Journal of Computational Physics 229 (2010) 3358–3381

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

High-order unconditionally stable FC-AD solvers for general smooth domains II. Elliptic, parabolic and hyperbolic PDEs; theoretical considerations

Mark Lyon [a], Oscar P. Bruno [b,*]

[a] University of New Hampshire, Department of Mathematics, Kingsbury Hall W348, NH 03824, United States
[b] California Institute of Technology, Applied and Computational Mathematics, MC 217-50, 1200 East California Blvd., CA 91125, United States

ARTICLE INFO

Article history:
Received 17 April 2009
Received in revised form 1 January 2010
Accepted 7 January 2010
Available online 20 January 2010

Keywords:
Spectral method
Complex geometry
Unconditional stability
Fourier series
Fourier Continuation
ADI
Partial differential equation
Numerical method

ABSTRACT

A new PDE solver was introduced recently, in Part I of this two-paper sequence, on the basis of two main concepts: the well-known Alternating Direction Implicit (ADI) approach, on one hand, and a certain "Fourier Continuation" (FC) method for the resolution of the Gibbs phenomenon, on the other. Unlike previous alternating direction methods of order higher than one, which only deliver unconditional stability for rectangular domains, the new high-order FC-AD (Fourier-Continuation Alternating-Direction) algorithm yields *unconditional stability for general domains*—at an $\mathcal{O}(N\log(N))$ cost per time-step for an N point spatial discretization grid. In the present contribution we provide an overall theoretical discussion of the FC-AD approach and we extend the FC-AD methodology to linear hyperbolic PDEs. In particular, we study the convergence properties of the newly introduced FC(Gram) Fourier Continuation method for both approximation of general functions and solution of the alternating-direction ODEs. We also present (for parabolic PDEs on general domains, and, thus, for our associated elliptic solvers) a stability criterion which, when satisfied, ensures unconditional stability of the FC-AD algorithm. Use of this criterion in conjunction with numerical evaluation of a series of singular values (of the alternating-direction discrete one-dimensional operators) suggests clearly that the fifth-order accurate class of parabolic and elliptic FC-AD solvers we propose is indeed unconditionally stable for all smooth spatial domains and for arbitrarily fine discretizations. To illustrate the FC-AD methodology in the hyperbolic PDE context, finally, we present an example concerning the Wave Equation—demonstrating sixth-order spatial and fourth-order temporal accuracy, as well as a *complete* absence of the debilitating "dispersion error", also known as "pollution error", that arises as finite-difference and finite-element solvers are applied to solution of wave propagation problems.

1. Introduction

The FC-AD (Fourier-Continuation Alternating-Direction) methodology introduced in [1] (Part I of this two-paper sequence) relies on two main elements: a novel spectral technique for general spatial domains (which is based on the one-dimensional Fourier Continuation method introduced in Part I) and the classical ADI approach pioneered by Douglas, Peaceman and Rachford [2–6]. Unlike previous alternating direction methods of order higher than one, which only deliver uncon-

DOI of original article: 10.1016/j.jcp.2009.11.020
* 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 © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.01.006

ditional stability for rectangular domains, the new high-order FC-AD algorithm yields *unconditional stability for general domains*—at an $\mathcal{O}(N\log(N))$ cost per time-step for an N point spatial discretization grid. In the present contribution we provide an overall theoretical discussion of the FC-AD approach concerning unconditional stability and accuracy in the (linear) parabolic and elliptic contexts, and we extend the FC-AD methodology to problems concerning wave propagation and scattering. In conjunction with numerical evaluation of a series of singular values (of the alternating-direction discrete one-dimensional operators), our theory suggests clearly that the fifth-order accurate class of parabolic and elliptic FC-AD solvers we propose is indeed unconditionally stable for all smooth spatial domains and for arbitrarily fine discretizations. To illustrate the FC-AD methodology in the hyperbolic PDE context, finally, we present an example concerning the Wave Equation—demonstrating sixth-order spatial and fourth-order temporal accuracy, as well as *complete* absence of the debilitating "dispersion error", also known as "pollution error", that arises as finite-difference and finite-element solvers are applied to solution of wave propagation problems.

(A number of attempts have been made to combine the unconditional stability of the alternating direction type schemes with the spectral character of Fourier bases [7–10]. We expect that, like our FC-AD method, these Fourier-based approaches do not suffer from pollution errors. These previous efforts did not provide stable Fourier-based alternating-direction solvers for non-rectangular geometries; a more detailed discussion in these regards as well as comments concerning related spectral and spectral-element methodologies are given in the introduction to Part I.)

The appeal of the implicit alternating direction algorithms lies in the efficiency that results from their achievement of unconditional stability at a reduced cost per time-step. An important limitation has hindered the usefulness of the ADI, however: previous alternating direction methods could not be directly applied to PDEs on arbitrary (non-rectangular) domains without reducing the truncation error near the boundary to first order [11]. We note that while the ADI has been applied to problems on non-rectangular geometries [12–14], these applications were based on mappings of the PDE domains to rectangular regions—a procedure that is generally prohibitively laborious. To our knowledge, the FC-AD approach provides the first high-order accurate unconditionally stable alternating-direction scheme for general domains that does not rely on domain mappings.

A general discussion of current research on finite-difference and finite-element methods in the parabolic case for both simple and complex geometries was provided in Part I; here it is useful to summarize some of the main conclusions we have drawn as we placed the parabolic FC-AD algorithms in the context of the underlying literature. For diffusion equations the most notable advantage provided by the FC-AD approach lies in its unconditional stability for general domains: in Part I we demonstrated, for example, an improvement of a factor of 1000 in computing times, for engineering accuracies, over the computing time required by state of the art methodologies. Another interesting comparison concerns the contribution [15], which proposes a SAT method of order four of spatial and temporal accuracy for the diffusion equation: to our knowledge, this work introduces the SAT parabolic solver of highest demonstrated order of spatial accuracy. (Unlike the CFL condition for regular finite-difference methods, the SAT CFL restrictions are not affected as severely by small distances between the boundary and the nearest discretization points in the computational domain.) In view of their explicit character, however, existing SAT methods for parabolic equations do require time-steps proportional to the square of the spatial mesh-size, thus giving rise to high computing costs. In a direct comparison with the numerical example put forth in [15], for instance, our parabolic FC-AD solver produced the solution with accuracies matching the values 3×10^{-4}, 5×10^{-5} and 1×10^{-5} shown in Fig. 13 of that reference, in computational times that we estimate to be of the order of 80–100 times faster than those required by the method introduced in that reference. Such improvement factors result mainly for the fact that our unconditionally stable solver can produce the prescribed accuracies with a number of approximately 100 times fewer time-steps than the, e.g. 50,000 time-steps used by the SAT method in conjunction with its coarsest spatial discretization. These improvement-factor estimates take into account the slightly super-linear FFT cost and the cost arising from the fourth-order Richardson extrapolation inherent in our solver, as well as the cost arising from the fourth-order Runge–Kutta and nine-point finite differences stencil used in the method [15].

As mentioned above, besides an analysis of the parabolic and elliptic FC-AD solvers introduced previously, in this paper we put forward new FC-AD algorithms for the Wave Equation in two and three spatial dimensions. As is well known, spectral approaches provide major advantages over other methodologies for the solution of wave propagation problems. Indeed, owing to the accumulation of phase errors over multiple wave-cycles in long wave-trains, finite-difference and finite-element methods typically give rise to significant "dispersion errors", also known as "pollution errors", and thus require use of very large numbers of points per wavelength (PPW) in large-scale problems [16]. This difficulty was discussed in detail in [17,18] in the contexts of finite-difference and finite-element methods (FEM), respectively. It has long been recognized, further, that spectral methods generally do not suffer from this difficulty. As might be expected in view of the spectral nature of the FC-AD algorithms, the same is true of our Wave Equation FC-AD approach. Thus, the new FC-AD Wave Equation solver combines the low PPW-requirements typical of spectral solvers together with the geometric flexibility, high-order accuracy and unconditional stability otherwise inherent in the parabolic and elliptic FC-AD solvers.

To demonstrate the significant advantages offered by the (essentially dispersionless) FC method in the hyperbolic context we compare its performance with that resulting from finite-difference solvers of second- and fourth-orders of accuracy. In order to avoid difficulties associated with enforcement of boundary conditions in the finite-difference context, the finite-difference tests we perform involve periodic geometries only; our FC simulations, in turn, involve non-periodic, complex-geometry cases. The relevance of such comparisons becomes apparent when one considers that second- and fourth-order is indeed the state of the art accuracy-order for finite-difference solvers in complex domains: general-domain solvers recently made

available for solution of the Wave Equation include the second-order SAT method [19] and the compact scheme with enforcement of boundary conditions along normals [20]. These contributions demonstrate second- and fourth-order accuracies, respectively, by considering test PDE problems in annular domains and solutions with sinusoidal time dependence: a geometry of 1.5 wavelengths in diameter in [19] and (in addition to convergence studies for travelling waves) a one-wavelength propagation problem (one-cycle of a 10-wavelength scattering problem) in [20]. The accuracies demonstrated (for complex geometries) in these contributions is in full agreement with the accuracy produced by the second- and fourth-order periodic finite-difference tests we have performed—indicating a high-quality enforcement of boundary conditions in [19,20]. In particular we are confident that the periodic finite-difference tests we have conducted provide a reliable reference for comparison of the performance of the FC methods with state of the art finite-difference methods for complex domains. As shown in Fig. 13, for example, the 15 point-per-wavelength FC method produces solutions with 1% L^∞-error for problems involving 200 wavelengths—and, even 300 wavelengths and beyond; our fourth-order periodic finite-difference comparison code, in turn, requires 40 points per wavelength to solve the 200-wavelength problem with the same 1% error. (Results of a 13 point-per-wavelength test on a computational domain 200 wavelengths in diameter were presented in [20], but no estimates of the error were provided for that solution; based on our own periodic-domain finite-difference tests we expect the associated L^∞-errors to be of the order of 40% or higher—since our fourth-order finite-difference periodic-geometry test displayed in Fig. 13, shows 40% L^∞-errors at 15 points-per-wavelength.) We thus estimate that, for a three-dimensional problem of 200-wavelengths in diameter at 1% L^∞-error, use of the FC method gives rise to a reduction by a *factor of nineteen* $(= (40/15)^3)$ in the number of discretization points over those required by competitive fourth-order finite-difference techniques—with corresponding savings in computing time and memory. The discretization-size improvement factors are much more significant when comparisons are made with the second-order finite-difference methods used most often in practice: at 1% L^∞-error, and, again, for a 200 wavelength problem in three dimensions, the number of discretization points required by the FC method is 50,000 times smaller than the corresponding number of points required by a second-order finite-difference approach; see Section 7 for details.

Other relevant recent references concerning hyperbolic solvers include [21–23]. The Discontinuous Galerkin methods [23], on one hand, are flexible and robust, but they can be expensive in execution time when compared to lower-order methods; in any case it seems reasonable to expect the gains resulting from the FC method when compared to finite-element approaches should be even more significant than those discussed in the comparison with finite-difference methods presented above. An innovative method for the wave propagation problem, based on a Green's function time-evolution approach, on the other hand, was put forth in [24]. While this *explicit* method was shown to be capable of evolving solutions without restrictions imposed by the dimensions of the smallest spatial cells, a number of examples [21] show that the method suffers from instabilities. An exploratory discussion of a related method is presented in [22], showing that instabilities indeed occur, and that these can be controlled, at least in some cases, by changing the discretization densities around boundaries; the paper concludes, however, by stating that "the general conditions for this [stable, high-order] behavior remain unknown. This problem is formidable...".

Our FC-AD methodology overlays the geometry with a Cartesian mesh $\mathcal{D}_\Omega$, as shown in Fig. 1, and it uses, in addition, all boundary points that lie at the intersection of the domain boundary with Cartesian discretization lines L: an example of a line parallel to the x axis together with corresponding boundary points x_ℓ and x_r is depicted in Fig. 2. Note that on each Cartesian discretization line L a certain line-dependent number $n = n(L)$ of equi-spaced interior points are positioned in an arbitrary manner with respect to x_ℓ and x_r, with the only limitation that $x_1 - x_\ell$ and $x_r - x_n$ are both less than or equal to h, where $h = x_{j+1} - x_j,\ j = 1, \ldots, n$ is the interior-point spacing. After use of alternating direction operator splittings to reduce the PDE to a series of ODEs (of the form (42), see e.g. Eq. (82) below and Part I), our FC-AD approach uses a mesh such as the one depicted in Fig. 2 together with the high-order Fourier Continuation algorithm detailed in Part I (the FC(Gram) method)

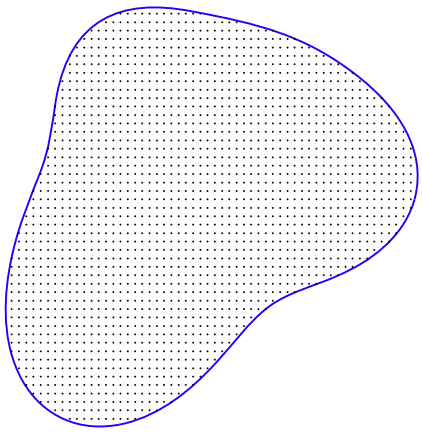


Fig. 1. Cartesian discretization $\mathcal{D}_\Omega = G \cap \Omega$ of a non-rectangular open domain Ω.

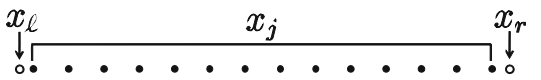


Fig. 2. One-dimensional discretization grid $(L \cap G \cap \Omega) \cup (L \cap \partial\Omega)$ on a typical discretization line L parallel to the x axis; similar discretizations are used on all discretization lines L parallel to each one of the Cartesian coordinate axes. Note the boundary points $L \cap \partial\Omega = \{x_\ell, x_r\}$ that generically do not lie on the regular one-dimensional Cartesian grid. A total of n discretization points $x_j,\ j = 1,\ldots,n$ are shown in addition to the boundary points x_ℓ and x_r.

to solve these ODEs with high-order accuracy at FFT speed—resulting, in all, in a high-order accurate solution to the PDE with unconditional stability and, for wave propagation problems, without the difficulties posed by dispersion/pollution errors.

This paper is organized as follows: after a brief description of the FC(Gram) approximation method, in Section 2 we analyze the errors that result from this algorithm. An estimate of the errors arising from the corresponding FC-ODE algorithm, in turn, is given in Section 3. The unconditional stability of the method for the Heat and Poisson Equations in a D-dimensional domain, $D \geqslant 2$, is studied in Section 4 via a reduction to a one-dimensional stability problem which can be solved through evaluation of certain singular value decompositions. As discussed in Section 4, unlike finite-difference eigenvalue-based stability tests, our stability criterion relies in an essential manner on use of singular values; see Remark 4.1. (Numerical results demonstrating unconditional stability and high-order accuracy of the FC-AD algorithm for the Heat and Poisson Equations, in agreement with the theoretical results of Sections 2–4, were presented in Part I.) In Section 5, we then extend the FC-AD methodology to the Wave Equation and, in Section 6, we present numerical results obtained from this approach—demonstrating, once again, unconditional stability and high-order accuracy. Results of an FC-AD algorithm which, in addition, produces a high-order of *temporal* accuracy, are also presented in this section. In Section 7, we present a comparative study on the numbers of PPW required by various methods showing, in particular, that the FC-AD algorithms do not suffer from dispersion/pollution effects. Our conclusions, finally, are put forward in Section 8.

