Sixth Hungarian Conference on Computer Graphics and Geometry, Budapest, 2012

New Schemes for Multi-sided Transfinite Surface Interpolation

Péter Salvi[†], Tamás Várady[†], Alyn Rockwood[‡]

[†] Budapest University of Technology and Economics
[‡] King Abdullah University of Science and Technology

Abstract

One of the major problems in CAGD is to create mathematical representations for complex free-form objects composed of several smoothly connected surfaces. An intuitive solution is curvenet-based design, where designers directly define feature curves of the object, which are interpolated by multi-sided surfaces. A convenient representation for these models is based on transfinite surface interpolation. This paper introduces two new schemes for transfinite surfaces: a true multi-sided generalization of the Coons patch, and another based on a natural combination of curved side interpolants. The two formulations are supported by new parameterizations that need to satisfy strict tangential constraints along the boundaries. A few examples comparing the two patches illustrate the results.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

1. Introduction

Curvenet-based design[9] is an intuitive approach to create complex free-form CAD models in three dimensions. The process begins with creating a curve network representing the edges and feature lines of the object. These curves may come from several sources, like traditional blueprints, 2D sketches, or directly by some 3D graphical user interface. Then surfaces are stretched over the curve loops interpolating these boundaries. *Transfinite interpolation surfaces* are particularly suitable for this purpose, since they are defined solely by the boundary curves and their cross-derivatives.

The Coons patch is a well-known example for four-sided (biparametric) configurations, widely used due to its natural curvature distribution. Multi-sided patches, however, cannot be defined by two parameters, and the known solutions — including the patches of Gregory[1] and Kato[4] — differ from the original Boolean sum logic proposed by Coons. These may also exhibit shape artifacts in certain situations with highly curved boundaries and uneven lengths.

In this paper we introduce two new surface representations that are “close” to Coons’ original idea. The first is a true generalization of the Coons patch, consisting of side interpolants and corner correction patches; the second is a surface combining curved side interpolants by a natural application of blend functions.

In Section 2 we will examine the Coons patch and rewrite it in an easily extensible formulation. Then in Section 3 we will review the most important traditional n-sided surface formulations. The generalized Coons patch will be presented in Section 4, along with two new parameterization methods. It is followed by the definition of the curved interpolant-based surface in Section 5, and some examples in Section 6.

This paper is a condensed version of a more-detailed journal paper[6], where all mathematical details and the related proofs are also presented. At the same time, we hope that the basic concept of these new representational forms can be understood based on the current paper, as well.

2. Coons patch reformulation

Let us assume that we have a four-sided surface patch S parameterized in the (u,v) plane ($u,v \in [0,1]$). Given positional side constraints $S(u,0)$, $S(u,1)$, $S(0,v)$, $S(1,v)$ and tangential (cross-derivative) constraints $S_v(u,0)$, $S_v(u,1)$, $S_u(0,v)$,

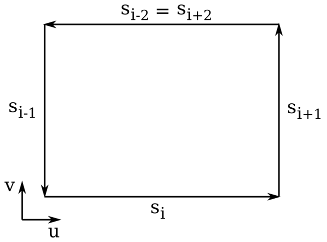


Figure 1: Domain of a Coons patch using side-based parameterization

$S_u(1,v)$, the patch equation suggested by Coons can be written as

$$
\begin{aligned}
U &= \begin{bmatrix} \alpha_0(u) & \beta_0(u) & \alpha_1(u) & \beta_1(u) \end{bmatrix}, \\
V &= \begin{bmatrix} \alpha_0(v) & \beta_0(v) & \alpha_1(v) & \beta_1(v) \end{bmatrix}, \\
S^u &= \begin{bmatrix} S(u,0) & S_v(u,0) & S(u,1) & S_v(u,1) \end{bmatrix}, \\
S^v &= \begin{bmatrix} S(0,v) & S_u(0,v) & S(1,v) & S_u(1,v) \end{bmatrix}, \\
S^{uv} &= \begin{bmatrix} S(0,0) & S_u(0,0) & S(1,0) & S_u(1,0) \\ S_v(0,0) & S_{uv}(0,0) & S_v(1,0) & S_{uv}(1,0) \\ S(0,1) & S_u(0,1) & S(1,1) & S_u(1,1) \\ S_v(0,1) & S_{uv}(0,1) & S_v(1,1) & S_{uv}(1,1) \end{bmatrix}, \\
S(u,v) &= V(S^u)^T + S^v U^T - V S^{uv} U^T.
\end{aligned}
$$

This is the well-known Boolean sum formulation[2], where α_0, α_1, β_0, β_1 are the cubic Hermite blending functions, needed to satisfy G^1 continuity:

$$
\begin{aligned}
\alpha_0(t) &= 2t^3 - 3t^2 + 1, & \alpha_1(t) &= -2t^3 + 3t^2, \\
\beta_0(t) &= t^3 - 2t^2 + t, & \beta_1(t) &= t^3 - t^2.
\end{aligned}
$$

