Box-spline based CSG blends

Jörg Peters Michael Wittman

Purdue University

Abstract

The zero set of a trivariate spline is used to blend basic CSG surfaces of algebraic degree up to four. The resulting volume-bounded blend surface is generically curvature continuous, and piecewise of algebraic degree four itself, independent of the number of surfaces joined. The algorithm consists of two parts: representing each of the n basic surfaces within the blend volume as a trivariate (box-)spline, and combining the information from the n 3D arrays of spline coefficients into one that represents a new spline. The zero set of this new spline defines the blend surface. It is traced using approximation by subdivision.

1 Introduction

The blending of surfaces that smoothly join primary surfaces has motivated extensive research both on parametrically and implicitly defined surfaces (see e.g. the surveys [27], [26]). This paper suggests a new implicit approach based on the zero set of the average of box splines. The approach may be used to smoothly join an arbitrary number of low-degree primary surfaces and results in a blend that is itself of a fixed low algebraic degree and consists of the zero set of a fixed number of polynomial pieces. Specifically, the blend surface has the following properties:

- The blend surface is curvature continuous.
- The blend surface joins C^2 with input surfaces of algebraic degree three (and some of degree four) that are separate outside the blend volume or join C^2 outside the blend volume.
- The approximation order to general smooth surfaces is $O(h^4)$.
- The algebraic degree of the blend surface is four, independent of the number of input surfaces.
- The blend surfaces can be rendered stably and moderately fast.
- The representation is compatible with set-theoretic representations; e.g. point classification (set membership determination) is supported.
- The surface representation has volume elements associated with it.
- The blend is volume bounded.

Related literature. Starting with [8], box splines have been developed during the last decade. With the notable exception of [24], [2], [4], and [10], most results on non-tensor-product box-splines have been published in journals on approximation theory and seem to not have entered the standard repertoire of CAD. This is unfortunate since the representation combines smoothness, efficient evaluation and high approximation order [9] as summarized in the appendix.

Besides the well-known implicit blending constructions of [12], [13], [15], [14], there are currently two main approaches to defining the individual pieces of a function whose zero set represents a surface. The first is to generate an approximate triangulation of the surface and then erect a shell-like structure of trivariate polynomial pieces over this parametrization [25], [6], [11], [5], [1], [16]. The second is to define a function on a regular, global lattice, for example, a piecewise triquadratic, C^1 tensor-product spline [17]. The regular lattice has the advantage that non-rectilinear features of the surface do not require special treatment and that no parametrization other than the lattice structure is imposed. The algorithm defined below is of the second kind. It can alternatively be viewed as a systematic way of creating a field in the spirit of 'blobby objects' used in animation [28] or as approximate version of constructive geometry in the sense of Ricci [23].

The paper is structured as follows. Section 2 explains the basic idea, discrete blending in coefficient space, the

