{span}\{1,\cos(\theta t),\sin(\theta t)\}$ *with* $\theta \in (0,\pi)$ *and* $t \in [0,1]$. *Then, for any* $\bar{t} \in [0,1]$,

$$p(\bar{t}) = \begin{pmatrix} 1-\omega_1^2 & \omega_1^2 \end{pmatrix} \begin{pmatrix} 1-\omega_1^1 & \omega_1^1 & 0 \\ 0 & 1-\omega_2^1 & \omega_2^1 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix},$$

where

$$\omega_1^1 = \frac{\cos\frac{\theta}{2}\sin\left(\frac{\theta}{2}\bar{t}\right)}{\sin\frac{\theta}{2}\cos\left(\frac{\theta}{2}\bar{t}\right)} = 1 - \frac{2\sin\left(\frac{\theta}{2}(1-\bar{t})\right)}{\sin\theta\cos\left(\frac{\theta}{2}(1-\bar{t})\right) + (1-\cos\theta)\sin\left(\frac{\theta}{2}(1-\bar{t})\right)},$$

$$\omega_2^1 = \frac{2\sin\left(\frac{\theta}{2}\bar{t}\right)}{\sin\theta\cos\left(\frac{\theta}{2}\bar{t}\right) + (1-\cos\theta)\sin\left(\frac{\theta}{2}\bar{t}\right)},$$

$$\omega_1^2 = \frac{\sin\left(\frac{\theta}{2}\bar{t}\right)\cos\left(\frac{\theta}{2}(1-\bar{t})\right)}{\sin\frac{\theta}{2}} = \sin\left(\frac{\theta}{2}\bar{t}\right)\left(\sin\left(\frac{\theta}{2}\bar{t}\right) + \frac{\cos\frac{\theta}{2}}{\sin\frac{\theta}{2}}\cos\left(\frac{\theta}{2}\bar{t}\right)\right).$$

Proposition 5.33. *Let us consider the space* $\operatorname{span}\{1, t, \cos(\theta t), \sin(\theta t)\}$ *with* $\theta \in (0, 2\pi)$ *and* $t \in [0,1]$. *Then, for any* $\bar{t} \in [0,1]$,

$$p(\bar{t}) = \begin{pmatrix} 1-\omega_1^3 & \omega_1^3 \end{pmatrix} \begin{pmatrix} 1-\omega_1^2 & \omega_1^2 & 0 \\ 0 & 1-\omega_2^2 & \omega_2^2 \end{pmatrix} \begin{pmatrix} 1-\omega_1^1 & \omega_1^1 & 0 & 0 \\ 0 & 1-\omega_2^1 & \omega_2^1 & 0 \\ 0 & 0 & 1-\omega_3^1 & \omega_3^1 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix},$$

where

$$\begin{aligned}
&\omega_1^1 = \frac{\theta\bar{t} - \sin(\theta\bar{t})}{(\theta - \sin\theta)\sigma_2}, \qquad \omega_2^1 = \frac{\theta(\sigma_2 - \bar{t}) + \sigma_0}{\theta(\sigma_2 - \sigma_1 - 1) + 2\sigma_0}, \qquad \omega_3^1 = \frac{\theta(\sigma_1 + (1-\bar{t})) - \sigma_0}{(\theta - \sin\theta)\sigma_1}, \\
&\omega_1^2 = \frac{\theta(\sigma_1 + \sigma_2 - 1)\bar{t} + 2\sigma_0}{\theta(\sigma_2 - \bar{t}) + \sigma_0}, \qquad \omega_2^2 = \frac{\theta(\sigma_1\bar{t} - \sigma_2(1-\bar{t})) + \sigma_0}{\theta(\sigma_1 + (1-\bar{t})) - \sigma_0}, \\
&\omega_1^3 = \frac{(\theta\bar{t} - \sin(\theta\bar{t}))\sigma_1}{\theta(\sigma_1\bar{t} - \sigma_2(1-\bar{t})) + \sigma_0},
\end{aligned}$$

with

$$\sigma_0 := \frac{\sin(\theta(1-\bar{t})) + \sin(\theta\bar{t}) - \sin\theta}{1 - \cos\theta}, \qquad \sigma_1 := -\frac{1 - \cos(\theta(1-\bar{t}))}{1 - \cos\theta}, \qquad \sigma_2 := \frac{1 - \cos(\theta\bar{t})}{1 - \cos\theta}.$$

Proposition 5.34. *Let us consider the space* $\operatorname{span}\{1, t, (1-t)^n, t^n\}$ *with* $n \geqslant 3$ *and* $t \in [0,1]$. *Then, for any* $\bar{t} \in [0,1]$,

$$p(\bar{t}) = \begin{pmatrix} 1-\omega_1^3 & \omega_1^3 \end{pmatrix} \begin{pmatrix} 1-\omega_1^2 & \omega_1^2 & 0 \\ 0 & 1-\omega_2^2 & \omega_2^2 \end{pmatrix} \begin{pmatrix} 1-\omega_1^1 & \omega_1^1 & 0 & 0 \\ 0 & 1-\omega_2^1 & \omega_2^1 & 0 \\ 0 & 0 & 1-\omega_3^1 & \omega_3^1 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix},$$

where

$$\begin{aligned}
&\omega_1^1 = 1 - \frac{(n-2)(1-\bar{t})^{n-2}\bar{t}}{1 - (1-\bar{t})^{n-2}}, \qquad \omega_2^1 = \bar{t}, \qquad \omega_3^1 = \frac{(n-2)\bar{t}^{n-2}(1-\bar{t})}{1 - \bar{t}^{n-2}}, \\
&\omega_1^2 = 1 - \frac{\left(1 - (1-\bar{t})^{n-2}\right)(1-\bar{t})}{(n-2)\bar{t}}, \qquad \omega_2^2 = \frac{\left(1 - \bar{t}^{n-2}\right)\bar{t}}{(n-2)(1-\bar{t})}, \\
&\omega_1^3 = \bar{t}.
\end{aligned}$$

Proposition 5.35. *Let us consider the space* $\operatorname{span}\left\{1, t, \frac{(1-t)^3}{1+(h-3)(1-t)t}, \frac{t^3}{1+(h-3)(1-t)t}\right\}$ *with* $h \geqslant 3$ *and* $t \in [0,1]$. *Then, for any* $\bar{t} \in [0,1]$,

$$p(\bar{t}) = \begin{pmatrix} 1-\omega_1^3 & \omega_1^3 \end{pmatrix} \begin{pmatrix} 1-\omega_1^2 & \omega_1^2 & 0 \\ 0 & 1-\omega_2^2 & \omega_2^2 \end{pmatrix} \begin{pmatrix} 1-\omega_1^1 & \omega_1^1 & 0 & 0 \\ 0 & 1-\omega_2^1 & \omega_2^1 & 0 \\ 0 & 0 & 1-\omega_3^1 & \omega_3^1 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix},$$

where

$$\omega_1^1 = \frac{h\bar{t}}{3(1-\bar{t}) + h\bar{t}} = 1 - \frac{3(1-\bar{t})}{3(1-\bar{t}) + h\bar{t}}, \qquad \omega_2^1 = \bar{t}, \qquad \omega_3^1 = \frac{3\bar{t}}{3\bar{t} + h(1-\bar{t})},$$

$$\begin{aligned}
\omega_1^2 &= \frac{h\bar{t}}{h-(h-3)(1-\bar{t})^2} = 1 - \frac{(3(1-\bar{t})+h\bar{t})\,(1-\bar{t})}{3(1-\bar{t})^2+h(1-(1-\bar{t})^2)},\\
\omega_2^2 &= 1 - \frac{h(1-\bar{t})}{h-(h-3)\bar{t}^2} = \frac{(3\bar{t}+h(1-\bar{t}))\,\bar{t}}{3\bar{t}^2+h(1-\bar{t}^2)},\\
\omega_1^3 &= \frac{\bar{t}\,(h-(h-3)\bar{t}^2)}{3\,(1+(h-3)(1-\bar{t})\bar{t})}.
\end{aligned}$$

In addition to these explicit formulae, the quantities $\{\omega_j^\ell\}$ can be evaluated in a recursive and efficient way by applying the following procedure.

Algorithm 3 (Coefficients for de Casteljau algorithm in span $\left\{1, t, \frac{(1-t)^3}{1+(h-3)(1-t)t}, \frac{t^3}{1+(h-3)(1-t)t}\right\}$).

```
Input: $\bar{t} \in [0,1]$;
Initialize: $d_1^0 = 3,\ d_2^0 = h,\ d_3^0 = h,\ d_4^0 = 3$;
for $\ell = 1, \ldots, 3$
    for $j = 1, \ldots, 4-\ell$
        $d_j^\ell = d_j^{\ell-1}(1-\bar{t}) + d_{j+1}^{\ell-1}\bar{t}$;
        $\omega_j^\ell = 1 - \frac{d_j^{\ell-1}}{d_j^\ell}(1-\bar{t}) = \frac{d_{j+1}^{\ell-1}}{d_j^\ell}\bar{t}$;
Output: coefficients $\omega_j^\ell$.
```

It is interesting to remark that this algorithm has a form similar to Algorithm 5.1 in [MP99].

Figure 5.8 shows examples of the application of the de Casteljau algorithms in some generalized spaces.

Remark 5.16. In particular, Figures 5.8(a), 5.8(b), 5.8(c) illustrate the comparison between the corner cutting algorithms in Propositions 5.33, 5.34, 5.35, and the de Casteljau algorithm proposed in [CM06]. In fact, these algorithms are in general different, though they produce of course the same final value. The algorithm obtained by means of our approach is a proper corner cutting procedure in the sense that the points computed in the intermediate steps are all obtained by convex combinations of the points of the previous step, thus they are all placed on the control polygon of the curve. On the contrary, the first step of the algorithm in [CM06] proceeds in a substantially different way, which entails that the first and last points are not computed as convex combinations of the initial ones. However, for the space considered in Proposition 5.34, we observe that the points computed at the first step lie on the edges of the control polygon computed at the first step by the algorithm proposed here (see Figure 5.8(b)), and in the successive steps both algorithms produce the same points.

Remark 5.17. Referring to Figure 5.8(d) for an illustrative example, if we consider the space span $\{1, \cos(\theta t), \sin(\theta t), \cos(2\theta t), \sin(2\theta t)\}$ with $\theta \in (0, \frac{2}{3}\pi)$ and $t \in [0,1]$, then the de Casteljau algorithm that derives from our procedure of knot insertion coincides at each step with the so-called B-algorithm proposed in [MP99].

