



## Notes on the CAD-Compatible Conversion of Multi-Sided Surfaces

Péter Salvi<sup>1</sup> , Tamás Várady<sup>2</sup>  and Alyn Rockwood<sup>3</sup>

<sup>1</sup>Budapest University of Technology and Economics, [salvi@iit.bme.hu](mailto:salvi@iit.bme.hu)

<sup>2</sup>Budapest University of Technology and Economics, [varady@iit.bme.hu](mailto:varady@iit.bme.hu)

<sup>3</sup>Boulder Graphics LLC, USA, [alynrock@gmail.com](mailto:alynrock@gmail.com)

Corresponding author: Péter Salvi, [salvi@iit.bme.hu](mailto:salvi@iit.bme.hu)

**Abstract.** We investigate genuine multi-sided surface representations that can be converted into standard tensor product format, such as NURBS. Multi-sided patches offer versatility in shape design, and permit smooth, watertight connections between adjacent patches using prescribed cross-derivatives; however, they can seldom be utilized by the majority of CAD/CAM systems, which handle surfaces only in standard data formats. Tensor product surfaces, on the other hand, may be too rigid for shape design, and trimmed patches can only be smoothly connected up to user-defined tolerances. This motivates the search for schemes that benefit from both representations.

We analyze four multi-sided surface representations that allow precise conversion into tensor product format, namely S-patches, Warren's patch, Kato's patch and a variant of the Charrot–Gregory patch. We compare these schemes from various aspects; in particular, we deal with surface equations, singularities, degrees of the converted surfaces, the control structure built and the computational efficiency of the conversion. Several examples help to gain deeper insights into the problem.

**Keywords:** multi-sided surfaces, trimmed surfaces, S-patch, Charrot–Gregory patch

**DOI:** <https://doi.org/10.14733/cadaps.2021.156-169>

## 1 INTRODUCTION

The majority of aesthetic objects are represented by free-form shapes, and modeling these naturally involves multi-sided (i.e., non-four-sided) surfaces. The mathematical representation of such patches is still an active area in CAGD, and although a great variety of approaches have been published, none of the genuine  $n$ -sided formulations have been standardized so far.

On the other hand, commercial CAD/CAM systems and related application programs only accept data in standard formats, such as tensor product NURBS surfaces. For this reason, a widely applied practice is to convert multi-sided surfaces into a CAD-compatible representation either by (i) approximating them with larger

quadrilaterals and *trimming* away the exterior part beyond the boundaries, or (ii) *splitting* them into smaller four-sided patches. Both techniques have their deficiencies. They only approximate the original multi-sided surface, and in general, trimming cannot ensure even precise  $C^0$  continuity between adjacent patches. In the splitting scheme the subdividing curves in the interior reduce the overall continuity of the surface.

Ideally, we would like to have an  $n$ -sided patch that:

- i) can be used for design (has intuitive controls),
- ii) can be attached to adjacent patches with  $G^1$  or higher continuity,
- iii) and can also be represented *accurately* as a tensor product NURBS surface.

The above problem can be resolved, if the multi-sided surfaces can be represented as rational polynomials of two parametric variables. Then they can be directly converted into NURBS form, without either changing the surfaces or harming continuity. The result will be a collection of *watertight trimmed surfaces*.

Some of the well-known multi-sided schemes allow computing a trimmed rational bi-parametric representation. Our goal in this paper is to review these and discuss the difficulties of the conversion process. We are going to provide further insights into specific computational and geometric problems that have not been discussed elsewhere and are useful for analyzing the “pros and cons” of these representations.

The paper is structured as follows. In Section 2, we review four pure multi-sided schemes, and investigate their conversions to tensor product NURBS. In Section 3, issues of the conversion process are discussed in details, with emphasis on the quality of the generated control network, and its relation to singularities. Some more complex test examples and comparisons follow in Section 4, showing actual high-degree examples with control grids, and a summary of our findings concludes the paper.

## 2 MULTI-SIDED SURFACES

In the following, we are going to review four multi-sided representations: the S-patch [6], and the surface schemes of Warren [12], Kato [5], and Charrot & Gregory [1]. Our conversion approach is presented in Sections 2.1.1–2.1.2 specifically for S-patches, but it will be adapted to other representations in Sections 2.3 and 2.4.

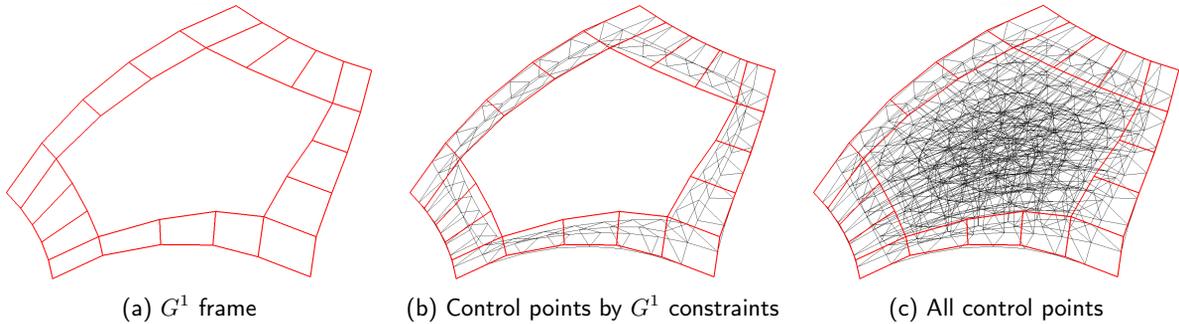
### 2.1 S-Patch

The S-patch of Loop & DeRose [6] is a generalization of the Bézier triangle, or, more precisely, a Bézier simplex mapping from  $(n - 1)$ D to 3D, where the  $n$  coordinates of the domain are supplied by generalized barycentric coordinates. The *depth* ( $d$ ) of an S-patch is the number of *de Casteljau* steps required to evaluate a surface point, i.e., something similar to the degree of a Bézier triangle (but not the degree of the S-patch itself). S-patches have many nice properties, and are known to be convertible into tensor product rational Bézier surfaces of degree  $d(n - 2)$ .

An  $n$ -sided S-patch is defined over a regular  $n$ -gon, parameterized by generalized barycentric coordinates  $\lambda = (\lambda_1, \dots, \lambda_n)$  [4]. Its control points  $\{P_s\}$  are labeled by  $n$  non-negative integers  $\mathbf{s} = (s_1, \dots, s_n)$ , whose sum is the depth of the surface. Then points on the surface are defined by the equation

$$S(\lambda) = \sum_{\mathbf{s}} P_{\mathbf{s}} \cdot B_{\mathbf{s}}^d(\lambda) = \sum_{\mathbf{s}} P_{\mathbf{s}} \cdot \binom{d}{\mathbf{s}} \cdot \prod_{i=1}^n \lambda_i^{s_i}, \quad (1)$$

where  $B_{\mathbf{s}}^d(\lambda)$  are Bernstein polynomials with multinomial coefficients, and the sum is computed over all possible values of  $\mathbf{s}$ .