2. Accuracy of the FC(Gram) approximation

2.1. Summary of the FC(Gram) continuation algorithm

The FC(Gram) method, the details of which are presented in Part I, provides an accelerated variant of the "continuation methods" introduced previously [25–27]; as discussed above, the FC(Gram) algorithm is a centerpiece of the FC-AD solver [1]. Briefly, an application of the FC(Gram) to a smooth function $y = f(x)$ defined on a bounded segment of the real line, proceeds by constructing a matching function f_{match} that joins curves obtained from portions of the graph of f near the boundaries of its domain of definition, see Fig. 3; using f_{match} the algorithm then produces a periodic function over a larger domain. Without loss of generality let the domain of the function f be the interval $[0,1]$, let Δ be the length of the boundary segments from which curves to be joined are obtained, and let the length the extended periodicity interval of the continuation function be $1 + d$. Further, let $x_j = (j-1)h,\ j = 1,\ldots,n$, $h = 1/(n-1)$, be a uniform discretization of the interval $[0,1]$ with $x_1 = 0$ and $x_n = 1$, and let

$$\Delta = I(n_\Delta - 1)h \quad \text{and} \quad d = I(n_d - 1)h \tag{1}$$

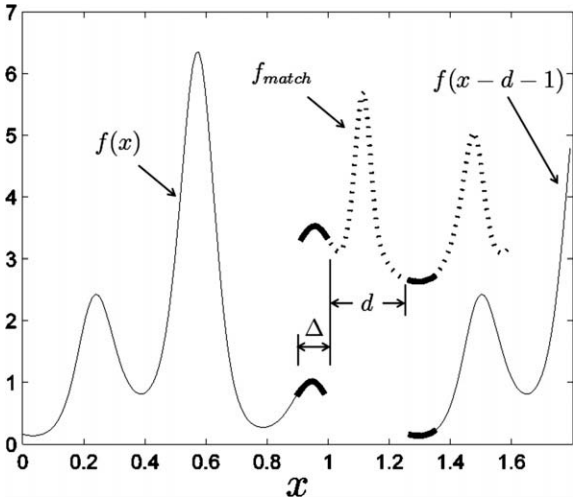


Fig. 3. Calculation of a periodic extension of $f(x) = e^{\sin(5.4\pi x - 2.7\pi) - \cos(2\pi x)}$ using only a small subset of function values $(n_\Delta = 10)$. Raised for visibility, the function $f_{\text{match}}(x)$ is displayed in the upper-right portion of the figure.

for certain integers I, n_Δ and n_d. The semi-positive-definite scalar products $(\cdot,\cdot)_{\text{left}}$ and $(\cdot,\cdot)_{\text{right}}$ – that are used in the prescription given below – are defined by

$$\begin{aligned}(h,k)_{\text{left}} &= \sum_{j\in S_{\text{left}}} h(x_j)k(x_j) \quad \text{and} \\ (h,k)_{\text{right}} &= \sum_{j\in S_{\text{right}}} h(x_j+1+d)k(x_j+1+d),\end{aligned} \tag{2}$$

where $S_{\text{right}} = \{1, I+1, 2I+1, \ldots, (n_\Delta-1)I+1\}$ and $S_{\text{left}} = \{n-(n_\Delta-1)I, n-(n_\Delta-2)I, n-(n_\Delta-3)I, \ldots, n\}$.

With reference to Fig. 3, the main elements of the FC(Gram) continuation method (which are presented in greater detail in [1]) are indicated in what follows:

1. For a given value of m (the resulting order of the approximation will be $m+1$), an orthonormal basis $\mathcal{B}_{\text{left}} = \{P^r_{\text{left}}(x)\}_{r=0}^m$ of the space of polynomials of degree $\leqslant m$ with respect to the scalar product $(\cdot,\cdot)_{\text{left}}$ is obtained by applying the $(\cdot,\cdot)_{\text{left}}$-based Gram–Schmidt orthogonalization process to the set $\{1, x, x^2, \ldots, x^m\}$ in order of increasing degree. Analogously, an orthonormal basis $\mathcal{B}_{\text{right}} = \left\{P^r_{\text{right}}(x)\right\}_{r=0}^m$ of the space of polynomials of degree $\leqslant m$ with respect to the scalar product $(\cdot,\cdot)_{\text{right}}$ is obtained by applying the $(\cdot,\cdot)_{\text{right}}$-based Gram–Schmidt orthogonalization process to the set $\{1, x, x^2, \ldots, x^m\}$ in order of increasing degree. (In practice, to avoid accuracy losses, our algorithms use the Gram–Schmidt orthogonalization process with partial reorthonormalization; see Part I.)

Remark 2.1. The "skipping-parameter" I determines the number $(I-1)$ of discretization points x_j in each one of the boundary intervals $[1-\Delta, 1]$ and $[1+d, 1+d+\Delta]$ that are "skipped" between any two discretization points used in the scalar product (2). As mentioned in Part I and as is established by the error estimate (37) below, slight increases of the parameter I are necessary to achieve convergence of the FC(Gram) algorithm to absolute zero errors (see also Remark 2.5). The necessary increases to achieve such convergence are actually extremely slow: in practice the value $I = 1$ (no skipping) is sufficient for high-order convergence to machine precision levels.

2. Given a function $f \in C^k[0,1]$, the coefficients

$$a^r_{\text{left}} = \left(f_{\text{left}}, P^r_{\text{left}}\right)_{\text{left}} \quad \text{and} \quad a^r_{\text{right}} = \left(f_{\text{right}}, P^r_{\text{right}}\right)_{\text{right}} \tag{3}$$

of the polynomial approximations (projections)

$$f^p_{\text{left}}(x) = \sum_{r=0}^m a^r_{\text{left}} P^r_{\text{left}}(x) \quad \text{and} \quad f^p_{\text{right}}(x) = \sum_{r=0}^m a^r_{\text{right}} P^r_{\text{right}}(x) \tag{4}$$

are obtained.

3. Highly accurate pre-computed FC(SVD) continuations $f^{P,Q} \in C^\infty_{per}[1-\Delta, 1+2d+\Delta]$ are used for certain pairs $\{P, Q\}$ of Gram Polynomials, where $P \in \mathcal{B}_{\text{left}}$ and $Q \in \mathcal{B}_{\text{right}}$, and where $f^{P,Q} \in C^\infty_{per}[1-\Delta, 1+2d+\Delta]$ is a Fourier Continuation of both P and Q. Various types of polynomial pairings are admissible as are methods to effect their joint continuation; full details concerning our prescriptions in these regards are presented in Section 2.3 of Part I. As indicated in that section, the method we use leads to certain continuation functions f^r_{even} and f^r_{odd}, $r = 0, \ldots, m$.
4. The function f_{match} is obtained as the following linear combination of the FC(SVD) continuations mentioned in point 3:

$$f_{\text{match}}(x) = \sum_{r=0}^m \frac{a^r_{\text{left}} + a^r_{\text{right}}}{2} f^r_{\text{even}}(x) + \frac{a^r_{\text{left}} - a^r_{\text{right}}}{2} f^r_{\text{odd}}(x). \tag{5}$$

5. A "discontinuous-projection" function f^{dp} is constructed according to the following formula (which is used in part on the basis of stability considerations presented in Section 4):

$$f^{\text{dp}}(x) = \begin{cases} f^p_{\text{right}}(x-1-d) & \text{for } x \in [0, \Delta], \\ f(x) & \text{for } x \in (\Delta, 1-\Delta), \\ f^p_{\text{left}}(x) & \text{for } x \in [1-\Delta, 1], \\ f_{\text{match}}(x) & \text{for } x \in (1, 1+d), \\ f^{\text{dp}}(x+1+d) = f^{\text{dp}}(x) & \text{for all } x \text{ in } \mathbb{R}. \end{cases} \tag{6}$$

6. The FC(Gram) continuation f^c of f is obtained, finally, as a trigonometric polynomial of periodicity interval $[0, 1+d]$ and appropriate degree, that interpolates f^{dp} at the equi-spaced mesh $x_j = (j-1)h$, $j = 1, \ldots, n + I(n_d-1) - 1$. (Notice that, although the function f^{dp} is discontinuous, see Remark 2.3, this function is close to a *smooth and periodic function* of period $1+d$.) In detail, f^c is defined as the Trigonometric Interpolant $(\mathcal{T})$

$$f^c(x) = \mathcal{T}(f^{\text{dp}})(x). \tag{7}$$

Here, the action of the operator $\mathcal{T}$ on a function $g \in L^2[0, 1+d]$ is defined by

$$\mathcal{T}(g)(x) = \frac{1}{1+d} \sum_{k \in t(n+I(n_d-1)-1)} c_k e^{-\Psi_k x}, \tag{8}$$

where $\Psi_k = \frac{2\pi i k}{1+d}$, $t(r) = \{k \in \mathbb{N} : -r/2+1 \leqslant k \leqslant r/2\}$ for r even and $t(r) = \{k \in \mathbb{N} : -(r-1)/2 \leqslant k \leqslant (r-1)/2\}$ for r odd, and where the coefficients c_k, which are given by the Discrete Fourier Transform

$$c_k = \frac{1+d}{n+I(n_d-1)-1} \sum_{j=1}^{n+I(n_d-1)-1} g(x_j) e^{\Psi_k x_j}, \tag{9}$$

can clearly be obtained rapidly by means of a Fast Fourier Transform (FFT). Although FFTs of size given by powers of two can generally not be used in this context, as discussed in the introduction to Part I, use of "adequate" FFT implementations enable evaluation of the discrete Fourier Transform (9) at a computational cost of $\mathcal{O}\{(n+I(n_d-1)-1)\log(n+I(n_d-1)-1)\}$ operations.

Remark 2.2. It is easy to check that, while for the error analysis below it is useful to define f_{match} and f^{dp} as functions of a continuous variable x (which requires f itself to be a function of the continuous variable x), the definition of the function $f^c(x)$ actually depends only on the values $(f_1, \ldots, f_n) = (f(x_1), \ldots, f(x_n))$. Thus, via the substitutions

$$f(x_j) \to f_j \quad \text{for } j = 1, \ldots, n, \tag{10}$$

the prescriptions above can be used to define a Fourier Continuation function $f^c(x)$ from discrete data of the form $f = (f_1, \ldots, f_n)$.

2.2. Error analysis in the interval [0,1]

The brief description above of the FC(Gram) continuation algorithm suffices for the purposes of this paper; a more detailed presentation in these regards is given in Part I. In what follows we provide an estimate of the error in FC(Gram) approximations of a smooth function $f \in C^k[0,1]$, where k is either a sufficiently large positive integer or $k = \infty$. In Section 2.3, we then extend this result to provide estimates of the extrapolation errors that occur as the FC(Gram) continuation method in the interval $[0,1]$ is used for a smooth function f that is defined in a "slightly extended interval" $[-\varepsilon_{\text{left}}, 1+\varepsilon_{\text{right}}] \supseteq [0,1]$. (The extended error estimate is needed for our analysis of the FC(ODE) error in Section 3.) To obtain our FC-Gram error estimates we evaluate the errors arising from each of the error-generating elements of the method, namely steps 2, 3–4 and 5–6 above.

Remark 2.3. In our analysis of the FC-AD algorithms a slight variation of the function f^{dp}, namely, the discontinuous extension function

$$f^{\text{de}}(x) = \begin{cases} f(x) & \text{for } x \in [0,1], \\ f_{\text{match}}(x) & \text{for } x \in (1, 1+d], \\ f^{\text{de}}(x+1+d) = f^{\text{de}}(x) & \text{for all } x \text{ in } \mathbb{R}, \end{cases} \tag{11}$$

is used in addition to f^{dp} itself. Clearly, the functions f^{de} and f^{dp} differ by amounts that tend to zero like h^{m+1} as $h \to 0$. At some points in our analysis it will prove necessary to perform certain blending operations on discontinuous functions which, like f^{de} and f^{dp}, differ by little from a smooth function. The smooth functions resulting from such blending procedures will be distinguished by superimposing a bar to the function name so that, e.g. the smooth, blended version of f^{de} will be denoted by $\overline{f^{\text{de}}}$. For example, the blended version of the function f^{de} is defined by

$$\overline{f^{\text{de}}}(x) = \begin{cases} f(x) + w(x/\Delta)(f_{\text{match}}(x+1+d) - f(x)) & \text{for } x \in [0, \Delta], \\ f(x) & \text{for } x \in (\Delta, 1-\Delta), \\ f(x) + w((1-x)/\Delta)(f_{\text{match}}(x) - f(x)) & \text{for } x \in [1-\Delta, 1+d], \\ \overline{f^{\text{de}}}(x+1+d) = \overline{f^{\text{de}}}(x) & \text{for all } x \in \mathbb{R}, \end{cases} \tag{12}$$

where $w(x)$ is a smooth infinitely differentiable windowing function satisfying $0 \leqslant w(x) \leqslant 1$, and such that $w(x) = 1$ for $x \leqslant 0$ and $w(x) = 0$ for $x \geqslant 1$. Fig. 4 demonstrates the blending procedure around $x = 1$ assuming $\Delta = 0.1$: the functions f and f_{match}, which are defined in $[0,1]$ and $[1-\Delta, 1+d+\Delta]$, respectively, are blended smoothly in the region $[1-\Delta, 1] = [0.9, 1]$. Although our analysis uses smooth blending functions such as $\overline{f^{\text{de}}}$, our final error estimates apply to the function f^c—whose definition does not include the use of blending procedures.

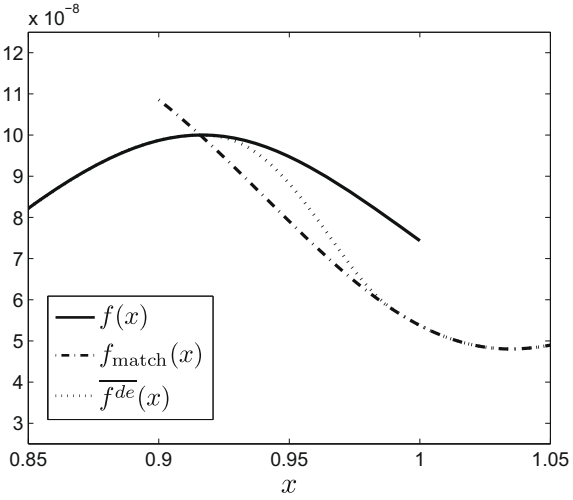


Fig. 4. Blended (smooth) version $\overline{f^{\mathrm{de}}}$ of f^{de} near $x = 1$, resulting from smooth blending of f and f_{match} according to Eq. (12).

