# Editing the interior of arbitrary surfaces using $C^\infty$ displacement blends

Péter Salvi

Budapest University of Technology and Economics

**Abstract**
*In curve network based design, surface patches are created automatically, based on positional and derivative constraints at their boundaries. This is often achieved by* transfinite interpolation *schemes, which have little or no control over the surface interior. In this paper we show a general framework for patch modification, which preserves the continuity of the surface both internally, and at the connections to other patches. The proposed method is applicable to any kind of surface – parametric, implicit or subdivision –, and gives the user flexible editing capabilities over the modification area.*

## 1. Introduction

*Transfinite surface interpolation*[11] is a very convenient tool for smooth hole-filling applications, and is the cornerstone of curve network based design—a paradigm where the designer draws feature curves, whose loops are automatically filled in with smooth patches by the modeling system.

A drawback of these surfaces, however, is that they lack interior control. Surface fullness is rarely adjustable, and for more detailed modifications, the user needs to insert additional curves, thereby (i) making the design more complex, (ii) increasing the number of patches, and (iii) harming overall surface quality, since continuity is normally only $G^1$ or $G^2$ between (sub)patches.

Control of the interior poses a problem sometimes even for control point based surfaces. The S-patch,[6] for example, has too many control points for interactive design, and while these can be arranged automatically,[8] it is not evident how they can be used to modify the patch inside. Our previous experience[10, 13] shows, that concave or highly curved configurations are also notoriously difficult to handle.

Here we propose a general method for patch modification. The idea is very simple: we want to allow the user to pick any point on the surface, and move it in a way that preserves continuity, and has a controllable region of influence. This will be accomplished by adding a *displacement surface* to the original patch, which vanishes at the boundaries. (Note that the designer will be able to move one point at a time, and subsequent edits modify the position of previously moved points.)

After reviewing some alternatives in Section 2, we go through the details of the construction in Section 3. A few examples and remarks on future work conclude the paper.

## 2. Previous work

Some specific transfinite interpolation surfaces do have interior control. Kato's patch,[4] for example, can easily be extended to interpolate *auxiliary objects*, such as internal points or curves, with adjustable range.[12] Another example is the *midpoint* patch,[9] which has a single central control point for fullness adjustments.

In the context of splines and subdivision surfaces, Kosinka et al.[5] propose an approach similar to ours, where *control vectors* correspond to what we will call *displacement*s .

We are not aware of any *representation-independent* method, however, for controlling the surface interior. The closest relatives are the mesh sculpting methods,[1, 2] where adaptive refinement is combined with decay functions to a similar effect.

## 3. The displacement surface

Let us assume that the initial surface is discretized as a triangular mesh with vertices $\mathbf{v}_i$, and we have an index set for the boundary vertices, $I_b$, and another for the displacements,

$I_d$ (assuming $I_b \cap I_d = \emptyset$). Finally, for each displacement index $j \in I_d$, we have a displacement vector $\mathbf{d}_j$.† The modified surface is then described by the displaced vertices

$$\hat{\mathbf{v}}_i = \mathbf{v}_i + \sum_{j \in I_d} \mathbf{d}_j B_j(i), \tag{1}$$

where $B_j(i) \in [0,1]$ is some kind of smooth blending function satisfying the following equations:

$$B_j(i) = \begin{cases} 0 & \text{for } i \in I_b, \\ 1 & \text{for } i = j. \end{cases} \tag{2}$$

In addition, all derivatives of $B_j$ should vanish at the boundary.

In the rest of this section, we will build up the blending function part by part.

### 3.1. Vanishing at the boundaries

First we need a function that satisfies Eq. (2), i.e., it gives 1 at the *footpoint* ($j$), and vanishes at the boundary. A simple solution is the harmonic function $H_j$ with these boundary constraints. It will only be $C^0$ at the footpoint, and will not satisfy the derivative constraints, but we will deal with those later.

