Smoothly Blending Successive Circular Interpolants in Two and Three Dimensions *

Richard Liska

Faculty of Nuclear Sciences and Physical Engineering

Czech Technical University in Prague

Břehová 7, 115 19 Prague 1, Czech Republic

e-mail: liska@siduri.fjfi.cvut.cz

Mikhail Shashkov

Los Alamos National Laboratory, T-7, Los Alamos

NM 87544, USA

e-mail: shashkov@lanl.gov

Blair Swartz

Los Alamos National Laboratory, T-7, Los Alamos

NM 87544, USA

e-mail: bks@lanl.gov

November 5, 1998

*The publisher recognizes the U. S. Government retains a non-exclusive royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U. S. Government purposes.

Abstract

Curvilinear edges and surfaces are developed to span a given logically-cuboid distribution of nodes. The underlying construction uses four successive nodes to define a curve spanning the middle pair as follows: One interpolates each of the two successive triples of nodes with the segment of a circle or straight line. Then one blends the two segments continuously between the middle pair of nodes. The blend is relatively linear in terms of arc-length along each segment. The union of such successive curve-sections forms a G^1 curve. Wire-frames of such curves define cell edges. Similar intermediate curvilinear interpolation of the wires defines cell faces, and their union defines G^1 coordinate-like surfaces. The Euclidean image of coordinate lines and surfaces of rectangular, cylindrical, and spherical coordinate systems are reconstructed exactly, given the nodes associated with tensor product data from the coordinate lines of each system. Otherwise—if the nodes are a tensor product grid associated with a sufficiently smooth reference coordinate system—the cell edges are third-order accurate. We append a natural two-dimensional blending of overlapping surface interpolants.

1 Introduction

Most numerical methods, including Finite-Element Methods (FEMs) and Finite-Difference Methods (FDMs), use a computational grid to solve partial differential equations. Typical grid generation methods (Knupp and Steinberg, 1993; George and Borouchaki, 1998) produce both the nodes of the grid (i.e., the coordinates of the grid-points) and the grid's structure (logically-rectangular, triangular, and so on). The structure of the grid tells us which pair of nodes is associated with an edge, which edges are associated with a face, and which faces are associated with a cell of the grid. But grid generators do not always specify the continua that connect the nodes to bound and separate the cells of the computation. Yet, edge lengths, facial areas, or cell volumes are often needed when approximating differential equations.

The simplest approach to these continua is to connect nodes by segments of straight lines. This is the most common approach for FDMs in two dimensions. The situation is more complicated in three dimensions. Consider a logically-cuboid grid, and connect the appropriate nodes by segments of straight lines to form the edges. The four edges associated with a face do not usually lie in the same plane, and there are a couple choices for surfaces that span such edges. Depending on this choice we will have different formulas for facial areas and cell volumes. The simplest approach in practice is to use the tri-linear map from the unit reference cube to a cell of the computational grid. This map retains the line segments as edges, while the faces become ruled surfaces, see, e.g., (Dukowicz, 1988).

Introducing curvilinear grids at curvilinear boundaries or interfaces can increase the accuracy of a simulation, as is well known from the FEM literature (Ciarlet and Raviart, 1972; Huyakorn and Andersen, 1986; Zlamal, 1973 a; Zlamal, 1973 b). A different goal motivated the curvilinear grid that underlies the difference approximations for the equations of Lagrangian gas-dynamics in (Margolin and Shashkov, 1998): namely, the preservation of the physical symmetry of the solution. For example, the ability to represent and preserve spherical symmetry in numerical simulations of implosion is critically important in the inertial confinement fusion program, (Town and Bell, 1991), where small departures from spherical symmetry due to discrete errors may be amplified by Rayleigh-Taylor instability, leading to unacceptably large errors in problems with strong compressions. Also, uncertainty as to whether a non-symmetric result is due to numerical errors or to the physical design inhibits our understanding of the dynamics of the implosion and so severely limits our predictive capabilities.

Curvilinear grids used for the construction of symmetry-preserving FDMs in two dimensions must satisfy some particular requirements: if points from the set corresponding to a logical direction belong to a straight line or a circle, then the edge-reconstruction algorithm must reproduce those curves. In (Margolin and Shashkov, 1998) a simple two-dimensional method that satisfies these requirements is described: points are connected either by a straight line segment or by the arc of some circle. But the tangent to the resulting curvilinear grid lines can be discontinuous at nodes—which can cause some problems in FDMs. Therefore, it is desirable to use a reconstruction algorithm that produces curves having continuously changing tangent lines. When developing symmetry-preserving methods for three dimensions, there are additional requirements for the reconstruction of surfaces: certain families of planes, spheres, cylinders and cones need to be reproduced; and we'd like unit surface normals to vary smoothly.

In this paper we describe a method for generating curves and surfaces in three dimensions that has the following properties:

- We obtain continua that interpolate the nodes and are G^1 continuous; i.e., tangent lines and planes vary continuously (although tangential derivatives can jump at nodes or cell edges)
- When given the nodes associated with a tensor product set of discrete locations on the coordinate lines of rectangular, polar, cylindrical, or spherical coordinates, the interpolants reproduce
  - the appropriate lines segments and circular arcs for the curve interpolant
  - the appropriate planes, spheres, cylinders and cones for the surface interpolant.

Concerning related work: we were inspired by the use that (Hill et al., 1990; Hall and Hibbs, 1990) made of Overhauser parabolic patches (Overhauser, 1968); see also, e.g., (Brewer and Anderson, 1977).

Overhauser blending has been successfully incorporated in boundary element methods in, e.g., (Hall and Hibbs, 1987; Hall and Hibbs, 1988); while (Sladek and Sladek, 1995) presents a deficiency of this approach related to normal vectors. A technique similar to Overhauser parabolic blending was used in (Kim and Nam, 1995) to blend circular quaternion curves. In two dimensions our way of blending two successive circles is similar to (Wenz, 1996)—but our choice of the circles determined by three successive points improves our space-curve approximation accuracy to third-order.

The paper is organized as follows. In section 2 we present the algorithm for our local blend of two successive circular interpolants. Section 3 describes the properties of the resulting curves, and section 4 presents several examples. Sections 5, 6 and 7 describe an algorithm, properties and examples for our surface interpolant—which is obtained using blended local circular nodal interpolation to form cell edges or "wires", followed by similar interpolation of the wires to define a surface interpolant. Appendix I provides proof of the third-order approximation of space curves by successive three-point circular interpolants and their blendings. Appendix II proposes a natural two-dimensional blend of four overlapping surface interpolants of four points.

2 Blending Successive Circular Interpolants

The curve we construct connects all but the first and last of a given set of nodes because our algorithm requires an additional point at each end. Specifically, given an ordered set of $n + 2$ distinct points $p_0, p_1, \ldots, p_n, p_{n+1}$, we shall define a curve C going through $p_1, \ldots, p_n$. Two successive circular interpolants—of the two overlapping triples of four successive points from this sequence—are blended between the middle pair to yield the segment of C between that pair. A similar blending in (Wenz, 1996)—but of different locally defined circles—is said there to yield lower-order accuracy. For completeness we describe the whole algorithm:

Algorithm 1 (*Construction of the curve interpolant*)

1. *For i:=1 to n-1 do (construct the interpolant between p_i and p_{i+1})*
   - (a) *determine the circle or line segment C_i^L going through the points p_{i-1}, p_i, p_{i+1}*
   - (b) *determine the circle or line segment C_i^R going through the points p_i, p_{i+1}, p_{i+2}*
   - (c) *between p_i and p_{i+1}, blend the two segments C_i^L and C_i^R into the resulting curve C_i, doing this in a manner ensuring that the tangents of C_i and C_i^L are the same at the point p_i, and the tangents of C_i and C_i^R are same at the point p_{i+1}*

2. The union of all such curve segments C_i defines the resulting interpolating curve C

Regarded as geometric objects, $C_{i+1}^L = C_i^R$. We use the algorithm both in two and three dimensions. If the three points are collinear we construct the line segment. If not, we construct the circle going through the points—but in three dimensions we first construct the plane going through the points.