**Figure 1:** Creating an S-patch from a  $G^1$  frame.

One drawback of this representation is its large number of control points, which renders it inconvenient for interactive design. For example, a five-sided patch of depth 5 has 126 control points, while a 4-sided quintic tensor product patch has only 36. A possible workaround is to use a  $G^1$  frame (Fig. 1a) for design that defines the tangent planes at the boundaries. After increasing the depth by 3, these boundary constraints can be interpolated [7], and the remaining interior control points can be set by some heuristic to generate a smooth surface [9], see Figures 1b–1c.

The CAD-compatible conversion presented in [6] is a two-step process: first convert the surface into a four-sided S-patch, and then to a tensor product patch. The first step is based on the composition of Bézier simplexes [2], which has very high complexity. Even using a more efficient simplex composition algorithm [3], converting a modest-sized S-patch still requires minutes of computation on today's machines [10].

### 2.1.1 Parameterization Using Implicit Line Equations

Here we propose an alternative conversion process. Since a Bézier simplex is polynomial, the only problem is how to express the generalized barycentric coordinates as a rational polynomial of the  $(u, v)$  parameters on the 2D domain. Using Wachspress coordinates [4],  $\{\lambda_i\}$  can be expressed as

$$\lambda_i(u, v) = \prod_{\substack{j=1 \\ j \notin \{i-1, i\}}}^n h_j(u, v) \bigg/ \sum_{k=1}^n \prod_{\substack{j=1 \\ j \notin \{k-1, k\}}}^n h_j(u, v), \quad (2)$$

where the indexing is cyclic, and  $h_j(u, v)$  is a *distance function* from the  $j$ -th side of the domain polygon. For the sake of brevity, we will use the notation

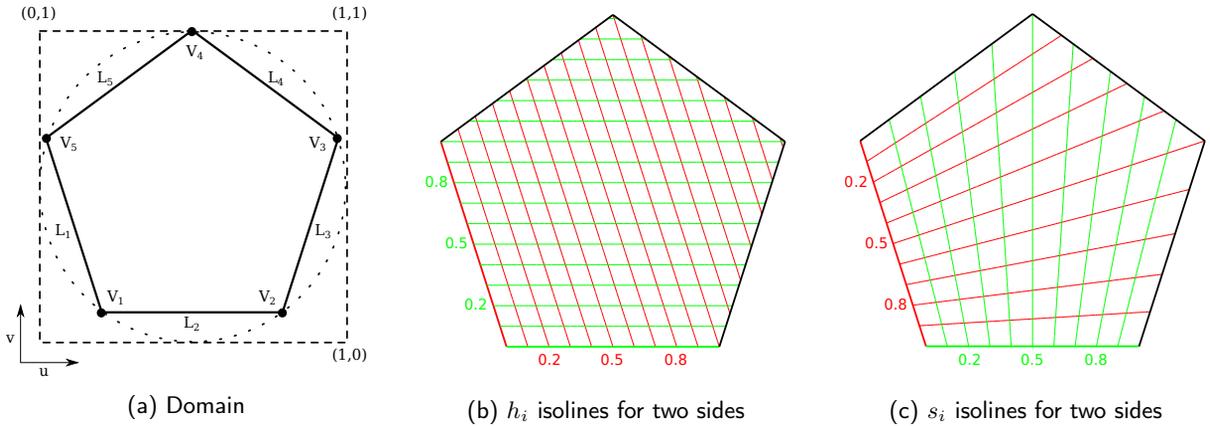
$$H_j^k(u, v) = \prod_{\substack{j=1 \\ j \notin J}}^n h_j^k(u, v), \quad (3)$$

where  $J$  is an index set to exclude certain terms. Then  $\lambda_i(u, v) = H_{i-1, i}^1(u, v) / \sum_{k=1}^n H_{k-1, k}^1(u, v)$ .

The distance function should vanish at the base domain edge, and increase monotonically as we get farther from it. The implicit equation of the line containing the base edge is suitable for this purpose, and is also a linear polynomial, so the Wachspress coordinates can be expressed as rational polynomials of degree  $n - 2$ . We normalize the distances so they take on the value 1 at the vertices adjacent to the side.

Formally, let

$$V_i = \left( \frac{1}{2} + \frac{1}{2} \cos(2\pi \cdot i/n), \frac{1}{2} + \frac{1}{2} \sin(2\pi \cdot i/n) \right), \quad i = 1 \dots n \quad (4)$$



**Figure 2:** Parameterization.

denote the vertices of the regular  $n$ -sided domain inside the  $[0, 1] \times [0, 1]$  square, see Figure 2a. Since a line  $L$  is defined by the implicit equation  $L(u, v) = Au + Bv + C = 0$ , the coefficients can be unambiguously defined by three equations. We define  $n$  lines  $L_i$  ( $i = 1 \dots n$ ) by the following constraints:

$$L_i(V_{i-1}) = L_i(V_i) = 0, \quad L_i(V_{i-2}) = L_i(V_{i+1}) = 1. \tag{5}$$

(Note that because of the symmetry of the regular polygon, the above four equations constrain only three degrees of freedom.) The distance function is then defined as

$$h_i(u, v) = L_i(u, v), \tag{6}$$

see constant parameter lines in Figure 2b.

### 2.1.2 Conversion to NURBS

With the above definition of generalized barycentric coordinates, the patch equation (1) becomes a rational vector polynomial in  $u$  and  $v$ . Since  $|s| = \sum_{i=1}^n s_i = d$ , all terms of the sum have the same denominator,

$$\left( \sum_{k=1}^n H_{k-1,k}^1(u, v) \right)^d, \tag{7}$$

which is a polynomial of degree  $d(n - 2)$ . It is easy to see that the rational degree  $\delta$  of the whole patch is also the same.

This means that we can represent an  $n$ -sided S-patch of depth  $d$  by a rational Bézier surface of  $\delta \times \delta$  degrees. In order to determine the positions of the control points, we still need to change from the power basis to the Bernstein basis.