This function can be computed by using a discrete Laplace operator,[7] solving

$$\mathbf{L} \cdot \mathbf{h} = \mathbf{0}, \tag{3}$$

subject to $h_i = 0$ for $i \in I_b$ and $h_j = 1$, which can be fixed using Lagrange multipliers. Here $\mathbf{L}$ is the symmetric matrix

$$L_{kl} = \begin{cases} -\sum_{i \neq k} L_{ki} & \text{when } k = l, \\ -\frac{1}{2}(\cot\alpha_{kl} + \cot\beta_{kl}) & \text{when } \mathbf{v}_k \text{ is adjacent to } \mathbf{v}_l, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

The angles $\alpha_{kl}$ and $\beta_{kl}$ are those opposite to the edge between $\mathbf{v}_k$ and $\mathbf{v}_l$ (or zero, at the boundary, where the angle does not exist). $H_j(i)$ is then defined simply as $h_i$.

### 3.2. $C^\infty$ continuity

The above function is not smooth at the footpoint, and its derivatives do not vanish at the boundary. These problems can be solved by composing it with a $C^\infty$ blending function—a monotonic function in $[0,1]$ that satisfies

$$f(0) = 0, \qquad f(1) = 1, \qquad f^{(k)}(0) = f^{(k)}(1) = 0 \tag{5}$$

for all $k > 0$.

† It is important to note, that the above is written in a discretized setting only for the sake of generality, and it could also be equivalently reformulated using parameters (for parametric surfaces), by associating displacements with fixed parameter values. In either way, the construction itself remains continuous.

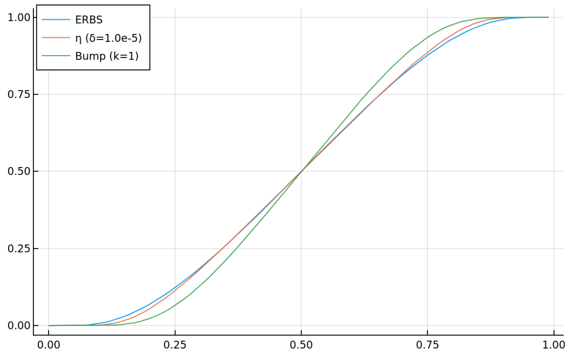


**Figure 1:** *Comparison of $C^\infty$ blending functions.*

There are a number of such functions, for example the *bump function*

$$f_{\text{Bump}}(t) = \frac{\psi(t)}{\psi(t) + \psi(1-t)}, \qquad \psi(t) = \exp\left(-\frac{1}{t^k}\right), \tag{6}$$

or the $\eta$ function[14]

$$f_\eta(t) = \begin{cases} 0 & 0 \le t \le \delta, \\ \frac{\phi(1-t)}{\phi(t)+\phi(1-t)} & \delta < t < 1-\delta, \\ 1 & 1-\delta \le t \le 1, \end{cases} \tag{7}$$

where

$$\phi(t) = \exp\left(2\exp\left(-\frac{1-2\delta}{t-\delta}\right) \cdot \frac{1-2\delta}{t+\delta-1}\right). \tag{8}$$

Here we use a simple *expo-rational B-spline*,[3] which is more computation-intensive, but has arguably better shape than the alternatives (see Fig. 1). It is defined as

$$f_{\text{ERBS}}(t) = \frac{\varphi(t)}{\varphi(1)}, \qquad \varphi(t) = \int_0^t \exp\left(-\frac{\left(s - \frac{1}{2}\right)^2}{s(1-s)}\right) ds. \tag{9}$$

### 3.3. Influence adjustment

Finally, we need to add some control over the range of the displacement. For this, we use the reparameterization

$$R_j(t) = 1 - (1-t)^{1/(1-r_j)}, \tag{10}$$

where $r_j \in [0,1)$ is the *range* of the displacement $\mathbf{d}_j$. When $r_j$ gets larger, an increasingly large neighborhood of the footpoint is getting high blend function values, see Fig. 2.

The final blend is defined as

$$B_j = f_{\text{ERBS}} \circ R_j \circ H_j. \tag{11}$$

Figure 3 shows the sum of two blending functions with different ranges over a 5-sided polygonal domain.