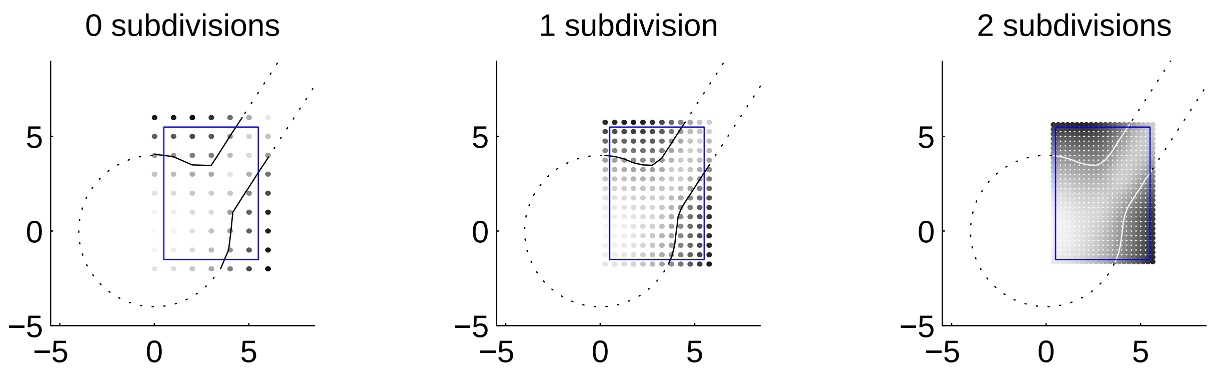

Figure 1: The grid of dots represents the array of coefficients that weigh shifts of the positive, symmetric function depicted in Figure 12(*bottom*). Darker dots correspond to negative array entries. The array is initialized as the maximum of two arrays containing the coefficients of a spline whose zero set is a circular disk, and another array that represents the intersection of two linearly bounded half spaces: the result is a C^1 quadratic blend of $-x^2-y^2+16>=0$ union $(-x+.65y+3.5>=0$ intersect $x-.65y-.75>=0)$. (cf. [19]).

third section gives a formal statement of the algorithm, Section 4 gives the details of the 3D construction and Section 5 discusses the properties of the representation and gives examples. A review of box spline properties used in this paper is appended.

2 Discrete blending in coefficient space

We start with an intentionally simple example in two rather than three dimensions. Consider two arrays with ± 1 entries:

$$M_1 := \begin{matrix} +1 & +1 \\ -1 & -1 \end{matrix} \qquad \text{and} \qquad M_2 := \begin{matrix} +1 & -1 \\ +1 & -1 \end{matrix}$$

and a third array generated as the componentwise maximum

$$M_3 := \max\{M_1, M_2\} := \begin{matrix} +1 & +1 \\ +1 & -1 \end{matrix}$$

We interpret each entry $M_k(i,j), k \in \{1,2,3\}, i,j \in \{1,2\}$ as the coefficient of a positive, symmetric function of compact support in two variables centered at (i,j), say the bell-shaped function displayed in Figure 12,*bottom* so that the array M_k represents the superposition of the four functions. We can visualize the zero set associated with M_1 as a horizontal line segment, the zero set of M_2 as a vertical line segment and the zero set of M_3 as a right angle, that has to be smoothed out since the four function shifts are each smooth and regular and hence so is their zero set. Thus, if we interpret the positive side of each line segment as the interior of some two dimensional object, then M_3 represents a smoothed-out union of the two objects. Correspondingly, we call this approach *discrete blending in coefficient space*. A number of detailed examples of this blending in two dimensions have been worked out in [19]. For example, in Figure 1,*left*, the grid of dots represents the array of coefficients weighing shifts of the positive, symmetric function depicted in Figure 12,*bottom*. The shading is proportional to the value of the array-entry, the darker the more negative, with black denoting “outside”. The array is initialized as the maximum of two arrays containing the coefficients of a spline whose zero set is a circular disk, and another array that represents the intersection of two linearly bounded half spaces. The details of the initialization of this two-dimensional example are given in [19], Section 3. The general algorithm for the initialization is explained in the next two sections.

A useful property of surfaces defined as shifts of a box-spline is that the surface is approximated well by the piecewise linear interpolant to the coefficients in the array, the better the less the local variation between the coefficients (see also Appendix 7.3). This is the basis for fast algorithms for graphic display and rendering because a simple averaging rule, called *subdivision*, increases the number of coefficients and decreases the variation between them quickly, so that the function values are well approximated by the coefficients after only a few steps of subdivision. Correspondingly, the zero set can be well approximated by the sign change in the coefficient array. This is illustrated by the sequence in Figure 1 where, from left to right, the zero level set is approximated at three consecutive subdivision steps and the white curve inside the rectangle of the second subdivision is a piecewise linear approximation to the quadratic blend curve. Note also that the ever denser set of coefficients shrinks towards a two dimensional ver-

