Pacific Graphics (2016)
E. Grinspun, B. Bickel, and Y. Dobashi (Editors)

Short Paper

Approximating Point Clouds by Generalized Bézier Surfaces

Péter Salvi[1,2], Tamás Várady[1], Kenjirō T. Miura[2]

[1] Budapest University of Technology and Economics
[2] Shizuoka University

Abstract

The Generalized Bézier or GB patch is a recently published, control point based multi-sided surface scheme [VS16]. It is fully compatible with tensor product Bézier patches and can interpolate prescribed boundaries and cross-derivatives of different degrees. It also has control points in the interior that can be used for editing and optimization. A typical application is when a large mesh needs to be approximated by a watertight collection of surface patches. Assuming that a general topology curve network has already partitioned the mesh into n-sided regions, we wish to approximate the data points in the interior. A least-squares minimization, similar to B-spline surface approximation, is applied, leading to a linear system of equations. We discuss the specific problems that need to be solved for GB patches, related to blending functions, parameterization, and smoothing. A few examples are shown to demonstrate the results.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

The representation of multi-sided surface patches is an important issue when complex free-form objects need to be defined. There is a wide variety of choices, and contradictory requirements need to be prioritized to select the "best" for a given application. A general topology patchwork can be represented by trimmed tensor-product surfaces, but difficulties arise when boundaries need to be accurately interpolated or smooth connections need to be ensured. Recursive subdivision is another option, but forcing the limit surface to interpolate boundary curves with cross-derivatives is a hard problem. Multi-sided transfinite surfaces ensure watertight and smooth connections, and define the surface interior automatically; but this may be a disadvantage when the interior needs to be edited and optimized. The control point based multi-sided patches attempt to combine the merits of the previous representations, facilitating boundary interpolation and interior shape control.

The Generalized Bézier or GB patch is a recently published multi-sided patch formulation [VS16]. It is fully compatible with tensor product Bézier patches and can interpolate various boundary curves and cross-derivatives given in Bézier form. The scheme produces a naturally distributed set of control points in the interior of the patch, that can be used for editing and optimization—however, no details have been supplied on these topics. In this paper our interest is how to approximate data points by GB patches. We assume that a large mesh has already been partitioned into smaller n-sided regions by a general topology curve network. This produces our input, i.e., boundaries to be interpolated and interior points to be approximated.

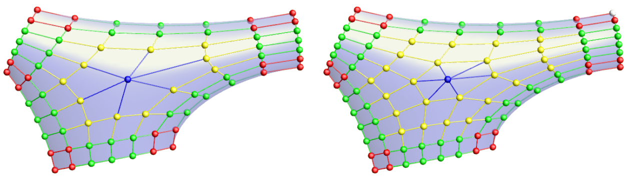
After reviewing related work (Section 2), we briefly present the definition of GB patches (Section 3). We discuss how standard least-squares approximation techniques can be adapted to GB patches, including blending functions, parameterization and smoothing (Section 4). Finally, we demonstrate our results using synthetic and real data examples (Section 5).

2. Previous Work

There are well-established techniques to approximate data points by means of quadrilateral Bézier or B-spline surfaces. Multi-sided transfinite patches are limited in this sense, as generally there are no free parameters for optimization, however, control point based multi-sided patches do comprise interior controls for this purpose. The majority of related papers focus solely on the surface representation and do not deal with applications. Nevertheless, we enumerate a few patch formulations, where approximation algorithms—similar to ours—might be applied.

(i) The Zheng–Ball surface [ZB97] has almost the same control grid as the GB patch, and exact degree elevation can be achieved through fairly complex algebra. Its parameterization is defined through implicit constraints, solutions of which are known only for $n \leq 6$.

(ii) The S-patch [LD89] is another generalization of the quadrilateral (and triangular) Bézier surface, with many nice mathematical properties. Due to the high number of control points even at relatively low degrees and their peculiar arrangement, these patches are hardly suitable for interactive design.



(a) $d = 6$, $l = 3$ (b) $d = 7$, $l = 4$

Figure 1: Control point structure of a 6-sided patch.