Assuming that the coefficients of a bi-degree  $\delta$  polynomial  $p(u, v)$  are given in a matrix  $M$  such that

$$p(u, v) = \begin{bmatrix} 1 & u & u^2 & \dots & u^\delta \end{bmatrix} M \begin{bmatrix} 1 & v & v^2 & \dots & v^\delta \end{bmatrix}^\top, \tag{8}$$

the Bézier coefficients are computed as  $N = C^\top M C$ , where  $C = \{c_{ij}\}$  is the upper triangular matrix with elements

$$c_{ij} = \binom{j}{i} / \binom{\delta}{i}, \quad i, j = 0 \dots \delta, \quad i \leq j. \tag{9}$$

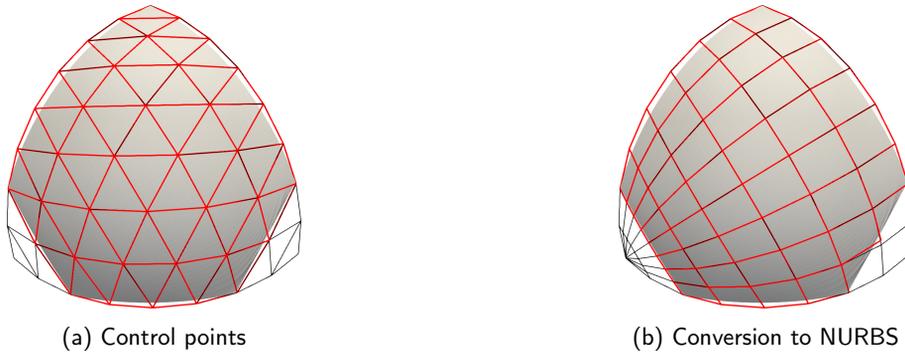


Figure 3: Warren's 5-sided patch.

Then the same polynomial is expressed as

$$p(u, v) = \left[ B_0^\delta(u) \ B_1^\delta(u) \ \dots \ B_\delta^\delta(u) \right] N \left[ B_0^\delta(v) \ B_1^\delta(v) \ \dots \ B_\delta^\delta(v) \right]^T. \quad (10)$$

Control point positions and the corresponding weights are computed as homogeneous coordinates, by calculating the coefficients for the numerator and denominator of Eq. (1) separately, and assigning the latter as the extra *weight* coordinate.

With this method, the Bézier control points of the tensor product representation can be located by straightforward computation, which takes only milliseconds.

## 2.2 Warren's Patch

Warren [12] created multi-sided patches from Bézier triangles by assigning  $0/0$  *base points* to some of the control points, essentially cutting off corners, and thus creating 5- and 6-sided surfaces; see an example in Figure 3a. We can characterize Warren's patch with the number  $m_i$  of control rows "trimmed" from each corner. Using the indices  $s = (s_1, s_2, s_3)$  introduced in Section 2.1,

$$P_s = 0/0, \quad \text{when } s_2 + s_3 < m_1 \text{ or } s_3 + s_1 < m_2 \text{ or } s_1 + s_2 < m_3. \quad (11)$$

For example, in Figure 3a two control rows are cut from two corners, so  $m_1 = 2$ ,  $m_2 = 2$  and  $m_3 = 0$ .

A simple NURBS conversion is also shown in the paper, using the degenerate transformation

$$\lambda_1 = (1 - u)v, \quad \lambda_2 = uv, \quad \lambda_3 = 1 - v. \quad (12)$$

Applying this to Eq. (1) results in a tensor product surface; its control points can be computed using Eqs. (8–10). Figure 3b shows an example. Note that the control points outside the patch boundaries are actually base points and are drawn here only for better comprehension of the control net structure.

A nice property of this patch is that the "remaining" control points define the behavior of the boundary in the same way as in a normal Bézier triangle, i.e., the first control row defines its position as a Bézier curve, the second its first derivatives etc.

It is important to see that not all degree configurations are available. The example in Figure 3, for example, has boundary curves with the following degrees, starting from the bottom side:  $[4, 2, 6, 6, 2]$ . In the common case when all boundaries have the same degree, a 6-sided patch is easily created from a triangle of degree  $3d$ , but due to its asymmetric construction, a 5-sided patch always has edge curves of different degrees. A  $3d$ -degree triangle suffices here, as well, but some of the boundaries need to be degree elevated. For example,

a 5-sided quintic patch can be generated from a 15-degree Bézier triangle by cutting 5 control rows from two corners, resulting in boundaries of degrees [5, 5, 10, 10, 5], so two boundary curves need to be elevated to degree 10.

In addition, using control points with zero weight is not a standard practice, and is not supported by many systems. Meshing also presents a problem, as a uniform grid on the domain would result in distorted triangles, since the “trimmed” boundaries correspond to corners of the domain triangle.

### 2.3 Kato’s Patch

Kato [5] proposed a surface defined as the transfinite interpolation of boundary curves with cross-derivatives:

$$S_{\text{Kato}}(u, v) = \sum_{i=1}^n R_i(s_i(u, v), h_i(u, v)) \Gamma_i^e(u, v), \quad (13)$$

where  $\Gamma_i^e(u, v)$  is a singular blending function (see below), and  $R_i(s_i, h_i)$  is a quadrilateral *ribbon* smoothly interpolating the  $i$ -th side. The cross-degree of the ribbon (and thus the number of cross-derivative constraints) is denoted by  $e$ .

Aside from the distance parameter  $h_i$ , defined as in Section 2.1.1, this representation also uses a *side parameter*  $s_i$  that takes on values from 0 to 1 as it sweeps from one adjacent side to the other, see Figure 2c. One such function is

$$s_i(u, v) = \frac{L_{i-1}(u, v)}{L_{i-1}(u, v) + L_{i+1}(u, v)}, \quad (14)$$

which gives a rational polynomial parameterization.

The blending function is given as

$$\Gamma_i^e(u, v) = H_i^{e+1}(u, v) / \sum_{k=1}^n H_k^{e+1}(u, v), \quad (15)$$

with  $H_i^{e+1}(u, v)$  defined as in Eq. (3). Note that the denominator vanishes for the corner points of the domain polygon, but the surface is well-defined there.