Remark 5.18. If we consider the space span $\{1, t, \cos(\theta t), \sin(\theta t), \cos(2\theta t), \sin(2\theta t)\}$ with $\theta \in (0, \frac{2}{3}\pi)$ and $t \in [0,1]$, then the de Casteljau algorithm that derives from our procedure of knot insertion is an alternative to the algorithm proposed in [RSA14] (we do not report the expressions for the quantities $\{\omega_j^\ell\}$ because of their excessive length). In particular, we observe that both proposals are corner cutting algorithms in the proper sense and at each step both place the points on the control polygon of the previous iteration, but in different locations (see Figure 5.8(e)). However, since our procedure is based on the (unique) ONTP basis and knot insertion, it provides at each step control points that are naturally associated with the ONTP

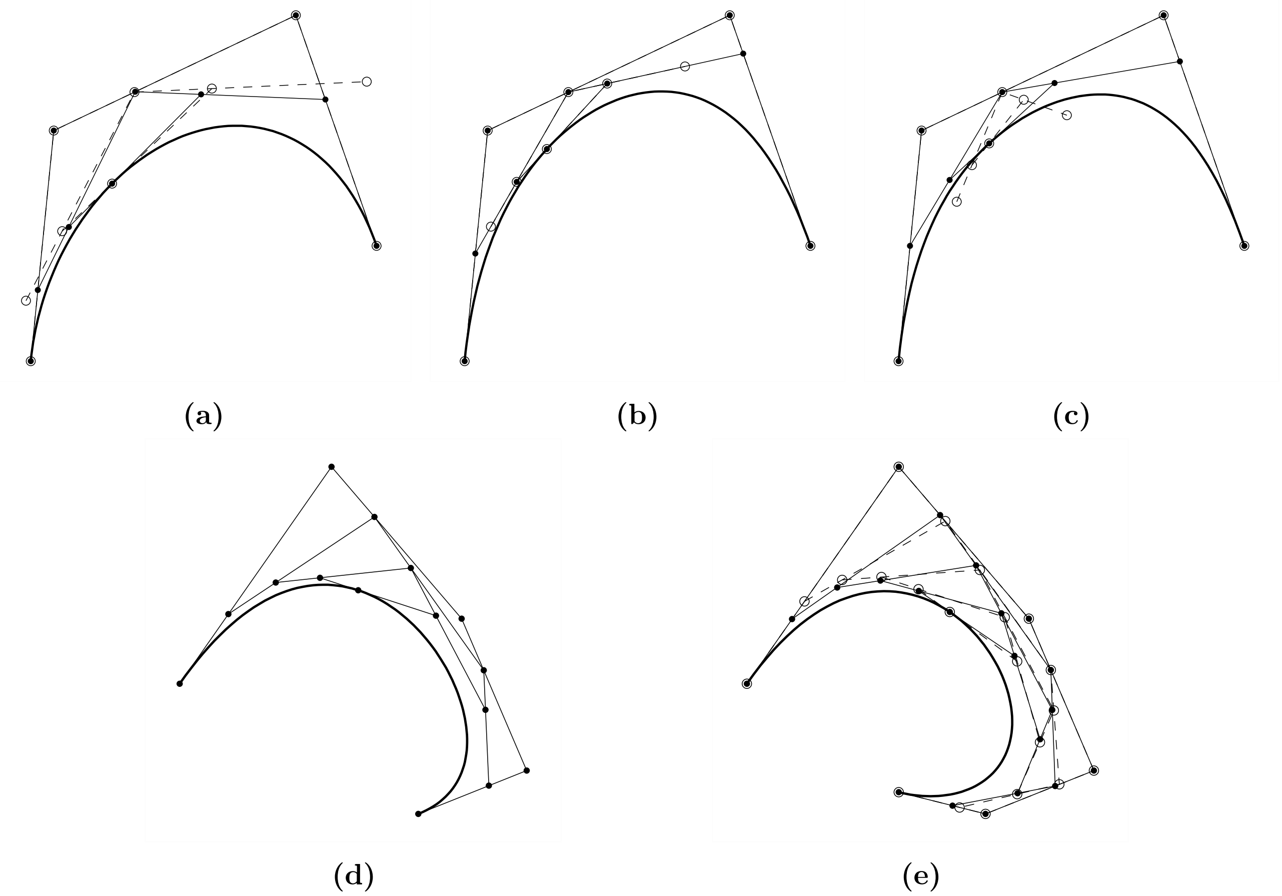


Figure 5.8: Examples of generalized Bézier curves and corner cutting algorithms in some QEC-spaces: (a) $\operatorname{span}\{1, t, \cos(\theta t), \sin(\theta t)\}$ with $\theta = \frac{\pi}{2}$, (b) $\operatorname{span}\{1, t, (1-t)^n, t^n\}$ with $n = 4$, (c) $\operatorname{span}\left\{1, t, \frac{(1-t)^3}{1+(h-3)(1-t)t}, \frac{t^3}{1+(h-3)(1-t)t}\right\}$ with $h = 6$, (d) $\operatorname{span}\{1, t, t^2, \cos(\theta t), \sin(\theta t)\}$ with $\theta = \frac{\pi}{2}$, (e) $\operatorname{span}\{1, t, \cos(\theta t), \sin(\theta t), \cos(2\theta t), \sin(2\theta t)\}$ with $\theta = \frac{\pi}{2}$. In all cases, the evaluation parameter is $\bar{t} = \frac{1}{3}$. (a)–(c) Comparisons between the corner cutting algorithm that derives from our procedure of knot insertion (dots, solid line) and the algorithm proposed in [CM06] for spaces of the form $\operatorname{span}\{1, t, u(t), v(t)\}$ (small circles, dashed line). (d) Steps of the corner cutting algorithm that derives from our procedure of knot insertion. (e) Comparison between the corner cutting algorithm that derives from our procedure of knot insertion (dots, solid line) and the algorithm proposed in [RSA14] (small circles, dashed line).

basis of the spline space obtained at any intermediate step, and this representation enjoys the variation diminishing property.

We conclude this section by emphasizing that a de Casteljau algorithm in the proper sense provides as a by-product a splitting of the given curve, as the points computed during the steps of the algorithm form the control polygons of the two segments in which the curve is split.

5.6.3.3 B-spline bases

In this section, we give the explicit expressions of the ONTP basis functions for some spaces $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ built on either a uniform or a non-uniform knot partition and such that all spaces $\{\mathcal{U}_{i,m}\}$ are equal to the one specified in each case.

Uniform knot partition. We start by considering the case of the uniform knot partition $\boldsymbol{\Delta} = \{x_i = i\}$.

Proposition 5.36. *Let us consider* 4*-dimensional QEC-spaces of the form* $\mathcal{U}_{i,4} = \operatorname{span}\{1, t, u(t), v(t)\}$ *with* $t \in [0,1]$ *for all* i*, where* $u(t)$ *and* $v(t)$ *are as follows:*

A) $u(t) = \cos(\theta t)$ *and* $v(t) = \sin(\theta t)$*, with* $\theta \in (0, 2\pi)$*;*

B) $u(t) = (1-t)^n$ *and* $v(t) = t^n$*, with* $n \geqslant 3$*;*

C) $u(t) = \frac{(1-t)^3}{1+(h-3)(1-t)t}$ *and* $v(t) = \frac{t^3}{1+(h-3)(1-t)t}$*, with* $h \geqslant 3$.

The B-spline basis functions of the associated spline spaces on uniform knots are respectively:

A)

$$N_{i,4}(x) = \begin{cases} \frac{\theta(x-i)-\sin(\theta(x-i))}{2\theta(1-\cos\theta)}, & i \leqslant x < i+1, \\ \frac{\theta(i+2-x)-\sin(\theta(i+2-x))-2\theta(x-(i+1))\cos\theta+2\sin(\theta(x-(i+1)))}{2\theta(1-\cos\theta)}, & i+1 \leqslant x < i+2, \\ \frac{\theta(x-(i+2))-\sin(\theta(x-(i+2)))-2\theta(i+3-x)\cos\theta+2\sin(\theta(i+3-x))}{2\theta(1-\cos\theta)}, & i+2 \leqslant x < i+3, \\ \frac{\theta(i+4-x)-\sin(\theta(i+4-x))}{2\theta(1-\cos\theta)}, & i+3 \leqslant x < i+4, \\ 0, & \textit{otherwise}; \end{cases}$$

B)

$$N_{i,4}(x) = \begin{cases} \frac{(x-i)^n}{2n}, & i \leqslant x < i+1, \\ \frac{(i+2-x)^n-2(x-(i+1))^n}{2n} + x - (i+1), & i+1 \leqslant x < i+2, \\ \frac{(x-(i+2))^n-2(i+3-x)^n}{2n} + i + 3 - x, & i+2 \leqslant x < i+3, \\ \frac{(i+4-x)^n}{2n}, & i+3 \leqslant x < i+4, \\ 0, & \textit{otherwise}; \end{cases}$$

C)

$$N_{i,4}(x) = \begin{cases} \frac{(x-i)^3}{2h(1+(h-3)(i+1-x)(x-i))}, & i \leqslant x < i+1, \\ \frac{(i+2-x)^3-2(x-(i+1))^3}{2h(1+(h-3)(i+2-x)(x-(i+1)))} + x - (i+1), & i+1 \leqslant x < i+2, \\ \frac{(x-(i+2))^3-2(i+3-x)^3}{2h(1+(h-3)(i+3-x)(x-(i+2)))} + i + 3 - x, & i+2 \leqslant x < i+3, \\ \frac{(i+4-x)^3}{2h(1+(h-3)(i+4-x)(x-(i+3)))}, & i+3 \leqslant x < i+4, \\ 0, & \textit{otherwise.} \end{cases}$$

Non-uniform knot partition. Now we consider the same QEC-spaces as above, but we give the explicit expressions of the B-spline basis functions for spline spaces on non-uniform knot partitions.

Proposition 5.37. *Being the setting as in Proposition 5.36, each B-spline basis function* $N_{i,4}$

has support $[t_i, t_{i+4}]$ *and is of the form:*

$$N_{i,4}(x) = \begin{cases} \dfrac{1}{H_i}\dfrac{d_i^2}{d_i+d_{i+1}}F(\tau_i), & t_i \leqslant x < t_{i+1}, \\ -\left(\dfrac{1}{H_i}+\dfrac{1}{H_{i+1}}\right)\dfrac{d_{i+1}^2}{d_{i+1}+d_{i+2}}F(\tau_{i+1}) \\ \quad -\dfrac{1}{H_i}\left(G_{i+1}(x)-\dfrac{d_{i+1}^2}{d_i+d_{i+1}}F(1-\tau_{i+1})\right), & t_{i+1} \leqslant x < t_{i+2}, \\ 1-\left(\dfrac{1}{H_i}+\dfrac{1}{H_{i+1}}\right)\dfrac{d_{i+2}^2}{d_{i+1}+d_{i+2}}F(1-\tau_{i+2}) \\ \quad -\dfrac{1}{H_{i+1}}\left(G_{i+2}(x)-\dfrac{d_{i+2}^2}{d_{i+2}+d_{i+3}}F(\tau_{i+2})\right), & t_{i+2} \leqslant x < t_{i+3}, \\ \dfrac{1}{H_{i+1}}\dfrac{d_{i+3}^2}{d_{i+2}+d_{i+3}}F(1-\tau_{i+3}), & t_{i+3} \leqslant x < t_{i+4}, \\ 0, & \textit{otherwise}; \end{cases}$$

where $d_i := t_{i+1}-t_i$, $\tau_i := \frac{x-t_i}{t_{i+1}-t_i}$, *and in the considered cases the other quantities are respectively defined as follows:*

A)

$$\begin{aligned} F(t) &= \theta t - \sin(\theta t), \\ G_i(x) &= (\theta - \sin\theta)(d_i - d_{i-1}) - \theta(1-\cos\theta)\,(x-t_i)\,, \\ H_i &= (\theta-\sin\theta)\,(d_i+d_{i+1}+d_{i+2}) + (3\sin\theta - \theta(2+\cos\theta))d_{i+1}; \end{aligned}$$

B)

$$\begin{aligned} F(t) &= t^n, \\ G_i(x) &= (d_i - d_{i-1}) - n\,(x-t_i)\,, \\ H_i &= (d_i+d_{i+1}+d_{i+2}) + (n-3)d_{i+1}; \end{aligned}$$

C)

$$\begin{aligned} F(t) &= \frac{t^3}{1+(h-3)(1-t)t}, \\ G_i(x) &= (d_i - d_{i-1}) - h\,(x-t_i)\,, \\ H_i &= (d_i+d_{i+1}+d_{i+2}) + (h-3)d_{i+1}. \end{aligned}$$

5.6.4 Subdivision schemes in generalized spline spaces

In this section we discuss some connections between knot insertion in generalized spline spaces and subdivision schemes. In particular, performing repeated knot insertion in a given spline space and thus passing from a coarser knot partition to a refined one, we work out the mask of a related subdivision scheme.