(iii) Toric patches [Kra02] are defined over a polygon on an integer lattice—an interesting, but slightly limited construction. G^1 continuity between surfaces is complex, but possible.

GB patches are suitable to apply standard techniques, using their multi-sided control net and special blending functions, as follows.

3. Preliminaries

In this section we summarize the formulation of the GB patch. For more details see the original paper [VS16].

The surface is defined over a regular polygon in the (u,v) domain plane. We assign local parameter functions s_i and h_i to each side of the polygon:

$$\begin{aligned} s_i(u,v) &= \lambda_i(u,v)/\left(\lambda_{i-1}(u,v)+\lambda_i(u,v)\right), \\ h_i(u,v) &= 1-\lambda_{i-1}(u,v)-\lambda_i(u,v), \end{aligned} \tag{1}$$

where $\lambda_i(u,v)$ is the Wachspress coordinate [Flo15] corresponding to the i-th vertex. The (u,v) arguments of these functions are henceforth omitted for ease of notation.

Note that s_i, the *side parameter*, takes values between 0 and 1 as it sweeps from the "left" to the "right" side; while h_i, the *distance parameter*, is zero on the base side, and increases monotonically to 1 as it approaches the "far" sides, see Figure 4a.

The surface equation is

$$S(u,v) = \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1} \mu^i_{j,k} B^d_{j,k}(s_i,h_i) C^{d,i}_{j,k} + C_0 B^d_0(u,v), \tag{2}$$

where n is the number of sides, $d \geq 3$ is the degree, and $l = \lceil d/2 \rceil$ is the number of control point layers (rows) for each side. Figure 1 shows a 6-sided control grid for even and odd degrees. The indexing scheme is side-based: $C^{d,i}_{j,k}$ refers to the j-th control point in the k-th row of the i-th side. Note that most control points have two indices, e.g. $C^{6,1}_{5,2} = C^{6,2}_{2,1}$. C_0 denotes the central control point.

The control points $C^{d,i}_{j,k}$ are multiplied by biparametric Bernstein polynomials $B^d_{j,k}(s,h) = B^d_j(s) \cdot B^d_k(h)$ and a rational function $\mu^i_{j,k}$. This latter function takes constant values (0, 0.5, or 1), except at 2×2 control points in the corners, as depicted in Figure 2, where α and β are defined as

$$\alpha_i = \frac{h_{i-1}}{h_{i-1}+h_i}, \qquad \beta_i = \frac{h_{i+1}}{h_{i+1}+h_i}. \tag{3}$$

These rational weights are needed to interpolate the boundary and first cross-derivative of the Bézier surface defined by the first two rows of the control net.

The central control point C_0 is multiplied by the weight deficiency B^d_0, such that the sum of all blending functions equals 1:

$$B^d_0(u,v) = 1 - \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1} \mu^i_{j,k} B^d_{j,k}(s_i,h_i). \tag{4}$$

Using weight deficiency as the central blending function raises interesting issues; these will be discussed in Section 4.4.

4. Approximation

The general framework of the approximation is shown in Algorithm 1. In the rest of this section, we will look at all required steps.

4.1. Parameterization and Initial Surface

Parameters associated with the data points are computed by projecting them to the current surface. We use a projection to a discrete triangular realization of the GB patch, linearly interpolating the parameters at the triangle vertices.

The initial surface is computed from the boundary conditions that determine the first two control rows, resulting in a cubic GB patch, which can be degree elevated as needed. (If only the boundary curves are given, we can assume zero twists to generate an initial control grid.)

A much better initial surface (and parameterization) can be obtained, however, if we know a "central" position in the region. Then we can constrain the middle point of the surface to interpolate that position [VS16], leading to a surface closer to the given data, compare Figures 6a and 6b, and their values in Table 1.