When the ribbons are given as Bézier surfaces of the form

$$R_i(s_i, h_i) = \sum_{j=0}^d \sum_{k=0}^e P_{j,k}^i B_j^d(s_i) B_k^e(h_i), \quad (16)$$

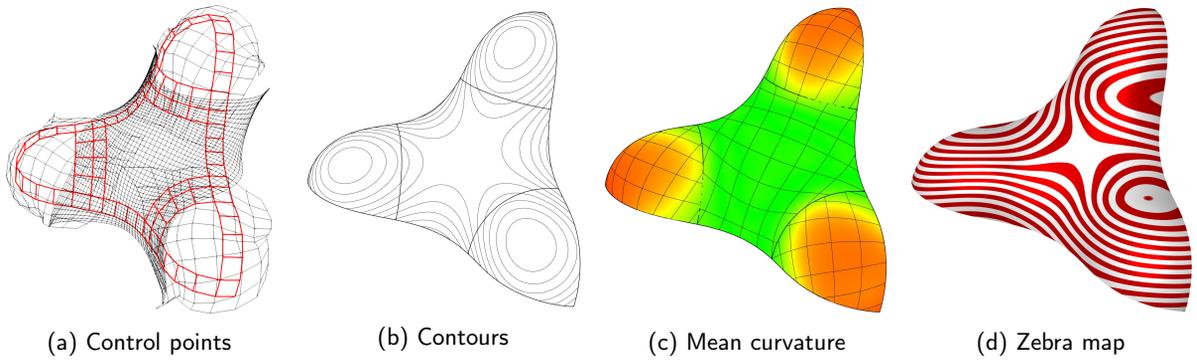
with  $P_{j,k}^i$  denoting the control points, the whole patch becomes a rational polynomial, and can be converted to a tensor product Bézier patch with rational degree  $nd + (n-1)(e+1) + e$ , see Appendix A for the details. In particular, using a  $G^1$  frame ( $e = 1$ ), the degree will be  $nd + 2n - 1$ .

### 2.4 Charrot–Gregory Patch

The Charrot–Gregory patch [1] combines corner interpolants, but it can also be formulated equivalently using ribbon surfaces [11]. Let  $I_{i-1,i}(s_{i-1}, s_i)$  denote the corner interpolant based on sides  $i-1$  and  $i$ , parameterized by the corresponding side parameters. (Note that on a regular domain the  $s_i$  parameters defined above will be the same as the radial parameterization in the original paper.)

The corner interpolant can be written as

$$I_{i-1,i}(s_{i-1}, s_i) = R_{i-1}(s_{i-1}, s_i) + R_i(s_i, 1 - s_{i-1}) - Q_{i-1,i}(s_{i-1}, s_i), \quad (17)$$



**Figure 4:** Trebol model.

where  $R_{i-1}$  and  $R_i$  are linear ribbons (Eq. (16) with  $e = 1$ ), and  $Q_{i-1,i}$  is a correction patch defined by

$$Q_{i-1,i}(s_{i-1}, s_i) = P_{00}^i + s_i d(P_{10}^i - P_{00}^i) + (1 - s_{i-1}) d(P_{01}^i - P_{00}^i) + s_i (1 - s_{i-1}) d^2(P_{11}^i - P_{10}^i - P_{01}^i + P_{00}^i). \quad (18)$$

Finally, the Charrot–Gregory surface itself is formulated as

$$S_{CG}(u, v) = \sum_{i=1}^n I_{i-1,i}(s_{i-1}(u, v), s_i(u, v)) \Gamma_{i-1,i}(u, v), \quad (19)$$

where  $\Gamma_{i-1,i}(u, v)$  is the blending function

$$\Gamma_{i-1,i}(u, v) = H_{i-1,i}^2(u, v) / \sum_{k=1}^n H_{k-1,k}^2(u, v). \quad (20)$$

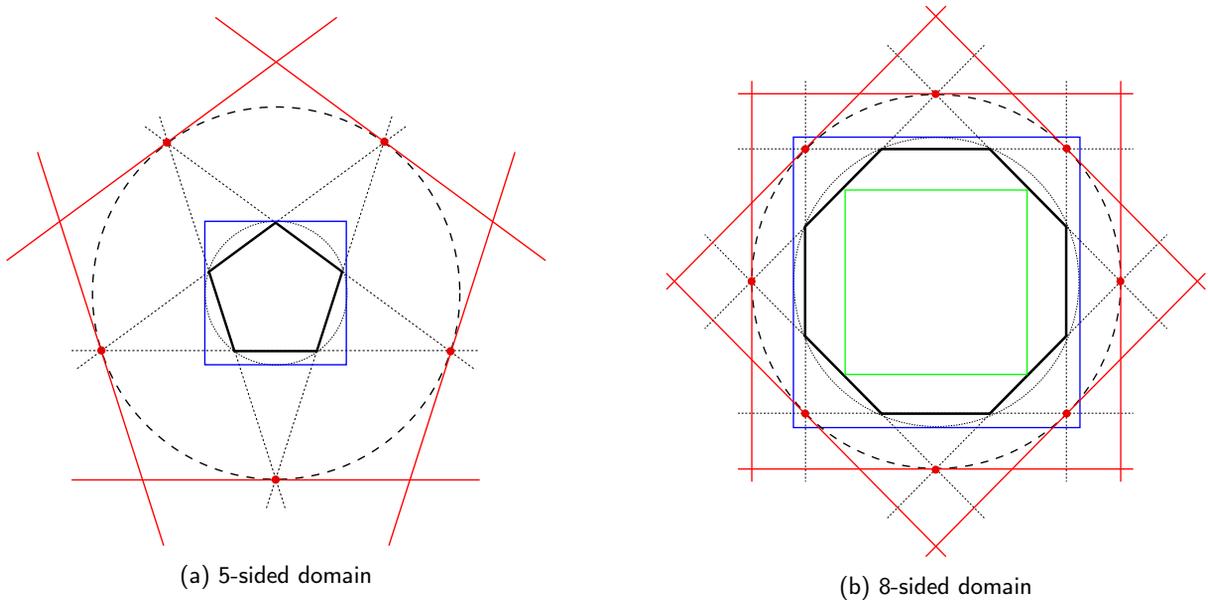
Conversion is done the same way as in Section 2.1.2; the converted patch is of degree  $nd + 2(n - 2)$ , see details in Appendix A. Note that the degree is relatively high because of the rational parameterization. For triangular surfaces, we can use distance parameters instead, thereby reducing the overall degree to  $d + 3$ , see Appendix B. Figure 4 shows an example with one 6-sided and three 3-sided surfaces.

### 3 DISCUSSION

One aspect of the tensor product conversion we have not touched on before is the *quality* of the control net. Aside from Warren's patch, which has singular control points, all the other representations have singularities in or outside their domains. When a singular point is close to the domain of the tensor product patch (i.e., the unit square), the control points in the vicinity show erratic behavior.

S-patches are singular on the circle that goes through the intersections of the lines containing the domain edges, as the denominator of Wachspress coordinates becomes zero on the *adjoint curve* of the domain, see details in [4]. The Charrot–Gregory patch is singular at lines parallel to the edges, touching the adjoint circle at the intersection points, since the denominator of  $s_i$  vanishes. Kato's patch also has these singularities, while it is also singular at the corners of the domain, where the denominator of the blending function vanishes.