2.2.1. Error arising from step 2 of the FC(Gram) algorithm

Step 2 of the FC(Gram) algorithm projects orthogonally the boundary functions f_{left} and f_{right} onto spaces of polynomials of degree $\leqslant m$ with respect to the semi-positive definite discrete scalar products (2). To proceed with our error analysis we consider optimal L^∞ polynomial approximations: we let p_{left} and p_{right} be the optimal maximum norm approximations of f_{left} and f_{right} by polynomials of degree m in the corresponding Δ-length boundary regions. A well known expression (cf. [28, p. 91]) gives the maximum-norm best polynomial approximation; in the present case we obtain, for example

$$\|f_{\mathrm{left}} - p_{\mathrm{left}}\|_{L^\infty[1-\Delta,1]} = \frac{\Delta^{m+1}|f^{(m+1)}(\xi)|}{2^{2m+1}(m+1)!} \tag{13}$$

for some point ξ in the Δ-length interval, with a similar identity for f_{right} and p_{right}. Since p_{left} is in the space of polynomials of degree $\leqslant m$, its orthogonal projection (according to (2)) into the subspace of polynomials is equal to itself. Thus, the maximum error made in projecting f_{left} into the subspace of polynomials of degree $\leqslant m$ over the interval $[1-\Delta, 1]$ is bounded as follows:

$$\left\|f_{\mathrm{left}} - f^p_{\mathrm{left}}\right\|_{L^\infty[1-\Delta,1]} \leqslant \|f_{\mathrm{left}} - p_{\mathrm{left}}\|_{L^\infty[1-\Delta,1]} + \left\|\sum_{r=0}^{m}\left(p_{\mathrm{left}} - f_{\mathrm{left}}, P^r_{\mathrm{left}}\right)_{\mathrm{left}} P^r_{\mathrm{left}}\right\|_{L^\infty[1-\Delta,1]}. \tag{14}$$

Introducing the quantity (related to the Lebesgue constant, cf. [28, p. 100])

$$L(m, n_\Delta) = \max_{x\in[1-\Delta,1]} \sum_{r=0}^{m}\left(1, |P^r_{\mathrm{left}}|\right)_{\mathrm{left}} |P^r_{\mathrm{left}}(x)| = \max_{x\in[1+d,1+d+\Delta]} \sum_{r=0}^{m}\left(1, \left|P^r_{\mathrm{right}}\right|\right)_{\mathrm{right}} \left|P^r_{\mathrm{right}}(x)\right| \tag{15}$$

(the definitions in terms of "left" and "right" polynomials coincide in view of the easily-established identity $P^r_{\mathrm{right}}(x) = P^r_{\mathrm{left}}(x - d - \Delta)$) it follows that

$$\left\|\sum_{r=0}^{m}\left(p_{\mathrm{left}} - f_{\mathrm{left}}, P^r_{\mathrm{left}}\right)_{\mathrm{left}} P^r_{\mathrm{left}}\right\|_{L^\infty[1-\Delta,1]} \leqslant L(m, n_\Delta)\|f_{\mathrm{left}} - p_{\mathrm{left}}\|_{L^\infty[1-\Delta,1]} \tag{16}$$

with a corresponding inequality with "left" substituted by "right" and $[1-\Delta, 1]$ substituted by $[1+d, 1+d+\Delta]$. Thus, denoting by M the maximum of $f^{(m+1)}(x)$ over the intervals $[1-\Delta, 1]$ and $[0, \Delta]$ (whose values coincide with those of the corresponding derivatives of f_{left} and f_{right} on the intervals $[1-\Delta, 1]$ and $[1+d, 1+d+\Delta]$, respectively), we obtain the bounds

$$\left\|f_{\mathrm{left}} - f^p_{\mathrm{left}}\right\|_{L^\infty[1-\Delta,1]} \leqslant (1 + L(m, n_\Delta))\frac{\Delta^{m+1}M}{2^{2m+1}(m+1)!} \quad \text{and} \tag{17}$$

$$\left\|f_{\mathrm{right}} - f^p_{\mathrm{right}}\right\|_{L^\infty[1+d,1+d+\Delta]} \leqslant (1 + L(m, n_\Delta))\frac{\Delta^{m+1}M}{2^{2m+1}(m+1)!}. \tag{18}$$

The parameter $L(m, n_\Delta)$ for values of the maximum polynomial degree m and the dimension n_Δ of the discrete scalar products (2) that are relevant to this work (which can be obtained by means of direct evaluations of the expression (15)) are displayed in Table 1.

Table 1
Lebesgue-type constants $L(m, n_\Delta)$ for $n_\Delta = 10$ and for various values of m of interest in the present section. The estimates of the parameter $L(m, n_\Delta)$ displayed in this table were obtained as the maximum of the first sum in Eq. (15) over 30,000 equi-spaced points in the interval $[1-\Delta, 1]$.

	$m=4$	$m=5$	$m=9$
$L(m, n_\Delta)$	1.7021	1.9261	17.849

2.2.2. Error arising from steps 3 and 4 of the FC(Gram) algorithm

An additional component of the error in the FC(Gram) continuation method arises as the FC(SVD) algorithm is used to produce the matching function f_{match}. Clearly this additional error can be evaluated as a linear combination of the errors in the approximation of the relevant even and odd Gram polynomial pairs by the corresponding FC(SVD) continuation functions f^r_{even} and f^r_{odd} for each $r \in \{0, \ldots, m\}$. Table 2 lists the latter errors for the even and odd polynomial pairs arising from the six Gram polynomials $r \in \{0, \ldots, 5\}$ that are used in the FC-AD approach, and for parameter values (see point 4 above) as detailed in Part I.

The errors inherent in the FC(SVD) continuations f^r_{even} and f^r_{odd} of the Gram basis functions (Table 2) can be combined according to Eq. (5) to produce an estimate of the error in the approximation of f_{left} and f_{right} by f_{match}: for $x \in [1-\Delta, 1]$, we obtain

$$f_{\text{match}}(x) - f^p_{\text{left}(x)} = \sum_{r=0}^{m} \frac{a^r_{\text{left}} + a^r_{\text{right}}}{2} \left(f^r_{\text{even}}(x) - P^r_{\text{left}}(x)\right) + \frac{a^r_{\text{left}} - a^r_{\text{right}}}{2} \left(f^r_{\text{odd}}(x) - P^r_{\text{left}}(x)\right) \tag{19}$$

with a similar result on the right for $x \in [1+d, 1+d+\Delta]$. For future use we introduce the notation

$$S_0(m) = \max\left(\left\| f_{\text{match}} - f^p_{\text{left}} \right\|_{L^\infty[1-\Delta,1]}, \left\| f_{\text{match}} - f^p_{\text{right}} \right\|_{L^\infty[1+d,1+d+\Delta]} \right), \tag{20}$$

which is used in our subsequent error analysis; note that $S_0(m)$ also depends on d/Δ, n_Δ, g and Υ: $S_0(m) = S_0(m, d/\Delta, n_\Delta, g, \Upsilon)$. In view of (19) and the corresponding equation involving $f^p_{\text{right}(x)}$, $S_0(m)$ can be bounded, for our choice of parameters, in terms of the approximation errors listed in Table 2. Clearly, owing to the observed spectral accuracy of the FC(SVD) continuation method [27], given a fixed n_Δ, $S_0(m)$ decays spectrally as g and Υ grow appropriately, see point 3 in Section 2.1, independently of I.

Remark 2.4. With reference to Table 2, we note that the errors arising in steps 3 and 4 are negligible in practice. Indeed the coefficients multiplying f^r_{even} and f^r_{odd} decay like $\mathcal{O}(h^r)$ and thus full machine precision approximations can be obtained from steps 3 and 4 despite entries in Table 2 which are larger than machine precision.

2.2.3. Error arising from the overall FC(Gram) algorithm: steps 1–6

Since $f^c = \mathcal{T}(f^{\text{dp}})$, to estimate $\|f - f^c\|_{L^\infty[0,1]} = \|f - \mathcal{T}(f^{\text{dp}})\|_{L^\infty[0,1]}$, we first construct a smooth approximation $\overline{f^{\text{de}}}$ (see Remark 2.3) of the discontinuous function f^{de} (using the smooth windowing function $w(x)$ scaled to a Δ-length interval, see e.g. Eq. (12) above). It is easy to obtain estimates, that are needed in what follows, of the error introduced by the smoothing process: from the fact that $0 \leqslant w(x) \leqslant 1$ and from the bounds (17), (18) and (20), we obtain the result

$$\|f - \overline{f^{\text{de}}}\|_{L^\infty[0,1]} \leqslant S_0(m) + (1 + L(m, n_\Delta)) \frac{\Delta^{m+1} M}{2^{2m+1}(m+1)!}. \tag{21}$$

Considering the definition (6) of f^{dp}, in turn, we obtain

$$\|f^{\text{dp}} - \overline{f^{\text{de}}}\|_{L^\infty[0,1]} \leqslant S_0(m) + (1 + L(m, n_\Delta)) \frac{\Delta^{m+1} M}{2^{2m+1}(m+1)!}. \tag{22}$$

Table 2
Maximum values over $[1-\Delta, 1] \cup [1+d, 1+d+\Delta]$ of the differences between the even and odd Gram polynomial pairs and their SVD continuations f^r_{even} and f^r_{odd} for the parameter values $n_\Delta = 10$, $d/\Delta = 26/9$, $g = 63$ and $\Upsilon = 150$. (Maximum values evaluated as maxima over 1800 points in the set $[1-\Delta, 1] \cup [1+d, 1+d+\Delta]$.)

r	$\max\left\lvert \left\{P^r_{\text{left}}, P^r_{\text{right}}\right\} - f^r_{\text{even}} \right\rvert$	$\max\left\lvert \left\{P^r_{\text{left}}, -P^r_{\text{right}}\right\} - f^r_{\text{odd}} \right\rvert$
0	5.6×10^{-17}	1.7×10^{-16}
1	6.1×10^{-16}	5.6×10^{-16}
2	2.9×10^{-15}	2.5×10^{-15}
3	1.4×10^{-14}	1.6×10^{-14}
4	1.4×10^{-13}	5.6×10^{-14}
5	4.2×10^{-13}	5.3×10^{-13}

The desired estimate of the error $\|f - \mathcal{T}(f^{\mathrm{dp}})\|_{L^\infty[0,1]}$ is based on use of the smooth and $(1+d)$-periodic function $\overline{f^{\mathrm{de}}}$: adding and subtracting both $\overline{f^{\mathrm{de}}}$ and its interpolant $\mathcal{T}(\overline{f^{\mathrm{de}}})$ we see, in view of (21), that it suffices to obtain adequate estimates for $\|\overline{f^{\mathrm{de}}} - \mathcal{T}(\overline{f^{\mathrm{de}}})\|_{L^\infty[0,1]}$ and $\|\mathcal{T}(\overline{f^{\mathrm{de}}}) - \mathcal{T}(f^{\mathrm{dp}})\|_{L^\infty[0,1]}$; we obtain such estimates in what follows.

In order to obtain a bound for $\|\overline{f^{\mathrm{de}}} - \mathcal{T}(\overline{f^{\mathrm{de}}})\|_{L^\infty[0,1]}$ (whose final form is given in Eq. (35) below) we consider the exact Fourier series of $\overline{f^{\mathrm{de}}}$ and, recalling the definition of the function t given below Eq. (8), its $t(n + I(n_d - 1) - 1)$-term truncation $\overline{f^{\mathrm{de}}_{\mathrm{trunc}}}$

$$\overline{f^{\mathrm{de}}}(x) = \frac{1}{1+d}\sum_{k=-\infty}^{\infty} c_k^{\mathrm{ex}} e^{-\Psi_k x}, \quad \overline{f^{\mathrm{de}}_{\mathrm{trunc}}}(x) = \frac{1}{1+d}\sum_{k\in t(n+I(n_d-1)-1)} c_k^{\mathrm{ex}} e^{-\Psi_k x}, \tag{23}$$

as well as its trigonometric interpolation

$$\mathcal{T}(\overline{f^{\mathrm{de}}})(x) = \frac{1}{1+d}\sum_{k\in t(n+I(n_d-1)-1)} c_k e^{-\Psi_k x}, \tag{24}$$

where again $\Psi_k = \frac{2\pi i k}{1+d}$. Clearly, the interpolation coefficients c_k (Eq. (9)) do not coincide with the exact Fourier coefficients of the function $\overline{f^{\mathrm{de}}}$, which are given by

$$c_k^{\mathrm{ex}} = \int_0^{1+d} \overline{f^{\mathrm{de}}}(x) e^{\Psi_k x}\, dx. \tag{25}$$

The error introduced in the approximation of the function $\overline{f^{\mathrm{de}}}$ by $\mathcal{T}(\overline{f^{\mathrm{de}}})$ can be estimated as follows:

$$|\overline{f^{\mathrm{de}}}(x) - \mathcal{T}(\overline{f^{\mathrm{de}}})(x)| \leqslant \left|\overline{f^{\mathrm{de}}}(x) - \mathcal{T}\left(\overline{f^{\mathrm{de}}_{\mathrm{trunc}}}\right)(x)\right| + \left|\mathcal{T}\left(\overline{f^{\mathrm{de}}} - \overline{f^{\mathrm{de}}_{\mathrm{trunc}}}\right)(x)\right|. \tag{26}$$

Now, from a classical result [29, p. 119, 30, p. 272] on trigonometric interpolation error, for a function $g \in L^\infty[0, 1+d]$ we have

$$|\mathcal{T}(g)(x)| \leqslant \widetilde{C_{\mathcal{T}}} \log(n + I(n_d - 1) - 1)\|g\|_{L^\infty[0,1+d]} \tag{27}$$

for some constant $\widetilde{C_{\mathcal{T}}}$. Letting $g = \overline{f^{\mathrm{de}}} - \overline{f^{\mathrm{de}}_{\mathrm{trunc}}}$ (the tail of the Fourier series) and recognizing that, since $\mathcal{T}\left(\overline{f^{\mathrm{de}}_{\mathrm{trunc}}}\right) = \overline{f^{\mathrm{de}}_{\mathrm{trunc}}}$, the first term on the right hand side of Eq. (26) equals $|g(x)|$, from Eqs. (26) and (27) we obtain

$$|\overline{f^{\mathrm{de}}}(x) - \mathcal{T}(\overline{f^{\mathrm{de}}})(x)| \leqslant C_{\mathcal{T}} \log(n + I(n_d - 1) - 1)\left\|\overline{f^{\mathrm{de}}} - \overline{f^{\mathrm{de}}_{\mathrm{trunc}}}\right\|_{L^\infty[0,1+d]}, \tag{28}$$

where $C_{\mathcal{T}} = \widetilde{C_{\mathcal{T}}} + 1$.

To express our estimate in terms of the parameters of the problem we note that, clearly

$$\left\|\overline{f^{\mathrm{de}}} - \overline{f^{\mathrm{de}}_{\mathrm{trunc}}}\right\|_{L^\infty[0,1+d]} \leqslant \sum_{k\notin t(n+I(n_d-1)-1)} \left|c_k^{\mathrm{ex}}\right|. \tag{29}$$

To bound the magnitude of the coefficients c_k^{ex}, we use integration by parts q times on Eq. (25); we obtain

$$c_k^{\mathrm{ex}} = \frac{(-1)^q}{\Psi_k^q}\int_0^{1+d} \overline{f^{\mathrm{de}}}^{(q)}(x) e^{\Psi_k x}\, dx, \tag{30}$$

where the boundary terms of the integration by parts procedure vanish since $\overline{f^{\mathrm{de}}}$ is a smooth periodic function. But, over the interval $[1, 1+d]$, the function $\overline{f^{\mathrm{de}}}$ coincides with f_{match}, which is given by the linear combination (5). It follows that, over that interval, the qth order derivative $\overline{f^{\mathrm{de}}}^{(q)}$ is given by the corresponding linear combination of $(f^r_{\mathrm{even}})^{(q)}$ and $(f^r_{\mathrm{odd}})^{(q)}$, $r = 0, \ldots, m$. Further, the polynomials in the bases $\mathcal{B}_{\mathrm{left}}$ and $\mathcal{B}_{\mathrm{right}}$ and, therefore, their FC(SVD) continuations f^r_{even} and f^r_{odd}, can be obtained as scaled versions of a set of C^∞ functions that are independent of d, Δ, I and h—provided that, as prescribed in point 4 of Section 2.1, n_Δ, d/Δ, g and Υ are kept fixed. It follows that, for all $q \geqslant 0$, $f^{(q)}_{\mathrm{match}}(x)$ is bounded by Δ^{-q} times a constant independent of h, d, I and Δ. Similarly, for any $q \geqslant 0$, the qth derivative $w^{(q)}$ of the windowing function used to construct $\overline{f^{\mathrm{de}}}$ is bounded, and, thus, the qth derivatives $d^q/dx^q(w(x/\Delta))$ and $d^q/dx^q(w((1-x)/\Delta))$ of the scaled functions over the respective intervals $[0, \Delta]$ and $[1-\Delta, 1]$ are bounded by Δ^{-q} times a constant independent of h, d, I and Δ. Thus, using the chain rule for differentiation we see that, for fixed $f \in C^q[0,1]$, there exist constants F_q and M_q, independent of h, d, I and Δ, such that

$$\left|\frac{d^q}{dx^q}\overline{f^{\mathrm{de}}}(x)\right| \leqslant \begin{cases} M_q\Delta^{-q} & \text{for } x \in [0, \Delta], \\ F_q & \text{for } x \in [\Delta, 1-\Delta], \\ M_q\Delta^{-q} & \text{for } x \in [1-\Delta, 1+d], \end{cases} \tag{31}$$

where M_q depends only mildly on the function being approximated. Eqs. (30) and (31) give rise to the following bound for the coefficients c_k^{ex}:

$$|c_k^{\mathrm{ex}}| \leqslant \frac{(1+d)^q}{(2\pi k)^q}F_q + \frac{(d+2\Delta)(1+d)^q}{(2\pi k\Delta)^q}M_q. \tag{32}$$