The blending in step *1 (c)* of the algorithm requires a parameterization of circular/line segments. The line segment L between the points p_i and p_{i+1} is parameterized by $L(t) := (1-t)p_i + tp_{i+1}, t \in [0,1]$, so that $L(0) = p_i$ and $L(1) = p_{i+1}$. The circular segment c with center $a = (x_c, y_c)$ and radius r is parameterized between p_i and p_{i+1} as $c(t) = (x(t), y(t))$, where

$$x(t) \quad = \quad r\sin(t\varphi - \vartheta) + x_c, \quad y(t) = r\cos(t\varphi - \vartheta) + y_c, \qquad t \in [0,1].$$

Here, $\varphi = \angle(p_i, a, p_{i+1})$, and ϑ is a rotational angle chosen so that $C(0) = p_i$ and $C(1) = p_{i+1}$. In three dimensions the parameterization is the same in principle, but necessarily involves the directional cosines for rotating a plane determined by three points in the coordinate frame.

As in (Wenz, 1996) we form the Overhauser-type blend C_i of our parameterized segments C_i^L and C_i^R, obtaining the curve-segment

$$C_i(t) = (1-t)\, C_i^L(t) + t\, C_i^R(t), \qquad t \in [0,1]. \tag{1}$$

An (extreme) example of such blending is shown in Fig. 1.

Finally, the curve interpolant $C(t)$, $t \in [1, n]$ is obtained by adjoining the segments C_i so that

$$C(t) = C_i(t-i), \quad t \in [i, i+1], \quad i = 1, \ldots, n-1.$$

3 Properties of the Blend of Successive Circular Interpolants

We list the properties of our blend of successive circular interpolants.

Locality: The segment of the interpolant between the points p_i and p_{i+1} depends only on the four points p_{i-1}, p_i, p_{i+1}, and p_{i+2}.

Exactness for straight lines: If four points $p_{i-1}, p_i, p_{i+1}, p_{i+2}$ lie linearly ordered on a line, then the interpolant between the points p_i, p_{i+1} is the segment of the line going through these two points. If all points $p_0, p_1, \ldots, p_n, p_{n+1}$ lie linearly ordered on a line, then the whole interpolant is a segment of that line.

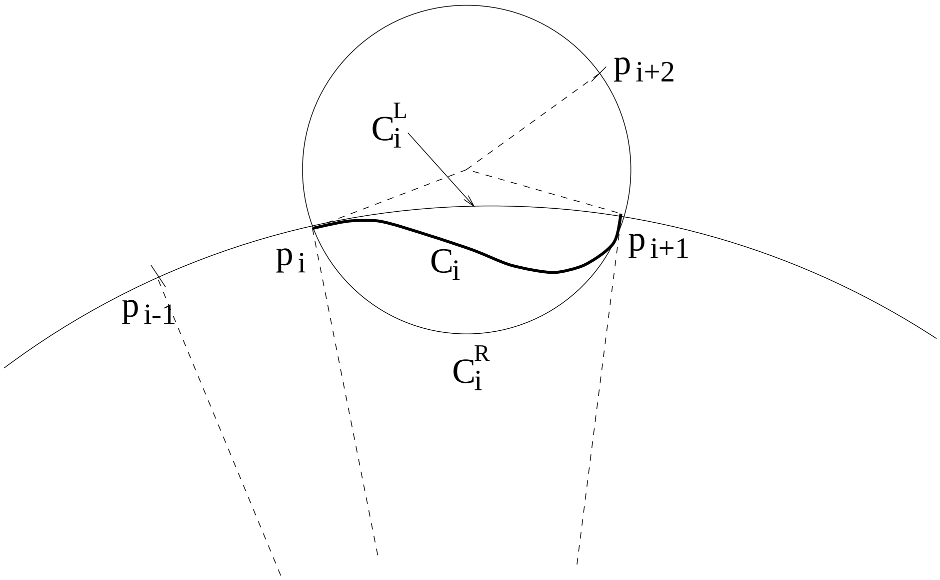


Figure 1: The successive interpolating arcs C_i^L, C_i^R, with their blend C_i in bold.

Exactness for circles: If four points $p_{i-1}, p_i, p_{i+1}, p_{i+2}$ lie linearly ordered on a circle then the interpolant between the points p_i, p_{i+1} is the corresponding arc of that circle. If all points $p_0, p_1, \ldots, p_n, p_{n+1}$ lie linearly ordered on a circle then the whole interpolant is a segment of that circle. In contrast to (Wenz, 1996) we do not need equidistant points for this property.

Exactness for Cartesian, polar, cylindrical, and spherical grid edges: A two-dimensional (2D) grid is given by points p_{ij}; and a 3D grid, by points p_{ijk}. If our set of interpolated points $p_0, p_1, \ldots, p_n, p_{n+1}$ is a subset of a 2D or 3D grid that consists of the images in Euclidean space of consecutive points along a line parallel to an axis of some set of spherical, cylindrical, polar, or Cartesian coordinates; then our interpolant coincides with the image in Euclidean space of that line of coordinates. Thus, we can exactly reconstruct by our interpolation the Euclidean image of the edges of a 2D or 3D tensor-product set of grid cells whose faces, in such curvilinear coordinates, are all parallel to the appropriate coordinate planes. As with the previous properties this one is a local property: if we have a grid of one of the above types which is exact in some region but deformed in the rest, then our interpolant reconstructs the grid edges in the interior of that region exactly.

Approximation: If the points $(p_i)_0^n$ lie successively on a thrice-differentiable space curve p, then the both blend and the local circular interpolants themselves are third-order $O(h^3)$ accurate as an approximation to p, where h is the maximum of the three successive secant distances between the four points of local data. See Appendix I for details.. In particular, the Euclidean distance $\|p(t) - C_i(t)\|$ is $O(h^3)$ in size—here the point $p(t)$ is the relative distance t from p_i to p_{i+1} along p.

4 Examples of Blends of Successive Circular Interpolants

For comparison we present in Fig. 2 our versions of all the examples from (Wenz, 1996)—which contains the relevant data, too. Originally, the data for Example (a) was from (Lee, 1989); Example (b), from (Carlson and Fritsch, 1980); and Examples (c) and (d), from (Irvine et al., 1986). In all these examples we have imposed a "natural" boundary condition by adjoining a (fictional) endpoint to each end of the given data, namely, $p_0 := p_1 - (p_2 - p_1)$ and $p_{n+1} := p_n + (p_n - p_{n-1})$. This has the effect that the curve's tangent line at each (real) endpoint also passes through the data point adjacent to it.

To show examples from our application area we present, in Fig. 3, two-dimensional non-uniform Cartesian and polar grids, non-deformed and deformed, where we used our curve interpolant to connect the nodes. The interior nodes of the deformed grids are randomly moved from their non-deformed locations. The edges of the non-deformed grids are reconstructed exactly—and edges of the deformed grids form G^1 continuous curves—as we sought.

Fig. 4 shows the edges of part of a uniform (a), and a deformed-uniform (b), 3D Cartesian grid as constructed by our curve interpolant. The edges delineate four grid cells. The deformation is done by random shift of the eight nodes corresponding to the closest cell, while the remaining ten points remain fixed in the original uniform Cartesian grid.

Fig. 5 shows the edges of a part of a uniform (a), and deformed uniform (b), 3D cylindrical grid as constructed by our curve interpolant. The edges delineate three grid cells. The deformation is done by random shift of the eight nodes corresponding to the middle cell, while the remaining eight points remain fixed in the original uniform cylindrical grid.

Fig. 6 shows the edges of a part of a uniform (a), and a deformed uniform (b), 3D spherical grid as constructed by our curve interpolant. The edges delineate four grid cells. The deformation is done by random shift of the eight nodes corresponding to the upper-right cell, while the remaining ten points remain fixed on the original uniform spherical grid.

