Genuine multi-sided parametric surface patches – A survey

Tamás Várady, Péter Salvi *, Márton Vaitkus

Budapest University of Technology and Economics, Budapest, Hungary

ARTICLE INFO

ABSTRACT

A state-of-the-art survey is presented on various formulations of multi-sided parametric surface patches, with a focus on methods that interpolate positional and cross-derivative information along boundaries.

1. Introduction

Modeling free-form surfaces is an important area in Computer Aided Geometric Design. The dominant part of related research describes four-sided parametric patches, such as Bézier and B-spline surfaces, or Coons patches and their variants. The term '*n*-sided' or 'multi-sided' surface generally refers to non-four-sided patches, indicating that alternative formulations are necessary, supplementing the classical tensor-product or Boolean sum approaches. In practical CAD/CAM, most operations involve four-sided patches, but their exclusive use may become inconvenient and cumbersome; it may degrade surface quality and require intricate artificial entities, such as splitting curves. It is evident that in many modeling situations the use of multi-sided surfaces is unavoidable. Amongst others, they emerge in aesthetic design of free-form objects, and are obtained through complex CAD operations such as lofting or blending. This has strongly inspired the development of various multi-sided surface representations.

Related research started in the golden era of CAGD in the 70s and 80s. First triangular patches were investigated, then 5- and 6-sided patches became the focus of interest. New surface types of crucial importance were introduced and later incorporated into commercial CAD/CAM systems. Several survey papers were published, including Várady (1987); Gregory et al. (1990); Malraison (1998, 2000); Karčiauskas and Krasauskas (2000); Peters (2003); Goldman (2004). The classical CAGD books, e.g. Hoschek and Lasser (1993) and Farin (2002) also devoted separate chapters to this topic.

While the momentum of research apparently slowed down in the 90s, the research community was aware of the deficiencies of the classical schemes, and this motivated the study of new, enhanced representations. In the last decade a large number of interesting multi-sided surface modeling techniques have been developed, which motivates the current survey paper. We intend to summarize the state-of-the-art on parametric, multi-sided surfacing schemes and attempt to impose some structure on the vast variety of representations from a set of unified viewpoints. Our survey is going to cover and analyze a large number of publications, and it is our hope that we were successful in not only including most of the relevant works, but also correctly interpreting their contents.

Our survey focuses on multi-sided surfaces employed within a *ribbon-based modeling* paradigm (Várady et al., 2011; Salvi et al., 2014). In this context, the term *ribbon* is fairly general; it defines boundaries and differential quantities to be interpolated *exactly*, ensuring geometric continuity (G^1, G^2, etc.) with neighboring surfaces (Peters, 2002; Kiciak, 2017). A ribbon-based multi-sided

---

* Corresponding author.
 E-mail address: salvi@iit.bme.hu (P. Salvi).
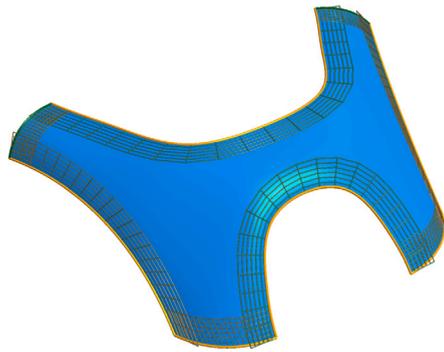
(a) Ribbons explicitly determined by given surfaces (vertex blending)
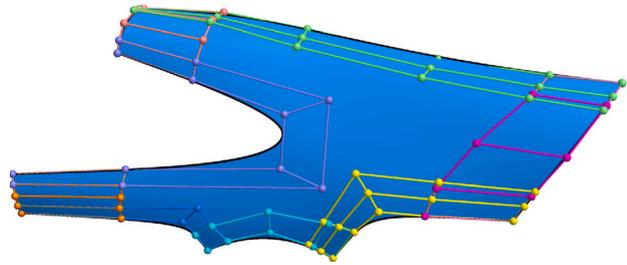
(b) Ribbons determined by grids of control points (curvenet-based design)
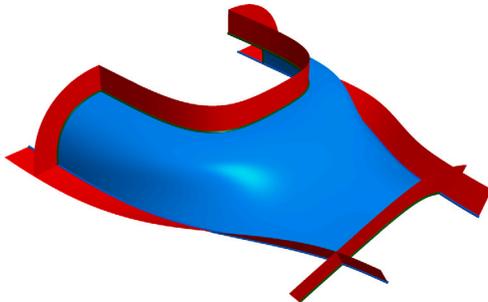
(c) Ribbons to be constrained by normal vector sweep (curvenet-based design)
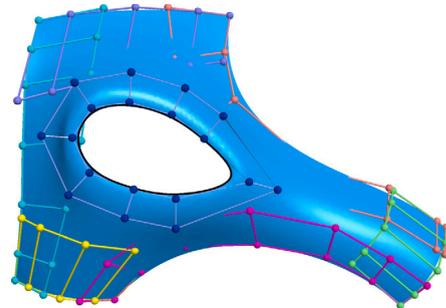
(d) Ribbons of a multiply connected patch

Fig. 1. Examples of ribbons and multi-sided patches.

patch is then generated *ab initio* by satisfying the given boundary constraints, with the possibility of additional interior control. In the rest of the introduction we would like to characterize where the ribbon-based approach has advantages over the dominant CAGD paradigms, i.e., modeling with *trimmed surfaces* and methods using *control polyhedra*.

In trimmed modeling, we generally work with an already existing four-sided surface, and a multi-sided patch is obtained through operations that indirectly define trimming boundaries. Accordingly, boundary constraints are typically inherited from the original surface, retaining its natural biparametric flow. Unfortunately, however, along trimming curves even watertight (i.e., C^0) connections to adjacent patches are achievable only in an *approximate* sense (see the recent survey of Marussig and Hughes, 2018). A recently published construction – boundary controlled trimming (Martin and Reif, 2022) – provides a promising solution to this problem, as presented in Section 6.1.

Methods for surface modeling based on control polyhedra include recursive subdivision (see survey in Cashman, 2012), macro-patches/generalized splines (Peters, 2019; Hughes et al., 2021), as well as 'manifold' approaches blending local surface pieces between overlapping 'charts' (Grimm and Zorin, 2006). Some subdivision and generalized spline surfaces have the advantage of being NURBS-compatible, but good shape can be difficult to achieve (Peters, 2003). Furthermore, classical subdivision methods do not interpolate predefined curve networks, though some solutions have been proposed over the years (Levin, 2000; Nasri and Sabin, 2002; Schaefer et al., 2004). Manifold-based blending can achieve a high degree of smoothness (even C^∞ in Ying and Zorin, 2004) while *approximating* a control polyhedron, and is similar to blending local interpolants (to be discussed in Section 4). Nevertheless, we argue that the patch-based formulations considered in this survey are preferable for curve network interpolation – see Southern and Dodgson (2007) for an attempt at reconciling these viewpoints.

Returning to the justification of *ribbon-based* modeling, there are two important areas where the ability to interpolate is crucial: (i) hole filling/vertex blending, and (ii) curve network based design. Related examples are shown in Fig. 1 with ribbons defined by (a) biparametric surfaces, (b) grids of control points, or (c) indirectly by means of non-parametric geometric quantities, such as normal vectors or curvature tensors. There may be ribbons in the patch interior, defining holes as seen in Fig. 1d. We will discuss ribbons in more detail in Section 3.1.

Multi-sided surface modeling is a complex task with often contradictory requirements and plenty of practical challenges. We are going to evaluate multi-sided patches with regard to degrees of geometric freedom, surface quality, ease of definition, smooth connections to adjacent patches, editing capabilities, efficiency of surface evaluation and different interrogations, and options to export through standard formats.

The paper is structured as follows. After classifying the major approaches in Section 2, we describe the most important components of multi-sided schemes in a general manner in Section 3. This will help us to discuss a wide range of specific techniques
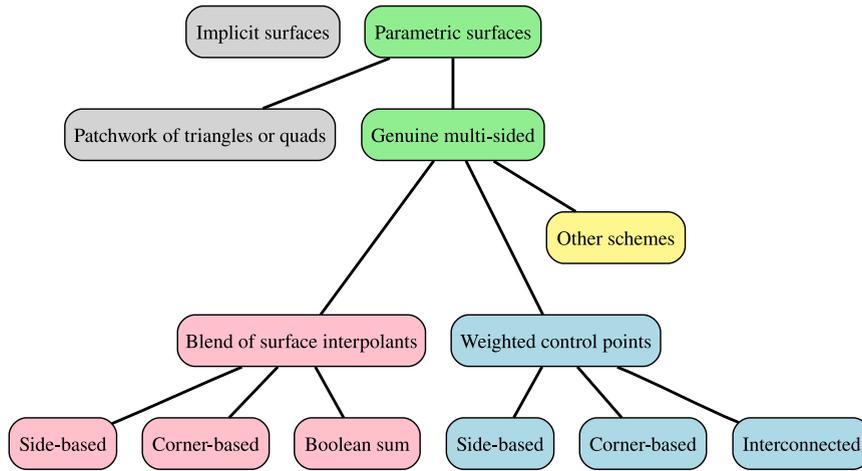
Fig. 2. Classification chart of multi-sided patches. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

in Sections 4–6. In Section 7 we will discuss the state of the art, and evaluate what has already been accomplished by the CAGD community. We close the survey with a list of challenges that remain for future research.

2. Classification

We are going to discuss multi-sided patches that interpolate well-defined boundaries with cross-derivative constraints. A related term – *transfinite interpolation* – was coined a long time ago, the first reference we have found is by Gordon and Hall (1973). The adjective 'transfinite' indicates methods that interpolate infinitely many points, in contrast to *discrete* interpolation methods that interpolate only a finite number of points – see Sabin (1996). Let us start with a classification of surfaces, see Fig. 2. While different divisions are possible, this particular view helps us to identify areas we wish to focus on; as for the remaining ones, only a few important references will be given.

As it is well-known, surfaces can be described either in *parametric* or *implicit* form. In the parametric case, generally (u, v) points of a given planar domain Γ are mapped onto a 3D surface $\mathbf{S}(u, v)$, i.e., $\mathbf{S} : \mathbb{R}^2 \supset \Gamma \to \mathbb{R}^3$. In some constructions the domain parameters are embedded into a multi-dimensional space, i.e., $\mathbf{S} : \mathbb{R}^n \supset \Gamma \to \mathbb{R}^3$. For implicit surfaces, the set of 3D points (x, y, z) satisfying a given equation $F(x, y, z) = 0$ constitute the surface, i.e., $F : \mathbb{R}^3 \to \mathbb{R}$. In this paper we focus on parametric patches, which dominate this area of research, though we would like to point out that implicit multi-sided constructions with various beneficial geometric and computational properties have also been published and have been used successfully for blending, hole filling, free-form or control polyhedron based design, and surface approximation. Important publications include Bloomenthal et al. (1997); Sederberg (1985); Bajaj et al. (1995); Hartmann (2001a); Várady et al. (2001). An up-to-date review of related work can be found in Sipos et al. (2020).

The term *genuine* multi-sided patch serves to distinguish schemes described by a single parametric equation, in contrast to those that are represented by a *collection* of smoothly connected triangular or quadrilateral patches. Research on composite patches started in the early eighties, and the related literature is huge. We refer to the fairly exhaustive surveys of Peters (2003, 2019) and Hughes et al. (2021), which present the state-of-the-art in this arena.

Following the diagram in Fig. 2, let us continue with genuine parametric patches, where we have identified three groups of surface constructions. The first group contains multi-sided surfaces that are composed of blended *interpolants*, i.e., a collection of biparametric surfaces that interpolate individual boundaries with cross-derivative constraints. In the multi-sided context, the interpolants are generally associated either with a *single* boundary, or with a *corner* (i.e., two consecutive boundaries of a patch), though interpolants with three constrained boundaries also exist (see later). Interpolant-based patches are generally defined as a linear combination of interpolants, weighted by appropriate blending functions to ensure interpolation along the boundaries. The classical Coons patch with the Boolean sum scheme (Coons, 1967) and its generalization, where side and corner interpolants are combined, also belong to this first group. In Sections 3 and 4 we will discuss how interpolants, correction surfaces, local parameterizations and various blending functions can be constructed.

The second group of patches is defined by means of a set of *control points*, and can be considered as generalizations of the classical tensor-product Bézier or B-spline patches. Patches in this group are defined as a combination of control points multiplied directly by corresponding blending functions, in contrast to the interpolant-based schemes where entire surfaces are blended together. The key issue is how to determine the structure of the control points and assign appropriate blending functions. There are side-based schemes, where control points associated with the boundaries determine the positional and differential behavior of the multi-sided patch, while in other cases the control points and the blend functions are grouped by corners. There are also more general control point structures – we refer to them with the adjective *interconnected* – where the control points are not directly associated with the boundaries, but cover the entire parametric domain. In Section 5.3 we will investigate various configurations, see Fig. 27 on page 21. For all schemes, it holds that the control points along and in the vicinity of the boundaries determine the positional and differential behavior of the multi-sided patch. These formulations will be discussed in Sections 3 and 5 in detail.

Ribbon constraints → Patch equation → Domain → Parameterizations → Blending functions → Editing capabilities
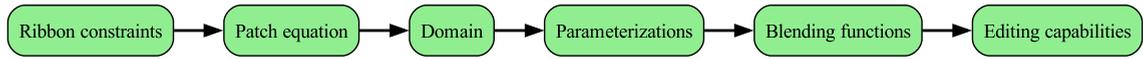
Fig. 3. Pipeline of multi-sided patch creation and modification.

The third group of surfaces, discussed in Section 6, is a collection of different schemes that do not belong to any of the above categories. First we deal with a special construction that blends an already existing, trimmed quadrilateral patch to given ribbons that *accurately* defines given trimming curves. Although surface creation and *trimming with boundary control* represent different modeling paradigms, the underlying mathematics shows certain similarities. This concept will be discussed in Section 6.1. In the remainder of Section 6, we are going to look at patches that *minimize energies* or satisfy differential equations, followed by formulations defined by *boundary integrals*.

A variety of parametric surface representations based on machine learning and neural networks have been proposed recently – see the recent work of Yang et al. (2023) for an overview. Instead of manual design, such methods are currently better suited for shape reconstruction and generative modeling applications, and are considered out of scope for this survey.

The above classification, while not always clear-cut, helps us to structure the rich variety of multi-sided patches as will be discussed in the following sections.

3. Basic components

From a bird's eye view, illustrated by the flowchart in Fig. 3, the majority of multi-sided constructions are created from six, strongly interrelated components, each involving a great variety of possible choices. Constructing a multi-sided surface is then like solving a puzzle: select a compatible set of components, and combine them in such a way that the resulting patch will satisfy the necessary constraints and produce a shape of acceptable quality. Our survey is about presenting already published combinations and evaluating their pros and cons.

3.1. Ribbons

As introduced earlier, ribbons represent input information associated with a given boundary to be interpolated. A ribbon can generally be expressed as a biparametric surface, which can be specified directly as user input, or by the Taylor expansion of an existing surface. Given a curve network, cross-derivative functions can be derived using the heuristic methods of e.g. Chiyokura (1986) and Salvi et al. (2023) for 1st, and Salvi and Várady (2014) for 2nd derivatives. Ribbons can also be defined using grids of control points representing tensor-product Bézier or B-spline surfaces.

Ribbons could also be determined by constraints on non-parametric geometric quantities, for example normal vector or curvature tensor fields associated and parameterized by the points of the boundary. From this geometric data a biparametric surface needs be created that satisfies the constraints approximately up to some tolerance. Note for example that the widely employed Rotation Minimizing Frames (RMFs) generally do not have closed-form expressions and need to be approximated by numerical methods (Wang et al., 2008). Some examples of ribbons were shown earlier in Fig. 1.

Ribbons may be used *directly* in certain surface representations, or the ribbon information needs to be *converted* into alternative structures that facilitate the application of more complex surface equations – see Fig. 4. In Section 4 we will describe interpolant-based schemes: ribbons can be used directly for one-sided (*side-based*) interpolants, but for two-sided (*corner-based*) interpolants some composition of ribbons is needed. Boolean sum constructions combine both side- and corner-based interpolants. Similarly, in Section 5 we will describe control point based representations: there are side-based schemes where the grid of control points is identical to the input, while others require regrouping the input control points around corners, or creating complex structures with control points that cover the patch interior.

We will also discuss multi-sided constructions in Section 6 where the ribbon-based logic is not directly applicable.

3.1.1. Ribbon compatibility and Gregory twists

At each corner of a ribbon-based patch two different cross-derivative functions must be interpolated at the same time. This might cause problems if the prescribed data is not compatible, e.g. when the mixed partial derivatives (*twists*) at the corner take different values, i.e., assuming a common (u, v) parameterization for ribbons $\mathbf{R}_{i-1}$ and $\mathbf{R}_i$, such that $u = v = 0$ at the corner, $\frac{\partial^2}{\partial u \partial v}\mathbf{R}_{i-1}(0,0) \neq \frac{\partial^2}{\partial u \partial v}\mathbf{R}_i(0,0)$. One solution might be to ensure compatibility of the ribbons, but the cyclic systems of constraints around n-valent vertices can be difficult to handle (Peters, 1991; Hermann et al., 2011, 2012). A commonly used alternative, known as the *Gregory twist* (Gregory, 1974), is to apply a rational blending to the different derivatives:

$$\mathbf{T}(u, v) = \frac{v}{u + v} \cdot \frac{\partial^2}{\partial u \partial v}\mathbf{R}_{i-1}(u, v) + \frac{u}{u + v} \cdot \frac{\partial^2}{\partial u \partial v}\mathbf{R}_i(u, v). \tag{1}$$

This expression evaluates to the twist of the $(i - 1)$-th or the i-th ribbon at the associated sides, i.e., when $u = 0$ or $v = 0$, respectively. At the corner point a singularity is introduced, where the value of the blended derivative depends on the direction from which the corner is approached. The construction can also be generalized to higher derivatives (Hermann, 1996).
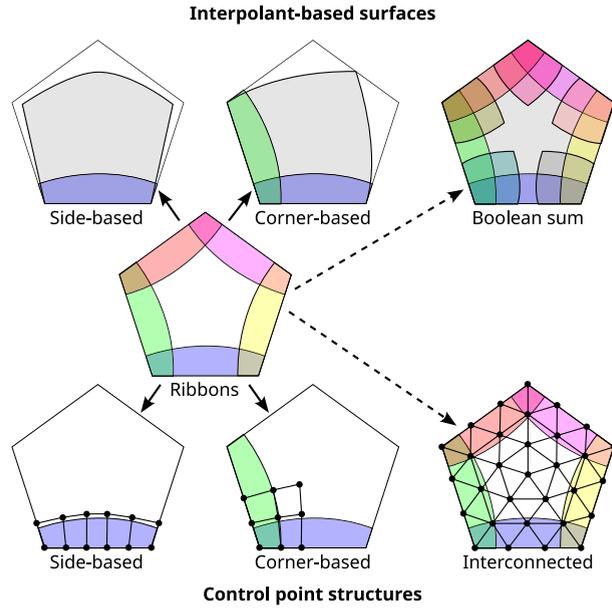
Fig. 4. Ways to create interpolants and control point structures from ribbons.

3.2. Surface equations

In this section we formulate surface equations in general; the methods of domain computation, appropriate parameterizations and blending functions will be discussed in the forthcoming sections. Note that while surfaces will be defined in terms of 2D domain coordinates denoted by (u, v), their components (e.g. interpolants and blend functions) might be defined in terms of local parameters that we chose to omit here to simplify our notation.

3.2.1. Interpolant-based patches

The general equation of a *side-based patch* is

$$\mathbf{S}(u,v) = \sum_{i=1}^{n} \mathbf{S}_i^{\text{Int}} \cdot W_i. \tag{2}$$

Here $\mathbf{S}_i^{\text{Int}}$ denotes a side-interpolant – generally a biparametric surface – defined in terms of local parameters associated with the i-th side. W_i is the i-th *weight* or *blend* function defined by a set of local variables, as will be explained later.

The general equation of a *corner-based patch* is

$$\mathbf{S}(u,v) = \sum_{i=1}^{n} \mathbf{S}_{i-1,i}^{\text{Int}} \cdot W_{i-1,i}. \tag{3}$$

$\mathbf{S}_{i-1,i}^{\text{Int}}$ denotes a corner-interpolant (generally also a biparametric surface of two local parameters associated with the i-th corner), and $W_{i-1,i}$ denotes a *corner blend* function.

The general equation of a *Boolean sum patch* is

$$\mathbf{S}(u,v) = \sum_{i=1}^{n} \mathbf{S}_i^{\text{Int}} \cdot W_i - \sum_{i=1}^{n} \mathbf{S}_{i-1,i}^{\text{Corr}} \cdot W_{i-1,i}. \tag{4}$$

These patches combine n side-interpolants as well as n corner-based *correction surfaces* that need to be subtracted, following the rules of Boolean algebra (Lancaster and Šalkauskas, 1986). These are inspired by the pioneering work of Coons (1967) – see Fig. 5. This approach was generalized in the above form by Várady et al. (2011).

3.2.2. Control point based patches

The general equation for a side-based control point patch is

$$\mathbf{S}(u,v) = \sum_{i=1}^{n} \sum_{j=0}^{m_i} \sum_{k=0}^{m_i^{\perp}} \mathbf{C}_{j,k}^i \cdot W_{j,k}^i. \tag{5}$$
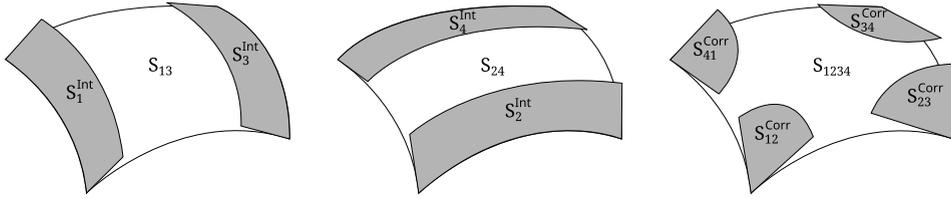
Fig. 5. The classical C^1 Coons patch combines three interpolants $\mathbf{S} = \mathbf{S}_{13} \oplus \mathbf{S}_{24} = \mathbf{S}_{13} + \mathbf{S}_{24} - \mathbf{S}_{1234}$ and applies cubic Hermite functions. This can be generalized for $n \geq 4$ by reinterpreting the constituent surfaces as the blended sum of side interpolants and corner correction surfaces.

| (a) By sides | (b) By corners | (c) Distributed over the full domain |
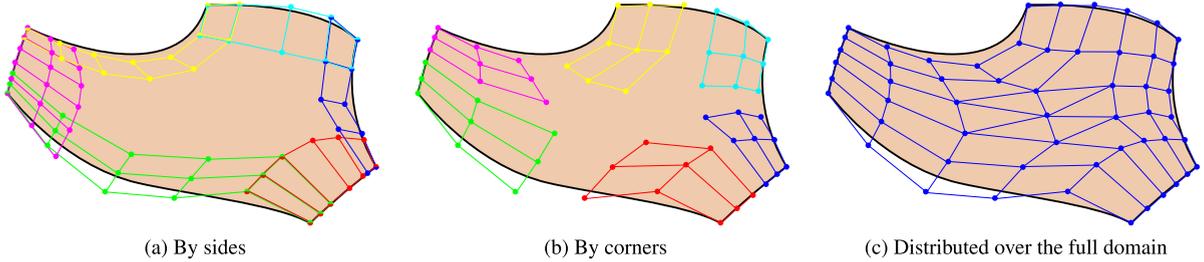
Fig. 6. Control point configurations.