4.2. Fitting

Given data points P_i with associated parameters (u_i,v_i), we wish to minimize the expression

$$\sum_i \left(S(u_i,v_i)-P_i\right)^2 = \min, \tag{5}$$

which leads to a linear system of equations in the unknown control points $C^{d,i}_{j,k}$ and C_0. Each control point except the central one appears with two indices in Equation (2), but we can avoid the ambiguity with the constraint $j \leq \lfloor d/2 \rfloor$. Now the blending function associated with $C^{d,i}_{j,k}$ is

$$\mu^i_{d-k,j} B^d_{d-k,j}(s_i,h_i) + \mu^{i+1}_{j,k} B^d_{j,k}(s_{i+1},h_{i+1}). \tag{6}$$

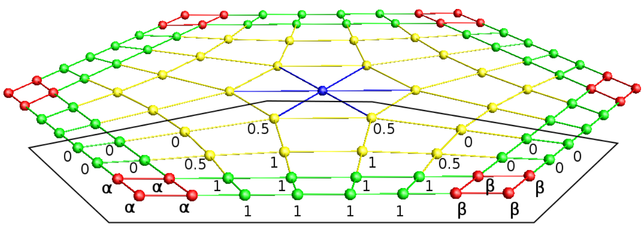


Figure 2: The $\mu^i_{j,k}$ weights of control points along one side.

Algorithm 1 The approximation framework.

create an initial surface
parameterize the points by the initial surface
while the deviation is above tolerance:
  fit a new surface on the points with smoothing terms
  if the deviation is still above tolerance:
    elevate the degree of the current surface
    re-parameterize the points by the elevated surface

According to the application at hand, we can fix the outermost one or two rows ($k = 0, 1$) to retain C^0 or G^1 interpolation properties. Then the equation system can be solved in a least-squares sense.

4.3. Smoothing

The approximation by the method in the previous section is optimal in a sense, but the computed control points may wiggle and thus be inappropriate for further design (see Figure 7a). It is standard practice to add some kind of smoothing term T to the system, that may slightly decrease accuracy, but yields a much better control grid. The expression to be minimized is thus adjusted to

$$\sum_i (S(u_i, v_i) - P_i)^2 + \sigma T = \min, \quad (7)$$

where σ is a smoothness parameter controlling the extent of fairing.

A well-known technique for reducing the oscillation of the control points is to pull each point towards the mass center of its neighbors. Figure 3 illustrates this effect on a curve. In the case of GB patches, this is done analogously by setting

$$T = \sum_{i=1}^{n} \sum_{j=1}^{\lfloor d/2 \rfloor} \sum_{k=1}^{l-1} T^i_{j,k} + T_0, \quad (8)$$

where

$$\begin{aligned} T^i_{j,k} &= \left(C^{d,i}_{j,k} - \frac{1}{4}\left(C^{d,i}_{j-1,k} + C^{d,i}_{j+1,k} + C^{d,i}_{j,k+1} + C^{d,i}_{j,k-1} \right) \right)^2, \\ T_0 &= \left(C_0 - \frac{1}{n} \sum_{i=1}^{n} C^{d,i}_{l,l-1} \right)^2. \end{aligned} \quad (9)$$

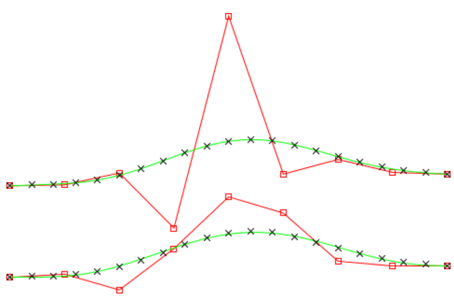
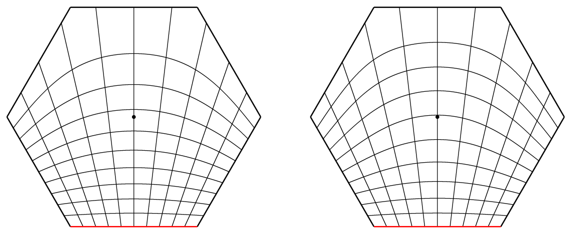


Figure 3: Fitting a Bézier curve on a set of points without smoothing (top) and with smoothing (bottom).