Based on the observations in Section 5.6.1, if the spline space admits refinability, then the subdivision scheme provides limit functions that belong to the same space. Conversely, if refinability does not hold, the limit functions of the scheme do not belong to the considered space.

Given a generalized spline space on a (bi-infinite) knot partition $\mathbf{\Delta} = \{x_i\}$ with distinct knots, let $\{N_{i,m}\}$ be the related B-spline basis, and let $\{\hat{N}_{2i+j,m}\}$ be the B-spline basis on a knot partition $\hat{\mathbf{\Delta}} = \{\hat{x}_i\}$ such that $\hat{x}_{2i} = x_i$, $\hat{x}_{2i+1} = \frac{1}{2}(x_i + x_{i+1})$. If the associated spline spaces are such that the inclusion holds true, we can express one basis function on the coarser partition as a linear combination of the basis functions on the refined partition as

$$N_{i,m}(x) = \sum_{j=0}^{m} \chi_{j,i}\, \hat{N}_{2i+j,m}(x), \qquad x \in \mathbb{R}, \tag{5.19}$$

where the coefficients $\chi_{j,i}$ are given by

$$\chi_{0,i} = f_{i,m}(\hat{x}_{2i+1}), \qquad \chi_{1,i} = 1, \qquad \chi_{2,i} = 1 - f_{i+1,m}(\hat{x}_{2i+3}),$$

if $m = 2$, and for $m > 2$ they can be determined explicitly by the recurrence relation

$$\chi_{0,i} = \frac{f_{i,m}(\hat{x}_{2i+1})}{\hat{f}_{2i,m}(\hat{x}_{2i+1})}, \tag{5.20}$$

$$\begin{aligned}\chi_{j,i} = \frac{1}{\hat{f}_{2i+j,m}(\hat{x}_{2i+j+1})} &\Bigg[f_{i,m}(\hat{x}_{2i+j+1}) - f_{i+1,m}(\hat{x}_{2i+j+1}) \\ &- \sum_{k=j-m+2}^{j-1} \chi_{k,i} \left(\hat{f}_{2i+k,m}(\hat{x}_{2i+j+1}) - \hat{f}_{2i+k+1,m}(\hat{x}_{2i+j+1}) \right) \Bigg] \\ = \chi_{j-1,i} + \frac{1}{\hat{f}_{2i+j,m}(\hat{x}_{2i+j+1})} &\Bigg[f_{i,m}(\hat{x}_{2i+j+1}) - f_{i+1,m}(\hat{x}_{2i+j+1}) - \chi_{j-m+2,i} \\ &+ \sum_{k=j-m+3}^{j-1} (\chi_{k-1,i} - \chi_{k,i}) \hat{f}_{2i+k,m}(\hat{x}_{2i+j+1}) \Bigg],\end{aligned}$$

for $j = 1, \ldots, m$, where we set $\chi_{j,i} = 0$ if $j < 0$ or $j > m$.
To work out relation (5.20), we write the $m \times m$ linear system which arises by requiring that (5.19) holds at each knot $\hat{x}_k$, $k = 2i+1, \ldots, 2i+m+1$. The system matrix is a banded matrix with bandwidth $m - 1$. From this property, and recalling the definition (5.3) of the B-spline basis functions in terms of the transition functions, (5.20) can be easily determined.

Remark 5.19. In general, expression (5.20) can provide the coefficients of a non-uniform non-stationary subdivision scheme.

As an application, we exploit the above relations to work out the refinement masks associated with some subdivision schemes, either in the situation where refinability holds or in the opposite one. In all cases, we shall consider a uniform knot partition $\mathbf{\Delta}$.

Remark 5.20. In the context of spline bases that admit nonstationary refinement, the knot insertion procedure gives rise to nonstationary subdivision schemes, as we see in the two examples below, which involve trigonometric functions.

Example 5.7. Let us consider the mixed algebraic/trigonometric space span$\{1, t, \cos(\theta t), \sin(\theta t)\}$, with $t \in [0, 1]$ and $\theta \in (0, \pi)$. In this space refinability holds, and the refinement by repeated knot insertion (i.e., placing a new knot at the midpoint of each knot interval) produces a nonstationary subdivision scheme (see [WW01, Section 4.4.3]) whose mask is

$$\left[\frac{1}{4(1+\sigma_k)}, \frac{1}{2}, 1 - \frac{1}{2(1+\sigma_k)}, \frac{1}{2}, \frac{1}{4(1+\sigma_k)} \right], \qquad \sigma_k := \cos\left(\frac{\theta}{2^k}\right),$$

where the "tension" parameters satisfy the recurrence $\sigma_k = \sqrt{\frac{1+\sigma_{k-1}}{2}}$ starting with $\sigma_0 \in (-1, 1)$, which is determined by the value of θ.

As showed in [WW01], this formula is valid also in the purely polynomial case span$\{1, t, t^2, t^3)\}$ (corresponding to the limit $\theta = 0$, thus $\sigma_k = 1\ \forall k$), and in the algebraic/hyperbolic (exponential) case span$\{1, t, \cosh(\theta t), \sinh(\theta t)\}$ with $\theta \geqslant 0$ (where now $\sigma_k := \cosh\left(\frac{\theta}{2^k}\right) > 1$).

In all three cases, the resulting limit curve is a C^2 spline.

Example 5.8. Now we consider the mixed algebraic/trigonometric space span$\{1, t, t^2, \cos(\theta t), \sin(\theta t)\}$, with $t \in [0, 1]$ and $\theta \in (0, 2\pi)$. Also this space admits refinability, and the refinement produces a nonstationary subdivision scheme whose mask is

$$\left[\frac{1}{8(1+\sigma_k)}, \frac{1}{4} + \frac{1}{8(1+\sigma_k)}, \frac{3}{4} - \frac{1}{4(1+\sigma_k)}, \frac{3}{4} - \frac{1}{4(1+\sigma_k)}, \frac{1}{4} + \frac{1}{8(1+\sigma_k)}, \frac{1}{8(1+\sigma_k)}\right],$$

$$\sigma_k := \cos\left(\frac{\theta}{2^k}\right),$$

where the "tension" parameters satisfy the recurrence $\sigma_k = \sqrt{\frac{1+\sigma_{k-1}}{2}}$ starting with $\sigma_0 \in (-1, 1)$, which is determined by the value of θ.

Similarly to the previous example, this formula is valid also in the purely polynomial case span$\{1, t, t^2, t^3, t^4)\}$ if $\theta = 0$, and in the algebraic/hyperbolic case span$\{1, t, t^2, \cosh(\theta t), \sinh(\theta t)\}$ with $\theta > 0$.

The resulting limit curve is a C^3 spline.

Example 5.9. Let us focus on the space of variable-degree polynomial splines [Cos00] span$\{1, t, (1-t)^{n_1}, t^{n_2}\}$ with $t \in [0, 1]$ and degrees $n_1 = n_2 = h \geqslant 3$. This space does not possess the refinability property. However, we can still perform knot insertion according to the first strategy described in Section 5.6.1, and after one step of refinement the coefficients that we obtain turn out to be the coefficients that form the mask of the stationary subdivision scheme studied in [GP00]:

$$\left[\frac{1}{2^h}, \frac{1}{2}, 1 - \frac{1}{2^{h-1}}, \frac{1}{2}, \frac{1}{2^h}\right], \qquad h \geqslant 3.$$

By the eigenanalysis of the local subdivision matrix, it can be proved that this scheme converges to a C^1 spline.

Example 5.10. Also the rational space span $\left\{1, t, \frac{(1-t)^3}{1+(\mu-3)(1-t)t}, \frac{t^3}{1+(\nu-3)(1-t)t}\right\}$ with $t \in [0, 1]$ and "tension" parameters $\mu = \nu = h \geqslant 3$, studied in [DG85], does not admit refinability. However, proceeding by the same knot insertion strategy recalled in the previous example, we obtain the mask:

$$\left[\frac{3}{2h(h+1)}, \frac{1}{2}, 1 - \frac{3}{h(h+1)}, \frac{1}{2}, \frac{3}{2h(h+1)}\right], \qquad h \geqslant 3.$$

To our knowledge, this scheme has never appeared in the literature, and, by the eigenanalysis of the local subdivision matrix, the convergence to a C^1 spline can be proved.

5.6.5 Computational methods

One of the main arguments in support of generalized splines is that, in some applications, they may be more powerful than the well-established polynomial and rational splines. In this view, their effective use requires that simple and efficient tools for computation be available.

An advantage of the proposed constructive approach is that it straightforwardly translates into procedures for computing with generalized splines. Based on this observation, the following list of algorithms outlines the main computational tools that can be drawn from the previous sections, including methods for evaluating a generalized B-spline with non-uniform and also multiple knots (and, analogously, its derivatives and integral), and for computing the spline coefficients of the representations associated either with a refined knot partition or with a raised order.

Algorithm 4 (Construction of transition functions)**.** Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a given generalized spline space and $\boldsymbol{\Delta}^*$ an extended partition. To determine the coefficients of the transition functions $f_{i,m}$, $i = 2, \ldots, m+K$, w.r.t. the QEC-systems $\{1, u_{j,2}, \ldots, u_{j,m}\}$, $j = 0, \ldots, k$:

1. solve the linear system (5.2) for each $f_{i,m}$, $i = 2, \ldots, m+K$;
2. assemble the matrix $B = (b_{i,j})$ of dimensions $(m+K) \times m(m-1)$ that in each row i stores the solution of the ith linear system, i.e., the coefficients that define the transition function $f_{i,m}$ w.r.t. to the QEC-systems $\{1, u_{i_j,2}, \ldots, u_{i_j,m}\}$, $j = 1, \ldots, p-1$.

Remark 5.21. Since $f_{1,m} = 1$, the first row of the matrix B above is given by $(1, 0, \ldots, 0, \ldots, 1, 0, \ldots, 0)$. However, note that they are not needed in the following algorithms.

Algorithm 5 (Evaluation of transition functions)**.** To evaluate at $\bar{x} \in [t_\ell, t_{\ell+1}] = [x_h, x_{h+1}]$ the (nontrivial) transition functions $f_{\ell-m+2,m}, \ldots, f_{\ell,m}$:

1. $r \leftarrow m(m-1)$;
2. **for** $i = \ell - m + 2, \ldots, \ell$
   - i. $r \leftarrow r - m$;
   - ii. $f_{i,m}(\bar{x}) \leftarrow \sum_{j=1}^{m} b_{i,r+j}\, u_{h,j}(\bar{x})$, where the $\{b_{i,j}\}$ are the entries of the matrix B constructed by Algorithm 4.