The control points $\mathbf{C}^i_{j,k}$ and the blends $W^i_{j,k}$ are associated with the i-th side of the patch, and are arranged into grids indexed by j and k, see also Fig. 6a. The size of the i-th grid in the side- and cross-directions are denoted by m_i and $m_i^\perp$, respectively, which might be different for the individual sides. These ribbons typically consist of rows of control points prescribing G^e continuity conditions at the boundary ($e \leq m_i^\perp$), and the blending functions $W^i_{j,k}$ generally ensure some degree of compatibility with tensor-product Bézier or B-spline patches.

The general equation for a corner-based control point patch is

$$\mathbf{S}(u,v) = \sum_{i=1}^{n} \sum_{j=0}^{m_i} \sum_{k=0}^{m_i} \mathbf{C}^{i-1,i}_{j,k} \cdot W^{i-1,i}_{j,k}. \tag{6}$$

The control points $\mathbf{C}^{i-1,i}_{j,k}$ and the blends $W^{i-1,i}_{j,k}$ are associated with the i-th corner of the patch, using certain subsets of the control points from the $(i-1)$-th and i-th sides, see Fig. 6b.

Interconnected patches are defined by structures of control points that cover the full domain, see Fig. 6c. Their general equation is

$$\mathbf{S}(u,v) = \sum_{i \in I} \mathbf{C}_i \cdot W_i, \tag{7}$$

where I is a set of indices. In contrast to the previous two schemes where the control points were associated with sides or corners, here the patch is defined by a complex graph of control points with related blend functions.

3.3. Domain

The surfaces we consider are always defined over a *parametric domain*, a (possibly multiply connected) subset of the Euclidean plane. The domain boundary is assumed to be composed of curve segments called *sides,* meeting at *corners* that can be convex, concave, or smooth (with 180-degree angle). Choosing a suitable domain is a critical issue when creating multi-sided surfaces. Highly complex boundary configurations might require a domain shape of comparable complexity. As a general rule, the domain should geometrically 'mimic' the surface shape as much as possible, since parametric distortion and unexpected shape artifacts are then alleviated. From a topological point of view, the number of 3D boundary curves needs to be equal to the number of sides of the domain; and if we have a multiply connected surface, the number of 3D hole loops needs to be equal to those within the domain.

In correlation with the boundary configurations, Fig. 7 depicts domains according to geometric complexity. Three- and four-sided surfaces are typically defined over triangles or rectangles. The great majority of published multi-sided formulations work over regular *convex polygonal* domains. Non-regular convex polygons may be necessary, when the lengths of the boundaries differ to a large extent. *Concave* polygonal domains are recommended, when adjacent 3D boundary curves meet in concave angles at the corners. The benefits of *curved* domains have come forward in the last few years, in particular for patches that have highly curved boundaries. Naturally, for multiply connected patches domains with interior hole loops should be used. We remark that in rare cases 1- or 2-sided domains may also occur, see constructions for example in Sabin (1996); Várady et al. (2012).

The following two examples illustrate how the shape of a domain imposes limits on the possible surface geometries that can be represented by a given scheme. In Fig. 8a, a surface, whose boundary loop has two *concave corners,* needs to be defined. Using a
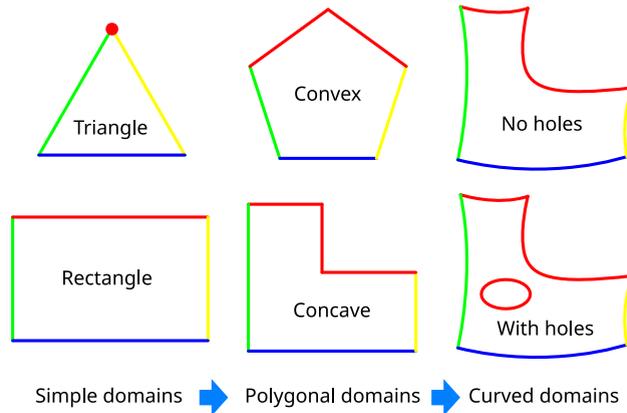
Fig. 7. Domain types. Edge coloring shows the mapping of a quadrilateral interpolant associated with the bottom side.

(a) Convex vs. concave domains

(b) Convex vs. curved domains
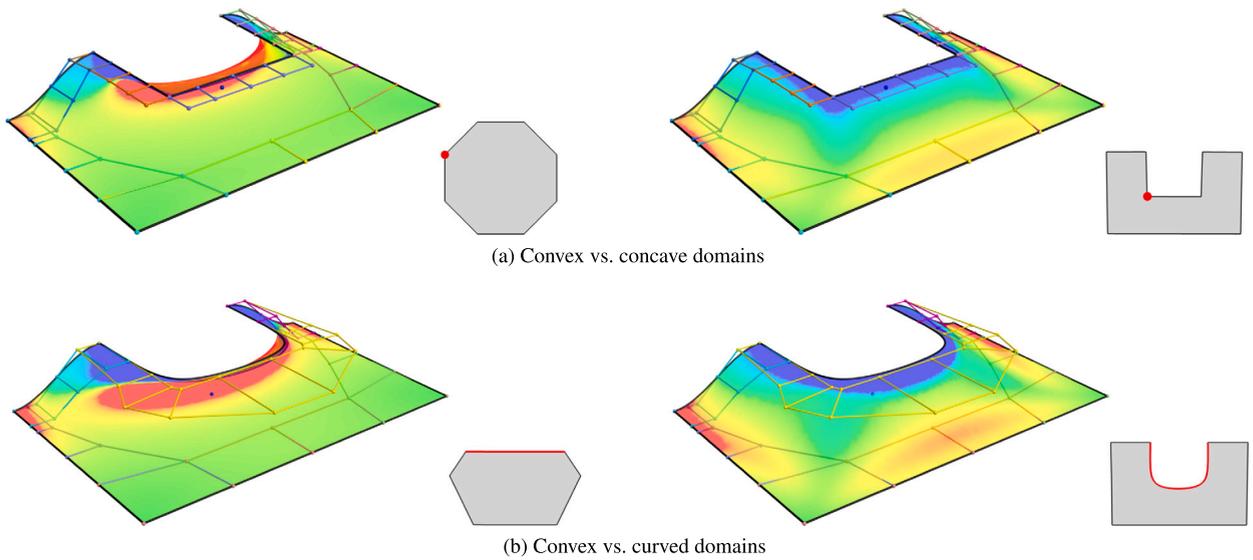
Fig. 8. Comparisons of different domain types; surface coloring indicates mean curvature.

convex polygon will locally turn the patch inside out, yielding an unacceptable result at the corners. The use of a concave domain polygon, however, allows a satisfactory resolution of the problem (see curvature maps). In Fig. 8b, a surface with a *strongly curved boundary* needs to be defined. Using a polygonal domain, this would be mapped to a straight edge of a convex polygon, leading to a surface that folds under itself. Using a curved domain, however, makes it possible to create a patch with more reasonable shape from the same inputs.

As the vast majority of multi-sided constructions are based on convex polygonal domains, the problem of domain creation has been largely disregarded in the literature. Algorithms for generating irregular convex polygons from 3D boundary curves were discussed in Várady et al. (2011). The simplest method is to place the vertices of the domain on the unit circle, where the angles are proportional to the arc lengths of the boundary curves. An alternative is to make the edge lengths of the cyclic polygon proportional to the arc lengths (provided that no boundary is longer than the others combined). In another approach, the angles between the boundary curves at the corners of the loop are also taken into consideration. Since this last algorithm has been extended to concave polygons (Salvi and Várady, 2018) and curved domains (Várady et al., 2020), it deserves to be explained in detail.

The basic idea is to first create an open polygon from segments with the same lengths as the associated 3D boundary curves, and then close the polygon by distributing the deviation vector **e** between the first and last points, see Fig. 9a. In the first phase, the angles between the segments are proportional to those between the boundary curves, normalized such that their sum is $(n-2)\pi$. In the second, closing-up phase, the distribution of the deviation can be either uniform or weighted by the segment lengths. This algorithm is applicable to the concave case, as well (Fig. 9b), with the caveat that it may produce polygons with very narrow bottlenecks or even self-intersections. In such cases, Salvi and Várady (2018) propose an iterative enlargement procedure.

The above concept can be extended to curved domains in a natural way, as discussed in Várady et al. (2020): the boundary curves are discretized into polylines, and at each point $\mathbf{p}_i$, the adjacent points $\mathbf{p}_{i-1}$ and $\mathbf{p}_{i+1}$ are projected onto the tangent plane (known

(a) Distribution of the deviation vector **e**   (b) Generating a concave domain   (c) Local flattening
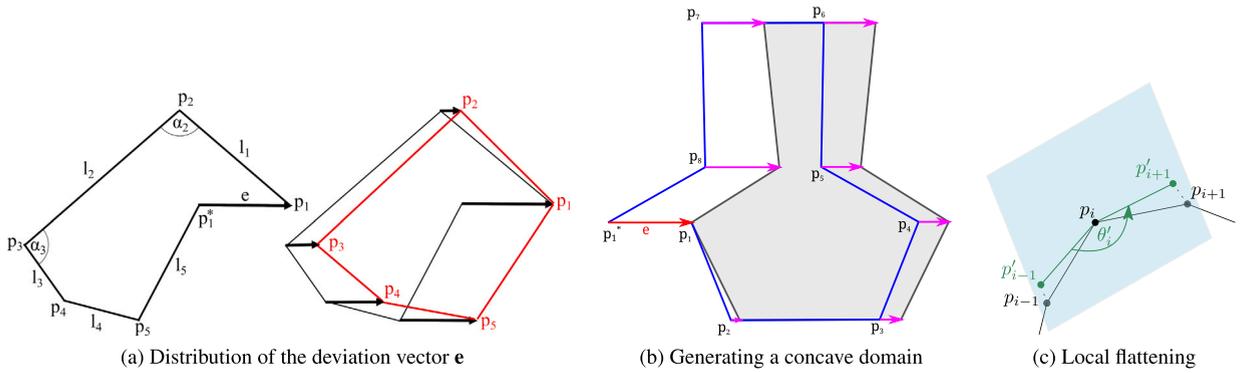
Fig. 9. Domain generation based on the arc lengths and angles of the boundary configuration.

from the boundary ribbon constraint), see Fig. 9c. This approach aims to preserve the shape of the curves 'within the surface', i.e., the local geodesic curvature.

In multiply connected domains the same algorithm can be used for interior loops, but then these need to be positioned and oriented inside the domain. Várady et al. (2020) suggest first creating a provisional patch without holes, and projecting and aligning the interior polylines to determine the associated domain coordinates.

We remark that other options also exist for domain generation. A common procedure is to project the boundary curves on a best-fit plane, but this can only be applied when the configuration is close to planar, see counter-examples in Salvi and Várady (2018). Instead of flattening the given 3D boundaries, one may first connect them by creating a coarse triangular mesh (Zou et al., 2013), and then the domain can be computed by simply flattening this mesh using standard methods for mesh parameterization (Hormann et al., 2008).

3.4. Parameterizations

The determination of local parameters is arguably the most crucial part of multi-sided patch generation, as it indirectly determines how the constituents of the patch are combined at a given domain point (u, v) when a surface point $\mathbf{S}(u, v)$ is evaluated. The shape of the domain designates the set of possible parameterization methods, which, in turn, have a fundamental influence on patch quality.

For interpolant-based patches, local parameters are used for *both* the interpolants and the blending functions, i.e., for each point in the domain we compute a set of associated points on the interpolants, and combine them by corresponding blending functions. For control point based patches local parameters are used *only* for the blending functions.

The side and corner interpolants in Eqs. (2)–(4), given as $\mathbf{S}_i^{\mathrm{Int}}(h_{i1}, h_{i2})$ or $\mathbf{S}_{i-1,i}^{\mathrm{Int}}(h_{i1}, h_{i2})$, are biparametric surfaces with side and corner parameters, respectively, where $h_{i1} = h_{i1}(u, v)$ and $h_{i2} = h_{i2}(u, v)$ denote the local parameter mappings. The *weight* or *blend* functions in Eqs. (2)–(7), given as $W_i(u, v) = W_i(h_1, h_2, \ldots, h_l)$, are evaluated at a set of local parameters $h_i = h_i(u, v)$, $i = 1 \ldots l$. We will see that there are blend functions that utilize all the local parameters ($l = n$), while others use only $l = 2$ or 4 local parameters.

3.4.1. Types of parameterizations

As discussed previously, most multi-sided constructions rely on local parameterizations that map the quadrilateral domain (*quad* for short) associated with a given side or corner onto a multi-sided domain. In case of side-based constructions this mapping can be done in a variety of ways, see Figs. 10a–10e. Coloring the edges of the unit square – base (blue), left (green), right (yellow) and opposite (red) – its image may (i) extend beyond the domain (*extended* coverage), or (ii) fit the domain perfectly (*full* coverage), or (iii) cover only a portion of it (*partial* coverage). The left and right sides of the quad are generally mapped to the $(i - 1)$-th and $(i + 1)$-th sides of the domain. In the case of full coverage, the opposite edge of the quad is mapped to all 'remote' sides ($j \neq i - 1, i, i + 1$).

Similarly, for corner-based mappings, three options exist, see Figs. 10f–10i. We can associate the $(i - 1)$-th and i-th sides with the left (green) and bottom (blue) edges of the quad. In the case of *extended* coverage we can disregard the mapping beyond the domain boundaries, as long as the associated interpolant will only be evaluated in its interior. In the *full* coverage case, the two opposite edges of the quad will be mapped onto the $(i - 2)$-th and $(i + 1)$-th sides, and the top-right corner of the quad is mapped onto the remaining sides of the domain ($j \neq i - 2, i - 1, i, i + 1$). *Partial* coverage is a possibility for corner-based parameters, as well.

3.4.2. Constructions of parameterizations

There is a great variety of methods for computing domain parameters, here we describe a few representative solutions.

Side and distance parameters are associated with a given side of the domain; the notations s_i and d_i will be used for the side and distance parameters of side i, respectively. The distance parameter d_i represents some distance measure between the given domain point (u, v) and a related footpoint $\mathbf{f}_i(s_i)$ on the i-th side of the domain. The distance parameter is zero along the corresponding side and increases in a monotone manner as we move towards the interior. Generally, d_i increases linearly along the $(i - 1)$-th and $(i + 1)$-th sides. The side parameter s_i measures a normalized distance between the 'left' corner point and a given footpoint on the

(a) Side-based parameterization   (b) Extended$_1$ coverage   (c) Extended$_2$ coverage   (d) Full coverage   (e) Partial coverage

(f) Corner-based parameteriza-   (g) Extended coverage   (h) Full coverage   (i) Partial coverage
tion (opposite corner in cyan)
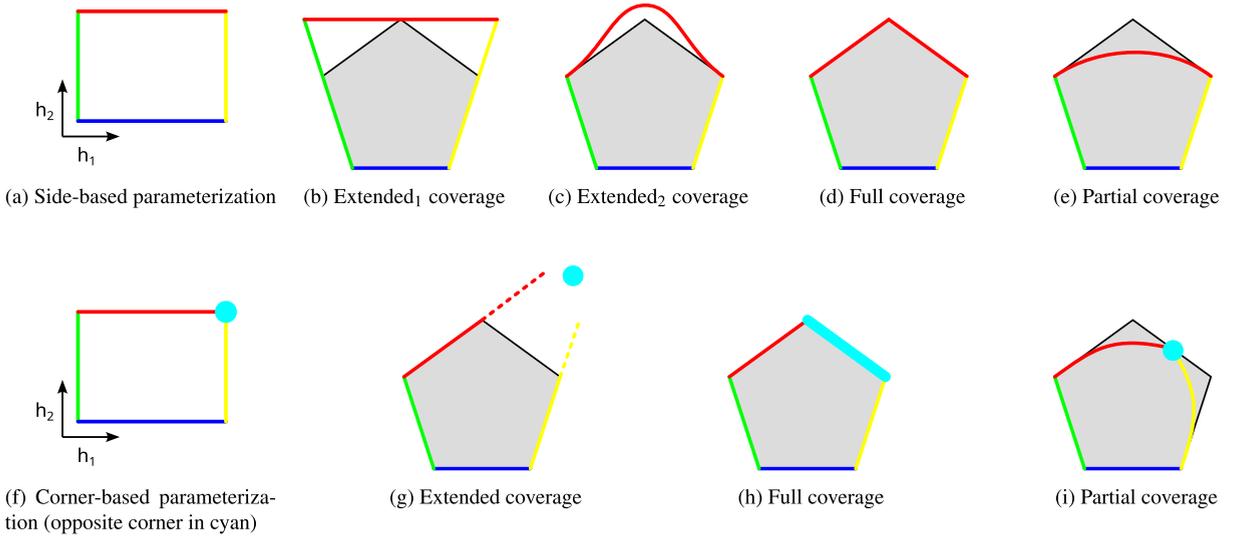
Fig. 10. Types of parametric coverage in the domain of side- and corner-based patches.

(a) Orthogonal distances   (b) Sweeping lines   (c) Biparametric mapping
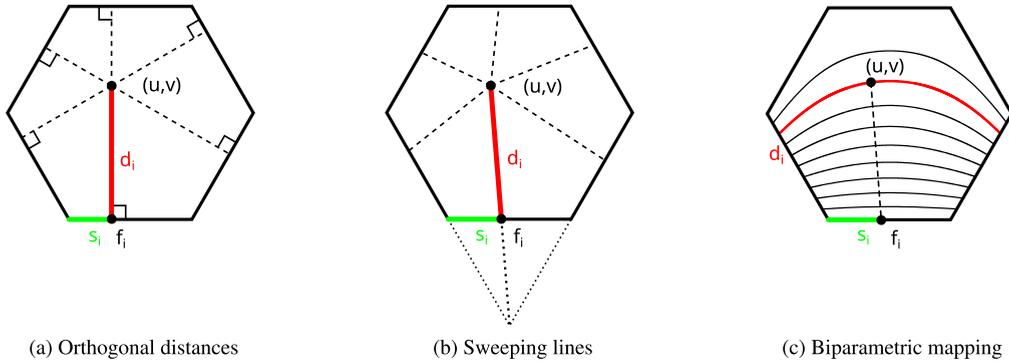
Fig. 11. Different parameter computations.

i-th side, i.e., $0 \le s_i \le 1$. We will see in the next section that there are constructions that require only distance parameters, or only side parameters, or some combination of both.

Fig. 11 depicts three examples. The first shows orthogonal projections onto the side, i.e., Euclidean distances to the footpoints. The second illustrates parameterization based on sweeping lines – pick a line that goes through point (u, v), and compute the intersection with side i to obtain footpoint $\mathbf{f}_i$. Then the related (normalized) distances yield the parameters s_i and d_i, respectively. Note that the well-known *radial sweep* method that forces the sweeping lines with index i to go through the intersection point of the extended $(i-1)$-th and $(i+1)$-th chords also belong to this category. The third example shows a map, which is defined not through explicit geometric entities, but a biparametric function. In case of concave or curved domains simple orthogonal projection or sweeping line constructions are not applicable, and more complex parameterization methods are needed. (An exception is when boundaries have an implicit representation, which might be sufficient to define parameters for some patches, e.g. Rvachev et al., 2001; Sabin et al., 2022.)

Generalized barycentric coordinates (GBC) represent a fundamental technique for constructing parameterizations. Detailed reviews of this topic can be found in Floater (2015) and Hormann and Sukumar (2017). Generalized barycentric coordinates λ_i $(i = 1 \ldots n)$ represent monotonically decreasing functions taking the value 1 at the associated corner point $\mathbf{p}_i$ of the domain, and vanishing at all other corners. Recall the following important properties:

$$\sum_{i=1}^{n} \lambda_i(u, v) = 1, \qquad \text{(partition of unity)} \tag{8a}$$

$$\sum_{i=1}^{n} \lambda_i(u, v)\mathbf{p}_i = (u, v), \qquad \text{(vertex reproduction)} \tag{8b}$$

$$\lambda_i(\mathbf{p}_j) = \delta_{ij}, \qquad \text{(Lagrange property)} \tag{8c}$$

(a) Generalized barycentric coordinate λ_2

(b) (s, d) parameters in a convex domain
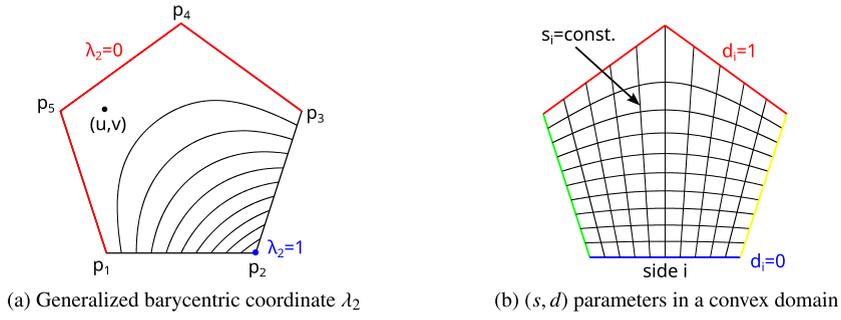
Fig. 12. Coordinates on a convex domain.