5 Surface Interpolation via Blended Circular Interpolation

As in the case of the circular curve interpolant we shall avoid treatment of boundaries and define the nodal interpolation problem for surfaces as follows. Given a matrix of $(m+2)(n+2)$ points $p_{i,j}$; $\quad i =$

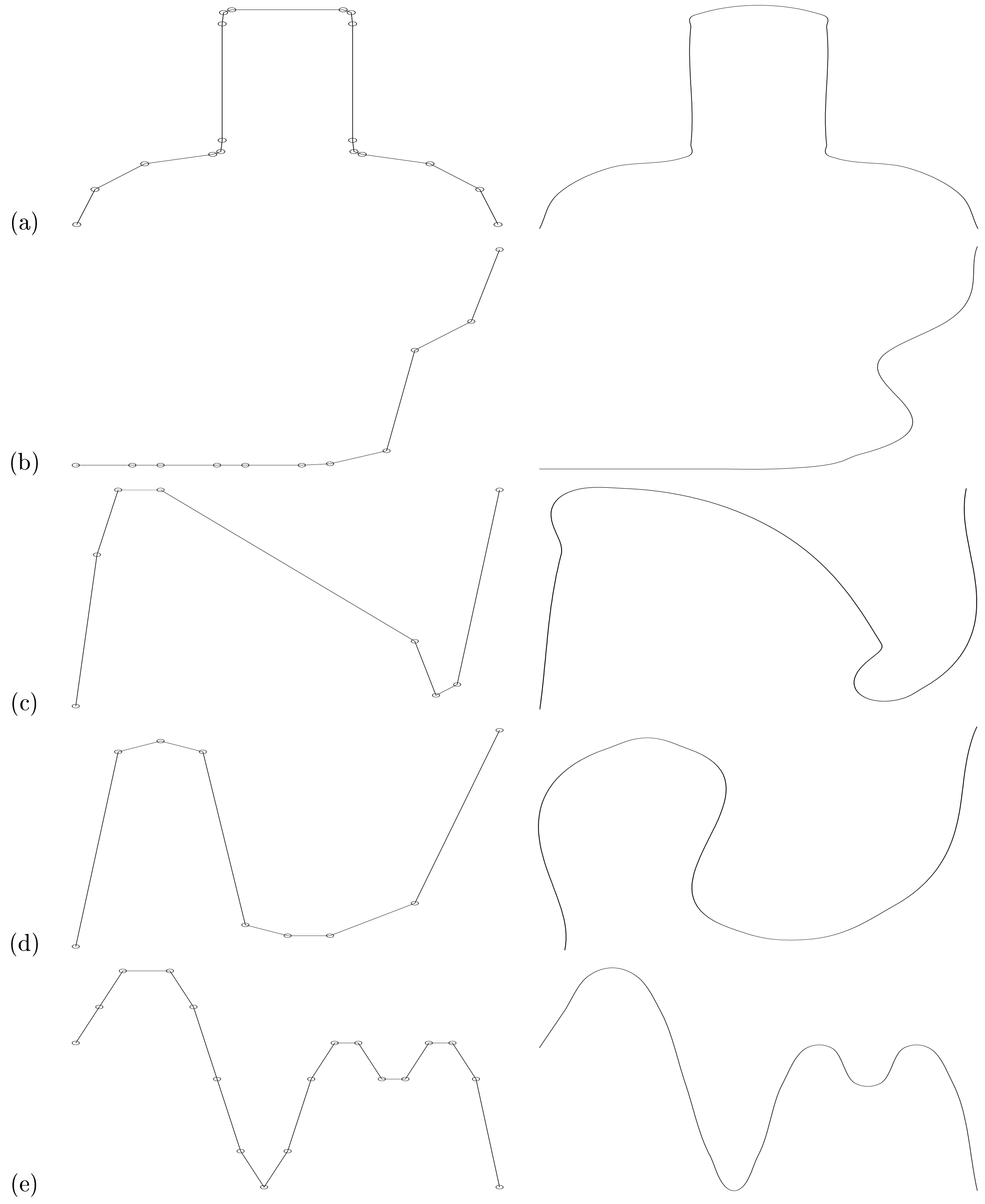


Figure 2: Examples from (Wenz,1996): (a) corresponds there to Fig. 2, (b) to Fig. 3, (c) to Fig. 4, (d) to Fig. 5, and (e) to Fig. 6. The data, linearly interpolated, are on the left; our interpolant is on the right.

$0, 1, \ldots, n, n+1; \quad j = 0, 1, \ldots, m, m+1$, we shall define a surface going through the points $p_{i,j}, \quad i = 1, \ldots, n; \quad j = 1, \ldots, m$. We do this by constructing our curve interpolant through each "column" of data points, thereby defining $m+2$ space curves or "wires"—and then interpolating corresponding points across

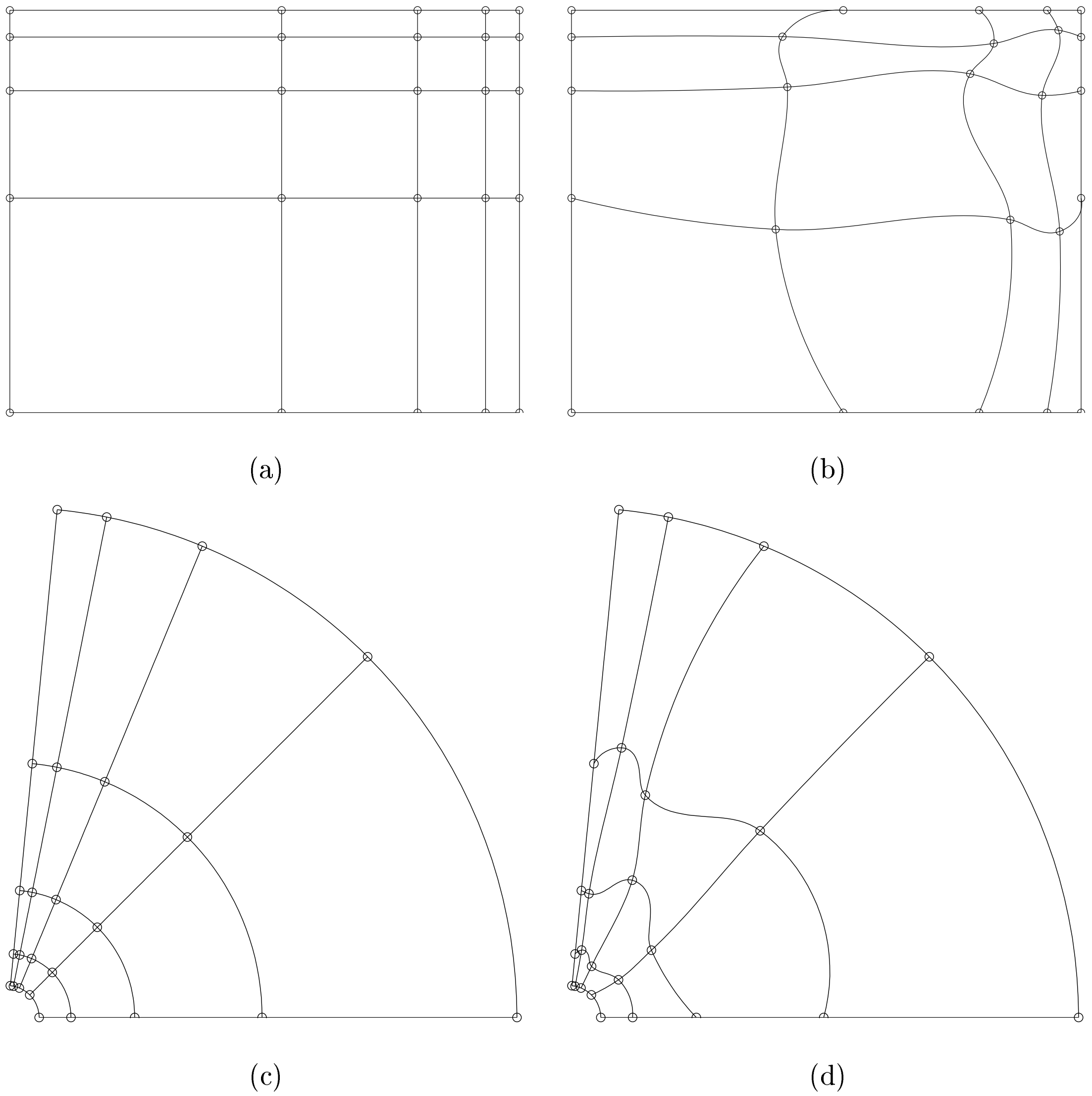


Figure 3: Our curve interpolant in two dimensions: cell edges of (a) a non-uniform Cartesian grid, (b) a deformed non-uniform Cartesian grid, (c) a non-uniform polar grid, (d) a deformed non-uniform polar grid.

the wires.

