Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2007) 1094–1125
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Accurate, high-order representation of complex three-dimensional surfaces via Fourier continuation analysis

Oscar P. Bruno [a,*], Youngae Han [a], Matthew M. Pohlman [b]

[a] *Applied and Computational Mathematics, 1200 E California Boulevard, M/C 217-50, Caltech, Pasadena, CA 91125, United States*
[b] *Areté Associates, Sherman Oaks, CA 91403, United States*

Received 11 December 2006; received in revised form 24 August 2007; accepted 28 August 2007
Available online 12 September 2007

Abstract

We present a new method for *construction of high-order parametrizations of surfaces*: starting from point clouds, the method we propose can be used to produce full surface parametrizations (by sets of local charts, each one representing a large surface patch – which, typically, contains thousands of the points in the original point-cloud) for complex surfaces of scientific and engineering relevance. The proposed approach accurately renders both smooth and non-smooth portions of a surface: it yields super-algebraically convergent Fourier series approximations to a given surface up to and including all points of geometric singularity, such as corners, edges, conical points, etc. In view of their C^∞ smoothness (except at true geometric singularities) and their properties of high-order approximation, the surfaces produced by this method are suitable for use in conjunction with high-order numerical methods for boundary value problems in domains with complex boundaries, including PDE solvers, integral equation solvers, etc. Our approach is based on a very simple concept: use of Fourier analysis to continue smooth portions of a piecewise smooth function into new functions which, defined on larger domains, are both smooth and *periodic*. The "continuation functions" arising from a function f converge super-algebraically to f in its domain of definition as discretizations are refined. We demonstrate the capabilities of the proposed approach for a number of surfaces of engineering relevance.

Keywords: Continuation method; Surface representation; Fourier series; Edge matching; Parametrization by projection

1. Introduction

The problem of producing high-order parametrizations of surfaces and, more generally, for given integers d and n ($d < n$), high-order parametrizations of d-dimensional manifolds in n-dimensional space, is one of great importance in a wide range of fields of science and engineering. The numerical solution of partial differential

* Corresponding author. Tel.: +1 626 395 4548; fax: +1 626 578 0124.
E-mail address: bruno@acm.caltech.edu (O.P. Bruno).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2007.08.029

equations (PDE) with *high-order accuracy* (see e.g. [4,6,7,9,10,14,36]), for example, requires construction of highly accurate, high-order representations of domain boundaries; it is indeed this particular application that motivated the work discussed in this paper. Engineering descriptions of surfaces are usually provided in some file format, such as might be created using a computer aided design (CAD) system; generally, the raw geometry representations available in engineering practice must be processed to satisfy the requirements of a high-order application. As an example of the geometry processing that might be required, consider the aircraft wing shown in Fig. 20: high-order solution of PDE boundary value problem associated with this geometry is only possible once the high-order information implicit in the low order definition of the bounding surface is appropriately extracted.

In this paper we introduce an algorithm for creation of high-order surface representations from given point data. The proposed approach is based on a innovative paradigm: representation of two-dimensional surfaces in three-dimensional space on the basis of *large patch surface parametrizations* that result from Fourier analysis and a certain "continuation method" [8,12] for construction of rapidly converging Fourier series. Briefly, the continuation method enables creation of rapidly convergent Fourier series from discrete point values of functions that are smooth but not necessarily periodic. The Fourier approximations resulting from the continuation method are C^∞ smooth (except at true geometric singularities) as well as highly accurate: function and derivatives are approximated with high-order accuracy. This method effectively eliminates the Gibbs ringing by *continuing* the function into a larger domain, thus allowing for periodicity to occur; the continuation functions, such as those depicted in Fig. 2, are produced automatically by the continuation method through use of a certain least-squares procedure, and need not in any way be provided by the user.

(The Fourier continuation method had thus far been considered, in a handful of contributions, as an element in solvers for partial differential equations. In fact we rediscovered the continuation method independently [3] while developing the surface representation methods described in this text. In the Appendix, further, we present novel uses of the idea of Fourier continuation giving rise to resolution of the Gibbs phenomenon for the problem of evaluation of Fourier series of non-periodic smooth functions.)

Although applications of techniques described in this text into various areas of computer aided design could be envisioned, including reverse engineering [17,18] and CAD/mesh-repair [5,28,31], the focus of the present contribution is to provide a methodology that enables construction of surface representations suitable for use in conjunction with efficient PDE solvers – for PDEs defined either on the surfaces themselves or on volumes bounded by them [4,6,7,9,10,14,36,40]. The types of surface representation methods that have emerged from the CAD field are not well adapted for solution of such PDE problems *with high-order accuracy*: an excellent discussion in these regards is given in [15]. Existing reverse engineering approaches, for example, can give rise to *small* NURBS parametrization patches with, usually, *smoothness of low order* across patches (typically smoothness of an order no higher than first occurs at certain points in the representation). The presence of either of these two characteristics virtually rules out the possibility of an effective high-order solver methodology – of desirable orders such as 5, 10, 20, even ∞: large patches allow for use of large locally-Cartesian grids, spectral representations, etc., while high-order surface representations are of course necessary to obtain a corresponding high-order solution accuracy. Smoothness of arbitrary order has previously been achieved on the basis of spline approximations [16,35,37], but the required polynomial degrees grow rapidly as smoothness requirements are increased. Use of interpolations involving such high-degree polynomials results in oscillations and low quality representations; thus explicit constructions often considered in practice are of C^1 or C^2 continuity; see e.g. [2,32,33].

These characteristics of previous reverse-engineering approaches stem mainly from two shared design traits: (a) use of *polynomial or rational approximants* and (b) use of representations which, in smooth portions of the surface are *smoothly matched along a system of curves*, so that the intersection between two patches is either empty or a portion of a matching curve. While maintaining their reliance on use of polynomial approximants (point (a)), previous manifold-representation methodologies [13,41] relax the requirement (b) by resorting to (c) use of overlapping patches; they achieve this by employing very small patches [30, p. 644]: the domains of the various local parametrizations are closely related to the stars of single vertices in the original mesh, and the size of a patch corresponds closely to the size of the corresponding star. While the resulting parametrizations are smooth, further, they follow closely the non-smooth discrete representations on which they are based – consider e.g. the left portion of Fig. 21, and they are thus likely to exhibit significant oscillations and departures from the

actual surface they approximate. Finally, the size of a patch resulting from these approaches is of the order of the discretization mesh, and, indeed, it tends to zero as the discretization mesh tends to zero – which, as discussed above, presents a difficulty if the representations are to be used as a basis for high-order PDE solvers.

The methodology proposed in this paper does not rely on either of the methodologies (a) or (b) just described. As mentioned above, instead of polynomials this approach utilizes a Fourier basis and the continuation method. The advantages of this strategy are very significant: very large continuation patches can be used without giving rise to undesirable oscillations or other unacceptable approximating properties. A wide range of examples of this fact are presented in this paper; a very simple one-dimensional example, presented in Table 10, may be particularly useful: using a given set of data, the errors in a continuation approximation of a function were reduced to the order of machine precision, with or without use of oversampling, while the corresponding polynomial approximations diverge grossly in the non-oversampled case, and fail to converge even in the oversampled case.

Concerning points (b) and (c), on the other hand, our methodology bears some connections with previous manifold-construction approaches, but in practice differs significantly from them. On one hand, the space of parameters used in the methodology proposed in this text, which consist of certain ''projection surfaces'' (that are in fact akin to the ''simple base-surfaces'' described in the reverse-engineering text [17, p. 659]), are not based in any way on underlying meshes. And, unlike previous manifold-representation approaches (which are based on construction of edge, vertex and face patches [13] or stars of vertices [41]) can give rise to parametrizations involving very large patches, see Section 3. On the other hand, the identification functions we use in the patch-blending step (at patch overlaps), cannot depend, unlike those used by previous manifold-representation approaches [13,41], on use of an underlying mesh – since, as mentioned above, in order to allow for construction of large patches, meshes are not used by our method. Instead, our identification procedures at smooth portions of a surface result from use of intersections of lines with two or more patches: roughly speaking, if a normal to one patch at a point intersect another patch, then the two intersection points are identified and subsequently blended by means of a C^∞ partition-of-unity. A corresponding matching of edges is introduced for description of surfaces around actual physical edges; see Sections 4 and 5 for details.

Since the methodology we present is based on use of point clouds, *correction of CAD inaccuracies as well as intelligent model defeaturing can be made to follow from use of the continuation method* (see e.g. Figs. 20 and 21). As discussed in Section 5, geometric singularities such as corners, edges, conical points, etc., which are of great importance in many applications, can be treated effectively in this framework as well: high-order accurate approximations of surfaces and their derivatives of various orders result from the proposed approach *up to and including points of geometric singularity*. In all, the methodology presented in this paper produces high-quality surface parametrizations consistent with the information available in a given geometry description; as finer and finer discretizations of the given surface are provided, further, the approximations converge spectrally to the underlying surface.

This paper is organized as follows. In Section 2 we present the continuation method and we discuss its convergence properties in presence of both noiseless and noisy data. A number of specific methodologies we use to parametrize given point data by means of the continuation method are then presented in Sections 3–5: in Section 3 we discuss the basic patching-and-parametrization strategy on which our approach is based; in Section 4 we present techniques that can be used to ensure perfect matching of patches in overlap regions and to construct smooth partitions-of-unity throughout the surface; and in Section 5 we extend the methodologies of Section 4 to enable perfect matching of patches at corners, edges and other geometric singularities. Examples demonstrating the character of the surfaces produced by this method are given throughout the text. As mentioned above, in Appendix A we present, for completeness, novel uses of the idea of Fourier continuation giving rise to resolution of the Gibbs phenomenon for the problem of evaluating Fourier series of discontinuous functions, and we place this methodology in the context of the existing related literature. In Appendix B, finally, we present an approach that can be used to improve some conditioning aspects of the Fourier continuation problem.

2. The continuation method

The continuation method results as one seeks to reproduce, for non-periodic functions, the excellent properties of convergence of Fourier series of smooth periodic functions. To gain a sense of the properties of the

continuation method let us at first consider Fig. 1: For a function $y = f(x)$ defined, say, in the interval $0 \leqslant x \leqslant 1$ (in our first example we take $f(x) = x$), the truncated Fourier series S_f^n of f oscillates around the points of discontinuity in a periodic extension of f with period 1 (the Gibbs phenomenon), and it thus converges poorly to f as $n \to \infty$. To overcome this difficulty, the continuation method seeks an approximating Fourier series of periodicity *larger* than the domain of definition of $f(x)$ – in our example, displayed in Fig. 2, the enlarged periodicity interval is $0 \leqslant x \leqslant 2$. An oversampled linear algebra problem is then used to obtain a number M of Fourier coefficients so that the corresponding Fourier series matches, in the least-squares sense, the values of the function at the given N possibly non-equispaced points ($N > M$) in the interval $0 \leqslant x \leqslant 1$. The continuation functions, such as the one depicted in Fig. 2 left, are produced automatically by the oversampled-interpolation algorithm: no values are provided to the algorithm beyond those available in the domain of definition of the function—in this case, the extension to the full interval $0 \leqslant x \leqslant 2$ results directly from the algorithm, without any recourse to user-supplied data in the interval $1 \leqslant x \leqslant 2$. From the depiction and tabular values presented in Fig. 2 we see that excellent approximations are obtained for the function and its derivatives throughout the interval of definition, including the endpoints.

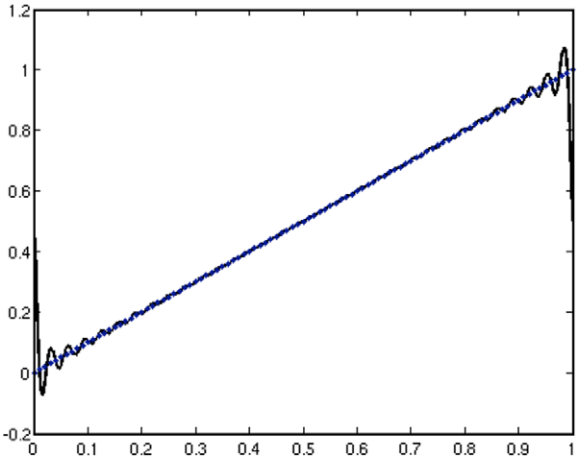
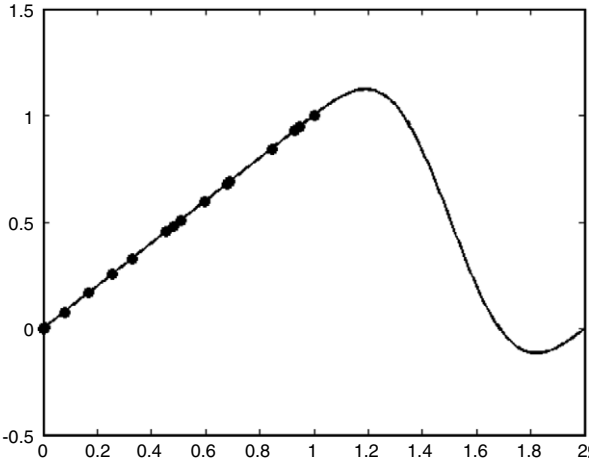


Fig. 1. The Gibbs phenomenon.



N	$f(x)$	df/dx	d^2f/dx^2
8	$1.4 \cdot 10^{-2}$	$4.6 \cdot 10^{-1}$	$4.8 \cdot 10^{0}$
16	$4.9 \cdot 10^{-4}$	$3.7 \cdot 10^{-2}$	$1.4 \cdot 10^{0}$
32	$6.2 \cdot 10^{-7}$	$1.3 \cdot 10^{-4}$	$1.7 \cdot 10^{-2}$
64	$2.4 \cdot 10^{-12}$	$1.4 \cdot 10^{-9}$	$5.1 \cdot 10^{-7}$
128	$1.5 \cdot 10^{-14}$	$1.1 \cdot 10^{-11}$	$8.6 \cdot 10^{-9}$
8	$1.7 \cdot 10^{-2}$	$4.8 \cdot 10^{-1}$	$4.9 \cdot 10^{0}$
16	$8.8 \cdot 10^{-4}$	$4.7 \cdot 10^{-2}$	$1.6 \cdot 10^{0}$
32	$1.7 \cdot 10^{-6}$	$2.7 \cdot 10^{-4}$	$3.0 \cdot 10^{-2}$
64	$1.5 \cdot 10^{-11}$	$6.0 \cdot 10^{-9}$	$1.7 \cdot 10^{-6}$
128	$2.9 \cdot 10^{-14}$	$2.4 \cdot 10^{-11}$	$1.3 \cdot 10^{-8}$

Fig. 2. Left: Interpolation of the function $f(x) = x$ using the continuation method on N data points in the interval $[0, 1]$, $M = N/2$. Tables: maximum interpolation errors in the interval $[0, 1]$ for the function and its first two derivatives, and for various values of N. Upper table: Uniform discretizations as in Eq. (1). Lower table: unevenly spaced points $x_j = (j-1)/(N-1) + \delta_j$ where $\delta_0 = \delta_N = 0$ and, for $1 \leqslant j \leqslant (N-1)$,$\delta_j$ are uniformly distributed random values with $|\delta_j| \leqslant 1/(N-1)$.

Remark 1. In Table 1 and throughout this paper, maximum errors are computed through comparison of approximations and exact functions at sufficiently large numbers (thousands) of points to insure complete resolution of all function features and accurate estimates of maximum errors.

A detailed description of the continuation method in two and three dimensions is presented in Sections 2.1 and 2.2. In Section 2.3 we then demonstrate the benefits brought about by the use of oversampling (see also [8]), and we provide a brief theoretical discussion of the properties of the continuation method. As mentioned above, in the subsequent Sections 3–5 we present and demonstrate a collection of techniques we developed which, in conjunction with the continuation method, make up an effective methodology for high-order parametrization of surfaces.

2.1. The continuation algorithm and "least-squares trigonometric interpolation": one-dimensional case

Let $y = f(x)$ be a smooth function defined in the interval $0 \leqslant x \leqslant 1$, $f \in C^k[0, 1]$ with either k a positive integer or $k = \infty$. Further, let x_j be an increasing sequence of points in that interval satisfying $0 \leqslant x_j \leqslant 1$, $j = 1, \ldots, N$, $x_1 = 0$, $x_N = 1$, and let $y_j = f(x_j)$. Note that the abscissae x_j need not be equally spaced; the case in which the sampling points are equispaced,

$$x_j = \frac{j-1}{N-1}, \quad j = 1, \ldots, N \tag{1}$$

is of practical and theoretical importance, if not the one that arises most often in surface representation applications.