(a) (s, d) parameters in a concave domain

(b) (s, d) parameters in a curved domain with a hole
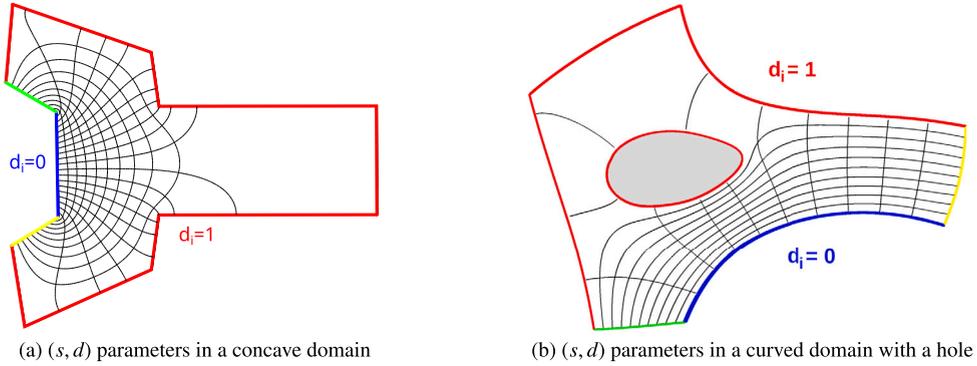
Fig. 13. Isolines of harmonic side and distance parameters.

where δ_{ij} is the Kronecker delta. On convex (and in some cases, concave) domains the coordinates are also non-negative, i.e., $\lambda_i \geq 0$ ($i = 1 \ldots n$). These properties can be exploited by treating $1 - \lambda_i$ as a distance measure between (u, v) and the corner $\mathbf{p}_i$. Fig. 12a shows isolines associated with the corner $\mathbf{p}_i$, using Wachspress coordinates (Wachspress, 1975).

Side and distance parameters from GBC-s. Barycentric coordinates can be used for the construction of evenly distributed side and distance parameters, as was suggested by Várady et al. (2016):

$$s_i(u, v) = \frac{\lambda_i}{\lambda_{i-1} + \lambda_i}, \qquad d_i(u, v) = 1 - \lambda_{i-1} - \lambda_i. \tag{9}$$

Fig. 12b shows (s_i, d_i) isolines within a convex polygon. The s_i lines sweep from the $(i-1)$-th side to the $(i+1)$-th, the d_i lines spread from the base side up to the opposite sides of the domain. One essential advantage of the GBC-based techniques is that they can easily be extended to concave, curved and even multiply connected domains. While some GBCs, like Wachspress coordinates, are strictly limited to convex polygons, others, such as Mean Value coordinates (Floater, 2003), are defined over more general domains (Hormann and Floater, 2006), but might take on negative values and have local minima. However, *harmonic coordinates* (Joshi et al., 2007), defined as solutions (with suitable boundary conditions) of the *Laplace equation*

$$\Delta \lambda_i(u, v) = \left(\frac{\partial^2}{\partial u^2} + \frac{\partial^2}{\partial v^2} \right) \lambda_i(u, v) = 0, \tag{10}$$

or, equivalently, as minimizers of the *Dirichlet energy*

$$\int \left\| \nabla \lambda_i(u, v) \right\|^2 du\, dv, \tag{11}$$

are guaranteed to be positive and without local minima over arbitrary domains, and using them within the above formulae, nice parametric mappings can be produced, see Fig. 13 and Salvi and Várady (2018); Várady et al. (2020). A closely related parameterization technique that also handles multiply connected domains is to compute s or d directly as a harmonic function that interpolates appropriate boundary conditions, see Vaitkus et al. (2021) and also the discussion in the next paragraph. Harmonic functions do not have a closed-form solution and their evaluation is possible only via numerical approximations, such as finite differences (Joshi et al., 2007), finite elements (Várady et al., 2020), boundary elements (Rustamov, 2008), or Monte Carlo methods (Sawhney and Crane, 2020). Other GBCs suitable for general domains include Maximum Entropy (Hormann and Sukumar, 2008), Iterative (Deng et al., 2020), and Maximum Likelihood (Chang et al., 2023) coordinates.

Biparametric mappings from PDEs or optimization. The problem of parameterization of planar domains – i.e., constructing a bijective mapping onto the unit square – has been studied at great depth within the context of quadrilateral mesh ('grid') generation (Liseikin, 2017). Transfinite interpolation methods, such as 2D Coons patches (Gordon and Hall, 1973; Gravesen et al., 2014), have also been

(a) Biparametric differential constraints

(b) (s, d) w/o constraint (c) (s, d) w/constraint (d) Sweep w/o constraint (e) Sweep w/constraint
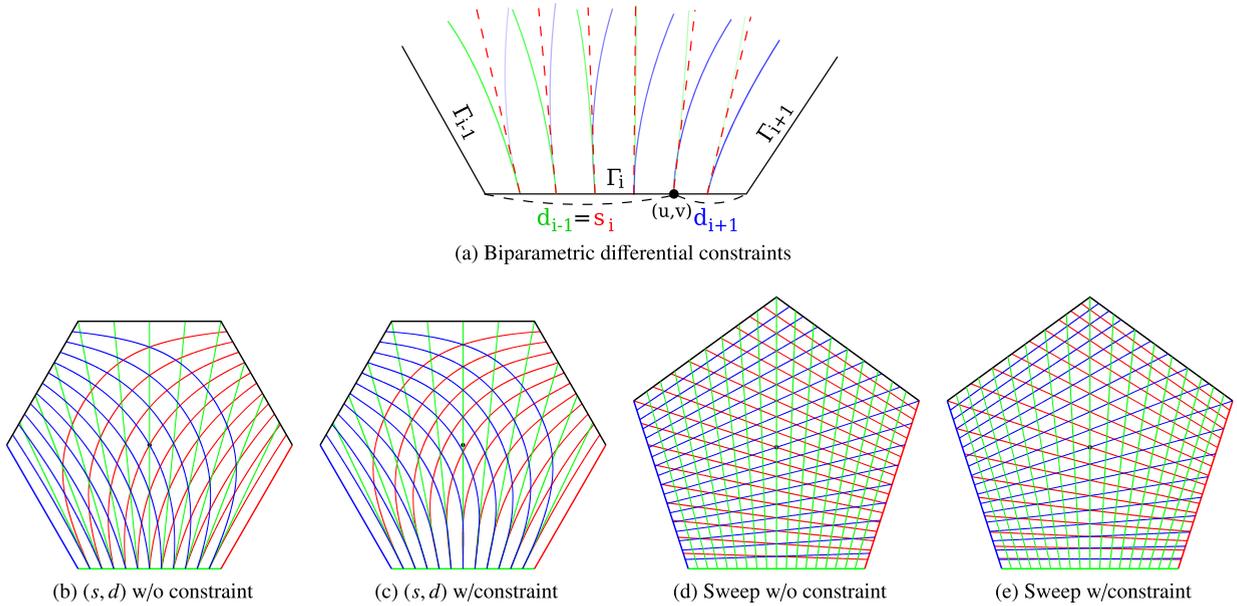
Fig. 14. Differentially constrained parameterizations.

used for this purpose. The aforementioned harmonic parameters of Vaitkus et al. (2021) belong to the family of Elliptic Grid Generation methods (Hinz et al., 2018). Harmonic boundary value problems can also be adapted to periodic hole loops within multiply connected domains, in analogy with methods for general topology surface parameterization (Aigerman et al., 2015). More generally, finding bijective maps between 2D domains involves iterative optimization of the mapping distortion – we refer to Garanzha et al. (2021) for an overview of this area.

Side and distance parameters with derivative constraints. There are constructions where further constraints need to be enforced on the local parametric variables to ensure the desired differential behavior along the patch boundary. An example is that the transversal derivatives of the $(i-1)$-th and $(i+1)$-th distance parameters at the i-th boundary must be equal to that of the side parameter, i.e., $d'_{i-1} = s'_i = -d'_{i+1}$, see e.g. Salvi et al. (2014) and Figs. 14a–14c. Another example is when G^1 or G^2 continuity of a corner-interpolant based patch is established through a constraint of the left and right side parameters, i.e., $s'_{i-1} = s'_{i+1}$, see e.g. Várady (1991); Salvi and Várady (2014) and Figs. 14d–14e.

N-dimensional parameter set with constrained equations. So far, we have described parameterizations determined by direct formulae. It is also possible to implicitly define a set of local coordinates by means of non-linear equations. A simple example due to Sabin (1983) is the following: the barycentric coordinates in a triangle must be normalized, i.e., $\lambda_1 + \lambda_2 + \lambda_3 = 1$, but we may define a parameterization satisfying an alternative constraint equation $\lambda_1 + \lambda_2 + \lambda_3 - 2\lambda_1\lambda_2\lambda_3 = 1$ (see details in Section 5.3.1). This idea can be extended for $n = 5, 6$, so the parameterization is defined by means of a 2D surface embedded in n-dimensional space.

3.5. Blending functions

The main purpose of blending functions is to assign weights to the constituents of the surface – interpolants or control points – while ensuring reproduction of positional and cross-derivative constraints along the boundaries.

3.5.1. Blends for interpolants

Blends with one non-zero side. In this case, the side interpolants $\mathbf{S}_i^{\text{Int}}$ are often identical to the ribbons. They are weighted by blends W_i^* that guarantee that the i-th ribbon (i) is reproduced on side i, and (ii) has no effect – in either positional or differential sense – on other sides of the domain ($j \neq i$), i.e., W_i^* is 1 on side i and 0 elsewhere. One possible solution is to apply a variation of Shepard's inverse distance weights (Shepard, 1968), as was suggested by several authors (Barnhill, 1977; Gregory, 1986; Kato, 2000; Rvachev et al., 2001; Gao and Rockwood, 2005; Várady et al., 2011):

$$W_i^* = \frac{1/d_i^e}{\sum_{j=1}^n 1/d_j^e}. \tag{12}$$

Exponent e sets the degree of continuity (G^{e-1}) for reproduction. Such a blend function is depicted in Fig. 15a. When $d_j = 0$ for some $j = 1 \ldots n$, the equation above seems to be singular, but this can be circumvented by multiplying both the numerator and the denominator by $\prod_{k=1}^n d_k^e$:

(a) Singular blend   (b) Corner blend   (c) Side blend
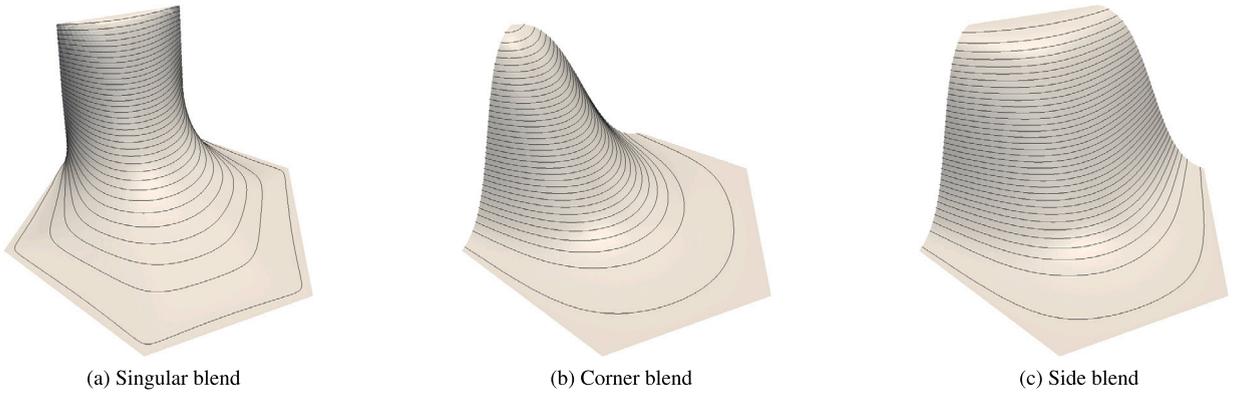
Fig. 15. Types of blending functions over a six-sided domain.

$$W_i^* = \left(\prod_{\substack{k=1 \\ k \neq i}}^{n} d_k^e \right) \Big/ \left(\sum_{j=1}^{n} \prod_{\substack{k=1 \\ k \neq j}}^{n} d_k^e \right). \tag{13}$$

Note that this equivalent expression is still singular at the corner points – fortunately the corner data are available *a priori* from the ribbons.

Blends with two non-zero sides. Corner interpolants $\mathbf{S}_{i-1,i}^{\text{Int}}$ interpolate two consecutive sides. Such an interpolant can be created by combining two adjacent ribbons associated with sides $i-1$ and i. The related corner blending function $W_{i-1,i}$ is supposed to act only on these two sides and vanish everywhere else ($j \neq i-1, i$). The blend has value 1 at the corner then decreases on the adjacent sides of the domain until it reaches 0, see Fig. 15b. The adjacent corner interpolant $\mathbf{S}_{i,i+1}^{\text{Int}}$ interpolates sides i and $i+1$; thus on side i two corner interpolants and two corner blends interact, and the expression $\mathbf{S}_{i-1,i}^{\text{Int}} W_{i-1,i} + \mathbf{S}_{i,i+1}^{\text{Int}} W_{i,i+1}$ will reproduce the i-th ribbon, see also Fig. 25b on page 18. A solution using distance parameters is the following:

$$W_{i-1,i} = \frac{1/(d_{i-1}d_i)^e}{\sum_{j=1}^{n} 1/(d_{j-1}d_j)^e} = \left(\prod_{\substack{k=1 \\ k \notin \{i-1,i\}}}^{n} d_k^e \right) \Big/ \left(\sum_{j=1}^{n} \prod_{\substack{k=1 \\ k \notin \{j-1,j\}}}^{n} d_k^e \right). \tag{14}$$

Note that for regular polygonal domains and Euclidean distances, this blend is equivalent to Wachspress barycentric coordinates (Wachspress, 1975) raised to the e-th power and normalized. These blends have no corner singularities. There are several alternative ways for creating blend functions with the required properties. An interesting solution was proposed in Salvi and Várady (2016), where instead of the n-tuples of distance parameters, only two pairs of parameters are used. The construction assumes full coverage parameterization (see also Section 4.2):

$$W_{i-1,i}^* = \frac{d_{i-1} H(s_i) H(d_i) + d_i H(1-s_{i-1}) H(d_{i-1})}{d_{i-1} + d_i}, \tag{15}$$

where $H(t)$ is a Hermite blend function, e.g. in the G^1 case the cubic polynomial $H(t) = 2t^3 - 3t^2 + 1$.

Blends with three non-zero sides. Here we combine three-sided interpolants composed of three ribbons associated with the sides $i-1$, i and $i+1$. The blend function

$$W_i = W_{i-1,i} + W_{i,i+1} \tag{16}$$

is non-zero only on these three sides; on side i it equals 1, on side $i-1$ and $i+1$ it decreases until it reaches 0 at the adjacent corners, see Fig. 15c. It is easy to see that in this case three interpolants will interact along side i, where the expression $\frac{1}{2}(\mathbf{S}_{i-1}^{\text{Int}} W_{i-1} + \mathbf{S}_i^{\text{Int}} W_i + \mathbf{S}_{i+1}^{\text{Int}} W_{i+1})$ reproduces ribbon i, see also Fig. 23b on page 17.

Blends for Boolean sum schemes. As discussed earlier, the generalized Coons patch combines side interpolants multiplied by side blends and correction surfaces multiplied by corner blends. Along a given side three side interpolants and two correction surfaces will interact to restore the original ribbon.

The above constructions are summarized in the schematic images of Fig. 16.

3.5.2. Blends for control points

In order to interpolate along the boundaries, the blending functions of control point based multi-sided patches must satisfy certain conditions, which are similar to those that are automatically guaranteed by the classical Bézier or B-spline tensor product surfaces. Loosely speaking, the *outermost* control points along side i fully determine the i-th boundary of the patch; the associated blending functions – apart from the common corners – vanish on the other sides ($j \neq i$). This holds in a differential sense, as well; as we move inwards, the first two *layers* of control points will determine G^1 continuity with adjacent patches, the first three G^2 continuity and

(a) One non-zero side   (b) Two non-zero sides   (c) Three non-zero sides   (d) Boolean sum (right: blend for the correction patch)
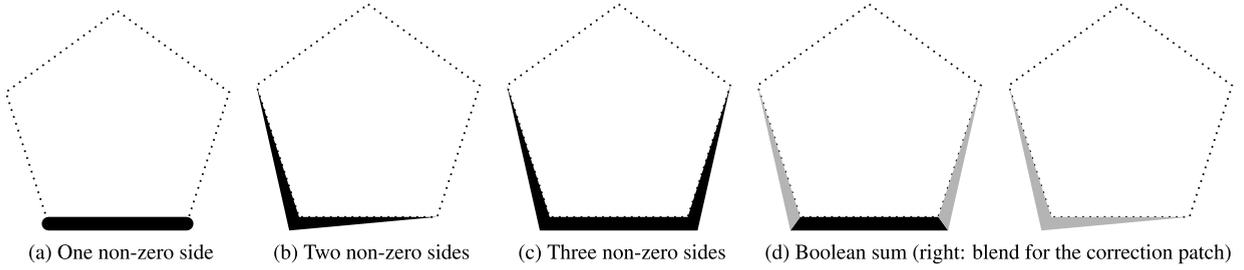
Fig. 16. Schematic depictions of blending functions. Light gray values show blends associated with non-interpolating boundary data.

(a) Side-based construction   (b) Corner-based construction
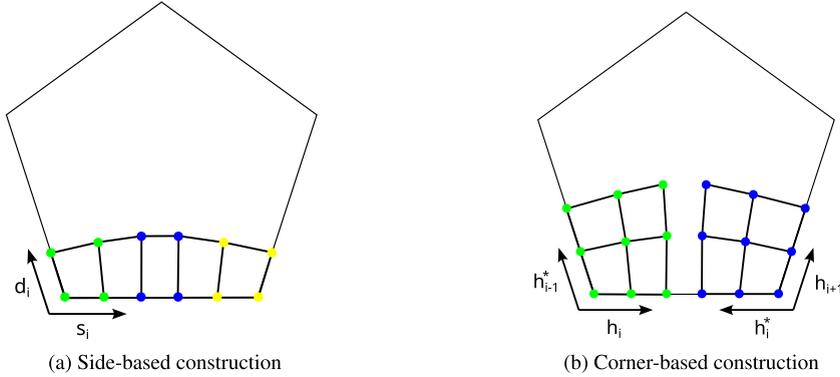
Fig. 17. Association of blending functions and control points.

so on; the related blend functions and the derivatives must also vanish on the other sides. The term 'layer' is also loosely defined, as it may mean different control point configurations in different constructions. As a rule of thumb, control points in the second layer include control points 'adjacent' to the outermost control points, and so on. The blending functions of the *interior* control points are supposed to influence the interior of the multi-sided patch, but must vanish (differentially) along the boundary.

In case of *side* and *corner* based control point structures the blend functions generally include biparametric tensor-product Bernstein or B-spline basis functions. In the next paragraphs we show two examples using Bernstein functions for simplicity.

Fig. 17a shows a G^1 ribbon with two layers, parameterized by s_i and d_i with full coverage over the domain. Assuming the ribbons have degrees p_i and $p_i^\perp > 2$ in the side- and cross-directions, respectively, the blending functions are defined as follows:

$$W_{j,k}^i = \mu_{j,k}^i \cdot B_j^{p_i}(s_i) \cdot B_k^{p_i^\perp}(d_i). \tag{17}$$

The *Bernstein polynomials* $B_j^{p_i}$, $B_k^{p_i^\perp}$ ensure ribbon interpolation on side i where $d_i = 0$, and vanish on the remote sides ($l \neq i-1, i, i+1$) where $d_i = 1$, but the first and last two columns are non-zero on sides $i-1$ and $i+1$. In order to eliminate this effect, the blend functions are multiplied by *localization terms* similar to Gregory twists (see Section 3.1.1):

$$\mu_{j,k}^i = \begin{cases} \alpha_i = d_{i-1}^2 / \left(d_{i-1}^2 + d_i^2 \right), & j = 0, 1, \\ \beta_i = d_{i+1}^2 / \left(d_{i+1}^2 + d_i^2 \right), & j = p_i - 1, p_i. \end{cases} \tag{18}$$

In this way proper interpolation for the i-th ribbon is guaranteed (Várady et al., 2020). G^{e_i} continuity works similarly, when $p_i^\perp > 2e_i$, using rational weights with exponent $e_i + 1$ instead of squares at $j = 0 \dots e_i$ and $j = p_i - e_i \dots p_i$.

In our second example (Fig. 17b) we assume that all degrees are odd and equal ($p_i = p_i^\perp = p$). Then we can associate biparametric blends with corner parameters (h_{i-1}, h_i) and (h_i, h_{i+1}) respectively. The control points are regrouped to be associated with the closest corner, so the respective blending functions are

$$W_{j,k}^{i-1,i} = B_j^p(h_j) \cdot B_k^p(h_k), \qquad j, k = 0 \dots p \div 2. \tag{19}$$

Note that the above blending function contains no localization terms, and local parameters must both (i) be differentially constrained and (ii) have full coverage to achieve ribbon interpolation (Várady, 1991).

For *interconnected* control point structures, the blend functions are generally composed of n parameters. The various possibilities will be discussed at length in Section 5.3.

3.5.3. Handling weight deficiency

In the basic formulae of both interpolant and control point based surfaces, geometric entities are weighted by blend functions. For affine invariance of the patch it is necessary that $\sum W_i = 1$ be satisfied everywhere within the domain. There are schemes where this automatically holds, but in general *weight deficiency* may occur. As a rule, weight deficiency vanishes along the domain boundary and increases towards the interior. Weight deficiency may be viewed as an advantage: we can use it to blend interior interpolants or control points into our patch equation, if we wish to gain extra control over the shape of the surface. For example, taking an arbitrary control point based patch with weight-deficient blend functions, we can introduce a central control point $\mathbf{C}_0$ and express the modified surface as

$$\mathbf{S}(u,v) = \sum_{i \in I} \mathbf{C}_i \cdot W_i + \mathbf{C}_0 \cdot \left(1 - \sum_{i \in I} W_i \right), \tag{20}$$

where $\mathbf{C}_0$ can be freely set (Várady et al., 2016). It is also possible to distribute the weight deficiency among several interior control points (Zheng and Ball, 1997; Várady et al., 2017; Salvi, 2024).

It often happens that we do not want to deal with underdetermined interior shape parameters. The simplest way to ensure that partition of unity holds is then to *normalize* the patch equation, thereby defining a new set of blend functions, e.g. as follows:

$$\mathbf{S}(u,v) = \frac{1}{\sum_{i \in I} W_i} \sum_{i \in I} \mathbf{C}_i \cdot W_i = \sum_{i \in I} \mathbf{C}_i \cdot \hat{W}_i. \tag{21}$$

Note that while normalization does not affect the interpolation properties of the scheme, it alters the shape of the blending functions.