Algorithm 2 (*Construction of the surface interpolant*)

1. *For j:=0 to m+1 do*

   - *construct the curve interpolant* $I_j(t), t \in [1, n]$ *of the points* $p_{0,j}, p_{1,j}, \ldots, p_{n,j}, p_{n+1,j}$ *by using Algorithm 1*

2. *fix* $t = t_0 \in [1, n]$; *and construct the curve interpolant* $S_{t_0}(u), u \in [1, m]$ *of the points*

$$I_0(t_0), I_1(t_0), \ldots, I_m(t_0), I_{m+1}(t_0)$$

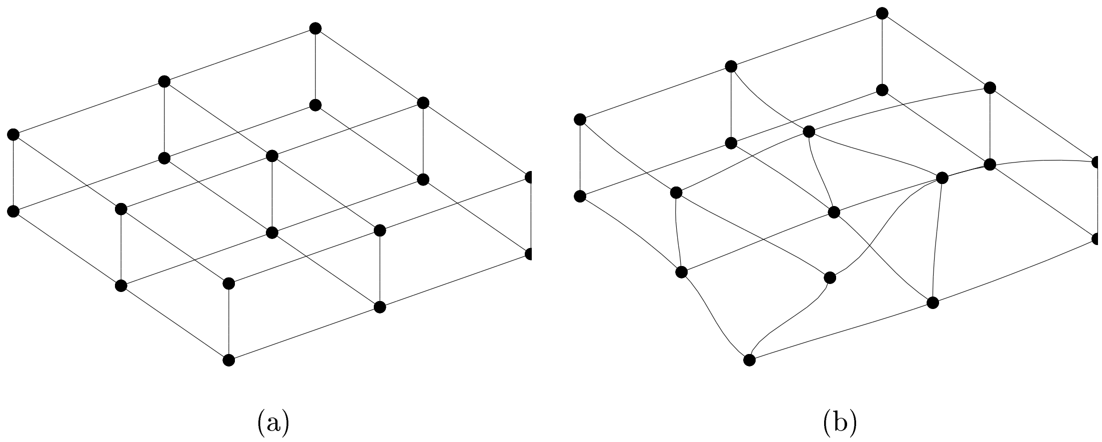


Figure 4: Cell edges of (a) a uniform Cartesian grid, and (b) a deformed uniform Cartesian grid via our curve interpolant in three dimensions.

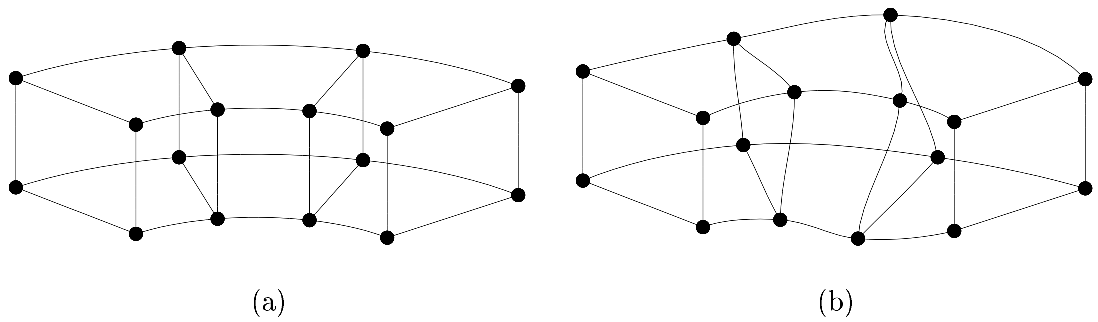
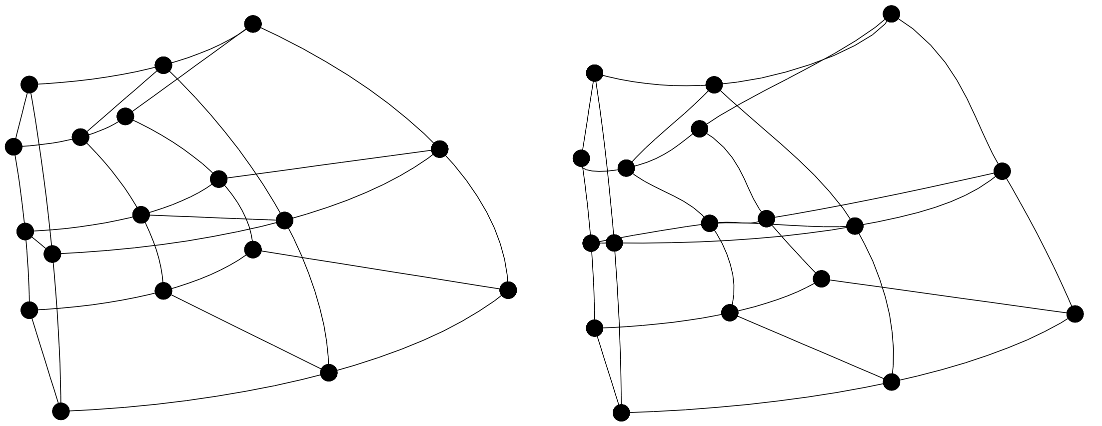


Figure 5: Cell edges of (a) a uniform cylindrical grid, and (b) a deformed uniform cylindrical grid via our curve interpolant in three dimensions.

by using Algorithm 1

3. *the surface S of blended circular interpolants is given by the parameterization $S(t,u) = S_t(u)$, $t \in [1,n]$, $u \in [1,m]$*

One can interpolate rows of nodes first instead of columns—that is, construct the initial curve interpolants (wires) through the points $p_{i,j}$ for fixed i (instead of fixed j as above). The resulting i-wires (explicitly produced by one process, implicitly by the other) coincide, as do the j-wires. However, the surfaces spanning the wires can differ in between the wires—according to our numerical experiments. In Appendix II we present a truly two-dimensional type of surface blending that can produce yet a third surface between the wires. The following properties hold for all these surfaces.



(a) (b)

Figure 6: Cell edges of (a) a uniform spherical grid, and (b) a deformed uniform spherical grid via our curve interpolant in three dimensions.

6 Properties of a Surface of Blended Successive Circular Interpolants

Here we list the properties of the surface of blended successive circular interpolants constructed by Algorithm 2.

Locality: The segment of the interpolant between the four points $p_{i,j}$, $i = I, I+1$, $j = J, J+1$ depends only on the 16 points $p_{i,j}$, $i = I-1, I, I+1, I+2$, $j = J-1, J, J+1, J+2$ for any I, J.

Exactness for the surfaces associated with a tensor product of data from Cartesian, cylindrical, or spherical coordinate lines: Suppose the 4×4 block of data being interpolated is the image in Euclidean space of the tensor product of four successive points each from two of the coordinate axes—and one fixed point from the third axis—of some set of spherical, cylindrical, or Cartesian coordinates. Then (a) the "middle segment" of each i-wire (and each j-wire) coincides with the Euclidean image of the appropriate coordinate line segment, and (b) the "middle square" of the surface interpolant coincides with the corresponding piece of the coordinate surface. In particular, the "middle square" reproduces appropriate sections of (a) spheres from a spherical coordinate tensor products of discrete longitude with discrete latitude data (and fixed radius), (b) circular cones from a spherical coordinate tensor product of longitude with radius (and fixed latitude), (c) circular cylinders from a cylindrical coordinate tensor product of longitude with z-axis data (and fixed radius), as well as (d) planes from the remaining tensor products of spherical, cylindrical and Cartesian coordinate lines. Consequently, given as nodes the image in

Euclidean space of a three-dimensional tensor product of discrete, not-necessarily-uniform set of spherical, cylindrical, or Cartesian coordinates; our surface interpolant reproduces the appropriate circular (or linear) edges that connect these nodes; and then spans them with the appropriate portions of spherical, conical, cylindrical, or planer surfaces. (Recall that a continuously dependent surface interpolant that handled these special cases exactly was the primary goal of this research.)