In order to reformulate the above definition on a per-side basis, we will use cyclic indices (with 1 coming after 4), and introduce the so-called *side parameters,* $s_i = s_i(u,v)$, associated with the i-th side of the domain, taking the values u, v, $1-u$ and $1-v$, as appropriate (see Fig. 1). Then, denoting the positional and tangential constraints by $P_i(s_i)$ and $T_i(s_i)$, respectively, as well as using $W_{i,i-1}$ for the twist vector, the Coons patch can be formally rewritten as

$$
S(u,v) = \sum_{i=1}^{4} \begin{bmatrix} \alpha_0(s_{i+1}) \\ \beta_0(s_{i+1}) \end{bmatrix}^T \begin{bmatrix} P_i(s_i) \\ T_i(s_i) \end{bmatrix} -
$$
$$
\sum_{i=1}^{4} \begin{bmatrix} \alpha_0(s_{i+1}) \\ \beta_0(s_{i+1}) \end{bmatrix}^T \begin{bmatrix} P_i(0) & T_{i-1}(1) \\ T_i(0) & W_{i,i-1} \end{bmatrix} \begin{bmatrix} \alpha_0(s_i) \\ \beta_0(s_i) \end{bmatrix}.
$$

We assume that twist vector compatibility is ensured, i.e.,

$$
W_{i,i-1} := \frac{\partial}{\partial s_i} T_i(0) = -\frac{\partial}{\partial s_{i-1}} T_{i-1}(1).
$$

For contradicting boundary constraints, this can be resolved by Gregory's rational twists (see Gregory[3] or Farin[2]).

This formulation can be further simplified, if we replace the tangential blending with linear interpolants or *ribbons*:

$$
R_i(s_i,d_i) = P_i(s_i) + d_i T_i(s_i).
$$

Here we introduced another local parameter, the so-called *distance parameter* $d_i = d_i(u,v)$, that measures some kind of distance from the i-th side. In this four-sided case $d_i(u,v) = s_{i+1}(u,v)$ is a simple choice. The resulting patch becomes

$$
\begin{aligned}
S(u,v) &= \sum_{i=1}^{4} R_i(s_i,d_i)\alpha_0(d_i) \\
&- \sum_{i=1}^{4} Q_{i,i-1}(s_i,s_{i-1})\alpha_0(d_i)\alpha_0(s_i), \qquad (1)
\end{aligned}
$$

where $Q_{i,i-1}$ is a corner correction patch given by

$$
\begin{aligned}
Q_{i,i-1}(s_i,s_{i-1}) &= P_i(0) + (1-s_{i-1})T_i(0) + s_i T_{i-1}(1) \\
&+ s_i(1-s_{i-1})W_{i,i-1}.
\end{aligned}
$$

This will not be identical to the Coons patch, but it is a very similar construction that trivially satisfies all the boundary constraints. This formulation will be the basis of the generalization presented in Section 4.

3. Multi-sided patches

We can interpret the same problem for an arbitrary number of boundary curves. Given n three-dimensional curves $P_i(s_i)$, $1 \le i \le n$, and cross-derivative functions $T_i(s_i)$ along these curves, we want to construct an n-sided patch that interpolates these boundary curves and derivatives. For this, we will need a domain polygon in the parameter plane. The simplest choice is a regular polygon; for enhanced, non-regular domain alternatives, see Várady et al.[8]

We will also need a parameterization scheme that transforms the (u,v) coordinates of the parameter plane into local parameters of the individual boundary curves. Two possible constructions are presented in Section 4.3. For a more complete overview, see Salvi[5].

The rest of this section concentrates on how these curves and cross-derivative functions can be combined to create an interpolating patch, and what kind of blending functions can be used. As a reminder, Figure 2 presents a bird's-eye view of the steps needed to evaluate a transfinite surface.

Charrot and Gregory[1] proposed an n-sided patch based on *corner interpolants*, i.e., surfaces that interpolate two consecutive boundary curves, defined as

$$
\begin{aligned}
&R^{\text{corner}}_{i,i-1}(s_i,s_{i-1}) = \\
&P_i(s_i) + (1-s_{i-1})T_i(s_i) + P_{i-1}(s_{i-1}) + s_i T_{i-1}(s_{i-1}) - \\
&P_i(0) - (1-s_{i-1})T_i(0) - s_i T_{i-1}(1) - (1-s_{i-1})s_i W_{i,i-1}.
\end{aligned}
$$

It is easy to see that these surfaces interpolate the i-th and $(i-1)$-th sides, and their cross-derivatives. A similar technique will be introduced in Section 5 for interpolating three