3.6. Editing capabilities

Editing ribbon-based surfaces requires a careful balance between ease of definition and degrees of freedom. On the one hand, with ribbon-based modeling only the boundary geometry needs to be specified, and then the interior of the surface is determined. On the other hand, it might be preferable to have additional degrees of freedom for design, fairing or approximation. Nevertheless, properly setting a large number of geometric quantities can be inconvenient and time-consuming. In light of these various considerations, let us now enumerate a few options to facilitate the editing of ribbon-based multi-sided patches.

3.6.1. Ribbon editing

For all schemes, it is an evident requirement that ribbons should be editable and any change must imply a simultaneous modification for the multi-sided patch. Completely unconstrained editing of the ribbons is rarely allowed, as geometric continuity with adjacent patches must be maintained. A recent paper by Salvi et al. (2023) investigated the problem of constrained editing of ribbon-based patches. By means of *control vectors* it is possible to manually edit cross-derivatives while maintaining tangent continuity with the input. In the same paper, a method for the initial setting of cross-derivatives is proposed that takes into account the shape of the domain as well as the local parameterizations.

3.6.2. Ribbon enrichment

To create the desired shape or satisfy certain constraints with some flexibility it might be necessary to *refine* the ribbon representation, by e.g. Bézier degree elevation or B-spline knot insertion. There are schemes that support individual refinement of the input ribbons along their boundaries, and ideally this results in no change in the surface shape. For schemes using an interconnected control structure this sort of refinement must be done in a global manner, i.e., every ribbon must be refined simultaneously and in both directions (Ball and Zheng, 2001; Li et al., 2021; Wang et al., 2023).

When ribbons are defined by grids of control points, they can also be enriched by adding extra layers. While this approach brings in new degrees of freedom, setting their default values can be a challenge. One example of this approach was suggested in Várady et al. (2020), where *ribbon control points* were added for better shape control, see Fig. 18.

3.6.3. Interior control

Using some of the interpolant-based schemes it is possible to add new *auxiliary* interpolants or even new interior structures. An example is shown in Fig. 19a, where a ribbon tangentially connecting two boundary points was incorporated to create a specific shape feature in the interior. Other constructions blend an entire surface with a ribbon-based patch to modify the interior – see Fig. 19b. This auxiliary surface can be a predefined feature (Várady et al., 2012; Martin and Reif, 2022), or may be modeled interactively using *displacement vectors* (Kosinka et al., 2015; Salvi, 2021). In a similar vein, Salvi (2024) defined *hybrid patches*, where interpolant-based surfaces are enriched by a tweakable set of interior control points arranged into an interconnected structure; see Fig. 20 and the discussion in Section 3.5.3. Determining default values for such interior degrees of freedom is a complex issue that relates to the topics of fairing and optimization – see our discussion in Section 7.6. One possibility is to use a refinement procedure to set the control points in a coarse-to-fine manner, as done in the case of convex Generalized Bézier patches (Várady et al., 2016). Another possibility is to optimize the control point positions according to some fairness energy while keeping the boundaries fixed, as done by e.g. Salvi (2019) for S-patches.

(a) Original patch	(b) Patch modified with ribbon control points
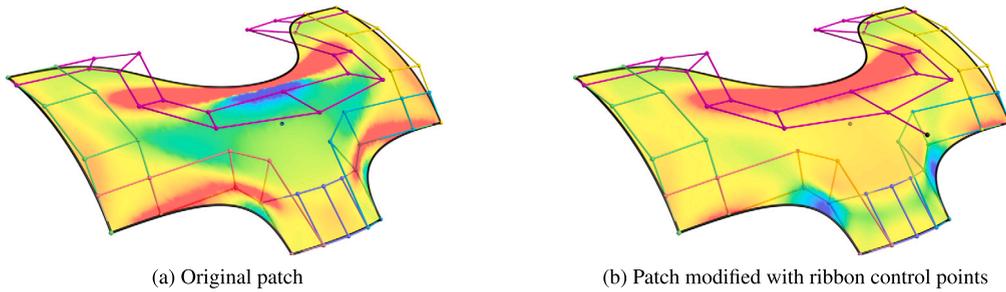
Fig. 18. Modifying the interior by adding extra ribbon control points (Várady et al., 2020).

(a) Snapping to an auxiliary curve	(b) Modifying the interior with an auxiliary surface
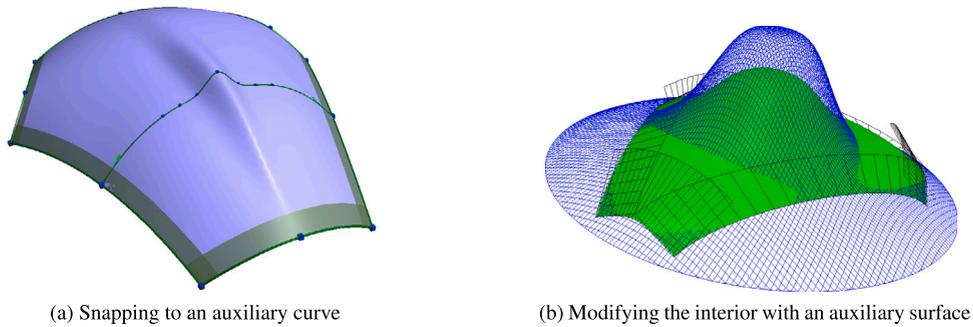
Fig. 19. Extensions to a singular-blend based patch (Várady et al., 2012).

(a) Ribbons and control points for tweaking the interior	(b) The control structure of a Generalized Bézier patch
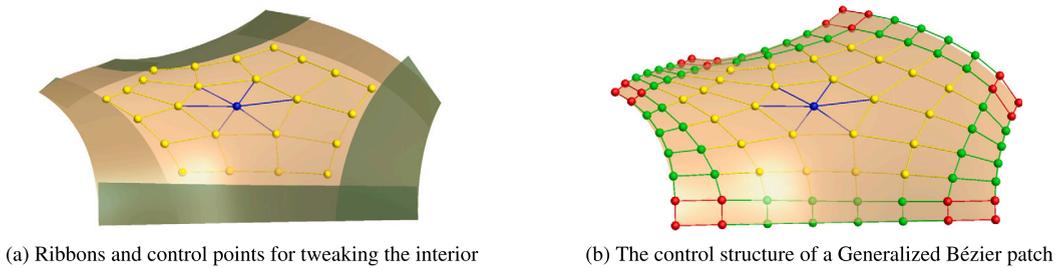
Fig. 20. The hybrid patch of Salvi (2024), based on the blending functions of Generalized Bézier patches.

4. Interpolant-based schemes

Initial efforts on multi-sided patches concentrated on hole filling and transfinite interpolation, and the major approach to this was to use a weighted sum of interpolant surfaces. This proved to be a fruitful line of research, which has produced many different schemes. There are many aspects by which we can classify these; here we are going to follow the division of Section 3.2.1.

4.1. Side-based interpolants

There is a simple and natural idea that seems to have been reinvented multiple times, which is to use the well-known inverse distance weights of Shepard (1968), usually associated with scattered data interpolation, but with the weights defined as the normalized inverse distance *in domain space*. In other words, given a point in the 2D domain, distances are computed to each side, which, in turn, are used to obtain the weights of the associated interpolants by taking some negative power (depending on the order of interpolation), see Eq. (12) and Fig. 15a in Section 3.5.1. In these methods, the interpolants are often simply defined to be the ribbon surfaces, and they are composed with a bivariate local parameterization map (see Section 3.4). One of the parameters is usually (but not necessarily) the same as the distance measure used for blending.

A common drawback of these surfaces is due to the choice of blending function: the Shepard-like weights decrease rapidly as we move away from the boundary, resulting in large curvature variations near the sides. Furthermore, the blends are singular in the corners. This singularity is removable, either by using twist-compatible input, or by applying Gregory twists, see Section 3.1.1. Another singular blending function with better characteristics, based on Generalized Bézier patches (see Section 5.1.1), has been proposed recently by Salvi (2024), where interior control points have also been added to facilitate shape editing.

(a) Distance is $d_i = \overline{\mathbf{p}\mathbf{p}_{i-1}} + \overline{\mathbf{p}\mathbf{p}_i} - \overline{\mathbf{p}_{i-1}\mathbf{p}_i}$; side parameters are $s_i = d_{i-1}/(d_{i-1} + d_{i+1})$

(b) (s, d) mapping of a side in a concave domain (cf. Fig. 13a)
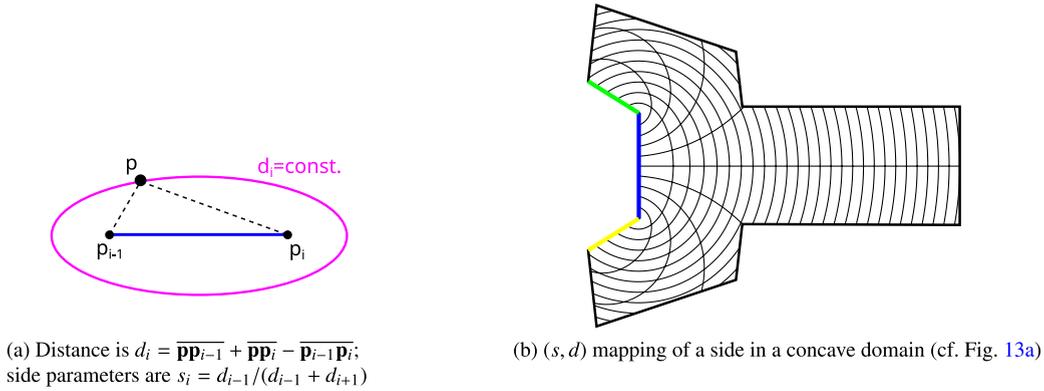
Fig. 21. Parameterization of concave and multiply connected patches in Kato (1991, 2000).
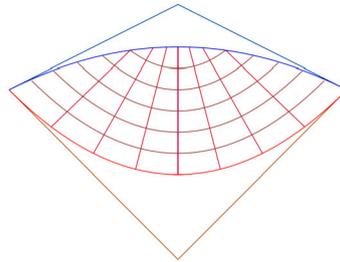
Fig. 22. Two-sided domains in Várady et al. (2012), parameterized by sweep lines and parabolic arcs.

4.1.1. Classical methods

An early incarnation of this procedure, in a triangular, scalar-valued setting, is found in Barnhill (1977), where it is referred to as the Brown–Little triangle. Barycentric coordinates are used as distances, and the input is treated as a single function over the right angle triangle $(0,0) - (1,0) - (0,1)$ instead of individual ribbons. This has been modified by Nielson (1979) to use radial parameterization and linear, quadratic or cubic side interpolants generated by Hermite blends of the vertex and its footpoint on the opposite side (hence the name *side–vertex* method). An application to the vector-valued case is described in Nielson (1987), along with an influential discretized version interpolating only the vertices and their normals. Interpolants for the G^2 case were developed by Hagen and Pottmann (1989), and the method was generalized to any number of sides in Gregory (1986).

Kuriyama (1994) uses Wachspress coordinates on a regular polygonal domain to generate side parameters that give 0 at one half of the domain vertices, and 1 at the other half (cf. the sweeping methods in Section 3.4.2, where only the adjacent sides are set to 0 and 1); distance parameters are computed as in Eq. (9). The interpolants are created from the boundary curves via energy optimization, and the parameter used in the inverse distance weights is $1/\lambda_{i-1}\lambda_i$. Several special cases are discussed, including interpolant creation in the presence of T-nodes, corners with 3 or more curves, patches missing one boundary, and 2-sided configurations (handled by splitting the boundaries in half).

Gao and Rockwood (2005) arrive at practically the same surface as Gregory (1986), but by a completely different route, coming from a weighted least squares fit of the interpolants. The distance parameter is computed as the distance on the radial sweepline.

4.1.2. Extensions to irregular, concave and multiply connected domains

A similar construction – although originally using a more complex blending function – was also described in Kato (1991, 2000), extending the scheme to handle multiple loops (holes, represented as polygons in the domain interior) and concave boundaries, respectively. A polygonal domain of the same topology as the defining edges is created, trying to preserve the scale and angles of the 3D configuration by projecting the corner points on a best-fit plane. Perpendicular distances are replaced (when needed) by a measure combining the distances from the endpoints of the base segment, thereby solving the problem of concave corners, see Fig. 21. Side parameters are generated by the ratio of the adjacent distance parameters.

In a work that serves as a basis for recent developments, Várady et al. (2011) demonstrate the need for irregular domains and propose an algorithm to generate a convex polygonal domain based on the boundary curves (see Fig. 9a in Sec. 3.3). Different parameterization schemes (central line sweep, biquadratic) are described, as well. One- and two-sided patches are also shown (see Fig. 22), but these are explained in detail only in Várady et al. (2012), in which the question of interior control is investigated. The representation is extended to include auxiliary objects with ribbons in the interior, e.g. single points, curves, or even surfaces, see Fig. 19. A related technique is introduced in Martin and Reif (2022), to be discussed in Section 6.1. A general framework for inverse distance weighting over fairly general domains was developed by Rvachev et al. (2001), based on the mathematical theory of R-functions (Shapiro, 2007).

(a) Construction of a composite side interpolant (cf. Fig. 25a).

(b) Schematic depiction of the blends associated with three sides
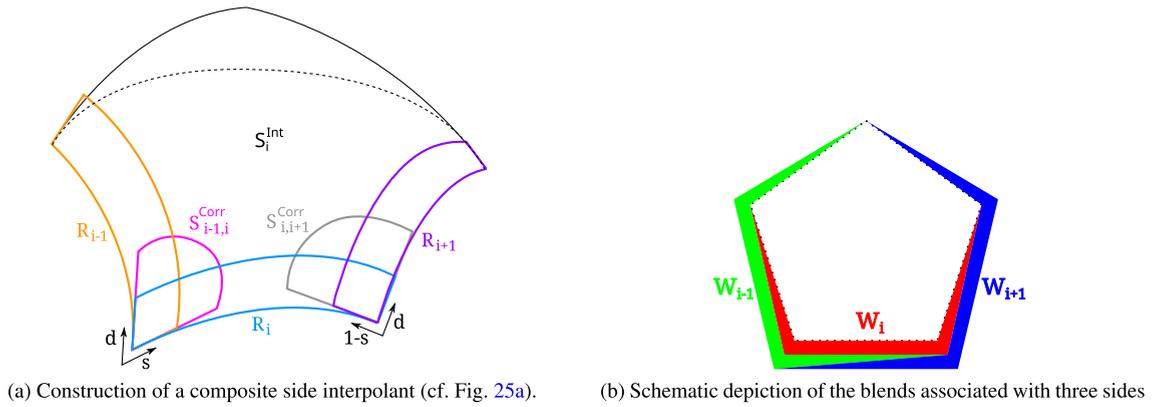
Fig. 23. Multi-sided surfaces using interpolants based on three consecutive sides (Salvi et al., 2014). Since each of interpolants $i-1$, i, $i+1$ reproduces the i-th ribbon, and the blending functions associated with all other ribbons vanish, the patch can be written as $\mathbf{S} = \frac{1}{2} \sum_i \mathbf{S}_i^{Int} W_i$.

(a) Constant parameter lines in polar (r, θ); the angular coordinate is adjusted such that $[-\pi/n, \pi/n]$ maps to $[-\pi/4, \pi/4]$

(b) Creating a quadratic curve segment by its Bézier polygon – the blue segment gradually vanishes as the radial coordinate approaches 1
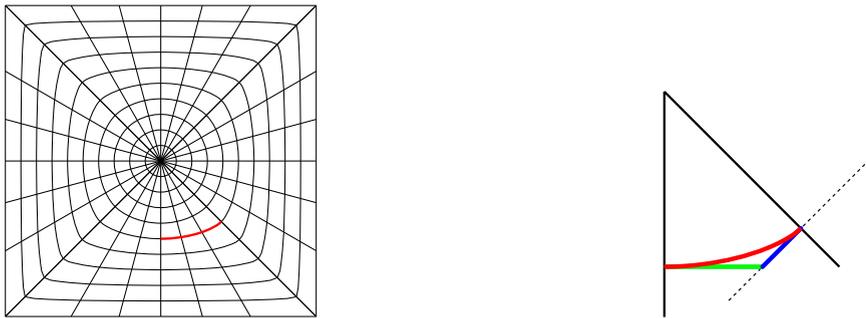
Fig. 24. Polar reparameterization of ribbons in Shi et al., 2010. The concentric constant parameter lines are defined by quadratic or cubic curve segments, connecting with G^1 or G^2 continuity, respectively.

4.1.3. Ribbons interpolating three sides

Using interpolants with (nearly) identical geometry in the vicinity of the common corner is highly beneficial, as demonstrated by the SuperD system (Rockwood et al., 2016) and the superior quality of the schemes discussed in Section 4.2. The culmination of this idea is when interpolants reproduce three consecutive boundaries. This is to be preferred especially when the number of sides is relatively low, since in this way the interior of the patch is geometrically well-defined. Salvi (2020) shows a construction with positional (C^0) interpolation, where the blending function is a very simple function of the distance parameter:

$$S = \frac{1}{2} \sum_{i=1}^{n} \mathbf{S}_i^{Int} \cdot (1 - d_i). \tag{22}$$

A similar surface with G^1 interpolation properties, called *Composite Ribbon patch*, was proposed by Várady et al. (2013); Salvi et al. (2014). These methods create the interpolants as partial Coons patches, and apply a blending function that vanishes on all but 3 sides, see Fig. 15c in Section 3.5.1. The logic of the construction is also demonstrated in Fig. 23.

4.1.4. Patches over a circular domain

Shi et al. (2010) combine NURBS ribbons using polar reparameterization, see Fig. 24. The individual ribbons are associated with the $[-\pi/n, \pi/n]$ arc of the unit circle, and then rotated by $2\pi k/n$ ($k = 1 \ldots n$) to create a common domain. The blending function is based on the function proposed by Hartmann (2001b); the representation can handle G^k interpolation. An approximation with n NURBS patches is also described. Using a circular domain for a boundary loop with sharp corners introduces large distortions that may harm surface quality; Salvi (2022) applies a trigonometric mapping to create overlap patches (see Section 5.2.1) with smooth corners. In a similar setting, Shi et al. (2013) creates a periodic NURBS surface, which can also be regarded as a one-sided patch. The handling of derivatives at such rounded corners is an intricate problem; some theoretical results have been published recently by Marussig and Reif (2022).

(a) $\mathbf{S}_{i-1,i}^{\text{Int}} = \mathbf{R}_{i-1} \oplus \mathbf{R}_i = \mathbf{R}_{i-1} + \mathbf{R}_i - \mathbf{S}_{i-1,i}^{\text{Corr}}$, where $\mathbf{S}_{i-1,i}^{\text{Corr}}$ is a corner correction patch

(b) Blends associated with two corners, reproducing the common side
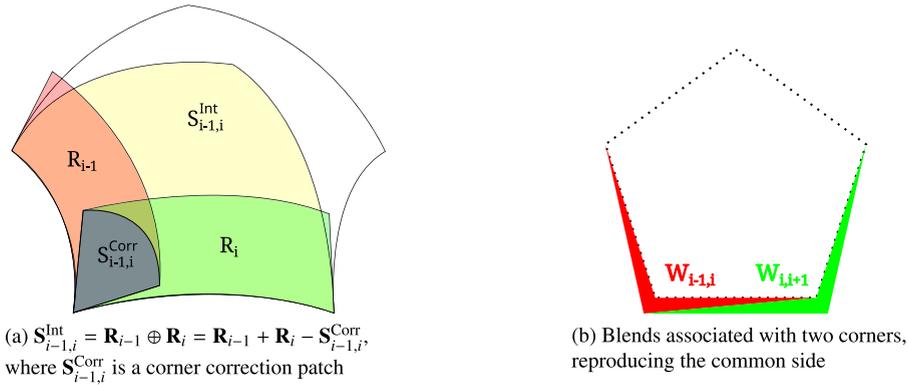
Fig. 25. Creation and blending of corner interpolants.

4.2. Corner-based interpolants

The patch representations in this section are defined as the weighted sum of *corner interpolants*, i.e., surfaces interpolating two consecutive ribbons. The first such (vector-valued) interpolant was published by Gregory and Charrot (1980), which used barycentric coordinates and a *polynomial* blending function to interpolate C^1 data at the boundaries of a triangle. The interpolants are created as the Boolean sum of two ribbons (see Fig. 25a), and parameterized by the barycentric coordinates associated with the other two corners of the domain. Farouki et al. (2010) used these patches to create triangular surfaces interpolating *geodesic* boundaries.

The above patch was extended to the 5-sided case by Charrot and Gregory (1984), replacing barycentric coordinates by radial sweep or perpendicular distance. The blending function is always based on the latter, and has the form of Eq. (14); see also Fig. 15b. A further generalization to an arbitrary number of sides was proposed in Gregory (1986), which was later generalized to G^2 interpolation by Gregory and Hahn (1987a,b, 1989). In a very influential paper, Plowman and Charrot (1996) introduced Gregory twists into the formulation (see Section 3.1.1), and proposed shape controls changing the fullness of the patch, tailored especially for creating vertex blends. Hall and Mullineux (1997) added similar controls, which they set automatically by minimizing a smoothness energy. Beyond vertex blending and hole filling, these patches have also found applications in finite element grid generation (Wang and Tang, 2004).

Hall and Mullineux (1999a) investigated the necessary conditions for G^2 continuity between such patches, and found that when there are adjacent surfaces with different number of sides, there are additional differential constraints on parameterization (similarly to those in Várady (1991), see Section 5.2.1). They recommended a suitable parameterization for $n > 3$, and showed that such a construction is not possible for triangular domains. A similar method is used by Salvi et al. (2013); Salvi and Várady (2014), where the generation of curved interpolants from an input curve network is also discussed.

Midpoint patches proposed in Salvi and Várady (2016) use the Hermite-like blending function shown in Eq. (15). This method results in high-quality surfaces in practice, and it also provides an additional degree of freedom to edit the patch interior, exploiting the weight deficiency.

4.3. Boolean sum patches

The original transfinite interpolation surface, the four-sided Coons patch (Coons, 1967), was defined as the Boolean sum of surfaces connecting opposing sides (see Section 3.2.1). Barnhill et al. (1973) applied the same idea to triangular patches, where each side has two 'opposite' sides, and the Boolean sum of any two such connecting surfaces satisfies the interpolation property. For reasons of symmetry, these are then further blended to create the final patch. The twist constraints of C^1 and higher order interpolation is developed in Barnhill and Gregory (1975).

The intuitive logic of the Coons patch is very appealing, but it does not lend itself easily to the extension to multiple sides. Although the patch of Charrot and Gregory (1984); Gregory (1986) – described in the previous section – is considered to be such by Sabin (1996), a more direct generalization (called the *Generalized Coons patch*) was published by Várady et al. (2011); Salvi et al. (2014), reinterpreting the connecting surfaces as side interpolants multiplied by Hermite blending functions, and the correction patch as the blended sum of derivatives at the vertices, see also Fig. 5.