Approximation: The "wires" retain the third-order accuracy previously mentioned for our curve interpolant in three dimensions, but we have no proof of corresponding accuracy for the surface between wires.

Smoothness: The surface interpolant is G^1 continuous.

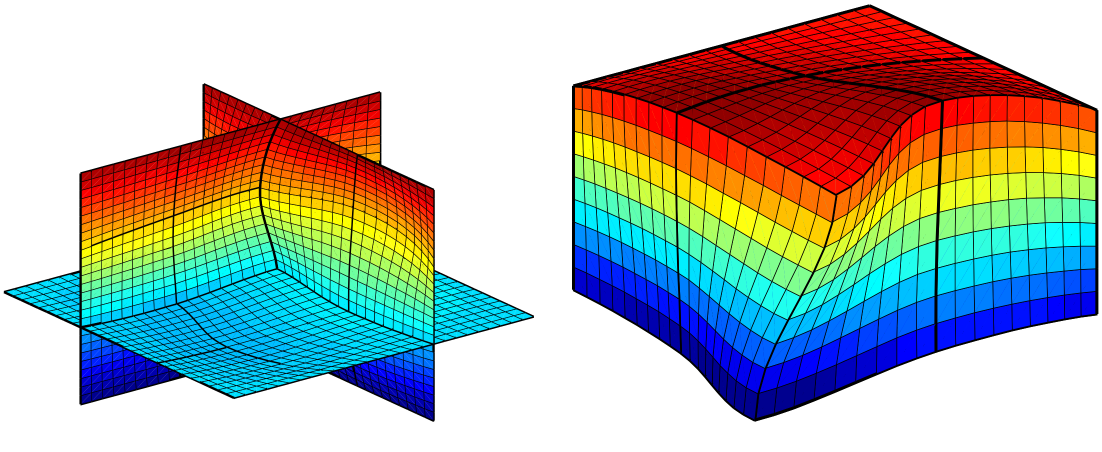
7 Examples of Surfaces of Blended Successive Circular Interpolants

As examples of our surface interpolant we construct the cell faces of uniform—and deformed uniform—Cartesian, cylindrical and spherical grids. This illustrates how the faces of uniform non-deformed coordinate grids are reconstructed exactly. The same is true for non-uniform non-deformed grids, but the results would be less apparent. The bold curves on the surfaces correspond to the grid edges that were shown earlier in Figs. 4 – 6.

Fig. 7 shows some grid faces of deformed uniform 3D Cartesian grids. Three deformed planes, corresponding to fixed indices of the grid, can be seen in Fig. 7(a). Fig. 7(b) shows the faces of four cells that we delineated earlier by their edges in Fig. 4(b).

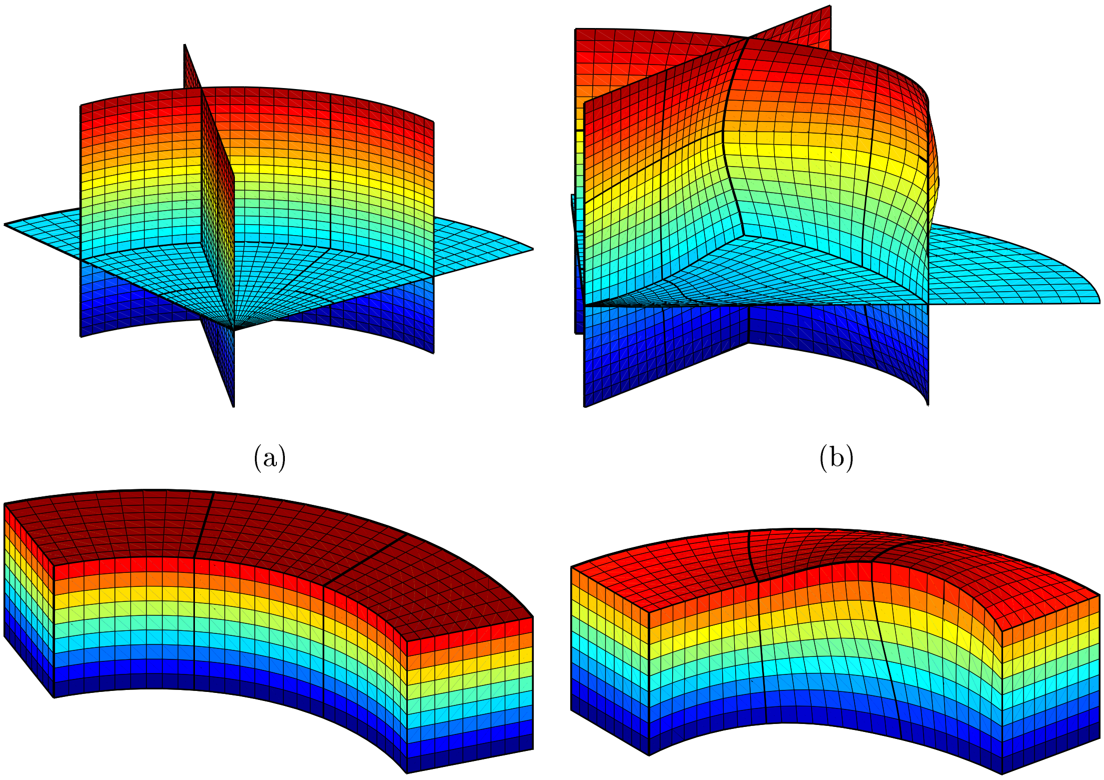
Fig. 8 shows some grid faces of a uniform, and a deformed uniform, cylindrical grid. Two planes and a cylinder, corresponding to fixed indices of the grid, can be seen in Fig. 8(a). Two deformed planes and a deformed cylinder are in Fig. 8(b). Fig. 8(c) shows the faces of the three cells of the non-deformed grid that we delineated earlier by their edges in Fig. 5(a). Fig. 8(d) shows the faces of the three cells of the deformed grid that we delineated earlier by their edges in Fig. 5(b).

Fig. 9 shows some grid faces of a uniform, and a deformed uniform, spherical grid. A sphere and a cone, corresponding to fixed indices of the grid, can be seen on Fig. 9(a). A deformed sphere and a deformed cone are in Fig. 9(b). Fig. 9(c) shows the faces of the four cells of the non-deformed grid that we delineated earlier by their edges in Fig. 6(a) (from different view-point). Fig. 9(d) shows the faces of the three cells of the deformed that we delineated earlier by their edges in Fig. 6(b) (from different view-point).



(a) (b)

Figure 7: Cell faces of deformed uniform Cartesian grids using our surface interpolant: (a) three deformed planes, (b) four deformed cubes, cf. Fig. 4(b).



(a) (b)

(c) (d)

Figure 8: Cell faces of uniform cylindrical grids (a),(c)—and the same grids deformed (b),(d)—using our surface interpolant: (a) a cylinder and two planes, (b) a deformed cylinder and two deformed planes, (c) three cells of a uniform cylindrical grid, (d) three cells of a deformed uniform cylindrical grid; cf. Fig. 5.