The continuation method produces certain *least-squares interpolations by trigonometric polynomials* of the point values (x_j, y_j). The essence of the continuation method lies in using trigonometric polynomials which

1. Have a periodicity interval $[0, b]$ larger than $[0, 1]$ (i.e. $b > 1$);
2. Contain a total of M Fourier modes, with M typically smaller than N; and which
3. Match the given data points in the sense of least-squares.

In detail, defining the column vectors $x = (x_1, \ldots, x_N)^T$ and $y = (y_1, \ldots, y_N)^T$, the continuation method produces trigonometric approximations with Fourier coefficients $a = (a_{-M/2}, \ldots, a_0, \ldots, a_{M/2-1})^T$ for M even and $a = (a_{-(M-1)/2}, \ldots, a_0, \ldots, a_{(M-1)/2})^T$ for M odd. Letting $t(M) = \{j \in \mathbb{N} : -(M-1)/2 \leqslant j \leqslant (M-1)/2\}$ for M odd and $t(M) = \{j \in \mathbb{N} : -M/2 \leqslant j \leqslant M/2 - 1\}$ for M even, the coefficients a_k are obtained as the *least-squares* solution of the (over-determined) system of linear equations

$$y_j = \sum_{k \in t(M)} a_k e^{\frac{2\pi i}{b} k x_j}, \quad j = 1, \ldots, N. \tag{2}$$

Table 1
Error in the interpolation of $f(x) = x$ shown in Fig. 2 by continuing the function to a smooth periodic function, with data given for N evenly spaced data points and M mode number

M	N	Max. error	Ratio	Max. d/dx err	Ratio	Max. d^2/dx^2 err	Ratio
4	8	1.4×10^{-2}		4.6×10^{-1}		4.8×10^{0}	
8	16	4.9×10^{-4}	2.9×10^{1}	3.7×10^{-2}	1.2×10^{1}	1.4×10^{0}	3.4×10^{0}
16	32	6.2×10^{-7}	7.9×10^{2}	1.3×10^{-4}	2.9×10^{2}	1.7×10^{-2}	8.3×10^{1}
32	64	2.4×10^{-12}	2.5×10^{5}	1.4×10^{-9}	9.4×10^{4}	5.1×10^{-7}	3.3×10^{4}
64	128	1.5×10^{-14}	1.7×10^{2}	1.1×10^{-11}	1.2×10^{2}	8.6×10^{-9}	5.9×10^{1}
8	8	8.3×10^{-4}		4.4×10^{-2}		1.5×10^{0}	
16	16	3.4×10^{-6}	2.5×10^{2}	4.5×10^{-4}	9.8×10^{1}	4.1×10^{-2}	3.7×10^{1}
32	32	1.5×10^{-10}	2.2×10^{4}	4.9×10^{-8}	9.3×10^{3}	1.1×10^{-5}	3.7×10^{3}
64	64	2.0×10^{-11}	7.5×10^{0}	1.3×10^{-8}	3.8×10^{0}	5.6×10^{-6}	2.0×10^{0}
128	128	1.4×10^{-10}	1.4×10^{-1}	1.9×10^{-7}	6.8×10^{-2}	1.7×10^{-4}	3.2×10^{-2}

Remark 2. For real valued functions f, significantly reduced computing times result from use of sin/cos expansions rather than expansions in terms of complex exponentials. For notational convenience, however, all of our descriptions and theoretical discussions use the complex exponential notation.

Clearly, the system of Eq. (2) can be re-expressed in the form

$$y = Aa \tag{3}$$

where A is a (typically not square) $N \times M$ matrix. The approximation produced by the continuation method is thus defined as

$$f^c(x) = f^c_{N,M}(x) = \sum_{k \in t(M)} a^c_k e^{\frac{2\pi i}{b} kx} \tag{4}$$

with coefficients $a^c = (a^c_k)_{k \in t(M)}$ equal to the least-squares solution of Eq. (3), that is, the solution of the minimization problem

$$\min_{(a_k)} \sum_{j=1}^{N} \left| \sum_{k \in t(M)} a_k e^{\pi i k x_j} - y_j \right|^2. \tag{5}$$

Eqs. (4) and (5) provide a complete description of the continuation method for functions defined in a one dimensional interval. Theoretical questions associated with the algorithm, including its convergence properties for a given continuation interval $[0, b]$ and number $M = M(N)$ of modes used for a given number N of discretization points, are addressed in Section 2.3. A discussion of the relative performance of the algorithm that result as various possible (e.g. iterative) solvers are used for least-squares problem 5 will be provided elsewhere; throughout this paper we use an SVD based least-square solver as described in [34, pp. 57–58]. Before considering theoretical questions and implementation issues, however, we briefly discuss the continuation algorithm in two and higher dimensions.

2.2. The continuation algorithm in dimensions $d \geqslant 2$

Clearly the algorithm described in the previous section generalizes directly to functions defined in d-dimensional cubes, $d \geqslant 2$. Interestingly, further, by the very nature of the approach, the function to be approximated needs not be defined in a d-dimensional cube, see e.g. Fig. 7. Thus, the data from which a function is to be approximated can be given as an irregular sampling of the function in an arbitrary region in $\mathbb{R}^d$; see also Remark 3.

2.3. Some theoretical considerations

In this section we discuss the convergence properties of the continuation method for a smooth function $f : [0, 1] \to \mathbb{R}$. For simplicity we (1) Restrict our theoretical discussion to functions of $d = 1$ variable (the extension to the case $d > 1$ is direct), and we (2) Assume the discretization points are equispaced – given by Eq. (1) for some N.

Since f is smooth ($f \in C^k[0,1]$ with either k a positive integer or $k = \infty$), for any $b > 1$ there exists a b-periodic smooth continuation $\tilde{f} : \mathbb{R} \to \mathbb{R}$ of f, that is, $\tilde{f}$ satisfies:

1. $\tilde{f} \in C^k(\mathbb{R})$
2. $\tilde{f}(x + b) = \tilde{f}(x)$ for all $x \in \mathbb{R}$,
3. $f(x) = \tilde{f}(x)$ for $x \in [0, 1]$.

The existence of such an extension can easily be established through consideration of the Taylor-series of f at $x = 0$ and $x = 1$ and use of appropriate smooth windowing functions around $x = 0$ and $x = 1$.

The function $\tilde{f}$ can be represented by its b-periodic Fourier series

$$\tilde{f}(x) = \sum_{k=-\infty}^{\infty} \tilde{a}_k e^{i n \frac{2\pi}{b} x}, \tag{6}$$

where

$$\tilde{a}_k = \frac{1}{b}\int_0^b \tilde{f}(x)\mathrm{e}^{-\mathrm{i}k\frac{2\pi}{b}x}\mathrm{d}x.$$

Since $\tilde{f}$ is k-times continuously differentiable (k a positive integer or $k=\infty$) its truncated Fourier series

$$S_M(x) = \sum_{k\in t(M)} \tilde{a}_k \mathrm{e}^{\mathrm{i}n\frac{2\pi}{b}x} \tag{7}$$

converges rapidly to $\tilde{f}$: for example, if $\tilde{f}\in C^\infty(\mathbb{R})$ then both the Fourier coefficients $\tilde{a}_k$ for $k\notin t(M)$ and the error in the approximation of $\tilde{f}$ by S_M are super-algebraically small, that is, of the order of $1/M^p$ for all integers p. The corresponding approximation error is of order $1/M^{s-1}$ if, say, $\tilde{f}$ is piecewise $\tilde{f}\in C^\infty$ and $\tilde{f}\in C^s(\mathbb{R})$ for a finite positive integer s. Thus there indeed exists a (highly non unique!) series of the form 4 that approximates f closely. The continuation method does not rely upon the numerically inaccurate Taylor-series-based construction of a b-periodic extension of f mentioned above, or any other previously existing extension technique. Rather, the procedure embodied in Eqs. (4) and (5) produces a continuation function $\tilde{f}=f^c$ and its b-periodic finite Fourier series *simultaneously* by means of a suitable least-squares computation.

To study the convergence of the series (4) for large values of M and N we consider various (linear) prescriptions $M=M(N)$ for the number M of modes used for a given discretization size N. Clearly, we might use $M(N)=N$ modes to produce all N modes with indexes $k\in t(M)$ through inversion of a $N\times N$ nonsingular matrix. Or, we may use $M(N)<N$ and a corresponding oversampled linear system of equations. A particular selection of the function $M(N)$ gives rise to a specific instance of the continuation algorithm; the merits of various such selections are discussed in what follows. A selection we have often found useful is $M(N)=[N/2]$, for which the total number of modes is roughly one-half of the number of sampling points used.

An understanding of the properties of the continuation method for some prescription $M=M(N)$ can be gained once knowledge is available on the behavior of the algorithm on b-periodic complex exponentials. Indeed, since $\tilde{f}(x)=f(x)$ for $0\leqslant x\leqslant 1$, Eq. (6) tells us that the approximation $f^c=f^c_{N,M}$ that arises from application of the (M,N) continuation method to f equals

$$f^c_{N,M}(x) = \sum_{n=-\infty}^{\infty} \tilde{a}_n \mathrm{e}^n_{N,M}(x), \tag{8}$$

where $\mathrm{e}^n_{N,M}$ denotes the function

$$\mathrm{e}^n_{N,M}(x) = \left\{\mathrm{e}^{\mathrm{i}n\frac{2\pi}{b}x}\right\}^c_{N,M}, \tag{9}$$

that is, the result of an application of the continuation method with N points and M modes to the exponential $\mathrm{e}^{\mathrm{i}n\frac{2\pi}{b}x}$. For future use we introduce notation concerning applications of the continuation method in finite precision arithmetic: we have

$$\dot{f}^c_{N,M}(x) = \sum_{n=-\infty}^{\infty} \tilde{a}_n \dot{\mathrm{e}}^n_{N,M}(x), \tag{10}$$

where $\dot{f}^c_{N,M}$ and $\dot{\mathrm{e}}^n_{N,M}$ denote the result of an application of the continuation method to the function f and to the exponential $\mathrm{e}^{\mathrm{i}n\frac{2\pi}{b}x}$, respectively, using *finite precision arithmetic* – and, as before, using N discretization points and M Fourier modes.

Comparing Eqs. (6) and (8) we see that the error in an exact arithmetic continuation approximation for a given pair (M,N) can be bounded in terms of the quantities

$$E_{N,M} = \max_{n\in t(M)} \max_{0\leqslant x\leqslant 1} \left|\mathrm{e}^n_{N,M}(x) - \mathrm{e}^{\mathrm{i}n\frac{2\pi}{b}x}\right| \quad \text{and} \quad B_{N,M} = \max_{n\notin t(M)} \max_{0\leqslant x\leqslant 1} \left|\mathrm{e}^n_{N,M}(x)\right|. \tag{11}$$

Clearly $E_{N,M}=0$ (compare $\dot{E}_{N,M}$ below, however), so that our main error estimate is given by

$$|f(x) - f^c_{N,M}(x)| \leqslant B_{N,M}\cdot \sum_{k\notin t(M)} |\tilde{a}_k| \tag{12}$$

for all $x \in [0,1]$. In view of the rapid decay of the coefficients $\tilde{a}_n$ that, as mentioned above, arises for smooth functions f, the sum in Eq. (12) converges rapidly to zero as $M \to \infty$. Thus, if for a given choice of the function $M = M(N)$ we have that $B_{N,M(N)}$ is uniformly bounded for all N, or, otherwise, it does not grow too rapidly as $N \to \infty$, the convergence of $f^c_{N,M}(x)$ to $f(x)$ results. As shown below, under the current assumption the N discretization points are equispaced and provided the quantity $b(N-1)$ is an integer, for a given value of N the coefficient $B_{N,M(N)}$ can be easily obtained through evaluation of a *finite number of quantities*.

The use of finite precision arithmetic requires additional considerations: as in the exact arithmetic case, here we have

$$|f(x) - \dot{f}^c_{N,M}(x)| \leqslant \dot{E}_{N,M} \cdot \sum_{k \in t(M)} |\tilde{a}_k| + \dot{B}_{N,M} \cdot \sum_{k \notin t(M)} |\tilde{a}_k| \tag{13}$$

for all $x \in [0,1]$, where

$$\dot{E}_{N,M} = \max_{n \in t(M)} \max_{0 \leqslant x \leqslant 1} \left| \dot{e}^n_{N,M}(x) - e^{i n \frac{2\pi}{b} x} \right| \quad \text{and} \quad \dot{B}_{N,M} = \max_{n \notin t(M)} \max_{0 \leqslant x \leqslant 1} \left| \dot{e}^n_{N,M}(x) \right|.$$

(For notational simplicity here and in what follows we do not differentiate between the exact coefficients $\tilde{a}_k$ and their numerical approximations.)

A similar discussion to that provided for the case of exact arithmetic applies here: In view of the rapid decay of the coefficients $\tilde{a}_n$ the first sum in Eq. (13) remains uniformly bounded as $M \to \infty$, while second sum in Eq. (13) converges rapidly to zero as $M \to \infty$. Thus, if for a given choice of the function $M = M(N)$ we have that $\dot{E}_{N,M(N)}$ remains almost vanishingly small (as it may be expected since $E_{N,M(N)} = 0$ and as is verified numerically in Table 2) and $B_{N,M(N)}$ is uniformly bounded for all N (or, otherwise, it does not grow too rapidly as $N \to \infty$), then a rapid convergence of $\dot{f}^c_{N,M}$ to f ensues.

As mentioned above, under the current assumption of an equispaced discretization and provided the quantity $b(N-1)$ is an integer, for a given value of N the coefficients $\dot{B}_{N,M(N)}$ and $\dot{E}_{N,M(N)}$ can be obtained through evaluation of a *finite number of quantities*. Indeed, we see that that for x on the *equispaced* set (1) of N discretization points, to which the material in the present section is restricted, a certain pure-mode $e^{i n \frac{2\pi}{b} x}$ with $n > b(N-1)$ coincides with $e^{i r \frac{2\pi}{b} x}$, where $r(|r| < b(N-1))$ is the remainder of the division of n by $b(N-1)$. Clearly, thus,

$$\dot{e}^n_{N,M} = \dot{e}^r_{N,M}$$

and the maximum in n in the definition of (Eq. (11)) can be produced by evaluating the finitely many quantities

$$\dot{e}^r_{N,M}, \quad \text{for } 0 \leqslant r < b(N-1).$$

Table 2 displays $\dot{E}$ and $\dot{B}$ for parameter values that allow evaluation of these quantities by means of finitely many operations, as described above. In conjunction with Eq. (13), this table gives rigorous estimates on the approximation errors that may be expected as the continuation method is applied to a given function f. We emphasize, however, that the convergence behavior of the algorithm is independent of the assumptions of this section; in view of a wealth of numerical experiments we expect the estimates arising from Table 2 and Eq. (13)

Table 2
The quantities $\dot{E}_{N,M(N)}$ and $\dot{B}_{N,M(N)}$ for extension of functions in the interval $[0,1]$ to the interval $[0,b]$, with $M(N) = (N-1)/2$ and for various values of b

b		$N = 9$	$N = 17$	$N = 33$	$N = 65$	$N = 129$	$N = 257$	$N = 513$
1.5	$\dot{E}_{N,M(N)}$	2×10^{-15}	4×10^{-15}	1×10^{-14}	4×10^{-14}	3×10^{-12}	1×10^{-11}	3×10^{-11}
	$\dot{B}_{N,M(N)}$	1×10^{0}	1×10^{0}	2×10^{0}	6×10^{0}	4×10^{2}	9×10^{2}	1×10^{3}
2	$\dot{E}_{N,M(N)}$	1×10^{-15}	2×10^{-15}	2×10^{-15}	3×10^{-14}	8×10^{-14}	6×10^{-13}	3×10^{-13}
	$\dot{B}_{N,M(N)}$	1×10^{0}	1×10^{0}	3×10^{0}	2×10^{1}	1×10^{2}	1×10^{2}	2×10^{2}
4	$\dot{E}_{N,M(N)}$	1×10^{-15}	2×10^{-15}	2×10^{-15}	6×10^{-15}	2×10^{-14}	4×10^{-14}	6×10^{-14}
	$\dot{B}_{N,M(N)}$	1×10^{0}	2×10^{0}	3×10^{0}	9×10^{0}	1×10^{1}	1×10^{1}	1×10^{1}

Uniform discretizations; integer values of $b(N-1)$.

portray the convergence behavior of the continuation method under rather generic conditions, including possibly non-equispaced discretizations and non-integer values of $b(N-1)$.