The domain of the patch is a convex polygon, and the interpolants are parameterized by a line sweep and a distance parameter. A new constraint is posited on the derivatives of these mappings, for which various *differentially constrained parameterizations* are explored, see Fig. 14c and the end of Section 3.4.

The Generalized Coons patch is visually very similar to that of Charrot and Gregory (1984); and it is shown in Salvi and Várady (2015) that aside from the parameterizations of the side interpolants, their patch equations are identical. Salvi et al. (2017) combined Generalized Coons patches with Midpoint patches (discussed above) to add interior control.
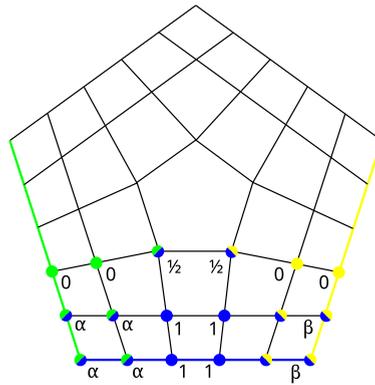
Fig. 26. Localization terms of Generalized Bézier patches. The notation α, β was defined previously in Eq. (18).

5. Control point based schemes

Free-form surfaces are traditionally defined by a collection of control points weighted by a suitable set of blending functions (see Section 3.5.2). Historically speaking, the first attempts at creating control point based multi-sided patches were based on interconnected control structures that allowed for some kind of generalization of Bernstein polynomials (Sabin, 1983; Loop and DeRose, 1989; Krasauskas, 2002; Karčiauskas, 2003). In a later stage, side- and corner-based control structures were investigated (Várady, 1991; Várady et al., 2016) and were found to be more flexible, enabling multi-sided generalizations of B-splines, as well (Pla-Garcia et al., 2006; Hettinga and Kosinka, 2020a; Vaitkus et al., 2021). In this chapter we assume that boundary ribbons are piecewise polynomial functions represented in Bernstein–Bézier or B-spline form, and consider control point based patches capable of interpolating this boundary data. The control points of the patch might coincide with those of the input ribbons, or might be derived from them by some procedure.

We mention a fundamental theoretical result by Warren (1994) stating that a rational parametric surface bounded by an arbitrary number of independent polynomial curves must possess base points (where the numerator and the denominator vanish simultaneously). As a consequence, multi-sided patches with sufficient flexibility are necessarily parameterized by rational polynomials with nontrivial denominator.

5.1. Side-based control structures

This family of multi-sided surfaces takes inspiration from the blending-based methods surveyed in Section 4, but instead of surface interpolants, it is Bézier or B-spline *basis functions* associated with each side that are blended together. An important precursor of these constructions were quadrilateral Gregory patches (Chiyokura and Kimura, 1983).

5.1.1. Generalized Bézier patches
Generalized Bézier (GB) patches were introduced by Várady et al. (2016). These were originally defined as multi-sided variants of tensor-product Bézier patches over regular and convex polygonal domains, but were later further generalized in various ways.

The control points of a GB patch are organized in the form of a rectangular spiderweb (see Fig. 27a), interpreted as *n* overlapping side-based interpolants. In the original construction of Várady et al. (2016), ribbons (with potentially different degrees) are first degree-reduced to cubic, and – after fixing a central control point – the resulting *base patch* is degree-elevated to the highest degree represented among the input ribbons. In this way ribbons can be reproduced exactly after applying suitable *displacements* to the degree-elevated control points, while interior control points are placed at reasonable default positions.

Patches in this family employ side-based biparametric mappings (s_i, d_i) to locally parameterize the multi-sided domain onto the unit square domain of each ribbon. These parameters are then used to pull back suitable tensor-product basis functions associated with the ribbons onto the domain. As discussed in Section 3.5.2, assuming that the local parameterizations have full coverage, only the basis functions of adjacent sides will interfere with each other, allowing ribbon interpolation with a relatively small degree of blending.

The definition of GB patches is based on tensor-product Bernstein functions, multiplied by Gregory-like localization terms, following Eq. (17) of Section 3.5.2. The localization terms at the corners are set according to Eq. (18) to ensure ribbon interpolation, while the remaining blend functions can also be weighted in various ways, as long as the weights associated with each control point sum to 1; one possibility, as recommended by Várady et al. (2017), is shown in Fig. 26.

Note that the modified basis functions do not satisfy partition of unity, and the weight deficiency can be assigned to the central control point (Várady et al., 2016), distributed among the interior control points, or normalized away (Várady et al., 2017) – see Section 3.5.3.

An important follow-up work by Várady et al. (2017) described enhancements of the original GB scheme related to the treatment of the weight deficiency and the central control point, and also introduced certain improvements to the local parameterization. Hettinga and Kosinka (2018) have demonstrated that GB patches can also handle ribbons with incompatible corner twists.

While GB patches can be degree-elevated in an analogous fashion to rectangular Bézier surfaces, the method described in Várady et al. (2016) had two serious issues: (i) the weight deficiency changed drastically between even and odd degrees, (ii) degree elevation did not preserve the surface shape due to the presence of correction terms. Problem (i) was successfully resolved in Várady et al. (2017) by slightly tweaking the usage of the central control point. To solve Problem (ii), Wang et al. (2023) have recently developed a method for exact degree elevation of GB patches that applies the usual rules to the products of control points and correction terms.

Convex GB patches have been applied to surface fitting (Salvi et al., 2018), polyhedral design (Szörfi and Várady, 2022), color interpolation (Hettinga et al., 2019) and isogeometric analysis (Wang et al., 2023).

The construction of the original GB surface scheme relied heavily on having a convex polygonal domain for both the GBC-based local parameters and the existence of a spiderweb control structure. Salvi and Várady (2018) generalized the method to polygonal domains with concave corners, using local parameters based on harmonic GBCs. Over non-convex domains, creating an interconnected control structure remains an open problem, so the ribbons were treated independently, allowing the degree and the number of layers to be different for each side. Várady et al. (2020) extended the GB construction to patches defined over curved, multiply connected domains using harmonic GBCs.

5.1.2. Generalized B-spline patches

A common method for creating multi-sided surfaces with B-spline boundaries is by piecing together 4- or 3-sided patches with geometric continuity – this approach comes with many challenges, as discussed in the Introduction. An early attempt at a genuinely multi-sided generalization of B-spline surfaces was by Pla-Garcia et al. (2006), where a special curve-based parameterization was used to pull back B-spline basis functions onto a multi-sided domain representing an extraordinary region of a polyhedral surface. This approach was limited to B-splines with uniform non-clamped knot vectors, and relatively simple configurations.

The basic definition of concave/curved domain GB patches as described in the previous chapter does not depend on the ribbons being represented in a specific basis, which has motivated Vaitkus et al. (2021) to define *Generalized B-spline (GBS)* patches over curved and multiply connected domains. GBS patches are constructed from ribbons that are still Bézier in the cross-direction, but are B-splines (with arbitrary degree and knot vector) in the longitudinal direction, which can be interpolated with arbitrary G^e continuity (for ribbons with $e + 1$ layers). The periodic parameterization of interior hole loops is achieved by solving an elliptic boundary value problem – see Section 3.4.2.

Hettinga and Kosinka (2020b) have generalized the quadrilateral spiderweb control structure of convex GB patches to interpolate uniform 2-segment B-spline ribbons, with the goal of filling holes around extraordinary vertices in subdivision surfaces. An important improvement upon the original scheme was the ability to ensure G^2 continuity with neighboring bicubic surfaces using special multi-degree spline functions in the cross-direction. This was generalized even further by Hettinga and Kosinka (2020a) defining surface patches that connect with G^e to degree-$(e + 1)$ uniform B-spline surfaces.

5.2. Corner-based control structures

We discuss two approaches to corner-based patches below.

5.2.1. Overlap patches

Overlap patches, introduced by Várady (1991), are defined over a polygonal domain by simple superposition of interpolants constructed from data at the corner points: position, two derivatives, and the twist vector. The data can be regarded as 4 control points at the corner of a cubic Bézier patch multiplied by their associated Bernstein polynomials. In other words, Overlap patches are in essence corner-based generalizations of cubic Bézier patches, see Fig. 17b and Section 3.5.2. A straightforward extension to higher-degree corners was described by Salvi (2022). To ensure G^k continuity, constrained parameters are required (see Section 3.4), for which Várady (1991) proposed two alternatives: one based on linear segments and conics (with G^1 connection), and another defined by the inverse of a planar biquadratic surface. Overlap patches have been applied to polyhedral design (Wang et al., 1999).

5.2.2. Blending Bézier patches

In a recent work Qin et al. (2023) introduced *Blending Bézier patches* defined over a rectangular spiderweb control structure (see Fig. 27a). Corner interpolants are first constructed as ordinary tensor-product Bézier patches, by simple averaging of the control points related to each corner. The corner interpolants are then blended together using GBCs raised to the appropriate power, making this a control point based variant of the Charrot–Gregory corner scheme (see Sec. 4.2). Connection to boundary ribbons with G^2 continuity requires constrained parameters (see Section 3.4), which was ensured using the method of Salvi and Várady (2014). Blending Bézier patches were applied by Qin et al. (2023) to hole filling and polyhedral design. In a less known work, Karčiauskas (2000) describes a different generalization of the Charrot–Gregory construction, related to the methods described below in Section 5.3.

5.3. Interconnected control structures

In this section, we survey various multi-sided patches based on interconnected control structures – see Fig. 27. These are general-izations of the classical Bernstein–Bézier representations, that can interpolate an arbitrary number of boundary ribbons with G^1 or G^2 continuity. The patches are (with a few exceptions) parameterized by rational functions over polygonal domains, and their boundary curves, as well as cross-derivatives, have a (possibly rational) Bézier representation. Note that some patches in previous sections were also defined over an interconnected (rectangular spiderweb) control structure, but were interpreted in a side- or corner-based manner.

(a) Rectangular spiderweb   (b) Triangular spiderweb   (c) Minkowski sum   (d) Lattice polygon
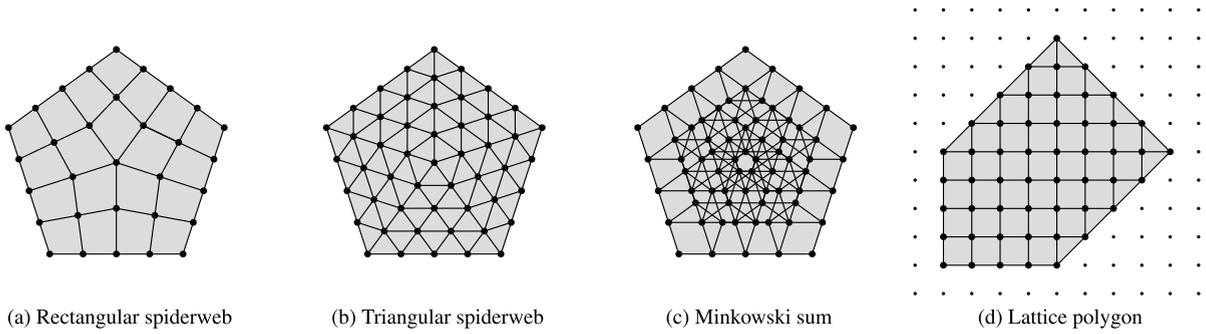
Fig. 27. Examples of interconnected control structures with degree-4 Bézier boundaries. Terminology adopted from Goldman (2004).

5.3.1. Sabin–Hosaka–Kimura–Zheng–Ball patches

The first attempt at a genuinely multi-sided surface scheme goes back to the work of Sabin (1983) and Hosaka and Kimura (1984), which has been generalized to an arbitrary number of sides by Zheng and Ball (1997). The control points of these surfaces form a rectangular spiderweb structure as shown in Fig. 27a and can be indexed by special n-tuples $\boldsymbol{i} = (i_1, \dots, i_n) \in I_{n,d}$, where d denotes the degree of the boundary curves. The surface is not defined explicitly in terms of the coordinates of a 2D domain, but via n local coordinates $\mathbf{s} = (s_1, \dots, s_n)$ that satisfy a set of non-linear differential equations. For details on the indexing scheme and the constraint equations refer to Zheng and Ball (1997). The surface is defined by a combination (Eq. (7)) of control points $\mathbf{C_i}$, weighted by blending functions similar to tensor-product Bernstein polynomials[1]:

$$
B_{\mathbf{i}}^d(\mathbf{s}) = \begin{cases} \binom{d}{i_{k-1}} s_{k-1}^{i_{k-1}} s_{k+1}^{i_{k+1}} \prod_{j \notin \{k-1, k, k+1\}} s_j^d \left(1 - d\, c_n \prod_{j=1}^n s_j\right), & \exists k. i_k = 0, \\ \binom{d}{i_{\ell-1}} \binom{d}{i_\ell} \prod_{j=1}^n s_j^{i_j}, & \text{otherwise,} \end{cases}
$$

(23)

where $i_{\ell-1}$ and i_ℓ are the two smallest elements in the index tuple, and c_n is a constant depending only on the number of sides. Note that these blending functions do not satisfy partition of unity, so the weight deficiency must be distributed among the interior control points (as proposed by Zheng and Ball, 1997), or normalized away. Sabin originally considered only 3- and 5-sided generalizations of quadratic Bézier patches. The construction was generalized to the cubic case, and also to 6-sided patches by Hosaka and Kimura (1984). Already for 6 sides the local parameters can only be expressed in terms of non-rational functions (Sabin, 1986, 1994) – note, however, that a finite portion of the non-rational surface might still possess a rational parameterization, as shown by Karčiauskas and Krasauskas (2000); Karčiauskas (2003). Building on the general construction of Zheng and Ball (1997), further generalizations allowing twist incompatibility were developed by Hall and Mullineux (1999b); Zheng (2001). A method for exact degree elevation was also presented by Ball and Zheng (2001). The evaluation of these patches can be non-trivial due to the implicitly defined coordinates – an evaluation method over a domain triangulation has been recently presented by Salvi and Gál (2023). While these patches have the advantage of possessing simple Bézier-like control structures, it should be noted that to the authors' knowledge no general formula for the coordinates has been discovered for domains with $n \geq 7$ sides. The parameter constraints could potentially be solved using numerical methods, but we are not aware of any (documented) attempt in the literature. These patches have found practical applications in polyhedral design (Zheng et al., 2005) and texture mapping (Zheng and Zhang, 2002).

5.3.2. S-patches

A direct multi-sided generalization of Bézier triangles called *S-patches* was proposed by Loop and DeRose (1989). The definition of an n-sided S-patch relies on generalized barycentric coordinates $\lambda = (\lambda_1, \dots, \lambda_n)$ over an n-sided polygonal domain. S-patch control points (forming a Minkowski sum structure, see Fig. 27c and Goldman 2004) are labeled by the set $L_{n,d} = \{\mathbf{s} = (s_1, \dots, s_n) \mid \sum_{i=1}^n s_i = d\}$ of n-tuples of non-negative integers, whose sum is the depth of the surface (d). The surface point corresponding to a domain point is given by a weighted combination (Eq. (7)) of control points with *multinomial Bernstein functions* of the barycentric coordinates:

$$
B_{\mathbf{s}}^d(\lambda(u,v)) = \binom{d}{\mathbf{s}} \cdot \prod_{i=1}^n \lambda_i^{s_i}.
$$

(24)

An equivalent geometric interpretation (illustrated on Fig. 28) is that via the GBCs the n-sided domain is embedded into an $(n-1)$-dimensional Bézier simplex (the higher-dimensional analogue of the Bézier triangle) and the degree of the Bernstein basis over the simplex determines the depth of the S-patch.

The original construction of Loop and DeRose (1989) employs a variant of Wachspress coordinates (Floater, 2015) suitable for regular polygons. Later Langer et al. (2008), and then Schaefer in Hormann and Sukumar (2017, Chapter 8) investigated S-patches using alternative GBCs, such as mean value coordinates (Floater, 2003), that are also defined for non-convex and multiply

---

[1] In case of $n = 3$ slightly different blending functions are required.
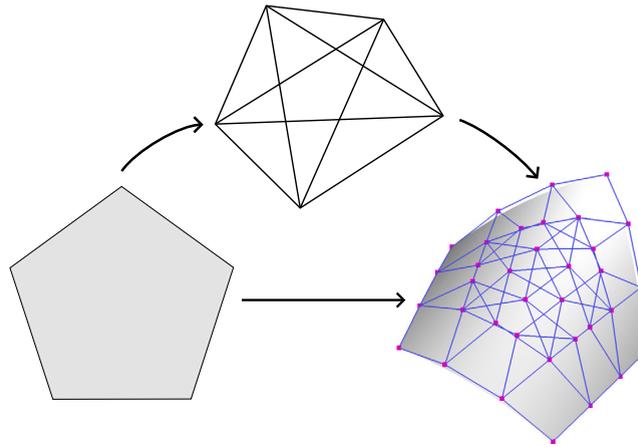
Fig. 28. Definition of an S-patch, adapted from Loop and DeRose (1989).

connected polygonal domains (Hormann and Floater, 2006). An S-patch is bounded by Bézier curves of degree d, however Smith and Schaefer (2015) have developed a method for selective degree elevation of each side (at the cost of losing positivity of the blending functions). Twist-incompatible S-patch variants using Gregory twists were introduced first by Lodha (1993), and developed further by Hettinga and Kosinka (2018). The S-patch inherits many properties of Bézier patches, including blossoming, exact degree (depth) elevation and recursive evaluation (De Casteljau's algorithm). The boundary cross-derivatives are determined by the control points near the boundary, and Loop and DeRose (1990) have shown how G^1 boundary interpolation can be achieved. S-patches have a large number of control points, which can be a challenge for manual design. Salvi (2019) has adapted the Discrete Coons method of Farin and Hansford (1999) to automatically position the interior control points. S-patches have found practical applications in polyhedral design (Loop and DeRose, 1990), hole filling (Lodha, 1993; Salvi, 2019), vertex blending (Zhou and Qian, 2010), cage-based editing (Smith and Schaefer, 2015), numerical analysis (Floater and Lai, 2016), and exact representation of spheres (Grošelj and Praprotnik, 2022).

5.3.3. Toric Bézier patches

A peculiar multi-sided generalization of the Bernstein–Bézier representation, the toric Bézier patch, was introduced by Krasauskas (2002). This line of research can be traced back to Warren (1992), who created multi-sided patches by setting the weights of a rational Bézier triangle to zero, effectively '*blowing up*' patch corners into rational boundary curves, which was later extended to tensor product patches, as well (Fiorot and Gibaru, 2002; Gibaru, 2004). Warren shows that quadrilateral Gregory patches (see Section 3.1.1) are also examples of this blow-up construction. For Warren's patches, the control points $\mathbf{C}_{(k,l)}$ correspond to the integer lattice points $(k,l) \in D \cap \mathbb{Z}^2$ contained within a convex polygonal domain $D \subset \mathbb{R}^2$ – see Fig. 27d. The closely related *toric Bézier patches* are defined over the same kind of domain in terms of perpendicular distances $h_i(u,v) = a_i u + b_i v + c_i$ from the domain sides (scaled to ensure integer values at lattice points), with control points weighted (Eq. (7)) by toric blending functions

$$B_{(k,l)}(u,v) = c_{(k,l)} \cdot \prod_{i=1}^{n} h_i(u,v)^{h_i(k,l)}. \tag{25}$$

Note that such blending functions might not define a partition of unity, so the patch equation is normalized analogously to rational Bézier patches, with positive weights $w_{(k,l)}$. The coefficients $c_{(k,l)}$ can be freely chosen (with the restriction that the boundaries should be rational Bézier curves defined by the boundary control points). Tensor-product and triangular Bézier patches are (reparameterized) toric patches over axis-aligned rectangles and triangles. For a given lattice polygon, Warren's blow-up construction and a toric Bézier patch are in fact different parameterizations of the same surface (Krasauskas, 2002). The order-k cross-derivatives are determined by the first k layers of control points from the boundary and the conditions for G^1 and G^2 continuity have been worked out by Sun and Zhu (2015, 2018).

The original works on toric Bézier patches (Zubė, 2000; Krasauskas, 2001, 2002) were inspired by the algebraic geometry of *toric varieties* – see e.g. Sottile (2003) for some relevant mathematical background. Further work has been done on evaluating and degree elevating toric patches (Li et al., 2021), on the linear precision property (Graf von Bothmer et al., 2010), as well as relating their control polyhedra (García-Puente et al., 2011) to the rational weights and the secondary polytope (De Loera et al., 2010) of the domain lattice.

Toric patches are strictly limited to domains that can arise as convex polygons within the integer lattice. Thus not every combination of sides and degrees can be represented using toric patches, and even when it *is* possible, the control structure and/or the parameterization might be asymmetric. While various generalizations have been investigated (Battaglia and Prato, 2022), the limitation to convex domains appears to be fundamental. Toric Bézier patches have been applied successfully to hole filling (Liu et al., 2012), isogeometric analysis (Zhu et al., 2020; Ji et al., 2022) – relying on previous work on injectivity (Yu et al., 2020) – and even algebraic statistics (Duarte et al., 2023).
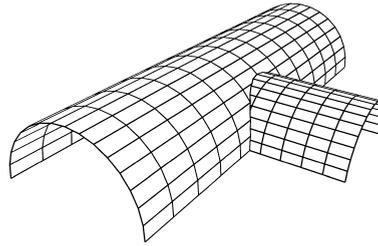
Fig. 29. A typical situation where trimming occurs between surfaces with natural (u, v) flow.

5.3.4. M-patches

Karčiauskas has proposed a multi-sided generalization of Bézier patches, called *M-patches*[2] first in a series of early papers (Karčiauskas, 1999a,b; Karčiauskas and Krasauskas, 2000), with a general theory elucidated in Karčiauskas (2003). M-patches employ generalized barycentric coordinates $\lambda = (\lambda_1, \dots, \lambda_n)$ over polygonal domains, similarly to S-patches, but their control points are organized in the form of a simple triangular spiderweb – see Fig. 27b. The n-sided polygonal domain is radially split into n triangles, that are further subdivided similarly to a degree-d Bézier triangle. The M-patch is defined as a weighted combination (Eq. (7)) of control points $\mathbf{C}_{(i;j,k)}$ indexed by tuples $(i; j, k) \in I_{n,d}$. To define the blending functions $B_{(i;j,k)}^d$, first so-called *key functions* $g_i(u, v)$ and $g(u, v)$ are assigned to the domain corners and the center point, respectively, which are then combined analogously to barycentric coordinates in a Bézier triangle:

$$B_{(i;j,k)}^d(u, v) = c_{(i;j,k)} \cdot g_i^j \, g_{i+1}^k \, g^{d-j-k}. \tag{26}$$

A natural choice for the key functions (using Wachspress GBCs over a regular domain) is $g_i = \lambda_i$ and $g = \prod_{i=1}^n \lambda_i$, but different choices are also possible, as explained by Karčiauskas (2003). The coefficients $c_{(i;j,k)}$ can be freely chosen, but certain choices help to reduce the polynomial degree of boundary cross-derivatives (Karčiauskas, 2003). Note that blending functions do not necessarily form a partition of unity, so the patch equation must be normalized.

