Journal of Computational Physics **170,** 696–719 (2001)
doi:10.1006/jcph.2001.6757, available online at http://www.idealibrary.com on IDEAL®

Transparent Boundary Conditions for Split-Step Padé Approximations of the One-Way Helmholtz Equation

Frank Schmidt,* Tilmann Friese,† and David Yevick†

**Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustr. 7, D-14 95 Berlin, Germany; and †Department of Physics, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada*
E-mail: frank.schmidt@zib.de

Received June 29, 2000; revised February 19, 2001

In this paper, we generalize the nonlocal discrete transparent boundary condition introduced by F. Schmidt and P. Deuflhard (1995, *Comput. Math. Appl.* **29**, 53–76) and by F. Schmidt and D. Yevick (1997, *J. Comput. Phys.* **134**, 96–107) to propagation methods based on arbitrary Padé approximations of the two-dimensional one-way Helmholtz equation. Our approach leads to a recursive formula for the coefficients appearing in the nonlocal condition, which then yields an unconditionally stable propagation method. © 2001 Academic Press

Key Words: Helmholtz equation; wide-angle approximation; transparent boundary conditions; finite-element method.

CONTENTS

1. INTRODUCTION

Scalar wave propagation in two dimensions is generally modeled by the Helmholtz equation

$$u_{zz} + u_{xx} + k^2 u = 0 \tag{1}$$

over the entire $\mathcal{R}^2$ domain, where the wavenumber $k = k(x, z)$ is generally position dependent. In many physical situations, we can further distinguish a principal propagation direction, here taken to be the z-direction, and a transverse x-direction. In the particular case of a position-independent wavenumber, the operator $\partial_z^2 + \partial_x^2 + k^2$ can be explicitly factorized, leading to the exact one-way Helmholtz equation

$$\partial_z u = ik\sqrt{1 + \frac{1}{k^2}\partial_x^2}\, u. \tag{2}$$

In the above expression, the formal square root operator is a pseudo-differential operator, which can be given a precise meaning in the Fourier representation. The associated initial value problem must generally be solved on the domain $\Omega = \mathcal{R} \times \mathcal{R}_0^+$. This requires that the pseudo-differential square root operator be evaluated in a basis of eigenfunctions of its operand, which are free-space Fourier components if k is x-independent.

While (2) is difficult to solve exactly for spatially varying $k(x)$, it provides a foundation for numerous beam propagation methods. Perhaps the most straightforward of these are the split-step fast Fourier transform methods, in which the square root operator is computed directly in Fourier space. Alternatively, approximations such as the fixed-point iteration or rational approximations to the square root operator $\sqrt{1+X}$ can be applied to transform (2) into more easily handled differential equations. These approximations can be formally written as

$$\partial_z u = ik\frac{\left(1 - b_0'\partial_x^2\right)\left(1 - b_1'\partial_x^2\right)\cdots\left(1 - b_m'\partial_x^2\right)}{\left(1 - b_0\partial_x^2\right)\left(1 - b_1\partial_x^2\right)\cdots\left(1 - b_n\partial_x^2\right)}u \tag{3}$$

with complex coefficients $b_0', \ldots, b_m'$ and $b_0, \ldots, b_n$. The approximation quality and the well-posedness of this formal equation have been extensively examined in [2, 16]. Additionally, the non-commutativity of the factors in (3) in the case of non-constant coefficients $b_0', \ldots, b_m'$ and $b_0, \ldots, b_n$ on the computational domain is the source of several theoretical issues which are not discussed here. Rather, we adopt the usual technique of "frozen coefficients," in which the functions are considered to be constant in deriving the rational approximation [2].

From a physical perspective, although originally developed in computational underwater acoustics, propagation methods based on (3) have proved critically important in many other contexts, as evident from some representative references, e.g. [4, 5, 7, 9, 17]. A high-order Padé approximation to the Helmholtz propagator ensures that all widely divergent beams are described to great accuracy. At the same time, replacing the Padé approximant by its partial fraction representation leads to an equivalent formalism that is ideally suited to parallel processing.

In the following sections, we will accordingly investigate the discrete solution to the following problems:

1. Let the coefficients $b_0', \ldots, b_m'$ and $b_0, \ldots, b_n$ be piecewise continuous real functions on the bounded computational domain $\Omega := [x_-, x_+] \times [0,\ z_{\max}]$ and real constants outside the computational domain.

2. Next, consider (3) on the unbounded x–z-domain $\mathcal{R} \times [0,\ z_{\max}]$ with the initial condition $u(x, 0) = u_0(x)$ compactly supported in the interval $[x_-, x_+] \subset \mathcal{R}$ such that the asymptotic boundary condition $\lim_{|x|\to\infty} u(x, z) = 0$ holds for all $0 \le z \le z_{\max}$.

3. Determine the solution $u(x)$ on the bounded computational domain Ω that agrees exactly with the unbounded result, restricted to Ω.

The goal of our paper is thus to construct transparent boundary conditions for arbitrary higher order evolution equations of the type (3) that ensure the realization of the third point above. Such problems occur frequently in various physical contexts. For example, electromagnetic field propagation due to radiating sources in the troposphere is distinguished by the presence of three regions: (1) The homogeneous region of the troposphere, corresponding to our left exterior domain, (2) the interior, inhomogeneous atmospheric region, and (3) the ground. The boundary condition between regions (2) and (3) is assumed to be given as an impedance boundary condition, leading to a straightforward modification of the three stipulations listed above. Various techniques for the solution of such boundary problems have been previously developed. In one such method, proposed by Marcus [11], the elliptic Helmholtz equation is approximated by a “parabolic” Padé (2, 0) approximation, after which the solutions in regions (1) and (2) are computed by an integral equation approach and a finite difference method and the two solutions are linked at the boundary between the two regions. In an example in acoustics, however, Mayfield [10] analyzed the numerical stability of an analogous hybrid propagation algorithm and determined that even if the implicit finite-difference propagation method in the interior region is unconditionally stable, the overall method may become unstable.

Recently, a specialization of the problem defined in this Introduction, based on a Padé (2, 2) approximation to the Helmholtz operator, has been successfully analyzed [1]. While the solution presented in this reference is restricted to cases for which certain inverse Laplace transforms can be inverted analytically, such a step is absent from our present method. We extend the methods described above as follows:

- We allow an arbitrary high-order Padé approximation of the square-root operator on the entire transversal axes.
- We do not need to introduce a Green’s function represenation for the choosen Padé approximation.
- We do not need to compute an explicit inverse Laplace transform.
- Consequentely, we do not need an analytic pretreatment.
- We prove unconditional stability of the algorithm.

As a result, we are able to develop efficient numerical techniques even for high Padé orders. Since our numerical implementation relies on non-trivial data structures, explicit pseudo-codes will also be presented below. It should be noted that this presentation is restricted to uniform step sizes in the propagation direction, so that we can apply the simplified shift-operator technique introduced in [15]. Similar formulas can, however, be derived for non-equidistant step sizes using the more direct procedure of [14] or the algebraic approach in [13], but the analysis and the resulting formulas would be far more complicated.

2. WIDE-ANGLE EQUATIONS

We first summarize the standard wide-angle approximation to the Helmholtz equation (1) that will form the basis for our subsequent considerations. The first step in this analysis is to define a new field variable $\tilde{u}(x, z) := u(x, z)\exp(-ik_0 z)$. The reference wavenumber k_0 is chosen to effectively minimize the mean phase velocity of the wavevector components of $\tilde{u}(x, z)$ in the z-direction. The resulting spectrally shifted Helmholtz equation, where we

TABLE I
Padé Coefficients

	(2, 0)	(2, 2)	(4, 2)	(4, 4)
j	0 2	0 2	0 2 4	0 2 4
c'_j	1 1/2	1 3/4	1 1 1/8	1 5/4 5/16
c_j	1	1 1/4	1 1/2	1 3/4 1/16

drop the tilde on $u(x, z)$ for notational simplicity, is

$$\left(\partial_z^2 + 2ik_0\partial_z + \partial_x^2 + k^2 - k_0^2\right) u(x, z) = 0. \tag{4}$$

Formally factorizing the above equation in analogy with (2) yields the following one-way equation after the transformations $z := z \cdot k_0$ and $x := x \cdot k_0$:

$$\partial_z u = -i(1 - \sqrt{1 + X^2})u, \quad \text{with } X^2 := \frac{k^2 - k_0^2}{k_0^2} + \partial_x^2 . \tag{5}$$

Since $k = \text{const}$ in the exterior domain, rational approximations of the form

$$\sqrt{1 + X^2} \simeq \frac{c'_0 + c'_2 X^2 + \cdots + c'_{2m} X^{2m}}{c_0 + c_2 X^2 + \cdots + c_{2n} X^{2n}} \tag{6}$$

can be applied to the square-root operator. The most accurate of these is generally obtained for $m = n$ or for $m = n + 1$; see [16]. We will derive a general procedure for deriving transparent boundary conditions that can be applied to all such approximations in succeeding sections. However, to simplify our numerical algorithm, we will consider explicitly Padé-type approximations of order $(2m, 2n)$. This yields an interpolation error $O(X^{2m+2n+2})$ for $X \to 0$.

For the simple Padé approximation of (6), the coefficients c'_{2i} and c_{2j}, $i = 0, \ldots, m$, $j = 0, \ldots, n$, can be obtained either by an explicit factorization [3] or by the Newman procedure [8]. The latter technique, on which our computer codes are based, can be extended to wide-angle equations other than those based on Padé-approximations. Some of the resulting coefficients can be found in Table I.

3. LONGITUDINAL DISCRETIZATION

The implicit midpoint discretization of (5) results in

$$\frac{u_i(x) - u_{i-1}(x)}{\Delta z} = -i(1 - \sqrt{1 + X^2})\frac{u_i(x) + u_{i-1(x)}}{2} . \tag{7}$$

Here $u_i(x)$, $0 < i \leq n$, denotes $u(x, z_0 + i\Delta z)$, where z_0 is the initial value of the longitudinal distance and Δz is the propagation step length. Note that here the subscript i denotes the longitudinal step number. The superscript i denotes the ith integer power of the corresponding quantity, while algebraic factors of i refer to the imaginary unit.