2.4. Parameter values

While the discussion of the Section 2.3 provides a clear rationale for the properties of the continuation method under various choices of oversampling ratio q and continuation length b, it is difficult in practice to select these parameters on the basis of the results of that section – since, for a given smooth function f, bounds on the values of the Fourier coefficients of a continuation of f are not know a priori. We have thus conducted extensive sets of experiments to gain a sense of the values that can generally be used for the parameters of the algorithm. We illustrate some typical situations in Tables 3–5; as can be seen in these tables, the rule-of-thumb $b = 2$ and $q = 0.5$ we commonly use is generally appropriate, although not necessarily optimal. Table 3 shows that for the simple function $f(x) = x$ our rule-of-thumb gives rise to significantly suboptimal convergence: use of parameter values $b = 4$ and $q = 1$ leads to much faster convergence in this case. The situation is reversed in other cases; one notable example is that considered in Table 4: the famous Runge function, for which polynomial approximation

Table 3
Interpolation errors for the function $y = x$ in the interval $[0,1]$ to the interval $[0,b]$, for various values of b, and using a number $M(N) = q \times N$ of Fourier modes

b	q	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$
1.5	0.25	2.7×10^{-1}	8.0×10^{-2}	6.6×10^{-3}	7.0×10^{-5}	1.3×10^{-8}
	0.50	6.2×10^{-2}	6.2×10^{-3}	9.4×10^{-5}	3.9×10^{-8}	5.1×10^{-14}
	0.75	2.1×10^{-2}	1.0×10^{-3}	4.7×10^{-6}	2.7×10^{-10}	2.8×10^{-11}
	1.00	9.5×10^{-3}	3.1×10^{-4}	7.7×10^{-7}	7.6×10^{-9}	1.8×10^{-8}
2	0.25	1.5×10^{-1}	2.1×10^{-2}	4.1×10^{-4}	3.9×10^{-7}	4.4×10^{-13}
	0.50	1.4×10^{-2}	4.9×10^{-4}	6.2×10^{-7}	2.4×10^{-12}	1.5×10^{-14}
	0.75	3.1×10^{-3}	2.6×10^{-5}	4.5×10^{-9}	8.1×10^{-13}	1.2×10^{-12}
	1.00	8.3×10^{-4}	3.4×10^{-6}	1.5×10^{-10}	2.0×10^{-11}	1.4×10^{-10}
4	0.25	3.8×10^{-2}	1.1×10^{-3}	1.2×10^{-6}	3.5×10^{-12}	2.0×10^{-15}
	0.50	7.7×10^{-4}	1.5×10^{-6}	6.8×10^{-12}	2.2×10^{-15}	2.3×10^{-15}
	0.75	4.2×10^{-5}	5.7×10^{-9}	1.8×10^{-14}	1.6×10^{-14}	1.3×10^{-14}
	1.00	3.0×10^{-6}	5.9×10^{-11}	3.3×10^{-13}	6.2×10^{-14}	1.2×10^{-13}

Uniform discretizations.

Table 4
Interpolation errors for the Runge function $y = \frac{1}{1+25(2x-1)^2}$ in the interval $[0,1]$ to the interval $[0,b]$, for various values of b, and using a number $M(N) = q \times N$ of Fourier modes

b	q	$N = 32$	$N = 64$	$N = 128$	$N = 256$
1.5	0.25	1.6×10^{-1}	2.3×10^{-2}	4.8×10^{-4}	4.5×10^{-7}
	0.50	2.3×10^{-2}	8.1×10^{-4}	2.8×10^{-5}	8.0×10^{-11}
	0.75	1.1×10^{-2}	8.4×10^{-3}	6.2×10^{-6}	7.7×10^{-11}
	1.00	1.2×10^{0}	3.7×10^{0}	7.9×10^{-6}	7.6×10^{-8}
2	0.25	1.9×10^{-1}	3.2×10^{-2}	9.1×10^{-4}	1.2×10^{-5}
	0.50	3.1×10^{-2}	5.5×10^{-3}	1.0×10^{-4}	5.9×10^{-9}
	0.75	7.6×10^{-2}	4.4×10^{-2}	3.8×10^{-5}	1.8×10^{-11}
	1.00	2.5×10^{1}	6.7×10^{-1}	1.4×10^{-5}	5.9×10^{-11}
4	0.25	2.2×10^{-1}	4.2×10^{-2}	3.7×10^{-3}	3.8×10^{-4}
	0.50	4.1×10^{-2}	1.5×10^{-2}	1.2×10^{-3}	3.7×10^{-6}
	0.75	3.9×10^{-1}	2.7×10^{-2}	3.1×10^{-4}	1.2×10^{-7}
	1.00	1.7×10^{0}	6.3×10^{-2}	2.0×10^{-4}	2.4×10^{-9}

Uniform discretizations.

Table 5
Interpolation errors for the function $y = \exp(\sin(5.4\pi\, x - 2.7\pi) - \cos(2\pi x))$ in the interval $[0,1]$ to the interval $[0,b]$, for various values of b, and using a number $M(N) = q \times N$ of Fourier modes

b	q	$N = 64$	$N = 128$	$N = 256$
1.5	0.25	4.2×10^{-2}	3.4×10^{-4}	9.2×10^{-9}
	0.50	1.3×10^{-3}	5.1×10^{-7}	8.1×10^{-14}
	0.75	2.3×10^{-3}	2.9×10^{-10}	4.7×10^{-11}
	1.00	1.7×10^{-2}	2.3×10^{-8}	3.5×10^{-8}
2	0.25	6.2×10^{-2}	10.0×10^{-4}	2.7×10^{-7}
	0.50	8.2×10^{-3}	9.4×10^{-7}	1.3×10^{-13}
	0.75	1.9×10^{-2}	8.6×10^{-8}	2.2×10^{-12}
	1.00	2.2×10^{-2}	1.1×10^{-10}	2.1×10^{-10}
4	0.25	8.1×10^{-2}	6.2×10^{-3}	3.1×10^{-4}
	0.50	8.7×10^{-3}	5.8×10^{-4}	2.2×10^{-7}
	0.75	3.0×10^{-2}	7.4×10^{-5}	9.6×10^{-11}
	1.00	4.2×10^{-2}	6.2×10^{-6}	6.4×10^{-14}

Uniform discretizations.

with increasing polynomial degrees fails to converge (see [7] as well as our discussion of approximations to the Runge function in Section A.2). In this case (in which the function is already periodic although the period-1 periodic extension is not smooth) use of $b = 1.5$ gives faster convergence than our rule-of-thumb value $b = 2$; the combination $b = 1.5$ and $q = 0.5$ yields some of the most accurate results amongst those considered here. In Table 5, finally, the rule-of-thumb choices $b = 2$ and $q = 0.5$ yields some of the best results. Our recommendation thus is that, in absence of information on the characteristics of the function approximated (as may be obtained, for example, by trial and error approximations), the rule-of-thumb parameters $b = 2$ and $q = 0.5$ be used. For functions that are periodic or nearly so but whose period-1 periodic extension is not smooth, a value of b smaller than 2 may be used to advantage. Of course, if the function admits a smooth periodic extension of period 1, we *can* use $b = 1$.

2.5. Continuation and oversampling in presence of noisy data

In view of Tables 3–5 we conclude that use of oversampling can be advantageous to some extent in high-precision arithmetic. The usefulness of oversampling is much more marked when the given data contain large errors, as is the case in the surface representation problems we consider in the following sections. Indeed, the high-frequencies present in the error components of the data tend to be amplified by the continuation algo-

Table 6
Approximation of the function $y = x$ *using noisy data* in the interval $[0,1]$ to the interval $[0,2]$. The noise is taken to equal either 10^{-2} or 10^{-3} times a data point dependent random number "rand" uniformly distributed in the interval $[-1,1]$

M	N	Data point error	Max. error	Noise
16	16	1.2×10^{-12}	2.1×10^{-1}	rand $\times 10^{-2}$
32	32	9.2×10^{-5}	2.4×10^{2}	rand $\times 10^{-2}$
16	16	8.8×10^{-14}	1.0×10^{-2}	rand $\times 10^{-3}$
32	32	4.1×10^{-5}	1.2×10^{2}	rand $\times 10^{-3}$
8	16	9.7×10^{-3}	1.0×10^{-2}	rand $\times 10^{-2}$
16	32	7.7×10^{-3}	8.6×10^{-3}	rand $\times 10^{-2}$
8	16	5.3×10^{-4}	1.2×10^{-3}	rand $\times 10^{-3}$
16	32	8.4×10^{-4}	9.6×10^{-4}	rand $\times 10^{-3}$

Max error is the error between the exact function $y = x$ (without noise) and the continuation obtained from the noisy data. Data point error is the corresponding error at evenly spaced discretization points. The first two rows in the upper table correspond to Fig. 3, while the first two rows in the lower table correspond to Fig. 4.

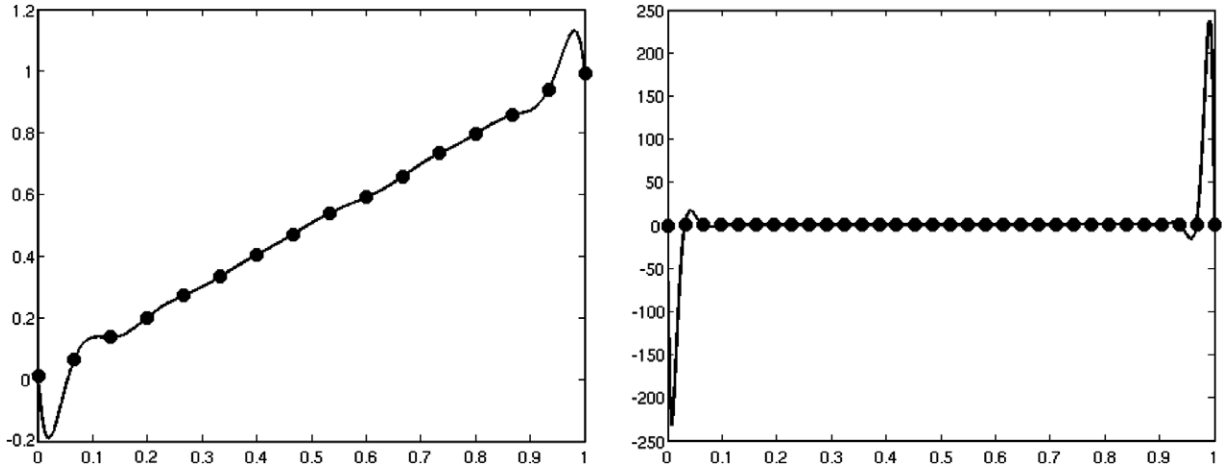


Fig. 3. Approximation of the function $f(x) = x$ using the continuation method on N data points with random noise of magnitude of 10^{-2} using $b = 2$ and $q = 1$ (no oversampling); see Table 6. Left: $N = 16$. Right: $N = 32$.

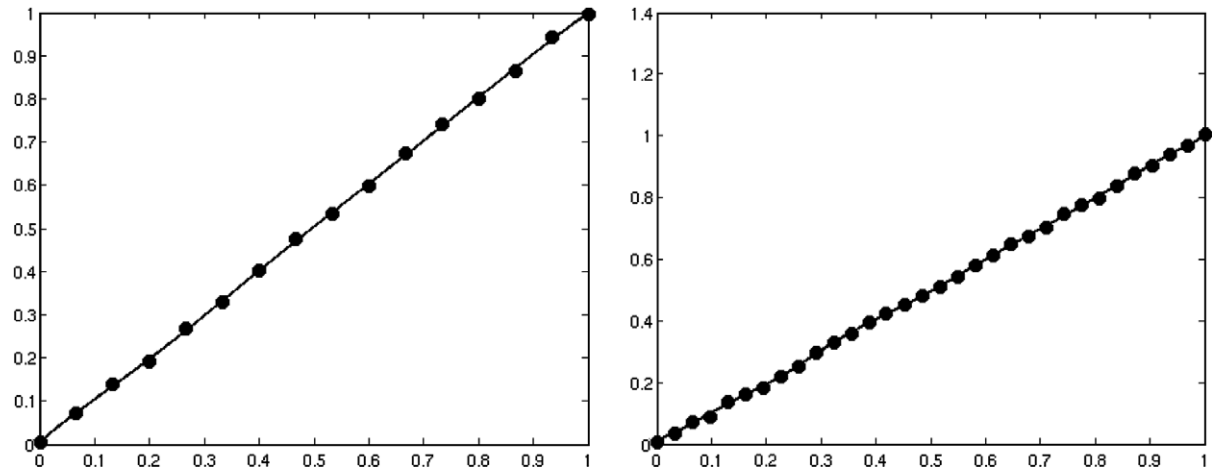


Fig. 4. Approximation of the function $f(x) = x$ using the continuation method on N data points with random noise of magnitude of 10^{-2} using $b = 2$ and $q = 0.5$ (oversampling); see Table 6. Left: $N = 16$. Right: $N = 32$.

rithm, and give rise to bad approximations; the use of oversampling significantly alleviates this difficulty (see Table 6).

We illustrate this point with a few examples, depicted in Figs. 3 and 4, which were obtained oversampling factors of $q = 1$ (no oversampling) and $q = 0.5$, respectively. These results are representative of the typical behavior observed: as illustrated in Fig. 4, oversampling is quite effective in eliminating undesirable artifacts such as those found in Fig. 3.

3. Parametrization by patching, projection and continuation

Accurate surface representations of complex bodies can indeed be obtained through suitable applications of the Fourier continuation method. As an example we present, in Fig. 5, *Fourier-based representation* of a complete Falcon airplane. The depiction of the Falcon aircraft on the right of Fig. 5 was obtained as a plot of a number of *explicit smooth functions*, each one of which, given by a *Fourier series*, represents a large portion of the aircraft.

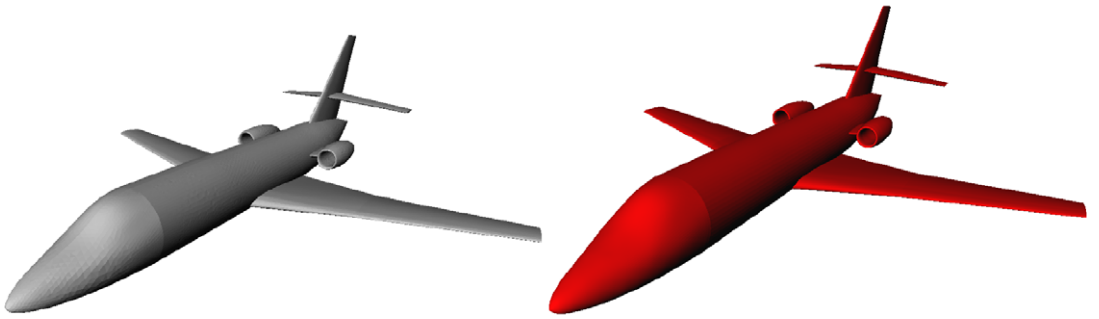


Fig. 5. Left: Falcon airplane, depicted directly from a triangulation obtained from the original CAD representation by means of a classical surface meshing algorithm. Right: Falcon airplane, depicted as a combination of *explicit* smooth functions given by Fourier series, each one of which represents a large portion of the aircraft surface.

Starting from raw point data, (i.e. discrete data such as point clouds, triangulations, etc.), our approach first seeks to extract portions or *patches* of the surface that subsequently are to be parametrized by the continuation method. Patches are to be selected in such a way that there is a simple, possibly closed form projection surface S upon which the discrete data can be projected – so that the given data amounts to a discrete-valued function defined on the surface S. (In the case of point clouds extracted from triangulations or CAD models, since the extracted point clouds do not need to satisfy stringent quality requirements, the procedure of obtaining discrete parametrizations from the original representation could be handled rapidly and effectively by means of an appropriate Graphical User Interface (GUI).) Naturally, it is best to select patches and projection surfaces that give rise to as a smooth variation of the discrete function as possible.

Subsequently, the projection surface S is mapped onto a planar region, again seeking as smooth a mapping as possible. The composition of the maps between the point cloud and the surface S with the map from the surface S to the $u - v$ plane thus gives rise to *discrete parametrizations from discrete sets of points in the* $u - v$ *plane to the given point cloud*. The continuation method is then applied to these discrete maps to produce local *smooth parametrizations* which, appropriately assembled, blended and matched, as explained in the following sections, make up the overall smooth surface representation. Our surface representation approach can thus be summarized as follows:

1. Starting from a discrete geometry representation, first obtain a point cloud that adequately describes the complete surface, and subdivide it into patches suitable for treatment as indicated in point 2 below. The patches may only contain geometric singularities (e.g. slope-discontinuities) at patch boundaries and, further, the points where geometric singularities exist must be labeled accordingly. In the case that the point cloud is obtained from a triangulation or a CAD model, correction of inaccuracies is not necessary at this stage, since such corrections result from use of the continuation method (see points 4 and 5 below as well as Figs. 20 and 21). If needed, intelligent defeaturing, as well as feature reduction and enhancement, which do need to be performed at this stage, can be performed with allowance for a degree of inaccuracy, and without requirements of water-tightness (defects these which, again, are corrected through application of the continuation method and the methodologies mentioned in points 4 and 5 below).
2. For each one of the patches mentioned in point 1 select appropriate (simple) projection surfaces (parabolic surfaces, cylinders, cones, aggregates of simple curves centered around other simple curves, etc.) and use them to produce a *discrete* parametrizations of the point clouds from planar surfaces.
3. To each such parametrization apply the continuation method described in Section 2 to obtain *explicit* functions describing each one of the patches. Assembling all of these functions, explicit (overlapping) parametrizations of the given surface are obtained; see, e.g. Figs. 5 and 14.
4. Overlapping regions which, owing to the discrete nature of the approximation, do not agree exactly, need to be ''blended'', as described in Section 4.
5. Edges, corners and other geometric singularities which, again, owing to the discrete nature of the approximation do not agree exactly, require ''matching'', as described in Section 5 below.