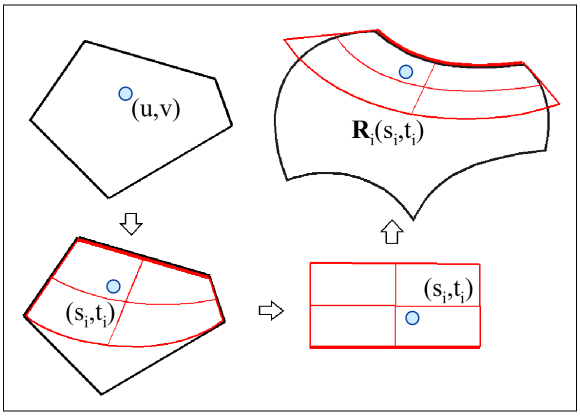


Figure 2: Evaluation of a transfinite surface: (i) point in the polygonal domain (ii) mapping into the domain of each biparametric ribbon (iii) evaluating each ribbon (iv) blending the evaluated points

sides. The Gregory patch is constructed in the following way:

$$S(u,v) = \sum_{i=1}^{n} R^{\text{corner}}_{i,i-1}(s_i(u,v),s_{i-1}(u,v)) \cdot B_{i,i-1}(d_1(u,v),\dots,d_n(u,v)).$$

Note, that we continue to use cyclic indices (e.g. $i-1$ for $i=1$ is n, and conversely, $i+1$ for $i=n$ is 1). Here a more complex blending function needs to be used, that is a rational polynomial function of the distance terms d_i:

$$B_{i,i-1}(d_1,\dots,d_n) = \frac{\prod_{k\notin\{i,i-1\}} d_k^2}{\sum_l \prod_{k\notin\{l,l-1\}} d_k^2} = \frac{D^2_{i,i-1}}{\sum_l D^2_{l,l-1}},$$

with the notation $D^e_{i_1\dots i_k} = \prod_{j\notin\{i_1\dots i_k\}} d_j^e$. This blending function will be used in Sec. 4.2 for the generalized Coons patch. The d_i parameters in an n-sided polygon are generally not equal to s_{i+1} (as we have seen for rectangles in the previous section), but can be independent "distance" parameters. For details, see Section 4.3.

Kato[4] also used the linear side interpolants R_i introduced above to define a patch as

$$S(u,v) = \sum_{i=1}^{n} R_i(s_i(u,v),d_i(u,v)) \cdot B_i^{\text{side}}(d_1(u,v),\dots,d_n(u,v)).$$

There are several variations on the blending function, one popular choice[8] being

$$B_i^{\text{side}}(d_1,\dots,d_n) = \frac{1/d_i^2}{\sum_j 1/d_j^2} = \frac{D_i^2}{\sum_j D_j^2}.$$

Note, that the blending function is singular at the corners; fortunately, the corner points are uniquely defined by the boundary constraints.

4. Generalized Coons patch

The next few sections deal with the generalization of Coons patches to n sides. The proposed interpolating surface uses the same principles as the original Coons patch, i.e., combines linear side interpolants and corner correction patches, and thus can be regarded as a natural generalization, even though the surface presented here will not revert to a Coons patch for quadrilateral configurations.

In Section 4.1 a new patch construction is introduced, and in Sections 4.2 and 4.3 we show suitable functions for blending and parameterization, respectively.

4.1. Coons Patch Generalization

With appropriate parameterization functions $s_i(u,v)$ and $d_i(u,v)$, Equation (1) can be generalized as

$$\begin{aligned} S(u,v) &= \sum_{i=1}^{n} R_i(s_i,d_i)\cdot B_i(d_1,\dots,d_n) \\ &- \sum_{i=1}^{n} Q_{i,i-1}(s_i,s_{i-1})\cdot B_{i,i-1}(d_1,\dots,d_n), \end{aligned}$$

where $B_i = B_{i,i-1} + B_{i+1,i}$ is a *side blend*, and $B_{i,i-1}$ is a *corner blend* function. $B_{i,i-1}$ is required to have the following properties:

$$B_{i,i-1}(d_1,\dots,d_j=0,\dots d_n) = 0,\ j\notin\{i-1,i\} \quad (2)$$

$$B_{i,i-1}(d_1,\dots,d_i=0,\dots,d_n) + B_{i+1,i}(d_1,\dots,d_i=0,\dots d_n) = 1, \quad (3)$$

$$\frac{\partial}{\partial d_j} B_{i,i-1}(d_1,\dots,d_j=0,\dots,d_n) = 0,\ \forall j. \quad (4)$$

Section 4.2 will show a possible construction. Equation (2) means that the blend function vanishes on all sides not connected to the corner. Boundary interpolation is satisfied due to Eq. (3), and finally Eq. (4) is needed for tangential interpolation.