Figure 5 illustrates this for 5-sided and 8-sided domains. Here the blue rectangle represents the domain of the tensor product surface; the dashed circle shows the singularities of S-patches, while the red lines are the singularities of Charrot–Gregory and Kato patches. The effect of these singularities depend on their proximity to the unit square – consequently quite disastrous for the 8-sided case, see Figure 6a.



**Figure 5:** Singularities around the domain.

We can summarize the effect of singularities on the different surface representations as follows. The triangular S-patch, which is not rational, has a stable control structure, while for  $n = 5$  and  $n = 6$  some control points may lie far from the multi-sided surface. For  $n \geq 7$  the converted tensor product surfaces are likely to have badly oscillating control points (possibly tending to infinity), which may lead to numerical issues. The situation is better for the Charrot–Gregory patch, where reliable control nets can be generated for  $n \leq 6$ . The control grid of Kato’s patch, and the above surfaces with more than 6 sides, can hardly be used when converted to NURBS form.

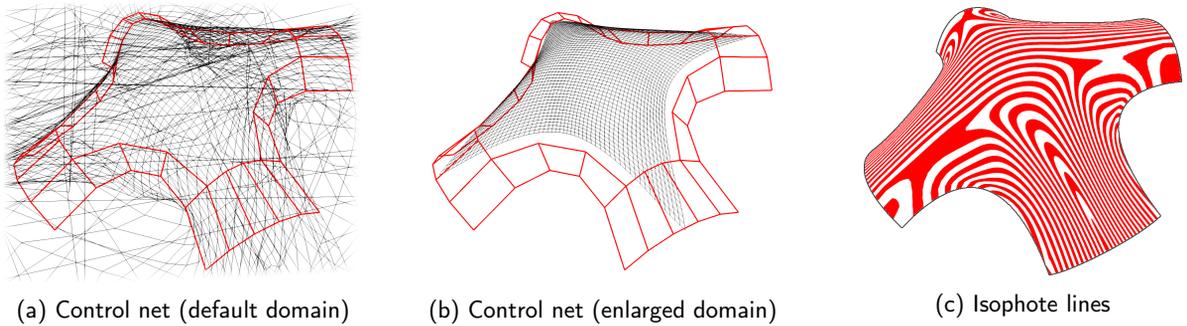
We present a solution to this problem. Normally the multi-sided domain is inside the unit square, so that the trimming curves will be inside the surface, but if we lift this constraint, we can create a larger multi-sided domain, thereby separating the unit square from the singularities. This means that the actual “trimmed” region will be outside the standard  $[0, 1] \times [0, 1]$  domain; this may not be supported by some applications, but the control structure will be close to the surface. An example with an 8-sided patch is presented in Figure 6b; the green rectangle in Figure 5b shows the unit square relative to the enlarged domain.

Rotating the domain, instead of scaling it, may also have a beneficial effect on the control grid. However, in our experience, rotation of the domain in itself is not sufficient to deal with 7- or 8-sided surfaces. Figure 7 shows several rotations of a 5-sided domain. We can define the optimal rotation as one that minimizes the functional

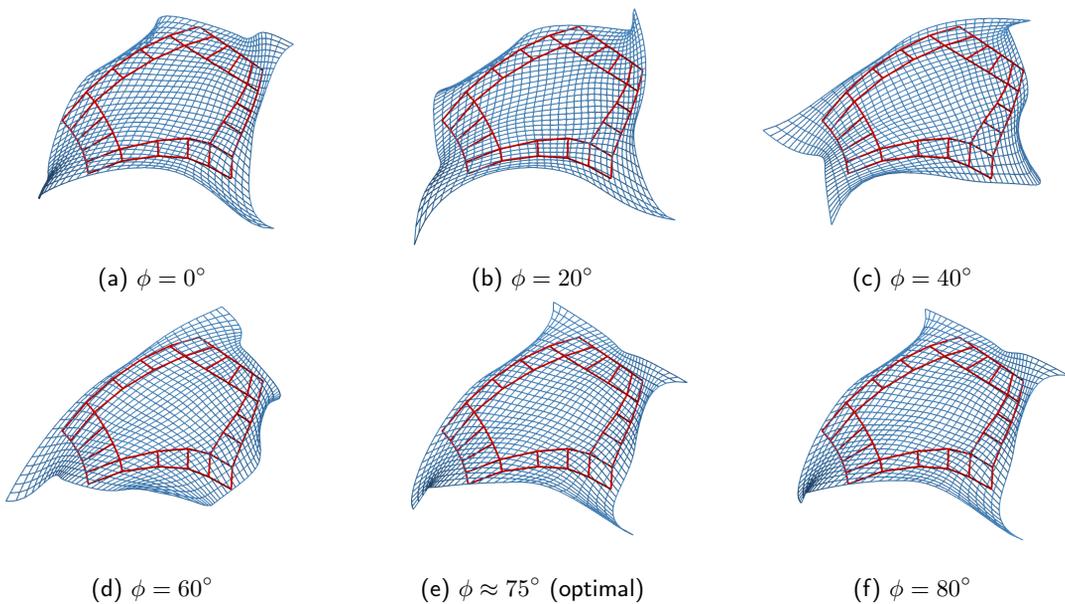
$$E = \sum_{i=0}^{\delta} \sum_{j=1}^{\delta-1} \left\| \frac{Q_{i,j-1} + Q_{i,j+1}}{2} - Q_{i,j} \right\|^2 + \sum_{i=1}^{\delta-1} \sum_{j=0}^{\delta} \left\| \frac{Q_{i-1,j} + Q_{i+1,j}}{2} - Q_{i,j} \right\|^2, \quad (21)$$

where  $Q_{i,j}$  are the control points of the bi-degree  $\delta$  tensor product surface, essentially selecting the smoothest control structure.

Table 1 summarizes the degrees of all four representations. It can be seen that these patches have relatively high degrees, in particular when the number of sides and the degree of the boundaries are raised. While Warren’s patch outperforms the others in this respect, the use of base points limits its usability in CAD systems. Kato’s surface always has singularities, and its degree is fairly high, but it is the only construction



**Figure 6:** An 8-sided Charrot–Gregory patch.



**Figure 7:** The effect of rotating a 5-sided domain.

where  $G^2$  continuity can be easily achieved. We found that while the Charrot–Gregory patch has a slightly higher degree than the S-patch, it has much lower computational cost in its multi-sided form, and have no control net quality problems for the most frequent 5- and 6-sided configurations.