From the definition of the function t (below Eq. (8)) we note that for $k \notin t(n + I(n_d - 1) - 1)$ we have $|k| \geqslant (1+d)/2h$. Clearly for all $q > 1$ and $n \geqslant 1$ we also have

$$\sum_{k=n}^{\infty} \frac{1}{k^q} \leqslant \frac{1}{n^q} + \int_n^{\infty} \frac{1}{k^q}\, dk = \frac{1}{n^q} + \frac{1}{(q-1)(n)^{q-1}} \leqslant \frac{q}{(q-1)n^q}. \tag{33}$$

Therefore from Eqs. (29) and (32), we obtain the bound

$$\left\| \overline{f^{\mathrm{de}}} - \overline{f^{\mathrm{de}}_{\mathrm{trunc}}} \right\|_{L^\infty[0,1+d]} \leqslant \frac{q(1+d)h^{q-1}}{(q-1)\pi^q} F_q + \frac{q(d/\Delta+2)(1+d)h^{q-1}\Delta}{(q-1)(\pi\Delta)^q} M_q \tag{34}$$

and, in view of (28), one of the two needed estimates follows:

$$\begin{aligned} \|\mathcal{T}(\overline{f^{\mathrm{de}}}) - \overline{f^{\mathrm{de}}}\|_{L^\infty[0,1+d]} &\leqslant C_{\mathcal{T}} \log(n + I(n_d-1) - 1) \frac{q(1+d)h^{q-1}}{(q-1)\pi^q} F_q + C_{\mathcal{T}} \log(n + I(n_d-1) - 1) \\ &\quad \times \frac{q(d/\Delta+2)(1+d)h^{q-1}\Delta}{(q-1)(\pi\Delta)^q} M_q. \end{aligned} \tag{35}$$

Noting that $\overline{f^{\mathrm{de}}} = f^{\mathrm{dp}}$ on the interval $[1, 1+d]$ and using (22) and (27), finally, we obtain

$$\|\mathcal{T}(\overline{f^{\mathrm{de}}}) - \mathcal{T}(f^{\mathrm{dp}})\|_{L^\infty[0,1]} \leqslant C_{\mathcal{T}} \log(n + I(n_d-1) - 1) \left(S_0(m) + (1 + L(m, n_\Delta)) \frac{\Delta^{m+1} M}{2^{2m+1}(m+1)!} \right). \tag{36}$$

The full estimate of the error of the FC(Gram) approximation now follows by combining Eqs. (21), (35) and (36):

$$\|f - f^c\|_{L^\infty[0,1]} \leqslant \log(n + I(n_d-1) - 1)(\zeta_1 + \zeta_2 + \zeta_3 + \zeta_4) \tag{37}$$

with

$$\zeta_1 = (C_{\mathcal{T}} + 1)(1 + L(m, n_\Delta)) \frac{\Delta^{m+1} M}{2^{2m+1}(m+1)!}, \quad \zeta_2 = (C_{\mathcal{T}} + 1) S_0(m), \tag{38}$$

$$\zeta_3 = C_{\mathcal{T}} \frac{q(1+d)h^{q-1}}{(q-1)\pi^q} F_q, \quad \text{and} \quad \zeta_4 = C_{\mathcal{T}} \frac{q(d/\Delta+2)(1+d)h^{q-1}\Delta}{(q-1)(\pi\Delta)^q} M_q. \tag{39}$$

Remark 2.5. Since, according to the prescription in Eq. (1), Δ equals $I(n_\Delta - 1)h$ with integer values of I and n_Δ (n_Δ fixed), the quantities ζ_1 and ζ_3 converge to zero like h^{m+1} as $h \to 0$, provided q is large enough. The quantities ζ_2 and ζ_4, in contrast, do not tend to zero as $h \to 0$: in this limit ζ_2 and ζ_4 converge to small values determined by the errors of approximation of Gram polynomials by FC(SVD) continuations. In view of the observed spectral convergence of the FC(SVD) approximation, (some details of such a convergence proof have been verified numerically but a rigorous proof is still unavailable [27]), given a fixed n_Δ, the quantity ζ_2 decays spectrally as g and Υ grow appropriately, see point 3 in Section 2.1, independently of I. The skipping parameter I, in turn, can be increased holding g and Υ fixed (and increasing $n = 1/h + 1$ as necessary), to make ζ_4 as small as desired. Thus, taking fixed values of n_Δ and d/Δ, the bound (37) implies convergence of the FC(Gram) algorithm as h tends to zero and I, g and Υ are allowed to grow appropriately. Our numerical experiments [1] indicate that, taking $I = 1$, suitable values for the FC(SVD) continuations parameters g and Υ can be selected (see Table 2 and Remark 2.4) to ensure $\mathcal{O}(h^{m+1})$ convergence of the FC(Gram) approximation to full machine precision. In view of these considerations, only $I = 1$ is used in our numerical examples and theoretical considerations for the remainder of this paper.

2.3. Error analysis in the slightly extended interval $[-\varepsilon_{\mathrm{left}}, 1 + \varepsilon_{\mathrm{right}}] \supseteq [0, 1]$

As mentioned at the beginning of Section 2.2, here we generalize the analysis provided in that section to obtain bounds on the extrapolation errors that occur as the FC(Gram) continuation method in the interval [0,1] is used to approximate a smooth function f on a "slightly extended interval" $[-\varepsilon_{\mathrm{left}}, 1 + \varepsilon_{\mathrm{right}}] \supseteq [0, 1]$ with $\varepsilon_{\mathrm{left}} = x_1 - x_\ell < h$ and $\varepsilon_{\mathrm{right}} = x_r - x_n < h$ (see Section 1). These bounds are used in Section 3 to estimate the accuracy of the FC-ODE algorithm.

To obtain bounds valid in the interval $[-\varepsilon_{\mathrm{left}}, 1 + \varepsilon_{\mathrm{right}}] \supseteq [0, 1]$, only the bounds (21) and (22) (and consequently, only the parameters ζ_1 and ζ_2 in Eq. (37)) need to be generalized to the interval $[-\varepsilon_{\mathrm{left}}, 1 + \varepsilon_{\mathrm{right}}]$: the estimate (35) is already valid over the slightly extended interval. Letting

$$\varepsilon = \max(\varepsilon_{\mathrm{left}}, \varepsilon_{\mathrm{right}}) \quad \text{and} \quad \Delta^\varepsilon = \Delta + \varepsilon, \tag{40}$$

and generalizing the analysis of the previous section to the slightly larger interval yields a bound of the form (37) with ζ_1 and ζ_2 substituted by

$$\zeta_1^\varepsilon = (C_T+1)(1+L^\varepsilon(m,n_\Delta))\frac{(\Delta^\varepsilon)^{m+1}M}{2^{2m+1}(m+1)!} \quad \text{and} \quad \zeta_2^\varepsilon = (C_T+1)S_0^\varepsilon(m), \tag{41}$$

where $L^\varepsilon(m,n_\Delta)$ is given by the obvious modification of (15), i.e., the maximum is now evaluated over the interval $[1-\Delta, 1+\varepsilon]$. Fig. 5 demonstrates, for the function $f(x)=e^x$, the extrapolation errors that result as the Fourier Continuation function f^c is used to approximate the original function f in the interval $[1,1+h]$—displaying, in particular, errors in the intervals $[1,1+\varepsilon_{\text{right}}]$ for all $\varepsilon_{\text{right}} \leqslant h$.

Clearly, the value of $L^\varepsilon(m,n_\Delta)$ (which we have evaluated numerically as the maximum of a large number of points in the interval $[1-\Delta,1+\varepsilon]$) is an increasing function of ε. The effect of the extrapolation is significant: our numerical calculations show that $L^\varepsilon(m,n_\Delta)$ increases rapidly with ε, going from about 1.7 for $\varepsilon=0$ to a limit as $\varepsilon \to h$ that is no larger than 8.5 for $m=4$, and from just less than 2 for $\varepsilon=0$ to a maximum of about 16 as $\varepsilon \to h$ for $m=5$. We only consider here the dependence of $L^\varepsilon(m,n_\Delta)$ on ε for $m=4$ and $m=5$ since, owing to the stability considerations presented in Section 4, these are the values of m we use within our FC-AD methodology. These bounds on $L^\varepsilon(m,n_\Delta)$ (8.5 for $m=4$ and 16 for $m=5$) are independent of the mesh and therefore constants for the convergence analysis of the FC(Gram) approximation. The value of $S_0^\varepsilon(m)$, in turn, is determined in the same manner as $S_0(m)$ (see the text below Eq. (20)); upper bounds for $S_0^\varepsilon(m)$ for all $0 \leqslant \varepsilon \leqslant h$ can be obtained as the values in Table 3 are used instead of those in Table 2: Table 3 assumes the worst possible case $\varepsilon = h$. Finally, we note that Remark 2.4 applies in the present context as well.

Remark 2.6. The results of this section provide estimates on the errors that result as the FC(Gram) continuation functions f^c are used to approximate a function f defined in an extended interval $[-\varepsilon_{\text{left}}, 1+\varepsilon_{\text{right}}]$. In particular, the errors in such (h-dependent) intervals converge to zero in the manner analogous to that indicated in Remark 2.5.

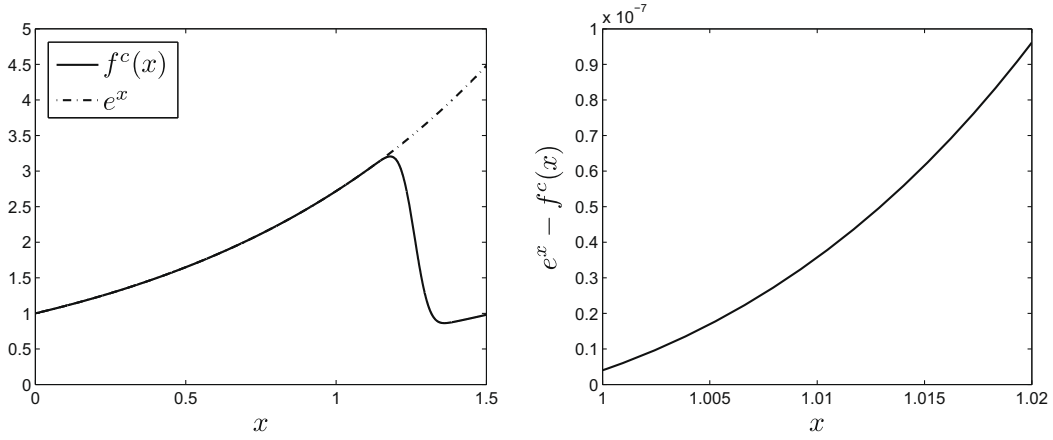


Fig. 5. Left: Fourier Continuation function f^c with periodicity interval $1+d=1.5$ for $f(x)=e^x$ obtained, with $m=4$, from $n=51$ equi-spaced points in the interval $[0,1]$ (including the points $x=0$ and $x=1$); the step-size is $h=0.02$. For reference both f^c and the exponential function e^x are displayed in the full continuation interval $[0,1+d]=[0,1.5]$. Right: Difference $e^x - f^c(x)$ for $x \in [1,1+h]=[1,1.02]$. The estimate (37) with $\zeta_1=\zeta_1^\varepsilon$ and $\zeta_2=\zeta_2^\varepsilon$ given by Eq. (41) and ζ_3 and ζ_4 given by Eq. (39) predicts that, in the present case, the departure between e^x and $f^c(x)$ for $x \in [1,1.02]$ is not larger than 8.0×10^{-7}. Clearly the errors displayed in the right figure are consistent with the theoretical error bound.

Table 3
Maximum values over $[1-\Delta,1+h]\cup[1+d-h,1+d+\Delta]$ of the differences between the even and odd Gram polynomial pairs and their SVD continuations f^r_{even} and f^r_{odd} for the parameter values $n_\Delta=10$, $d/\Delta=26/9$, $g=63$ and $\Upsilon=150$. (Maximum values evaluated as maxima over 2000 points in the set $[1-\Delta,1+h]\cup[1+d-h,1+d+\Delta]$.)

r	$\max\left\lvert\left\{P^r_{\text{left}}, P^r_{\text{right}}\right\} - f^r_{\text{even}}\right\rvert$	$\max\left\lvert\left\{P^r_{\text{left}}, -P^r_{\text{right}}\right\} - f^r_{\text{odd}}\right\rvert$
0	5.6×10^{-17}	2.9×10^{-15}
1	4.8×10^{-14}	5.6×10^{-16}
2	6.7×10^{-15}	1.2×10^{-13}
3	2.2×10^{-12}	2.3×10^{-14}
4	4.8×10^{-13}	6.0×10^{-12}
5	1.2×10^{-10}	2.2×10^{-12}

3. Accuracy of the FC-ODE algorithm

As mentioned in Section 1, the overall FC-AD approach proceeds by reducing a PDE to a series of ODEs that are subsequently solved by means of the FC(Gram)-based ODE solver FC-ODE. In what follows we briefly review the main lines of the FC-ODE algorithm for ODEs arising from the Heat, Wave and Poisson Equations, and we present an error analysis establishing its high-order convergence.

Given an ODE of the form

$$-\alpha^2 u''(x) + u(x) = F(x), \quad u(x_\ell) = B_\ell, \quad u(x_r) = B_r, \tag{42}$$

the FC-ODE algorithm produces an approximate solution $\breve{u}$,

$$\breve{u} = \mathcal{S}^{x_\ell, x_r; B_\ell, B_r}_{\alpha^2, x}[f] \tag{43}$$

(where $\breve{u} = (\breve{u}_j)$ is an approximation of the exact-solution values $u(x_j)$) for a given approximation $f = (f_j)$ of the ODE right-hand-side, $f_j \approx F(x_j)$. We describe our FC-ODE algorithm in what follows.

Remark 3.1. In our PDE applications, α is a simple function of the time-step Δt, see Eq. (85) and Part I. Some of the prescriptions for the FC-ODE solver are introduced to give rise to stability in our overall FC-AD PDE solver.