Rather than providing general rules on how items 1–5 can be tackled, we provide details on how this was done to produce the full airplane parametrization shown in Fig. 5; such details are presented in Sections 3.1 and 3.2 below.

3.1. A smooth wing patch

With reference to points 1 to 5 above;

1. The patch we consider in this section is a smooth curved portion of the front of the wing that spans outward (away from the fuselage) from the wing slope-discontinuity line depicted in Fig. 20.
2.
   (i) Fig. 6 right displays the point cloud that describes the wing portion selected and a section of a parabolic projection surface S – the surface S is actually infinite in extent in all directions. Each red point $\mathbf{r}_{\mathrm{cl}}$ in the point cloud is then projected onto the surface S in a direction orthogonal to S; call r_{s} the resulting point on S. This procedure gives a preliminary discrete function defined on S which maps $r_{\mathrm{s}} \to \mathbf{r}_{\mathrm{cl}}$. To produce a discrete function defined on a flat $u - v$ region we then seek a mapping from S onto a plane; in this case it was found that a simple projection onto the $z = 0$ plane was perfectly adequate. In Fig. 7 left we show the result of projecting all the points r_{s} onto this plane. This procedure has thus produced a discrete parametrization of the point cloud depicted in Fig. 6 from the domain depicted on the left portion of Fig. 7. (Of course, the projection onto the $z = 0$ plane mentioned above is only useful after the point cloud has been positioned appropriately, e.g. as shown in Fig. 7 left. Throughout this paper we assume a capability is available that permits to select a region of a given surface, extract points in it, rotate, translate, etc. In practice this can be accomplished by means of some appropriate software, commercial or otherwise.)
   (ii) For future reference we note that the points along the left and right boundaries of the discrete domain in Fig. 6 left correspond to the left and right edges of the patch; the top and bottom boundary points, on the other hand, resulted as a somewhat arbitrary choice was made to pare the back portion of the wing. Notice that the left and right boundary points ought to delineate a smooth curve, corresponding to the true, physical edges. The top and bottom curves, on the other hand, do not delineate a smooth curve

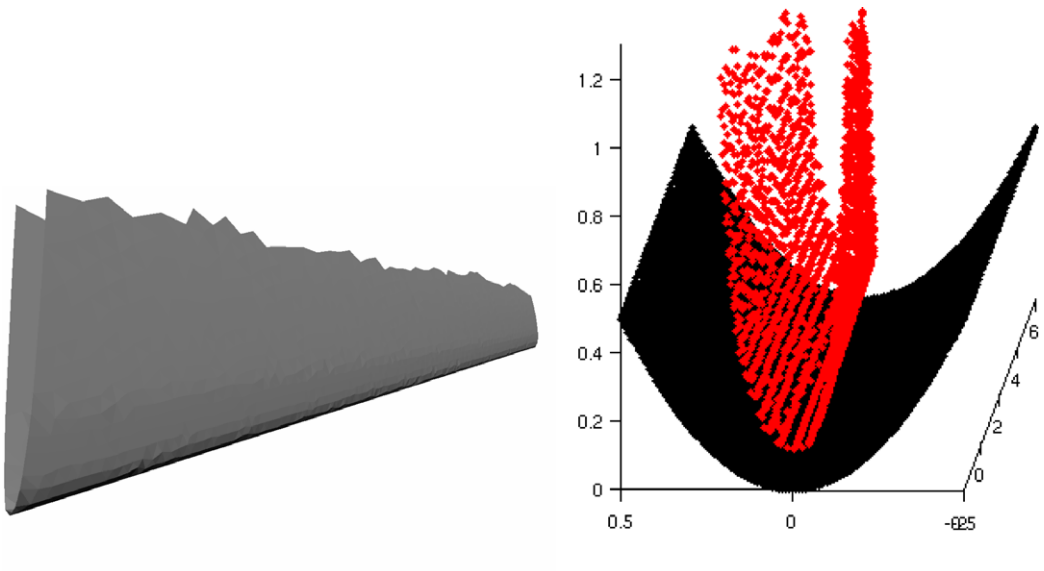


Fig. 6. Left: A wing portion selected from the original mesh representation. Right: Point cloud obtained from the original representation and a parabolic projection surface.

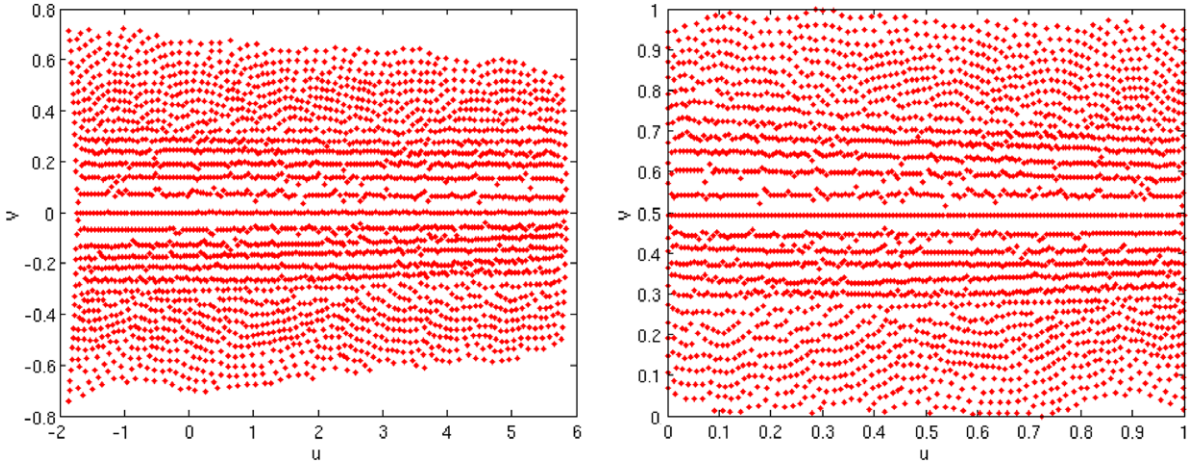


Fig. 7. Discrete $u - v$ parameter plane. Left: before additional $u - v$ transformations. Right: result of $u - v$ transformations applied to the left figure. Note the perfectly straight left and right bounding lines, and the rugged bounding "curves" determined by the top and bottom boundary points.

since these points lie in the interior of wing, away from all physical edges. Call R the smallest rectangle that contains all the points in this $u - v$ projection, and let (u_0, v_0), $(u_0 + H, v_0)$, $(u_0, v_0 + K)$ and $(u_0 + H, v_0 + K)$ be the four vertices of R.

3.

(i) At this point the continuation method could be applied using, say, the continuation region R_{cont} with vertices (u_0,v_0), $(u_0 + 2H, v_0)$, $(u_0,v_0 + 2K)$ and $(u_0 + 2H, v_0 + 2K)$, to yield good representations of the surface under consideration. Prior to this step, however, we introduce additional transformations in $u - v$ space to produce a $u - v$ regions that match a rectangular domain as closely as possible: this procedure simplifies the necessary subsequent trimming, application of the cosine transform (see Appendix B) as well as all construction of partitions-of-unity, surface blending and edge matching (Points 4 and 5). See Remark 3 below.

(ii) Once the discrete parametrization implicit in Fig. 7 right has been obtained we are ready to apply the continuation method using the continuation region R_{cont} to produce a parametrization. We note that there are number of ways in which this could be done. For example, one could directly apply the continuation method to each one of the components (x, y, z) of the discrete map. Or, alternatively, one could apply the continuation algorithm to the function $\rho(u, v) = |\mathbf{r}_{\text{cl}}(u, v) - r_{\text{s}}(u, v)|$. See Remark 4 for considerations concerning the relative advantages of using one or the other approach; here we demonstrate the results we obtained by applying the continuation method to the function ρ. The continuation method was applied to this discrete map (which is defined by a total of 2954 ρ values, from as many (u, v) parameter-space points) using fifteen modes in each, the u and v directions, for a total of 225 Fourier modes in two-dimensional $u - v$ space. Thus we used roughly half as many modes as discretization points in the v direction, but only about one sixth as many modes as discretization points in the u direction. This is in contrast with the comments in Section 2, where we suggested that use of half as many modes as discretization points is usually a reasonable choice. The explanation is simple: the parameter ρ varies rather slowly in the the u direction: use of larger numbers of modes in the u direction leads to no additional improvements in the representation. (We have checked that use of thirty modes, for example, results in an essentially identical parametrization; use of significantly less than fifteen modes, however, gives rise to loss of accuracy. In accordance with the discussion of Section 2.5, in turn, use of inappropriately large numbers of modes, e.g. one-hundred in the u direction in this case, may give rise to poor representations, arising as oscillations near patch boundaries.) The result of this application of the continuation method is shown in Fig. 8 left. Clearly, some departures occur between the upper left portion of the surface produced by the continuation method and the corresponding section of the wing. These

Fig. 8. Left: The initial parametrization of the front of a Falcon airplane. Right: The final parametrization after trimming procedures were applied to the figure on the left.

departures are expected: they are caused by a lack of data in the corresponding region of the $u - v$ plane – around the horizontal lines $v = 0$ and $v = 1$ in the right portion of Fig. 7. This portion of the representation is simply incorrect, and should be trimmed, as indicated in what follows.

(iii) In accordance with the comments in item (i) above, it is easy to trim inaccurate portions of the surface: after trimming the region of $v < .1$ or $v > .9$, the right picture of Fig. 8 is obtained. The lines $u = 0$ or $u = 1$ could not possibly be trimmed: they are physical edges. Fortunately, however, they do not need to be trimmed, since no missing data occurs along an actual physical edge.

Remark 3. Although the continuation method can be applied to data given on an arbitrary planar region, the accuracy of the algorithm can be significantly improved by using data distributed in a region that is roughly rectangular in shape. Note that a one-dimensional version of the continuation method or other approaches can be used to produce approximate bounding curves, and, thus, by means of appropriate subtractions, to transform the domains of definition of various patches into roughly rectangular regions.

Remark 4. Surface approximation based on the distance $\rho = \rho(u, v)$ (see point 3 (ii) above) are generally preferable over approximation of each one of the three functions $\mathbf{r}(u, v) = (x(u, v), y(u, v), z(u, v))$ since, in the former case the inverse of $\mathbf{r}$ (which is needed e.g.; in expression (19) can often be computed in closed form – this is not so in the latter case, for which use of Newton solver thus becomes necessary. An additional (minor) advantage concerns computing times: the computational cost of approximating the single function ρ is, obviously, one-third of that required to approximate the functions (x, y, z).

3.2. Nose patches

1. We now consider a smooth patch of the nose of the airplane, depicted in left portion of Fig. 13.
2. A point cloud of the nose of the airplane, containing 2003 points is depicted in Fig. 9 left; here a projection surface is constructed as a collection of circles centered at the curve depicted within the nose point-cloud; the right portion of the figure shows the corresponding $u - v$ parameter points. Since at $v = 1$ only one data point is available, (the nose tip, $u = 0.5$, $v = 1$), additional data points are included with $0 \leqslant u \leqslant 1$, $v = 1$ the image of all of which is the tip point itself. This procedure does not modify the given point cloud in any way, of course, but it is beneficial in that it provides additional data points for the continuation method along the boundary $v = 1$ in the $u - v$ plane; see Fig. 11.
3. The continuation method is applied to the complete discrete parametrization (augmented at the tip as indicated in point 1 above). We used twenty modes in the u direction, and twenty five modes in the v direction. Since the surface to be parametrized is periodic in the u direction we used $b = 1$; see Section 2.4. In the v direction, in turn, the value $b = 2$ was used. The reduced sampling rate that occurs around $v = 1$ line (compared to the denser sampling around $v = 0$ combined with the fact that $v = 1$ is a boundary for the continuation method gives rise to some oscillations in the area surrounding the nose tip: the data in this region is not sufficient to determine 20×25 Fourier modes. The data points do describe the surface satisfactorily, however, so that the continuation method should reproduce it correctly if used in an appropriate manner. We thus remove from the nose patch just obtained the portion corresponding to $v \geqslant 0.9$, see Fig. 10, and we represent this portion by means of a separate application of the continuation method. We then apply the continuation to the data in the region $0.6 \leqslant v \leqslant 1$, which contains 510 data points. Using

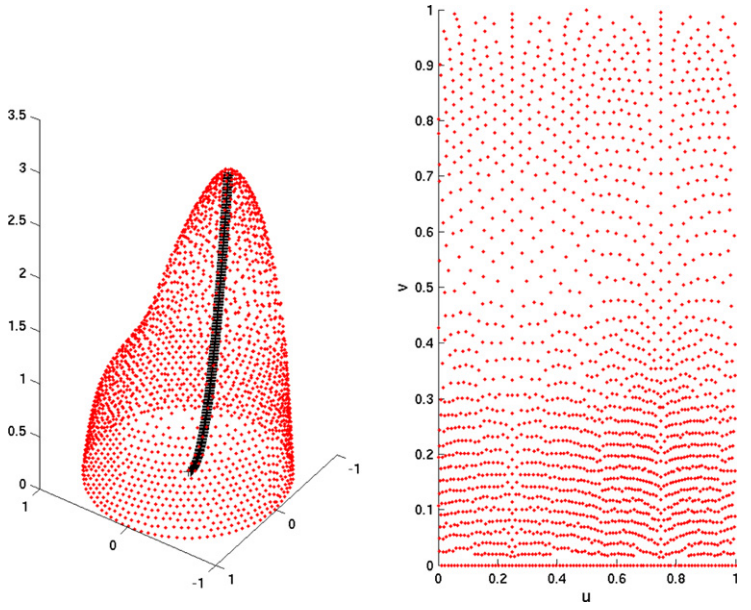


Fig. 9. Discrete $u - v$ parametrization of the nose of a Falcon airplane.

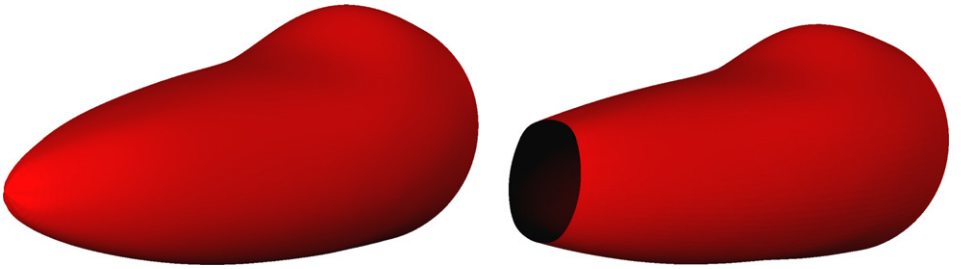


Fig. 10. Left: The initial parametrization of the nose of the Falcon airplane. Right: The final parametrization after the trimming procedures indicated in point 3. above were applied to the parametrization displayed on the left.

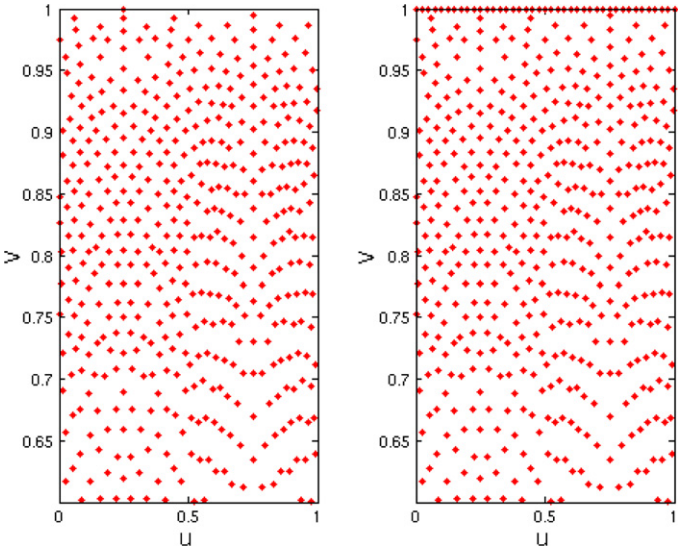


Fig. 11. Discrete $u - v$ parametrization of the tip of the nose of the Falcon airplane. The right figure coincides with the left one, except for a row of points at $v = 1$ on the right figure, all of which are mapped to the nose tip.

eight modes in the u direction and eight modes in the v direction the representation shown in Fig. 12 results; no trimming procedure is necessary in this case. After blending (see Section 4), the combination of the two overlapping patches just constructed gives rise to an adequate representation of the airplane nose – shown in Fig. 13 right. Note that a total of two patches sufficed to parametrize the surface under consideration.