Note that this patch, similarly to the Coons patch, combines five surfaces to evaluate a point on the boundary: three linear ribbons and two corner correction patches. This imposes several requirements for the (s_i,d_i) parameterization. First, for s_i we need

$$s_i \in [0,1], \quad (5)$$

and for a point on the i-th side, the parameterization has to satisfy that

$$d_i = 0, \quad (6)$$

$$s_{i-1} = 1, \qquad s_{i+1} = 0, \quad (7)$$

$$d_{i-1} = s_i, \qquad d_{i+1} = 1 - s_i, \quad (8)$$

$$\frac{\partial d_{i-1}}{\partial u} = \frac{\partial s_i}{\partial u}, \qquad \frac{\partial d_{i-1}}{\partial v} = \frac{\partial s_i}{\partial v}, \quad (9)$$

$$\frac{\partial d_{i+1}}{\partial u} = -\frac{\partial s_i}{\partial u}, \qquad \frac{\partial d_{i+1}}{\partial v} = -\frac{\partial s_i}{\partial v}. \quad (10)$$

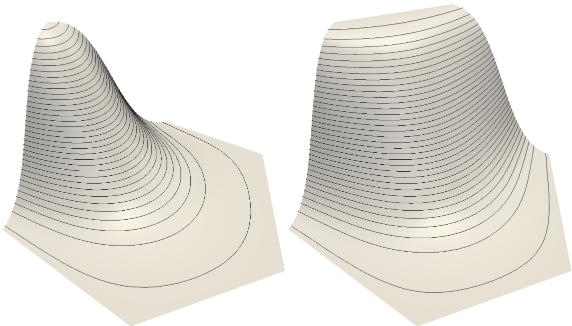


Figure 3: Blending functions $B_{i,i-1}$ and B_i over a six-sided domain.

The patches reviewed in Section 3 only require (5)–(7), so we need to satisfy more strict restrictions here, but in Section 4.3 two techniques will be presented for creating eligible parameterizations.

4.2. Blending functions

Recall the blending function $B_{i,i-1}$ described in Charrot and Gregory[1]:

$$B_{i,i-1}(d_1,\dots,d_n) = \frac{\prod_{k\notin\{i,i-1\}} d_k^2}{\sum_l \prod_{k\notin\{l,l-1\}} d_k^2} = \frac{D_{i,i-1}^2}{\sum_l D_{l,l-1}^2}.$$

Assumptions (2) and (3) are trivially true. A closer examination of the difference quotient leads to a straightforward proof of (4), thus this blend function adheres to all requirements. Fig. 3 shows a graphical representation of $B_{i,i-1}$, as well as the side blend $B_i = B_{i,i-1} + B_{i+1,i}$.

4.3. Parameterizations

In the following we will propose two parameterizations that satisfy all the requirements (5)–(10).

4.3.1. Interconnected Parameterizations

Take functions $s_i(u,v)$ that give 0 for every point on the $(i-1)$-th side and 1 for those on the $(i+1)$-th side; for all other points inside the convex domain, they return a value in $[0,1]$. For example, the s coordinates of the bilinear or radial line sweeps[1, 5] are such functions. These naturally satisfy (5) and (7). Define a blending function $\alpha(t) \in [0,1] \to [0,1]$ with $\alpha(0) = 1$ and $\alpha(1) = \alpha'(0) = \alpha'(1) = 0$. Examples are the Hermite function $\alpha_0(t)$ from Section 2, or a variation of the rational blend function presented in the previous section: $\alpha(t) = \frac{(1-t)^2}{t^2+(1-t)^2}$. Now we can define d_i by means of s_{i-1} and s_{i+1} as follows:

$$d_i(u,v) = (1-s_{i-1}(u,v))\cdot\alpha(s_i) + s_{i+1}(u,v)\cdot\alpha(1-s_i).$$

If we are on the i-th side, $s_{i-1} = 1$ and $s_{i+1} = 0$, so $d_i = 0$, satisfying (6).

Still on the i-th side, d_{i-1} and its derivative are the same as s_i:

$$\begin{aligned}
d_{i-1} &= (1-s_{i-2})\cdot\alpha(s_{i-1}) + s_i\cdot\alpha(1-s_{i-1}) = s_i,\\
\frac{\partial d_{i-1}}{\partial u} &= \frac{\partial}{\partial u}(1-s_{i-2})\cdot\alpha(s_{i-1}) + \frac{\partial}{\partial u}s_i\cdot\alpha(1-s_{i-1})\\
&= \frac{\partial s_i}{\partial u},
\end{aligned}$$

because the derivatives of the blend function vanish. The same reasoning works for the derivative by v. Similarly

$$\begin{aligned}
d_{i+1} &= (1-s_i)\cdot\alpha(s_{i+1}) + s_{i+2}\cdot\alpha(1-s_{i+1}) = 1-s_i,\\
\frac{\partial d_{i+1}}{\partial u} &= \frac{\partial}{\partial u}(1-s_i)\cdot\alpha(s_{i+1}) + \frac{\partial}{\partial u}s_{i+2}\cdot\alpha(1-s_{i+1})\\
&= \frac{\partial}{\partial u}(1-s_i) = -\frac{\partial s_i}{\partial u},
\end{aligned}$$

so the requirements (8), (9) and (10) are all satisfied.

Fig. 4 shows constant s and d lines for this parameterization (using the central line sweep parameterization[10] as a basis). The first image is based on the right side of the polygon; the second image is based on the small side at the top-right; and the third image is based on the top side. Note that all lines of the second image start the same way (in a differential sense) as their counterparts in the first and third images.