From (7), we immediately obtain the discrete evolution equation

$$\left(1+\frac{i\Delta z}{2}(1-\sqrt{1+X^2})\right)u_i(x)=\left(1-\frac{i\Delta z}{2}(1-\sqrt{1+X^2})\right)u_{i-1}(x)\,.$$

After replacing the square-root operator by its rational approximant, we obtain an equation of the form

$$u_i(x)=\frac{P'(X^2)}{P(X^2)}u_{i-1}(x)\,. \tag{8}$$

Here $P(X^2)$ and $P'(X^2)$ are the following polynomials of degree m in the variable X^2:

$$\begin{aligned} P'(X^2) &= \left(1-i\frac{\Delta z}{2}\right)C(X^2)+i\frac{\Delta z}{2}C'(X^2)\\ P(X^2) &= \left(1+i\frac{\Delta z}{2}\right)C(X^2)-i\frac{\Delta z}{2}C'(X^2)\,. \end{aligned}$$

The quantities $C'(X^2)$ and $C(X^2)$ are themselves polynomials defined by $C'(X^2)=c'_0+c'_2X^2+\cdots+c'_{2m}X^{2m}$ and $C(X^2)=c_0+c_2X^2+\cdots+c_{2n}X^{2n}$; see (6) and Table I. Applying a complex root finder yields

$$P'(X^2)=c'\prod_{j=1}^{k}(1-a'_jX^2) \qquad \text{and} \qquad P(X^2)=c\prod_{j=1}^{k}(1-a_jX^2), \tag{9}$$

where c and c' are constants. Choosing $X^2=0$, it follows that $P'(0)/P(0)=c'/c$ and comparing with (7) shows $c'/c=1$. Finally, inserting $X^2:=(k^2-k_0^2)/k_0^2+\partial_x^2$ leads to the desired factorization.

4. DISCRETE EVOLUTION EQUATION

From the rational approximation (8) and the factorization (9), we obtain the longitudinally discretized form of the evolution equation,

$$u_i(x)=\left(\frac{1-a'_k\partial_x^2}{1-a_k\partial_x^2}\right)\cdots\left(\frac{1-a'_2\partial_x^2}{1-a_2\partial_x^2}\right)\left(\frac{1-a'_1\partial_x^2}{1-a_1\partial_x^2}\right)u_{i-1}(x)\,, \tag{10}$$

which is the exact counterpart of the continuous evolution equation (3). In terms of the intermediate functions $g_i^{(1)}(x),\ldots,g_i^{(k-1)}(x)$ given by

$$\begin{aligned} g_i^{(1)}(x) &= \left(\frac{1-a'_1\partial_x^2}{1-a_1\partial_x^2}\right)u_{i-1}(x)\\ g_i^{(2)}(x) &= \left(\frac{1-a'_2\partial_x^2}{1-a_2\partial_x^2}\right)g_i^{(1)}(x)\\ &\vdots\\ g_i^{(k-1)}(x) &= \left(\frac{1-a'_{k-1}\partial_x^2}{1-a_{k-1}\partial_x^2}\right)g_i^{(k-2)}(x)\\ u_i(x) &= \left(\frac{1-a'_k\partial_x^2}{1-a_k\partial_x^2}\right)g_i^{(k-1)}(x)\,, \end{aligned}$$

the factorized discrete evolution problem (10) of order $2k$ can be recast into the following system of k second-order differential equations:

$$\begin{aligned}
\left(1 - a_1\partial_x^2\right) g_i^{(1)}(x) - \left(1 - a_1'\partial_x^2\right) u_{i-1}(x) &= 0 \\
\left(1 - a_2\partial_x^2\right) g_i^{(2)}(x) - \left(1 - a_2'\partial_x^2\right) g_i^{(1)}(x) &= 0 \\
&\vdots \\
\left(1 - a_{k-1}\partial_x^2\right) g_i^{(k-1)}(x) - \left(1 - a_{k-1}'\partial_x^2\right) g_i^{(k-2)}(x) &= 0 \\
\left(1 - a_k\partial_x^2\right) u_i(x) - \left(1 - a_k'\partial_x^2\right) g_i^{(k-1)}(x) &= 0\,.
\end{aligned} \tag{11}$$

This sequence of equations may be rewritten conveniently in the matrix form

$$\left[\begin{pmatrix} 1 & & & & \\ \ddots & \ddots & & & \\ & -1 & 1 & & \\ & & -1 & 1 & \\ & & & \ddots & \ddots \end{pmatrix} + \begin{pmatrix} -a_1\partial_x^2 & & & & \\ \ddots & \ddots & & & \\ & a_k'\partial_x^2 & -a_k\partial_x^2 & & \\ & & a_1'\partial_x^2 & -a_1\partial_x^2 & \\ & & & \ddots & \ddots \end{pmatrix}\right] \begin{pmatrix} g_1^{(1)} \\ \vdots \\ u_1 \\ g_2^{(1)} \\ \vdots \end{pmatrix} = 0. \tag{12}$$

The unknown functions are then assembled into the following sequence:

$$G(x) = \left\{ g_1^{(1)}, \ldots, g_1^{(k-1)}, u_1, \ldots, \underbrace{g_i^{(1)}, \ldots, g_i^{(k-1)}, u_i}_{i\text{th macro step}}, \ldots \right\}.$$

Here the quantities distinguished by a subscript i are evaluated during the ith physical propagation step, while the $k-1$ intermediate partial steps that compose each physical step and are associated with different terms in the Padé approximation are distinguished by their superscripts.

Given an initial condition with respect to x, at some fixed point, say x_+, we can solve the system (12) equation by equation. Thus at step i the function $u_{i-1}(x)$ maps over $k-1$ intermediate functions $g_i^{(j)}(x)$ to the solution $u_i(x)$. Hence we consider a mapping

$$u_{i-1}(x) \mapsto \underbrace{g_i^{(1)}(x) \mapsto g_i^{(2)}(x) \mapsto \cdots \mapsto g_i^{(k-1)}(x) \mapsto u_i(x)}_{\mathbf{g}_i(x) := \left(g_i^{(1)}(x), g_i^{(2)}(x), \ldots, g_i^{(k-1)}(x), u_i(x)\right)} \tag{13}$$

$$u_{i-1}(x) \mapsto \mathbf{g}_i(x)\,, \tag{14}$$

which depends on the initial conditions (with respect to x). In the lowest order case, $k = 1$, the vector $\mathbf{g}_i(x)$ consists of the single function $u_i(x)$.

To write (11) or (12) as a simple matrix equation with finite dimensions, we introduce the discrete shift-operator s_k with the property that $s_k u_i(x) = u_{i-1}(x)$. This operator shifts our sequence by one physical step, corresponding to k terms in G. That is, if s is the shift operator which shifts the sequence in G by one place, as introduced in Definition A.2, we have $s_k = s^k$. Since in the following case the index shift occurs only in powers of s_k, we redefine $s := s_k$ to yield the more intuitive formulation denoted by $su_i(x) = u_{i-1}(x)$.

Introducing the notation $\dot{g}$ for $\partial_x g$ and eliminating $u_{i-1}(x)$ in (11) in terms of $u_i(x)$ then yields

$$\left(\mathbf{E}(s) + \mathbf{A}(s)\partial_x^2\right)\mathbf{g}_i(x) = 0, \quad \text{for all macro steps } i \geq 1, \tag{15}$$

which must be solved subject to the boundary conditions

$$\dot{\mathbf{g}}_{i,+} = \mathbf{B}_+(s)\mathbf{g}_{i,+} \qquad \text{and} \qquad \dot{\mathbf{g}}_{i,-} = \mathbf{B}_-(s)\mathbf{g}_{i,-}. \tag{16}$$

A comparison with the infinite-dimensional system (12) demonstrates that the $k \times k$ matrices $\mathbf{E}(s)$ and $\mathbf{A}(s)$ and the k-element vectors $\mathbf{g}_i(x)$, $\mathbf{g}_{i,\pm}$, and $\dot{\mathbf{g}}_{i,\pm}$ are given by

$$\mathbf{E}(s) = \begin{pmatrix} 1 & & & & -s \\ -1 & 1 & & & \\ & & \ddots & & \\ & & -1 & 1 & \\ & & & -1 & 1 \end{pmatrix}, \qquad \mathbf{A}(s) = \begin{pmatrix} -a_1 & & & & sa_1' \\ a_2' & -a_2 & & & \\ & & \ddots & & \\ & & a_{k-1}' & -a_{k-1} & \\ & & & a_k' & -a_k \end{pmatrix},$$

$$\mathbf{g}_i(x) = \begin{pmatrix} g_i^{(1)}(x) \\ g_i^{(2)}(x) \\ \vdots \\ g_i^{(k-1)}(x) \\ u_i(x) \end{pmatrix}, \qquad \mathbf{g}_{i,\pm} = \begin{pmatrix} g_i^{(1)} \\ g_i^{(2)} \\ \vdots \\ g_i^{(k-1)} \\ u_i \end{pmatrix}_{x=x_\pm}, \qquad \dot{\mathbf{g}}_{i,\pm} = \begin{pmatrix} \dot{g}_i^{(1)} \\ \dot{g}_i^{(2)} \\ \vdots \\ \dot{g}_i^{(k-1)} \\ \dot{u}_i \end{pmatrix}_{x=x_\pm}.$$

We refer to $\mathbf{E}(s)$ and $\mathbf{A}(s)$ as operators, as they depend explicitly on the shift operator. Further, they may be represented as $\mathbf{D}(s) = \sum_{j=0}^{i} \mathbf{D}_j s^j$, where the $k \times k$ matrices $\mathbf{D}_j,\ j = 0, \ldots, i$, are complex. We will label operators of this general type as convolution operators or convolution matrices; the properties of such matrices that are required to derive our subsequent algorithms are collected in the Appendix. The Dirichlet-to-Neumann operators $\mathbf{B}_\pm(s)$ that implement the desired boundary conditions must be constructed such that the asymptotic boundary condition $\lim_{|x|\to\infty} u_i(x) = 0$ is fulfilled for all propagation steps. We will show later by construction that these boundary operators are also of convolution type.

5. DISCRETE TRANSPARENT BOUNDARY CONDITIONS

We now present a derivation of transparent boundary conditions for general wide-angle methods. To simplify the discussion, we consider $\mathbf{g}_i(x)$ only in the right exterior domain, $x \geq x_+$, and further shift the position of the right boundary x_+ to the origin according to $x \mapsto x + x_0$ and omit the ± subscript. Also, we designate the boundary value $\mathbf{g}_{i,+}$ by $\mathbf{g}_{i,0}$. All our results of course apply equally to the left exterior domain.

