sion of the blend volume.

We are now ready for a complete example in three dimensions illustrated by Figure 2. On the left, we see the primary surfaces, and below, a blend volume, here a simple brick shape, that covers the volume to be blended. The blend volume serves as spline domain. The zero set of this spline is shown on the right, both in place as blending surface and enlarged, by itself. The blend surface is rendered just like the white blend curve in the two-dimensional example. The next section is devoted to explaining the details of the algorithm.

3 The algorithm

The *input* to the algorithm are

a. the n defining polynomials $p_i(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^3$, of the primary surfaces.

b. a blend volume and its partition (default 5 by 5 by 5),

c. a map that combines n array entries to one, and scaling factors (default 1), and

d. a number of averaging steps (default 0).

For example, the defining polynomial of the unit sphere $p_1 \geq 0$ is $p_1(\mathbf{x}) = p_1(x,y,z) = -x^2 - y^2 - z^2 + 1$; that is, the interior corresponds to positive values of the defining polynomial. The *blend volume* is a typically block-shaped region in 3-space that encloses the blend surface to be constructed. The blend volume need not be aligned with the global xyz coordinate axes and can be the image of a cube under a more general map. An example of an operation is to choose the maximum of a set of coefficients. This will correspond to an approximate union or blend, while a minimum operation approximates an intersection.

The three *steps of the algorithm* are as follows. First, each of the n basic primitives within the blend volume is automatically expanded as a trivariate spline. That is, the defining polynomial of each basic primitive, is written as a linear combination of shifts of the spline basis function M. For example, the first defining polynomial is $p_1(\mathbf{x}) = \sum a^1(\alpha)M(\mathbf{x} - \alpha)$, where $\alpha = (\alpha_x, \alpha_y, \alpha_z) \in \mathbb{R}^3$ is a point on a lattice that sub-divides the blend volume. The explicit formulas for the constants $a^1(\alpha) \in \mathbb{R}$ in terms of the coefficients of the defining polynomial are derived in Section 4. The zero set of the spline then defines the surface. Second, the operation specified generates from the n 3D arrays of spline coefficients one entry in each position and hence a new 3D array and associated spline, $p(\mathbf{x}) = \sum a(\alpha)M(\mathbf{x} - \alpha)$. Typically, when blending, $a(\alpha)$ is defined as $a(\alpha) := \max_\ell\{a^\ell(\alpha)\}$ except for α in the boundary set. The *boundary set* of the blend consists of α within a certain range of a sign change on a face of one of the arrays where all other arrays have negative entries. A face of the array is the 2 dimensional subarray where one of the three indices is either maximal or minimal. Thus where a single input surface meets the blend volume, its coefficients are preferred. This preference also holds during the optional third and final step, local averaging that allows to smoothen features, and thus guarantees a smooth transition between primary surface and blend surface. When surfaces are added to the blend ensemble, blends on blends are avoided by regenerating the blend based on all primitives.

4 Blend Surfaces

According to the first step of the algorithm, we select a suitable box-spline in three variables. A poor choice of box-spline can result in artifacts as shown in Figure 4. We choose for M the centered 7-direction quartic box spline which is a serendipitous element among the trivariate box-splines in that it combines a high degree of symmetry, with high reproduction of polynomials, almost maximal smoothness and low degree. The specific properties, degree four, curvature continuity and reproduction of all cubics and some quartics, are derived in the appendix.

The first step requires representing the defining polynomial p_ℓ of each primary surface in terms of shifts of M:

$$p_\ell(\mathbf{x}) = \sum a^\ell(\alpha)M(\mathbf{x} - \alpha).$$

Since each p_ℓ decomposes into monomials $x^i y^j z^k$, namely $p_\ell = \sum c^\ell_{ijk} x^i y^j z^k$, it suffices to determine scalar coefficients $a_{ijk}(\alpha)$ such that the following (Marsden) identity holds

$$x^i y^j z^k = \sum a_{ijk}(\alpha)M(\mathbf{x} - \alpha).$$

Then

$$a^\ell(\alpha) := \sum_{i+j+k} c^\ell_{ijk} a_{ijk}(\alpha).$$

For the centered 7-direction box-spline, the scalar coefficients for $i + j + k < 3$ are particularly simple:

$$a_{ijk}(\alpha) = \alpha_x^i \alpha_y^j \alpha_z^k - \begin{cases} 5/12 \text{ if } i = 2 \text{ or } j = 2 \text{ or } k = 2 \\ 0 \text{ else} \end{cases}.$$