4.3.2. Biquadratic Parameterization

The patch defined in Section 4.1 is using side-based linear interpolants, but also corner-based correction patches. In the following, two parameterizations will be used, one for the ribbons (side-based parameterization), and one for the correction patches (corner-based parameterization). The idea is that a linear side interpolant should have the same parameterization near the side adjacent to its base as the correction patch of the corresponding corner, enabling the correction patch to cancel out the extra terms generated by the ribbons. This will be achieved by biquadratic maps similar to the overlap patch parameterization[7].

We place a three-by-three control net on the domain, and the resulting planar surface defines a parameterization (see Fig. 5). We need the inverse of this biquadratic map, which can be computed by numerical methods, such as the Newton–Raphson algorithm. Note that the control points on the left-hand side are the same in both the side-based and the corner-based variants, so the two parameterizations behave identically near the left boundary. In the following we will examine the construction of these biquadratic control nets.

Imagine a domain with its i-th side on the u axis. For the side-based case, the control points are defined as

$$\begin{bmatrix} p_i^l & p_i^m & p_i^r \\ p_{i-1}^m & c & p_{i+1}^m \\ p_{i-1}^l & p_i^o & p_{i+1}^r \end{bmatrix},$$

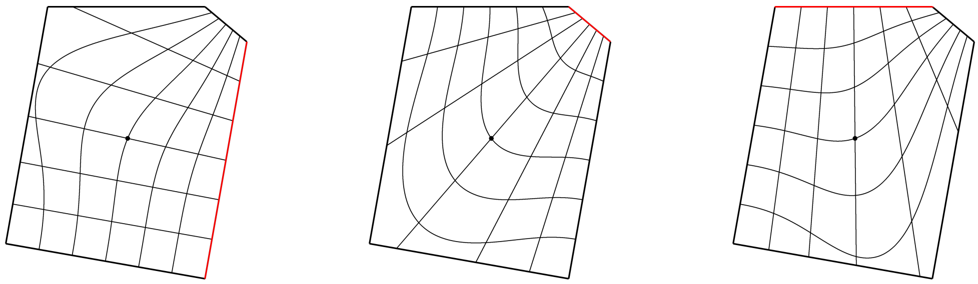
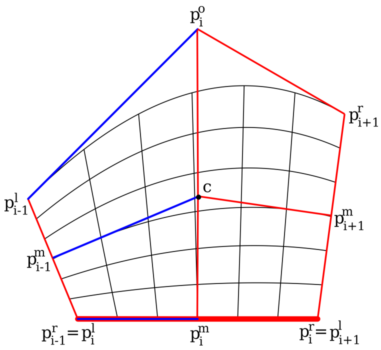
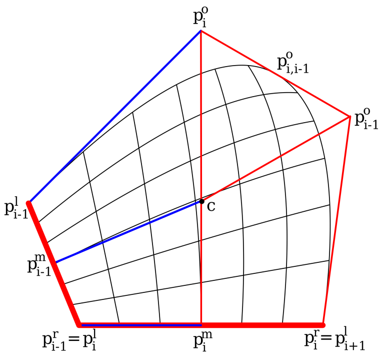


Figure 4: Constant parameter lines of the interconnected parameterization.



(a) Side-based construction



(b) Corner-based construction

Figure 5: Control net and constant parameter lines of the biquadratic parameterization.

where p_j^l, p_j^r and p_j^m are the left and right endpoints and the midpoint of the j-th side, respectively; c is the center of the domain and p_j^o is the domain vertex opposite to the j-th side (for even-sided polygons p_j^o is the midpoint of the opposite side).

When the basis is the corner at p_i^l, the control points are

$$\begin{bmatrix} p_i^l & p_i^m & p_i^r \\ p_{i-1}^m & c & p_{i-1}^o \\ p_{i-1}^l & p_i^o & p_{i,i-1}^o \end{bmatrix},$$

where $p_{i,i-1}^o$ is the vertex opposite to the vertex defined as the intersection of the i-th and $(i-1)$-th sides (for odd-sided polygons $p_{i,i-1}^o$ is the midpoint of the opposite side).

Note that the first two columns are the same as in the side-based case. Also the first two rows have their counterparts in the $(i-1)$-th side-based biquadratic as well. These relationships tell us that the parameterization based on the $i, i-1$ corner behaves the same way for a point on the $(i-1)$-th side as the parameterization based on the i-th side; and for a point on the i-th side as the parameterization based on the $(i-1)$-th side. This is important, because it means that all superfluous data that comes into the equations from the ribbon interpolation can be eliminated by corner-parameterized correction patches.

There is one special case: three-sided segments have a very intuitive, singular parameterization (depicted in Figure 6).

So in our transfinite surface scheme, side-based parameterizations are used for ribbons (R_i) and blends ($B_{i,i-1}$), but corner-based parametrizations are used for the correction patches ($Q_{i,i-1}$). We will use the convention of referring to corner-based parameters by (s_i^*, d_i^*), while retaining (s_i, d_i) for side-based ones.