Consider (15) as an initial value problem in the right exterior domain with data given on the shifted boundary $x_+ = 0$. Our objective is to construct an operator with the property that the corresponding exterior solution decays asymptotically for any given Dirichlet data $\mathbf{g}_{i,0}$. To do this, we construct the Laplace transform $\widehat{\mathbf{g}}_i(p) := \int_0^\infty \exp(-px)\,\mathbf{g}_i(x)\,dx$, $p \in \mathcal{C}$

with $\Re(p) >$ const, of the exterior solution vector $\mathbf{g}_i(x)$. In the exterior domain, the Laplace transform of the equation system (15) is

$$(\mathbf{E}(s) + p^2\mathbf{A}(s))\widehat{\mathbf{g}_i}(p) = \mathbf{A}(s)(p\mathbf{g}_{i,0} + \dot{\mathbf{g}}_{i,0}).$$

The operator $\mathbf{A}(s)$ is invertible, since $a_j \neq 0,\ j = 1, \dots, k$ (cf. Section 3 and Proposition A.1). Therefore we can write equivalently

$$(p^2\mathbf{I} - \mathbf{C}^2(s))\widehat{\mathbf{g}_i}(p) = p\mathbf{g}_{i,0} + \dot{\mathbf{g}}_{i,0}\,,$$

where the $k \times k$ matrix $\mathbf{C}^2(s) := -\mathbf{A}(s)^{-1}\mathbf{E}(s)$ is of convolution type and $\mathbf{I}$ is the $k \times k$ identity matrix. If we assume for the moment that we can obtain the square roots, $\pm\mathbf{C}(s)$, of the matrix $\mathbf{C}^2(s)$, we have $(p^2\mathbf{I} - \mathbf{C}^2(s)) = (p\mathbf{I} + \mathbf{C}(s))\,(p\mathbf{I} - \mathbf{C}(s))$, since $\mathbf{C}(s)$ and $\mathbf{I}$ commute. Thus imposing the ansatz $\dot{\mathbf{g}}_{i,0} = \mathbf{B}\mathbf{g}_{i,0}$, see (15), and the boundary operator $\mathbf{B}(s) = -\mathbf{C}(s)$ leads to a special solution of the matrix equation, namely

$$\widehat{\mathbf{g}_i}(p) = (p\mathbf{I} + \mathbf{C}(s))^{-1}\,\mathbf{g}_{i,0} \quad \text{subject to boundary conditions } \dot{\mathbf{g}}_{i,0} = \mathbf{B}(s)\mathbf{g}_{i,0}. \tag{17}$$

To derive the nonlocal boundary conditions, which is equivalent to finding the boundary operator $\mathbf{B}(s)$, we apply the same procedure as in [14] or [15]. That is, we construct the matrix $\mathbf{C}(s)$ such that all poles $p_j,\ j = 1, \dots, k$, of $(p\mathbf{I} + \mathbf{C}(s))^{-1}$ are located in the right half of the complex plane, i.e., $\Re p_j > 0,\ j = 1, \dots, k$. This ensures that the exterior solution decays appropriately as $x \to \pm\infty$. The operator $(p\mathbf{I} + \mathbf{C}(s))^{-1}$ possesses a pole in p if the matrix $(p\mathbf{I} + \mathbf{C}(s)|_{s=0})^{-1}$ possesses a pole in p (cf. the properties of operators of convolution type given in the Appendix.)

The square root of $\mathbf{C}^2(s) := -\mathbf{A}^{-1}(s)\mathbf{E}(s)$ is obtained by decomposing the matrices $\mathbf{A}(s)$ and $\mathbf{E}(s)$ into its components with respect to s. The first of these is independent of the shift operator s, namely, $\mathbf{A}_0 := \mathbf{A}|_{s=0}$ and $\mathbf{E}_0 := \mathbf{E}|_{s=0}$, while the second is s dependent. We thus have $\mathbf{A} = \mathbf{A}_0 + s\mathbf{A}_1$ and $\mathbf{E} = \mathbf{E}_0 + s\mathbf{E}_1$ with

$$\mathbf{A}_1 := \begin{pmatrix} 0 & \cdots & 0 & a_1' \\ & & & 0 \\ & \mathbf{0} & & \vdots \\ & & & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{E}_1 := \begin{pmatrix} 0 & \cdots & 0 & -1 \\ & & & 0 \\ & \mathbf{0} & & \vdots \\ & & & 0 \end{pmatrix}.$$

For convenience, we define new matrices:

$$\tilde{\mathbf{A}}_1 := \mathbf{A}_0^{-1}\mathbf{A}_1, \qquad \tilde{\mathbf{E}}_0 := \mathbf{A}_0^{-1}\mathbf{E}_0, \qquad \tilde{\mathbf{E}}_1 := \mathbf{A}_0^{-1}\mathbf{E}_1\,.$$

From the previous definitions together with the ansatz $\mathbf{C}(s) = \sum_{j=0}^{i-1} \mathbf{C}_j s^j$, where $\mathbf{C}_j \in \mathcal{C}^{k\times k},\ 0 \leq j \leq i$, we observe that we must find matrices $\mathbf{C}_j$ such that

$$(\mathbf{I} + \tilde{\mathbf{A}}_1 s)(\mathbf{C}_0 + \mathbf{C}_1 s + \mathbf{C}_2 s^2 + \cdots)(\mathbf{C}_0 + \mathbf{C}_1 s + \mathbf{C}_2 s^2 + \cdots) = -\tilde{\mathbf{E}}_0 - \tilde{\mathbf{E}}_1 s. \tag{18}$$

It is a property of matrices of convolution type that the expression (18) can be computed elementwise and further ordered with respect to equal powers of s such that the resulting operator is of convolution type. That is, we are allowed to reorder, for example, $\mathbf{C}_1 s\mathbf{C}_0$ to $\mathbf{C}_1\mathbf{C}_0 s$. (This property, of course, is not generally true.) Now, as is evident by comparing coefficients, to solve (18) we must first find $\mathbf{C}_0$ such that $\mathbf{C}_0^2 = -\tilde{\mathbf{E}}_0$. Because the matrix $\tilde{\mathbf{E}}_0$ is a quotient of two lower triangular matrices, the matrix $\mathbf{C}_0$ is also lower triangular. Further, the diagonal entries $\mathbf{C}_0$ can be chosen such that $\Re(\mathbf{C})_{jj} > 0,\ j = 1, \ldots, k$, using the Algorithm 1. Hence all poles corresponding to $\mathbf{C}_0$, with $\mathbf{C}_0$ constructed according to Algorithm 1, are located in the right half of the complex plane.

ALGORITHM 1. Calculate $\mathbf{C_0} = \sqrt{-\tilde{\mathbf{E}}_0}$.

```
for $i = 1$ to $k$ do
  $c_{ii} = \sqrt{-\tilde{\mathbf{E}}_{0,ii}}$
  if $i > 1$ then
    for $j = i - 1$ to 1 do
      $c_{ij} = (-\tilde{\mathbf{E}}_{0,ij} - \sum_{m=j+1}^{i-1} c_{im} c_{mj})/(c_{ii} + c_{jj})$
    end for
  end if
end for
```

Remark. Since $\mathbf{C}_0 = \mathbf{C}(s)$ at $s = 0$, the condition $\mathbf{B} = -\mathbf{C}_0$ is the desired boundary condition for the first step; that is, for the first $k - 1$ intermediate solutions, $g_1^{(1)}(x), \ldots, g_1^{(k-1)}(x)$, and also for the solution $u_1(x)$ after the first step. By construction, all eigenvalues of the square root have a positive real part. Therefore both the intermediate solutions and $u_1(x)$ decay asymptotically in the external domain.

Subsequently the sequence $\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_{n-1}$ for the following n propagation steps is obtained by comparing coefficients of equal powers of s in (18). The corresponding pseudocode is given in Algorithm 2, and consists mainly of solutions of Sylvester equations. From the structure of the algorithm we observe that if $\mathbf{C}_0$ is computed, the entire sequence is uniquely determined.

ALGORITHM 2. Recursive calculation of $\mathbf{C}_j,\ j = 1, \ldots, n - 1$.

```
$\mathbf{Z} := -(\mathbf{E}_1 + \mathbf{A}_1 \mathbf{C}_0^2)$
Compute $\mathbf{C}_1$ from $\mathbf{C}_1 \mathbf{C}_0 + \mathbf{C}_0 \mathbf{C}_1 = \mathbf{Z}$
for $k = 2$ to $n - 1$ do
  $\mathbf{Z} \Leftarrow -\mathbf{A}_1 \mathbf{Z}$
  Compute $\mathbf{C}_k$ from $\mathbf{C}_k \mathbf{C}_0 + \mathbf{C}_0 \mathbf{C}_k = \mathbf{Z} - \sum_{j=1}^{k-1} \mathbf{C}_j \mathbf{C}_{k-j}$
end for
```

Finally, the nonlocal boundary condition at every step $0 < i \leq n$ is obtained from $\mathbf{C}_j$, $j = 0, \ldots, n - 1$, by employing the definitions $\mathbf{B} = -\mathbf{C}$ and

$$\begin{aligned} \dot{\mathbf{g}}_{i,0} &= \mathbf{B}(s)\mathbf{g}_{i,0} \\ &= (\mathbf{B}_0 + s\mathbf{B}_1 + \cdots + s^{i-1}\mathbf{B}_{i-1})\mathbf{g}_{i,0} \\ &= \mathbf{B}_0 \mathbf{g}_{i,0} + \mathbf{B}_1 \mathbf{g}_{i-1,0} + \cdots + \mathbf{B}_{i-1} \mathbf{g}_{1,0}. \end{aligned} \tag{19}$$

Equation (19) provides the algorithmic basis for constructing nonlocal boundary conditions for any wide-angle approximation and discrete propagation method, as different propagation methods can be distinguished simply through the values of the defining coefficients $a_1', \ldots, a_k'$ and $a_1, \ldots, a_k$. Because the operator $\mathbf{B}(s)$ possesses a Taylor representation in s, its action can be represented by matrix–vector multiplications of the Taylor coefficients $\mathbf{B}_j$ with boundary values describing the history of the evolution process. In our numerical implementation we order the boundary vectors $\mathbf{g}_{i,0}$ from smaller to larger step numbers and introduce a composite boundary vector $\mathbf{g}_0 = (\mathbf{g}_{1,0}^T, \dots \mathbf{g}_{i-1,0}^T, \mathbf{g}_{i,0}^T)^T$. Similarly, we generate the composite boundary matrices

$$\mathbf{B} = (\mathbf{B}_{i-1}, \mathbf{B}_{i-2}, \dots \mathbf{B}_0), \quad 1 \le i \le n \tag{20}$$

$$\mathbf{C} = -\mathbf{B}, \tag{21}$$

in place of the boundary operator $\mathbf{B}(s)$, after which the normal derivative $\dot{\mathbf{g}}_{i,0}$ from system (19) is computed through a matrix–vector multiplication. This procedure for implementing the discrete boundary condition in terms of a composite matrix $\mathbf{C}$ is summarized in Algorithm 5.