For example, to represent the cylinder

$$(x - y)^2 + z^2 = 1$$

the array entry $(\alpha_x, \alpha_y, \alpha_z) \in \mathbb{N}^3$ for a unit partitioned blend volume at the origin is

$$(\alpha_x^2 - \frac{5}{12}) - 2\alpha_x\alpha_y + (\alpha_y^2 - \frac{5}{12}) + (\alpha_z^2 - \frac{5}{12}) - 1.$$

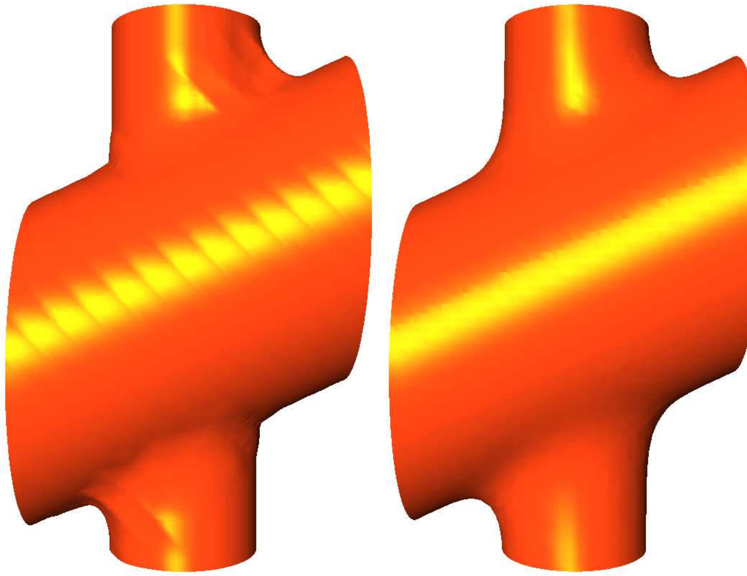

Figure 4: Artifacts on a blend surface generated as the zero set of the 5-direction box spline (*left*) with the unsymmetric direction matrix $\Xi := \begin{bmatrix} 1 & 0 & 0 & -1 & -1 \\ 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & -1 & 1 \end{bmatrix}$ The zero set of this box-spline and hence the blend consists of quadrics. The zero set of the 7-direction box spline is shown (*right*).

To map the integer indices of the array onto the partition of the blend volume in the general case let A be the transformation that maps a box of size $n_1 \times n_2 \times n_3$ at the origin to the blend volume. Then the 3D array V is initialized as

$$V(\beta) := a(\alpha) = a(A(S_M\beta)), \quad \beta \in [0..n_1] \times [0..n_2] \times [0..n_3].$$

Here S_M is a shift by 1.5 followed by scaling by $(n_i + 1)/n_i$ in the ith component so that evaluation by subdivision converges exactly to the blend volume.

For additional smoothing, an optional averaging step may be added. None of the figures displayed in this paper required this extra smoothing. With the boundary set kept fixed and positive and negative coefficients are averaged separately by a Gaussian. The averaging is not expensive since it involves at most the $n_1 \times n_2 \times n_3$ interior coefficients of the array V. The boundary set of the 7-direction box-spline blend consists of the indices within an index range of 2 from sign changes on the array boundary.

5 Discussion of the blend surfaces

5.1 Smoothness and reproduction

Within the blend volume the blend surface is the zero set of a C^2 function. If this function is regular within the blend volume then, by the implicit function theorem, its zero set is also C^2. Using singular defining polynomials, it is also possible to represent singularities such as the apex of a cone. A C^2 join with the input surface across the boundary of the blend volume is guaranteed if the blend surface matches the input surface up to second order across the boundary. This is in particular the case if the boundary sets do not overlap and the defining polynomial is of degree less than four or equal to four with no fourth order mixed monomials. *Cyclides*, whose defining polynomial is $p(x, y, z) = (x^2+y^2+z^2-m^2+b^2)^2 - 4(ax-cm)^2-4b^2y^2$ (see e.g.[22]), have a combination of mixed terms $x^2y^2 + x^2z^2 + y^2z^2$ that is not reproduced exactly. Yet, the box-spline approximation is visually indistinguishable from the correct zero set (c.f. Figure 5) traced out using the explicit parametrization. This may be explained by the $O(h^4)$ approximation order of the box spline. If we choose an (n, n, n)-partition of the blend volume, then the error in the mixed terms is of the order n^{-4} with respect to the size of the bounding box. Moreover, for the smooth blend, global reproduction is not necessary. Local reproduction of the first three Taylor terms across a plane (or other blend volume boundary) suffices. For example, if we choose the blend volume boundary to coincide with a line of curvature of the cyclide then only a quadratic has to be reproduced for continuity and a cone for tangent continuity.