In its first step, FC-ODE obtains the FC(Gram) continuation Fourier series $f^c(x)$ of (f_j) (see Section 2.1 and Remark 2.2)

$$f^c(x) = \sum_{k \in t(n+n_d-2)} a_k e^{2\pi i \frac{xk}{(x_r - x_\ell)(1+d)}}. \tag{44}$$

Next, including the appropriate solution of the associated homogeneous problem, the function v obtained as

$$v(x) = \sum_{k \in t(n+n_d-2)} \frac{a_k}{1 + \frac{4\alpha^2\pi^2k^2}{(x_r - x_\ell)^2(1+d)^2}} e^{2\pi i \frac{xk}{(x_r - x_\ell)(1+d)}} + c_1 h_1(x) + c_2 h_2(x), \tag{45}$$

solves Eq. (42) with F replaced by f^c, where c_1 and c_2 are constants chosen to fit the boundary conditions in (42) and where the homogeneous solutions are given by,

$$h_1(x) = e^{x/|\alpha|} \quad \text{and} \quad h_2(x) = e^{-x/|\alpha|}. \tag{46}$$

Then, a discrete correction η_j is introduced in the FC-ODE (for full consistency), which is obtained as the solution of the equations

$$-\alpha^2 \frac{\eta_{j+1} - 2\eta_j + \eta_{j-1}}{h^2} + \eta_j = f_j - f^c(x_j), \quad \text{for } j \in \{1, \ldots, n_\Delta\} \cup \{n - n_\Delta + 1, \ldots, n\}, \tag{47}$$

$$\eta_j = 0 \quad \text{otherwise} \tag{48}$$

with boundary conditions $\eta_0 = 0$, $\eta_{n_\Delta+1} = 0$, $\eta_{n-n_\Delta} = 0$ and $\eta_{n+1} = 0$. The discrete solution $\breve{u}_j$ of the ODE under consideration, finally, is constructed as

$$\breve{u}_j = \eta_j - \eta_j^p + (1-\chi)v_j^p + \chi v_j^b, \quad \chi = \min(25\alpha^2/h^2, 1), \tag{49}$$

where the superscripts p and b denote boundary projections (open and closed, respectively) into a polynomial basis of degree m. The open boundary projection (superscript p), which results from use of the inner product defined in Eq. (2) (applied to the discrete data v_j, compare Remark 2.2) coincides with the projection used in step 2 of the FC(Gram) algorithm. The closed boundary projections (superscript b) are defined and computed similarly, using scalar products defined over the sets $\{x_j : j \in S_{\text{left}}\} \cup \{x_r\}$ and $\{x_j : j \in S_{\text{right}}\} \cup \{x_\ell\}$ instead of those given in Eq. (2); see Part I for details. We note that use of this combination of open and closed boundary projections was experimentally determined to insure that the error in the FC-ODE solution tends to zero as α tends to zero, while yielding unconditional stability in the overall FC-AD solver; an extended description in these regards is presented in Part I. This concludes our brief description of the FC-ODE algorithm, and we thus turn to our error analysis of this method.

In order to determine the error of the approximate solution $\breve{u} = (\breve{u}_j)$ given by Eq. (49), assuming $F \in C^k[x_\ell, x_r]$ ($u \in C^{k+2}[x_\ell, x_r]$), we consider the Green's function associated with the differential operator in Eq. (42). For clarity, and without loss of generality, in this section we let $x_\ell = 0$ and $x_r = 1$; notice this notational simplification is different from the one made in Section 2.1, where we set $x_1 = 0$ and $x_n = 1$ instead. (The interval $[0, 1]$ here corresponds to the extended interval $[-\varepsilon_{\text{left}}, 1 + \varepsilon_{\text{right}}]$ of Section 2.3!) The Green's function for zero Dirichlet data for the interval $[x_\ell, x_r] = [0, 1]$ is given by

$$G(x,\tilde{x}) = \begin{cases} \frac{2e^{1/|\alpha|}}{|\alpha|(e^{2/|\alpha|}-1)} \sinh\left(\frac{\tilde{x}}{|\alpha|}\right) \sinh\left(\frac{1-x}{|\alpha|}\right) & \text{for } \tilde{x} < x, \\ \frac{2e^{1/|\alpha|}}{|\alpha|(e^{2/|\alpha|}-1)} \sinh\left(\frac{1-\tilde{x}}{|\alpha|}\right) \sinh\left(\frac{x}{|\alpha|}\right) & \text{for } x < \tilde{x}. \end{cases} \tag{50}$$

Since the function v (defined in Eq. (45)) is the exact solution of

$$-\alpha^2 v''(x) + v(x) = f^c(x), \quad v(x_\ell) = B_\ell, \quad v(x_r) = B_r, \tag{51}$$

it follows that

$$u(x) - v(x) = \int_0^1 (F(\tilde{x}) - f^c(\tilde{x}))G(x,\tilde{x})\, d\tilde{x}. \tag{52}$$

Clearly this quantity is bounded by

$$\|u - v\|_{L^\infty[0,1]} \leqslant \|F - f^c\|_{L^\infty[0,1]} \int_0^1 |G(x,\tilde{x})t|\, d\tilde{x}. \tag{53}$$

Noting that $G(x,\tilde{x}) \geqslant 0$ for $x,\ \tilde{x} \in [0,1]$, and that, further,

$$0 \leqslant \int_0^1 G(x,\tilde{x})\, d\tilde{x} = 1 - \frac{\sinh\left(\frac{1-x}{|\alpha|}\right) + \sinh\left(\frac{x}{|\alpha|}\right)}{\sinh\left(\frac{1}{|\alpha|}\right)} \leqslant 1, \tag{54}$$

we obtain

$$\|u - v\|_{L^\infty[0,1]} \leqslant \|F - f^c\|_{L^\infty[0,1]}. \tag{55}$$

We now consider the error resulting from the open and closed stability-projections. Let u^p and u^b be the open and closed boundary projections of the function u; these functions are approximations of u by polynomials of degree m. The approximation error in these projections can be bounded by a procedure analogous to that used in the context of the error analysis for the FC(Gram) algorithm. Recalling the definitions of ε and Δ^ε from Eq. (40), we have

$$\|u - u^p\|_{L^\infty[0,1]} \leqslant (1 + L^\varepsilon(m, n_\Delta)) \frac{(\Delta^\varepsilon)^{m+1} M_u}{2^{2m+1}(m+1)!} \tag{56}$$

and

$$\|u - u^b\|_{L^\infty[0,1]} \leqslant (1 + L_B^\varepsilon(m, n_\Delta)) \frac{(\Delta^\varepsilon)^{m+1} M_u}{2^{2m+1}(m+1)!}, \tag{57}$$

where M_u is the maximum of the $(m+1)$th derivative of u over the set $[0, x_{n_\Delta}] \cup [x_{n-n_\Delta+1}, 1]$. The new constant $L_B^\varepsilon(m, n_\Delta)$ in Eq. (57) is a quantity analogous to $L^\varepsilon(m, n_\Delta)$ which uses a set of discrete sampling points that includes the relevant boundary point (x_ℓ or x_r) in the boundary projection. A direct evaluation of this quantity, performed in a manner similar to that described in the caption of Table 1, shows that with $n_\Delta = 10$ we have $L_B^\varepsilon(m, n_\Delta) \leqslant 2.1$ for $m = 4$ and $m = 5$.

The discrete correction η_j, which, as discussed in Part I, is used to ensure full consistency of the FC-AD algorithm, is obtained as the solution to Eq. (47). We clearly have

$$\sum_{j=1}^{n_\Delta} (\eta_j)^2 \leqslant \sum_{j=1}^{n_\Delta} (f_j - f^c(x_j))^2 \quad \text{and} \quad \sum_{j=n-n_\Delta+1}^{n} (\eta_j)^2 \leqslant \sum_{j=n-n_\Delta+1}^{n} (f_j - f^c(x_j))^2, \tag{58}$$

since η_j is obtained for each of the two sets $j \in \{1, \ldots, n_\Delta\}$ and $j \in \{n - n_\Delta + 1, \ldots, n\}$ by inverting a symmetric positive definite matrix with eigenvalues greater than or equal to one. Further, $\eta_j - \eta_j^p$ is bounded by

$$\max_{j=1,\ldots,n} \left|\eta_j - \eta_j^p\right| \leqslant \sqrt{n_\Delta} \|F - f^c\|_{L^\infty[0,1]}, \tag{59}$$

since η_j^p is calculated by an orthogonal projection for which the bounds

$$\sum_{j=1}^{n_\Delta} \left(\eta_j - \eta_j^p\right)^2 \leqslant \sum_{j=1}^{n_\Delta} (\eta_j)^2 \quad \text{and} \quad \sum_{j=n-n_\Delta+1}^{n} \left(\eta_j - \eta_j^p\right)^2 \leqslant \sum_{j=n-n_\Delta+1}^{n} (\eta_j)^2 \tag{60}$$

hold. (Note that $\eta_j = \eta_j^p = 0$ for $j = n_\Delta + 1, \ldots, n - n_\Delta$.) The error in the approximation $\breve{u}_j$, defined in Eq. (49) as

$$\breve{u}_j = \eta_j - \eta_j^p + (1 - \chi) v_j^p + \chi v_j^b, \tag{61}$$

can then be bounded by combining the previous bounds. With some algebra using Eqs. (54)–(56) and (58) we obtain

$$\max_{j=1,\ldots,n} |u(x_j) - \check{u}_j| \leqslant \left(\sqrt{n_\Delta + 1} + 2\sqrt{n_\Delta} + 1\right) \|F - f^c\|_{L^\infty[0,1]} + \left(1 + L^\varepsilon(m, n_\Delta) + L_B^\varepsilon(m, n_\Delta)\right) \frac{(\Delta^\varepsilon)^{m+1} M_u}{2^{2m+1}(m+1)!}. \tag{62}$$

Since n_Δ is fixed, $L_B^\varepsilon(m, n_\Delta)$ and $L^\varepsilon(m, n_\Delta)$ are uniformly bounded, and $\Delta^\varepsilon = \mathcal{O}(h)$, we conclude that $\max_{j=1,\ldots,n}|u(x_j) - \check{u}_j| = \mathcal{O}(h^{m+1})$ plus a constant times $\|F - f^c\|_{L^\infty[0,1]}$, as desired, and thus the accuracy of the solution is directly related to the accuracy of the FC(Gram) continuation.

4. Stability and singular-value decompositions

In this section we establish the stability of FC-AD algorithms introduced in Part I for the Heat and Poisson Equations. Our analysis relates stability to a bound on the largest singular values of the matrices associated with the discrete one-dimensional operators $\mathcal{S}_{\alpha^2,x}^{x_\ell,x_r;B_\ell,B_r}$. The conclusions of our analysis are drawn from a numerical evaluation of such singular values for a complete range of all but two of the parameters involved: the two parameters n_{x_i} and n_{y_j} defined below (the numbers of discretization points within the PDE domain in the vertical and horizontal lines passing through x_i and y_j, respectively) are taken to span a large range but, naturally, not the complete (infinite) range of possible values.

Remark 4.1. The stability of finite-difference algorithms can generally be established through consideration of the eigenvalues of certain symmetric matrices, which control the growth and accumulation of errors over the number of time-steps necessary to advance the solution up to a given final time T. In the case of the ADI algorithm a stability analysis *for rectangular D-dimensional domains* ($D = 2, 3, \ldots$) can in some cases be reduced to study of eigenvalues of *one-dimensional* finite-difference matrices. Indeed we have (1) for *rectangular domains*, ADI time-stepping amounts to application of a product of fixed one-dimensional operators [31, p. 599] (the same is in fact true of FC-AD time-stepping for rectangular domains), and (2) each one of the one-dimensional ADI operators mentioned in point (1) are symmetric matrices. Under these two conditions, the Euclidean operator norm of the overall D-dimensional ADI time-stepping matrix (that is used in the ADI stability analysis [31, p.599]) is bounded by the product of the largest eigenvalues of the underlying one-dimensional operators. If the one-dimensional alternating direction operators are not Hermitian, as is the case for both the one-dimensional operators associated with the FC-AD algorithm and with the D-dimensional ADI operator for a non-rectangular domain, the largest eigenvalue does not equal the Euclidean operator norm. As shown in what follows, the FC-AD stability analysis can still be performed by appealing to singular values instead of eigenvalues: below in this section we use singular values of one-dimensional operators to establish stability of the overall D-dimensional FC-AD solver for the heat equation *in general non-rectangular domains*.

In order to facilitate reference to the grid points in the interior of the domain, we introduce some additional notations. Let the bounded open set Ω (the domain of the PDE, see Fig. 1) be contained in $[a_x, b_x] \times [a_y, b_y]$, let

$$\{(x_i, y_j) : 1 \leqslant i \leqslant N_x,\ 1 \leqslant j \leqslant N_y\} \tag{63}$$

be a Cartesian mesh in the rectangle $[a_x, b_x] \times [a_y, b_y]$, and call $\mathcal{D}_\Omega$ the set of mesh points interior to Ω:

$$\mathcal{D}_\Omega = \{(x_i, y_j) \in [a_x, b_x] \times [a_y, b_y] : (x_i, y_j) \in \Omega\}. \tag{64}$$

For a given point $(x_i, y_j) \in \mathcal{D}_\Omega$, let n_{x_i} be the number of points in $\mathcal{D}_\Omega$ on the same vertical line as (x_i, y_j), and let n_{y_j} be the corresponding number of points in $\mathcal{D}_\Omega$ on the same horizontal line.

We also introduce some vector spaces associated with these meshes: we let $\ell_2(n) = \mathbb{R}^n$ be the usual n-dimensional vector space with the Euclidean norm (in our constructions n may be either n_{x_i} or n_{y_j} for some x_i or y_j) and, calling N_Ω the total number of points in $\mathcal{D}_\Omega$, we define the space $\ell_2(\mathcal{D}_\Omega) = \mathbb{R}^{D_\Omega}$—that is, the set of all functions from $\mathcal{D}_\Omega$ to $\mathbb{R}$—with the usual Euclidean norm: for a given $\theta \in \ell_2(\mathcal{D}_\Omega),\ \theta_{ij} = \theta(x_i, y_j)$, the norm of θ is given by

$$\|\theta\|_{\ell_2(\mathcal{D}_\Omega)} = \sqrt{\sum_{(x_i,y_j)\in\mathcal{D}_\Omega} \frac{\theta_{ij}^2}{N_\Omega}}. \tag{65}$$

4.1. Stability: reduction to evaluation of FC-ODE singular values

To start our study we introduce the following definition concerning the FC-ODE solver.

Definition 4.1. Let an n point discretization of the interval $[x_\ell, x_r]$, as depicted in Fig. 2, be given. We define L_1 as the FC-ODE solution operator with boundary-values B_ℓ and B_r: $L_1 = \mathcal{S}_{\alpha^2,x}^{x_\ell,x_r;B_\ell,B_r}$ (see Eq. (43)). Further we define L_2 as the linear map from $\ell_2(n) \to \ell_2(n)$ given by $L_2 f = (2L_1 - I)f$. Note that $\check{q}_j = \sum_{j=1}^n L_{2,ij} f_j$ is an approximation of $q(x) = 2u(x) - f(x)$. (Here and in what follows, the matrices of the operators L_k, $k = 1, 2$, in the underlying canonical basis, are denoted by $L_k = (L_{k,ij})$.)

The FC-ODE solution operator $\mathcal{S}^{x_\ell,x_r;B_\ell,B_r}_{\alpha^2,\lambda}$, which is described in Section 3 and, in more detail, in Section 4 of Part I, is defined in terms of (1) the FC(Gram) Fourier Continuation algorithm; (2) exact solution of ODEs with right-hand-sides given by Fourier series, and (3) certain corrections introduced to insure stability and convergence when FC-ODE is used as part of the FC-AD PDE solvers.

Remark 4.2. The linear maps L_1 and L_2 depend on the FC(Gram) and FC-ODE parameters n, h, n_Δ, m, d/Δ, g, Υ, $\varepsilon_\ell = x_1 - x_\ell$, $\varepsilon_r = x_r - x_n$, B_ℓ, B_r and α. As is appropriate in stability studies, the boundary values B_ℓ, B_r are taken to vanish for the stability analysis presented in this section. For a given domain Ω, a selected time-step Δt (which determines α; see (42) and Part I), a given Cartesian mesh (64), and selected values of n_Δ, m, d/Δ, g, Υ (such as those listed in the caption of Table 2, which are, in fact, the values generically recommended in Part I), the operators L_k depend on three parameters only, namely, the number n of discretization points used and the values ε_ℓ and ε_r; when necessary, in what follows, we make these dependencies explicit through the notations $L_k = L_k^{n,\varepsilon_\ell,\varepsilon_r}$ for $k = 1, 2$.