The parametrization requirements (5)–(7) are trivially satisfied for side-based schemes. Linearity on the sides also guarantees (8), even between a side-based and a corner-based patch. The remaining constraints do not hold for either type, but are valid in-between: the d_{i-1} side-parameters are equal to the s_i^* corner-parameters in derivative sense (a corner-parameter with index i is associated with the biquadratic patch based on the i-th and $(i-1)$-th edges, using the i-th edge as constant $d_i^* = 0$). Similarly all requirements are satisfied in this way.

Using this new double parameterization system, we have to alter the original definition of the surface, substituting $Q_i(s_i^*, d_i^*) = Q_{i,i-1}(s_i^*, 1-d_i^*)$ for $Q_{i,i-1}(s_i, s_{i-1})$:

$$\begin{aligned} S(u,v) &= \sum_i R_i(s_i, d_i) \cdot B_i(d_1, \dots, d_n) \\ &\quad - \sum_i Q_i(s_i^*, d_i^*) \cdot B_{i,i-1}(d_1, \dots, d_n). \end{aligned}$$

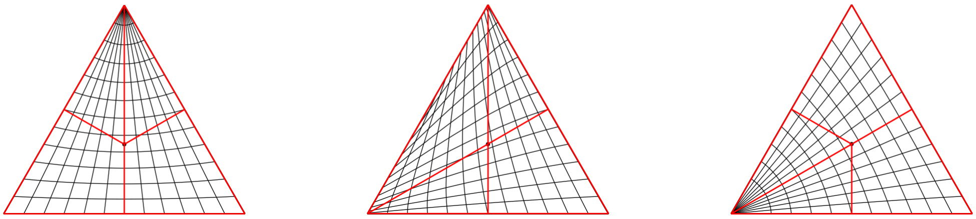


Figure 6: Control nets for triangular domains, based on the bottom side (left image), the right side (right image) and both (middle image).

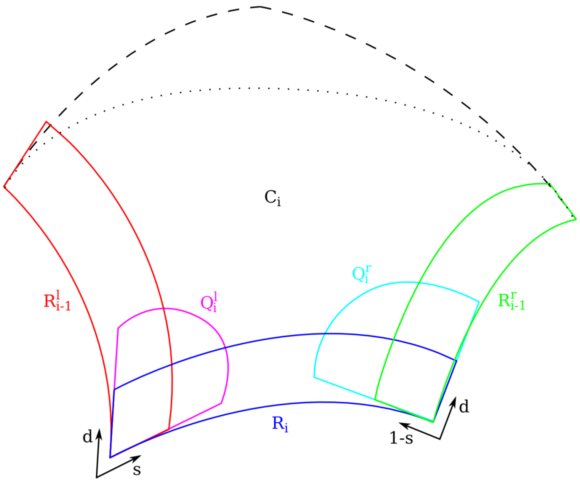


Figure 7: Constituents of a curved ribbon.

5. Curved ribbons

The patch defined in the previous section is based on linear ribbons. These have the advantage of being very simple and can be computed efficiently. On the other hand, highly curved surfaces may deviate far from their linear interpolants, thereby decreasing stability. Unintuitive bulges, coming from contradicting cross-derivatives, pose a common problem. Recall, however, that conventional side interpolant-based transfinite interpolation surfaces do not depend on the linearity of the ribbons. We can thus define curved ribbons that follow the shape of the surface more closely.

Let $C_i(s_i, d_i)$ denote the curved ribbon for the i-th side. In order to simplify the notation, we will drop the indices of s and d when talking about only a single curved ribbon, as it does not cause any ambiguity. The definition of C_i is as follows (see Fig. 7):

$$\begin{aligned} C_i(s,d) &= R_i^l(s,d)H(s) + R_i(s,d)H(d) + R_i^r(s,d)H(1-s) \\ &- Q_i^l(s,d)H(s)H(d) - Q_i^r(s,d)H(1-s)H(d), \end{aligned}$$

where

$$\begin{aligned} R_i^l(s,d) &= R_{i-1}(1-d,s) = P_{i-1}(1-d) + sT_{i-1}(1-d) \\ R_i^r(s,d) &= R_{i+1}(d,1-s) = P_{i+1}(d) + (1-s)T_{i+1}(d) \\ Q_i^l(s,d) &= Q_{i,i-1}(s,1-d) \\ &= P_i(0) + sT_{i-1}(1) + dT_i(0) + sdW_{i,i-1} \\ Q_i^r(s,d) &= Q_{i+1,i}(d,s) \\ &= P_{i+1}(0) + dT_i(1) + (1-s)T_{i+1}(0) + \\ & \quad d(1-s)W_{i+1,i} \end{aligned}$$

and $H(t)$ is a blend function, for example the Hermite blend function $\alpha_0(t)$.