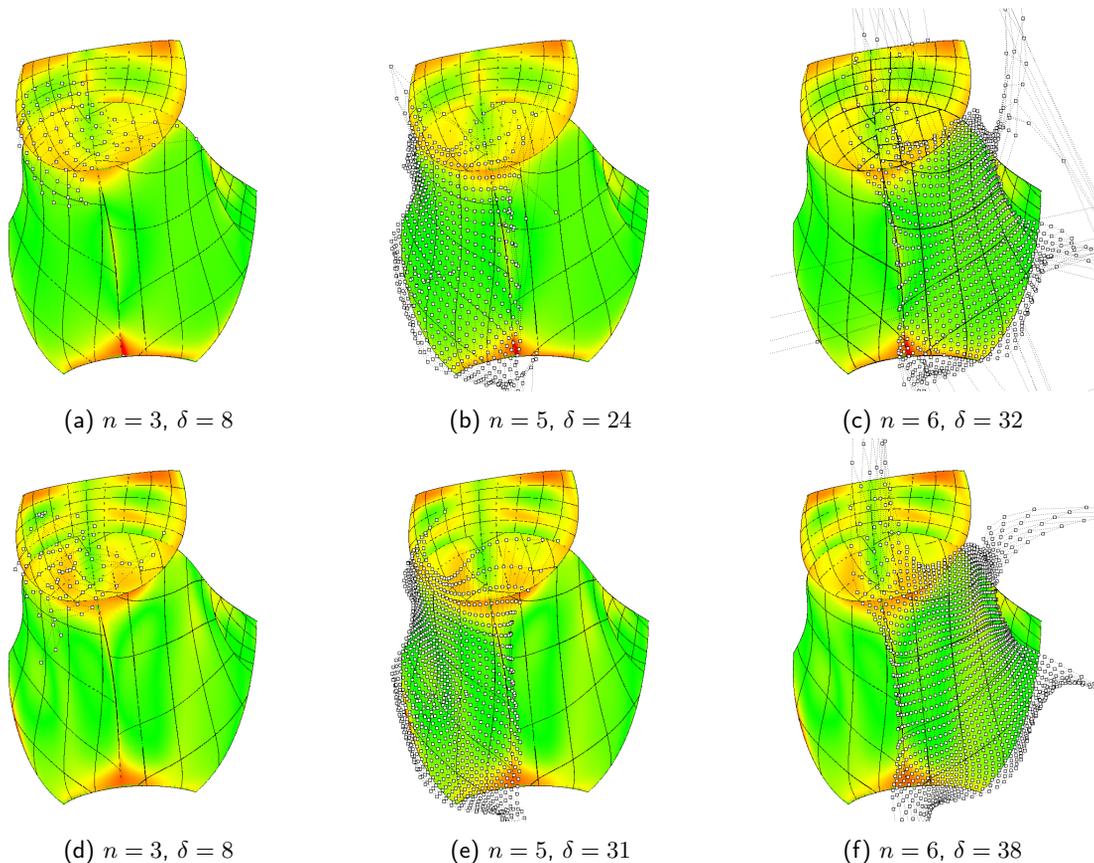
#### 4 TEST CASES

In this section we will show the rational Bézier conversion of more complex objects. Since the tensor product surface is exactly the same as the multi-sided original, there is little point in investigating the quality of the patches; we are going to concentrate on the control net quality instead. The domain-scaling method outlined above always generates nice control structures, so a comparison using the default domain is more informative. (We will show only S-patches and Charrot–Gregory patches, because the other two surfaces always have singularities.)

$n$	S-patch [6]	Warren [12]	Kato [5]	Charrot–Gregory [1]
3	$d[+3]$ (both Bézier triangles)		$3d + 5$	$d + 3$
5	$3d[+9]$	$\approx 3d$	$5d + 9$	$5d + 6$
6	$4d[+12]$	$3d$	$6d + 11$	$6d + 8$
7+	$(n - 2)(d[+3])$	N/A	$nd + 2n - 1$	$nd + 2n - 4$

**Table 1:** Rational polynomial degrees of the converted surfaces for different number of sides, assuming boundary curves of degree  $d$ . Gray cells indicate that the surface is susceptible to the singularity issue. For S-patches, the number in brackets is applied when the surface is generated by a degree- $d$   $G^1$  frame.

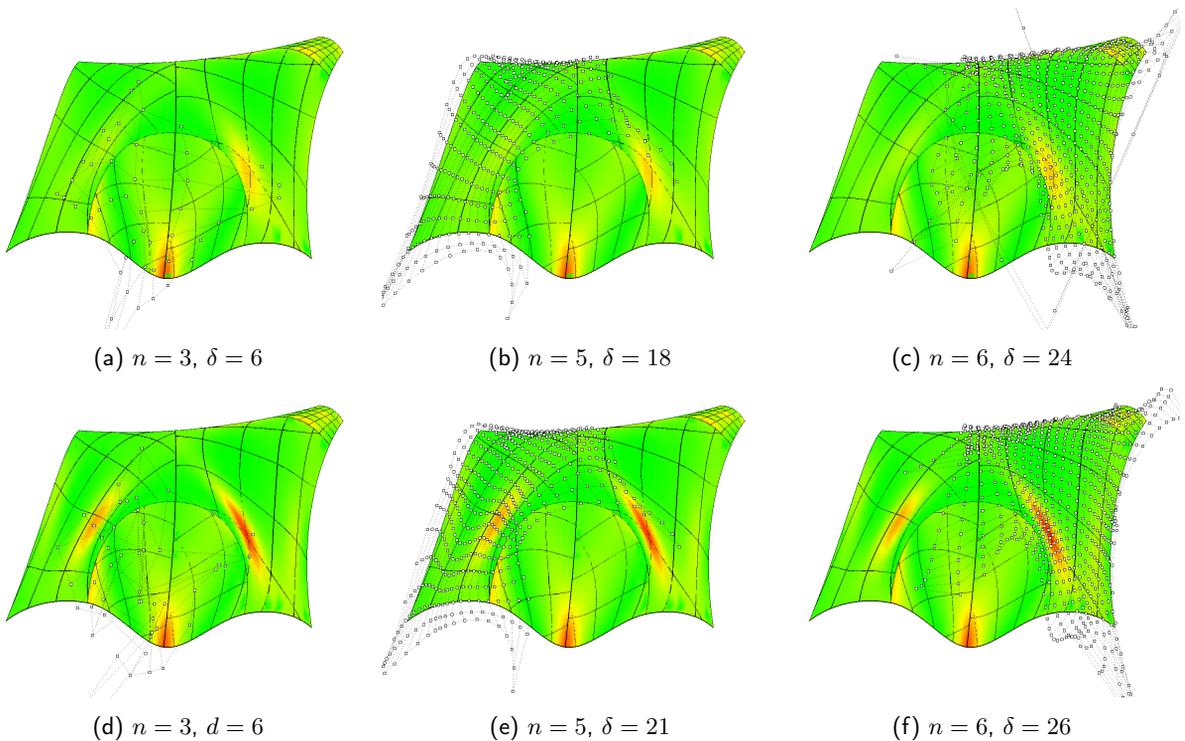
All images in this section were created by a commercially available CAD system, Rhinoceros 3D [8]; the colors show mean curvature distribution, patch boundaries and isocurves are drawn with black lines.