Our analysis proceeds through consideration of the following versions, defined in all of $\mathcal{D}_\Omega$, of the operators L_1 and L_2.

Definition 4.2. For a given $\theta \in \ell_2(\mathcal{D}_\Omega)$, $\theta_{ij} = \theta(x_i, y_j)$, we define the operators $\mathcal{L}^1_x : \ell_2(\mathcal{D}_\Omega) \to \ell_2(\mathcal{D}_\Omega)$, $\mathcal{L}^1_y : \ell_2(\mathcal{D}_\Omega) \to \ell_2(\mathcal{D}_\Omega)$, $\mathcal{L}^2_x : \ell_2(\mathcal{D}_\Omega) \to \ell_2(\mathcal{D}_\Omega)$ and $\mathcal{L}^2_y : \ell_2(\mathcal{D}_\Omega) \to \ell_2(\mathcal{D}_\Omega)$ by

$$\begin{aligned}
\mathcal{L}^1_x[\theta](x_i, y_j) &= \sum_{k=1}^{n_{y_j}} L_{1,ik}\theta_{kj}, \\
\mathcal{L}^1_y[\theta](x_i, y_j) &= \sum_{k=1}^{n_{x_i}} L_{1,jk}\theta_{ik}, \\
\mathcal{L}^2_x[\theta](x_i, y_j) &= \sum_{k=1}^{n_{y_j}} L_{2,ik}\theta_{kj}, \quad \text{and} \\
\mathcal{L}^2_y[\theta](x_i, y_j) &= \sum_{k=1}^{n_{x_i}} L_{2,jk}\theta_{ik}.
\end{aligned} \tag{66}$$

We note that the Heat Equation FC-AD algorithm can easily be expressed in terms of the operators (66) with $\alpha^2 = \frac{k\Delta t}{2}$ (see Definition 4.1): the approximate solution $\breve{u}^n = \left(\breve{u}^n_{ij}\right)$ for time $t_n = n\Delta t$ is given by

$$\breve{u}^n = \mathcal{L}^1_y\left[\mathcal{L}^2_x[\breve{w}^{n-1}]\right], \tag{67}$$

where $\breve{w}^n$ is evolved according to

$$\breve{w}^n = \mathcal{L}^2_y\left[\mathcal{L}^2_x[\breve{w}^{n-1}]\right] = 2\breve{u}^n - \mathcal{L}^2_x[\breve{w}^{n-1}] \tag{68}$$

with initial condition $\breve{w}^0_{ij} = w_0(x_i, y_j)$, where

$$w_0(x, y) = \left(1 + \frac{k\Delta t}{2}\frac{\partial^2}{\partial y^2}\right)u_0(x, y). \tag{69}$$

Similarly, the approximate solution $\breve{u}^n = \left(\breve{u}^n_{ij}\right)$ produced after n iterations of the FC-AD algorithm for the Poisson Equation (see Section 3.3 of Part I), is given by

$$\breve{u}^n = \mathcal{L}^1_y\left[\mathcal{L}^2_x[\breve{w}^{n-1}]\right], \tag{70}$$

where $\breve{w}^n$ is evolved according to

$$\breve{w}^n = \left(1 + \frac{\gamma_n}{\gamma_{n-1}}\right)\breve{u}^n - \frac{\gamma_n}{\gamma_{n-1}}\mathcal{L}^2_x[\breve{w}^{n-1}] \tag{71}$$

with initial condition $\breve{w}^0_{ij} = 0$.

The operator norms $\|\mathcal{L}^1_x\|_{\ell_2(\mathcal{D}_\Omega)}$, $\left\|\mathcal{L}^1_y\right\|_{\ell_2(\mathcal{D}_\Omega)}$, $\|\mathcal{L}^2_x\|_{\ell_2(\mathcal{D}_\Omega)}$ and $\left\|\mathcal{L}^2_y\right\|_{\ell_2(\mathcal{D}_\Omega)}$ arise from the norm (65) of the space $\ell_2(\mathcal{D}_\Omega)$ of functions of a two-dimensional discrete variable. As it happens, however, these operator norms can be bounded by a maximum of the singular values of a set of operators defined on one-dimensional meshes. Indeed, the "horizontal data lines" $\{\theta : \theta_{ij} = 0 \text{ for } j \neq j_0\}$ form a complete set of invariant subspaces for the operators $\mathcal{L}^1_x$ and $\mathcal{L}^2_x$, while the "vertical data lines" $\{\theta : \theta_{ij} = 0 \text{ for } i \neq i_0\}$ form a complete set of invariant subspaces for the operators $\mathcal{L}^1_y$ and $\mathcal{L}^2_y$. Therefore, as is easily checked, the norms of each one of the four two-dimensional operators (66) is bounded by

$$\max\left\{\left\|L_k^{n,\varepsilon_\ell,\varepsilon_r}\right\| : k = 1, 2,\ 0 \leqslant n \leqslant \max\{N_x, N_y\},\ 0 \leqslant \varepsilon_\ell,\ \varepsilon_r \leqslant h\right\}, \tag{72}$$

see Eq. (63) and Remark 4.2, where $\|\cdot\|$ denotes the operator norm in the space $\ell_2(n)$. The quantity $\|L_k\|$ in Eq. (72), which, as is the case for all finite dimensional operators, equals the maximum singular value of the matrix $(L_{k,ij})$, can be evaluated easily using readily-available linear algebra algorithms.

It follows from the evolution Eqs. (68), (70) and (71) that stability of the FC-AD algorithms for both the Heat Equation and the Poisson Equation is ensured provided the relations

$$\|L_1\|_{\ell_2} \leqslant 1 \quad \text{and} \quad \|L_2\|_{\ell_2} \leqslant 1 \tag{73}$$

are satisfied along with the condition that $\gamma_n \leqslant \gamma_{n-1}$ as prescribed in Section 3.3 of Part I; in Section 4.2, we verify that these stability conditions do indeed hold for the recommended values of the parameters that define our algorithms.

4.2. Evaluation of the maximum singular values $\|L_1\|$ and $\|L_2\|$

As shown in the previous section (see e.g. Eq. (72) and associated narrative), the maximum singular values $\|L_1\|$ and $\|L_2\|$ determine the stability of the FC-AD algorithm for the Heat and Poisson Equations. While, unfortunately, we have not been able to evaluate these singular values in closed form, we found it possible to obtain these quantities numerically in an essentially comprehensive manner. Indeed, as mentioned in Remark 4.1, the linear maps L_1 and L_2 are functions of the parameters n, h, n_Δ, m, d/Δ, g, Υ, $\varepsilon_\ell = x_1 - x_\ell$, $\varepsilon_r = x_r - x_n$ and α. All our applications of FC-AD solver use parameters values n_Δ, d/Δ, g and Υ as shown in the caption of Table 2 (which coincide with the values generically recommended in Part I); based on a range of numerical tests we performed we believe these parameter choices to be essentially optimal for the types of problems we have thus far considered. Therefore we take these parameter values as fixed, and we evaluate the singular values of the operators L_1 and L_2 for adequate ranges of values of the parameters n, α, m, ε_ℓ and ε_r. Using adequately refined discretizations for the parameters α, ε_ℓ and ε_r, we have verified that the stability conditions (73) hold for all values of these parameters, for all $n \leqslant 200$ and a sampling of a wide range of larger values of n, for all $m \leqslant 4$ for the operators L_1 and L_2, and for $m \leqslant 5$ for the operator L_1 (the operator L_2 does not enter in the Wave Equation FC-AD formulation described in Section 5, for which the value $m = 5$ leads to stable solutions in the context of the parameter prescriptions we use otherwise). In what follows we describe some of the numerical experiments that we performed to effect this verification; see also [32].

Fig. 6 displays, as functions of α and for several values of n, numerical approximations of the quantities $1 - \max_{\varepsilon_\ell,\varepsilon_r}\|L_1\|$ and $1 - \max_{\varepsilon_\ell,\varepsilon_r}\|L_2\|$, where the maxima are taken for all admissible values of ε_ℓ and ε_r. This figure was produced by obtaining numerically, for each value on a fine α-mesh, the maxima, over all admissible values of ε_ℓ and ε_r, of each of the singular values $\|L_1\|$ and $\|L_2\|$. The maximum singular values that coincide with the operator norms $\|L_1\|$ and $\|L_2\|$ were calculated using Singular Value Decompositions (SVD) [33], and the maxima over ε_ℓ and ε_r were estimated as maxima over all 121 possible combinations of 11 values for ε_ℓ and 11 values for ε_r—which in view of our experiments were deemed sufficient to produce accurately the maxima of the singular values over all admissible values of ε_ℓ and ε_r. Results shown on the left portion of Fig. 6 correspond to $m = 5$ (sixth-order accuracy); in fact, just like the few $m = 5$ curves shown in this figure, the $m = 5$ curves essentially coincide for all values of n. Thus, these numerical calculations provide strong evidence that the stability relation $\|L_1\| < 1$ is satisfied for all possible values of the parameters n, ε_ℓ and ε_r and α for $m \leqslant 5$. Similarly, the results for the linear map L_2 suggest that the stability condition $\|L_2\| < 1$ is satisfied for $m \leqslant 4$ (up to fifth-order accuracy) for all admissible parameter values. As mentioned

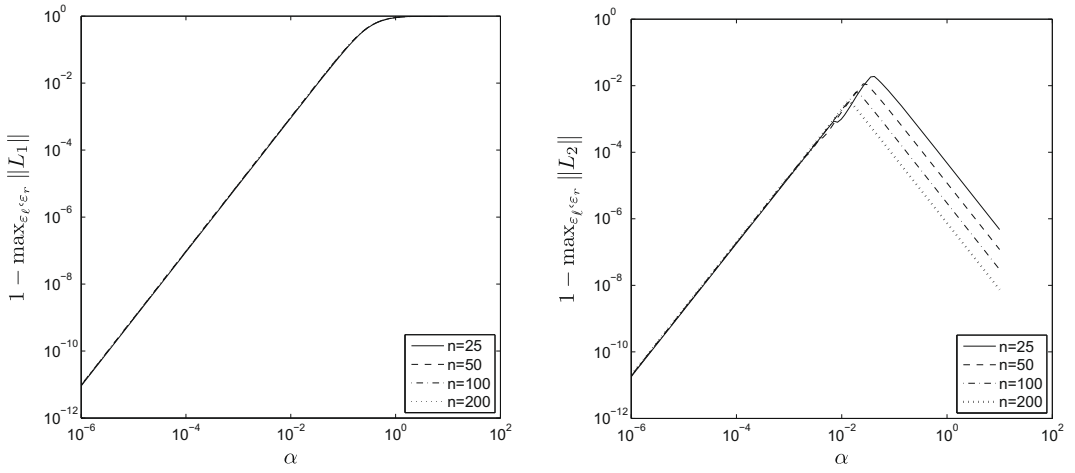


Fig. 6. Left: $1 - \max_{\varepsilon_\ell,\varepsilon_r}\|L_1\|$ for $m = 5$ (sixth-order accuracy), as a function of α for four values of n. Right: $1 - \max_{\varepsilon_\ell,\varepsilon_r}\|L_2\|$ for $m = 4$ (fifth-order accuracy), as a function of α for four values of n. Maxima with respect to ε_ℓ and ε_r were obtained by maximization over a discrete sampling of $\varepsilon_\ell = (x_1 - x_\ell)$ and $\varepsilon_r = (x_r - x_n)$, as indicated in the text, for each fixed value of α.

above, for example, we have performed these calculations for all values $n \leqslant 200$ and many other values larger than 200; the results of some of these calculations are shown on the right portion of Fig. 6. We have never encountered values of the parameters under consideration for which the stability conditions $\|L_1\| < 1$ and $\|L_2\| < 1$ (for $m \leqslant 5$ and $m \leqslant 4$, respectively) are not satisfied.

In contrast, our calculations show that, for $m \geqslant 6$, the stability conditions (73) are violated for some values of α—at least under our present selection $n_\Delta = 10$ of the sampling parameter—and, thus, for this value of n_Δ, the choice $m \geqslant 6$ leads to conditional stability only. Conceivably other choices of the sampling parameter n_Δ might restore stability for polynomial degrees $m \geqslant 6$. A study of the dependence of the stability domain on the ratio n_Δ/m is left for future work.

5. FC-AD algorithm for the Wave Equation

We now consider an extension of the FC-AD methodology to the Wave Equation in two- and three-dimensional spatial domains, and we present a variety of numerical results obtained from this algorithm—demonstrating high-order accuracy and unconditional stability. We consider at first the two-dimensional Wave Equation with Dirichlet boundary data

$$\begin{aligned} &u_{tt} = k^2(u_{xx} + u_{yy}) + Q(x,y,t), \quad \text{in } \Omega \times (0,T], \\ &u(x,y,t) = G(x,y,t), \quad (x,y) \in \partial\Omega, \quad t \in (0,T], \\ &u(x,y,0) = W(x,y), \quad (x,y) \in \Omega, \\ &u_t(x,y,0) = V(x,y), \quad (x,y) \in \Omega, \end{aligned} \tag{74}$$

where $\Omega \subset \mathbb{R}^2$ is a smoothly bounded domain, and where Q, G, W and V are given smooth functions defined either in Ω, in $\Omega \times (0,T]$ or in $\partial\Omega \times (0,T]$, as appropriate. Our experiments indicate that use of a centered second-order accurate time discretization

$$\frac{u^{n+1} - 2u^n + u^{n-1}}{\Delta t^2} = k^2\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\frac{u^{n+1} + u^{n-1}}{2} + Q^n, \tag{75}$$

similar to that used for the Heat Equation in Part I, leads to long time instability due to the occurrence of eigenvalues with small imaginary components in the resulting fully-discrete scheme. To avoid this difficulty we use, instead, the first order scheme (76) below, whose first order truncation error gives rise to stability. It is easy to recover the accuracy that is lost as a result of the additional diffusion, however: as shown in Section 6, use of Richardson extrapolation can be made to regain second-order accuracy and, in fact, to obtain even higher orders of temporal accuracy (compare [34]).