6. FINITE-ELEMENT DISCRETIZATION

From the representation (11), we will now generate a finite-element discretization on the interior domain. For illustrative purposes, consider the first equation of the system (11) at step i, $0 < i \le n$. Multiplying this equation by a trial function $v \in H^1(\Omega)$, $\Omega = (x_{-,}x_+)$, integrating over Ω, and integrating by parts yields

$$\begin{aligned}&\left(v, g_i^{(1)}\right) + \left(\partial_x v, a_1 \partial_x g_i^{(1)}\right) - \left(\bar{v} a_1 \partial_x g_i^{(1)}\right)\Big|_{x_-}^{x_+}\\ &\quad = (v, u_{i-1}) + (\partial_x v, a_1' \partial_x u_{i-1}) - (\bar{v} a_1' \partial_x u_{i-1})|_{x_-}^{x_+}.\end{aligned} \tag{22}$$

The variational problem corresponding to this equation is therefore to find a function $g_i^{(1)} \in H^1(\Omega)$ such that (22) holds for any $v \in H^1(\Omega)$. The other equations of the system (11) can be reformulated similarly. The resulting system is then discretized by replacing the infinite-dimensional function space $H^1(\Omega)$ by a finite-dimensional space $V_h = \operatorname{span}\{v_1, v_2, \dots, v_N\}$ with $V_h \subset H^1(\Omega)$. Hence the corresponding discrete problem is to determine a discrete approximation $g_{h,i}^{(1)}$ of $g_i^{(1)}$ with $g_{h,i}^{(1)} \in V_h$ such that for all $v_h \in V_h$

$$\begin{aligned}&\left(v_h, g_{h,i}^{(1)}\right) + \left(\partial_x v_h, a_1 \partial_x g_{h,i}^{(1)}\right) - \left(\bar{v}_h a_1 \partial_x g_{h,i}^{(1)}\right)\Big|_{x_-}^{x_+}\\ &\quad = (v_h, u_{h,i-1}) + (\partial_x v_h, a_1' \partial_x u_{h,i-1}) - (\bar{v}_h a_1' \partial_x u_{h,i-1})|_{x_-}^{x_+}.\end{aligned} \tag{23}$$

To solve the above system, we employ standard linear C^0-elements. Let v_1 and v_N denote the leftmost and the rightmost finite elements, respectively. A compact notation for (23) results if we set $v_1|_{x_-} = 1$, $v_N|_{x_+} = 1$, define the vectors $\mathbf{b}_i^{(1)}, \mathbf{b}_i'^{(1)} \in \mathcal{C}^N$ with $N = \dim V_h$ by

$$\mathbf{b}_i^{(1)} = \begin{pmatrix} \left(-a_1 \partial_x g_{h,i}^{(1)}\right)\big|_{x=x_-} \\ 0 \\ \vdots \\ 0 \\ \left(a_1 \partial_x g_{h,i}^{(1)}\right)\big|_{x=x_+} \end{pmatrix} \quad \text{and} \quad \mathbf{b}_i'^{(1)} = \begin{pmatrix} (-a_1' \partial_x u_{h,i-1})|_{x=x_-} \\ 0 \\ \vdots \\ 0 \\ (a_1' \partial_x u_{h,i-1})|_{x=x_+} \end{pmatrix},$$

and introduce the mass and stiffness matrices $\mathbf{M} \in \mathcal{R}^{N\times N}$, $\mathbf{A}_1, \mathbf{A}_1' \in \mathcal{C}^{N\times N}$ in standard fashion as $(\mathbf{M})_{i,j} = (v_{h,i}, v_{h,j})$ and $(A)_{i,j} = (\partial_x v_{h,i}, a_1 \partial_x v_{h,j})$. Defining as well the vectors $\mathbf{g}_i^{(1)} = (g_{h,i,1}^{(1)}, \dots, g_{h,i,N}^{(1)})^T \in \mathcal{C}^N$ and $\mathbf{u}_{i-1} = (u_{h,i-1,1}, \dots, u_{h,i-1,N})^T \in \mathcal{C}^N$, which are the discrete counterparts of the continuous functions $g_i^{(1)}(x)$ and $u_i(x)$, we have

$$(\mathbf{M} + \mathbf{A}_1)\,\mathbf{g}_i^{(1)} - \mathbf{b}_i^{(1)} = (\mathbf{M} + \mathbf{A}_1')\mathbf{u}_{i-1} - \mathbf{b}_i'^{(1)}. \tag{24}$$

If we know the solution $\mathbf{u}_{i-1}$ in the interior domain together with its normal derivative on the boundary and the normal derivative of $\mathbf{g}_i^{(1)}$, we can obtain the unknown intermediate vector $\mathbf{g}_i^{(1)}$. Repeating this procedure for each of the equations of (11), we generate the following block matrix equation in terms of the matrices and vectors introduced in the preceding paragraph:

$$\begin{pmatrix} \mathbf{M}+\mathbf{A}_1 & & & \\ & \mathbf{M}+\mathbf{A}_2 & & \\ & & \ddots & \\ & & & \mathbf{M}+\mathbf{A}_k \end{pmatrix} \begin{pmatrix} \mathbf{g}_i^{(1)} \\ \mathbf{g}_i^{(2)} \\ \vdots \\ \mathbf{u}_i \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(1)} \\ \mathbf{b}_i^{(2)} \\ \vdots \\ \mathbf{b}_i^{(k)} \end{pmatrix}$$
$$= \begin{pmatrix} \mathbf{M}+\mathbf{A}_1' & & & \\ & \mathbf{M}+\mathbf{A}_2' & & \\ & & \ddots & \\ & & & \mathbf{M}+\mathbf{A}_k' \end{pmatrix} \begin{pmatrix} \mathbf{u}_{i-1} \\ \mathbf{g}_i^{(1)} \\ \vdots \\ \mathbf{g}_i^{(k-1)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i'^{(1)} \\ \mathbf{b}_i'^{(2)} \\ \vdots \\ \mathbf{b}_i'^{(k)} \end{pmatrix}. \tag{25}$$

To solve the system (25), the vectors $\mathbf{b}_i^{(j)}$, $\mathbf{b}_i'^{(j)}$, $j = 1, \ldots, k$, must be constructed in accordance with the boundary conditions. The relationship between the discretized evolution equation (25) and the boundary condition (19) is determined by decomposing the boundary condition at each boundary according to $\mathbf{b}_i^{(j)} = \mathbf{b}_{i,-}^{(j)} + \mathbf{b}_{i,+}^{(j)}$ with

$$\mathbf{b}_{i,-}^{(j)} = \begin{pmatrix} \left(-a_j\partial_x g_{i,h}^{(j)}\right)\big|_{x=x_-} \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} \quad \text{and} \quad \mathbf{b}_{i,+}^{(j)} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \left(a_j\partial_x g_{i,h}^{(j)}\right)\big|_{x=x_+} \end{pmatrix}, \ \mathbf{b}_{i,\pm}^{(j)} \in \mathcal{C}^N.$$

Performing the same decomposition for each vector $\mathbf{b}_i'^{(j)}$ and assembling all nonzero entries of the vectors $\mathbf{b}_{i,\pm}^{(j)}$, $\mathbf{b}_{i,\pm}'^{(j)}$, $j = 1, \ldots, k$, into the four vectors $\mathbf{b}_{i,\pm}$, $\mathbf{b}_{i,\pm}' \in \mathcal{C}^k$, we arrive at

$$\mathbf{b}_{i,\pm} = \begin{pmatrix} \left(\pm a_1\partial_x g_{i,h}^{(1)}\right)\big|_{x=x_\pm} \\ \left(\pm a_2\partial_x g_{i,h}^{(2)}\right)\big|_{x=x_\pm} \\ \vdots \\ \left(\pm a_{k-1}\partial_x g_{i,h}^{(k-1)}\right)\big|_{x=x_\pm} \\ (\pm a_k\partial_x u_i)\big|_{x=x_\pm} \end{pmatrix} \quad \text{and} \quad \mathbf{b}_{i,\pm}' = \begin{pmatrix} \pm(a_1'\partial_x u_{i-1})\big|_{x=x_\pm} \\ \pm\left(a_2'\partial_x g_{i,h}^{(1)}\right)\big|_{x=x_\pm} \\ \vdots \\ \pm\left(a_{k-1}'\partial_x g_{i,h}^{(k-2)}\right)\big|_{x=x_\pm} \\ \pm\left(a_k'\partial_x g_{i,h}^{(k-1)}\right)\big|_{x=x_\pm} \end{pmatrix}. \tag{26}$$

We now derive an equation relating the vectors $\mathbf{b}_{i,\pm}$ and $\mathbf{b}'_{i,\pm}$ to the boundary condition (19). For $\mathbf{b}_{i-1,+}$, we have from (26) and (19)

$$\begin{aligned}\mathbf{b}_{i,+} &= \operatorname{diag}(a_1, \dots, a_k)\dot{\mathbf{g}}_{0,i}\\ &= \operatorname{diag}(a_1, \dots, a_k)\mathbf{B}(s)\mathbf{g}_{0,i}\\ &= \operatorname{diag}(a_1, \dots, a_k)(\mathbf{B}_0 + \mathbf{B}_1 s + \mathbf{B}_2 s^2 + \cdots + \mathbf{B}_{i-1}s^{i-1})\mathbf{g}_{0,i}\ .\end{aligned}$$

All of the expressions $\mathbf{B}_j s^j \mathbf{g}_n(0) = \mathbf{B}_j \mathbf{g}_{n-j}(0)$ above with $j = 1, \dots, n-1$ can be immediately evaluated based on the observation that the shift operator s decreases the index of $\mathbf{B}_j$ by unity. Further, since the matrix $\mathbf{B}_0$ is a lower triangular matrix, we can arrange the algorithm so that the boundary condition at the current step depends only on boundary values in the previous steps. To this end we decompose $\mathbf{b}_\pm$ as

$$\begin{aligned}\mathbf{b}_\pm \ &= \ \mathbf{B}_{\mathbf{d},\pm}\mathbf{g}_{0,i} + \mathbf{B}_{\mathbf{r},\pm}\mathbf{g}_i \quad \text{with}\\ \mathbf{B}_{\mathbf{d},\pm} &:= \pm\operatorname{diag}(a_1, \dots, a_k)|_{x=x_\pm}\ \operatorname{diag}(\mathbf{B}_{0,\pm})\end{aligned} \tag{27}$$

and

$$\begin{aligned}\mathbf{B}_{\mathbf{r},\pm}(s) :=\ & \pm\operatorname{diag}(a_1, \dots, a_k)|_{x=x_\pm}\\ & \cdot (\mathbf{B}_0 - \operatorname{diag}(\mathbf{B}_0) + s\mathbf{B}_1 + \cdots + s^{i-1}\mathbf{B}_{i-1})|_{x=x_\pm},\end{aligned} \tag{28}$$

where the matrices $\operatorname{diag}(\mathbf{B}_{0,\pm})$ contain only the main diagonals of $\mathbf{B}_{0,\pm}$. The diagonal matrices $\mathbf{B}_{\mathbf{d},\pm}$ are then inserted into the matrix of the final system which yields updated matrices $\mathbf{A}_j$ satisfying

$$\mathbf{A}_j = \mathbf{A}_j - \begin{pmatrix} \mathbf{B}_{\mathbf{d},-}(j,j) & & \\ & \mathbf{0} & \\ & & \mathbf{B}_{\mathbf{d},+}(j,j) \end{pmatrix}, \quad j = 1, \dots, k. \tag{29}$$

The reduced matrix $\mathbf{B}_{\mathbf{r},\pm}$, which is a lower triangular matrix with a zero diagonal, couples only previously determined boundary values and is therefore placed on the right-hand side of the evolution equation (cf. Algorithm 5.).