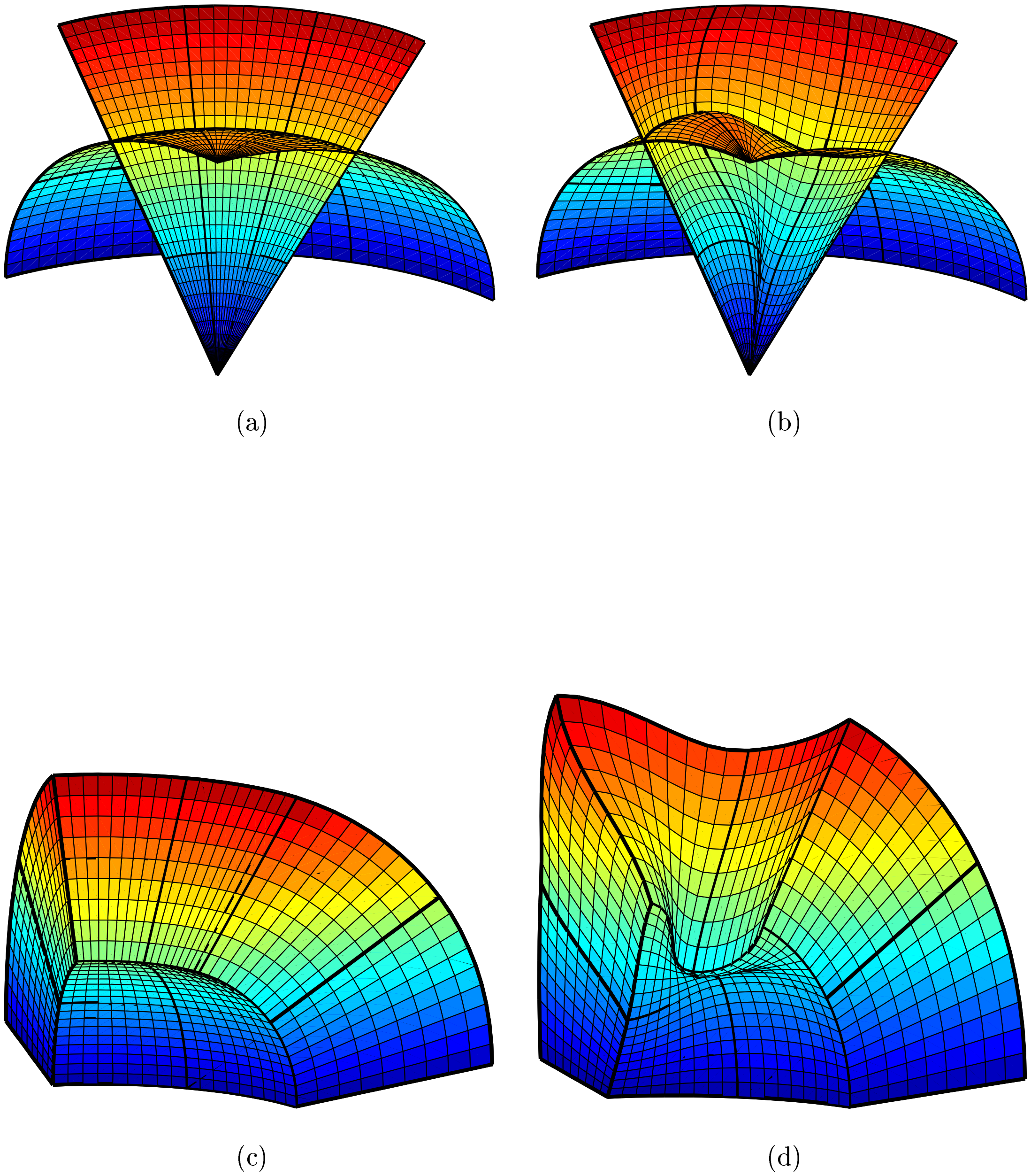


Figure 9: Cell faces of uniform spherical grids (a),(c)—and the same grids deformed (b),(d)—using our surface interpolant: (a) a sphere and a cone, (b) a deformed sphere and a deformed cone, (c) four cells of a uniform spherical grid, (d) four cells of a deformed uniform spherical grid; cf. Fig. 6.

8 Acknowledgments

This work was performed under the auspices of the US Department of Energy under contract W-7405-ENG-36 and the DOE/BES Program in the Mathematical, Information, and Computational Sciences

(MICS) Contract KC-07-01-01. The last two authors also acknowledge the support of DOE's Accelerated Strategic Computing Initiative (ASCI).

R. Liska was supported also in part by the National Science Foundation grant CCR-9531828, by the Ministry of Education of Czech Republic program Kontakt (Project ME 050 (1997)) and would like to thank the Institute for Geophysics and Planetary Physics of the Los Alamos National Laboratory for hosting his visit at Los Alamos.

We appreciate fruitful conversation with J. M. Hyman, L. Margolin, and P. Knupp.

Appendix I: Third-order accuracy for the relatively linear, Overhauser-type blend of two overlapping circular interpolants of a smooth space curve

Proposition. (*Approximation by blended circular interpolants*). *The two successive circular interpolants C^L and C^R of the two overlapping triples of four successive points $(p_i)_0^3$ taken along a thrice-differentiable space curve p are blended between the middle pair of points p_1 and p_2. The (relatively) linear blending used—of Overhauser type—constructs a curve C connecting p_1 to p_2. C is defined by the parameterization*

$$C(t) := (1-t)\,C^L(t) + t\,C^R(t), \qquad t \in [0,1],$$

in which $C^L(t)$ is the point a relative distance t along C^L from p_1 to p_2, and $C^R(t)$ is the point a relative distance t along C^R from p_1 to p_2. Like C^L and C^R, the accuracy of the curve C is third-order $O(h^3)$ as an approximation to p between p_1 and p_2. Here h is the size of the largest of the successive secant distances between the four successive points. More generally: any blending

$$(1-w(t))\,C^L(t) + w(t)\,C^R(t),$$

with w some increasing map of $[0,1]$ onto itself, is also third-order accurate. For example: if all derivatives of w vanish at 0 and at 1, then the union of successive blends associated with a sequence of points along p is an infinitely differentiable curve approximating p to within $O(h^3)$. As normals of non-planer space curves are not easily estimated in $o(1)$ neighborhoods of zeroes of the curvature, our error estimates are not uniform there unless this $o(1)$ is, in fact, $O(h)$.

The proof goes roughly as follows: Each circle is third-order accurate between the three successive points on p that it interpolates. So, the two successive circles are each third-order accurate between the common pair they interpolate. The arc-length distances between these two points—along p or either of the circles—are within third-order of each other. Consequently, any convex combination of two points that lie at the same relative distance along the two circles from the common data points is within third order of p at that same relative distance along p.

We devote most of our effort to plane curves. To prove the Proposition in this case, we first develop certain properties of three-point circular interpolation; for omitted details see, e.g., (Mjolsness and Swartz, 1987, esp. pp. 217–219). Existence of a circle c interpolating three successive points p_0, p_1, p_2 along the curve p is ensured. We now consider the local approximation of p by c. We shall prove further on that the space curves of the Proposition are equally well approximated.

Lemma. (*Approximation by three-point circular interpolation*). *Let h be small enough. Then the orthogonal projection Π of the segment of the curve p between p_0 and p_2 onto the secant line-segment $\overline{p_0p_2}$ is one-to-one. For each point P in this segment of p, there is a point c_P on the circle c that also projects onto $\Pi(P)$. We compare the two curves p and c—and their derivative(s) $f' := df/ds$ with respect to arc-length s—at these two points. This yields the error estimates (measured in the Euclidean distance $\| \cdot \|$)*

$$\|P - c_P\| = O(h^3), \quad \|p'(P) - c'(c_P)\| = O(h^2), \quad \|p''(P) - c''(c_P)\| = O(h). \tag{1}$$

Proof: Let the x-axis contain the secant segment $\overline{p_0p_2}$. We can and shall assume that h is small enough so that the x-coordinate mappings—which (orthogonally) project both p and c onto $\overline{p_0p_2}$—are one-to-one. Even more: we assume that x is not stationary on $\overline{p_0p_2}$, i.e., $x' := dx/ds$ is bounded away from zero on the relevant parts of c, uniformly for h small enough. Let $Y(x)$ be the y-coordinate of the point on p whose x-coordinate is x; and, similarly, let $y(x)$ be the y-coordinate for points on the circle c. The third derivative of Y is continuous by assumption.