To visualize the sizes of the Fourier patches and the quality and degree of smoothness enjoyed by the parametrizations that result from the Fourier continuation method we present, in Figs. 13, 15, 20 and 21 close-ups of some patches making up the representation of the Falcon airplane. In Fig. 20, in particular, we show a close-up of the wing crease, making evident the beneficial effect the Fourier continuation method can have as a geometry-repair agent. The mesh-repair property of the continuation method is highlighted, once again, in Fig. 21: in this case a single Fourier series, in appropriate coordinates, was used to represent the aft tip of the airplane.

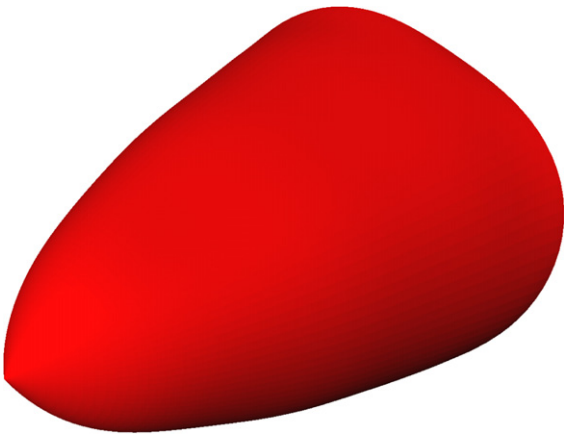


Fig. 12. Parametrization of the nose tip.

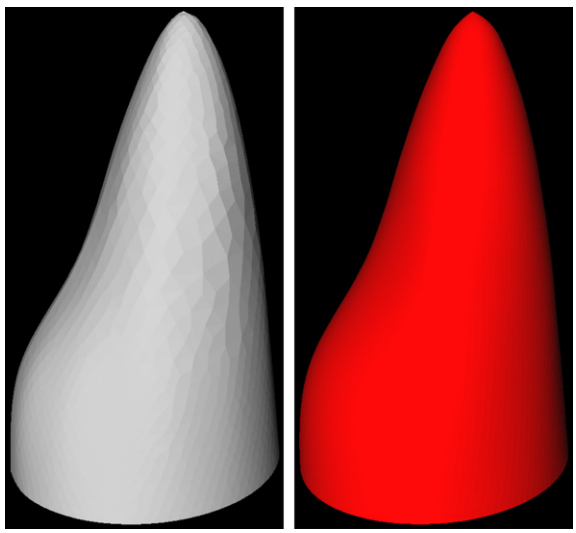


Fig. 13. Left: Original representation of the Falcon aircraft nose. Right: two patch parametrization of the nose produced by the method described in Section 3 after blending using the methodology presented in Section 4.

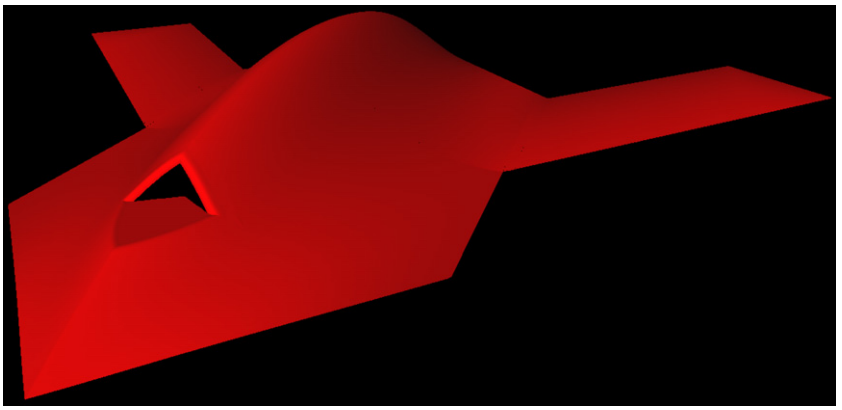


Fig. 14. Parametrization of an unmanned aerial vehicle (UAV).

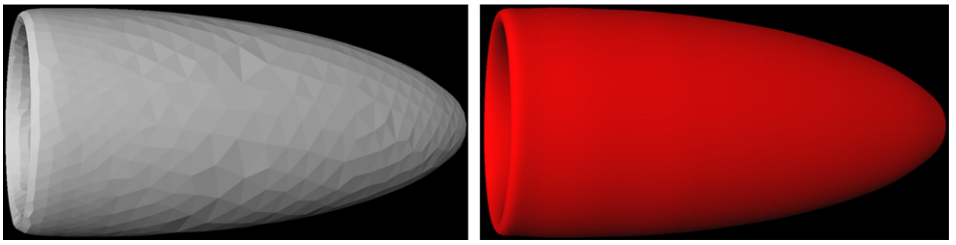


Fig. 15. Two views of the nacelle surface. The full surface is made out of three Fourier series patches, one for the front and two for the back inside and outside.

3.3. Complete patch systems

Proceeding as indicated previously in this section, approximating parametrizations can be constructed that include representations of every portion of a given piecewise smooth surface S; let us assume a number K of patches $\mathcal{P}^j,\ j = 1, \ldots, K$, have thus been produced. Here we summarize the characteristics of the result of such a construction; for generality we phrase our summary to include parametrizations of d-dimensional sets, that is, curves in the plane or in space for $d = 1$ and surfaces in space for $d = 2$. (Clearly, the methodology described in the previous sections applies to both curves and surfaces in space.) Given a surface or curve S our approach thus produces

1. A collection of sets or "patches" $\mathcal{P}^j,\ j = 1, \ldots, K$, which collectively provide approximations of each portion of S.
2. Each patch $\mathcal{P}^j$ is the image of a region $\mathcal{H}^j = \mathcal{U}^j \cup \mathcal{B}^j \subseteq \overline{\mathcal{H}^j}$ contained in $\mathbb{R}^d$ (where $\overline{\mathcal{H}^j}$ denotes the closure of $\mathcal{H}^j$ in $\mathbb{R}^d$) via a smooth and invertible map

   $$\mathbf{r}^j : \overline{\mathcal{H}^j} \to \mathbb{R}^3, \quad j = 1, \ldots, K,$$

   which admits a smooth inverse, and such that for $d = 1$ the derivative of $\mathbf{r}^j$ does not vanish, and for $d = 2$ the vector product

   $$\mathbf{V}^j = \mathbf{V}^j(u^j, v^j) = \frac{\partial \mathbf{r}^j}{\partial u^j} \times \frac{\partial \mathbf{r}^j}{\partial v^j} \tag{14}$$

   is bounded away from zero in $\mathcal{H}^j$. Here for $j = 1, \ldots, K, \mathcal{U}^j \subseteq \mathbb{R}^d$ *is an open set and* $\mathcal{B}^j$ *is a (possibly empty) portion of the boundary set* $\partial\mathcal{U}^j$. The boundary portion $\mathcal{B}^j$ is itself a union of smooth curves for which the algorithms in this section provide explicit parametrizations.
3. Portions of S may be approximated by two or more of the patches $\mathcal{P}^j$.

Owing to numerical errors patches may not (usually do not!) exactly overlap as they should in regions that are approximated by more than one patch. In the next section we describe a methodology to eliminate this problem and produce a fully conforming approximating surface from the collection of patches resulting from the continuation method.

Remark 5. Boundaries $\partial\mathcal{V}$ and closures $\overline{\mathcal{V}}$ of sets $\mathcal{V} \in \mathbb{R}^d$ were considered in point 2 above; the boundary and closure operators used there are those defining closures and boundaries of subsets of the topological space $\mathbb{R}^d$ with its standard topology: $\overline{\mathcal{V}}$ is the set of points in $\mathbb{R}^d$ at zero distance from $\mathcal{V}$ and $\partial\mathcal{V}$ equals the closure of $\mathcal{V}$ minus its interior. The boundary $\partial\mathcal{S}$ and closure $\overline{\mathcal{S}}$ of a set $\mathcal{S} \subseteq \mathcal{P}^j \subseteq \mathbb{R}^3$ are hereby defined as the image under the map $\mathbf{r}^j$ of the boundary and closure within $\mathbb{R}^d$ of the set $(\mathbf{r}^j)^{-1}(\mathcal{S})$, respectively.

4. Overlaps, mismatches and blending: the smooth case

Given a point cloud or some other discrete description of a d-dimensional set in three-dimensional space (a surface for $d = 2$, a curve for $d = 1$), the approach described in Section 3 can be used to produce a description of the complete set by a collection of *overlapping* approximating patches $\{\mathcal{P}^j,\ j = 1, \ldots, K\}$. Each one of these patches represents a portion of the given set, with adjacent patches enjoying a degree of overlap – except along corners, edges and other geometric singularities; see Section 5 for a treatment of such cases. Overlaps are useful in a number of ways; in particular for $d = 2$ they allow for high-order parametrization of surfaces by local patches without recourse to cumbersome surface-partitioning along, say, a class of smooth curves – which may be difficult to produce. And, when used in conjunction with a smooth partition-of-unity, which can be constructed as indicated in what follows, they facilitate evaluation of surface integral and differential operators, solution of nonlinear functional, integral and differential equations, etc; see e.g. [4,26].

As illustrated in Fig. 16 however, numerical errors present in the given description of a d-dimensional set and its approximations necessarily give rise to mismatches amongst the various representations of any overlap portion. Therefore, at overlaps, the approximation of a surface or curve thus constructed is not uniquely specified and a surface that ought to be water-tight may not be so; clearly two undesirable characteristics. In particular, the classical differential-geometry change-of-variables relationship

$$t^k = (\mathbf{r}^k)^{-1}(\mathbf{r}^j(t^j)) \tag{15}$$

that gives the k-th chart coordinates $t^k \in \mathbb{R}^d$ of the point with j-chart coordinates $t^j \in \mathbb{R}^d$ is not meaningful in this case – since, owing to numerical errors, a point $\mathbf{r}^j(t^j)$, which should otherwise lie in the intersection of the j-th and k-th patches, may not even lie in the image of the map $\mathbf{r}^k$; see Fig. 16. Fortunately it is not difficult to

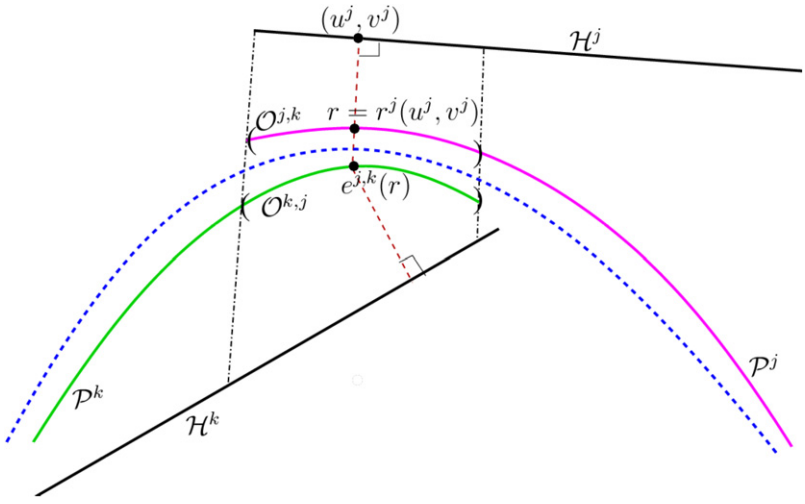


Fig. 16. 1–1 correspondence between patches for overlapping region.

blend the various overlapping parametrizations to produce matching and, if appropriate, water-tight representations while, at the same time, producing an approximate substitute for the change-of-variables relationship (15) as well as a global partition-of-unity. The approach presented in this section, which conceptually follows [13] (although its actual implementation concerning, e.g. explicit equivalence relations, differs from the latter in essential ways), applies to cases in which the underlying surface contains no geometric singularities such as corners, edges, etc.; the methodology is then enhanced in Section 5 to include cases in which geometric singularities exist – which, to our knowledge, had not been considered previously.

4.1. Equivalence relation in a patch system

To carry out the program outlined above we begin by specifying an equivalence between points in regions of different patches. We thus assume that for each $1 \leqslant j, k \leqslant K$ a set $\mathcal{O}^{j,k} \subseteq \mathcal{P}^j$ has been determined (with the interpretation that in absence of error we would have $\mathcal{O}^{j,k} = \mathcal{O}^{k,j} = \mathcal{P}^j \cap \mathcal{P}^k$); in particular, we assume $\mathcal{O}^{j,j} = \mathcal{P}^j$ for all j. Together with these sets we assume "identification" maps $e^{j,k} : \mathcal{O}^{j,k} \mapsto \mathcal{O}^{k,j}$ $(1 \leqslant j, k \leqslant K)$ are provided (in absence of error $e^{j,k}$ would equal the identity map in $\mathcal{P}^j \cap \mathcal{P}^k$). These sets and maps are assumed to satisfy the following requirements:

1. For each $1 \leqslant j, k, \ell \leqslant K$ we have

$$e^{j,j} = \text{Identity}_{\mathcal{P}^j}, \quad e^{j,k} = (e^{k,j})^{-1} \quad \text{and} \tag{16}$$

for $\mathbf{r} \in \mathcal{O}^{j,k}$ such that $e^{j,k}(\mathbf{r}) \in \mathcal{O}^{k,\ell}$ we have $\mathbf{r} \in \mathcal{O}^{j,\ell}$ and

$$e^{k,\ell}(e^{j,k}(\mathbf{r})) = e^{j,\ell}(\mathbf{r}). \tag{17}$$

(It follows from this assumption that, defining a relation $\approx$ in the set $\cup_{1\leqslant j,k\leqslant K}\mathcal{O}^{j,k} = \cup_{1\leqslant j,k\leqslant K}\mathcal{P}^{j,k}$ by $\mathbf{r}_1 \approx \mathbf{r}_2$ if and only if $\mathbf{r}_2 = e^{j,k}(\mathbf{r}_1)$ for some pair (j,k), the relation $\approx$ is an equivalence relation.)

2. The function $e^{j,k}$ maps the set $\partial\mathcal{O}^{j,k} \setminus \partial\mathcal{P}^j$ into $\partial\mathcal{P}^k$; see Remark 5 for the operative definition of the boundary operator ∂.

It is possible to construct equivalence functions $\{e^{j,k}\}_{j,k}$ by means of simple projection surfaces (planar surfaces, parabolic surfaces, surfaces of revolution, etc.) such as those introduced in Section 3. A possible construction is demonstrated in Fig. 16: given a point $\mathbf{r} \in \mathcal{P}^j$ we may define $e^{j,k}(\mathbf{r})$ as the intersection (evaluated by say, Newton's method) of the patch $\mathcal{P}^k$ with the projection line that joins (u^j, v^j) and $\mathbf{r} = \mathbf{r}(u^j, v^j)$. Note that some care must be exercised: for example in order for the second of the conditions (16) to be satisfied in the configuration depicted in Fig. 16 we must use projection lines orthogonal to a single projections surface, say $\mathcal{H}^j$, in both, the definition of $e^{j,k}$ and $e^{k,j}$. In order for condition (17) to be satisfied, analogously, we must use a single projection surface for the definition of all three equivalence functions $e^{j,k}$, $e^{k,\ell}$ and $e^{j,\ell}$. Of course, this may well be unfeasible; in such cases alternative approaches may be used – including, for example, use of composite projection manifolds resulting from smooth combinations of various projection surfaces, etc. Such complications did not arise in any of the examples presented in this paper, however: in all cases use of projection surfaces underlying the patch definitions sufficed to produce the needed overlap equivalence relationships $e^{j,k}$.