A corresponding expression for $\mathbf{b}'_+$ results from the observation that the vector $\mathbf{g}_i$ is ordered as $(g_i^{(1)}, g_i^{(2)}, \dots\ g_i^{(k-1)}, u_i)^T$, reflecting the algebraic structure of the boundary condition (19). The discrete evolution system (25), however, requires the alternate ordering (26) $(u_{i-1,} g_i^{(1)}, g_i^{(2)}, \dots\ g_i^{(k-1)})^T$. Thus rearranging the columns and rows of the operator $\mathbf{B}$ appropriately, we obtain a condition of the form

$$\begin{pmatrix} \partial u_{i-1} \\ \vdots \\ \partial g_i^{(k-1)} \end{pmatrix} = (\mathbf{B}'_0 + \mathbf{B}'_1 s + \cdots + \mathbf{B}1_{i-1}s^{i-1}) \begin{pmatrix} u_{i-1} \\ \vdots \\ g_i^{(k-1)} \end{pmatrix},$$

with the operator $\mathbf{B}'(s) := \mathbf{B}'_0 + \mathbf{B}'_1 s + \cdots + \mathbf{B}'_{i-1}s^{i-1}$.

In the first step of this transformation, which is illustrated in Fig. 1, we remove the last row and the last column of the composite boundary matrix $\mathbf{B} = -\mathbf{C}$ to obtain the

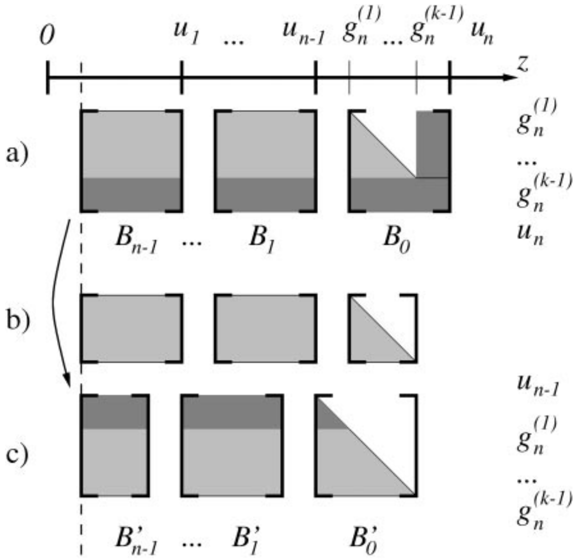
0
u_1 ... u_{n-1} $g_n^{(1)}$ ... $g_n^{(k-1)}$ u_n
z
a)
b)
c)
B_{n-1} ... B_1 B_0
B'_{n-1} ... B'_1 B'_0
$g_n^{(1)}$
...
$g_n^{(k-1)}$
u_n
u_{n-1}
$g_n^{(1)}$
...
$g_n^{(k-1)}$

FIG. 1. Construction of the boundary operator $\mathbf{B}'$ from the boundary operator $\mathbf{B}$.

reduced matrix shown in Fig. 1b. We then place the former last row of $\mathbf{B}$ at the top of the reduced matrix, after shifting the row to the left and adjusting it to the dimension of the reduced matrix, as illustrated in Fig. 1c. Finally, the resulting matrix is multiplied with the diagonal matrix $\mathrm{diag}(a'_1, \dots a'_k)$ according to (26). The pseudo-code for these operations is given in Algorithm 3.

Remark. To evolve the field over n propagation steps, we must first initialize the two boundary operators $\mathbf{B}_+$ and $\mathbf{B}_-$ with dimensions $k \times (kn)$, acting on the right and left boundary, respectively. At the end of the simulation we will possess two vectors $\mathbf{g}_+$ and $\mathbf{g}_-$ of dimensions $nk + 1$ that contain the required boundary values.

ALGORITHM 3. Computation of the operator $\mathbf{B}'$ (see Fig. 1).

```
1. Compute $\mathbf{B}' := -\mathbf{C}(1 : k-1, 1 : nk-1)$
2. Compute $\mathbf{b}' := -[\mathbf{C}(k, k+1 : nk), \underbrace{0, 0, \dots 0}_{k-1}]$
3. Compute $\mathbf{B}' \Leftarrow \mathrm{diag}(a'_1, \dots a'_k)\left[\begin{smallmatrix} 0 & \mathbf{b}' \\ \mathbf{0} & \mathbf{B}' \end{smallmatrix}\right]$
```

The initialization of the propagation algorithm, which includes the computation of the standard finite element matrices, the updating of these matrices (which corresponds to incorporating the boundary conditions), and the computation of the boundary matrices $\mathbf{B}_\pm$ are summarized in Algorithm 4:

ALGORITHM 4. Computation of the finite element matrices $\mathbf{A}_j$, $\mathbf{M}$ and the boundary matrices $\mathbf{B}_\pm$.

```
Compute $\mathbf{C}_{0,\pm}$                                        {acc. to Alg. 1}
Compute $\mathbf{C}_{j,\pm},\ j = 1 : n-1$                          {acc. to Alg. 2}
Compute $\mathbf{C}_\pm := [\mathbf{C}_{n-1}, \mathbf{C}_{n-2}, \dots \mathbf{C}_0]_\pm$   {acc. to Eq. (21)}
Compute $\mathbf{A}_j,\ j = 1, \dots, k$ and $\mathbf{M}$              {acc. to Eq. (23) and Eq. (24)}
Compute $\mathbf{B}'_\pm$                                           {acc. to Alg. 3}
Compute $\mathbf{B}_{\mathbf{d}}$                                   {acc. to Eq. (27)}
Update $\mathbf{A}_j,\ j = 1 : k$ using $\mathbf{B}_{\mathbf{d}}$              {acc. to Eq. (29)}
Compute $\mathbf{B}_{\pm} := -\mathrm{diag}(a_1, \ldots, a_k)_{\pm}[\mathbf{C}_{n-1},$
   $\mathbf{C}_{n-2}, \ldots\ \mathbf{C}_0 - \mathrm{diag}(\mathbf{C}_0)]_{\pm}$                {acc. to Eq. (28)}
$\mathbf{B}_{\pm} \Leftarrow [\mathbf{0}, \mathbf{B}_{\pm}(:, 1 : nk - 1)]$
$\mathbf{B}_{\pm} := [\mathbf{B}_{n-1}, \mathbf{B}_{n-2}, \ldots\ \mathbf{B}_0] \Leftarrow \mathbf{B}_{\pm} - \mathbf{B}'_{\pm}$
```

The structure and the numerical details of the resulting propagation algorithm is finally described in Algorithm 5.

ALGORITHM 5. Propagation algorithm.

```
$\mathbf{g} := \mathbf{u}_0$                                        {set the initial values}
$\mathbf{g}_- := (\mathbf{g}(1)),\ \mathbf{g}_+ := (\mathbf{g}(N))$                  {save the boundary values}
for $i = 1$ to $n$ do {propagate $n$ steps}
   for $j = 1$ to $k$ do {solve $k$ intermediate problems}
      $\mathbf{b} = (\mathbf{M} + \mathbf{A}'_j)\mathbf{g}$
      $c_- = \mathbf{B}_-(j, (n-i)k + 1 : (n-1)k + j)\mathbf{g}_-$
      $c_+ = \mathbf{B}_+(j, (n-i)k + 1 : (n-1)k + j)\mathbf{g}_+$
      $\mathbf{b} \Leftarrow \mathbf{b} + \begin{pmatrix} c_- \\ \mathbf{0} \\ c_+ \end{pmatrix}$
      Compute $\mathbf{g}$ from $(\mathbf{M} + \mathbf{A}_j)\mathbf{g} = \mathbf{b}$
      $\mathbf{g}_- \Leftarrow (\mathbf{g}_-,\ \mathbf{g}(1))^T, \mathbf{g}_+ \Leftarrow (\mathbf{g}_{+,}\ \mathbf{g}(N))^T$      {save the boundary values}
   end for
   $\mathbf{u}_i := \mathbf{g}$                                     {solution of the $i$th step}
end for
```

7. STABILITY

The stability of the wide-angle transparent boundary conditions can be verified by a natural extension of our earlier analysis for the Schrödinger-type Padé (2, 0) approximant [15]. Assume that the algorithm is implemented with exact arithmetic and consider the representation of the exterior solution given by (17), namely,

$$\widehat{\mathbf{g}}_i(p) = (p\mathbf{I} + \mathbf{C}(s))^{-1}\mathbf{g}_{i,0}, \quad \text{with } \mathbf{C}(s) = \mathbf{C}_0 + \mathbf{C}_1 s + \cdots + \mathbf{C}_{i-1}s^{i-1},$$

in which by construction (Algorithm 1) $\Re\,(\mathrm{diag}(\mathbf{C}_0)) > 0$. The operator $p\mathbf{I} + \mathbf{C}(s)$ is of convolution type. Its inverse exists and is again of convolution type $(p\mathbf{I} + \mathbf{C}(s))^{-1} = \mathbf{D}_0 + \mathbf{D}_1 s + \cdots$, as long as $\det(p\mathbf{I} + \mathbf{C}_0) \neq 0$. Further, it holds $\mathbf{C}_0^{-1} = \mathbf{D}_0$. Hence the zeros of $\det\,(p\mathbf{I} + \mathbf{C}_0)$ are exactly the poles of the solution vector $\widehat{\mathbf{g}}_i\,(p)$. Consequently, $\mathbf{g}_i\,(x)$ consists only of exponentially decaying functions.

We now prove that our boundary condition conserves the $\mathrm{L}^2(-\infty, \infty)$ norm over the infinite domain in which we employ the continuous form of the L^2 norm within the right and left exterior domains $\Omega_- = (-\infty, x_-)$ and $\Omega_+ = (x_+, \infty)$, respectively, and the discrete L^2 norm on $\Omega = (x_-, x_+)$. The unconditional stability of the propagation algorithm follows directly from this conservation law.

We consider first the discrete variational equation (23), and abbreviate $g_{h,i}^{(1)}$ by g and $u_{h,i-1}$ by u and a_1, a_1' by a, a'. Accordingly, (23) reads

$$(v_h, g) + (\partial_x v_h, a\partial_x g) - (\overline{v}_h a\partial_x g)|_{x_-}^{x_+} = (v_h, u) + (\partial_x v_h, a'\partial_x u) - (\overline{v}_h a'\partial_x u)|_{x_-}^{x_+}.$$

We must compute the function g from its predecessor u. If we regard the special choice $v_h = g$ and $v_h = u$, we obtain the equation system

$$(g, g) + (\partial_x g, a\partial_x g) - (\bar{g} a \partial_x g)|_{x_-}^{x_+} = (g, u) + (\partial_x g, a'\partial_x u) - (\bar{g} a'\partial_x u)|_{x_-}^{x_+}$$
$$(u, g) + (\partial_x u, a\partial_x g) - (\bar{u} a \partial_x g)|_{x_-}^{x_+} = (u, u) + (\partial_x u, a'\partial_x u) - (\bar{u} a'\partial_x u)|_{x_-}^{x_+}.$$

We compute the conjugate complex of the second equation

$$(g, u) + (a\partial_x g, \partial_x u) - (u\overline{a\partial_x g})|_{x_-}^{x_+} = (u, u) + (a'\partial_x u, \partial_x u) - (u\overline{a'\partial_x u})|_{x_-}^{x_+}.$$

If a' and a are complex conjugates with $\Im(a) \neq 0$, which is the essential requirement that ensures stability, the sum of the last and the first equations of the above system yields

$$(g, g) + (\partial_x g, a\partial_x g) - (\bar{g} a\partial_x g + u\overline{a\partial_x g})|_{x_-}^{x_+}$$
$$= (u, u) + (a'\partial_x u, \partial_x u) - (\bar{g} a'\partial_x u + u\overline{a'\partial_x u})|_{x_-}^{x_+},$$