M-patches are rational surfaces with a relatively simple structure and the ability to connect with tensor-product patches with G^k continuity. They are also closely related to other interconnected patches in some special cases, as explored by Karčiauskas (2003). However, their construction is tailored to regular or convex polygonal domains and has some ambiguity regarding the choice of key functions and interior coefficients. M-patches have been successfully applied to hole-filling, and as guide surfaces approximated by macro-patches (Karčiauskas and Peters, 2005).

6. Other schemes

In this section we give a brief overview of several other multi-sided constructions that did not fit into the interpolant or control point based categories.

6.1. Boundary controlled trimming

In CAD it is often the case that a four-sided surface needs to be trimmed where it connects to adjacent surfaces. The *trimming curves* generally cannot be exactly represented, and approximations are used instead (Marussig and Hughes, 2018). Consequently, the surfaces have only numerical C^0 connection, but exact C^0, G^1 or even G^2 continuity would be desirable, suggesting a transfinite interpolation problem.

Note that this setting is different from that of the rest of this survey, since here the (ultimately multi-sided) patch is to be derived from an originally four-sided *base surface* with a natural (u, v) flow, lying very close to the boundary constraints (represented as ribbons) – e.g. a trimmed cylinder, see Fig. 29. Given the above application, exact CAD-compatible (i.e., NURBS) export is also of crucial importance.

Martin and Reif (2022) introduced *Accurate Boundary Control* (ABC) *patches*, using inverse distance weights (see Sec. 4.1) to blend the ribbons with an interior surface (similarly to Várady et al. (2012), see Section 3.6.3). Local parameterization maps are created as $\mathbb{R}^2 \to \mathbb{R}^2$ NURBS surfaces defined over the (u, v) domain of the base patch, by fitting (s_i, d_i) parameters of projected ribbon sample points. The boundaries of the multi-sided domain are defined *implicitly* (by $d_i = 0$), however in practice these curves need not ever be explicitly computed. The resulting parameterizations only have partial coverage, where the $(i - 1)$-th and $(i + 1)$-th sides are not constrained to $s_i = 0$ and $s_i = 1$, which suffices for this particular construction. The distance parameters are clamped to 1 in the interior, creating a *plateau*. This has two beneficial effects: (i) it decreases the overall degree, and, more importantly, (ii) limits the influence of individual ribbons, so the base surface is exactly reproduced over the plateau. The resulting procedural surface can be exported to a collection of NURBS patches by subdividing it at interior knots of the boundary, and near the edges of the plateau.

---

[2] Originally referred to as T-patches in e.g. Karčiauskas and Krasauskas (2000).

(a) Footpoints in the domain   (b) Patch with isolines
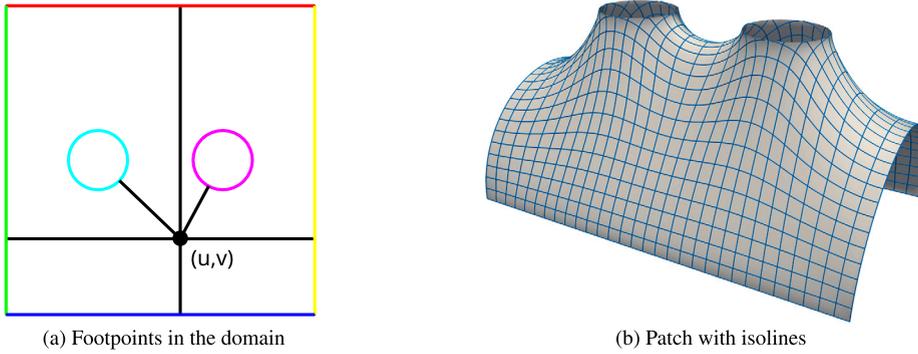
Fig. 30. Four-sided hyperbolic patch with two holes (Sabin, 1998).

Blending boundary conditions with an interior representation is also common when solving boundary value problems (Höllig et al., 2001; Marussig and Hughes, 2018), a practice with a long history going back to the work of Kantorovich (Shapiro, 2007).

In a somewhat similar approach, Sabin et al. (2022) also make use of implicit boundaries in the parametric domain to add displacements (Kosinka et al., 2015) to NURBS surfaces, using the implicit equations to define the blending functions.

6.2. General transfinite interpolation

A general conception of transfinite interpolation is given in Sabin (1996), where it is defined as a mapping $\mathcal{M} : \mathbf{B} \rightarrow \mathbf{S}$ between a parametric 3D boundary $\mathbf{B}(t) : \mathbb{R} \rightarrow \mathbb{R}^3$ and a parametric surface $\mathbf{S}(u, v) : \mathbb{R}^2 \rightarrow \mathbb{R}^3$. This formulation allows for domains with curved boundaries or cases with only one or two 'sides' (discontinuities in the boundary); it is also straightforward to extend it to handle multiple loops. While in the paper only C^0 interpolation is discussed, the discussion can easily be generalized to higher-order continuity, as well. Four major classes of methods for interpolation are identified: (1) functional optimization, (2) solving partial differential equations (PDEs), (3) boundary integrals, and (4) boundary sums. PDEs might arise in the form of optimality conditions ('variational design'), but could also be postulated without such justification. Variational surface definitions have an *implicit* character, and their evaluation might require a discretization of the domain and iterative numerical methods, see e.g. Moreton and Séquin (1992); Schneider and Kobbelt (2000); Joshi and Séquin (2007); Pan et al. (2015); Soliman et al. (2021). Solutions of PDEs can in certain cases be expressed as boundary integrals (Evans, 2010).

Following Sabin, we consider surfaces defined by an integral formula of the following general linear form:

$$\mathbf{S}(u, v) = \int_t \mathbf{B}(t) \cdot \Phi(t, u, v) \, dt, \tag{27}$$

where $\Phi(t, u, v)$ is a scalar blending function. A special case is when instead of a full integral, only a few associated points (*footpoints*) per side are used:

$$\mathbf{S}(u, v) = \sum_{i=1}^{n} \mathbf{B}(t_i(u, v)) \cdot \Psi_i(u, v). \tag{28}$$

Here $t_i(u, v)$ maps a point in the domain to the i-th side, similarly to the *side parameter* of the previous sections, and $\Psi_i(u, v)$ is the scalar blending function associated with the same side. The above equations are referred to as *elliptic* and *hyperbolic* forms, respectively, based on their similarity to solution formulae for elliptic and hyperbolic PDEs (Evans, 2010). These surfaces are often defined to minimize some kind of energy. Generally speaking, hyperbolic methods propagate boundary discontinuities along characteristic lines, while elliptic ones smooth them out. In the rest of this section, we are going to survey patches of both hyperbolic and elliptic form.

6.2.1. Side-based ('hyperbolic') patches

Sabin (1996) gives two concrete examples of hyperbolic surfaces with C^0 interpolation properties. A hyperbolic surface capable of G^1 interpolation was later proposed in Sabin (1998) that – remarkably – also handled multiply connected configurations, but was undeservedly forgotten. Each point of the surface is defined by minimizing the bending energy of curves connecting it to the footpoints on the boundary, with arc lengths fixed based on domain distances to the footpoints. A common Jacobian matrix is assumed at the point of evaluation and the bending energy is linearized as the integral of the squared second derivative. Assuming a cubic Hermite form for the curves, and inverting a 3×3 matrix (that involves familiar inverse distance weights), closed-form expressions can be derived for the surface point as well as the Jacobian. Internal holes are represented as circles, and footpoints are found using perpendicular projection, see Fig. 30. This leads to several open questions (e.g. how to handle internal loops with multiple sides or concave domains) that were not explored further in the paper.

Note that interpolant-based surfaces share some similarities with the hyperbolic formula Eq. (28), and in fact most of the patches seen in Section 4 can be written in this form. The bilinear Coons patch, notably, minimizes the functional (Farin and Hansford, 1999)
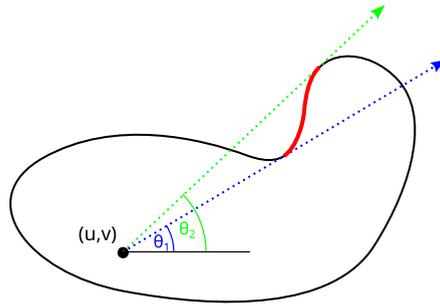
Fig. 31. Handling concave domains in mean value interpolation (Dyken and Floater, 2009). The interval shown in red has negative sign.

$$\int \left(\frac{\partial^2}{\partial u \partial v} S(u,v) \right)^2 , \tag{29}$$

and, equivalently, is the solution of the hyperbolic PDE called the draftsman's equation (Birkhoff and Gordon, 1968)

$$\frac{\partial^4}{\partial u^2 \partial v^2} S(u,v) = 0. \tag{30}$$

Sabin (1996) has also pointed out that corner discontinuities might propagate into the interior of hyperbolic patches, unless (i) a single footpoint per side is weighted by a function with corner singularities, or (ii) all corner points are included in the formula, with negative signs. This suggests that singular side blends and Boolean sums (within either interpolants or the patch equation) actually cover the range of practical hyperbolic methods.

6.2.2. Integral-based ('elliptic') patches

The idea to use a *biharmonic* PDE of the form

$$\left(\frac{\partial^2}{\partial u^2} + \frac{\partial^2}{\partial v^2} \right)^2 S(u,v) = 0 \tag{31}$$

for transfinite interpolation appeared early on in e.g. Bloor and Wilson (1989). Many subsequent works have solved similar linear PDEs using a finite element discretization (e.g. Welch and Witkin, 1992; Botsch and Kobbelt, 2004; Jacobson et al., 2010; Andrews et al., 2011) – we refer to Botsch and Sorkine (2007) for a survey of such methods. Note that to pose the biharmonic interpolation problem, not only values, but also transversal (e.g. normal) derivatives need to be prescribed along the boundary – see Vaitkus (2023) for a ribbon-based definition of biharmonic patches. As previously mentioned, solutions of linear elliptic PDEs can be expressed as boundary integrals, which plays an important role in boundary element (Ilbery et al., 2013) and Monte Carlo (Sawhney and Crane, 2020) methods.

The *moving least squares coordinates* of Manson and Schaefer (2010) also belong in this category. Similarly to Gao and Rockwood (2005), discussed in Section 4.1, they solve for a weighted least squares fit of the boundary data with inverse distance weights. Here boundary values and derivatives, as well as the locally fitted function, are assumed to be *polynomials* of degree p, and approximation is performed by integration over the whole boundary (instead of just the footpoints). The method thus exactly reproduces degree-p surfaces. Interpolation of normal derivatives is argued to be possible using appropriately chosen weights, although a proof of this claim has not been given. When the domain is polygonal, these coordinates can also be expressed in closed form.

Mean value interpolation (Floater, 2003; Hormann and Floater, 2006), originally defined on polygons, was generalized to arbitrary boundary curves with interpolation of normal derivatives in Dyken and Floater (2009). The domain can be concave and multiply connected (see Fig. 31), the only restriction is that the domain boundary cannot intersect its external medial axis, which practically excludes concave polygons (and the mild condition that a line intersects the domain boundary only finite times). The basic idea is, once again, using a (normalized) inverse distance weight based on Euclidean distances to points on the boundary.

The above method has quadratic precision, but a variant with cubic precision is published in Floater and Schulz (2008). In this work, for each point, we find a linear polynomial over the domain minimizing the integral of the linearized bending energy of all Hermite curves interpolating this linear polynomial at one end, and the boundary constraints at the other. So this can be regarded as the elliptic version of the patch proposed in Sabin (1998). C^1 interpolation is demonstrated through numerical examples, but a proof has not been given.

Specializations of this method for interpolation over polygons with cubic (Li et al., 2013) and quadratic precision (Beatson et al., 2018) were published subsequently, where function values and derivatives are prescribed only at the vertices and at edge midpoints. While the aforementioned general transfinite methods involve complicated integrals that need to be evaluated numerically, these later developments have closed-form solutions.

A recent treatise on generalized barycentric coordinates (Hormann and Sukumar, 2017, esp. Chapter 3) gives a nice overview of this line of research, although in a somewhat different context, including such (until recently) forgotten gems as the cubic precision Hermite interpolant of Gordon and Wixom (1974). This approach works on convex domains, by taking the mean of all Hermite interpolants connecting the two opposing boundary points defined by the intersection with a revolving line centered on the point
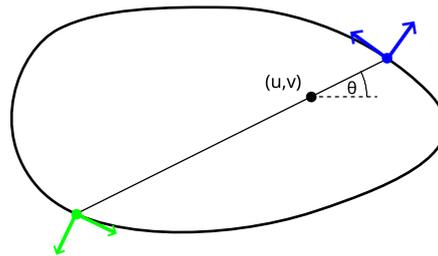
Fig. 32. The C^1 transfinite interpolation method of Gordon and Wixom (1974). A revolving line intersects the boundary of the convex domain in two points that define an Hermite interpolant based on the normal derivative of the function. This can be evaluated at the ratio of the distances to the two boundary points, resulting in a candidate value at the domain point. Then the surface is defined as the mean of all such candidate values.

of evaluation, see Fig. 32. When the domain is the unit disk, biharmonic surfaces are reproduced (Floater, 2015). A variant of this method, with superior approximation properties, was proposed by Belyaev and Fayolle (2015), who also extended it to non-convex domains (similarly to Dyken and Floater (2009), see Fig. 31).

7. Discussion

We have collected a few important points of view to compare and analyze the presented surfacing schemes.

7.1. Comparing representations

For the evaluation of the surveyed representations, we selected various criteria considered important for practical applications.

In Table 1(a) interpolant-based representations are compared. Side-based patches with singular inverse distance weights (Gregory, 1986) are very flexible and easy to implement, but occasionally suffer from quality problems. Composite ribbon patches (Salvi et al., 2014) are a great choice when n is relatively low. Midpoint patches (Salvi and Várady, 2016; Salvi et al., 2017) are especially recommended for larger n, and are also quite efficient to evaluate. Generalized Coons patches (Várady et al., 2011) can also produce excellent results, but require more complex, differentially constrained parameterizations.

Table 1(b) compares control point based representations. Toric (Krasauskas, 2002) and Zheng–Ball patches (Zheng and Ball, 1997) have some important limitations, as discussed previously. S-patches (Loop and DeRose, 1989) are less constrained, but can have a very large number of control points. M-patches (Karčiauskas, 2003), in contrast, have a quite simple structure, and might deserve more attention. The Generalized Bézier family (Várady et al., 2016) is probably a practical choice for many applications, and this scheme has been extended to multiply connected curved domains and B-spline boundaries, see an example in Fig. 1d.

The representations reviewed in Section 6 (boundary controlled trimming, hyperbolic and elliptic patches) have less in common than what would warrant a comparison, so here we discuss only elliptic patches (Table 2). Note that some of these interpolation methods are developed for different applications, but we evaluate them here in the context of ribbon-based surfacing.

The moving least squares coordinates patch (Manson and Schaefer, 2010), depending on the degree of its internal fit, can have arbitrary precision. On the other hand, its interpolation property in the general case is not proven, and while it does not assume a convex domain, in a highly concave configuration negative weights might be produced for the boundary positions. The pointwise radial minimization of Floater and Schulz (2008) can match derivatives of arbitrary order, but also lacks a formal proof. Although transfinite mean value interpolation (Dyken and Floater, 2009) supports only G^1 constraints, it can handle multiply connected surfaces, and should probably be the first choice for a surface of this type. The pseudo-harmonic interpolation method of Gordon and Wixom (1974) is very intuitive, and with its extension to concave domains, may be a good competitor. Several of these methods are derived from approximations of biharmonic surface interpolation (Jacobson et al., 2010; Vaitkus, 2023), which may be preferable when surface quality is of primary importance.

7.2. Importance of input quality

As for all free-form surfaces, appropriate input for multi-sided patches is essential to ensure that good quality surfaces are created. Boundary imperfections can propagate across the entire patch, and neither algebraic nor optimization-based methods can hope to compensate for arbitrarily bad input. It is also crucial that ribbons are 'harmonized', meaning they are constructed with some awareness of each other. This might be obvious for derivatives at patch corners – though minor incompatibilities between adjacent ribbons can be compensated by Gregory's twist – but practical experience suggests that the effect of non-adjacent ribbons should be considered as well.

It must also be noted, that any method, no matter how sophisticated, may be capable of producing poor quality results. Academic examples typically involve 'convex' patches with geometrically simple boundaries, which are easy to handle using almost any multi-sided scheme. When complex, uneven ribbons need to be interpolated, however, most methods might give unacceptable results and manual 'engineering' might become necessary. The recent work of Salvi et al. (2023) describes methods to initialize the cross-derivatives based on the given domain and modify them via control vectors, but obviously more work could be done in this direction.

Table 1
Comparison of interpolant and control point based representations.

	Boundary	Interpolation order	Interior control
Gregory (1986) Side-based patch (4.1.1)	Concave, multiply connected (Kato, 2000)	Any	By auxiliaries (Várady et al., 2012) By control points (Salvi, 2024)
Gregory (1986) Corner-based patch (4.2)	Convex	G^2 (Hall and Mullineux, 1999a)[a]	Fullness control (Salvi and Várady, 2016) [Midpoint patch]
Shi et al. (2010) Polar patch (4.1.4)	Convex[b]	Any	—
Várady et al. (2011) Generalized Coons patch (4.3)	Convex	G^1	Fullness control (Salvi et al., 2017) [Midpoint Coons patch]
Salvi et al. (2014) Composite ribbon patch (4.1.3)	Convex	G^1	—

(a) Interpolant-based representations

	Boundary (default: polynomial/convex)	Interpolation order	Refinement
Loop and DeRose (1989) S-patch (5.3.2)	Concave (Hormann and Sukumar, 2017, Chapter 8)	G^1 (Salvi, 2019)[c]	p
Várady (1991) Overlap patch (5.2.1)	—	Any (Salvi, 2022)	—
Zheng and Ball (1997) Zheng–Ball patch (5.3.1)	—	Any	p (Ball and Zheng, 2001)
Krasauskas (2002) Toric patch (5.3.3)	—	G^1 (Sun and Zhu, 2015); G^2 (Sun and Zhu, 2018)	p
Karčiauskas (2003) M-patch (5.3.4)	—	G^2	p
Várady et al. (2016) GB patch (5.1.1)	B-spline (Hettinga and Kosinka, 2020b); Concave (Salvi and Várady, 2018); Multi-connected (Várady et al., 2020)	Any	h, p (Wang et al., 2023)
Vaitkus et al. (2021) GBS patch (5.1.2)	B-spline; Multiply connected	Any	h, p

(b) Control point based representations; in the 'Refinement' column, p and h denote degree elevation and knot insertion, respectively

[a] See also Salvi and Várady (2014) on how to create suitable ribbons.
[b] Large distortions possible at the corners when angles are not near π.
[c] Shows explicit equations; higher-order interpolation is also possible theoretically.

Table 2
Comparison of elliptic representations (cf. Sec. 6.2.2).

	Domain	Interpolation order	Precision
Gordon and Wixom (1974) Pseudo-harmonic interpolation	Concave (Belyaev and Fayolle, 2015)	Any	3 (cubic case)
Floater and Schulz (2008) Pointwise radial minimization	Concave, multiply connected	Any?	3 (cubic case)
Dyken and Floater (2009) Transfinite mean value interpolation	Concave, multiply connected	G^1	2
Manson and Schaefer (2010) Moving least squares coordinates	Convex / Mildly concave	G^1?	Any
Jacobson et al. (2010) Polyharmonic surface	Curved, multiply connected	Any	3 (cubic case)

7.3. Relevance of different patch types

While quadrilateral patches dominate free-form design, in the majority of cases 3-, 5- and 6-sided patches appear most frequently within general topology surfaces. Patches with more than 6 sides rarely occur in practice, mostly due to the perceived difficulties and/or limitations of the mathematical representations, so when such surfaces arise they are often artificially subdivided. Some

designers may not trust existing tools for creating many-sided surfaces and rely on custom workflows instead. Two- or one-sided patches occur only in exceptional situations.

Concerning surfaces with concave corners we have some doubts. Concave angles represent a mathematical abstraction; they cannot be practically manufactured and lead to extreme stress concentrations, thus typically such corners are artificially split or rounded off into small fillets. As a result, often concave boundary arcs with high curvature variation are produced, and these create a challenge for the multi-sided schemes (cf. Fig. 8). Patches that can represent multiply connected surfaces also constitute a recent development. We hope that once the design community becomes aware of methods supporting complex boundaries and internal holes, they will adopt these modeling tools in place of their currently used alternatives.

7.4. Continuity

The degree of geometric continuity between multi-sided patches (or equivalently, between patches and their ribbons) is another important issue to consider. The ability to create watertight connections, i.e., C^0 continuity, is a key advantage over trimming techniques, and is essential for many downstream applications. Exact G^1 continuity can also be important in practice. We would like to point out that surfaces (e.g. ribbons) represented as industry-standard C^2 cubic B-splines cannot meet with G^1 continuity in a general configuration, unless regularity is reduced to C^1 (Peters and Fan, 2010) or degree is elevated to at least 5 (Shi et al., 2004). Regarding curvature continuity, we advise caution, as algebraic G^2 constraints can be difficult to handle and typically degrade surface fairness. Class-A surfacing might demand G^3 continuity (Autodesk, 2023), which would be even more problematic to enforce. Recall, however, that analytic smoothness is neither sufficient nor necessary to achieve good shape, and small imperfections of the mathematical representation might be imperceptible on an actually manufactured object. It is our experience that relaxing geometric continuity to be satisfied only approximately (NG^k) can help to substantially improve surface quality, which is also supported by similar observations reported in e.g. Karčiauskas and Peters (2015); Kiciak (2017); Mosbach et al. (2022); Salvi et al. (2023).