4.2. Smooth partition-of-unity on a patch system

Once the functions $e^{j,k}$ are available a partition-of-unity on the non-conforming atlas may be constructed. We define a partition-of-unity associated with the non-conforming atlas $\{(\mathcal{H}^j, \mathcal{P}^j, \mathbf{r}^j),\ j = 1, \ldots, K\}$ to be a set of functions $\{w^j(u^j, v^j), j = 1, \ldots, K\}$, such that

(i) w^j is defined, smooth and non-negative in $\mathcal{H}^j$, and it vanishes together with all of its derivatives in $\partial\mathcal{H}^j$
(ii) For each j and for $\mathbf{r} \in \mathcal{P}^j$ we have

$$\sum_{\{k:\mathbf{r}\in\mathcal{O}^{j,k}\}} w^k((\mathbf{r}^k)^{-1}(e^{j,k}(\mathbf{r}))) = 1$$

Such partition-of-unity can be constructed by utilizing functions W^j, $j = 1, \ldots, K$ satisfying

1. W^j is a non-negative infinitely smooth function defined in $\mathcal{H}^j$
2. W^j vanishes together with all of its derivatives on $\partial\mathcal{H}^j \setminus \mathcal{B}^j$
3. For each j and each $t^j \in \mathcal{H}^j$ at least one of the quantities $W^k((\mathbf{r}^k)^{-1}(e^{j,k}(\mathbf{r}^j(t^j))))$, $(k = 1, \ldots, K)$ does not vanish.

Indeed, if functions W^j satisfying points (1)–(3) are available then the functions w^j given by

$$w^j(t^j) = W^j(t^j) \Big/ \sum_{\{k:\mathbf{r}^j(t^j)\in\mathcal{O}^{j,k}\}} W^k((\mathbf{r}^k)^{-1}(\mathbf{e}^{j,k}(\mathbf{r}^j(t^j)))) \tag{18}$$

form a partition-of-unity. Functions W^j satisfying points (1)–(3) above can easily be constructed by means of exponentials in square domains, star-shaped domains, etc.; see e.g. [4]. For efficiency it is often preferable to use partitions-of-unity with small derivatives, which can be arranged by allowing for substantial overlap of the patches $\mathcal{P}^j$.

4.3. Blending of patch systems

Once a partition-of-unity has been constructed it is an easy matter to modify the parametrizations $\mathbf{r}^j$ so that their images match perfectly at overlaps. Indeed, defining

$$\tilde{\mathbf{r}}^j(t^j) = \sum_{\{k:\mathbf{r}^j(t^j)\in\mathcal{O}^{j,k}\}} \mathbf{e}^{j,k}(\mathbf{r}^j(t^j)) w^k((\mathbf{r}^k)^{-1}(\mathbf{e}^{j,k}(\mathbf{r}^j(t^j)))), \tag{19}$$

$\widetilde{\mathcal{P}^j}$ = Image of $\tilde{\mathbf{r}}^j$ and $\widetilde{S}$ to equal the union of $\widetilde{\mathcal{P}^j}, j = 1, \ldots, K, \widetilde{S}$ is a classical differentiable manifold with (conforming) atlas

$$\{(\mathcal{H}^j, \widetilde{\mathcal{P}^j}, \tilde{\mathbf{r}}^j),\ j = 1, \ldots, K\}.$$

Examples of a blended surfaces resulting from the method described above are shown in Figs. 17 and 18; the match in the overlap regions is exact to machine precision.

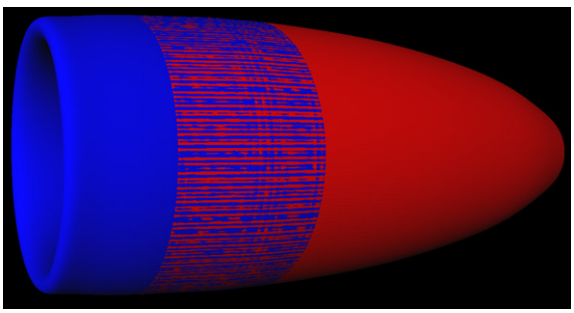


Fig. 17. Blending of smooth surfaces.

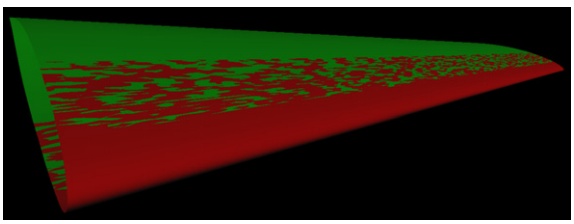


Fig. 18. Blending of smooth surfaces.

5. Blending of singular surfaces

As mentioned above, at points where slope-discontinuities of a surface exist a special treatment is necessary: for any reasonable truncation size a single finite Fourier series cannot accurately reproduce a sharp edge, corner, etc. Clearly, however, the parametrization algorithm described in Section 3 can be used to produce two or more *smooth* vector valued representations for the smooth surfaces that meet at a geometric singularity: a line where the slope is discontinuous, for example, is thus given as the intersection of two smooth surfaces. For water-tightness, the boundaries of the various surfaces meeting at singularities must match *exactly* – to machine precision. The surfaces obtained by the methods of Section 3 only represent the corresponding body within the accuracy inherent in the given point cloud, however, and they are therefore bound to differ by that amount at singularities. As in the case of patch overlaps considered in the previous section this presents a difficulty: water-tightness is essential for adequate numerical resolution of partial differential equations throughout a domain boundary, and particularly so around edges, corners, etc.

The problem of matching at singularities can be solved by means of appropriately defined error-correcting functions defined in all the patches containing geometric singularities. We detail the procedure for curves first and then, *using* corner-matched singularity *curves*, we introduce a methodology for matching of surface singularities.

5.1. Corner-matching for curves

We consider a curve for which a complete patch system was produced by the methods of Section 3, with overlaps blended by the methods of Section 4, and, without loss of generality, we assume the patches defined by the parametrizations $\mathbf{r}^j(t^j)$ $(a^j < t^j \leqslant b^j)$ and $\mathbf{r}^k(t^k)$ $(a^k < t^k \leqslant b^k)$, which should have intersected at a corner point for $t^j = b^j$ and $t^k = b^k$, do not actually intersect there: we have

$$\mathbf{r}^j(b^j) \neq \mathbf{r}^k(b^k).$$

The difference between these points is small: of the order $\mathcal{O}(\varepsilon)$ of the error inherent in the curve representations.

To proceed we introduce a new point $\mathbf{r}_c$ which is to become the exact corner point in the matched representation. There is considerable latitude in the choice of the point $\mathbf{r}_c$; the choices $\mathbf{r}_c = \mathbf{r}^j(b^j)$, $\mathbf{r}_c = \mathbf{r}^k(b^k)$ or $\mathbf{r}_c$= any other point in a neighborhood of these points of a diameter $\mathcal{O}(\varepsilon)$ is generally equally adequate.

The matching procedure is completed using smooth windowing functions $p^j = p^j(t^j)$ and $p^k = p^k(t^k)$ which, defined in the intervals $a^j < t^j \leqslant b^j$, satisfy $p^j(b^j) = 1$ and $p^k(b^k) = 1$, and which vanish a small distance (say, of the order of 10% of the corresponding interval lengths $b^j - a^j$ and $b^k - a^k$) away from the points b^j and b^k, respectively. These windowing functions allow us to construct error-correcting functions

$$e^j(t^j) = (\mathbf{r}_c - \mathbf{r}^j(b^j))p^j(t^j) \quad \text{and} \quad e^k(t^k) = (\mathbf{r}_c - \mathbf{r}^k(b^k))p^k(t^k). \tag{20}$$

Clearly, the parametrizations $\mathbf{r}^j(t^j) + \mathbf{e}^j(t^j)$ and $\mathbf{r}^k(t^k) + \mathbf{e}^k(t^k)$ are perfectly matched at the corner point.

5.2. Singularity matching for surfaces

A generalization of the corner-matching methodology for curves described in Section 5.1 can be used to produce perfectly matched representations of surfaces along edges and other geometric singularities. The corner-matching method considered above proceeds by constructing error-correcting functions (20) – which correct departures that prevent preliminary parametrizations from meeting at corners. As discussed above, within the margins of the numerical tolerance there is significant freedom in the selection of the corner point $\mathbf{r}_c$. In the present problem of matching surfaces along edges, corners and other singularities, in turn, we must identify appropriate matching curves and points – whose selection, like that of corner points $\mathbf{r}_c$, involves a degree of flexibility.

Our singularity matching methodology for surfaces can be summarized as follows:

1. Using the methodology described in Section 5.1, parametrize, blend and corner-match all edge curves.
2. Using projection surfaces, establish a correspondence (akin to that discussed in Section 4 for overlapping surfaces) between boundaries of surface patches and edges parametrized as indicated in point 1; note that a point on an edge may project (correspond) with points in two or more patches. In this approach *the projection of an edge curve on one patch defines the trimming curve on that patch*.
3. Use correction functions analogous to those defined by Eq. (20) to make all patches match the edges.
4. After all edges are matched blend all surfaces using smooth windowing functions, in a direct generalization of the procedure detailed in Section 4.

This procedure gives rise to edges that are perfectly matched at corners, and surfaces that are perfectly matched at edges and corners.

Fig. 19 shows two slightly different on-edge views of the wing around the quadruple slope-discontinuity point, demonstrating the results of our edge matching technique can offer: edges are matched to machine precision. Fig. 22 shows the image of Cartesian grids under the parametrization maps. Similar methods have been applied to all surface representation problems we considered in regions where slope discontinuity lines or points exist. Other types of singularities can be addressed in a similar manner; for example the

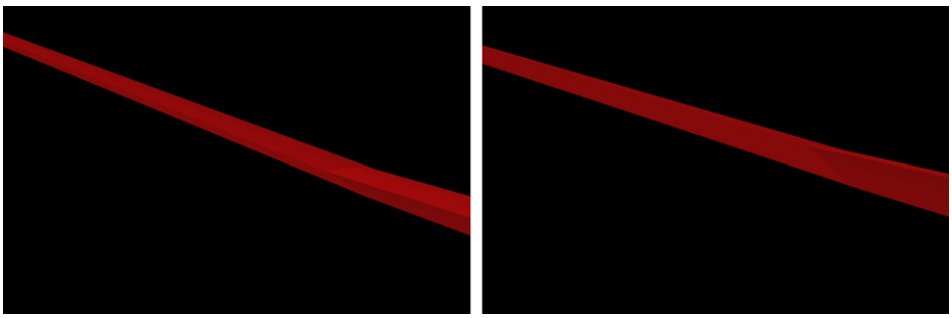


Fig. 19. Two slightly different on-edge views of the wing around the quadruple slope-discontinuity point, demonstrating the effectiveness of the edge matching technique to produce perfectly sharp edges and water-tight geometries.

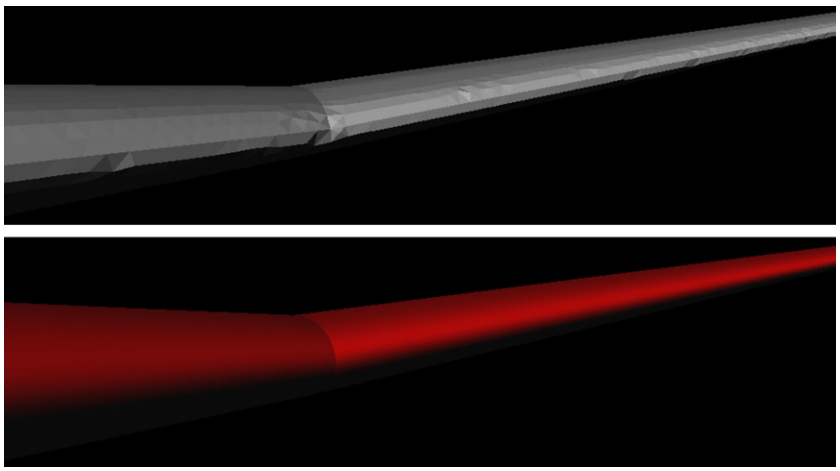


Fig. 20. Repair of bad-quality triangulation by means of the continuation method. Top: Original triangulation of the slope-discontinuity area of the wing, showing poor triangulations. Bottom: Result of an application of the continuation method to this geometry, making evident the beneficial effect of the Fourier continuation method as a geometry-repair agent.

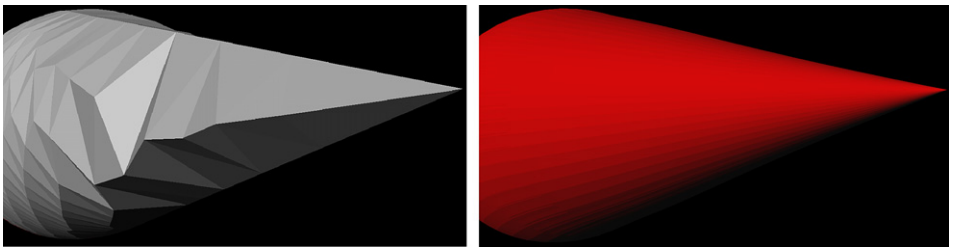


Fig. 21. Close-ups on a large portion of the unstructured and parametrized representations of the fuselage tail.

corner at the aft tip of the aircraft can be effectively resolved by using a cone as a projection surface and appropriate oversampling, in such a way that a number of different points in the $u - v$ plane correspond to the corner in the cone. A constant boundary condition along the line corresponding the fuselage tip is enforced. To produce a sufficiently large data set three copies of the $u - v$ plane were placed side by side in the angular direction u and the composite discrete parametrization, which turns around the conical section three times, was used as input of the continuation method. The result of this procedure is shown in Fig. 21 – which, in addition to demonstrating the effectiveness of the conical algorithm, shows, in a different context than considered previously in this text, the effectiveness of the continuation method to repair defects in point-cloud inputs.

Acknowledgements

This work was supported in part by the Air Force Office of Scientific Research, the National Science Foundation and the National Aeronautics and Space Administration.

Appendix A. Evaluation algorithm

It is interesting to note that a procedure related to that used in the main body of this paper to approximate point clouds can also be used to evaluate with spectral accuracy truncated Fourier series of non-periodic smooth functions. Again, the idea is to find a continuation of the given function to a smooth and periodic function in a larger interval. We note that, although the approximation algorithm described in Section 2 and evaluation algorithm presented here share a conceptual basis, the actual details of the two algorithms differ significantly.

We describe our procedure using our usual spatial/frequency extension factors $b = 2$ and $q = 0.5$; see Section 2.4. Let $\hat{p}_k$ be the exact Fourier coefficients of a non-periodic C^n function $p(x)$ (n an integer or $n = \infty$) defined on the unit interval $[0,1]$. If $\hat{q}_\ell$ are the Fourier coefficients of some continuation $q(x)$ of $p(x)$ to the interval $[0,2]$, then the corresponding Fourier coefficients of the continuation $q(x) = \sum_\ell \hat{q}_\ell e^{\pi i \ell x}$ restricted to the interval $[0,1]$ can be determined analytically: The functions $p(x)$ and $q(x)$ will coincide in the interval $[0,1]$ if and only if

$$\hat{p}_k = \int_0^1 q(x) e^{-2\pi i k x} dx = \sum_\ell \hat{q}_\ell \int_0^1 e^{\pi i (\ell - 2k) x} dx = \sum_\ell D_{k,\ell} \hat{q}_\ell \tag{21}$$

where

$$D_{k,\ell} = \begin{cases} 1 & \text{if } 2k = \ell \\ 2i/\pi(\ell - 2k) & \text{if } \ell \text{ odd} \\ 0 & \text{otherwise.} \end{cases} \tag{22}$$

Clearly, the values $\hat{q}_\ell$ can be obtained through solution of the linear system

$$D\hat{q} = \hat{p} \tag{23}$$

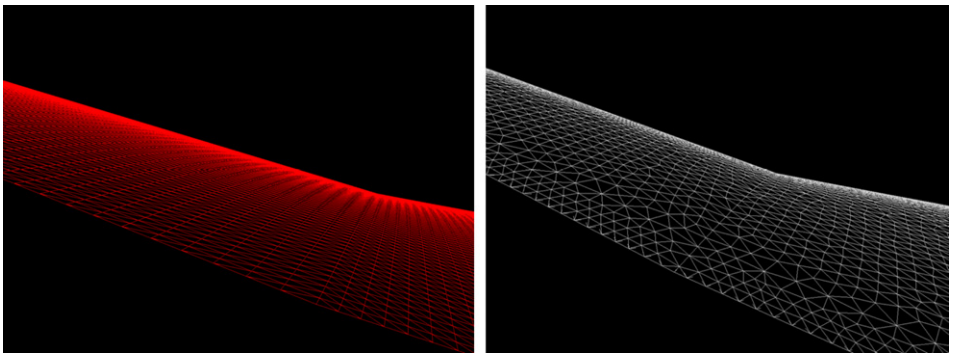


Fig. 22. Left: Logically Cartesian surface mesh induced by the parametrization of the surface wing around the slope-discontinuity line. Right: unstructured mesh. Clearly, the logically Cartesian mesh can be refined easily near the edges, by simple incorporating additional Cartesian lines in parameter-space, as needed.

for the coefficients $\hat{q}$. If an over-determined system is used, a solution of this problem is obtained in a least-squares sense. To evaluate p in the interval $[0,1]$ we sum a truncated Fourier series for the function q – whose Fourier coefficients $\hat{q}_k$ were obtained as the solution of the problem (23).