The same procedure applies to the two exterior domains. For the right exterior domain

$$(g, g)_+ + (\partial_x g, a\partial_x g)_+ - (\bar{g} a\partial_x g + u\overline{a\partial_x g})|_{x_+}^{\infty}$$
$$= (u, u)_+ + (a'\partial_x u, \partial_x u)_+ - (\bar{g} a'\partial_x u + u\overline{a'\partial_x u})|_{x_+}^{\infty},$$

with an analogous result for the left exterior domain. Summing all three contributions and noting that the terms at infinity vanish while the normal derivatives at $x_\pm$ cancel, we obtain

$$(g, g)_- + (g, g) + (g, g)_+ + (\partial_x g, a\partial_x g)_- + (\partial_x g, a\partial_x g) + (\partial_x g, a\partial_x g)_+$$
$$= (u, u)_- + (u, u) + (u, u)_+ + (a'\partial_x u, \partial_x u)_- + (a'\partial_x u, \partial_x u) + (a'\partial_x u, \partial_x u)_+.$$

The imaginary part of the above equation yields

$$\Im(a)[(\partial_x g, \partial_x g)_- + (\partial_x g, \partial_x g) + (\partial_x g, \partial_x g)_+]$$
$$= -\Im(a')[(\partial_x u, \partial_x u)_- + (\partial_x u, \partial_x u) + (\partial_x u, \partial_x u)_+].$$

Since a and a' are complex conjugates with $\Im(a) \neq 0$, we obtain the conservation law

$$(\partial_x g, \partial_x g)_- + (\partial_x g, \partial_x g) + (\partial_x g, \partial_x g)_+ = (\partial_x u, \partial_x u)_- + (\partial_x u, \partial_x u) + (\partial_x u, \partial_x u)_+$$

from which we conclude that

$$(g, g)_- + (g, g) + (g, g)_+ = (u, u)_- + (u, u) + (u, u)_+.$$

Since the global L^2 norm is conserved for every intermediate step, it is conserved over the entire discrete evolution. We have proven that similar considerations apply to the energy norm. Since the initial data u_0 is required to be compactly supported on the interior domain, we have additionally that

$$\|g_i\|_{L^2, i\geq 0} \leq \|u_0\|_{L^2},$$

in which the norm is restricted to the interior domain, establishing the uniqueness of the discrete solution.

8. NUMERICAL EXPERIMENTS

We now verify our theoretical considerations by computing the reflection from the computational window boundary of a Gaussian input beam, propagating in air, described by

$$u(x,0) = u_0(x) = \text{const}\exp(-(x/10)^2)\exp(ik_0x\sin\phi).$$

We set $k(x) = k_0 = 2\pi/\lambda$, where the free space wavelength $\lambda = 1.55\ \mu$m, the propagation step size $\Delta z = 0.4\ \mu$m, and $\phi = \pi/4$. In our calculations which are meant to duplicate the corresponding numerical experiments in [6], the computational domains are either $\Omega = (-50, 50) \times (0, 400)\ \mu\text{m}^2$ or $\Omega = (-50, 50) \times (0, 100)\ \mu\text{m}^2$, while the transverse step sizes Δx vary between $0.01\ \mu\text{m} \leq \Delta x \leq 0.2\ \mu$m. We consider first the intrinsic error associated with application of the implicit midpoint rule to plane wave solutions of the Helmholtz equation. Recall that for the exact one-way Helmholtz propagator

$$u(\Delta z) = u(0)\exp(-i\Delta zk_0\sqrt{1-\sin^2\phi}),$$

for which the implicit midpoint rule yields

$$u_{\text{IMR}}(\Delta z) = u_{\text{IMR}}(0)\frac{1 - i\Delta zk_0/2(1-\sqrt{1-\sin^2\phi})}{1 + i\Delta zk_0/2(1-\sqrt{1-\sin^2\phi})}.$$

In this expression ϕ, $-\pi/2 < \phi < \pi/2$, is the angle between the propagation direction and the z-axis. This yields a resulting phase error $\log(u(\Delta z)) - \log(u_{\text{IMR}}(\Delta z))$ which we compare in Fig. 2 with that obtained by applying the Padé (2, 0) (Schrödinger type) approximation to the exact propagator. While at Padé order 2 the phase error of the Padé approximation is far greater, the opposite is true for the Padé (8, 8) approximant, as evident from Fig. 3.

Next we display in Fig. 4 the spectral norms of the matrices $\mathbf{B}_i$ as a function of the number of propagation steps for the boundary conditions associated with both the (2, 0) and (8, 8) Padé approximants. Here the matrices $\mathbf{B}_i$ are defined and computed as in Algorithm 4. Both approximations decay asymptotically as $\|\mathbf{B}_i\|_2 = \text{const}\, i^{-3/2}$, independent of the order of the approximation. Note that every second coefficient of the Padé (2, 0) approximation vanishes.

We now propagate the field from $z = 0$ to $z = 400\ \mu$m with the (8, 8) Padé procedure. Here the transverse grid spacing is $\Delta x = 0.2\ \mu$m while the computational domain is $\Omega = (-50, 50) \times (0, 400)\ \mu\text{m}^2$. The contour lines for the logarithmic amplitude over the first $100\ \mu$m of propagation are shown in Fig. 5. While the incident field propagates as expected along the $\theta = \pi/4$ direction, residual reflections are generated by the finite transverse discretization error.

To underline the wide-angle property of the Padé (8, 8) approximant, we first note that employing the (2, 0) in place of the (8, 8) Padé approximation for the square root operator leads to considerable phase errors, as evident from Fig. 6.

We perform a numerical experiment in the presence of a second Gaussian beam that describes an angle of $+\pi/4$ with respect to the z-axis. Figure 7 demonstrates the wide-angle nature of the underlying propagation method.

The influence of the discretization error with respect to the transverse step length Δx is evident if we repeat our previous (8, 8) Padé simulation (Fig. 5 with $\Delta x = 0.01\ \mu$m, cf. Fig. 8). The boundary reflection, which vanishes in the $\Delta x \to 0$ limit is indeed significantly reduced. The actual dependence of the reflection from the discretization error can be deduced

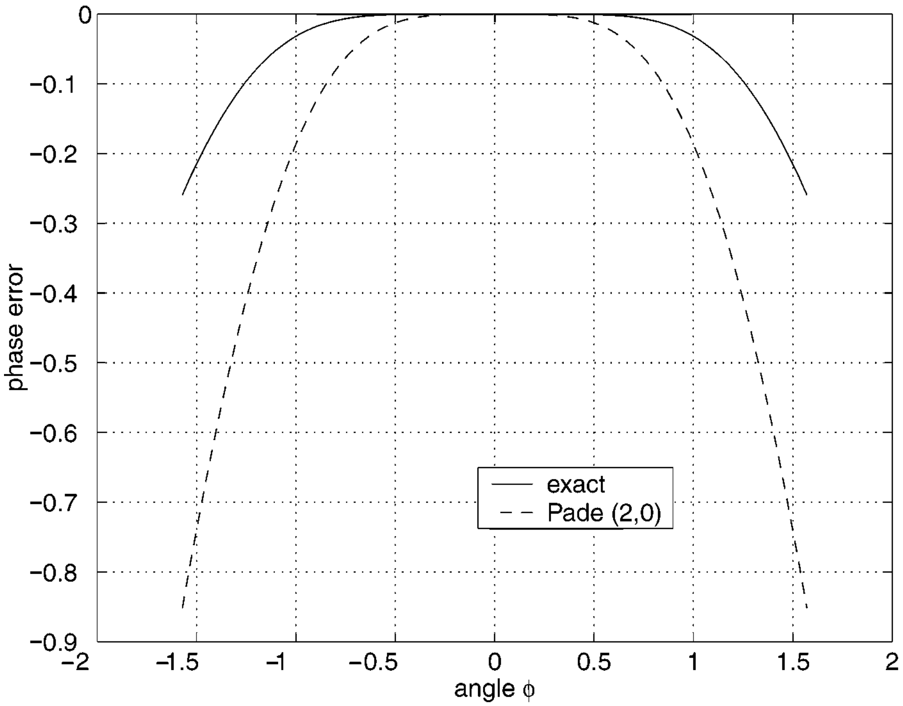

FIG. 2. The phase error associated with the exact implicit midpoint discretization (solid line) compared to that of the corresponding Padé (2, 0) approximant (dashed line).

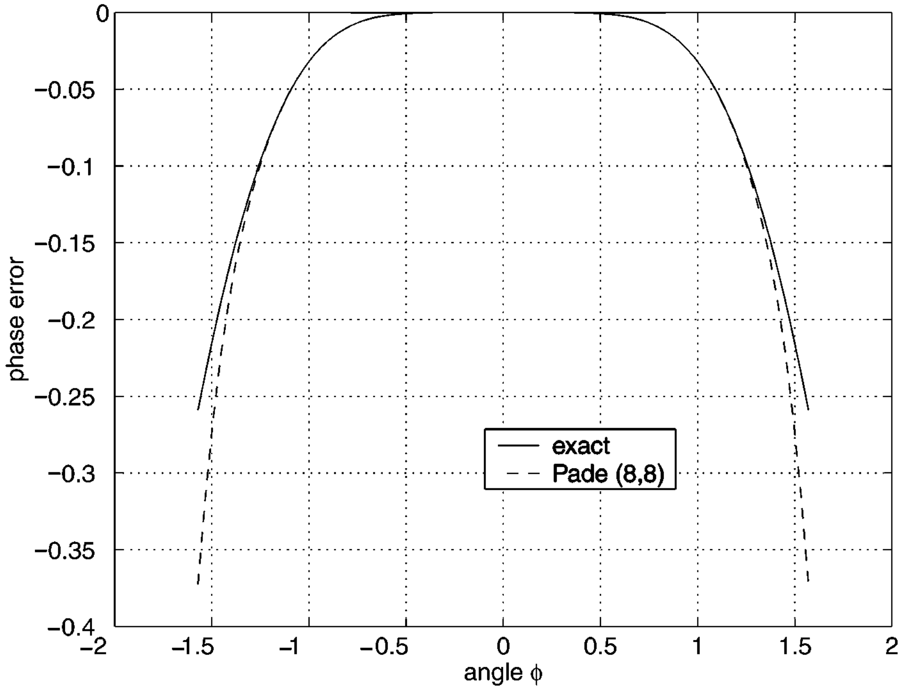

FIG. 3. As in Fig. 2, but for a (8, 8) Padé approximant.