Thus, to derive our FC-AD splitting for the Wave Equation we first consider the time discretization

$$\frac{u^{n+1} - 2u^n + u^{n-1}}{\Delta t^2} = k^2\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)u^{n+1} + Q^n + \frac{E_1(x,y,t)}{\Delta t^2}, \tag{76}$$

where

$$E_1(x,y,t) \leqslant k^2\Delta t^3\|u_{txx}\|_{L^\infty(\Omega\times(t^n,t^{n+1}))} + k^2\Delta t^3\|u_{tyy}\|_{L^\infty(\Omega\times(t^n,t^{n+1}))} + \frac{\Delta t^4}{12}\|u_{tttt}\|_{L^\infty(\Omega\times(t^n,t^{n+1}))}. \tag{77}$$

Eq. (76) can be re-expressed in the form

$$\left(1 - k^2\Delta t^2\frac{\partial^2}{\partial x^2} - k^2\Delta t^2\frac{\partial^2}{\partial y^2}\right)u^{n+1} = 2u^n - u^{n-1} + \Delta t^2 Q^n + E_1(x,y,t). \tag{78}$$

The operator on the left hand side of Eq. (78) may be split creating an additional error

$$\left(1 - k^2\Delta t^2\frac{\partial^2}{\partial x^2}\right)\left(1 - k^2\Delta t^2\frac{\partial^2}{\partial y^2}\right)u^{n+1} = 2u^n - u^{n-1} + \Delta t^2 Q^n + E_1(x,y,t) + E_2(x,y,t), \tag{79}$$

where

$$E_2(x,y,t) = \frac{k^4\Delta t^4}{4}\frac{\partial^2}{\partial x^2}\frac{\partial^2}{\partial y^2}u^{n+1} \leqslant \frac{k^4\Delta t^4}{4}\|u_{xxyy}\|_{L^\infty(\Omega\times(t^n,t^{n+1}))}. \tag{80}$$

Then by inverting the operators on the left hand side and discarding the error terms we obtain

$$\tilde{u}^{n+1} = \left(1 - k^2\Delta t^2\frac{\partial^2}{\partial y^2}\right)^{-1}\left(1 - k^2\Delta t^2\frac{\partial^2}{\partial x^2}\right)^{-1}(2u^n - u^{n-1} + \Delta t^2 Q^n) \tag{81}$$

corresponding to the discrete alternating direction algorithm

$$\tilde{u}^{n+1} = \mathcal{L}_y^1 \mathcal{L}_x^1 (2u^n - u^{n-1} + \Delta t^2 Q^n), \tag{82}$$

where the operators $\mathcal{L}_y^1$ and $\mathcal{L}_x^1$ are those introduced in Definition 4.2. The boundary conditions for each inverse operation are given by $G(x, y, t^{n+1})$; all together, the application of each one of these inverse operators, including their corresponding boundary conditions, amounts to the solution of an uncoupled set of ODEs of the form (42). As in the Heat Equation algorithm presented in Part I, this choice of boundary data yields additional truncation error, E_3, which is bounded by

$$E_3(x, y, t) \leqslant \frac{k^2 \Delta t^2}{4} \|u_{yy}\|_{L^\infty(\Omega \times (t^n, t^{n+1}))}. \tag{83}$$

The overall error that arises as the discrete algorithm (82) is used to evolve a numerical solution up to a certain time T results, naturally, as a composite of occurrences and propagation of truncation errors (77), (80) and (83) at various time-steps. Unfortunately, if it was established that the algorithm under consideration amplifies single time-step errors by a factor that grows linearly with time, as is typically done in the analysis of finite-difference methods (see e.g. Eq. (8.2.1) in [35, p. 193]) then the convergence of the algorithm could not be established. Indeed, under such linear-growth condition, if an error E is introduced in a given step of the algorithm, the error after $\frac{T}{\Delta t}$ additional steps may be amplified by a factor of the order of $\frac{T}{\Delta t}$. Assuming an error of $|E_1| + |E_2| + |E_3|$ error is introduced at each of the $\frac{T}{\Delta t}$ steps of the algorithm, the ultimate error at time T would be estimated to not exceed a constant times $\frac{|E_1|+|E_2|+|E_3|}{\Delta t^2}$. Owing to the E_3 error term, and in contrast with the behavior observed in practice, such an approach would not establish the convergence of the algorithm: the error bound thus obtained does not tend to zero as $\Delta t \to 0$.

In practice it is observed that, for the basic algorithm, without use of Richardson extrapolation, the overall error at a fixed time T is a quantity of order $\mathcal{O}(\Delta t)$ in spite of the presence of the $\mathcal{O}(\Delta t)^2$ error E_3. Without embarking in a full theoretical study of the properties of the algorithm (82), we suggest the observed convergence should relate to the fact that, as is established by energy bounds such as [36, p. 381] for the continuous Wave Equation, the overall error resulting from evolution up to a fixed time T of boundary errors originating at a single time-step should be bounded by an adequate norm of the boundary error itself—and, thus, the overall error arising from all of the needed $\frac{T}{\Delta t}$ time-steps should be of the order of $\mathcal{O}(\Delta t)$, as observed in our numerical experiments.

The FC-AD algorithm for three-dimensional problems takes a similar form

$$\tilde{u}^{n+1} = \mathcal{L}_z^1 \mathcal{L}_y^1 \mathcal{L}_x^1 (2u^n - u^{n-1} + \Delta t^2 Q^n). \tag{84}$$

The solutions of Eqs. (82) and (84) are obtained by solving ODEs of the form (42), with

$$\alpha = k\Delta t, \tag{85}$$

by means of the FC-ODE algorithm. In this paper we use $m = 5$ for all cases concerning wave equations; see Section 4.2 for a discussion concerning stability for various PDEs and various values of m.

We have demonstrated numerically the unconditional stability of the FC-AD Wave Equation algorithm for a variety of geometries by both, explicit calculation of eigenvalues of the full D-dimensional time-stepping operators, and through a wide range of long-time runs; a sampling of such results are presented in Section 6. While not reducing the Wave Equation stability analysis to properties of one-dimensional geometry-independent operators (which was indeed accomplished for the Heat- and Poisson-Equation algorithms, see Section 4) the results of these tests are significant: after very many computational runs for a variety of geometries, we have never observed an instability—for any time-step/spatial-mesh-size whatsoever.

6. Numerical results for the Wave Equation

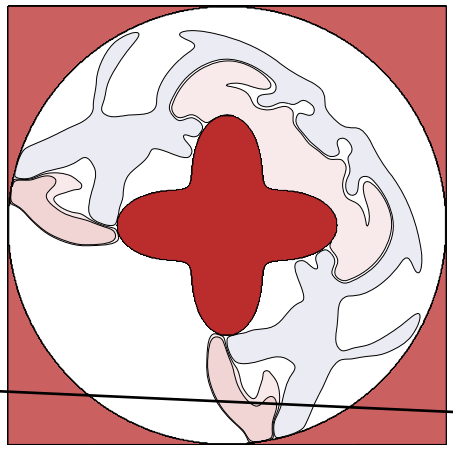
In what follows we demonstrate the convergence and unconditional-stability properties of the Wave Equation FC-AD algorithms of sixth-order spatial accuracy ($m = 5$) and various orders of temporal accuracy—the latter of which are obtained by means of Richardson extrapolation, see e.g. [34]. (The Richardson extrapolation technique combines a number N of first order solutions for various time-steps, $\Delta t = \Delta t_1, \Delta t_2, \ldots, \Delta t_N$ to produce a solution with error of the order of $\mathcal{O}(\Delta t^N)$.)

Remark 6.1. Our numerical experiments indicate that, in order to obtain full N-order convergence from the Richardson extrapolation algorithm, it must be ensured that the parameter χ (49) is either taken to equal one for all of the time-steps Δt used, or taken to be less than one for all of the time-steps Δt used. In practice this does not amount to a significant restriction.

In our first example we consider the Wave Equation with $k = 1$ on a complex two-dimensional domain contained within a circle of radius 0.5 centered at $x = 0.5$ and $y = 0.5$ and outside the curve defined parametrically by

$$x(\theta) = \frac{(1 + \cos(2\theta))\cos(\theta)}{8} - \frac{1}{2}, \quad y(\theta) = \frac{(1 + \cos(2\theta))\sin(\theta)}{8} - \frac{1}{2} \tag{86}$$

for $\theta \in [0, 2\pi]$; see Fig. 7. We assumed Gaussian initial data of the form



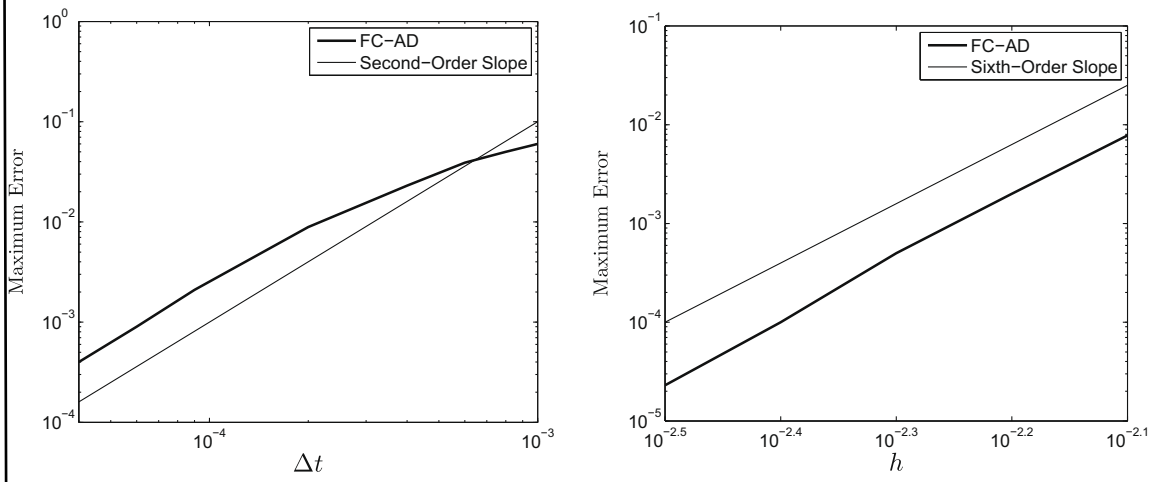


Fig. 8. Maximum error for the Wave Equation with Gaussian initial data, with fixed $h = 0.002$ on the left and $\Delta t = 1/3000$ on the right.

$$W(x,y) = u(x,y,0) = e^{-1000\{(x-0.75)^2+(y-0.75)^2\}}, \tag{87}$$

together with zero values for the boundary condition G, the forcing function Q and the $t = 0$ time derivative V. The solution was calculated using Richardson extrapolation to second-order up to a final time $T = 1$, at which point the wave resulting from the pulse had travelled through most of the domain; a snapshot at $T = 1$ is displayed in Fig. 7.

A variety of convergence studies were performed to assess the rates of convergence of the solutions provided by the FC-AD algorithm for this problem. At first a (fine) spatial resolution was fixed at $h_x = h_y = 0.002$ and time-steps ranging from $\Delta t = 1/800$ to $\Delta t = 1/25,600$ were used to study the order of temporal accuracy; solution errors were obtained through comparisons with the $\Delta t = 1/51,200$ solution. (In each case, Richardson extrapolation was performed using calculations up to time T using time-steps Δt and $\Delta t/2$.) The temporal convergence is demonstrated on the left portion of Fig. 8; the expected second order accuracy in time is observed. To study the convergence as the spatial discretization is refined, the value $\Delta t = 1/3000$ was fixed and the value of $h_x = h_y$ was allowed to vary from 0.008 to 0.003 and compared with the reference solution calculated with $h_x = h_y = 0.001$. The resulting errors are shown on the right portion of Fig. 8, demonstrating the high-order spatial convergence of the FC-AD algorithm. The unconditional stability of the method enables accurate and stable solution in spite of the extremely small mesh segments that arise next to the domain boundaries.

As an additional example we consider once again the $k = 1$ Wave Equation over the domain whose boundary is defined by the curve $(x-0.5)^4 + (y-0.5)^4 = 1/16$. The functions Q, G, W and V were chosen in such a way that the exact solution equals

$$u(x,y,t) = \sin\left(\sqrt{85}\pi(x-t)\right) + \sin\left(\sqrt{85}\pi(y-t)\right). \tag{88}$$

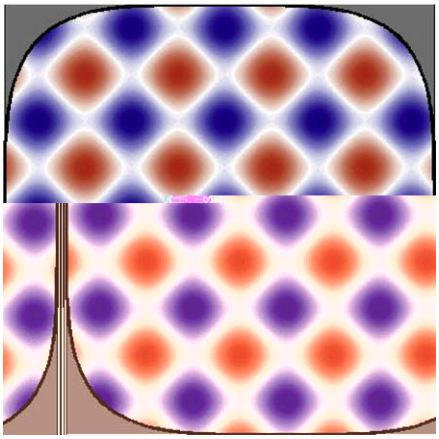


Fig. 9. Solution u of a PDE problem used in convergence studies of the FC-AD Wave Equation algorithm.

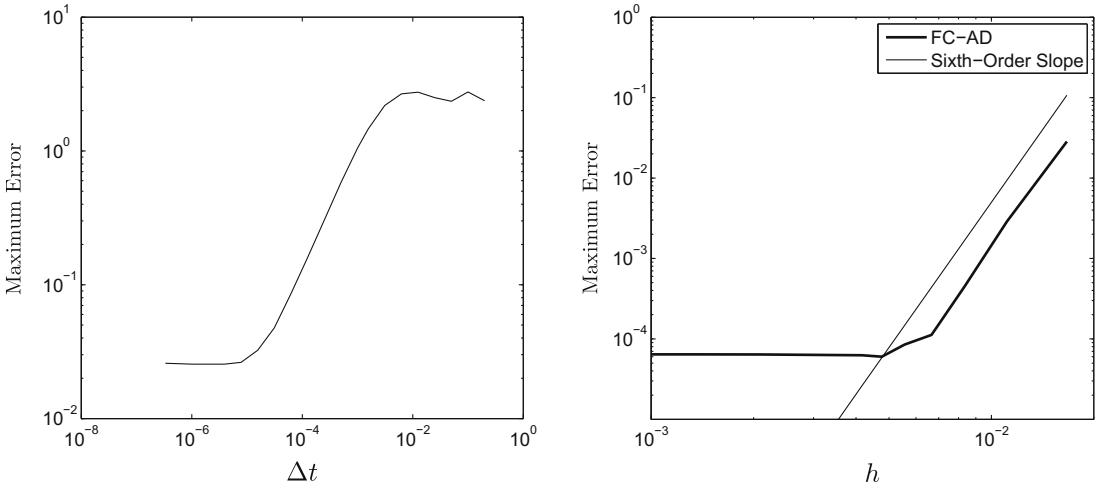


Fig. 10. Maximum errors resulting from applications of the FC-AD Wave Equation algorithm to a problem with solution given by Eq. (88). Left: Errors with $h = 0.01667$ fixed, as a function of Δt. Right: Errors resulting from use of the FC-AD algorithm in conjunction with fourth-order Richardson extrapolation in time.

The function $u(x, y, t)$ at the final time $T = 1$ is depicted in Fig. 9. For a fixed mesh with $h = 0.01667$, various time-steps were chosen ranging from 0.2 to 3.33×10^{-7}; the maximum error at any time-step during the calculation is shown on the left of Fig. 10. Clearly, stability occurs for a very wide range CFL numbers—certainly, owing to the smallness of the boundary mesh-segments, much larger, in fact, than the nominal CFL ratio implicit in the largest $\Delta t/h$ quotient, namely, $\Delta t/h = 0.2/0.01667 = 12 \gg 1$—giving yet another indication of unconditional stability.

The solution of this problem was subsequently obtained using Richardson extrapolation: in Fig. 11 we show the errors resulting from calculations using fourth-order Richardson extrapolation in conjunction with our FC-AD scheme, in which both the time-step and the spatial mesh-size are refined simultaneously. The error is primarily dependent on the time-step and therefore the fourth-order Richardson-extrapolation convergence is observed. Additional improvements should result from use of a restarted version [34] of the Richardson extrapolation methodology.

The full sixth-order spatial convergence can, of course, be observed in this example provided sufficiently small time-steps are used. To demonstrate this, for each element in a representative set of h values in the range $0.001 \leqslant h \leqslant 0.01667$, we used first order solutions obtained for $\Delta t = 5 \times 10^{-4},\ 2.5 \times 10^{-4},\ 1.667 \times 10^{-4}$ and 1.25×10^{-4} to produce fourth-order Richardson solutions with sufficiently small temporal errors so that the six-order spatial convergence is realized. The errors thus obtained are displayed on the right portion of Fig. 10.