A.1. Evaluation problem: implementation and numerical results

An example applying the periodic continuation idea to the evaluation problem is demonstrated by the evaluation of a truncated Fourier series for the non-periodic function $f(x) = x$. Of course, the direct summation of the truncated Fourier series

$$\hat{f}_k = \begin{cases} 1/2 & k = 0 \\ i/(2\pi k) & k \neq 0 \end{cases} \tag{24}$$

suffers from the Gibbs phenomenon. In contrast, using the coefficients obtained for the continuation to a periodic function on $[0,2]$ does not exhibit such oscillations. This is demonstrated by the plots in Fig. 23, while the error results in Table 7 display the expected super-algebraic convergence of the method.

More generally, the continuation method can be applied to non-periodic functions with additional discontinuities. To demonstrate this we consider the function

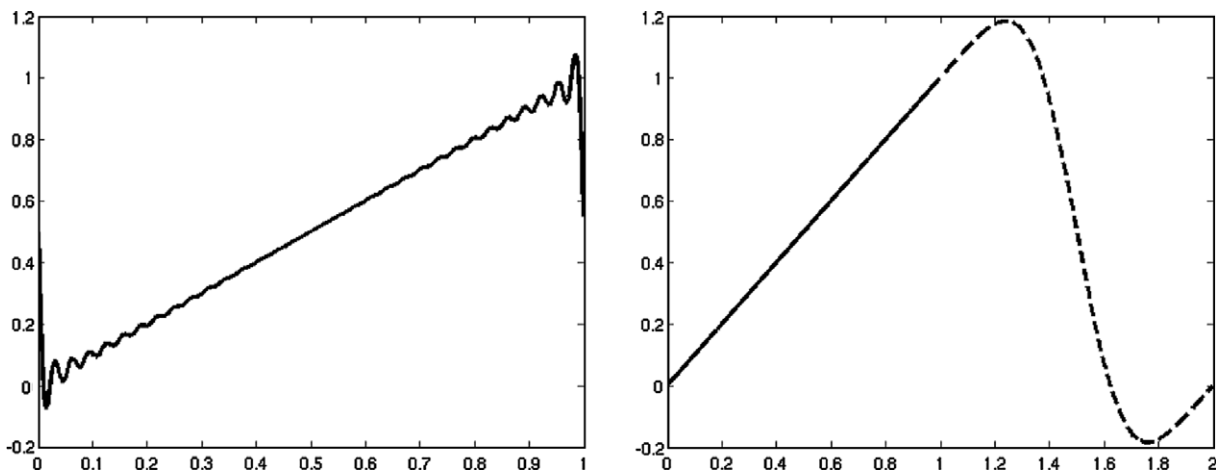


Fig. 23. Left: Evaluation of the 128-mode truncated Fourier series of the function $f(x) = x$ in the interval $[0,1]$. Right: evaluation of the C^∞ continuation of f to the interval $[0,2]$ obtained from the same 128 Fourier coefficients (resolving the Gibbs Phenomenon).

Table 7
Errors for the solution of the evaluation problem for the function $f(x) = x$ on the basis of N given Fourier coefficients $\hat{f}_0 = 1/2, \hat{f}_k = i/(2\pi k)$ for $k \neq 0$

M	N	Error	Ratio	d/dx err	Ratio	d^2/dx^2 err	Ratio
4	8	5.0×10^{-2}		7.5×10^{-1}		5.3×10^{0}	
8	16	2.0×10^{-3}	2.5×10^{1}	8.7×10^{-2}	8.6×10^{0}	2.2×10^{0}	2.4×10^{0}
16	32	3.8×10^{-6}	5.3×10^{2}	4.6×10^{-4}	1.9×10^{2}	3.7×10^{-2}	5.9×10^{1}
32	64	1.5×10^{-11}	2.5×10^{5}	5.0×10^{-9}	9.1×10^{4}	1.2×10^{-6}	3.0×10^{4}
64	128	3.1×10^{-14}	4.8×10^{2}	2.1×10^{-11}	2.4×10^{2}	1.1×10^{-8}	1.1×10^{2}

M denotes the number of Fourier coefficients used for the continuation of f to the interval $[0, 2]$.

$$f(x) = \begin{cases} x & x \in [0, \pi/8) \\ 1/2 - x & x \in [\pi/8, 1] \end{cases} \tag{25}$$

whose Fourier coefficients can be calculated exactly and whose continuation is shown in Fig. 24. Each piecewise smooth portion of the function is continued smoothly to a periodic function defined on a larger interval, and the convergence is super-algebraically fast, as demonstrated by Table 8. As a final example we consider the discontinuous non-periodic function

$$f(x) = \begin{cases} \sin(\cos(2x - 1/2)) & x \in [0, 1/2) \\ 0 & x \in [1/2, 1] \end{cases} \tag{26}$$

with Fourier coefficients calculated numerically. Table 9 demonstrates the expected super-algebraic convergence of the continuation method (see Fig. 25).

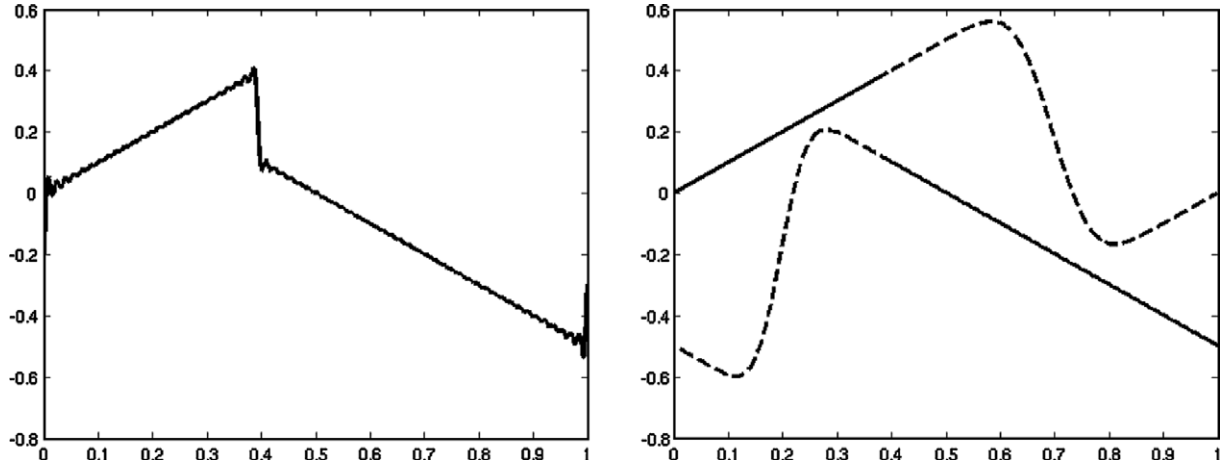


Fig. 24. Left: Evaluation of the 128-mode truncated Fourier series of the piecewise smooth function 25. Right: Evaluation of the C^{∞} continuations of the two branches of f obtained from the same 128 Fourier coefficients (resolving the Gibbs Phenomenon).

Table 8
Errors for the solution of the evaluation problem for the function (25), using N given Fourier coefficients

M_1	M_2	N	Error	Ratio	d/dx err	Ratio	d^2/dx^2 err	Ratio
6	10	32	1.6×10^{-3}		6.1×10^{-2}		1.3×10^{0}	
12	20	64	4.3×10^{-6}	3.8×10^{2}	4.7×10^{-4}	1.3×10^{2}	3.4×10^{-2}	3.9×10^{1}
24	40	128	3.7×10^{-11}	1.1×10^{5}	1.2×10^{-8}	4.0×10^{4}	2.7×10^{-6}	1.3×10^{4}
48	80	256	6.3×10^{-14}	5.8×10^{2}	5.9×10^{-11}	2.0×10^{2}	3.4×10^{-8}	7.7×10^{1}

M_1 and M_2 denote the numbers of Fourier modes used for the continuation of the left and right branches of the given function.

A.2. Comparison with polynomial interpolation and other methods for resolution of the Gibbs phenomenon

Comparison with high-degree polynomial interpolation. The properties of the continuation method differ significantly from those of high-order polynomial interpolation; to demonstrate this we consider the problem of approximating the famous Runge function $y = \frac{1}{1+25x^2}$ for $-1 \leqslant x \leqslant 1$. Table 10 displays results of applications of the continuation method and polynomial approximation to the Runge function; the upper portion of this table shows results obtained by approximations *with oversampling*, while the lower portion displays results that result from approximations *not using oversampling*. The use of oversampling alleviates the severity of the Runge phenomenon for polynomial interpolation but does not eliminate it: in neither case does the polynomial approximation converge as N is increased. The continuation method converges in both cases: with and without oversampling.

Table 9
Errors for the solution of the evaluation problem for the function 26, using N given Fourier coefficients

M_1	M_2	N	Error	Ratio	d/dx err	Ratio	d^2/dx^2 err	Ratio
4	4	16	5.0×10^{-3}		8.3×10^{-2}		6.8×10^{-1}	
8	8	32	9.2×10^{-5}	5.5×10^{1}	4.4×10^{-3}	1.9×10^{1}	1.2×10^{-1}	5.6×10^{0}
16	16	64	7.9×10^{-8}	1.2×10^{3}	1.1×10^{-5}	4.2×10^{2}	9.3×10^{-4}	1.3×10^{2}
32	32	128	1.5×10^{-13}	5.3×10^{5}	5.5×10^{-11}	1.9×10^{5}	1.4×10^{-8}	6.6×10^{4}

M_1 and M_2 denote the numbers of Fourier modes used for the continuation of the left and right branches of the given function.

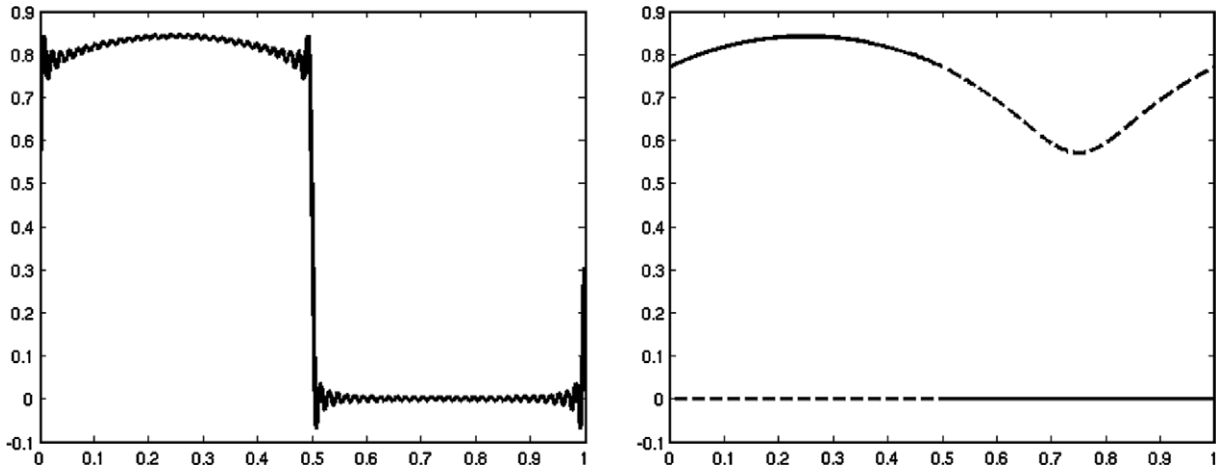


Fig. 25. Left: Evaluation of the 128-mode truncated Fourier series of the piecewise smooth function (26). Right: Evaluation of the C^∞ continuations of the two branches of f obtained from the same 128 Fourier coefficients (resolving the Gibbs Phenomenon).

Table 10
Approximation errors for continuation and polynomial approximation

	$N = 64$	$N = 128$	$N = 256$	$N = 512$
Continuation	5.5×10^{-3}	1.0×10^{-4}	5.9×10^{-9}	2.3×10^{-13}
Polynomial	1.4×10^{-2}	1.8×10^{-2}	1.5×10^{-2}	1.4×10^{-2}
Continuation	6.7×10^{-1}	1.4×10^{-5}	5.9×10^{-11}	2.6×10^{-10}
Polynomial	1.5×10^{8}	1.1×10^{19}	1.0×10^{41}	1.4×10^{85}

Upper table: Continuation and polynomial approximation using oversampling, using N data points and $N/2$ Fourier/polynomial coefficients, respectively. Lower table: Continuation and polynomial *interpolation* (without use of oversampling), using N data points and N Fourier/polynomial coefficients, respectively.

Comparison with other methods for the resolution of the Gibbs phenomenon. In effect, both the continuation based approximation and evaluation algorithms we have considered in this paper provide a resolution of the well known Gibbs phenomenon – that is, these methods eliminate the oscillatory behavior exhibited by Fourier series of discontinuous functions at points of discontinuity, and the associated slow convergence throughout the interval of periodicity. The Gibbs phenomenon has attracted much attention over the last quarter century; here we mention work based on filtering of high-order Fourier coefficients [27]; integration rule based on Chebyshev-like quadrature points [29]; smoothing [23]; Gegenbauer polynomials [22,24]; built in singularities [19,20]. Of these approaches, the Gegenbauer-polynomial method is the only one that, like ours, resolves the Gibbs phenomenon throughout the interval of definition of the function, including the endpoints of the interval *without* constructing and/or using the jumps of the function and its derivatives at the edges. This is the type of algorithm we need in our problem of interpolation of surfaces around edges, and we thus may confine our comparative discussion to references of the Gegenbauer-polynomial method.

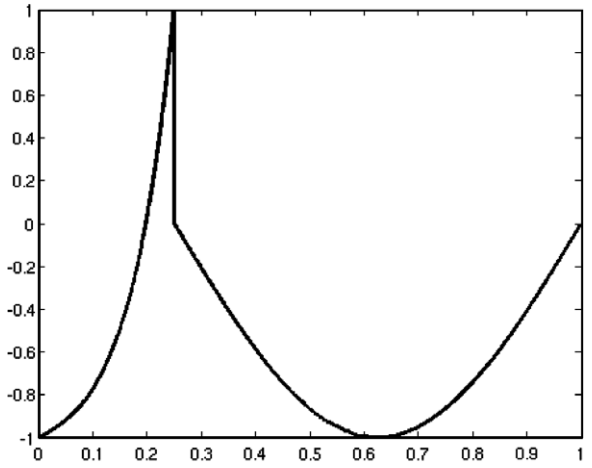
Before moving to explicit performance comparisons we point out a few issues of interest. We note that, (i) although the theoretical considerations presented in Section 2.3 provide evidence for the rapid convergence of the continuation method, the discussion of that section (the only theoretical discussion of the continuation method of which we are aware) falls short of a proof of convergence. In contrast, convergence proofs have been given for the Gegenbauer approach; see [24] and references therein. In addition, (ii) the present implementation of the continuation method is based on a direct least-square solve requiring $\mathcal{O}(n^3)$ operations, while implementations of the Gegenbauer approach have been given with a cost of $\mathcal{O}(n\log(n))$ operations. With regards to point (ii) we mention that we have produced robust $\mathcal{O}(n\log(n))$ implementations of the continuation method – on the basis of iterative least-square solvers [1]. A description of the accelerated methodology is not incorporated in the present text since, for the small values of n relevant to the work considered in this paper, such accelerated algorithms offer limited benefits. With regards to point (i), on the other hand, we suggest that such proofs should be pursued – in view of the excellent convergence properties of the continuation method demonstrated throughout this text and, in particular, in following set of comparisons.

Fortunately it is not difficult to find a good basis for comparison between the continuation method and the Gegenbauer approach: in Fig. 3 Right of reference [21], for example, a demonstration is provided of the approximation properties of the Gegenbauer-polynomial method through application to non-smoothly-periodic sinusoids with frequencies $\omega = 1.4$, 2.4, 3.4 and 4.4. By consideration of that figure we see that, indeed, the *error* arising from the Gegenbauer-polynomial method actually increases substantially (*to a value of* 10,000) before it starts its exponential decrease. No such undesirable behavior is observed in the results provided by the continuation method, which, in this case, actually exhibits a monotone super-algebraic decrease and yields full double-precision accuracy through use of approximately 80 Fourier modes. Use of a full 125 modes with the Gegenbauer-polynomial method, in turn, gives rise to errors of the order of 10^{-2} for the highest frequency. Even more importantly, use of a small number of modes (20–40) in the continuation method gives rise to useful interpolations, while, in this regime, the corresponding results given by the Gegenbauer-polynomial method contain 100% errors at best, and 1,000,000% errors in some cases. The more recent

Table 11
Comparison between errors for the solution of the evaluation problem for the interpolation of a discontinuous non-periodic function in Fig. 26 using N Fourier coefficients to determine continuations from the subintervals $[0,1/4)$ and $[1/4,1]$

N	Continuation	Gegenbauer	Freud
16	8.0×10^{-2}	–	–
32	1.0×10^{-2}	–	–
64	1.1×10^{-3}	6.1×10^{-1}	8.9×10^{-1}
128	2.6×10^{-6}	5.7×10^{-1}	1.4×10^{-1}
256	3.9×10^{-12}	1.3×10^{-4}	1.8×10^{-4}
512	5.3×10^{-13}	1.5×10^{-6}	1.0×10^{-7}
1024	1.8×10^{-12}	1.4×10^{-7}	5.3×10^{-13}
2048	4.5×10^{-13}	9.0×10^{-7}	5.2×10^{-14}

Second column is obtained using $N/4$ for M_1 and M_2 Fourier modes, respectively. Second and third columns are reproduced from reference [25].



$$f(x) = \begin{cases} \frac{2\exp(4\pi x) - 1 - \exp(\pi)}{\exp(\pi) - 1} & \text{if } x < .25 \\ -\sin(4\pi x/3 - \pi/3) & \text{if } x \geq .25 \end{cases}$$

Fig. 26. Evaluation of the truncated Fourier series with 128 modes for a discontinuous non-periodic function.