(a) $\delta = 0$ (b) $\delta = 4$

Figure 4: Constant parameter lines of the 6-sided domain.

Here indexing is naturally extended to include control points $C^{d,i}_{j,l}$, with $C^{d,i}_{l,l} = C_0$ for even degrees. On the adjustment of the σ parameter, see examples in Section 5.

4.4. Central Blending Function

We have found that for larger n or d, the central blending function B^d_0 becomes overweighted, and accordingly the effect of the other interior control points is shrinking. This is unsatisfactory for fitting, as well as from a designer's perspective.

The problem is rooted in the parameterization of the patch. At the domain center, where B^d_0 is maximal, its value depends solely on the h_i parameters. These tend to condense near the edges, as n gets larger. We propose the alternative formulation

$$\hat{h}_i = (1 - \lambda_{i-1} - \lambda_i) \cdot (1 - \delta \cdot \lambda_{i-2}\lambda_{i+1}), \quad (10)$$

where the dilation factor δ "drags" the constant parameter lines towards the distant sides of the domain, and $\delta = 0$ gives back the original parameterization, see Figure 4. Note that this works only when $n \geq 5$.

There is a theoretical maximum for δ, such that $\hat{h}_i \geq 0$. An even stronger criterion is to require that $B^d_0 \geq 0$ on the whole domain. It is easy to estimate these maximum values.

We believe that for each (n, d) pair, there is an optimal δ—this issue is part of future work. As a rule of thumb, we have used settings that give a specified maximum for B^d_0, such as 0.2.

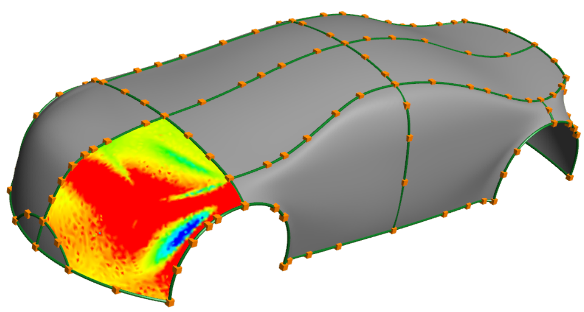
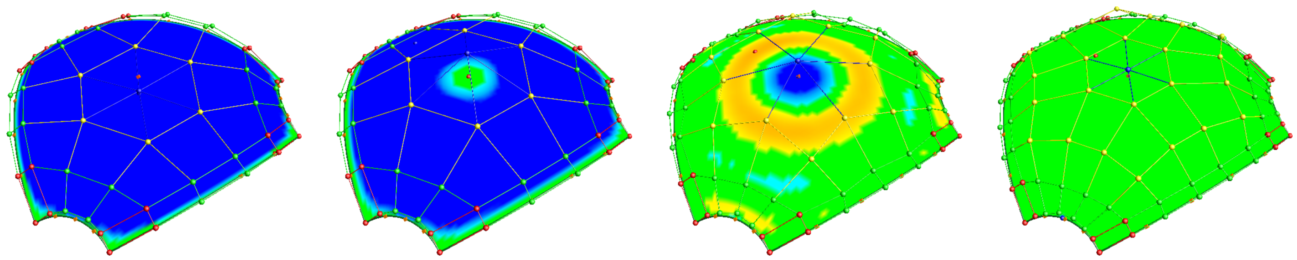
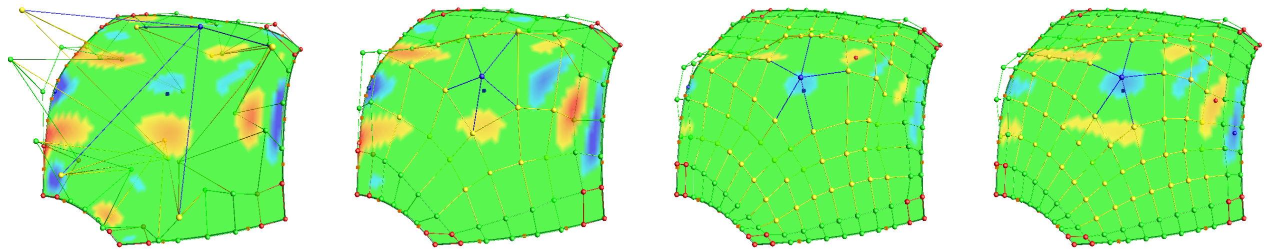


Figure 5: The concept car model. The mesh to be approximated is shown with its mean curvature map.