Algorithm 6 (Evaluation of a generalized spline)**.** Given $\bar{x} \in [a, b]$, to evaluate the spline (5.10) at $\bar{x}$:

1. determine ℓ such that $\bar{x} \in [t_\ell, t_{\ell+1})$;
2. use Algorithm 5 to evaluate the transition functions $f_{\ell-m+2,m}, \ldots, f_{\ell,m}$ at $\bar{x}$;
3. $N_{\ell-m+1,m}(\bar{x}) \leftarrow 1 - f_{\ell-m+2,m}(\bar{x})$;
   $N_{i,m}(\bar{x}) \leftarrow f_{i,m}(\bar{x}) - f_{i+1,m}(\bar{x})$ for $i = \ell - m + 2, \ldots, \ell - 1$;
   $N_{\ell,m}(\bar{x}) \leftarrow f_{\ell,m}(\bar{x})$.
4. compute $s(\bar{x})$ as the linear combination (5.10).

Remark 5.22 (Derivatives and integral of a generalized spline). By linearity, evaluating the derivatives and integral amounts to differentiating and integrating the transition functions and ultimately the functions $\{u_{i,j}\}$ of the QEC-systems. Thus this computation follows the same outline of Algorithms 5 and 6.

Algorithm 7 (Knot insertion)**.** Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a given generalized spline space, $\boldsymbol{\Delta}^*$ an extended partition, and s a spline function of the form (5.10). Given $\hat{t} \in [a, b]$, to determine the coefficients $\hat{c}_i$, $i = 1, \ldots, m+K+1$, obtained by inserting $\hat{t}$ in $\boldsymbol{\Delta}^*$ once:

1. determine ℓ such that $\hat{t} \in [t_\ell, t_{\ell+1})$;
2. compute the coefficients α_i, $i = \ell - m + 2, \dots, \ell - r + 1$ through (5.7);
3. use (5.11) to compute the coefficients $\hat{c}_i$, $i = 1, \dots, m + K + 1$.

Algorithm 8 (Order elevation)**.** Let $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ be a given generalized spline space, $\boldsymbol{\Delta}^*$ an extended partition, and s a spline function of the form (5.10). To represent s in the space $S(\boldsymbol{\mathcal{U}}_{m+n}, \tilde{\boldsymbol{M}}, \boldsymbol{\Delta})$ of order elevated by $n = 1, 2$:

1. repeatedly apply Algorithm 7 to the space $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$ until all the knots have multiplicity $m - 1$;
2. on each non-degenerate knot interval $[t_\ell, t_{\ell+1}]$, determine the m transition functions $f_{i,m}$ related to the generalized Bernstein basis for $\mathcal{U}_{\ell,m}$, and evaluate at t_ℓ the derivative of order $i - 1$ if $n = 1$, and also of order i if $n = 2$;
3. on each non-degenerate knot interval $[t_\ell, t_{\ell+1}]$, determine the $m + n$ transition functions $f_{i,m+n}$ related to the generalized Bernstein basis for $\mathcal{U}_{\ell,m+n}$, and evaluate at t_ℓ the derivative of order $i - 1$ if $n = 1$, and also of order i if $n = 2$;
4. if $n = 1$, determine the coefficients γ_i, $i = 2, \dots, m$, through (5.13), or, if $n = 2$, determine the coefficients γ_i, $i = 2, \dots, m$, through (5.16a) and the coefficients δ_i, $i = 2, \dots, m + 1$, through (5.16b);
5. determine the coefficients $\tilde{c}_i$, $i = 1, \dots, m + n$, either through (5.14) if $n = 1$ or through (5.18) if $n = 2$;
6. apply a knot removal algorithm to the spline space $S(\boldsymbol{\mathcal{U}}_{m+n}, \hat{\boldsymbol{M}}, \boldsymbol{\Delta})$, in which all the knots have multiplicity m, until multiplicity $m_i + n$, where m_i is the initial multiplicity of each knot in $S(\boldsymbol{\mathcal{U}}_m, \boldsymbol{M}, \boldsymbol{\Delta})$.

5.6.6 Application examples

This section presents an instructive example on the proposed constructive approach and some numerical examples requiring the combined use of several algorithms proposed above.

Example 5.11 (Construction of a B-spline basis function)**.** In this example we will show the construction of a single B-spline basis function, defined on a given sequence of QEC-spaces. In particular, we consider the three spaces

$$\begin{aligned}
\mathcal{U}_{i,3} &= \operatorname{span}\{1, t, t^2\}, & t &= t(x) := x - x_i, & x &\in I_i,\\
\mathcal{U}_{i+1,3} &= \operatorname{span}\{1, \cos(\theta t), \sin(\theta t)\}, & t &= t(x) := x - x_{i+1}, & x &\in I_{i+1},\\
\mathcal{U}_{i+2,3} &= \operatorname{span}\{1, \cosh(\phi t), \sinh(\phi t)\}, & t &= t(x) := x - x_{i+2}, & x &\in I_{i+2},
\end{aligned}$$

and, assuming for simplicity an extended partition $\boldsymbol{\Delta}^* = \{t_j\}$ with distinct knots $t_j < t_{j+1}$, we derive the expression of the B-spline basis function $N_{i+3,3} = f_{i+3} - f_{i+4}$ whose support is $[t_{i+3}, t_{i+6}] = [x_i, x_{i+3}]$.

We observe that, while $D\mathcal{U}_{i,3}$ and $D\mathcal{U}_{i+2,3}$ are EC-spaces on $\mathbb{R}$, for $D\mathcal{U}_{i+1,3}$ to be an EC-space on the related interval I_{i+1} we need to require that $d_{i+1} := x_{i+2} - x_{i+1} \in (0, \pi)$ and $\theta \in (0, 1)$.

We can now express each piece of the transition functions $f_{j,3}$, $j = i+3, i+4$, in the QEC-system $\{u_{h,1}, u_{h,2}, u_{h,3}\}$ that generates the corresponding $\mathcal{U}_{h,3}$, $h = i, i+1, i+2$, on the related knot interval as

$$f_{j,3}(x) = \begin{cases} 0, & x < t_j, \\ \displaystyle\sum_{k=1}^{3} b_{j,k} u_{j-3,k}(t(x)), & t_j \leqslant x < t_{j+1}, \\ \displaystyle\sum_{k=1}^{3} b_{j,k+3} u_{j-2,k}(t(x)), & t_{j+1} \leqslant x < t_{j+2}, \\ 1, & x \geqslant t_{j+2}, \end{cases}$$

and compute the coefficients $\{b_{j,k}\}$ of the transition function $f_{j,3}$, $j = i+3, i+4$, by solving the 6×6 linear system of the form (5.2) that arises from the continuity conditions

$$\begin{aligned} &f_{j,3}^{[1]}(x_{j-3}) = 0, && Df_{j,3}^{[1]}(x_{j-3}) = 0, \\ &f_{j,3}^{[1]}(x_{j-2}) = f_{j,3}^{[2]}(x_{j-2}), && Df_{j,3}^{[1]}(x_{j-2}) = Df_{j,3}^{[2]}(x_{j-2}), \\ &f_{j,3}^{[2]}(x_{j-1}) = 1, && Df_{j,3}^{[2]}(x_{j-1}) = 0. \end{aligned}$$

For $f_{i+3,3}$, this gives

$$\begin{aligned} &b_{i+3,1} = b_{i+3,2} = 0, \qquad b_{i+3,3} = \frac{1}{d_i^2 + 2\frac{d_i}{\theta}\tan\left(\frac{\theta}{2}d_{i+1}\right)}, \\ &b_{i+3,4} = 1 + \frac{1}{\cos(\theta d_{i+1}) - \frac{\theta}{2}d_i \sin(\theta d_{i+1}) - 1}, \qquad b_{i+3,5} = (1 - b_{i+3,4})\cos(\theta d_{i+1}), \\ &b_{i+3,6} = \frac{1}{\frac{\theta}{2}d_i + \tan\left(\frac{\theta}{2}d_{i+1}\right)}, \end{aligned}$$

while, for $f_{i+4,3}$,

$$\begin{aligned} &b_{i+4,1} = -b_{i+4,2} = \frac{1}{1 - \cos(\theta d_{i+1}) + \frac{\theta}{\phi}\sin(\theta d_{i+1})\tanh\left(\frac{\phi}{2}d_{i+2}\right)}, \qquad b_{i+4,3} = 0, \\ &b_{i+4,4} = 1 + \frac{1}{\cosh(\phi d_{i+2}) + \frac{\phi}{\theta}\tan\left(\frac{\theta}{2}d_{i+1}\right)\sinh(\phi d_{i+2}) - 1}, \\ &b_{i+4,5} = (1 - b_{i+4,4})\cosh(\phi d_{i+2}), \qquad b_{i+4,6} = \frac{1}{\frac{\phi}{\theta}\tan\left(\frac{\theta}{2}d_{i+1}\right) + \tanh\left(\frac{\phi}{2}d_{i+2}\right)}. \end{aligned}$$

Recalling Proposition 5.21, when the dimension of the QEC-spaces is 3, like in this case, the spline space always admits the ONTP basis. For this reason, it turns out that the transition functions are always positive and monotonically increasing and that the difference between two of them is always positive as well. Based on all the results in the previous sections, we can thus conclude that the function $N_{i+3,3} = f_{i+3,3} - f_{i+4,3}$ is a member of the B-spline (ONTP) basis.

Example 5.12 (Explicit formulae for B-spline bases in the case $m = 3$)**.** If we consider spaces with a relatively simple structure (i.e., with low dimension m and basic expressions for the generators of QEC-systems), the theoretical framework presented in Section 5.3 also allows for computing symbolically explicit expressions of the basis functions. As an example, we have computed the generic expression of a transition function in a generalized spline space

$S(\boldsymbol{\mathcal{U}}_3, \boldsymbol{M}, \boldsymbol{\Delta})$ with $\mathcal{U}_{i,3} = \operatorname{span}\{1, u_i(x), v_i(x)\}$ for all i:

$$f_{i,3}(x) = \begin{cases} 0, & x < t_i, \\ -\dfrac{G_i(x)}{H_i(t_{i+2})}, & t_i \leqslant x < t_{i+1}, \\ \dfrac{H_i(x)}{H_i(t_{i+2})}, & t_{i+1} \leqslant x < t_{i+2}, \\ 1, & x \geqslant t_{i+2}, \end{cases}$$

where

$$\begin{aligned} &C_i := u_i'(t_{i+1})v_i'(t_i) - u_i'(t_i)v_i'(t_{i+1}), \\ &F_i(x,y) := (v_i(t_i) - v_i(x))u_i'(y) - (u_i(t_i) - u_i(x))v_i'(y), \\ &G_i(x) := C_{i+1}F_i(x,t_i), \qquad H_i(x) := C_iF_{i+1}(x,t_{i+2}) - C_{i+1}F_i(t_{i+1},t_i), \end{aligned}$$

and the B-spline basis functions are defined as $N_{i,3}(x) := f_{i,3}(x) - f_{i+1,3}(x)$.

Example 5.13 (Tools for Isogeometric Analysis)**.** In this example we discuss how the B-spline basis functions defined by our approach can be conveniently exploited in the context of Isogeometric Analysis.