Moreover, the third derivative of y is bounded uniformly for h small enough. For this we first note that $\dot{y} := dy/dx = y'/x'$ is bounded since $(x')^2 + (y')^2 \equiv 1$ on the relevant part of c, and x' is bounded away from zero there. Next: $\ddot{y} = (x'\,y'' - y'\,x'')/(x')^3$ is bounded, since the curvature $\hat{\kappa}$ of c converges to the local curvature of p (according, e.g., to the analysis leading to eqn. (2.8) in (Mjolsness and Swartz, 1987, pp. 217–219), and since $(x'')^2 + (y'')^2 \equiv \hat{\kappa}^2$. Similarly, $\dddot{y}$ is uniformly bounded—use $(x''')^2 + (y''')^2 \equiv \hat{\kappa}^4$.

We are now allowed the bounds

$$e(x) := Y(x) - y(x) = O(h^3), \quad \dot{e}(x) = O(h^2), \quad \text{and} \quad \ddot{e}(x) = O(h) \tag{2}$$

for x along the segment $\overline{p_0p_2}$. The proof is a typical one using Rolle's theorem, namely: e vanishes at the projections x_i of the three p_i onto the secant segment. Hence its first derivative vanishes at two distinct points $a_{1/2}$ and $a_{3/2}$ strictly between x_0 and x_2. So its second derivative vanishes at some interior point b. Integrating $\dddot{e}$ once from b, and then this integral from (say) $a_{1/2}$, and then *this* integral from one of the x_i, the boundedness of $\dddot{e}$ proves (2). The Lemma now follows using $e' = \ddot{e}\, x'$ and $e'' = \ddot{e}\,(x')^2 + \dot{e}\, x''$. $\square$

The (relatively) linear blending of the Proposition's two successive circular interpolants uses the relative distance between pairs of data points along the two circles, and we shall need to know

Corollary. *The point p_x on the plane curve p, and the point c_x on the three-point interpolating circle c, that each project onto the same x-coordinate along the secant segment $\overline{p_0p_2}$, lie at the same (to within $O(h^3)$) locations relative to the distances from p_0 to p_1 along their respective arcs.*

Proof. The relevant arc-lengths from p_0 along p and c, respectively, are

$$s_p(x) := \int_{x_0}^{x} (1+\dot{Y}^2)^{1/2}\, d\xi \quad \text{and} \quad s_c(x) := \int_{x_0}^{x} (1+\dot{y}^2)^{1/2}\, d\xi.$$

Using (2),

$$\begin{aligned} |(s_p - s_c)(x)| &= O\left(h^2\, |x - x_0|\right) \\ &= O\left(h^2\, |s_p(x)|\right) = O\left(h^2\, |s_c(x)|\right). \end{aligned}$$

Thus, with $t_p(x) := s_p(x)/(\Delta s)_p$ the relative distance that p_x is from p_0 using arc-lengths along the curve p, and with $t_c(x) := s_c(x)/(\Delta s)_c$ similarly the relative distance that c_x is from p_0 along the interpolating circle c,

$$|(s_p - s_c)(x)| = O(h^3\, t_p(x)) = O(h^3\, t_c(x))\,. \quad \square \tag{3}$$

Continuing with the Proposition for plane curves p: A given $t \in [0,1]$ determines (a) the point $P = P(t)$ on p that is a relative distance t from p_1 to p_2 along p, (b) an associated $x_1 = x_1(t)$ from the projection of P onto $\overline{p_0p_2}$, and thereby (c) an associated point $C^L(x_1)$ on the circle C^L having the same projection x_1 that P does. Similarly, we have an x_2 associated with the projection of (the same) $P(t)$ onto $\overline{p_1p_3}$, and thereby an associated point $C^R(x_2)$ on C^R. From the Lemma we conclude

$$\|P(t) - C^L(x_1)\| = O(h^3) = \|P(t) - C^R(x_2)\|.$$

But $C^L(x_1)$ and $C^R(x_2)$ are not quite at the necessary locations t relative to their respective arcs. Let $C^L(t) \in C^L$ be located at the relative distance t from p_1 to p_2 along C^L; and analogously, for $C^R(t) \in C^R$, along C^R. From the Corollary (especially (A.3)) and the boundedness of $\|dC^L/ds\|$ and $\|dC^R/ds\|$ on the relevant intervals, we have

$$\|C^L(x_1) - C^L(t)\| = O(h^3) = \|C^R(x_2) - C^R(t)\|.$$

Then, for arbitrary $w \in [0,1]$, we conclude the error estimate for the "w-weighted blend of $C^L(t)$ and $C^R(t)$", namely,

$$\begin{aligned} \|P(t) - [(1-w)\,C^L(t) + w\,C^R(t)]\,\| &\leq (1-w)\,\|P(t) - C^L(t)\| + w\,\|P(t) - C^R(t)\| \\ &\leq (1-w)\,\|P(t) - C^L(x_1)\| + w\,\|P(t) - C^R(x_2)\| \\ &\quad +(1-w)\,\|C^L(x_1) - C^L(t)\| + w\,\|C^R(x_2) - C^R(t)\| \\ &\leq O(h^3). \end{aligned}$$

The Proposition—for plane curves p—follows immediately.

Third-order approximation to curves p in three dimensions is now relatively easy to prove. First: the circular interpolant c of three successive points p_0, p_1, p_2 along p is $O(h^3)$ accurate in any $O(h)$-size neighborhood along p in which its curvature is not $o(1)$. Our proof will make repeated use of the fact that, while smooth curves are locally linear to second order $O(h^2)$, they are locally planer to third-order [see, e.g., (Struik, 1950, Chap. 1, esp. §6), or (Boehm, 1990)]. Indeed, both the curve and the p_i lie in an $O(h)$ by $O(h^2)$ by $O(h^3)$ rectangular box in the local tangential/normal/binormal directions. Secant lines are $O(h^2)$ accurate, and (in the difference notation of (Mjolsness and Swartz, 1987)) the approximate unit tangent $t_\Delta := \Delta p/\Delta t$ is $O(h)$ accurate. Since the local curvature of p is not $o(1)$, the approximate unit normal n_Δ—constructed using t_Δ and $\Delta^2 p/\Delta t^2$, and spanning (with t_Δ) the "secant plane" $\mathcal{P}_\Delta$ that contains the interpolating circle c—converges to the local normal vector of p. Therefore, the approximate binormal $b_\Delta := t_\Delta \times n_\Delta$, converges to the local binormal, too. It follows that the (orthogonal) projection $\Pi_\mathcal{P}(p)$ of p onto $\mathcal{P}_\Delta$ has a uniformly bounded third derivative as $h \to 0$. From our Lemma about the circular interpolation of smooth plane curves,

$$|||c - \Pi_\mathcal{P}(p)||| = O(h^3), \quad |||(c - \Pi_\mathcal{P}(p))'||| = O(h^2), \quad |||(c - \Pi_\mathcal{P}(p))''||| = O(h) \tag{4}$$

between p_0 and p_2—here, $|||A - B|||$ is the (Hausdorff) distance between the point-sets A and B. Next: Choose coordinates x along $\overline{p_0 p_2}$, y along n_Δ, and z along b_Δ. With $Y(x)$ (respectively, $y(x)$) the y-coordinate of p (resp., $\Pi_\mathcal{P}(p)$), and since $Y - y$ has three distinct zeroes on $\overline{p_0 p_2}$, Rolle's Theorem applies as it did for our Lemma. The same argument applies to $Z - z$. So the curve and its projection are suitably close:

$$|||\Pi_\mathcal{P}(p) - p||| = O(h^3), \quad |||(\Pi_\mathcal{P}(p) - p)'||| = O(h^2), \quad |||(\Pi_\mathcal{P}(p) - p)''||| = O(h).$$

With (4) we conclude—as promised—that the $O(h^3)$ error bounds (1) hold for space curves p, too. Finally: with a proof much as in two-dimensions, if two successive circular interpolants (of four successive points

p_0, p_1, p_2, p_3 from p) are averaged at the same relative distance between p_1 and p_2, the resulting curve approximates p (at that same relative distance) to within $O(h^3)$—even though the planes of the two circles may have only the line through $\overline{p_1p_2}$ in common.