To conclude this section we demonstrate the FC-AD techniques for linear hyperbolic equations in complex *three-dimensional* domains. We thus present results for a three-dimensional Wave Equation, with $k = 1$, in the domain given by

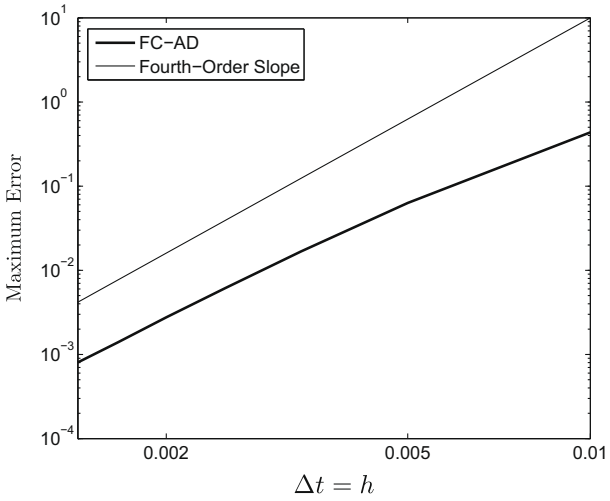


Fig. 11. Numerical errors arising in the solution of a Wave Equation, with exact solution given in Eq. (88), as a function of $\Delta t = h$. Fourth-order Richardson extrapolation was used. The coarsest time-step of the extrapolation was taken to equal h; the other time-steps used were $\Delta t = h/2,\ h/3$ and $h/4$. Fourth-order convergence is observed as h and therefore Δt are refined.

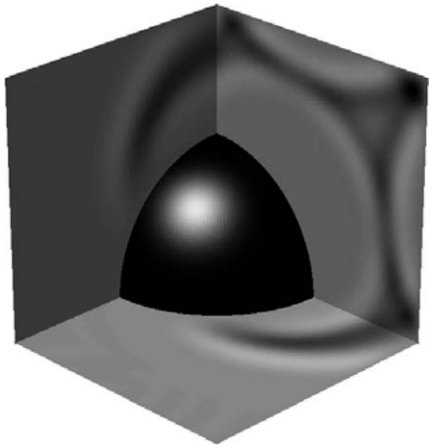


Fig. 12. Solution to the Wave Equation with Gaussian initial data in a domain consisting of the complement of a sphere within a cube. The gray-scale on the three planar sections $x = 0.5025,\ y = 0.5025$ and $z = 0.5025$ display the planer values of the solution.

the complement of the sphere of radius 1/4 centered at $(1/2, 1/2, 1/2)$ in the unit cube $(0, 1)^3$. The solution is defined by initial data given by the Gaussian function

$$u(x, y, z, 0) = e^{-2000\{(x-0.5)^2+(y-0.75)^2+(z-0.75)^2\}},$$

a vanishing initial time derivative, and zero boundary conditions on both the surface of the sphere and the surrounding cube. A spatial step $h = 1/199$ was used giving rise to approximately 7.3 million unknowns. Each time-step required less than ten seconds of computing time on a single core of a 2.33 GHz processor. Fig. 12 displays the values of the solution on the three planes $x = 0.5025,\ y = 0.5025$ and $z = 0.5025$ for a selected point in time. Without Richardson Extrapolation, the three-dimensional algorithm is first-order accurate in time with sixth-order spatial accuracy. Our experiments indicate that the present three-dimensional FC-AD Wave Equation solver is unconditionally stable and, like its two-dimensional counterpart, is fast (it runs in times that grow only linearly with the size of the discretization, at approximately one second per time-step per million unknowns in a single-processor of present day PC) as well as highly accurate.

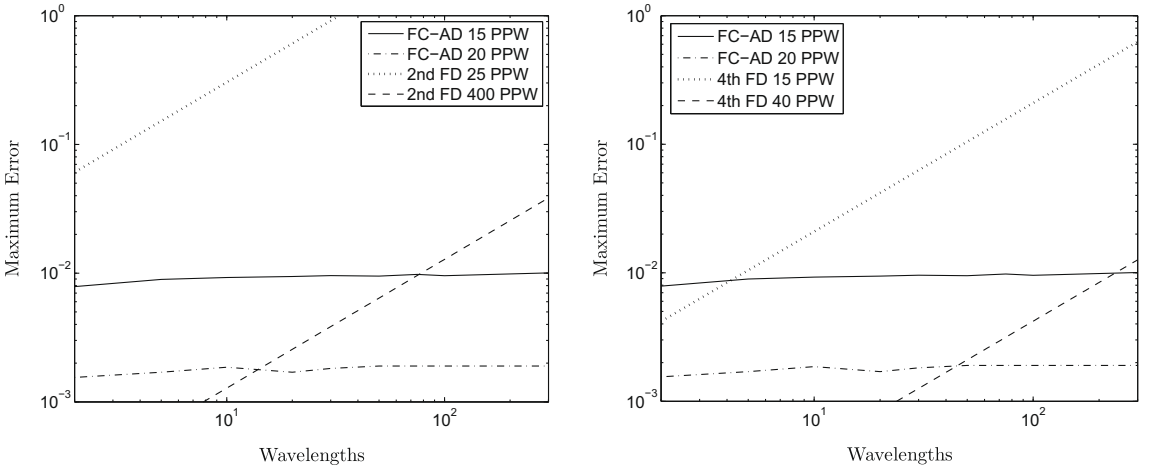


Fig. 13. Maximum errors arising from applications of FC-AD and finite-difference algorithms for increasing number of wavelengths and various fixed numbers of PPW. Second-order finite differences on the left and fourth-order finite differences on the right.

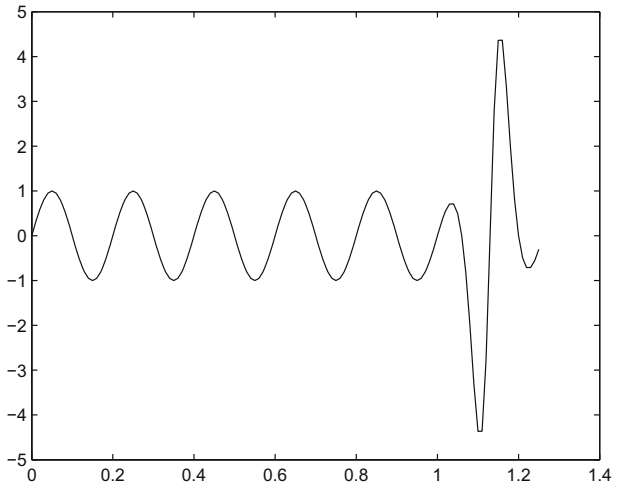


Fig. 14. FC(Gram) continuation of the otherwise periodic function $\sin(10\pi x)$.

7. Transient and time-harmonic wave propagation: pollution error

As mentioned in Section 1, classical FEM and FDM wave-propagation algorithms typically require very large numbers of points-per-wavelength (PPW) in large-scale problems [18,17,16]: the errors produced in finite-difference and finite-element representations compound over the lengths of PDE domains, requiring, for a given error tolerance, increasing numbers of discretization points *per wavelength* as the number of wavelengths is increased. As demonstrated in this section, the FC-AD algorithms do not suffer from this significant drawback.

To demonstrate the advantages arising from use of FC-AD algorithms, we consider a simple wave-propagation problem: the one-dimensional Wave Equation $u_{tt} = u_{xx}$ in the unit intervals of time and space—with initial and periodic boundary conditions selected in such a way that the exact solution is given by $u(x,t) = \sin(2\pi w(x-t))$. In view of the periodicity of the solution and the regularity of the spatial grid used, the finite-difference algorithm exhibits nearly optimal performance in this case. The FC-AD implementation, in contrast, does not benefit from periodicity as it was set to use continuation functions such as the one displayed in Fig. 14. On the left portion of Fig. 13, we display the errors arising for this problem from both, a spatially second-order explicit finite-difference scheme and our FC-AD algorithm. The time-step used for both algorithms was taken to be small enough so as to have no significant effect on the numerical error for any of the wavelengths considered. The maximum error for all time-steps at all discretization points is reported in the figure for 15 and 20 PPW for the FC-AD method, and for 25 and 400 PPW for the second-order finite-difference method. We note that, unlike the FDM, the FC-AD algorithm yields similar performance for an arbitrarily shaped domain in two- and three-dimensional space. In any case, Fig. 13 demonstrates the significant performance advantages that arise from use of the FC-AD algorithm for wave propagation problems, even for the present, extremely simple (and highly advantageous for FDM) one-dimensional configuration. A

similar graph, presented on the right portion of Fig. 13, compares the FC-AD method with results obtained by means of a fourth-order finite-difference method.

Clearly, lower-order methods are exceedingly costly for problems whose spatial dimensions span many wavelengths. The improvement in the performance from second- to fourth-order finite-difference methods is remarkable, but even with fourth-order finite-difference methods the growth in the required number of points per wavelength makes 3D calculations prohibitive. As mentioned in Section 1, current research on finite-difference methods on complex domains without use of domain mappings (including the SAT and Embedded Boundary methods) still focuses on development of second- and fourth-order accurate formulations (typical EB formulations of higher order of accuracy on non-rectangular domains have not proven reliably stable thus far), for which the dispersion/pollution error presents a significant challenge. In contrast, the number of points per wavelength required by the FC-AD methodology for a given accuracy remains essentially constant as the sizes of the problems are increased—and therefore, the FC-AD approach offers as significant a capability for solution of wave propagation and scattering problems, as it does for the parabolic and elliptic problems discussed in Part I.

8. Conclusions

In this contribution we studied the properties of the previously-introduced FC-AD algorithm for the Heat and Poisson Equations, and we extended the FC-AD method to hyperbolic PDEs through applications to the Wave Equation in one, two and three spatial dimensions. In particular, an analysis of the spatial accuracy of the FC(Gram) approximation (a central component to the FC-AD) was presented, along with a corresponding study of the accuracy of the FC-ODE solvers for the types of ODEs that result from the alternating direction splitting of the Heat, Poisson and Wave Equations. A key contribution in this paper is a stability criterion for the parabolic and elliptic algorithms that was used to establish unconditional stability of these solvers for general domains on the basis of numerical evaluation of singular values of certain one-dimensional (geometry independent) operators. The unconditionally stability and high-order spatial and temporal accuracy for the Wave Equation FC-AD solver (with high-order temporal accuracy achieved by means of Richardson extrapolation) were demonstrated through a variety of numerical examples. It was also demonstrated that the FC-AD methodology does not suffer from the debilitating dispersion/pollution effect that underlies finite-difference and finite-element formulations.

Acknowledgments

We gratefully acknowledge support by the Air Force Office of Scientific Research and the National Science Foundation.

References

[1] O. Bruno, M. Lyon, High-order unconditionally-stable FC-AD solvers for general smooth domains I: Basic elements, J. Comput. Phys. 229 (6) (2010) 2009–2033.
[2] D. Peaceman, H. Rachford Jr., The numerical solution of parabolic and elliptic differential equations, J. Soc. Ind. Appl. Math. 3 (1) (1955) 28–41.
[3] J. Douglas Jr., Alternating direction methods for three space variables, Numer. Math. 4 (1962) 41–63.
[4] J. Douglas Jr., C. Pearcy, On convergence of alternating direction procedures in the presence of singular operators, Numer. Math. 5 (1963) 175–184.
[5] J. Douglas Jr., J. Gunn, A general formulation of alternating direction methods. Part I. Parabolic and hyperbolic problems, Numer. Math. 6 (1964) 428–453.
[6] J. Douglas Jr., H. Rachford Jr., On the numerical solution of heat conduction problems in two and three space variables, Trans. Am. Math. Soc. 82 (2) (1956) 421–439.
[7] A. Averbuch, L. Vozovoi, Two-dimensional parallel solver for the solution of Navier–Stokes equations with constant and variable coefficients using ADI on cells, Parallel Comput. 24 (1998) 673–699.
[8] K. Eckhoff, On a high order numerical method for solving partial differential equations in complex geometries, J. Sci. Comput. 12 (2) (1997) 119–138.
[9] O. Næss, K. Eckhoff, A modified Fourier Galerkin method for the Poisson and Helmholtz equations, J. Sci. Comput. 17 (2002) 529–539.
[10] G. Zhao, Q. Liu, The unconditionally stable pseudospectral time-domain (PSTD) method, IEEE Microw. Wireless Compon. Lett. 13 (11) (2003) 475–477.
[11] K. Morton, D. Mayers, Numerical Solution of Partial Differential Equations, second ed., Cambridge University Press, Cambridge, 2005.
[12] N. Kantartzis, T. Zygiridis, T. Tsiboukis, An unconditionally stable higher order ADI-FDTD technique for the dispersionless analysis of generalized 3-D EMC structures, IEEE Trans. Magn. 40 (2) (2004) 1436–1439.
[13] W. Dyksen, Tensor product generalized ADI methods for separable elliptic problems, SIAM J. Numer. Anal. 24 (1) (1987) 59–75.
[14] R. Lynch, J. Rice, D. Thomas, Direct solution of partial difference equations by tensor product methods, Numer. Math. 6 (1964) 185–199.
[15] S. Abarbanel, A. Ditkowski, Asymptotically stable fourth-order accurate schemes for the diffusion equation on complex shapes, J. Comput. Phys. 133 (1997) 279–288.
[16] T. Colonius, S. Lele, Computational aeroacoustics: progress on nonlinear problems of sound generation, Prog. Aerospace Sci. 40 (2004) 345–416.
[17] L. Jameson, High order schemes for resolving waves: number of points per wavelength, J. Sci. Comput. 15 (4) (2000) 417–433.
[18] I. Babuska, S. Sauter, Is the pollution effect of the FEM avoidable for the Helmholtz equation considering high wave numbers?, SIAM J Numer. Anal. 34 (6) (1997) 2392–2423.
[19] S. Abarbanel, A. Ditkowski, A. Yefet, Bounded error schemes for the wave equation on complex domains, J. Sci. Comput. 26 (2006) 67–81.
[20] D. Appelö, N. Petersson, A fourth-order accurate embedded boundary method for the wave equation, preprint.
[21] J. Li, L. Greengard, High order marching schemes for the wave equation in complex geometry, J. Comput. Phys. 198 (2004) 295–309.
[22] S. Wandzura, Stable, high-order discretization for evolution of the wave equation in 2 + 1 dimensions, J. Comput. Phys. 199 (2004) 763–775.
[23] J. Hesthaven, T. Warburton, Nodal Discontinuous Galerkin Methods, Springer, New York, 2007.
[24] B. Alpert, L. Greengard, T. Hagstrom, An integral evolution formula for the wave equation, J. Comput. Phys. 162 (2) (2000) 536–543.
[25] O. 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, pp. 43–82.
[26] J. Boyd, A comparison of numerical algorithms for Fourier extension of the first, second, and third kinds, J. Comput. Phys. 178 (2002) 118–160.

[27] O. Bruno, Y. Han, M. Pohlman, Accurate, high-order representation of complex three-dimensional surfaces via Fourier-continuation analysis, J. Comput. Phys. 227 (2007) 1094–1125.
[28] G.M. Phillips, Interpolation and Approximation by Polynomials, CMS Books in Mathematics, Springer, New York, 2003.
[29] D. Jackson, The Theory of Approximation, vol. 11, American Mathematical Society Colloquium Publications, American Mathematical Society, New York, 1930.
[30] C. Canuto, M. Hussaini, A. Quarteroni, T. Zang, Spectral Methods Fundamentals in Single Domains, Scientific Computation, Springer, Berlin, 2006.
[31] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, second ed., Springer, New York, 2004.
[32] M. Lyon, High-order unconditionally-stable FC-AD PDE solvers for general domains, Ph.D. Thesis, California Institute of Technology, 2009.
[33] G. Golub, C. Van Loan, Matrix Computations, third ed., The John Hopkins University Press, Baltimore, 2006.
[34] B. Fornberg, J. Zuev, J. Lee, Stability and accuracy of time-extrapolated ADI-FDTD methods for solving wave equations, J. Comput. Appl. Math. 200 (2007) 178–192.
[35] J. Strikwerda, Finite Difference Schemes and Partial Differential Equations, second ed., SIAM, Philadelphia, 2004.
[36] L. Evans, Partial Differential Equations, American Mathematical Society, Providence, 1998.