A first advantageous aspect is related to the fact that the fundamental operations of differentiation and integration can be performed straightforwardly, by following the outline of Algorithms 5 and 6, which only require to know the derivatives and primitive functions of the functions $\{u_{i,j}\}$. These, in turn, take a simple form since for spaces of generalized splines useful for design the QEC-systems consist of basic trigonometric, polynomial, or simple rational functions.

Moreover, the tools of knot insertion and order elevation provided in Sections 5.6.1 and 5.6.2 can be combined to perform the so-called k-refinement [HCB05, Section 2.6]. Such a refinement approach results in a lower number of basis functions and higher continuity with respect to the h-p-refinement, as illustrated by Figure 5.9, which is the counterpart of Figure 10 in [HCB05], exploiting a space of generalized splines instead of polynomial splines.

Example 5.14 (2-order elevation)**.** As discussed in Section 5.6.2, when the target space contains an additional couple of trigonometric or hyperbolic functions, it has to be reached by elevating by two orders at once. Figure 5.10 shows two examples of 2-order elevations for both generalized Bézier and B-spline curves built on spaces of mixed type.

Example 5.15 (Modeling with generalized B-splines)**.** This example illustrates two essential tools for geometric modeling and design. The first is the progressive refinement of a parametric curve, which is shown in Figures 5.11(a) and 5.11(b). The second is the procedure of conversion into standard representation (i.e., with clamped knot partition) of a closed generalized spline curve with a periodic non-uniform knot partition. It has been obtained by exploiting the knot insertion formula (5.7) at the left and the right endpoint of the knot partition. The standard representation is shown in Figure 5.11(c).

5.7 Concluding remarks

To summarize, in this chapter we proposed a general approach that allows us to construct the optimal normalized totally positive basis for a generalized spline space, if it admits one.

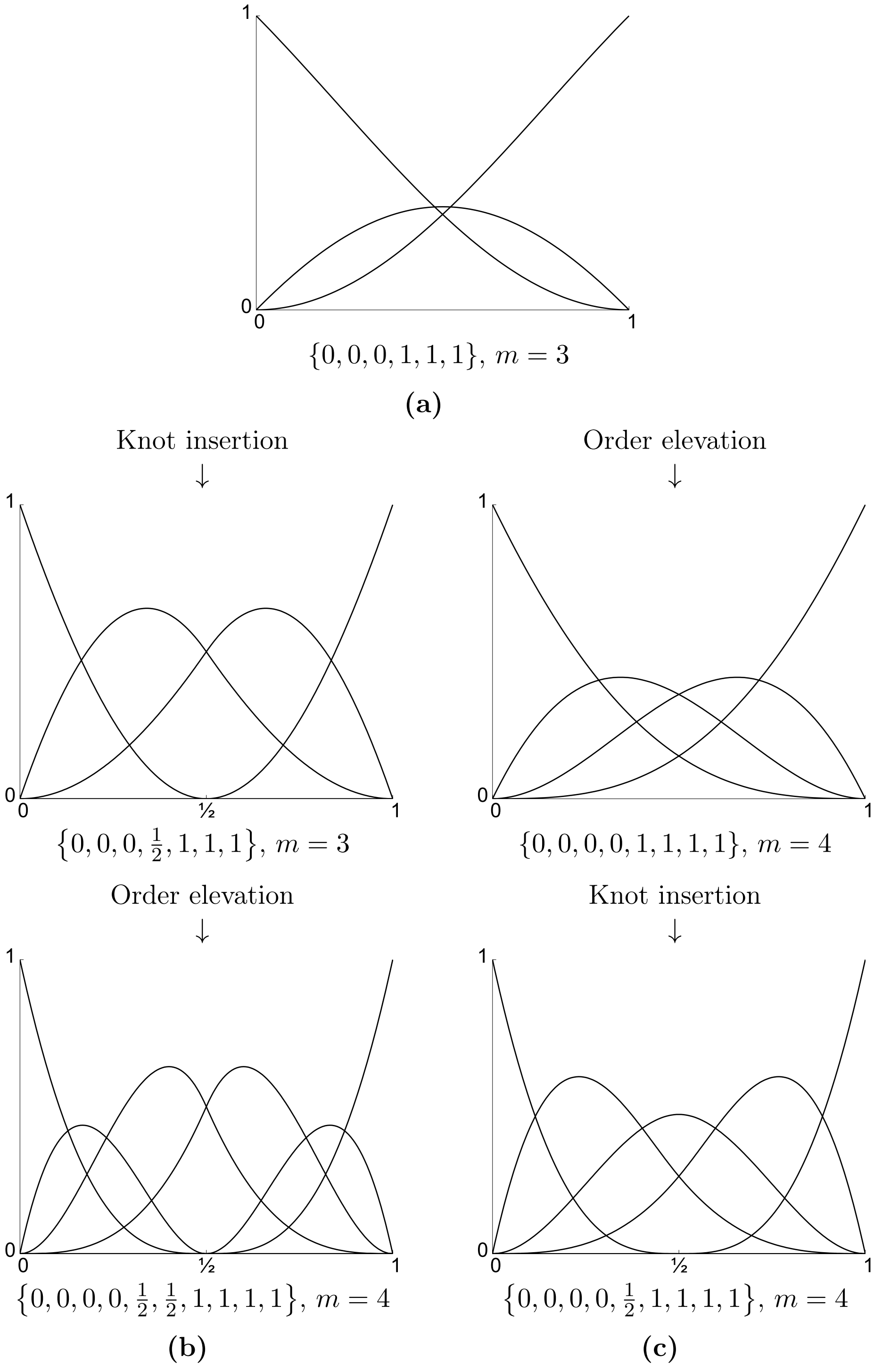


Figure 5.9: k-refinement takes advantage of the fact that knot insertion and order elevation do not commute. (a) Base case of one element in the EC-space span $\{1, \cos(\theta t), \sin(\theta t)\}$ with $\theta = 2$ and the specified extended partition. The EC-space is to be order-elevated to span $\{1, t, \cos(\theta t), \sin(\theta t)\}$. (b) h-p-refinement: knot insertion followed by order elevation results in six basis functions that are C^1 at internal knots. (c) k-refinement: order elevation followed by knot insertion results in five basis functions that are C^2 at internal knots.

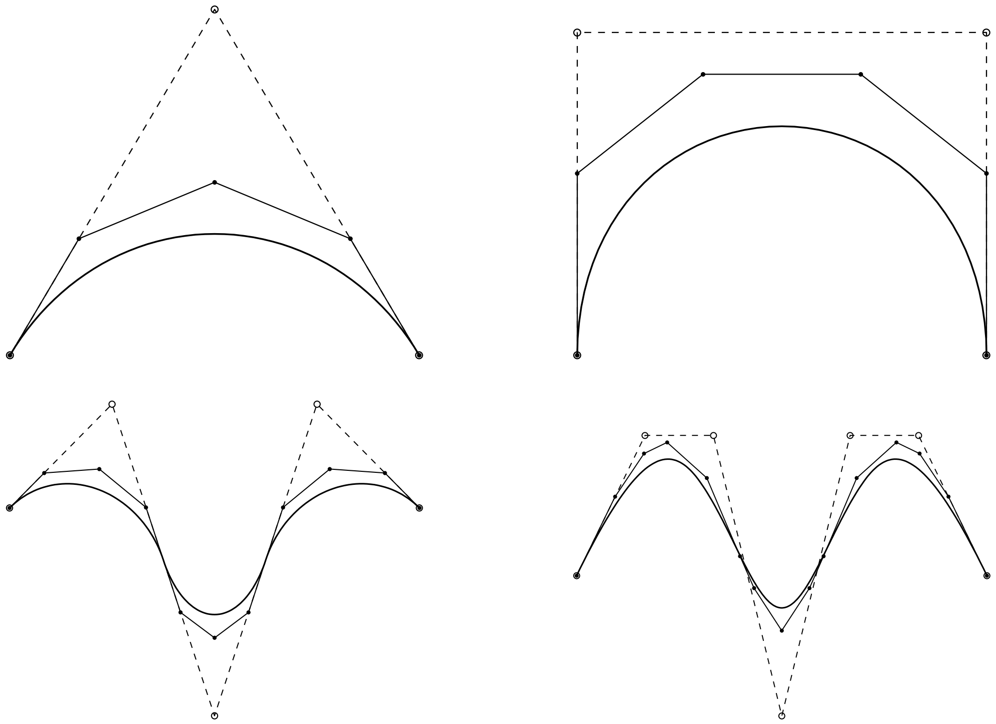


Figure 5.10: Elevation by two orders on generalized Bézier (top) and B-spline (bottom) curves for different spaces: from span $\{1, \cos(\theta_i t), \sin(\theta_i t)\}$ to span $\{1, \cos(\theta_i t),\ \sin(\theta_i t),\ \cosh(\phi_i t),\ \sinh(\phi_i t)\}$ with $t \in [0, 3.14]$ and $\theta_i = 0.5$, $\phi_i = 1$, $\forall i$ (left), and from span $\{1, t,\ \cos(\theta_i t),\ \sin(\theta_i t)\}$ to span $\{1, t,$ $\cos(\theta_i t),\ \sin(\theta_i t),\ t\cos(\theta_i t),\ t\sin(\theta_i t)\}$ with $t \in [0, 6.28]$ and $\theta_i = 0.5$, $\forall i$ (right). For the B-spline curves in the bottom left and right figures, the knot partitions are $\{0, 0, 0, 3.14, 6.28, 9.42, 9.42, 9.42\}$ and $\{0, 0, 0, 0, 3.14, 6.28, 9.42, 12.56, 12.56, 12.56, 12.56\}$ respectively.

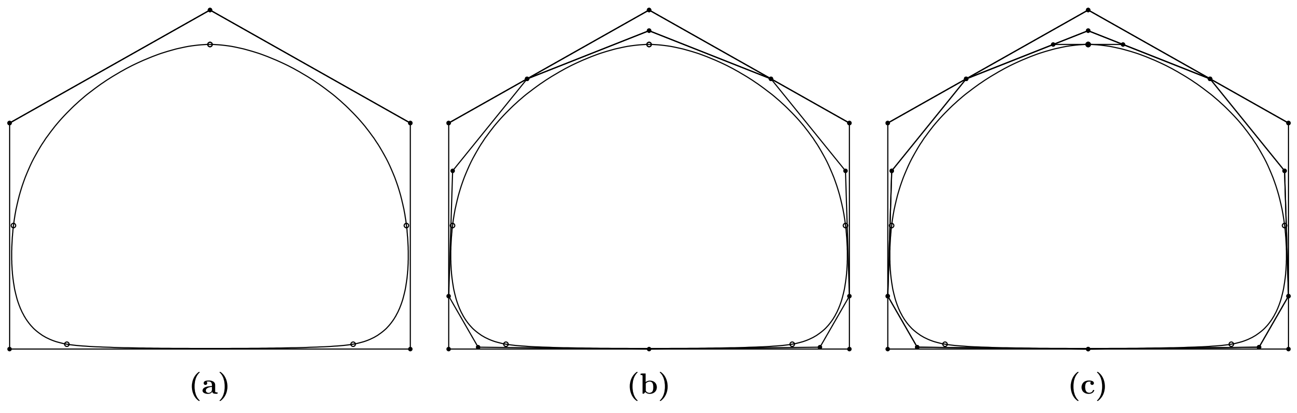


Figure 5.11: Example of a closed spline curve with periodic non-uniform knot partition. All the underlying QEC-spaces are span $\left\{1, t, \frac{(1-t)^3}{1+(h_i-3)(1-t)t}, \frac{t^3}{1+(h_i-3)(1-t)t}\right\}$, where the tension parameter h_i is 8 for one space and 4 for all others. Left: curve and control polygon corresponding to the knot partition $\{-0.55, -0.45, -0.35, 0, 0.35, 0.45, 0.55, 0.65, 1, 1.35, 1.45, 1.55\}$; center: one step of refinement, obtained by inserting the midpoint of each knot interval; right: standard representation.