7.5. Degrees of freedom

In the classical view of free-form modeling, degrees of freedom that determine the geometry are also used for manual shape design. One reason for the popularity of control point (e.g. Bézier and NURBS) representations is their built-in fairness: evenly placed control points unambiguously define shapes of reasonable quality that react in an intuitive manner to the designer's actions. Note that for the highest quality Class-A surfaces, designers are advised to use Bézier representations of degree 7 at most (Autodesk, 2023). It is also well-established among practitioners that interpolating degrees of freedom (e.g. interpolating polynomial splines) are less suitable for creating high-quality shapes compared to control points. These observations carry over to the multi-sided setting as well, but new challenges also arise, namely the degrees of freedom could become too numerous for manual design, even in simple situations. A possible solution might be to impose a separation on the degrees of freedom: some DoFs are used to satisfy interpolation constraints, the remaining are either made available for manual design, or set automatically by some optimization procedure (see the next section). Surfaces defined by interconnected control structures often possess a natural hierarchy of representations (organized by degree or 'depth') that could be exploited in the spirit of multi-resolution or subdivision methods (Zorin, 2006). Defining interconnected, hierarchical control structures for non-convex, multiply connected configurations is one of the most important open problems in this area.

7.6. Evaluation vs. optimization

Designing with geometric elements – including control points, lines, planes and various vector quantities – is a procedure that is considered natural and intuitive by most users. Most of the surveyed methods generate a surface from these entities using algebraic formulae that might have geometric interpretation, but global properties (e.g. fairness) are difficult to predict or control in this manner. Defining shapes by functional optimization is another possibility that comes with its own set of challenges, namely how to define a suitable functional for the task at hand, and how the shape can be tuned in an intuitive manner. The first problem is already quite daunting as a large variety of functionals could be chosen and even combined together to formalize the (ultimately subjective and application-dependent) notion of 'fairness'. As for the second problem, experience shows that fairness and design intent may sometimes be in conflict, and if a particular notion of fairness is 'baked' into the representation, it might be difficult achieve the particular effect intended by the designer. Similar observations have been made by Peters (2003); Peters and Karčiauskas (2008). Our view is that some combination of the two approaches would be beneficial, as mentioned in the previous section.

7.7. Conversion into standard formats

The great majority of genuine multi-sided patches cannot be easily exported into a standard NURBS representation, which is considered a fundamental limitation of these methods. In principle, many of the patches (over convex polygonal domains) can be represented exactly as high-degree rational Bézier patches (Salvi et al., 2021), but the complexity of these surfaces is generally considered impractical. Some patches (particularly those over non-polygonal domains) do not have closed-form expressions, requiring numerical methods for their evaluation. One might argue that numerical methods could be tuned to achieve whatever tolerance is required by a given application (at least if the mathematical objects to be approximated are sufficiently well-behaved). Our view, however, is even more optimistic: inspired by existing work on guided surfaces (Peters and Karčiauskas, 2008), we consider

it plausible that a patchwork of standard NURBS surfaces can be associated with multi-sided (even multiply connected) patches that maintain ribbon interpolation while approximating the patch interior. When C^0 discontinuities within a given tolerance are acceptable, trimmed NURBS approximation is also a viable option. For certain applications (e.g. when the surface only needs to be displayed or 3D printed) even a polygonal mesh discretization might suffice, which can easily be produced for any multi-sided representation.

8. Conclusion and open problems

In this survey, we have given an overview of the state of the art in parametric, genuinely multi-sided surface representations. While the field has shown remarkable progress in recent years, we conclude with a list of open problems that remain to be solved.

1. *Better local parameters*. Local parameterizations fundamentally determine the quality of multi-sided surfaces. In our practical experience interpolants are more sensitive to the parameters than blending functions, which results in poor performance when interpolant-based methods are generalized to non-convex domains. The theoretical understanding of these effects could be improved, and enhanced parameterization methods could also be developed, with better control over the properties of the mapping. New methods for differentially constrained parameters (see Section 3.4.2) should be investigated as well.

2. *Interior control for general domains*. As already mentioned in Section 7.5, we believe that finding natural interior control structures and associated blending functions for patches over curved, multiply connected domains would constitute a major achievement in this area.

3. *Conversion into standard formats*. The non-standard nature of multi-sided representations remains one of their major shortcomings. See our discussion in Section 7.7 on potential remedies.

4. *Efficient and accurate interrogations*. Calculating partial derivatives of the complex expressions defining multi-sided surfaces, as required for computation of normal vectors, curvatures, etc., can be challenging. Modern automatic differentiation tools might help alleviating these difficulties. Certain constructions (such as those based on solving PDEs) might even need domain discretization for evaluation. Instead of the typical C^0 mesh, a discretization with higher regularity (Hughes et al., 2021) could be used, allowing derivatives to be directly evaluated.

5. *Shape optimization*. In current practice, ribbon generation relies on relatively simple heuristics (Salvi and Várady, 2014; Salvi et al., 2023). Global shape optimization is an attractive, if challenging alternative – see the discussion in Section 7.6 – but multi-sided generalizations of procedural/local interpolation methods (Antonelli et al., 2016; Shirman and Séquin, 1990) should also be investigated in the future.

6. *Application to analysis*. Some of the surveyed representations have already been used to define function spaces for the purpose of (isogeometric) numerical analysis, including quadrilateral Gregory patches (Puso and Laursen, 2002; Greco and Cuomo, 2021), toric patches (Zhu et al., 2020; Ji et al., 2022) and Generalized Bézier patches (Wang et al., 2023). An important related issue is efficient numerical integration (quadrature) – we mention the pioneering recent work of Zhou et al. (2023) on Gregory patches – but given the high regularity of the representations, it might also be possible to solve PDEs by collocation.

7. *Volumetric representations*. With the development of additive manufacturing and isogeometric analysis, there is an increased need for volumetric representations (Elber, 2023). Due to the difficulty of hexahedral mesh generation, it is important to allow general topology configurations in this context. We are aware of only a few attempts to extend multi-sided constructions to a 3D setting, such as Generalized Coons patches (Randrianarivony, 2011) and Charrot–Gregory patches (Hu and Lin, 2019), leaving much to be done in this area.

We have presented the great variety of genuinely multi-sided surface representations, and hope that this will lead to wider practical adoption in the near future; in particular, when further progress is made on the above open issues.

CRediT authorship contribution statement

Tamás Várady: Writing – original draft, Conceptualization, Writing – review & editing. **Péter Salvi:** Writing – original draft, Conceptualization, Writing – review & editing. **Márton Vaitkus:** Writing – original draft, Conceptualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

References

Aigerman, N., Poranne, R., Lipman, Y., 2015. Seamless surface mappings. ACM Trans. Graph. 34, 72. https://doi.org/10.1145/2766921.

Andrews, J., Joshi, P., Carr, N., 2011. A linear variational system for modelling from curves. Comput. Graph. Forum 30, 1850–1861. https://doi.org/10.1111/j.1467-8659.2011.01966.x.

Antonelli, M., Beccari, C.V., Casciola, G., 2016. High quality local interpolation by composite parametric surfaces. Comput. Aided Geom. Des. 46, 103–124. https://doi.org/10.1016/j.cagd.2016.06.005.

Autodesk, 2023. Alias golden rules. https://help.autodesk.com/view/ALIAS/2023/ENU/. Getting Started chapter.

Bajaj, C.L., Chen, J., Xu, G., 1995. Modeling with cubic A-patches. ACM Trans. Graph. 14, 103–133. https://doi.org/10.1145/221659.221662.

Ball, A.A., Zheng, J., 2001. Degree elevation for n-sided surfaces. Comput. Aided Geom. Des. 18, 135–147. https://doi.org/10.1016/S0167-8396(01)00020-6.

Barnhill, R.E., 1977. Representation and approximation of surfaces. In: Mathematical Software III, pp. 69–120.

Barnhill, R.E., Birkhoff, G., Gordon, W.J., 1973. Smooth interpolation in triangles. J. Approx. Theory 8, 114–128. https://doi.org/10.1016/0021-9045(73)90020-8.

Barnhill, R.E., Gregory, J.A., 1975. Compatible smooth interpolation in triangles. J. Approx. Theory 15, 214–225. https://doi.org/10.1016/0021-9045(75)90104-5.

Battaglia, F., Prato, E., 2022. Nonrational polytopes and fans in toric geometry. arXiv:2205.00417.

Beatson, R., Floater, M.S., Kåshagen, C.E., 2018. Hermite mean value interpolation on polygons. Comput. Aided Geom. Des. 60, 18–27. https://doi.org/10.1016/j.cagd.2018.01.002.

Belyaev, A.G., Fayolle, P.A., 2015. On transfinite Gordon–Wixom interpolation schemes and their extensions. Comput. Graph. 51, 74–80. https://doi.org/10.1016/j.cag.2015.05.010.

Birkhoff, G., Gordon, W.J., 1968. The draftsman's and related equations. J. Approx. Theory 1, 199–208. https://doi.org/10.1016/0021-9045(68)90024-5.

Bloomenthal, J., Bajaj, C., Blinn, J., Wyvill, B., Cani, M.P., Rockwood, A., Wyvill, G., 1997. Introduction to Implicit Surfaces. Morgan Kaufmann.

Bloor, M.I., Wilson, M.J., 1989. Generating blend surfaces using partial differential equations. Comput. Aided Des. 21, 165–171. https://doi.org/10.1016/0010-4485(89)90071-7.

Graf von Bothmer, H.C., Ranestad, K., Sottile, F., 2010. Linear precision for toric surface patches. Found. Comput. Math. 10, 37–66. https://doi.org/10.1007/s10208-009-9052-6.

Botsch, M., Kobbelt, L., 2004. An intuitive framework for real-time freeform modeling. ACM Trans. Graph. 23, 630–634. https://doi.org/10.1145/1015706.1015772.

Botsch, M., Sorkine, O., 2007. On linear variational surface deformation methods. IEEE Trans. Vis. Comput. Graph. 14, 213–230. https://doi.org/10.1109/TVCG.2007.1054.

Cashman, T.J., 2012. Beyond Catmull–Clark? A survey of advances in subdivision surface methods. Comput. Graph. Forum 31, 42–61. https://doi.org/10.1111/j.1467-8659.2011.02083.x.

Chang, Q., Deng, C., Hormann, K., 2023. Maximum likelihood coordinates. Comput. Graph. Forum 42, e14908. https://doi.org/10.1111/cgf.14908.

Charrot, P., Gregory, J.A., 1984. A pentagonal surface patch for computer aided geometric design. Comput. Aided Geom. Des. 1, 87–94. https://doi.org/10.1016/0167-8396(84)90006-2.

Chiyokura, H., 1986. Localized surface interpolation method for irregular meshes. In: Advanced Computer Graphics: Proceedings of Computer Graphics Tokyo'86, pp. 3–19.

Chiyokura, H., Kimura, F., 1983. Design of solids with free-form surfaces. Comput. Graph. 17, 289–298. https://doi.org/10.1145/964967.801160.

Coons, S.A., 1967. Surfaces for computer-aided design of space forms. Technical Report MAC-TR-41. MIT.

De Loera, J., Rambau, J., Santos, F., 2010. Triangulations: Structures for Algorithms and Applications. Algorithms and Computation in Mathematics, vol. 25. Springer.

Deng, C., Chang, Q., Hormann, K., 2020. Iterative coordinates. Comput. Aided Geom. Des. 79, 101861. https://doi.org/10.1016/j.cagd.2020.101861.

Duarte, E., Hollering, B., Wiesmann, M., 2023. Toric fiber products in geometric modeling. In: Geometric Science of Information, pp. 494–503.

Dyken, C., Floater, M.S., 2009. Transfinite mean value interpolation. Comput. Aided Geom. Des. 26, 117–134. https://doi.org/10.1016/j.cagd.2007.12.003.

Elber, 2023. A review of a B-spline based volumetric representation: design, analysis and fabrication of porous and/or heterogeneous geometries. Comput. Aided Des. 163, 103587. https://doi.org/10.1016/j.cad.2023.103587.

Evans, L.C., 2010. Partial Differential Equations, 2nd ed. Graduate Studies in Mathematics, vol. 19. AMS.

Farin, G., 2002. Curves and Surfaces for CAGD: A Practical Guide, 5th ed. Morgan Kaufmann.

Farin, G., Hansford, D., 1999. Discrete Coons patches. Comput. Aided Geom. Des. 16, 691–700. https://doi.org/10.1016/S0167-8396(99)00031-X.

Farouki, R.T., Szafran, N., Biard, L., 2010. Construction and smoothing of triangular Coons patches with geodesic boundary curves. Comput. Aided Geom. Des. 27, 301–312. https://doi.org/10.1016/j.cagd.2010.01.004.

Fiorot, J.C., Gibaru, O., 2002. Blowing up: application to G^2-continuous 8-sided filling patch. Numer. Math. 92, 257–287. https://doi.org/10.1007/s002110100366.

Floater, M.S., 2003. Mean value coordinates. Comput. Aided Geom. Des. 20, 19–27. https://doi.org/10.1016/S0167-8396(03)00002-5.

Floater, M.S., 2015. Generalized barycentric coordinates and applications. Acta Numer. 24, 161–214. https://doi.org/10.1017/S0962492914000129.

Floater, M.S., Lai, M.J., 2016. Polygonal spline spaces and the numerical solution of the Poisson equation. SIAM J. Numer. Anal. 54, 797–824. https://doi.org/10.1137/15M101155X.

Floater, M.S., Schulz, C., 2008. Pointwise radial minimization: Hermite interpolation on arbitrary domains. In: Computer Graphics Forum, pp. 1505–1512.

Gao, K., Rockwood, A., 2005. Multi-sided attribute based modeling. In: Mathematics of Surfaces XI. IMA, pp. 219–232.

Garanzha, V., Kaporin, I., Kudryavtseva, L., Protais, F., Ray, N., Sokolov, D., 2021. Foldover-free maps in 50 lines of code. ACM Trans. Graph. 40, 102. https://doi.org/10.1145/3450626.3459847.

García-Puente, L.D., Sottile, F., Zhu, C., 2011. Toric degenerations of Bézier patches. ACM Trans. Graph. 30, 110. https://doi.org/10.1145/2019627.2019629.

Gibaru, O., 2004. Tensorial rational surfaces with base points via massic vectors. SIAM J. Numer. Anal. 42, 1415–1434. https://doi.org/10.1137/S0036142903420972.

Goldman, R., 2004. Multisided arrays of control points for multisided Bézier patches. Comput. Aided Geom. Des. 21, 243–261. https://doi.org/10.1016/j.cagd.2003.10.003.

Gordon, W.J., Hall, C.A., 1973. Transfinite element methods: blending-function interpolation over arbitrary curved element domains. Numer. Math. 21, 109–129. https://doi.org/10.1007/BF01436298.

Gordon, W.J., Wixom, J.A., 1974. Pseudo-harmonic interpolation on convex domains. SIAM J. Numer. Anal. 11, 909–933. https://doi.org/10.1137/0711072.

Gravesen, J., Evgrafov, A., Nguyen, D.M., Nørtoft, P., 2014. Planar parametrization in isogeometric analysis. In: 8th International Conference on Mathematical Methods for Curves and Surfaces, pp. 189–212.

Greco, L., Cuomo, M., 2021. An implicit G^1-conforming bi-cubic interpolation for the analysis of smooth and folded Kirchhoff–Love shell assemblies. Comput. Methods Appl. Mech. Eng. 373, 113476. https://doi.org/10.1016/j.cma.2020.113476.

Gregory, J.A., 1974. Smooth interpolation without twist constraints. In: Computer Aided Geometric Design. University of Utah, pp. 71–87.

Gregory, J.A., 1986. *n*-sided surface patches. In: The Mathematics of Surfaces. IMA, pp. 217–232.

Gregory, J.A., Charrot, P., 1980. A C^1 triangular interpolation patch for computer-aided geometric design. Comput. Graph. Image Process. 13, 80–87. https://doi.org/10.1016/0146-664X(80)90117-3.

Gregory, J.A., Hahn, J.M., 1987a. Geometric continuity and convex combination patches. Comput. Aided Geom. Des. 4, 79–89. https://doi.org/10.1016/0167-8396(87)90026-4.

Gregory, J.A., Hahn, J.M., 1987b. Polygonal patches of high order continuity. Technical Report TR/01/87. Brunel University.

Gregory, J.A., Hahn, J.M., 1989. A C^2 polygonal surface patch. Comput. Aided Geom. Des. 6, 69–75. https://doi.org/10.1016/0167-8396(89)90007-1.

Gregory, J.A., Lau, V.K., Zhou, J., 1990. Smooth parametric surfaces and *n*-sided patches. In: Computation of Curves and Surfaces, pp. 457–498.

Grimm, C.M., Zorin, D., 2006. Surface modeling and parameterization with manifolds. In: SIGGRAPH '06 Courses. AMC, pp. 1–81.

Grošelj, J., Praprotnik, A.Š., 2022. Exact sphere representations over Platonic solids based on rational multisided Bézier patches. Comput. Aided Geom. Des. 98, 102148. https://doi.org/10.1016/j.cagd.2022.102148.

Hagen, H., Pottmann, H., 1989. Curvature continuous triangular interpolants. In: Mathematical Methods in Computer Aided Geometric Design, pp. 373–384.

Hall, R., Mullineux, G., 1997. Shape modification of Gregory patches. In: The Mathematics of Surfaces VII. IMA, pp. 393–408.

Hall, R., Mullineux, G., 1999a. Continuity between Gregory-like patches. Comput. Aided Geom. Des. 16, 197–216. https://doi.org/10.1016/S0167-8396(98)00044-2.

Hall, R., Mullineux, G., 1999b. The Zheng–Ball construction without twist constraints. Comput. Aided Geom. Des. 16, 165–175. https://doi.org/10.1016/S0167-8396(98)00041-7.

Hartmann, E., 2001a. Implicit G^n-blending of vertices. Comput. Aided Geom. Des. 18, 267–285. https://doi.org/10.1016/S0167-8396(01)00030-9.

Hartmann, E., 2001b. Parametric G^n blending of curves and surfaces. Vis. Comput. 17, 1–13. https://doi.org/10.1007/PL00013398.

Hermann, 1996. G^2 interpolation of free form curve networks by biquintic Gregory patches. Comput. Aided Geom. Des. 13, 873–893. https://doi.org/10.1016/S0167-8396(96)00013-1.

Hermann, T., Peters, J., Strotman, T., 2011. A geometric constraint on curve networks suitable for smooth interpolation. Comput. Aided Des. 43, 741–746. https://doi.org/10.1016/j.cad.2010.05.007.

Hermann, T., Peters, J., Strotman, T., 2012. Curve networks compatible with G^2 surfacing. Comput. Aided Geom. Des. 29, 219–230. https://doi.org/10.1016/j.cagd.2011.10.003.

Hettinga, G.J., Brals, R., Kosinka, J., 2019. Colour interpolants for polygonal gradient meshes. Comput. Aided Geom. Des. 74, 101769. https://doi.org/10.1016/j.cagd.2019.101769.

Hettinga, G.J., Kosinka, J., 2018. Multisided generalisations of Gregory patches. Comput. Aided Geom. Des. 62, 166–180. https://doi.org/10.1016/j.cagd.2018.03.005.

Hettinga, G.J., Kosinka, J., 2020a. Multisided B-spline patches over extraordinary regions. In: Smart Tools and Apps for Graphics. Eurographics Association.

Hettinga, G.J., Kosinka, J., 2020b. A multisided C^2 B-spline patch over extraordinary vertices in quadrilateral meshes. Comput. Aided Des. 127, 102855. https://doi.org/10.1016/j.cad.2020.102855.

Hinz, J., Möller, M., Vuik, C., 2018. Elliptic grid generation techniques in the framework of isogeometric analysis applications. Comput. Aided Geom. Des. 65, 48–75. https://doi.org/10.1016/j.cagd.2018.03.023.

Höllig, K., Reif, U., Wipper, J., 2001. Weighted extended B-spline approximation of Dirichlet problems. SIAM J. Numer. Anal. 39, 442–462. https://doi.org/10.1137/S0036142900373208.

Hormann, K., Floater, M.S., 2006. Mean value coordinates for arbitrary planar polygons. ACM Trans. Graph. 25, 1424–1441. https://doi.org/10.1145/1183287.1183295.

Hormann, K., Polthier, K., Sheffer, A., 2008. Mesh parameterization: theory and practice. In: ACM SIGGRAPH ASIA 2008 Courses, pp. 1–87.

Hormann, K., Sukumar, N., 2008. Maximum entropy coordinates for arbitrary polytopes. Comput. Graph. Forum 27, 1513–1520. https://doi.org/10.1111/j.1467-8659.2008.01292.x.

Hormann, K., Sukumar, N., 2017. Generalized Barycentric Coordinates in Computer Graphics and Computational Mechanics. CRC Press.

Hosaka, M., Kimura, F., 1984. Non-four-sided patch expressions with control points. Comput. Aided Geom. Des. 1, 75–86. https://doi.org/10.1016/0167-8396(84)90005-0.

Hoschek, J., Lasser, D., 1993. Fundamentals of Computer Aided Geometric Design. A K Peters.

Hu, C., Lin, H., 2019. Gregory solid construction for polyhedral volume parameterization by sparse optimization. Appl. Math. J. Chin. Univ. 34, 340–355. https://doi.org/10.1007/s11766-019-3697-y.

Hughes, T.J., Sangalli, G., Takacs, T., Toshniwal, D., 2021. Smooth multi-patch discretizations in isogeometric analysis. Handb. Numer. Anal. 22, 467–543. https://doi.org/10.1016/bs.hna.2020.09.002.

Ilbery, P., Kendall, L., Concolato, C., McCosker, M., 2013. Biharmonic diffusion curve images from boundary elements. ACM Trans. Graph. 32, 219. https://doi.org/10.1145/2508363.2508426.

Jacobson, A., Tosun, E., Sorkine, O., Zorin, D., 2010. Mixed finite elements for variational surface modeling. Comput. Graph. Forum 29, 1565–1574. https://doi.org/10.1111/j.1467-8659.2010.01765.x.

Ji, Y., Li, J., Yu, Y., Zhu, C., 2022. *h*-refinement method for toric parameterization of planar multi-sided computational domain in isogeometric analysis. Comput. Aided Geom. Des. 93, 102065. https://doi.org/10.1016/j.cagd.2022.102065.

Joshi, P., Meyer, M., DeRose, T., Green, B., Sanocki, T., 2007. Harmonic coordinates for character articulation. ACM Trans. Graph. 26, 71. https://doi.org/10.1145/1276377.1276466.

Joshi, P., Séquin, C., 2007. Energy minimizers for curvature-based surface functionals. Comput-Aided Des. Appl. 4, 607–617. https://doi.org/10.1080/16864360.2007.10738495.