**Figure 8:** Control networks of S-patches (top) and Charrot–Gregory surfaces (bottom).

The model in Figure 8, defined by quintic  $G^1$  frames, has three 3-sided, two 4-sided, one 5-sided and one 6-sided surface. The triangular S-patch shown in Fig. 8a has a more regular control net than its counterpart in Fig. 8d, but the 5-sided patch in Fig. 8b shows some outliers compared to Fig. 8e, and the 6-sided S-patch in

Fig. 8c has some control points close to infinity, causing numerical evaluation errors in the bottom-left corner, while the control net in Fig. 8f, deviating to some extent, is still manageable.



**Figure 9:** “Pocket” model (top: S-patches, bottom: Charrot–Gregory patches).

We find similar results in the “Pocket” model (Figure 9), where the surfaces were defined by cubic  $G^1$  frames, and are of a lower degree. In this case the irregularities in the control net of the 6-sided S-patch did not lead to any evaluation problems.

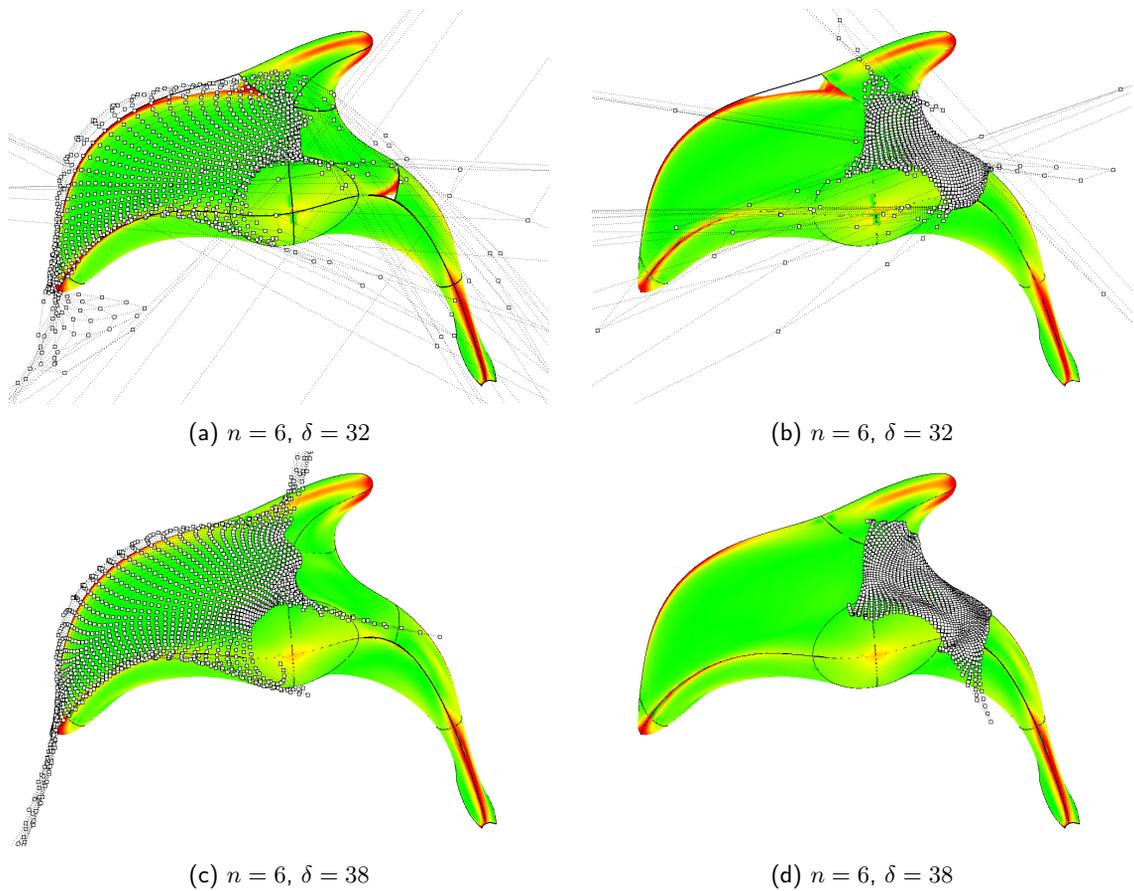
Finally, in Figure 10, a dolphin model is shown, defined by quintic  $G^1$  frames. The two 6-sided S-patches are not evaluated correctly here due to the erratic control points. One of the Charrot–Gregory patches also stretches far near the corners, but is still within the limits of exact evaluation.

## 5 CONCLUSIONS

We have described and analyzed four genuine multi-sided surface representations that can be converted into standard, tensor product (rational Bézier) form. The conversion generally leads to high-degree patches, and the control points of the resulting surfaces may oscillate, due to singularities, to an extent that prevents practical usability in industrial CAD/CAM systems.

We have investigated the occurrence of singularities for these schemes, and introduced a technique to avoid nonsense control structures. Using the proposed exact conversion procedures one can build watertight and smoothly connected models with trimmed surface patches.

All four schemes presented in this paper have their deficiencies; however, our practical experience would assign some advantage to Charrot–Gregory patches. We are going to continue searching for other patch representations that possess both multi-sided shape control and low-degree tensor product form.



**Figure 10:** Dolphin model (top: S-patches, bottom: Charrot-Gregory patches).

## ACKNOWLEDGEMENTS

This project has been supported by the Hungarian Scientific Research Fund (OTKA, No. 124727), and the National Research, Development and Innovation Fund (TUDFO/51757/2019-ITM, Thematic Excellence Program).

## ORCID

Péter Salvi <http://orcid.org/0000-0003-2456-2051>

Tamás Várady <http://orcid.org/0000-0001-9547-6498>

## A CONVERSION EQUATIONS

The following appendices show the patch equations with the numerator and denominator separated, so the tensor product conversion can be done according to Eqs. (8–10).