It is interesting to note that the curved corner interpolants of the Gregory patch[1] can be defined in terms of ribbons and correction patches, as well:

$$\begin{aligned} R_{i,i-1}^{\text{Gregory}}(s_i, s_{i-1}) &= R_{i-1}(s_{i-1}, s_i) + R_i(s_i, 1-s_{i-1}) \\ &- Q_{i,i-1}(s_i, s_{i-1}). \end{aligned}$$

The curved ribbon defined above interpolates three consecutive boundary curves using blends, which makes it essentially a three-sided Coons patch. These ribbons can be used as side interpolants for conventional transfinite interpolation patches, but there is also a more natural blending scheme, as proposed in the next section.

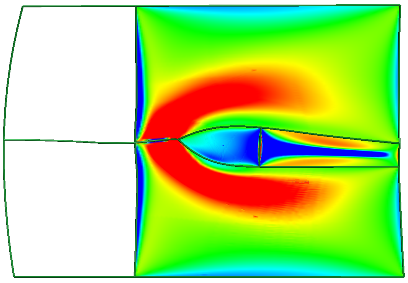
5.1. Composite ribbon patch

The generalization of Coons patches, as introduced in Section 4, does not work with curved side interpolants. However, we propose a new representation that combines curved ribbons in a different way, and renders the elimination of correction patches possible, yielding a much simpler formula:

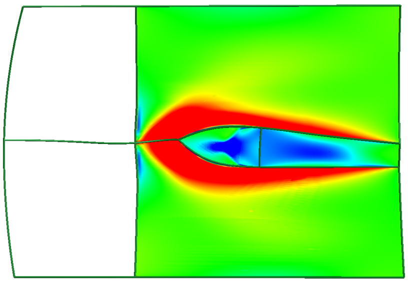
$$S(u,v) = \frac{1}{2}\sum_{i=1}^{n} C_i(s_i(u,v), d_i(u,v))B_i(d_1(u,v), \dots, d_n(u,v)),$$

where the parameterization satisfies the requirements listed in (5)–(10).

According to the characteristics of the B_i blend function, for any point on the i-th boundary all addends of the sum vanish except for C_{i-1}, C_i and C_{i+1}. Since each of these ribbons also interpolates the adjacent curves, the three ribbon



(a) Conventional linear interpolant-based patch



(b) Composite ribbon patch

Figure 8: Mean map comparison of a model with four surfaces.

points are the same. Their cumulative blend is

$$\begin{aligned} B_{i-1} + B_i + B_{i+1} &= B_{i,i-1} + B_i + B_{i+1,i} \\ &= (B_{i,i-1} + B_{i+1,i}) + B_i \\ &= 1 + 1 = 2, \end{aligned}$$

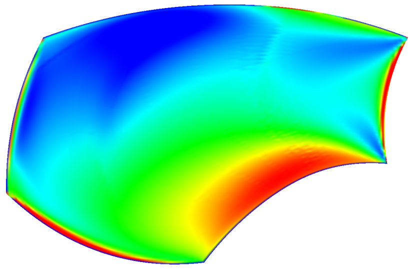
hence the division by two in the surface equation.

It can be proven that the parameterization constraints can be loosened. If we leave out the last two requirements (9 and 10), the patch remains valid, i.e., it will have the same tangent plane for every boundary point as the respective Coons ribbon, but it won't have the exact same tangent vector. This enables the use of many other parameterizations, such as the bilinear or the central line sweep parameterizations.

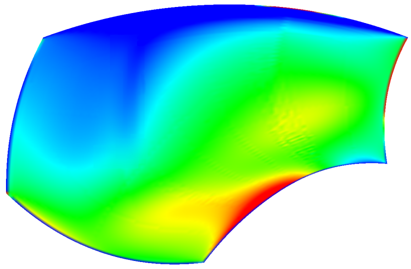
In other words, we have created a new transfinite surface representation that (i) has the same computational complexity as other conventional methods, (ii) uses curved side interpolants, and (iii) employs non-singular blend functions.

6. Examples

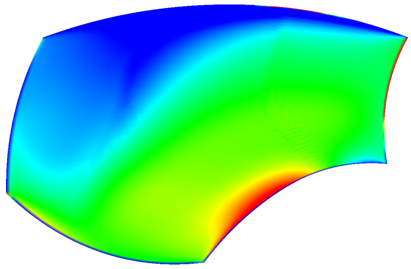
The first example in Fig. 8 shows the mean map of a model containing three-, four- and five-sided surfaces. The first image was created by a conventional linear interpolant-based patch. Note the blue areas near the sides — these are artifacts of the blending function, which abruptly drops as we move away from the boundaries. The composite ribbon patch below is free of these unwanted curvature changes.