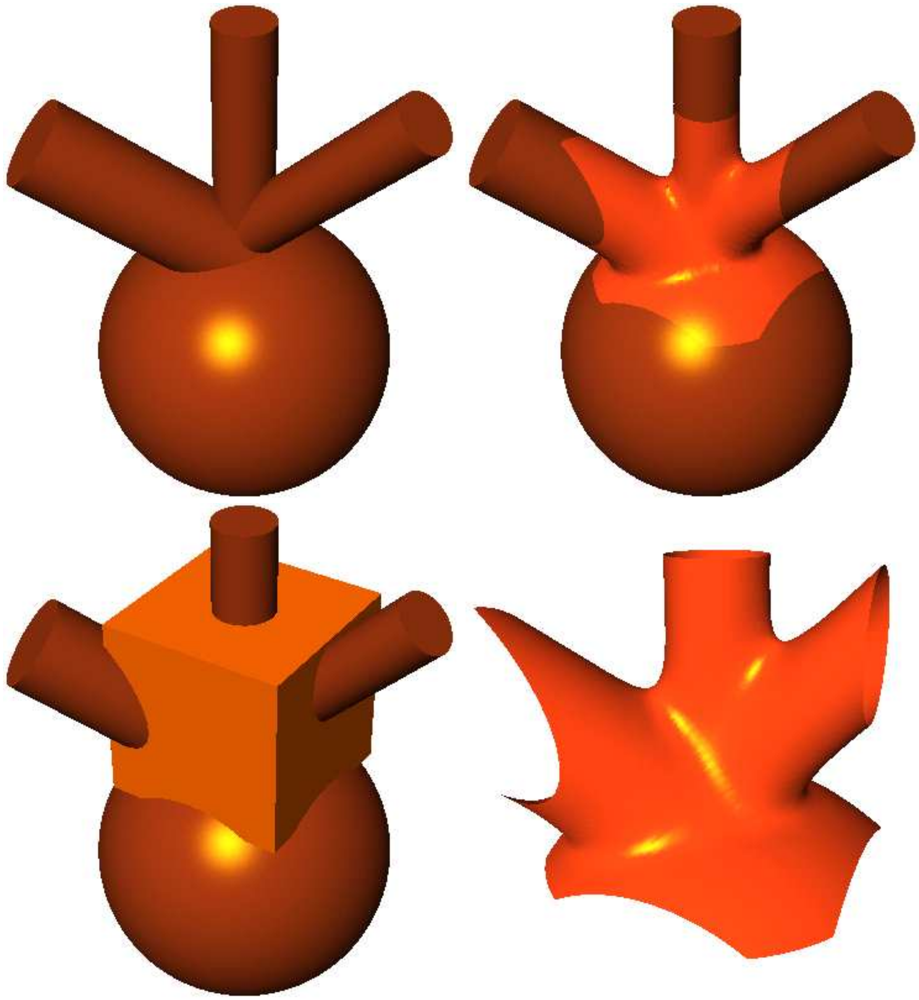

Figure 2: (*Upper left*) Primary surfaces, (*lower left*) blend volume in place, (*lower right*) zero set of the box-spline, (*upper right*) blended ensemble.

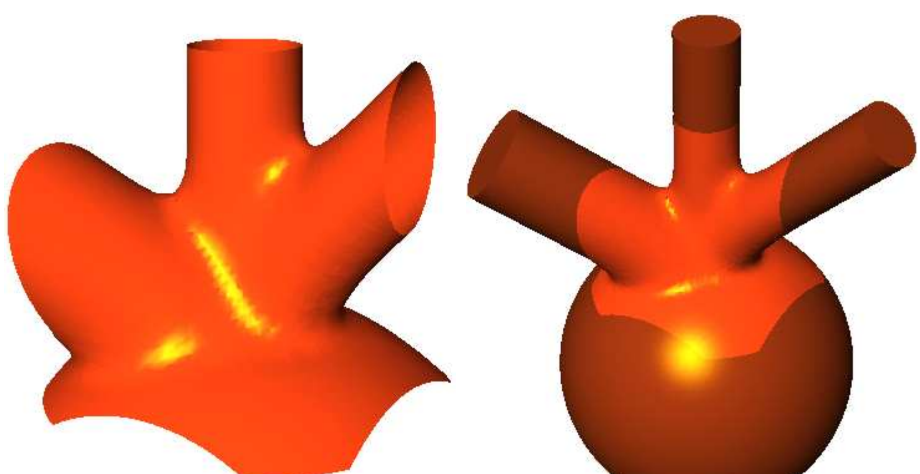

Figure 3: Termination of the blend with the left cylinder altered by subtracting a half space.