Moreover, based on this method, we provided a practical criterion for determining whether a given spline space possesses the ONTP basis, and we showed how this result can be translated into a general numerical methodology for the verification of the existence of such a basis. In addition, the idea of transition functions, which is the key ingredient of the proposed construction, proved to be extremely convenient also to formulate computational algorithms and express essential operations which are commonly used in the context of geometric modeling.

Although we presented the framework in the setting of QEC-spaces all having the same dimension and parametric continuity conditions at the knots, the approach lends itself to the generalization to the cases where the QEC-spaces may differ in dimension and geometric continuity conditions are required.

Chapter 6

Conclusion

In this dissertation we have addressed several problems that are of interest in the context of geometric modeling, both theoretical and practical. In particular, the contributions of our work cover some intimately connected topics, namely subdivision schemes, non-uniform local interpolation with curves and surfaces, and generalized splines.

In Chapter 2 we have presented an effective way of integrating subdivision surfaces in CAD systems, making them available as a very intuitive modeling tool that at the same time meets the requirements needed in a CAD framework. We remark that the proposed strategy for such an integration was acknowledged as innovative at the time of its realization. The implementation was heavily tested on real-world cases and then included as a stable part of a CAD software on the market. According to the users that have exploited it in industrial design projects, it has produced very good results.

Besides Chapter 2, also all the other constructions proposed throughout the dissertation bear in mind the importance of application and implementation. In fact, alongside discussing the theoretical aspects of the undertaken work, a relevant amount of space has been committed to providing algorithms drawn from the presented methods that can be translated into numerical procedures and successfully implemented.

In Chapter 3 we have considered the definition of locally supported piecewise polynomial interpolants on non-uniform knot partitions that feature prescribed design properties (support width, degree, orders of continuity and approximation). The proposed approach represents an advancement with respect to the existing literature in that it extends a preceding construction by allowing us to both increasing the flexibility of the considered spline interpolants and reducing their computational cost by minimizing their degree. Moreover, pursuing the intent of providing the reader with as many tools as possible from a design viewpoint, in Chapter 3 we have devoted sections to addressing the often neglected issue of endpoint corrections for open data sets, as well as to offering examples of how to exploit the increased flexibility in order to achieve particular shape effects.

Local spline interpolants of the above kind have been exploited in Chapter 4 to work out a network of curves interpolating a given mesh of points, which is one of the main ingredients of the framework introduced there for the construction of smooth composite interpolation surfaces with a good shape quality. The other point of novelty in the approach presented in Chapter 4 is the use of a peculiar parameterization strategy for the surface that at the same time guarantees the interpolation of the curves of the network (each of which with its own parameterization) and allows the surface to achieve good aesthetic properties.

Moreover, the idea behind the definition of the blending functions exploited in Chapter 3 is the concept of transition function. The same notion has been transferred to the more general setting of generalized spline spaces suitable for design purposes as the key tool for the construction of optimal normalized totally positive bases, as discussed in detail in Chapter 5. Additionally, a large variety of the algorithms presented for computation and modeling with generalized splines take advantage of the formulation by means of transition functions.

In view of the results presented in this dissertation, a natural development consists in the actual integration of methods for generalized splines in CAD systems. In fact, such classes of splines would generously enrich the batch of representable entities, since they would allow users to achieve particular tension effects or an exact representation of relevant shapes. As an example, we may be interested in reproducing conics and quadrics, which are essential for modeling mechanical parts and objects with rotational symmetry, as well as surfaces of revolution. Working with the suitable trigonometric (or mixed polynomial/trigonometric) spaces, we would be able to obtain this kind of shapes without resorting to NURBS and their inherent rational nature.

Furthermore, motivated by the positive experience reported in Chapter 2, another interesting advancement would be the integration in CAD systems of univariate and bivariate subdivision schemes based on generalized splines, for the same reasons as above.

Also the proposed construction of local interpolating surfaces with augmented parameterization described in Chapter 4 would provide benefits in the context of arbitrary-topology CAD modeling, and compete with (if not outclass) the existing interpolatory methods in terms of shape quality.

References

[ABC$^+$13] Michele Antonelli, Carolina Vittoria Beccari, Giulio Casciola, Roberto Ciarloni, and Serena Morigi. Subdivision surfaces integrated in a CAD system. *Computer-Aided Design*, 45(11):1294–1305, 2013. doi:10.1016/j.cad.2013.06.007.

[ABC14a] Michele Antonelli, Carolina Vittoria Beccari, and Giulio Casciola. A general framework for the construction of piecewise-polynomial local interpolants of minimum degree. *Advances in Computational Mathematics*, 40(4):945–976, 2014. doi:10.1007/s10444-013-9335-y.

[ABC14b] Michele Antonelli, Carolina Vittoria Beccari, and Giulio Casciola. High-quality local interpolation of arbitrary-topology meshes and curve networks by composite parametric surfaces. Submitted, 2014.

[ABCR14a] Michele Antonelli, Carolina Vittoria Beccari, Giulio Casciola, and Lucia Romani. Computation and modeling in spaces of generalized splines. Submitted, 2014.

[ABCR14b] Michele Antonelli, Carolina Vittoria Beccari, Giulio Casciola, and Lucia Romani. A constructive approach to generalized splines. Submitted, 2014.

[ADS06] Ursula H. Augsdörfer, Neil A. Dodgson, and Malcolm A. Sabin. Tuning subdivision by minimising Gaussian curvature variation near extraordinary vertices. *Computer Graphics Forum*, 25(3):263–272, 2006. doi:10.1111/j.1467-8659.2006.00945.x.

[ANW67] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh. *The Theory of Splines and Their Applications*, volume 38 of *Mathematics in Science and Engineering*. Academic Press, New York, NY, USA, 1967.

[Bar96] P. J. Barry. de Boor-Fix dual functionals and algorithms for Tchebycheffian B-spline curves. *Constructive Approximation*, 12(3):385–408, 1996. doi:10.1007/BF02433050.

[BCR11a] Carolina Vittoria Beccari, Giulio Casciola, and Lucia Romani. Non-uniform interpolatory curve subdivision with edge parameters built-upon compactly supported fundamental splines. *BIT Numerical Mathematics*, 51(4):781–808, 2011. doi:10.1007/s10543-011-0328-2.

[BCR11b] Carolina Vittoria Beccari, Giulio Casciola, and Lucia Romani. Polynomial-based non-uniform interpolatory subdivision with features control. *Journal of Computational and Applied Mathematics*, 235(16):4754–4769, 2011. doi:10.1016/j.cam.2010.09.014.

[BCR13a] Carolina Vittoria Beccari, Giulio Casciola, and Lucia Romani. Construction and characterization of non-uniform local interpolating polynomial splines. *Journal of Computational and Applied Mathematics*, 240:5–19, 2013. doi:10.1016/j.cam.2012.06.025.

[BCR13b] Carolina Vittoria Beccari, Giulio Casciola, and Lucia Romani. Non-uniform non-tensor product local interpolatory subdivision surfaces. *Computer Aided Geometric Design*, 30(4):357–373, 2013. doi:10.1016/j.cagd.2013.02.002.

[BFL$^+$10] Carolina Vittoria Beccari, Elisabetta Farella, Alfredo Liverani, Serena Morigi, and Marco Rucci. A fast interactive reverse-engineering system. *Computer-Aided Design*, 42(10):860–873, 2010. doi:10.1016/j.cad.2010.06.001.

[BLZ00] Henning Biermann, Adi Levin, and Denis Zorin. Piecewise smooth subdivision surfaces with normal control. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '00, pages 113–120, New York, NY, USA, 2000. ACM/Addison-Wesley. doi:10.1145/344779.344841.

[BM03] B. Buchwald and Günter Mühlbach. Construction of B-splines for generalized spline spaces generated from local ECT-systems. *Journal of Computational and Applied Mathematics*, 159(2):249–267, 2003. doi:10.1016/S0377-0427(03)00533-8.

[BM12] Martine Brilleaud and Marie-Laurence Mazure. Mixed hyperbolic/trigonometric spaces for design. *Computers and Mathematics with Applications*, 64(8):2459–2477, 2012. doi:10.1016/j.camwa.2012.05.019.

[BM14] Martine Brilleaud and Marie-Laurence Mazure. Design with L-splines. *Numerical Algorithms*, 65(1):91–124, 2014. doi:10.1007/s11075-013-9697-8.

[BMZ04] Ioana Boier-Martin and Denis Zorin. Differentiable parameterization of Catmull-Clark subdivision surfaces. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry processing*, SGP '04, pages 155–164, New York, NY, USA, 2004. ACM. doi:10.1145/1057432.1057453.

[BP97] Daniel Bister and Hartmut Prautzsch. A new approach to Tchebycheffian B-splines. In *Curves and Surfaces with Applications in CAGD*, pages 387–394, Nashville, TN, USA, 1997. Vanderbilt University Press.

[BTU03] Thierry Blu, Philippe Thévenaz, and Michael Unser. Complete parameterization of piecewise-polynomial interpolation kernels. *IEEE Transactions on Image Processing*, 12(11):1297–1309, 2003. doi:10.1109/TIP.2003.818018.

[CADS09] Thomas J. Cashman, Ursula H. Augsdörfer, Neil A. Dodgson, and Malcolm A. Sabin. NURBS with extraordinary points: high-degree, non-uniform, rational subdivision schemes. *ACM Transactions on Graphics*, 28(3):46:1–9, 2009. doi:10.1145/1531326.1531352.

[Cas10] Thomas J. Cashman. NURBS-compatible subdivision surfaces. Technical Report UCAM-CL-TR-773, Computer Laboratory, University of Cambridge, UK, March 2010. Ph.D. thesis. URL: http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-773.pdf.

[Cas11] Thomas J. Cashman. *NURBS-compatible Subdivision Surfaces*. British Informatics Society, 2011.

[CC78] Edwin Catmull and James Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design*, 10(6):350–355, 1978. doi:10.1016/0010-4485(78)90110-0.

[CDV96] Charles K. Chui and Johan M. De Villiers. Applications of optimally local interpolation to interpolatory approximations and compactly supported wavelets. *Mathematics of Computation*, 65(213):99–114, 1996. doi:10.1090/S0025-5718-96-00672-2.

[Chu90] Kai Ching Chu. B3-splines for interactive curve and surface fitting. *Computers & Graphics*, 14(2):281–288, 1990. doi:10.1016/0097-8493(90)90039-Z.

[CM06] Paolo Costantini and Carla Manni. Geometric construction of generalized cubic splines. *Rendiconti di Matematica e delle sue Applicazioni, Serie VII*, 26:327–338, 2006. URL: http://www1.mat.uniroma1.it/ricerca/rendiconti/2006(3-4)/327-338.pdf.