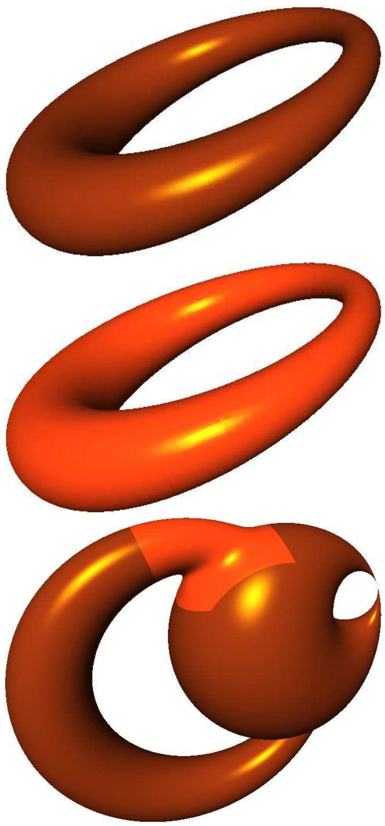

Figure 5: Exact cyclide, approximate cyclide as zero set of box-splines and two blended cyclides.

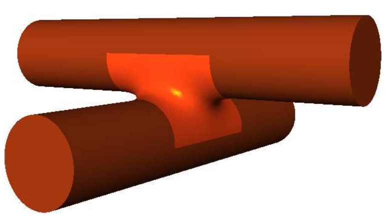

Figure 6: Dominance of the interior can join two non-intersecting primary surfaces inside the blend volume if they are sufficiently close. To keep the surfaces separate, a finer partition of the blend volume suffices, or, of course, removal of the blend volume.

5.2 Zero sheets, rendering, evaluation and point-classification

In the approximate union or blend operation represented by the the maximum operation in Step 2 of the algorithm, positive entries dominate. That is, if any of the entries in a one of the 3D arrays is positive, the resulting entry will be positive. Conversely, a negative entry will only appear in the resulting array, if all contributing entries are negative. Thus interior volume dominates and hence zero sheets can disappear but no more sheets can appear than were present in the original n primary surfaces. At worst, as Figure 6 illustrates, separate zero sheets can join if they are sufficiently close within the blend volume. The analogous argument holds for smoothed intersections – here the intersection is smoothed by removing volume in the interior of some but not all primary surfaces.

To extract a continuous piecewise linear approximation of the zero sheet of the spline, we traverse the coefficient array to detect sign changes. In standard fashion, here additionally motivated by the tetrahedral support of the polynomial pieces, each cube with a sign change is split into tetrahedra according to Figure 10, associating the average of the values at the vertices with the center of each cube and cube face. For each edge whose endpoints have an opposite sign, we mark the midpoint. Each tetrahedron has either zero, three or four marked edge-midpoints. Correspondingly, we add no, one or two (coplanar) triangles connecting the midpoints to a list of triangles. The union of the triangles in the list then form the surface approximation.

The surface approximation is refined by averaging the array entries according to the subdivision rules of the 7-direction box spline (cf. the Appendix). That is, each value is replicated over a cube of half the edge length and then the values on this refined lattice are averaged consecutively in each of the four diagonal directions of the box spline. With each step of the subdivision, the approximation gains two additional digits of accuracy.

To test for intersection, we subdivide depth-first to obtain a nested sequence of bounding boxes. If a single point is to be classified, the check is first against the blend volume then against the subcubes and tetrahedra already generated. If this check fails because the point is very close to the boundary, the spline is exactly evaluated. A stable evaluation algorithm and code for the box-spline basis functions are given in [7].

5.3 Shape and detail control

Since the spline is the limit of the coefficients generated by the subdivision process, and subdivision reduces variation through averaging, its rough shape and hence