Gegenbauer and Freud results presented in Table 11 for the function displayed in Fig. 26 above (see Ref. [39]) are consistent with this observation; note the favorable performance exhibited by the continuation method in this case. (The Freud method is a variant of the Gegenbauer polynomial which, unlike the original algorithm, is stable for large values of N.) For two-dimensional problems, finally, all three of these methods require the discretization domain to be a square in the plane – a condition not required by the continuation approach. This fact is central in our surface representation applications of the continuation method; see e.g. the right graph in Fig. 7.

Appendix B. Chebyshev transform

Unlike a function's Fourier coefficients, the series coefficients produced by the continuation method cannot be bounded in terms of the maximum absolute value of the function approximated: the continuation coefficients can be quite large. This fact can be appreciated by considering the column labeled $C_{\max}$ in the upper portion of Table 12, which displays the maximum absolute values of the continuation coefficients of a certain function for various discretization sizes N. (The function used in this example was chosen as one for which this effect is particularly noticeable; for the function $y = x$ and using our usual prescriptions, a

Table 12
Errors in continuation–approximations of the function $y = \exp(\sin(5.4\pi x - 2.7\pi) - \cos(2\pi x))$ in the interval $[0,1]$ – with continuation interval $[0,2]$ and numbers N and $M = N/2$ of discretization points and modes, respectively

N	Error	Ratio	d/dx err	Ratio	d^2/dx^2 err	Ratio	$C_{\max}$	Δy_1	Δy_2
16	3×10^{-1}		7×10^{-1}		8×10^{-1}		5×10^{1}	2×10^{-13}	2×10^{-4}
32	6×10^{-2}	5×10^{0}	8×10^{-1}	9×10^{-1}	3×10^{0}	3×10^{-1}	3×10^{3}	9×10^{-13}	5×10^{-4}
64	8×10^{-3}	7×10^{0}	4×10^{-1}	2×10^{0}	5×10^{0}	6×10^{-1}	10×10^{7}	6×10^{-8}	5×10^{-4}
128	9×10^{-7}	9×10^{3}	6×10^{-5}	8×10^{3}	2×10^{-3}	3×10^{3}	1×10^{8}	1×10^{-7}	5×10^{-4}
256	1×10^{-13}	7×10^{6}	3×10^{-11}	2×10^{6}	2×10^{-9}	1×10^{6}	2×10^{0}	5×10^{-13}	5×10^{-4}
16	3×10^{-1}		7×10^{-1}		8×10^{-1}		1×10^{0}	1×10^{-13}	1×10^{-4}
32	6×10^{-2}	5×10^{0}	8×10^{-1}	9×10^{-1}	3×10^{0}	3×10^{-1}	1×10^{0}	5×10^{-15}	6×10^{-6}
64	8×10^{-3}	7×10^{0}	4×10^{-1}	2×10^{0}	5×10^{0}	6×10^{-1}	1×10^{0}	3×10^{-14}	3×10^{-5}
128	9×10^{-7}	9×10^{3}	6×10^{-5}	7×10^{3}	4×10^{-3}	1×10^{3}	1×10^{0}	3×10^{-14}	3×10^{-5}
256	1×10^{-13}	7×10^{6}	4×10^{-11}	2×10^{6}	9×10^{-9}	5×10^{5}	1×10^{0}	3×10^{-14}	3×10^{-5}

Uniform discretizations. Upper table: unmodified continuation method. Lower table: continuation method followed by Chebyshev expansion in the interval $[0,1]$. $C_{\max}$ denotes the maximum absolute value of the coefficients in the expansions via continuation (upper table) and continuation-Chebyshev (lower table). $\Delta y_1 = |f_{\text{approx}}(.5 + 10^{-14}) - f(.5)|$ and $\Delta y_2 = |f_{\text{approx}}(.5 + 10^{-5}) - f(.5)|$, where f_{approx} is the continuation approximation (upper table) or the continuation-Chebyshev approximation (lower table).

double-size continuation interval and $M = N/2$, for example, the continuation coefficients are uniformly bounded by one for all the values of N needed to reach machine precision accurate approximations of the function.) Remarkably, in spite of the large values of the continuation coefficients, the approximations of the function and its derivatives are of high-quality throughout the approximation interval $[0, 1]$: the maximum errors displayed in Table 12, which were evaluated through comparison of the approximations with the exact values at 10,000 points, make an eloquent case in these regards.

For cases in which use of the continuation method does give rise to large coefficients (as is the case, say, for $N = 32$, $N = 64$, $N = 128$ in the upper portion of Table 12) it is possible, if desired, to transform the continuation series into a series representation which, while bearing small coefficients, is as accurate as the original continuation series. Such an alternative series can be obtained, quite simply, by sampling an N-point continuation series at an N-point Chebyshev grid (respectively equispaced grid in the periodic case, such as, e.g. the functions of the u variable in the aircraft nose considered in Section 3.2) and evaluating the corresponding Chebyshev (respectively Fourier) expansion. The results of this procedure are demonstrated in the lower portion of Table 12: the approximations obtained are essentially as accurate as those given by the continuation series. Of course, the Chebyshev coefficients are nicely bounded in terms of the maximum value of the function approximated.

Depending on the application at hand, use of series representations containing small coefficients may or may not be a matter of importance. A simple illustration of the import of use of a series with large coefficients can be appreciated in the two rightmost columns of Table 12, which display the values $\Delta y_1 = |f_{\text{approx}}(.5 + 10^{-14}) - f_{\text{approx}}(.5)|$ and $\Delta y_2 = |f_{\text{approx}}(.5 + 10^{-5}) - f_{\text{approx}}(.5)|$ computed from the continuation series (upper portion) and the Chebyshev series derived from it (lower table). Clearly, owing to the large values of the continuation coefficients and associated cancellation errors, the quantity Δy_1 arising from the continuation series is orders of magnitude larger than both, the actual variation in the function and the variation that would be exhibited by the continuation series under exact arithmetic. The Chebyshev series obtained from the continuation series does not suffer from this difficulty, of course, note that the quantity Δy_2 was correctly computed even without recourse to a Chebyshev re-expansion.

The value of this approach may be significant under certain circumstances. Although, the Chebyshev series is not more accurate than its parent continuation approximation, its use in conjunction with high-order numerical methods may allow for convergent results beyond levels that could be possible by using the continuation method alone. Although this would not result in any better approximations of, say, the solution of an associated physical problem under consideration, it would allow to establish the convergence of the numerical method at hand and, thus, it could be used to relegate any potential uncertainties in numerical results to corresponding uncertainties in the surface representation – which could then be tackled through use of additional measurements, appropriate refinements, etc. In short, this would allow to produce convergent results and help

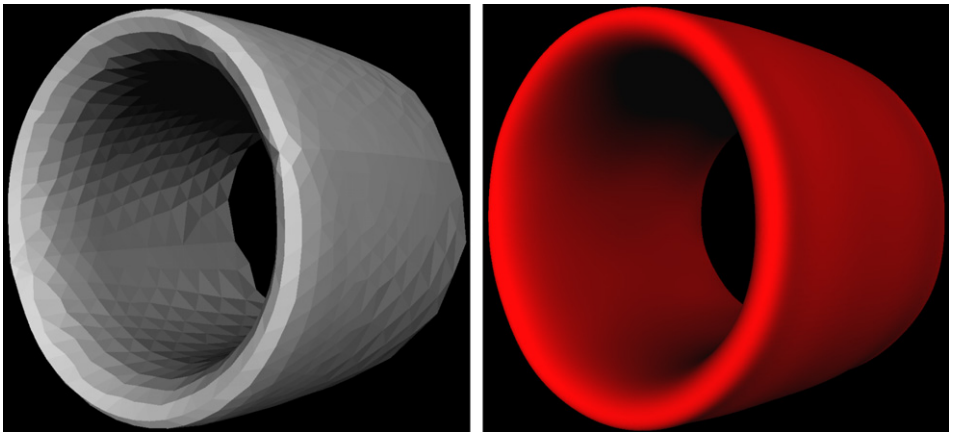


Fig. 27. Left: original nacelle front mesh. Right: a patch for the mesh on the left figure obtained from the continuation method followed by a Chebyshev transform.

focus on the cause of, say, discrepancies between numerics and experiments, as lying not in a numerical solver but rather on the quality of the surface representation.

In Fig. 27 we demonstrate the results of this Chebyshev re-interpolation method on a portion of the aircraft nacelle. Starting from the mesh depicted in Fig. 27 left, we produce a single patch, depicted in Fig. 27 right. The magnitude of Fourier coefficients arising from the continuation method is of the order of 10^6; after use of a Chebyshev transform, the results of which are rendered in Fig. 27 right, the magnitude of the Chebyshev coefficients is of order 1.

References

[1] A. Bjorck, Numerical Methods for Least Squares Problems, Soc. Ind. Appl. Math. (1996).
[2] H. Bohl, U. Reif, Degenerate Bézier patches with continuous curvature, Comput. Aided Geom. Design 14 (1997) 749–761.
[3] O.P. Bruno, Fast, High-Order, High-Frequency Integral Methods for Computational Acoustics and Electromagnetics, in: M. Ainsworth, P. Davies, D. Duncan, P. Martin, B. Rynne, (Eds.), Topics in Computational Wave Propagation Direct and Inverse Problems Series, Lecture Notes in Computational Science and Engineering, vol. 31 (2003) ISBN: 3-540-00744-X. Also available at http://www.acm.caltech.edu/~bruno/bruno_springer_vol.pdf.
[4] O. Bruno, L. Kunyansky, A fast, high-order algorithm for the solution of surface scattering problems: Basic implementation, tests and applications, J. Comp. Phys. 169 (2001) 80–110.
[5] P. Borodin, M. Novotni, R. Klein, Progressive Gap Closing for Mesh Repairing (2002).
[6] O. Bruno, C. Geuzaine, J. Monro, F. Reitich, Prescribed error tolerances within fixed computational times for scattering problems of arbitrarily high frequency: the convex case, Phil. Trans. Roy. Soc. London A 362 (2004) 629–645.
[7] J. Boyd, Chebyshev and Fourier Spectral Methods, Dover Publications, New York, 2001.
[8] J. Boyd, A comparison of numerical algorithms for fourier extension of the first, second, and third kinds, J. Comp. Phys. 178 (2002) 118–160.
[9] L. Canino, J. Ottusch, M. Stalzer, J. Visher, M. Wandzura, Numerical solution of the Helmholtz equation in 2D and 3D using a high-order nystrom discretization, J. Comp. Phys. 146 (1998) 627–663.
[10] G. Chesshire, W.D. Henshaw, Composite overlapping meshes for the solution of partial differential equations, J. Comp. Phys. 90 (1990) 1–64.
[12] M. Elghaoui, R. Pasquetti, A spectral embedding method applied to the advection-diffusion equation, J. Comp. Phys. 125 (1996) 464–476.
[13] C.M. Grimm, J.F. Hughes, Modeling surfaces of arbitrary topology using manifolds, in: Proceedings of SIGGRAPH 95, Computer Graphics Proceedings, Annual Conference Series, 359368, 1995.
[14] D. Gottlieb, M. Hussaini, S. Orszag, Theory and application of spectral methods, in: Spectral Methods for Partial Differential Equations, SIAM, Philadelphia, 1984, pp. 1–54.
[15] W.D. Henshaw, Automatic grid generation, Acta Numer. (1996) 121–148.
[16] C.T. Loop, T.D. DeRose, A Multisided Generalization of Bézier Surfaces, ACM Trans. Graph. 8 (1989) 204–234.
[17] T. Varady, R.R. Martin, Reverse engineering, in: G. Farin, J. Hoschek, M.S. Kim (Eds.), Handbook of Computer Aided Geometric Design, Springer, 2002, pp. 651–681.
[18] P. Benko, G. Kos, T. Varady, L. Andor, R.R. Martin, Constrained fitting in reverse engineering, Comput. Aided Geomet. Des. 19 (2002) 173–205.
[19] J. Geer, Rational trigonometric approximations using Fourier series partial sums, J. Sci. Comp. 10 (1995) 325–356.
[20] J. Geer, N. Banerjee, Exponentially accurate approximations to piece-wise smooth periodic functions, J. Sci. Comp. 12 (1997) 253–287.
[21] D. Gottlieb, C. Shu, Resolution properties of the Fourier method for discontinuous waves, Comput. Meth Appl. Mech. Engrg. 116 (1994) 27–37.
[22] D. Gottlieb, C. Shu, A. Solomonoff, H. Vandeven, On the Gibbs phenomenon I: recovering exponential accuracy from the Fourier partial sum of a nonperiodic analytic function, J. Comput. Appl. Math. 43 (1992) 81–98.
[23] D. Gottlieb, E. Tadmor, Recovering pointwise values of discontinuous data within spectral accuracy, in: E.M. Murman, S.S. Abarbanel (Eds.), Progress and Supercomputing in Computational Fluid Dynamics, Birkhauser, Boston, 1985, pp. 357–375.
[24] D. Gottlieb, C. Shu, On the Gibbs phenonmenon and its resolution, SIAM Review 39 (1997) 644–668.
[25] A. Gelb, J. Tanner, Robust reprojection methods for the resolution of the Gibbs phenomenon, Appl. Comput. Harmon. Anal. 20 (2006) 3–25.
[26] J. Melenk, I. Babuska, The partition-of-unity finite element method: Basic theory and applications, Comput. Methods Appl. Mech. Engrg. 139 (1996) 289–314.
[27] A. Majda, J. McDonough, S. Osher, The Fourier method for nonsmooth initial data, Math. Comp. 32 (144) (1978) 1041–1081.
[28] A.A. Mezentsev, T. Woehler, Methods and Algorithms of Automated CAD Repair for Incremental Surface Meshing (1999).
[29] M. Mock, P. Lax, The computation of discontinuous solutions of linear hyperbolic equations, Commun. Pure Appl. Math. 31 (1978) 423–430.
[30] J.C. Navau, N.P. Garcia, Modeling surfaces from meshes of arbitrary topology, Comput. Aided Geom. Des. 17 (2000) 643671.

[31] P.S. Patel, D.L. Marcum, M.G. Remotigue, Automatic CAD Model Topology Generation, Int. J. Numer. Methods Fluids 52 (2006) 823–841.
[32] J. Peters, Curvature continuous spline surfaces over irregular meshes, Comput. Aided Geom. Des. 13 (1996) 101–131.
[33] J. Peters, C^2 free-form surfaces of degree (3,5), Comput. Aided Geom. Des. 19 (2002) 113–126.
[34] W. Press, S. Teukolsky, W. Vetterling, B. Flannery, Numerical recipes in Fortran, second ed., Cambridge University Press, 1992.
[35] U. Reif, TURBSTopologically Unrestricted Rational B-Splines, Constr. Approx. 14 (1998) 57–77.
[36] J. Strain, Locally-corrected multidimensional quadrature rules for singular functions, SIAM J. Sci. Comput. 16 (1995) 992–1017.
[37] H.P. Seidel, Polar Forms and Triangular B-Spline Surfaces, in: D.-Z. Du, F. Hwang (Eds.), Euclidean Geometry and Computers, second ed., World Scientific Publishing Company, 1994, pp. 235–286.
[39] J. Tanner, Optimal filter and mollifier for piecewise smooth spectral data, Math. Comp. 75 (2006) 767–790.
[40] G. Xu, Discrete Laplace–Beltrami operator on sphere and optimal spherical triangulations, Int. J. Comp. Geomet. Appl. 16 (2006) 75–93.
[41] L. Ying, D. Zorin, A simple manifold-based construction of surfaces of arbitrary smoothness, ACM Trans. Graph. 23 (2004) 271–275.