[CMP03] Jesús M. Carnicer, Esmeralda Mainar, and Juan Manuel Peña. Critical length for design purposes and Extended Chebyshev spaces. *Constructive Approximation*, 20(1):55–71, 2003. doi:10.1007/s00365-002-0530-1.

[CMSR98] Giulio Casciola, Serena Morigi, and Javier Sánchez-Reyes. Degree elevation for p-Bézier curves. *Computer Aided Geometric Design*, 15(4):313–322, 1998. doi:10.1016/S0167-8396(97)00034-4.

[Cos00] Paolo Costantini. Curve and surface construction using variable degree polynomial splines. *Computer Aided Geometric Design*, 17(5):419–446, 2000. doi:10.1016/S0167-8396(00)00010-8.

[CP94] Jesús M. Carnicer and Juan Manuel Peña. Totally positive bases for shape preserving curve design and optimality of B-splines. *Computer Aided Geometric Design*, 11(6):633–654, 1994. doi:10.1016/0167-8396(94)90056-6.

[CR74] Edwin Catmull and Raphael Rom. A class of local interpolants. In Robert E. Barnhill and Richard F. Riesenfeld, editors, *Computer Aided Geometric Design*, pages 317–326. Academic Press, New York, NY, USA, 1974.

[dB02] Carl de Boor. Spline basics. In Gerald Farin, Josef Hoschek, and Myung-Soo Kim, editors, *Handbook of Computer Aided Geometric Design*, pages 141–164. Elsevier Science, Amsterdam, The Netherlands, 2002.

[DFH09] Nira Dyn, Michael S. Floater, and Kai Hormann. Four-point curve subdivision based on iterated chordal and centripetal parameterizations. *Computer Aided Geometric Design*, 26(3):279–286, 2009. doi:10.1016/j.cagd.2008.09.006.

[DG85] Roger Delbourgo and John A. Gregory. Shape preserving piecewise rational interpolation. *SIAM Journal on Scientific and Statistical Computing*, 6(4):967–976, 1985. doi:10.1137/0906065.

[DGM88] Wolfgang Dahmen, T.N.T. Goodman, and Charles A. Micchelli. Compactly supported fundamental functions for spline interpolations. *Numerische Mathematik*, 52(6):639–664, 1988. doi:10.1007/BF01395816.

[DS78] D.W.H. Doo and Malcolm A. Sabin. Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*, 10(6):356–360, 1978. doi:10.1016/0010-4485(78)90111-2.

[DV12] Johan M. De Villiers. *Mathematics for Approximation.* Atlantis Press, Paris, France, 2012.

[Far02] Gerald Farin. *Curves and surfaces for CAGD: a practical guide.* Morgan Kaufmann, San Francisco, CA, USA, fifth edition, 2002.

[Far12] Rida T. Farouki. The Bernstein polynomial basis: A centennial retrospective. *Computer Aided Geometric Design*, 29(6):379–419, 2012. doi:10.1016/j.cagd.2012.03.001.

[FdB57] Francesco Faà di Bruno. Note sur une nouvelle formule de calcul différentiel. *The Quarterly Journal of Pure and Applied Mathematics*, 1:359–360, 1857. URL: http://gdz.sub.uni-goettingen.de/en/dms/loader/img/?PPN=PPN600494829_0001&DMDID=DMDLOG_0071.

[FHK02] Gerald Farin, Josef Hoschek, and Myung-Soo Kim, editors. *Handbook of Computer Aided Geometric Design.* Elsevier Science, Amsterdam, The Netherlands, 2002.

[Flo08] Michael S. Floater. On the deviation of a parametric cubic spline interpolant from its data polygon. *Computer Aided Geometric Design*, 25(3):148–156, 2008. doi:10.1016/j.cagd.2007.08.001.

[FS06] Michael S. Floater and Tatiana Surazhsky. Parameterization for curve interpolation. In Kurt Jetter, Martin D. Buhmann, Werner Haussmann, Robert Schaback, and Joachim Stöckler, editors, *Topics in Multivariate Approximation and Interpolation*, volume 12 of *Studies in Computational Mathematics*, pages 39–54. Elsevier Science, Amsterdam, The Netherlands, 2006. doi:10.1016/S1570-579X(06)80004-2.

[GP00] Laura Gori and Francesca Pitolli. A class of totally positive refinable functions. *Rendiconti di Matematica e delle sue Applicazioni, Serie VII*, 20:305–322, 2000. URL: http://www.dmmm.uniroma1.it/~francesca.pitolli/Ricerca/Rendiconti_2000.pdf.

[GU07] Ingo Ginkel and Georg Umlauf. Tuning subdivision algorithms using constrained energy optimization. In Ralph Martin, Malcolm A. Sabin, and Joab Winkler, editors, *Mathematics of Surfaces XII*, volume 4647 of *Lecture Notes in Computer Science*, pages 166–176. Springer Berlin Heidelberg, 2007. doi:10.1007/978-3-540-73843-5_11.

[Hay74] J. Geoff Hayes. New shapes from bicubic splines. Retyped by Malcolm A. Sabin in March 2010, as a note of the Rainbow Group (Computer Laboratory, University of Cambridge, UK), from a photocopy of the microfiche recording the proceedings of the CAD74 conference, 1974.

[HCB05] Thomas J.R. Hughes, J. Austin Cottrell, and Yuri Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39–41):4135–4195, 2005. doi: 10.1016/j.cma.2004.10.008.

[Her96] Thomas Hermann. G^2 interpolation of free form curve networks by biquintic Gregory patches. *Computer Aided Geometric Design*, 13(9):873–893, 1996. doi:10.1016/S0167-8396(96)00013-1.

[HKD93] Mark Halstead, Michael Kass, and Tony DeRose. Efficient, fair interpolation using Catmull-Clark surfaces. In *Proceedings of the 20th annual conference on Computer Graphics and interactive techniques*, SIGGRAPH '93, pages 35–44, New York, NY, USA, 1993. ACM. doi:10.1145/166117.166121.

[Joh02] Warren P. Johnson. The curious history of Faà di Bruno's formula. *American Mathematical Monthly*, 109(3):217–234, 2002. doi:10.2307/2695352.

[JS71] Joseph W. Jerome and Larry L. Schumaker. Local bases and computation of g-splines. *Methoden und Verfahren der Mathematischen Physik*, 5:171–199, 1971.

[JS76] Joseph W. Jerome and Larry L. Schumaker. Local support bases for a class of spline functions. *Journal of Approximation Theory*, 16(1):16–27, 1976. doi:10.1016/0021-9045(76)90092-7.

[Kar68] Samuel Karlin. *Total Positivity*, volume 1. Stanford University Press, Stanford, CA, USA, 1968.

[KFK94] Mitsuru Kuroda, Susumu Furukawa, and Fumihiko Kimura. Controllable locality in C^2 interpolating curves by B2-splines/S-splines. *Computer Graphics Forum*, 13(1):49–55, 1994. doi:10.1111/1467-8659.1310049.

[KK88] E. Kaufmann and R. Klass. Smoothing surfaces using reflection lines for families of splines. *Computer-Aided Design*, 20(6):312–316, 1988. doi:10.1016/0010-4485(88)90112-1.

[KL89] Per Erik Koch and Tom Lyche. Exponential B-splines in tension. *Approximation Theory VI*, 2:361–364, 1989.

[KP13] Kęstutis Karčiauskas and Jörg Peters. Non-uniform interpolatory subdivision via splines. *Journal of Computational and Applied Mathematics*, 240:31–41, 2013. doi:10.1016/j.cam.2012.07.004.

[KPR04] Kęstutis Karčiauskas, Jörg Peters, and Ulrich Reif. Shape characterization of subdivision surfaces: case studies. *Computer Aided Geometric Design*, 21(6):601–614, 2004. doi:10.1016/j.cagd.2004.04.005.

[KS99] Boris Ilyich Kvasov and Pairote Sattayatham. GB-splines of arbitrary order. *Journal of Computational and Applied Mathematics*, 104(1):63–88, 1999. doi: 10.1016/S0377-0427(98)00265-9.

[KSD13] Jiří Kosinka, Malcolm A. Sabin, and Neil Dodgson. Cubic subdivision schemes with double knots. *Computer Aided Geometric Design*, 30(1):45–57, 2013. doi: 10.1016/j.cagd.2012.06.004.

[KY06] E. B. Kuznetsov and A. Yu. Yakimovich. The best parameterization for parametric interpolation. *Journal of Computational and Applied Mathematics*, 191(2):239–245, 2006. doi:10.1016/j.cam.2005.06.040.

[LB07] Dylan Lacewell and Brent Burley. Exact evaluation of Catmull-Clark subdivision surfaces near B-spline boundaries. *Journal of Graphics, GPU, and Game Tools*, 12(3):7–15, 2007. doi:10.1080/2151237X.2007.10129243.

[LBW05] Hong-Wei Lin, Hu-Jun Bao, and Guo-Jin Wang. Totally positive bases and progressive iteration approximation. *Computers and Mathematics with Applications*, 50(3-4):575–586, 2005. doi:10.1016/j.camwa.2005.01.023.

[LC06] Shuhua Lai and Fuhua Cheng. Parametrization of general Catmull-Clark subdivision surfaces and its applications. *Computer-Aided Design & Applications*, 3(1–4):513–522, 2006. doi:10.1080/16864360.2006.10738490.

[Lee89] E. T. Y. Lee. Choosing nodes in parametric curve interpolation. *Computer-Aided Design*, 21(6):363–370, 1989. doi:10.1016/0010-4485(89)90003-1.

[Lev06a] Adi Levin. C^2 modified subdivision surfaces by optimal blending. http://www.math.tau.ac.il/~levin/adi/sig2006.htm, 2006. Addition to [Lev06b].

[Lev06b] Adi Levin. Modified subdivision surfaces with continuous curvature. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 1035–1040, New York, NY, USA, 2006. ACM. doi:10.1145/1179352.1141990.

[Lin10] Hongwei Lin. The convergence of the geometric interpolation algorithm. *Computer-Aided Design*, 42(6):505–508, 2010. doi:10.1016/j.cad.2010.01.006.

[Loo87] Charles Loop. Smooth subdivision surfaces based on triangles. Master's thesis, Department of Mathematics, University of Utah, August 1987. URL: http://research.microsoft.com/en-us/um/people/cloop/thesis.pdf.

[Loo02] Charles Loop. Bounded curvature triangle mesh subdivision with the convex hull property. *The Visual Computer*, 18(5):316–325, 2002. doi:10.1007/s003710100148.

[LS08] Charles Loop and Scott Schaefer. G^2 tensor product splines over extraordinary vertices. In *Proceedings of the 6th Eurographics Symposium on Geometry Processing*, SGP '08, pages 1373–1382, Aire-la-Ville, Switzerland, 2008. Eurographics Association.

[LW79] Tom Lyche and Ragnar Winther. A stable recurrence relation for trigonometric B-splines. *Journal of Approximation Theory*, 25(3):266–279, 1979. doi:10.1016/0021-9045(79)90017-0.

[Ma05] Weiyin Ma. Subdivision surfaces for CAD – an overview. *Computer-Aided Design*, 37(7):693–709, 2005. doi:10.1016/j.cad.2004.08.008.

[Maz08] Marie-Laurence Mazure. Which spaces for design? *Numerische Mathematik*, 110(3):357–392, 2008. doi:10.1007/s00211-008-0164-8.