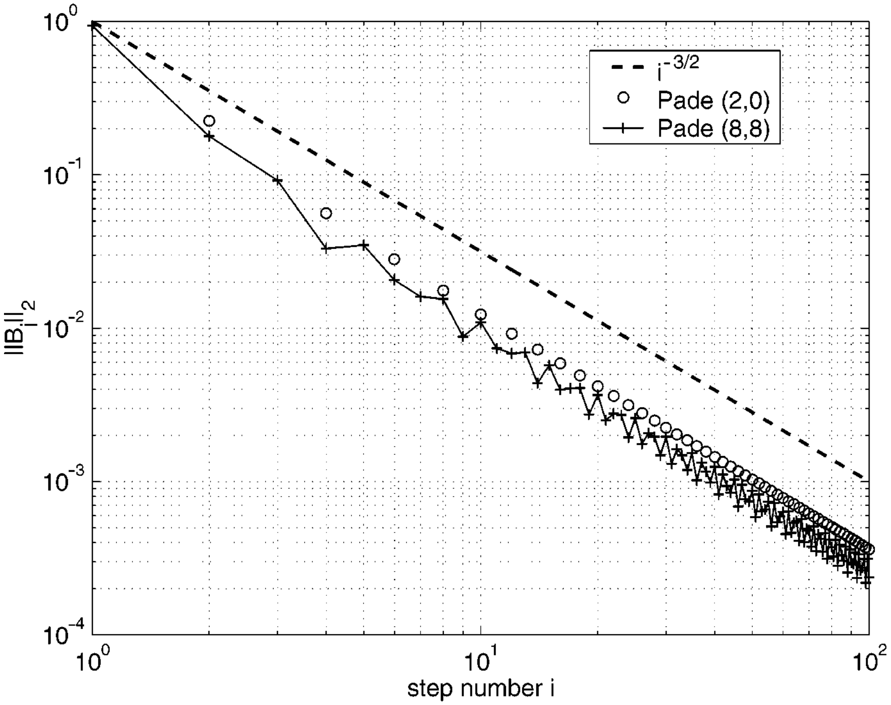

FIG. 4. Spectral norm of the boundary matrices $\mathbf{B}_i$ as a function of the number of propagation steps.

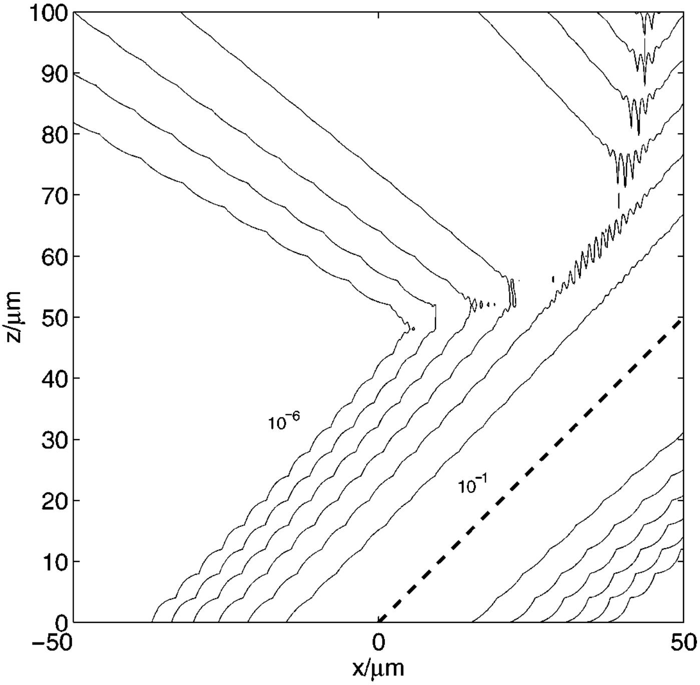

FIG. 5. Gaussian beam propagation calculated with a (8, 8) Padé propagator and $\Delta x = 0.2$ μm. The dashed line represents the exact propagation angle $\theta = \pi/4$. The reflected field vanishes as $\Delta x \to 0$.

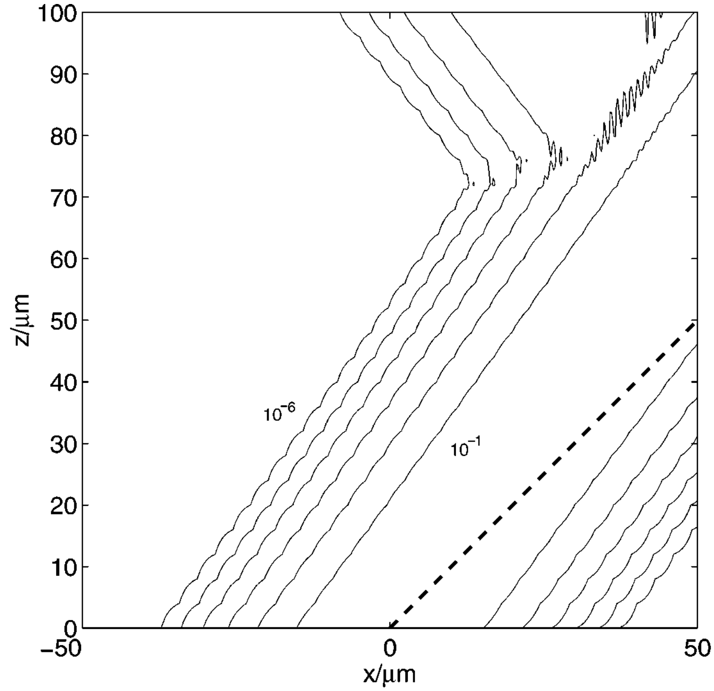

FIG. 6. As in Fig. 5 except for a Padé (2, 0) propagator. Note the error in the propagation angle.

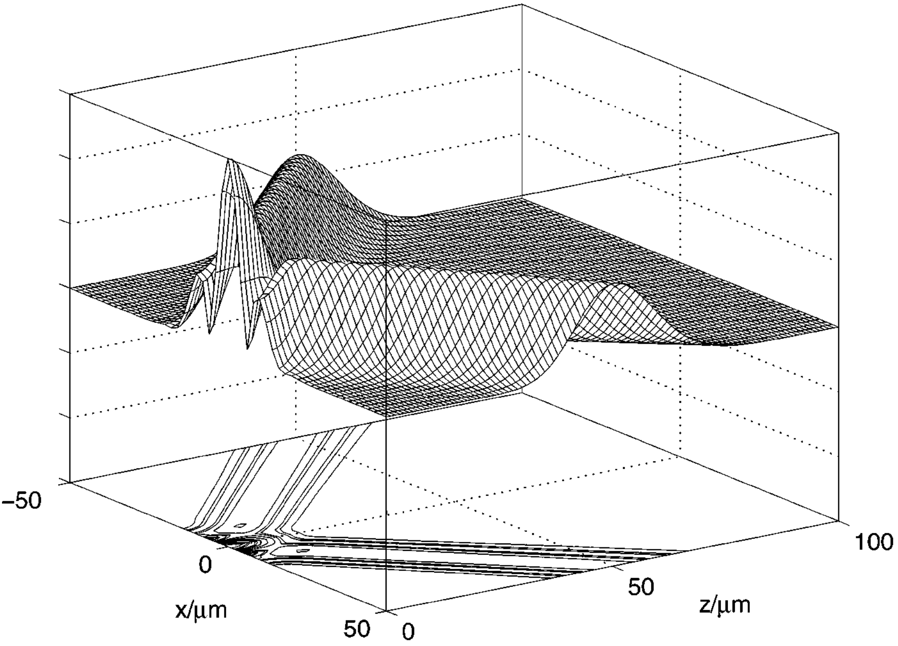

FIG. 7. Propagation of two Gaussian beams at a relative angle of $\pi/2$.

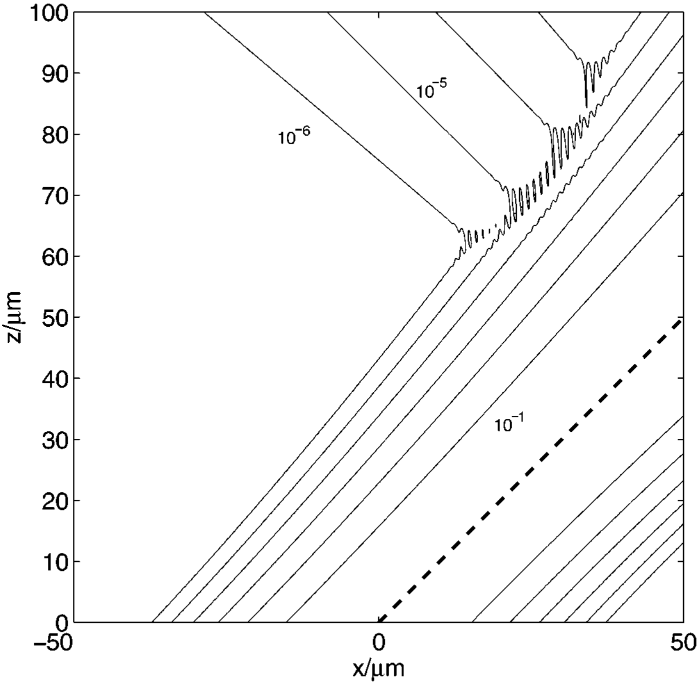

FIG. 8. As in Fig. 5, but with $\Delta x = 0.01\ \mu$m.

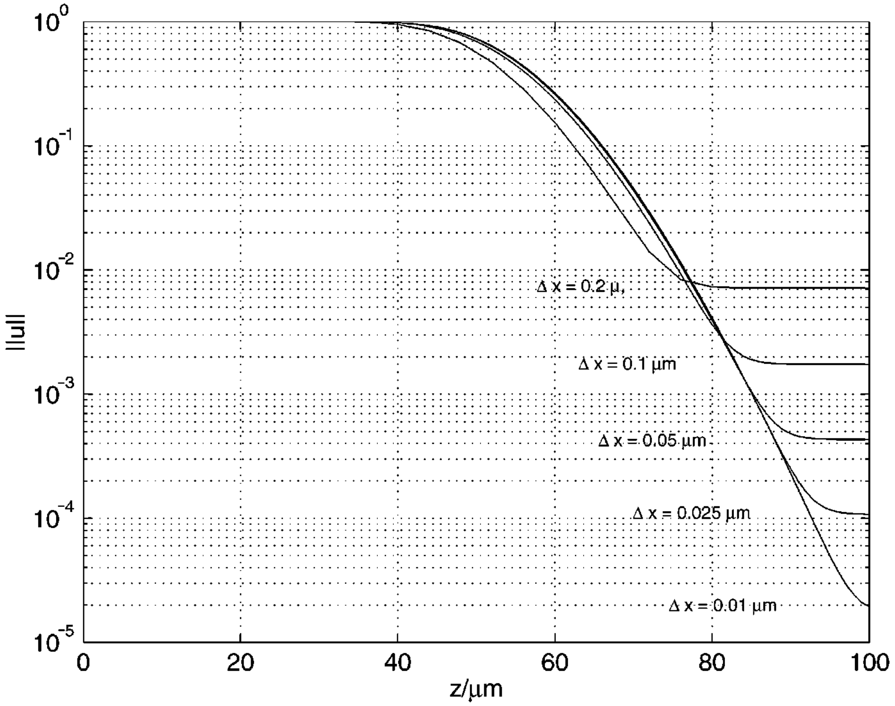

FIG. 9. The $L^2(-50, 50)$ norm $\|u\|$ as a function of the propagation distance for varying step sizes Δx.