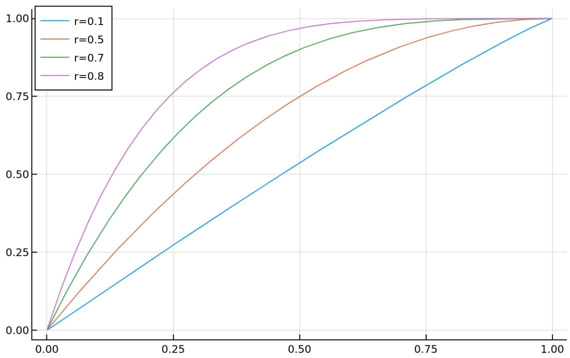


**Figure 2:** *Reparameterization with various ranges.*

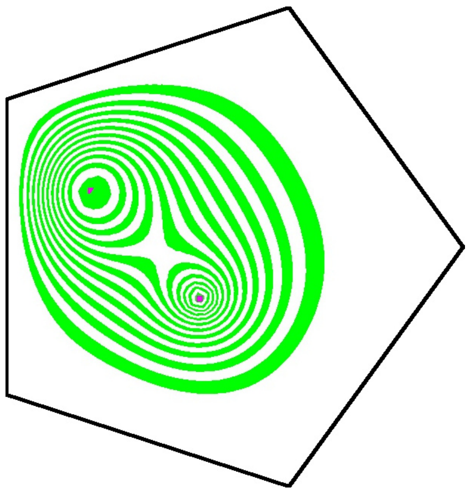

**Figure 3:** *Sum of two blends with ranges* $r_1 = 0.7$ *(left) and* $r_2 = 0.25$ *(right) over a 5-sided polygonal domain.*

## 4. Examples

In the following, we demonstrate some basic operations on a six-sided transfinite surface. Due to its highly curved shape, the interior becomes too flat. In the first example (Fig. 4), we add a single control point to adjust the central region. The displacement vector is very small here, with short range. The mean curvature maps show a highly localized change.

In Figure 5, we added a new feature to the surface by placing two control points with medium range.

Finally, we show a change in fullness by adding several control points of various ranges (Fig. 6).

## Conclusion and future work

We have shown a simple and very general method for adding interior control to any surface. With this approach, an arbi-

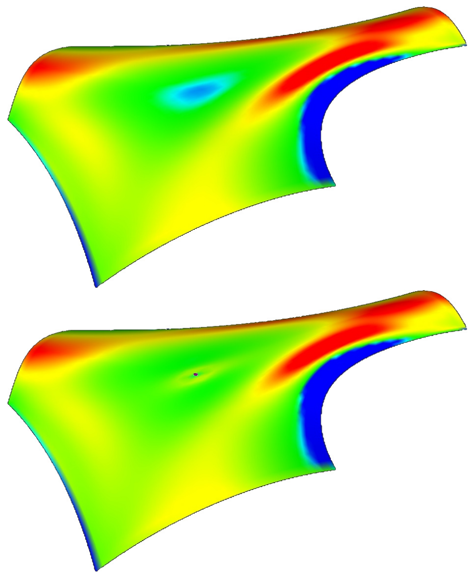

**Figure 4:** *Adjusting the center with a single control point.*

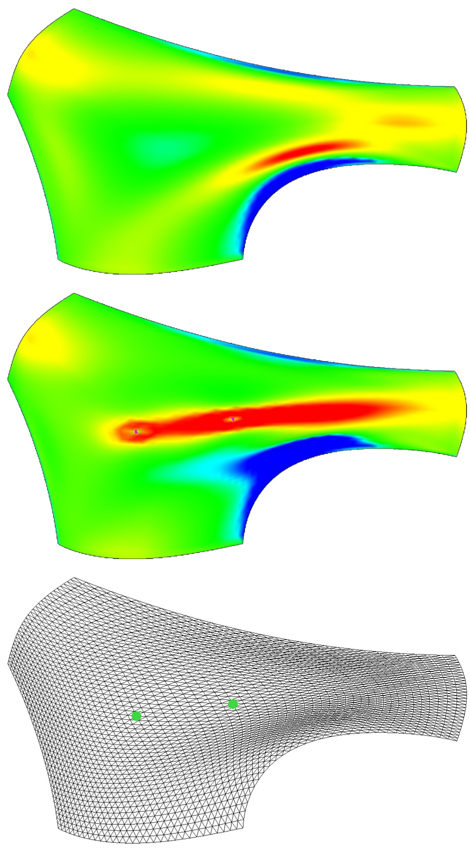

**Figure 5:** *Adding a feature using two control points. The bottom image shows the triangulation and the control positions (in green).*