Karčiauskas, K., 1999a. On five- and six-sided rational surface patches. Preprint.

Karčiauskas, K., 1999b. Rational *m*-sided Sabin and Hosaka–Kimura like surface patches. Preprint.

Karčiauskas, K., 2000. Algebraic version of convex combination patches. In: LMD Mokslo Darbai III. VU-MII, pp. 178–182.

Karčiauskas, K., 2003. Rational *M*-patches and tensor-border patches. In: Topics in Algebraic Geometry and Geometric Modeling. AMS, pp. 101–130.

Karčiauskas, K., Krasauskas, R., 2000. Comparison of different multisided patches using algebraic geometry. In: Curve and Surface Design. AFA, pp. 163–172.

Karčiauskas, K., Peters, J., 2005. Polynomial C^2 spline surfaces guided by rational multisided patches. In: Computational Methods for Algebraic Spline Surfaces. ESF, pp. 119–134.

Karčiauskas, K., Peters, J., 2015. Can bi-cubic surfaces be class A? Comput. Graph. Forum 34, 229–238. https://doi.org/10.1111/cgf.12711.

Kato, K., 1991. Generation of *n*-sided surface patches with holes. Comput. Aided Des. 23, 676–683. https://doi.org/10.1016/0010-4485(91)90020-W.

Kato, K., 2000. N-sided surface generation from arbitrary boundary edges. In: Curve and Surface Design. AFA, pp. 173–182.

Kiciak, P., 2017. Geometric Continuity of Curves and Surfaces. Synthesis Lectures on Visual Computing. Morgan & Claypool Publishers.

Kosinka, J., Sabin, M.A., Dodgson, N.A., 2015. Control vectors for splines. Comput. Aided Des. 58, 173–178. https://doi.org/10.1016/j.cad.2014.08.028.

Krasauskas, R., 2001. Shape of toric surfaces. In: Proceedings of Spring Conference on Computer Graphics. IEEE, pp. 55–62.

Krasauskas, R., 2002. Toric surface patches. Adv. Comput. Math. 17, 89–113. https://doi.org/10.1023/A:1015289823859.

Kuriyama, S., 1994. Surface modelling with an irregular network of curves via sweeping and blending. Comput. Aided Des. 26, 597–606. https://doi.org/10.1016/0010-4485(94)90102-3.

Lancaster, P., Šalkauskas, K., 1986. Curve and Surface Fitting: An Introduction. Academic Press.

Langer, T., Belyaev, A., Seidel, H.P., 2008. Mean value Bézier maps. In: Advances in Geometric Modeling and Processing, pp. 231–243.

Levin, A., 2000. Filling *n*-sided holes using combined subdivision schemes. In: Curve and Surface Design. AFA, pp. 221–228.

Li, J., Ji, Y., Zhu, C., 2021. De Casteljau algorithm and degree elevation of toric surface patches. J. Syst. Sci. Complex. 34, 21–46. https://doi.org/10.1007/s11424-020-9370-y.

Li, X., Ju, T., Hu, S., 2013. Cubic mean value coordinates. ACM Trans. Graph. 32, 126. https://doi.org/10.1145/2461912.2461917.

Liseikin, V.D., 2017. Grid Generation Methods, 3rd ed. Scientific Computation. Springer.

Liu, H., Zhu, C., Li, C., 2012. Constructing *n*-sided toric surface patches from boundary curves. J. Inf. Comput. Sci. 9, 737–743.

Lodha, S., 1993. Filling *n*-sided holes. In: Modeling in Computer Graphics. IAC-CNR, pp. 319–345.

Loop, C.T., DeRose, T.D., 1989. A multisided generalization of Bézier surfaces. ACM Trans. Graph. 8, 204–234. https://doi.org/10.1145/77055.77059.

Loop, C.T., DeRose, T.D., 1990. Generalized B-spline surfaces of arbitrary topology. In: SIGGRAPH '90. ACM, pp. 347–356.

Malraison, P., 1998. A bibliography for *n*-sided surfaces. In: The Mathematics of Surfaces VIII. IMA, pp. 419–430.

Malraison, P., 2000. *n*-sided surfaces: a survey. In: Curve and Surface Design. AFA, pp. 247–256.

Manson, J., Schaefer, S., 2010. Moving least squares coordinates. In: Computer Graphics Forum, pp. 1517–1524.

Martin, F., Reif, U., 2022. Trimmed spline surfaces with accurate boundary control. In: Geometric Challenges in Isogeometric Analysis, pp. 123–148.

Marussig, B., Hughes, T.J., 2018. A review of trimming in isogeometric analysis: challenges, data exchange and simulation aspects. Arch. Comput. Methods Eng. 25, 1059–1127. https://doi.org/10.1007/s11831-017-9220-9.

Marussig, B., Reif, U., 2022. Surface patches with rounded corners. Comput. Aided Geom. Des. 97, 102134. https://doi.org/10.1016/j.cagd.2022.102134.

Moreton, H.P., Séquin, C.H., 1992. Functional Optimization for Fair Surface Design. ACM SIGGRAPH Computer Graphics, vol. 26, pp. 167–176.

Mosbach, D., Schladitz, K., Hamann, B., Hagen, H., 2022. A local approach for computing smooth B-spline surfaces for arbitrary quadrilateral base meshes. J. Comput. Inf. Sci. Eng. 22. https://doi.org/10.1115/1.4051121.

Nasri, A.H., Sabin, M.A., 2002. Taxonomy of interpolation constraints on recursive subdivision surfaces. Vis. Comput. 18, 382–403. https://doi.org/10.1007/s003710100155.

Nielson, G.M., 1979. The side-vertex method for interpolation in triangles. J. Approx. Theory 25, 318–336. https://doi.org/10.1016/0021-9045(79)90020-0.

Nielson, G.M., 1987. A transfinite, visually continuous, triangular interpolant. In: Geometric Modeling: Algorithms and New Trends. SIAM, pp. 235–246.

Pan, H., Liu, Y., Sheffer, A., Vining, N., Li, C., Wang, W., 2015. Flow aligned surfacing of curve networks. ACM Trans. Graph. 34, 127. https://doi.org/10.1145/2766990.

Peters, J., 1991. Smooth interpolation of a mesh of curves. Constr. Approx. 7, 221–246. https://doi.org/10.1007/BF01888155.

Peters, J., 2002. Geometric continuity. In: Handbook of Computer Aided Geometric Design, pp. 193–227.

Peters, J., 2003. Smoothness, fairness and the need for better multi-sided patches. In: Topics in Algebraic Geometry and Geometric Modeling. AMS, pp. 55–64.

Peters, J., 2019. Splines for meshes with irregularities. SMAI J. Comput. Math. 5, 161–183. https://doi.org/10.5802/smai-jcm.57.

Peters, J., Fan, J., 2010. On the complexity of smooth spline surfaces from quad meshes. Comput. Aided Geom. Des. 27, 96–105. https://doi.org/10.1016/j.cagd.2009.09.003.

Peters, J., Karčiauskas, K., 2008. An introduction to guided and polar surfacing. In: 7th International Conference on Mathematical Methods for Curves and Surfaces, pp. 299–315.

Pla-Garcia, N., Vigo-Anglada, M., Cotrina-Navau, J., 2006. *N*-sided patches with B-spline boundaries. Comput. Graph. 30, 959–970. https://doi.org/10.1016/j.cag.2006.05.001.

Plowman, D., Charrot, P., 1996. A practical implementation of vertex blend surfaces using an *n*-sided patch. In: The Mathematics of Surfaces VI. IMA, pp. 67–78.

Puso, M.A., Laursen, T.A., 2002. A 3D contact smoothing method using Gregory patches. Int. J. Numer. Methods Eng. 54, 1161–1194. https://doi.org/10.1002/nme.466.

Qin, K., Li, Y., Deng, C., 2023. Blending Bézier patch for multi-sided surface modeling. Comput. Aided Geom. Des. 105, 102222. https://doi.org/10.1016/j.cagd.2023.102222.

Randrianarivony, M., 2011. On transfinite interpolations with respect to convex domains. Comput. Aided Geom. Des. 28, 135–149. https://doi.org/10.1016/j.cagd.2010.10.003.

Rockwood, A., Gao, K., Farrell, M., 2016. SuperD: SubD modeling without subdivision. In: SIGGRAPH '16 Talks. ACM, p. 8.

Rustamov, R.M., 2008. Boundary element formulation of harmonic coordinates. Technical Report. Purdue University.

Rvachev, V.L., Sheiko, T.I., Shapiro, V., Tsukanov, I., 2001. Transfinite interpolation over implicitly defined sets. Comput. Aided Geom. Des. 18, 195–220. https://doi.org/10.1016/S0167-8396(01)00015-2.

Sabin, M.A., 1983. Non-rectangular surface patches suitable for inclusion in a B-spline surface. In: Eurographics Conference Proceedings. Eurographics Association.

Sabin, M.A., 1986. Some negative results in *n* sided patches. Comput. Aided Des. 18, 38–44. https://doi.org/10.1016/S0010-4485(86)80009-4.

Sabin, M.A., 1994. A symmetric domain for a 6-sided patch. In: The Mathematics of Surfaces IV. IMA, p. 185.

Sabin, M.A., 1996. Transfinite surface interpolation. In: The Mathematics of Surfaces VI. IMA, pp. 517–534.

Sabin, M.A., 1998. Further transfinite surface developments. In: The Mathematics of Surfaces VIII. IMA, pp. 161–174.

Sabin, M.A., Fellows, C., Kosinka, J., 2022. CAD model details via curved knot lines and truncated powers. Comput. Aided Des. 143, 103137. https://doi.org/10.1016/j.cad.2021.103137.

Salvi, P., 2019. G^1 hole filling with S-patches made easy. In: Proceedings of the 12th Conference of the Hungarian Association for Image Processing and Pattern Recognition. KÉPAF. arXiv:2002.11109.

Salvi, P., 2020. A multi-sided generalization of the C^0 Coons patch. In: Proceedings of the Workshop on the Advances of Information Technology. BME-IIT, pp. 110–111. arXiv:2002.11347.

Salvi, P., 2021. Editing the interior of arbitrary surfaces using C^∞ displacement blends. In: Proceedings of the Workshop on the Advances of Information Technology. BME, pp. 35–38.

Salvi, P., 2022. A circular parameterization for multi-sided patches. In: Proceedings of the Tenth Hungarian Conference on Computer Graphics and Geometry. NJSZT, pp. 22–26. arXiv:2305.07534.

Salvi, P., 2024. Intuitive interior control for multi-sided patches with arbitrary boundaries. Comput-Aided Des. Appl. 21, 143–154. https://doi.org/10.14733/cadaps.2024.143-154.

Salvi, P., Gál, P., 2023. Tessellation of Zheng–Ball patches. In: Proceedings of the Workshop on the Advances of Information Technology. BME, pp. 35–39.

Salvi, P., Kovács, I., Várady, T., 2017. Computationally efficient transfinite patches with fullness control. In: Proceedings of the Workshop on the Advances of Information Technology. BME, pp. 96–100. arXiv:2002.11212.

Salvi, P., Vaitkus, M., Várady, T., 2023. Constrained modeling of multi-sided patches. Comput. Graph. 114, 86–95. https://doi.org/10.1016/j.cag.2023.05.020.

Salvi, P., Várady, T., 2014. G^2 surface interpolation over general topology curve networks. Comput. Graph. Forum 33, 151–160. https://doi.org/10.1111/cgf.12483.

Salvi, P., Várady, T., 2015. Comparison of two *n*-patch representations in curve network-based design. In: Proceedings of the 10th Conference of the Hungarian Association for Image Processing and Pattern Recognition. KÉPAF, pp. 612–624.

Salvi, P., Várady, T., 2016. Multi-sided surfaces with fullness control. In: Proceedings of the Eighth Hungarian Conference on Computer Graphics and Geometry. NJSZT, pp. 61–69.

Salvi, P., Várady, T., 2018. Multi-sided Bézier surfaces over concave polygonal domains. Comput. Graph. 74, 56–65. https://doi.org/10.1016/j.cag.2018.05.006.

Salvi, P., Várady, T., Miura, K.T., 2018. Approximating point clouds by generalized Bézier surfaces. In: Proceedings of the Workshop on the Advances of Information Technology. BME, pp. 61–66.

Salvi, P., Várady, T., Rockwood, A., 2013. Transfinite surface interpolation over specific curvenet configurations. In: 14th IMA Conference on Mathematics of Surfaces. IMA, pp. 309–326.

Salvi, P., Várady, T., Rockwood, A., 2014. Ribbon-based transfinite surfaces. Comput. Aided Geom. Des. 31, 613–630. https://doi.org/10.1016/j.cagd.2014.06.006.

Salvi, P., Várady, T., Rockwood, A., 2021. Notes on the CAD-compatible conversion of multi-sided surfaces. Comput-Aided Des. Appl. 18, 156–169. https://doi.org/10.14733/cadaps.2021.156-169.

Sawhney, R., Crane, K., 2020. Monte Carlo geometry processing: a grid-free approach to PDE-based methods on volumetric domains. ACM Trans. Graph. 39, 123. https://doi.org/10.1145/3386569.3392374.

Schaefer, S., Warren, J., Zorin, D., 2004. Lofting curve networks using subdivision surfaces. In: Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, pp. 103–114.

Schneider, R., Kobbelt, L., 2000. Generating fair meshes with G^1 boundary conditions. In: Geometric Modeling and Processing 2000. IEEE, pp. 251–261.

Sederberg, T.W., 1985. Piecewise algebraic surface patches. Comput. Aided Geom. Des. 2, 53–59. https://doi.org/10.1016/0167-8396(85)90007-X.

Shapiro, V., 2007. Semi-analytic geometry with R-functions. Acta Numer. 16, 239–303. https://doi.org/10.1017/S096249290631001X.

Shepard, D., 1968. A two-dimensional interpolation function for irregularly-spaced data. In: Proceedings of the 1968 23rd ACM National Conference. ACM, pp. 517–524.

Shi, K., Yong, J., Sun, J., Paul, J.C., 2010. G^n blending multiple surfaces in polar coordinates. Comput. Aided Des. 42, 479–494. https://doi.org/10.1016/j.cad.2009.11.009.

Shi, K., Yong, J., Tang, L., Sun, J., Paul, J.C., 2013. Polar NURBS surface with curvature continuity. In: Computer Graphics Forum, pp. 363–370.

Shi, X., Wang, T., Yu, P., 2004. A practical construction of G^1 smooth biquintic B-spline surfaces over arbitrary topology. Comput. Aided Des. 36, 413–424. https://doi.org/10.1016/S0010-4485(03)00111-8.

Shirman, L.A., Séquin, C.H., 1990. Local surface interpolation with shape parameters between adjoining Gregory patches. Comput. Aided Geom. Des. 7, 375–388. https://doi.org/10.1016/0167-8396(90)90001-8.

Sipos, Á., Várady, T., Salvi, P., Vaitkus, M., 2020. Multi-sided implicit surfacing with I-patches. Comput. Graph. 90, 29–42. https://doi.org/10.1016/j.cag.2020.05.009.

Smith, J., Schaefer, S., 2015. Selective degree elevation for multi-sided Bézier patches. In: Computer Graphics Forum, pp. 609–615.

Soliman, Y., Chern, A., Diamanti, O., Knöppel, F., Pinkall, U., Schröder, P., 2021. Constrained Willmore surfaces. ACM Trans. Graph. 40, 112. https://doi.org/10.1145/3450626.3459759.

Sottile, F., 2003. Toric ideals, real toric varieties, and the moment map. In: Topics in Algebraic Geometry and Geometric Modeling. AMS, pp. 225–240.

Southern, R., Dodgson, N.A., 2007. A smooth manifold based construction of approximating lofted surfaces. Technical Report UCAM-CL-TR-699. University of Cambridge, Computer Laboratory.

Sun, L., Zhu, C., 2015. G^1 continuity between toric surface patches. Comput. Aided Geom. Des. 35, 255–267. https://doi.org/10.1016/j.cagd.2015.03.017.

Sun, L., Zhu, C., 2018. Curvature continuity conditions between adjacent toric surface patches. Comput. Graph. Forum 37, 469–477. https://doi.org/10.1111/cgf.13583.

Szörfi, J., Várady, T., 2022. Polyhedral design with concave and multi-connected faces. In: Proceedings of the Tenth Hungarian Conference on Computer Graphics and Geometry. NJSZT, pp. 28–35.

Vaitkus, M., 2023. A note on ribbon-based biharmonic surface patches. arXiv:2311.12822.

Vaitkus, M., Várady, T., Salvi, P., Sipos, Á., 2021. Multi-sided B-spline surfaces over curved, multi-connected domains. Comput. Aided Geom. Des. 89, 102019. https://doi.org/10.1016/j.cagd.2021.102019.

Várady, T., 1987. Survey and new results in n-sided patch generation. In: The Mathematics of Surfaces II. IMA, pp. 203–235.

Várady, T., 1991. Overlap patches: a new scheme for interpolating curve networks with n-sided regions. Comput. Aided Geom. Des. 8, 7–27. https://doi.org/10.1016/0167-8396(91)90046-E.

Várady, T., Benkő, P., Kós, G., Rockwood, A., 2001. Implicit surfaces revisited – I-patches. In: Geometric Modelling, pp. 323–335.

Várady, T., Rockwood, A., Salvi, P., 2011. Transfinite surface interpolation over irregular n-sided domains. Comput. Aided Des. 43, 1330–1340. https://doi.org/10.1016/j.cad.2011.08.028.

Várady, T., Salvi, P., Karikó, G., 2016. A multi-sided Bézier patch with a simple control structure. Comput. Graph. Forum 35, 307–317. https://doi.org/10.1111/cgf.12833.

Várady, T., Salvi, P., Kovács, I., 2017. Enhancement of a multi-sided Bézier surface representation. Comput. Aided Geom. Des. 55, 69–83. https://doi.org/10.1016/j.cagd.2017.05.002.

Várady, T., Salvi, P., Rockwood, A., 2012. Transfinite surface interpolation with interior control. Graph. Models 74, 311–320. https://doi.org/10.1016/j.gmod.2012.03.003.

Várady, T., Salvi, P., Rockwood, A., 2013. Transfinite surface patches using curved ribbons. In: Eurographics 2013 – Short Papers. Eurographics Association, pp. 5–8.

Várady, T., Salvi, P., Vaitkus, M., Sipos, Á., 2020. Multi-sided Bézier surfaces over curved, multi-connected domains. Comput. Aided Geom. Des. 78, 101828. https://doi.org/10.1016/j.cagd.2020.101828.

Wachspress, E.L., 1975. A Rational Finite Element Basis. Academic Press.

Wang, C.C., Tang, K., 2004. Non-self-overlapping structured grid generation on an n-sided surface. Int. J. Numer. Methods Fluids 46, 961–982. https://doi.org/10.1002/fld.791.

Wang, M., Ji, Y., Zhu, C., 2023. Degree elevation and knot insertion for generalized Bézier surfaces and their application to isogeometric analysis. J. Comput. Math. https://doi.org/10.4208/jcm.2301-m2022-0116.

Wang, W., Jüttler, B., Zheng, D., Liu, Y., 2008. Computation of rotation minimizing frames. ACM Trans. Graph. 27, 2. https://doi.org/10.1145/1330511.1330513.

Wang, X., Cheng, F., Barsky, B.A., 1999. Blending, smoothing and interpolation of irregular meshes using N-sided Várady patches. In: SMA '99. ACM, pp. 212–222.

Warren, J., 1992. Creating multisided rational Bézier surfaces using base points. ACM Trans. Graph. 11, 127–139. https://doi.org/10.1145/130826.130828.

Warren, J.D., 1994. Multi-sided rational surface patches with independent boundary control. In: The Mathematics of Surfaces V. IMA, pp. 281–294.

Welch, W., Witkin, A., 1992. Variational surface modeling. Comput. Graph. 26, 157–166. https://doi.org/10.1145/133994.134033.

Yang, L., Liang, Y., Li, X., Zhang, C., Lin, G., Sheffer, A., Schaefer, S., Keyser, J., Wang, W., 2023. Neural parametric surfaces for shape modeling. arXiv:2309.09911.

Ying, L., Zorin, D., 2004. A simple manifold-based construction of surfaces of arbitrary smoothness. ACM Trans. Graph. 23, 271–275. https://doi.org/10.1145/1015706.1015714.

Yu, Y., Ji, Y., Zhu, C., 2020. An improved algorithm for checking the injectivity of 2D toric surface patches. Comput. Math. Appl. 79, 2973–2986. https://doi.org/10.1016/j.camwa.2020.01.001.

Zheng, J., 2001. The n-sided control point surfaces without twist constraints. Comput. Aided Geom. Des. 18, 129–134. https://doi.org/10.1016/S0167-8396(01)00019-X.

Zheng, J., Ball, A.A., 1997. Control point surfaces over non-four-sided areas. Comput. Aided Geom. Des. 14, 807–821. https://doi.org/10.1016/S0167-8396(97)00007-1.

Zheng, J., Zhang, J., 2002. Texture mapping on irregular topology surface. In: Proceedings Sixth International Conference on Information Visualisation. IEEE, pp. 323–330.

Zheng, J., Zhang, J., Zhou, H., Shen, L., 2005. Smooth spline surface generation over meshes of irregular topology. Vis. Comput. 21, 858–864. https://doi.org/10.1007/s00371-005-0345-8.

Zhou, J., Barendrecht, P.J., Bartoň, M., Kosinka, J., 2023. Numerical quadrature for Gregory quads. Appl. Math. Comput. 453, 128051. https://doi.org/10.1016/j.amc.2023.128051.

Zhou, P., Qian, W.H., 2010. Polyhedral vertex blending with setbacks using rational S-patches. Comput. Aided Geom. Des. 27, 233–244. https://doi.org/10.1016/j.cagd.2010.01.001.

Zhu, X., Ji, Y., Zhu, C., Hu, P., Ma, Z.D., 2020. Isogeometric analysis for trimmed CAD surfaces using multi-sided toric surface patches. Comput. Aided Geom. Des. 79, 101847. https://doi.org/10.1016/j.cagd.2020.101847.

Zorin, D., 2006. Modeling with multiresolution subdivision surfaces. In: ACM SIGGRAPH 2006 Courses, pp. 30–50.

Zou, M., Ju, T., Carr, N., 2013. An algorithm for triangulating multiple 3D polygons. Comput. Graph. Forum 32, 157–166. https://doi.org/10.1111/cgf.12182.

Zubè, S., 2000. The n-sided toric patches and $\mathcal{A}$-resultants. Comput. Aided Geom. Des. 17, 695–714. https://doi.org/10.1016/S0167-8396(00)00024-8.