[Maz11a] Marie-Laurence Mazure. How to build all Chebyshevian spline spaces good for geometric design? *Numerische Mathematik*, 119(3):517–556, 2011. doi:10.1007/s00211-011-0390-3.

[Maz11b] Marie-Laurence Mazure. On a general new class of quasi chebyshevian splines. *Numerical Algorithms*, 58:399–438, 2011. doi:10.1007/s11075-011-9461-x.

[Maz11c] Marie-Laurence Mazure. Quasi Extended Chebyshev spaces and weight functions. *Numerische Mathematik*, 118(1):79–108, 2011. doi:10.1007/s00211-010-0312-9.

[Maz12] Marie-Laurence Mazure. On a new criterion to decide whether a spline space can be used for design. *BIT Numerical Mathematics*, 52(4):1009–1034, 2012. doi: 10.1007/s10543-012-0390-4.

[MFR$^+$10] Kerstin Müller, Christoph Fünfzig, Lars Reusche, Dianne Hansford, Gerald Farin, and Hans Hagen. DINUS: Double insertion, nonuniform, stationary subdivision surfaces. *ACM Transactions on Graphics*, 29(3):25:1–21, 2010. doi:10.1145/1805964.1805969.

[MMN07] Takashi Maekawa, Yasunori Matsumoto, and Ken Namiki. Interpolation by geometric algorithm. *Computer-Aided Design*, 39(4):313–323, 2007. doi:10.1016/j.cad.2006.12.008.

[MP99] Esmeralda Mainar and Juan Manuel Peña. Corner cutting algorithms associated with optimal shape preserving representations. *Computer Aided Geometric Design*, 16(9):883–906, 1999. doi:10.1016/S0167-8396(99)00035-7.

[MP10] Esmeralda Mainar and Juan Manuel Peña. Optimal bases for a class of mixed spaces and their associated spline spaces. *Computers and Mathematics with Applications*, 59(4):1509–1523, 2010. doi:10.1016/j.camwa.2009.11.009.

[MRF06] Kerstin Müller, Lars Reusche, and Dieter Fellner. Extended subdivision surfaces: building a bridge between NURBS and Catmull-Clark surfaces. *ACM Transactions on Graphics*, 25(2):268–292, 2006. doi:10.1145/1138450.1138455.

[Müh06] Günter Mühlbach. ECT-B-splines defined by generalized divided differences. *Journal of Computational and Applied Mathematics*, 187(1):96–122, 2006. doi: 10.1016/j.cam.2005.03.040.

[MZ00] Weiyin Ma and Nailiang Zhao. Catmull-Clark surface fitting for reverse engineering applications. In *Proceedings of the Geometric Modeling and Processing 2000*, GMP '00, pages 274–283, Washington, DC, USA, 2000. IEEE Computer Society.

[MZ02] Weiyin Ma and Nailiang Zhao. Smooth multiple B-spline surface fitting with Catmull-Clark subdivision surfaces for extraordinary corner patches. *The Visual Computer*, 18(7):415–436, 2002. doi:10.1007/s003710100159.

[Nas01] Ahmad H. Nasri. Constructing polygonal complexes with shape handles for curve interpolation by subdivision surfaces. *Computer-Aided Design*, 33(11):753–765, 2001. doi:10.1016/S0010-4485(01)00093-8.

[Pet00] Jörg Peters. Patching Catmull-Clark meshes. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '00, pages 255–258, New York, NY, USA, 2000. ACM/Addison-Wesley. doi: 10.1145/344779.344908.

[PR98] Jörg Peters and Ulrich Reif. Analysis of algorithms generalizing B-spline subdivision. *SIAM Journal on Numerical Analysis*, 35(2):728–748, 1998. doi:10.1137/S0036142996304346.

[PR04] Jörg Peters and Ulrich Reif. Shape characterization of subdivision surfaces: basic principles. *Computer Aided Geometric Design*, 21(6):585–599, 2004. doi:10.1016/j.cagd.2004.04.006.

[PR08] Jörg Peters and Ulrich Reif. *Subdivision Surfaces*. Springer, 2008.

[Pra98] Hartmut Prautzsch. Smoothness of subdivision surfaces at extraordinary points. *Advances in Computational Mathematics*, 9(3-4):377–389, 1998. doi:10.1023/A:1018945708536.

[PS77] Michael James David Powell and Malcolm A. Sabin. Piecewise quadratic approximations on triangles. *ACM Transactions on Mathematical Software*, 3(4):316–325, 1977. doi:10.1145/355759.355761.

[PT95] Les Piegl and Wayne Tiller. *The NURBS book*. Springer-Verlag, London, UK, 1995.

[PU98] Hartmut Prautzsch and Georg Umlauf. A G^2-subdivision algorithm. In Gerald Farin, Hanspeter Bieri, Guido Brunnett, and Tony DeRose, editors, *Geometric Modelling, Dagstuhl, Germany, 1996*, volume 13 of *Computing Supplement*, pages 217–224. Springer Vienna, London, UK, 1998. doi:10.1007/978-3-7091-6444-0_17.

[Rab02] Christophe Rabut. On Pierre Bézier's life and motivations. *Computer-Aided Design*, 34(7):493–510, 2002. doi:10.1016/S0010-4485(01)00121-X.

[Rei95] Ulrich Reif. A unified approach to subdivision algorithms near extraordinary vertices. *Computer Aided Geometric Design*, 12(2):153–174, 1995. doi:10.1016/0167-8396(94)00007-F.

[RSA14] Lucia Romani, Laura Saini, and Gudrun Albrecht. Algebraic-trigonometric Pythagorean-hodograph curves and their use for Hermite interpolation. *Advances in Computational Mathematics*, 2014. To appear. doi:10.1007/s10444-013-9338-8.

[Sab91] Malcolm A. Sabin. Cubic recursive division with bounded curvature. In Pierre-Jean Laurent, Alain Le Méhauté, and Larry L. Schumaker, editors, *Curves and surfaces*, pages 411–414. Academic Press Professional, San Diego, CA, USA, 1991.

[SB03] Malcolm A. Sabin and Loïc Barthe. Artifacts in recursive subdivision surfaces. In Albert Cohen, Jean-Louis Merrien, and Larry L. Schumaker, editors, *Curve and Surface Fitting: Saint-Malo 2002*, pages 353–362. Nashboro Press, Brentwood, TN, USA, 2003. URL: http://www.irit.fr/recherches/VORTEX/publications/rendu-geometrie/BOOK2002_Sabin_Barthe.pdf.

[SCAD07] Malcolm A. Sabin, Thomas J. Cashman, Ursula H. Augsdörfer, and Neil A. Dodgson. Bounded curvature subdivision without eigenanalysis. In Ralph Martin, Malcolm A. Sabin, and Joab Winkler, editors, *Mathematics of Surfaces XII*, volume 4647 of *Lecture Notes in Computer Science*, pages 391–411. Springer Berlin Heidelberg, 2007. doi:10.1007/978-3-540-73843-5_24.

[Sch64] Isaac Jacob Schoenberg. On trigonometric spline interpolation. *Journal of Mathematics and Mechanics*, 13(5):795–825, 1964.

[Sch07] Larry L. Schumaker. *Spline Functions: Basic Theory.* Cambridge University Press, Cambridge, UK, third edition, 2007. doi:10.1017/CBO9780511618994.

[SDHI03] Malcolm A. Sabin, Neil A. Dodgson, Mohamed F. Hassan, and Ioannis P. Ivrissimtzis. Curvature behaviours at extraordinary points of subdivision surfaces. *Computer-Aided Design*, 35(11):1047–1051, 2003. doi:10.1016/S0010-4485(02)00113-6.

[SES04] Jordan Smith, Doug Epps, and Carlo Séquin. Exact evaluation of piecewise smooth Catmull-Clark surfaces using Jordan blocks. Technical report, University of California, Berkeley, CA, USA, January 2004. URL: http://www.jordansmith.org/pubs/pre_sharp.pdf.

[Sta98] Jos Stam. Exact evaluation of Catmull-Clark subdivision surfaces at arbitrary parameter values. In *Proceedings of the 25th annual conference on Computer Graphics and interactive techniques*, SIGGRAPH '98, pages 395–404, New York, NY, USA, 1998. ACM. doi:10.1145/280814.280945.

[SW10] Wanqiang Shen and Guozhao Wang. A basis of multi-degree splines. *Computer Aided Geometric Design*, 27(1):23–35, 2010. doi:10.1016/j.cagd.2009.08.005.

[SWZ04] Scott Schaefer, Joe Warren, and Denis Zorin. Lofting curve networks using subdivision surfaces. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, SGP '04, pages 103–114, New York, NY, USA, 2004. ACM. doi:10.1145/1057432.1057447.

[SZBN03] Thomas W. Sederberg, Jianmin Zheng, Almaz Bakenov, and Ahmad H. Nasri. T-splines and T-NURCCs. In *ACM SIGGRAPH 2003 Papers*, SIGGRAPH '03, pages 477–484, New York, NY, USA, 2003. ACM. doi:10.1145/1201775.882295.

[SZSS98] Thomas W. Sederberg, Jianmin Zheng, David Sewell, and Malcolm A. Sabin. Non-uniform recursive subdivision surfaces. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98, pages 387–394, New York, NY, USA, 1998. ACM. doi:10.1145/280814.280942.

[The01] Holger Theisel. Are isophotes and reflection lines the same? *Computer Aided Geometric Design*, 18(7):711–722, 2001. doi:10.1016/S0167-8396(01)00063-2.

[VRS11] Tamás Várady, Alyn Rockwood, and Péter Salvi. Transfinite surface interpolation over irregular n-sided domains. *Computer-Aided Design*, 43(11):1330–1340, 2011. doi:10.1016/j.cad.2011.08.028.

[VSR12] Tamás Várady, Péter Salvi, and Alyn Rockwood. Transfinite surface interpolation with interior control. *Graphical Models*, 74(6):311–320, 2012. doi:10.1016/j.gmod.2012.03.003.

[WF08] Guozhao Wang and Mei'e Fang. Unified and extended form of three types of splines. *Journal of Computational and Applied Mathematics*, 216(2):498–508, 2008. doi:10.1016/j.cam.2007.05.031.

[Woo87] Charles D. Woodward. B2-splines: a local representation for cubic spline interpolation. *The Visual Computer*, 3(3):152–161, 1987. doi:10.1007/BF01962896.

[WW01] Joe Warren and Henrik Weimer. *Subdivision Methods for Geometric Design: A Constructive Approach*. Morgan Kaufmann, San Francisco, CA, USA, 2001.

[Yam01] Yasushi Yamaguchi. A basic evaluation method of subdivision surfaces. *Journal for Geometry and Graphics*, 5(2):145–155, 2001. URL: http://www.heldermann-verlag.de/jgg/jgg01_05/jgg0515.pdf.

[YSK11] Cem Yuksel, Scott Schaefer, and John Keyser. Parameterization and applications of Catmull-Rom curves. *Computer-Aided Design*, 43(7):747–755, 2011. doi:10.1016/j.cad.2010.08.008.

[Zor06] Denis Zorin. Constructing curvature-continuous surfaces by blending. In Konrad Polthier and Alla Sheffer, editors, *Proceedings of the 4th Eurographics Symposium on Geometry Processing*, SGP '06, pages 31–40, Aire-la-Ville, Switzerland, 2006. Eurographics Association. URL: http://mrl.nyu.edu/~dzorin/papers/zorin2006ccc.pdf.