### Kato's Patch

The blending functions have the same denominators, but the side parameters do not. Introducing the notation  $\mathcal{L}_i = L_{i-1} + L_{i+1}$ , and omitting the parameters for brevity, the patch equation becomes

$$S_{\text{Kato}} = \frac{1}{\sum_{i=1}^n H_i^{e+1}} \cdot \sum_{i=1}^n \sum_{j=0}^d \sum_{k=0}^e P_{j,k}^i \cdot \binom{d}{j} \frac{L_{i-1}^j L_{i+1}^{d-j}}{\mathcal{L}_i^j \mathcal{L}_i^{d-j}} \cdot \binom{e}{k} L_i^k (1 - L_i)^{e-k} \cdot H_i^{e+1}. \quad (22)$$

Dividing and multiplying the equation with  $\prod_{i=1}^n \mathcal{L}_i^d$ , we arrive at the form

$$S_{\text{Kato}} = \frac{1}{\sum_{i=1}^n H_i^{e+1}} \cdot \frac{1}{\prod_{i=1}^n \mathcal{L}_i^d} \cdot \sum_{i=1}^n \sum_{j=0}^d \sum_{k=0}^e P_{j,k}^i \cdot \left[ \binom{d}{j} \cdot L_{i-1}^j L_{i+1}^{d-j} \prod_{\substack{r=1 \\ r \neq i}}^n \mathcal{L}_r^d \right] \cdot \binom{e}{k} L_i^k (1 - L_i)^{e-k} \cdot H_i^{e+1}. \quad (23)$$

### Charrot–Gregory Patch

Similarly to Kato's patch, straightforward calculation leads to the form

$$S_{\text{CG}} = \frac{1}{\sum_{i=1}^n H_{i-1,i}^2} \cdot \frac{1}{\prod_{i=1}^n \mathcal{L}_i^d} \cdot \sum_{i=1}^n \left[ \hat{R}_{i-1} + \hat{R}_i - \hat{Q}_{i-1,i} \right] \cdot H_{i-1,i}^2 \cdot \prod_{\substack{r=1 \\ r \neq i-1,i}}^n \mathcal{L}_r^d, \quad (24)$$

where

$$\hat{R}_{i-1} = \sum_{j=0}^d \left[ P_{j0}^{i-1} \mathcal{L}_i + L_{i-1} d(P_{j1}^{i-1} - P_{j0}^{i-1}) \right] \cdot \binom{d}{j} L_{i-2}^j L_i^{d-j} \mathcal{L}_i^{d-1}, \quad (25)$$

$$\hat{R}_i = \sum_{j=0}^d \left[ P_{j0}^i \mathcal{L}_{i-1} + L_i d(P_{j1}^i - P_{j0}^i) \right] \cdot \binom{d}{j} L_{i-1}^j L_{i+1}^{d-j} \mathcal{L}_{i-1}^{d-1}, \quad (26)$$

$$\hat{Q}_{i-1,i} = \left[ P_{00}^i \mathcal{L}_{i-1} \mathcal{L}_i + L_{i-1} d(P_{10}^i - P_{00}^i) \mathcal{L}_{i-1} + L_i d(P_{01}^i - P_{00}^i) \mathcal{L}_i + L_{i-1} L_i d^2 (P_{11}^i - P_{10}^i - P_{01}^i + P_{00}^i) \right] \cdot \mathcal{L}_{i-1}^{d-1} \mathcal{L}_i^{d-1}. \quad (27)$$

## B TRIANGULAR CHARROT–GREGORY PATCH

Substituting  $h_{i-1}$  and  $h_i$  for  $s_i$  and  $1 - s_{i-1}$ , respectively, the patch equation becomes

$$S_{\text{CG}\Delta} = \frac{1}{\sum_{i=1}^3 H_{i-1,i}^2} \cdot \sum_{i=1}^3 \left[ R_{i-1}(1 - h_i, h_{i-1}) + R_i(h_{i-1}, h_i) - Q_{i-1,i}(1 - h_i, h_{i-1}) \right] \cdot H_{i-1,i}^2, \quad (28)$$

which is directly convertible to tensor product form. While this is of a much lower degree than the original parameterization, it cannot be used for  $n > 3$ , because then  $h_i$  may take on values larger than 1, and thus the ribbons would be evaluated outside the  $[0, 1]$  interval, which can damage surface quality.

## REFERENCES

- [1] Charrot, P.; Gregory, J.A.: A pentagonal surface patch for computer aided geometric design. *Computer Aided Geometric Design*, 1(1), 87–94, 1984. [http://doi.org/10.1016/0167-8396\(84\)90006-2](http://doi.org/10.1016/0167-8396(84)90006-2).
- [2] DeRose, T.D.: Composing Bézier simplexes. *ACM Transactions on Graphics*, 7(3), 198–221, 1988. <http://doi.org/10.1145/44479.44482>.
- [3] DeRose, T.D.; Goldman, R.N.; Hagen, H.; Mann, S.: Functional composition algorithms via blossoming. *ACM Transactions on Graphics*, 12(2), 113–135, 1993. <http://doi.org/10.1145/151280.151290>.
- [4] Floater, M.S.: Generalized barycentric coordinates and applications. *Acta Numerica*, 24, 161–214, 2015. <http://doi.org/10.1017/S0962492914000129>.
- [5] Kato, K.: Generation of  $n$ -sided surface patches with holes. *Computer-Aided Design*, 23(10), 676–683, 1991. [http://doi.org/10.1016/0010-4485\(91\)90020-W](http://doi.org/10.1016/0010-4485(91)90020-W).
- [6] Loop, C.T.; DeRose, T.D.: A multisided generalization of Bézier surfaces. *ACM Transactions on Graphics*, 8(3), 204–234, 1989. <http://doi.org/10.1145/77055.77059>.
- [7] Loop, C.T.; DeRose, T.D.: Generalized B-spline surfaces of arbitrary topology. In *17th Conference on Computer Graphics and Interactive Techniques, SIGGRAPH*, 347–356. ACM, 1990. ISBN 0-89791-344-2. <http://doi.org/10.1145/97879.97917>.
- [8] Robert McNeel & Associates: Rhinoceros 3D. <https://www.rhino3d.com>.
- [9] Salvi, P.:  $G^1$  hole filling with S-patches made easy. In *Proceedings of the 12th Conference of the Hungarian Association for Image Processing and Pattern Recognition*, 2019. <https://arxiv.org/abs/2002.11109>.
- [10] Salvi, P.: On the CAD-compatible conversion of S-patches. In *Proceedings of the Workshop on the Advances of Information Technology*, 72–76, 2019. <https://arxiv.org/abs/2002.11111>.
- [11] Salvi, P.; Várady, T.; Rockwood, A.: Ribbon-based transfinite surfaces. *Computer Aided Geometric Design*, 31(9), 613–630, 2014. <http://doi.org/10.1016/j.cagd.2014.06.006>.
- [12] Warren, J.: Creating multisided rational Bézier surfaces using base points. *ACM Transactions on Graphics*, 11(2), 127–139, 1992. <http://doi.org/10.1145/130826.130828>.