This almost completes proof of the Proposition for space curves p. There remains the caveat concerning small curvatures: Since $|Y(x)| = O(\kappa h^2) = |y(x)|$ for h, $\kappa \to 0$, we see that $\kappa = O(h)$ does not alter $O(h^3)$ error bounds—in spite of ill-determined normals n_Δ and b_Δ in that case. The case $\kappa = o(1)$ but $\kappa \neq O(h)$ may require more delicacy, and we have not pursued it further. As a practical matter, however, we have only lost the uniformity of our error estimates in $o(1)$ neighborhoods of a zero of κ. $\square$

Appendix II. A natural two-dimensional blending of overlapping surface interpolants

Consider a 4×4 pile of data consisting of points in three dimensions. Suppose $S_{00} = S_{00}(u, v)$, $(u, v) \in [-1, 1] \times [-1, 1]$, is a surface interpolating the 3×3 subset in the "lower left" corner (of the 4×4 pile) at the appropriate integral values of u and v. And suppose $S_{01} = S_{01}(u, v)$, $(u, v) \in [-1, 1] \times [0, 2]$ similarly interpolates the "upper left" 3×3 pile. And, similarly, for the lower right and upper right 3×3 piles of data. Then, if $U = U(u)$ and $V = V(v)$ are increasing maps of $[0, 1]$ onto itself, then the blended surface (defined for $(u, v) \in [0, 1] \times [0, 1]$) given by

$$S := (1 - U)(1 - V)\, S_{00} + U(1 - V)\, S_{10} + (1 - U)V\, S_{01} + UV\, S_{11}$$

also interpolates the middle 2×2 pile of data. Moreover, if surfaces spanning each successive 3×3 subset of $M \times N$ data points are prescribed, then similar blending, by defining surfaces spanning neighboring 2×2 piles of data, constructs a single-valued surface, continuous across cell boundaries (u or v integers), that interpolates all the data. More than this: if the six surfaces S_{ij} relevant to a boundary of the middle cell coincide on that boundary, e.g., if $S_{ij}(u, 0) \equiv S_{mn}(u, 0)$ for the six relevant pairs of indices; then the tangent plane to the blend also varies continuously across that boundary; i.e., the surface is G^1 there. (Actually, this remains true if the two appropriate triples of the six surfaces each has this property.)

Example: Such S_{ij} can be constructed as we now describe for S_{01}: let "horizontal wires", or "j-wires", be constructed for $j = 0, 1, 2$ and $u \in [-1, 1]$ by circular interpolation of the (i, j)-data, $i = -1, 0, 1$; here (the fractional part of) u represents the relative distance along each circular arc between the data "horizontally", i.e., from i to $i+1$. Then, let A_{01} be the surface defined for each $u \in [-1, 1]$ by interpolating with a circle the three points on the three j-wires associated with that u-value—here (the fractional part of) $v \in [0,2]$ is associated with the relative distance along this circle between the j-wires. In the same way, define the surface B_{01} by forming three circular "i-wires", $i = 0,1,2$, along each of which $v \in [0,2]$ varies; then, for each such v, interpolate the three i-wires with a circle—along which u varies from -1 to 1. Finally, set $S_{01} := (A_{01} + B_{01})/2$. Then, if we set $U(u) := u$ and $V(v) := v$, the surface S is the average of the two possible surfaces we used to interpolate an $I \times J$ set of nodes—the one interpolating the i-wires, and the other interpolating the j-wires.

References

[1] Boehm, W. (1990), Differential Geometry I, pp. 173–183, in: Farin, G.E., *Curves and Surfaces for Computer Aided Geometric Design*, Second Edition, Academic Press, New York.

[2] Brewer, J.A. and Anderson, D.C. (1977), Visual interaction with Overhauser curves and surfaces, Computer Graphics 11, 132–137.

[3] Carlson, R.E. and Fritsch, F.N. (1980), Monotone piecewise cubic interpolation, SIAM J. Num. Anal. 17, 238–246.

[4] Ciarlet, P. G. and Raviart, P.-A. (1972), Interpolation Theory over Curved Elements, with Applications to Finite Element Methods, Computer Methods in Applied Mechanics and Engineering 1, 217–249.

[5] Dukowicz, J. K. (1988) Efficient Volume Computation for Three-Dimensional Hexahedral Cells, J. Comput. Phys. 74, 493–496.

[6] George, P.-L. and H. Borouchaki, H. (1998), *Delaunay Triangulation and Meshing*, Hermes, Paris.

[7] Hall, W. S. and Hibbs, T.T. (1987), The treatment of singularities and the application of the Overhauser C^1 continuous quadrilateral boundary element to three dimensional elastostatics, in: Cruse, T.A. ed., *Advanced boundary element methods, Proceedings of the IUTAM Symposium, San Antonio, Texas, April 13-16,1987*, 135–143, Springer-Verlag, Berlin, New York.

[8] Hall, W.S. and Hibbs, T.T. (1988), C^1 continuous, quadrilateral boundary elements applied to three dimensions, in: Brebbia, C.A. ed., *Boundary elements X, Vol.2: Heat Transfer, Fluid Flow and Electrical Applications, 10th International Conference on Boundary Element Methods, Southampton, England*, 181–192, Computational Mechanics Publications; Springer-Verlag. Southampton, Boston; Berlin, New York.

[9] Hall, W.S. and Hibbs, T.T. (1990), Continuous, quadrilateral, and triangular surface patches, in: Creasy, C.F.M. and Craggs, C., editors, *Applied surface modeling*, 129–138, E. Horwood, New York.

[10] Hill, D.J.M., Fletcher, E.J., Moscardini, A.O. and Wilkinson T.S. (1990), Overhauser patches for generally curved surfaces, in: Creasy, C.F.M. and Craggs, C. eds., *Applied surface modeling*, 139–149, E.Horwood, New York.

[11] Huyakorn, P. S. and Andersen P. F. (1986), A Curvilinear Finite Element Model for Simulating Two-Well Tracer Tests and Transport in Stratified Aquifer, Water Resources Research 22, 663–678.

[12] Irvine, D., Marin, S.P. and Smith, P.W. (1986), Constrained interpolation and smoothing, Constr. Approx., 129–151.

[13] Kim, M.S. and Nam, K.W. (1995), Interpolating solid orientations with circular blending quaternion curves, Computer Aided Design 27, 385–398.

[14] Knupp, P. and Steinberg, S. (1993), *Fundamentals of Grid Generation*, CRC Press, Boca Raton, FL.

[15] Lee, E.T.Y. (1989), Choosing nodes in parametric curve interpolation, Computer Aided Design 21, 363–370.

[16] Margolin, L. and Shashkov, M. (1998) *Using a Curvilinear Grid to Construct Symmetry-Preserving Discretizations for Lagrangian Gas Dynamics*, LA-UR-98-3480, Report of Los Alamos National Laboratory, Los Alamos, NM, USA, submitted to Journal of Computational Physics.

[17] Mjolsness, R.C. and Swartz, B. (1987), Some plane curvature approximations, Math. Comp. 49, 215–230.

[18] Overhauser, A.W. (1968), *Analytic definition of curves and surfaces by parabolic blending*, Technical Report SL 68-40, Ford Motor Company, Scientific Res. Staff Publ.

[19] Sladek, V. and J. Sladek, J. (1995), Smooth modeling of the normal vector by using C^0-continuous Lagrange elements in BEM, Communications in Numerical Methods in Engineering 11, 981–993.

[20] Struik, D.J. (1950), *Lectures on Classical Differential Geometry*, Addison-Wesley.

[21] Town, R.P.J. and Bell, A.R. (1991), Three-Dimensional Simulations of the Implosion of Inertial Confinement Fusion Targets, Phys. Rev. Lett. 67, 1863–1866.

[22] Wenz, H.-J. Interpolation of curve data by blended generalized circles, Computer Aided Geometric Design 13, 673–680.

[23] Zlamal, M. (1973 a), The Finite Element Method in domains with curved boundaries, Int. J. Numer. Meth. Engineering 5, 367–373.

[24] Zlamal, M. (1973 b), Curved elements in the Finite Element Method I, SIAM J. Numer. Anal. 10, 229–240.