(a) Conventional linear ribbon patch



(b) Generalized Coons patch

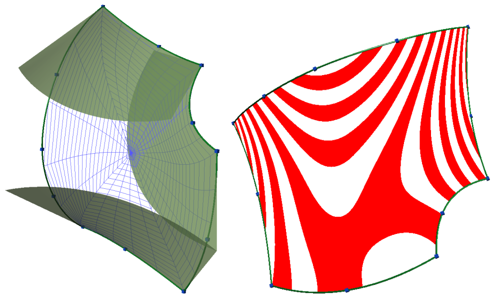


(c) Composite ribbon patch

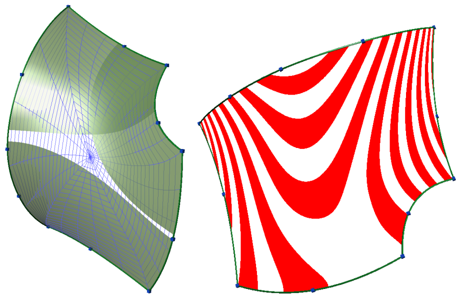
Figure 9: Mean map comparison of a surface using different patch types.

In Fig. 9 three schemes are contrasted using a single surface. The low-curvature areas at the right-hand corners are smoothed out in both of the new patches. The generalized Coons patch and the composite ribbon patch are very similar, but the latter has even more smooth curvature transitions.

In order to understand the essential difference between the two new patches, we should examine their ribbons. Figure 10 depicts them over the constant parameter lines of the patches. In the linear ribbon case, the interpolants start to deviate from the surface at a very "early phase", thus points in the interior are actually computed as the affine combination of relatively distant positions. On the other hand, the curved interpolants on the right-hand side image are going close to the predictable surface position, and in this way, points in the interior are combinations of ribbon points that are relatively close, keeping curvature variation at a low level. The slicing maps show that both surfaces have sufficiently good quality.



(a) Generalized Coons patch



(b) Composite ribbon patch

Figure 10: Ribbons and slicing map of a five-sided boundary configuration.

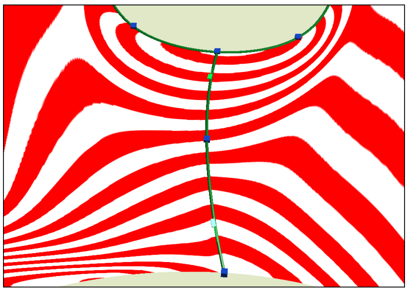


Figure 11: A G^1 model with two composite ribbon patches, using isophote line visualization.

Finally, Fig. 11 shows isophote lines on two composite ribbon patches connected by a smooth edge. The ends of the lines match along the boundary, confirming G^1 continuity, and in most points they are also unbroken, exhibiting approximately G^2 behavior.

Conclusion

Two new surface representations for transfinite surface interpolation have been presented. The first scheme can be considered as the true generalization of the Coons patch, using classical Boolean sum logic. The second is a transfinite surface combining curved side-interpolants. In order to satisfy the boundary constraints (positions, cross-derivatives), special blend functions and parameterizations have been applied. Future research work is going to be directed towards G^2 ribbons and using non-convex parametric domains.

Acknowledgements

This work was partially supported by the scientific program “Development of quality-oriented and harmonized R+D+I strategy and functional model at the Budapest University of Technology and Economics” (UMFT-TÁMOP-4.2.1/B-09/1/KMR-2010-0002) and a grant by the Hungarian Scientific Research Fund (No. 101845). The pictures in this paper were generated by the Sketches system developed by ShapEx Ltd, Budapest; the contribution of György Karikó is highly appreciated. The authors also acknowledge support from the Geometric Modeling and Scientific Visualization Research Center of KAUST, Saudi-Arabia.

References

1. P. Charrot, J. A. Gregory, A pentagonal surface patch for computer aided geometric design, Computer Aided Geometric Design 1 (1) (1984) 87–94.
2. G. Farin, Curves and surfaces for CAGD: a practical guide, 5th ed., Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.
3. J. A. Gregory, Smooth interpolation without twist constraints, in: R. E. Barnhill, R. F. Riesenfeld (eds.), Computer Aided Geometric Design, Academic Press, Inc., 1974, pp. 71–88.
4. K. Kato, n-sided surface generation from arbitrary boundary edges, in: P.-J. Laurent, P. Sablonnière, L. L. Schumaker (eds.), Curve and Surface Design: Saint-Malo 1999, Innovations in Applied Mathematics, Vanderbilt University Press, Nashville, TN, 2000, pp. 173–181.
5. P. Salvi, Fair curves and surfaces, Ph.D. thesis, Eötvös Loránd University, Budapest (2012).
6. P. Salvi, T. Várady, A. Rockwood, New multi-sided transfinite surfaces, Computer Aided Geometric Design (submitted).
7. T. Várady, Overlap patches: a new scheme for interpolating curve networks with n-sided regions, Computer Aided Geometric Design 8 (1) (1991) 7–27.
8. T. Várady, A. Rockwood, P. Salvi, Transfinite surface interpolation over irregular n-sided domains, Computer Aided Design 43 (2011) 1330–1340.
9. T. Várady, P. Salvi, A. Rockwood, 3d shape design using curve networks with ribbons, in: this proceedings.
10. T. Várady, P. Salvi, A. Rockwood, Transfinite surface interpolation with interior control, Graphical Models (under revision).