(a) Initial surface ($d = 5$) (b) Same as (a), with fixed center (c) Optimized surf. ($d = 6$, $\sigma = 0.1$) (d) Optimized surf. ($d = 7$, $\sigma = 0.1$)

Figure 6: Fitting a 6-sided patch on a sphere showing deviations (red $> 0.12\%$, $-0.04\% <$ green $< 0.04\%$, $-0.12\% >$ blue).



(a) Optimized ($d = 6$, $\sigma = 0$) (b) Optimized ($d = 7$, $\sigma = 0.2$) (c) Optimized ($d = 10$, $\sigma = 0.2$) (d) Optimized ($d = 10$, $\sigma = 0.5$)

Figure 7: Approximating a car body part showing deviations (red $> 0.26\%$, $-0.13\% <$ green $< 0.13\%$, $-0.26\% >$ blue).

5. Examples

As a first, synthetic example, we approximate points sampled from a sphere with a 6-sided patch (Figure 6). The images show deviation maps with red and blue markers at the maximal loci. Here we can supply the exact location of the surface center, leading to a better initial surface (6b). Subsequent degree elevations and optimizations lead to a high-quality fit (6d); see also Table 1.

In the second example, we approximate part of a concept car model (Figure 5). The curve network is fairly complex, and so is the input mesh, as can be seen from its curvature map. A good fit is achieved by a 10-degree surface (Figure 7c); choosing a larger smoothing value gives a nicer control structure (7d), while deviations increase only to a small extent, see Table 1.

Model	Fig.	d	σ	max. dev.	avg. dev.
Sphere	6a	5	-	6.390%	2.510%
	6b	5	-	0.817%	0.360%
	6c	6	0.1	0.171%	0.030%
	6d	7	0.1	0.035%	0.006%
Car	7a	6	0	0.516%	0.083%
	7b	7	0.2	0.423%	0.073%
	7c	10	0.2	0.236%	0.048%
	7d	10	0.5	0.260%	0.063%

Table 1: Maximal and average deviations of the examples with various settings. All deviations are percentages of the bounding box.

Conclusion

We have presented an algorithm for approximating data points within an n-sided loop of boundaries. Generalized Bézier patches, due to their simple control structure and blending functions, make it possible to extend standard surface fitting techniques. The algorithm is driven by two internal parameters, one controls the parameterization of the domain, the other adds smoothing energies to avoid the oscillation of the control structure. The optimal combination of these two quantities is subject of future research.

Acknowledgements

This work was partially supported by a grant from the Hungarian Scientific Research Fund (No. 101845) and the Suzuki Foundation. All images were generated by the Sketches system (ShapEx Ltd, Budapest); the authors appreciate the help of György Karikó.

References

[Flo15] FLOATER M. S.: Generalized barycentric coordinates and applications. *Acta Numerica 24* (2015), 161–214. 2

[Kra02] KRASAUSKAS R.: Toric surface patches. *Advances in Computational Mathematics 17*, 1 (2002), 89–113. 2

[LD89] LOOP C. T., DEROSE T. D.: A multisided generalization of Bézier surfaces. *Transactions on Graphics 8* (1989), 204–234. 1

[VS16] VÁRADY T., SALVI P.: A multi-sided Bézier patch with a simple control structure. *Computer Graphics Forum 35*, 2 (2016), 307–317. 1, 2

[ZB97] ZHENG J., BALL A. A.: Control point surfaces over non-four-sided areas. *Computer Aided Geometric Design 14*, 9 (1997), 807–821. 1