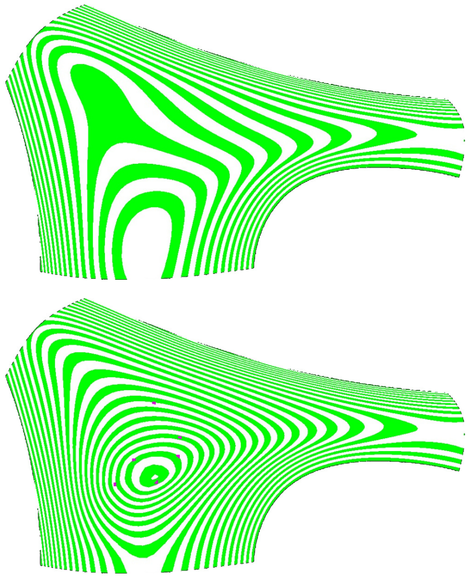

**Figure 6:** *Changing the fullness (surfaces shown with contouring).*

trary point of the surface can be repositioned while retaining the original smoothness.

Forcing the surface to go through multiple fixed points, or even an auxiliary curve or surface, may be achieved by modifying the harmonic equation. The related equations and a corresponding intuitive interface is the subject of future research. Different reparameterizations are also possible: the range of the displacement effect is currently controlled with a single number, but it would be interesting to add a directional component for anisotropic effects.

**Acknowledgements**

This work was supported by the Hungarian Scientific Research Fund (OTKA, No. 124727). The author thanks Tamás Várady for his valuable comments.

**References**

1. Jeff B. Allan, Brian Wyvill, and Ian H. Witten. A methodology for direct manipulation of polygon meshes. In *New Advances in Computer Graphics*, pages 451–469. Springer, 1989.
2. James R. Bill and Suresh K. Lodha. Computer sculpting of polygonal models using virtual tools. Technical report, University of California at Santa Cruz, 1994.
3. Lubomir T. Dechevsky, Arne Lakså, and Børre Bang. Expo-rational B-splines. *International Journal of Pure and Applied Mathematics*, 27(3):319–369, 2006.
4. Kiyokata Kato. Generation of $n$-sided surface patches with holes. *Computer-Aided Design*, 23(10):676–683, 1991.
5. Jiří Kosinka, Malcolm A. Sabin, and Neil A. Dodgson. Control vectors for splines. *Computer-Aided Design*, 58:173–178, 2015.
6. Charles T. Loop and Tony D. DeRose. A multisided generalization of Bézier surfaces. *ACM Transactions on Graphics (TOG)*, 8(3):204–234, 1989.
7. Ulrich Pinkall and Konrad Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2(1):15–36, 1993.
8. Péter Salvi. $G^1$ hole filling with S-patches made easy. In *Proceedings of the 12th Conference of the Hungarian Association for Image Processing and Pattern Recognition*. KÉPAF, 2019.
9. Péter Salvi, István Kovács, and Tamás Várady. Computationally efficient transfinite patches with fullness control. In *Proceedings of the Workshop on the Advances of Information Technology*, pages 96–100. BME, 2017.
10. Péter Salvi and Tamás Várady. Multi-sided Bézier surfaces over concave polygonal domains. *Computers & Graphics*, 74:56–65, 2018.
11. Tamás Várady, Alyn Rockwood, and Péter Salvi. Transfinite surface interpolation over irregular $n$-sided domains. *Computer-Aided Design*, 43(11):1330–1340, 2011.
12. Tamás Várady, Péter Salvi, and Alyn Rockwood. Transfinite surface interpolation with interior control. *Graphical Models*, 74(6):311–320, 2012.
13. Tamás Várady, Péter Salvi, Márton Vaitkus, and Ágoston Sipos. Multi-sided Bézier surfaces over curved, multi-connected domains. *Computer Aided Geometric Design*, 78:101828, 2020.
14. Lexing Ying and Denis Zorin. A simple manifold-based construction of surfaces of arbitrary smoothness. *ACM Transactions on Graphics (TOG)*, 23(3):271–275, 2004.