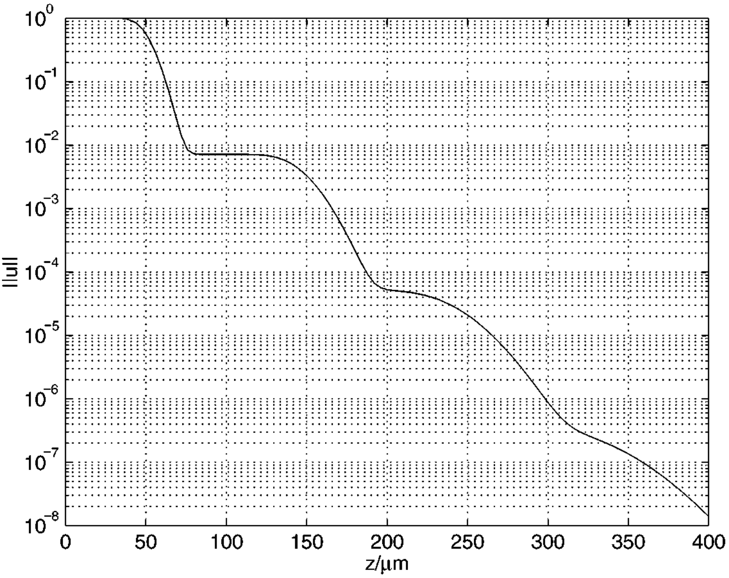

FIG. 10. The discrete L^2-norm of the field within the computational window as a function of longitudinal distance for $0 \leq z \leq 400$ μm for the parameters of Fig. 5.

from Fig. 9 which graphs the discrete $L_2(-50, 50)$ norm as a function of propagation distance for transverse step sizes of 0.2, 0.1, 0.05, 0.025, and 0.01 μm. The figure clearly shows that halving the transverse grid point spacing reduces the norm of the reflected field by a factor of 4. This behavior is entirely consistent with the $O(\Delta x^2)$ discretization error of the underlying linear finite elements. Finally, to demonstrate the stability of our algorithm (subject to arithmetic error), we display in Fig. 10, $\|u\|$ computed over a longer longitudinal interval $0 \leq z \leq 400$ μm for an (8, 8) Padé propagator. The parameters are the same as in Fig. 5. Clearly the resulting curve, which displays successive plateaus corresponding to integer number of reflections of the Gaussian beam from the computational window boundaries is completely free of numerical divergences.

CONCLUSIONS

We have presented the theoretical and algorithmic details required to derive and implement transparent boundary conditions for arbitrary rational approximations of the one-way Helmholtz equation in two dimensions. Our approach directly generalizes our earlier work on Schrödinger-type equations. Additionally, we have proven the unconditional stability of propagation methods based on our technique that in turn are based on longitudinal discretization with the implicit midpoint rule. The proof requires only that the rational approximations to the square-root operator obey the condition $a_j = \overline{a}'_j,\ j = 1, \ldots, k$, and that the finite-element space V_h of the interior discrete problem is nonadaptive; that is, it remains unchanged over the entire longitudinal propagation length. Given the generality of our results, we conclude that the associated family of transparent boundary conditions will find considerable application in a wide variety of numerical wave propagation problems.

APPENDIX: DISCRETE OPERATIONAL CALCULUS

Here we define an algebra that contains infinite sequences as elements in such a manner that the operations we need to derive the algorithms of this paper are well defined. Let U denote the infinite sequence of complex numbers $\{u_0, u_1, \ldots, u_j, \ldots\}$, $|u_j| < \infty$ for all $j \geq 0$, and $\mathcal{C}_s$ be the set of all such sequences. In particular, if u_j are functions, $u_j : \mathcal{C} \to \mathcal{C}$, of a complex argument p, we write $U(p) := \{u_0(p), u_1(p), \ldots, u_j(p), \ldots\}$.

DEFINITION A.1. Define the null and unity elements of $\mathcal{C}_s$ by

$$[0] = \{0, 0, \ldots, \} \qquad \text{and} \qquad [1] = \{1, 0, 0, \ldots, \}.$$

We write 0 and 1 in place of [0] and [1] if no ambiguity results. Addition and multiplication of two elements $U, V \in \mathcal{C}_s$ are then defined by

$$U + V = \{u_0 + v_0,\ u_1 + v_1,\ \ldots\} \qquad \text{and} \qquad UV = \{g_0, g_1, \ldots\} \ \text{ with } g_j = \sum_{i=0}^{j} u_{j-i} v_i.$$

From the above definitions, if $U, V \in \mathcal{C}_s$, then $U + V \in \mathcal{C}_s$ and $UV \in \mathcal{C}_s$. Further, $UV = VU$, $U(VW) = (UV)W$, and $U(V + W) = UV + UW$ so that $\mathcal{C}_s$ is a commutative ring. By induction, for $U, V \in \mathcal{C}_s$ the equation $UV = 0$ implies either $U = 0$ or $V = 0$ or both. Accordingly, the ring $\mathcal{C}_s$ can be extended to a commutative ring of fractions in the sense of Mikusiński's operational calculus. This step is however unnecessary for our present purpose.

We will label certain special elements of $\mathcal{C}_s$ with dedicated symbols. (1) The multiplication operator is $[a] := \{a,\ 0,\ 0, \ldots\}$, $a \in \mathcal{C}$. The operation $[a]U$ that multiplies each element of U by a, is for simplicity abbreviated aU.

DEFINITION A.2. (2) The shift operator s is defined by $s := \{0,\ 1,\ 0,\ 0, \ldots\}$.

By definition, $sU = \{0,\ u_0,\ u_1, \ldots, \}$, $ss = s^2 = \{0,\ 0,\ 1,\ 0,\ \ldots, \}, \ldots$

Two sequences $U, V \in \mathcal{C}_s$ are equal if their elements are equal, i.e., $u_i = v_i$ for all $i \geq 0$. An elementwise comparison of sequences will be denoted by $(sU)_{i+1} = (U)_i = u_i$ or by $su_{i+1} = u_i$, which means that $V := s^k U$ implies $v_j = 0$ for $j = 0, \ldots, k-1$. We also state that $U(p)$ has a complex pole λ if $\lim_{p \to \lambda} |u_j(p)| = \infty$ for some $j \geq 0$.

DEFINITION A.3. A linear mapping $\mathbf{A} : \mathcal{C}_s \to \mathcal{C}_s$ is defined through the infinite matrices $\mathbf{A} = (a_{ij})_{i,j=0,\ldots,\infty}$, $a_{ij} \in \mathcal{C}$ that map U into V, $U \mapsto V = \mathbf{A}U$ through a matrix–vector multiplication $V_i = \sum_{j=0}^{i} a_{ij} U_j$. We denote the set of all of such matrices by $\mathcal{L}$.

In our application, equations between sequences appear in the form

$$\left[\begin{pmatrix} \mathbf{A} & & & \\ & \mathbf{A} & & \\ & & \mathbf{A} & \\ & & & \ddots \end{pmatrix} + \begin{pmatrix} \mathbf{B} & & & \\ & \mathbf{B} & & \\ & & \mathbf{B} & \\ & & & \ddots \end{pmatrix} s^k \right] U = V\,, \tag{A.1}$$

where $\mathbf{A}$ and $\mathbf{B}$ are lower triangular, complex $k \times k$ matrices. Each row of this equation corresponds to one full propagation step (also termed a physical step) that consists of k intermediate or partial steps, where k is the order of the underlying Padé approximation.

Accordingly, we separate the sequences U and V into subsequences of lengths k and write these as column vectors. In other words, motivated by the matrix form of the underlying equations, we compose $U = \{\mathbf{u}_0^T, \mathbf{u}_1^T, \ldots\}$ and $V = \{\mathbf{v}_0^T, \mathbf{v}_1^T, \ldots\}$ with $\mathbf{u}_j, \mathbf{v}_j \in \mathcal{C}^k$. We can now replace (A.1) by the shorter form

$$(\mathbf{A} + \mathbf{B}s^k)\mathbf{u}_j = \mathbf{v}_j, \quad j \geq 0\,. \tag{A.2}$$

The second important matrix type, appearing in this paper as lower triangular blocks is matrices of *convolution* type

$$\begin{pmatrix} \mathbf{B}_0 & & & \\ \mathbf{B}_1 & \mathbf{B}_0 & & \\ \mathbf{B}_2 & \mathbf{B}_1 & \mathbf{B}_0 & \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}. \tag{A.3}$$

The terms $\mathbf{B}_j$, $j \geq 0$, are $k \times k$ complex matrices. In the special case $k = 1$ we assign the sequence $B = \{\mathbf{B}_0, \mathbf{B}_1, \mathbf{B}_2, \ldots\}$ to each matrix $\mathbf{B}$ of convolution type with $B \in \mathcal{C}_s$ such that $\mathbf{B}\,U = BU$ for all $U \in \mathcal{C}_s$. A typical mapping $U \mapsto V$ within our factorization procedure possesses the form

$$\left[\begin{pmatrix} \mathbf{A} & & & \\ & \mathbf{A} & & \\ & & \mathbf{A} & \\ & & & \ddots \end{pmatrix} + \begin{pmatrix} \mathbf{B}_0 & & & \\ \mathbf{B}_1 & \mathbf{B}_0 & & \\ \mathbf{B}_2 & \mathbf{B}_1 & \mathbf{B}_0 & \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}\right] U = V. \tag{A.4}$$

In the above equation, $\mathbf{A}$ and $\mathbf{B}_0$ are $k \times k$ lower triangular complex matrices and $\mathbf{B}_j$, $j \geq 1$, are general complex $k \times k$ matrices. In our notation, the jth row of the above equation for $(j \geq 0)$ reads

$$\left(\mathbf{A} + \sum_{i=0}^{j} \mathbf{B}_i s^{ik}\right) \mathbf{u}_j = \mathbf{v}_j, \quad j \geq 0\,, \tag{A.5}$$

which we further abbreviate as $(\mathbf{A} + \mathbf{B}(s))\,\mathbf{u}_j = \mathbf{v}_j$ with $\mathbf{B}(s) := \sum_{i=0}^{j} \mathbf{B}_i s^{ik}$. Equations of this type that are formulated in terms of the shift operator will accordingly be labled operator equations.

The inverse of the operator appearing in brackets on the left-hand side of (A.4) possesses a pole if $\det(\mathbf{A} - \mathbf{B}_0) = 0$. Since (A.4) is equivalent to the operator equation (A.5), the operator equation (A.2) possesses a pole if $\det(\mathbf{A} - \mathbf{B}_0) = 0$.

The following properties of matrices of convolution type are important to our subsequent derivations. They may be proved by direct computation.

Consider operators $\mathbf{B}(s)$ of convolution type with

$$\mathbf{B}(s) = \mathbf{B}_0 + \sum_{i=1}^{j} \mathbf{B}_i s^{ik}, \quad \mathbf{B}_i \in \mathcal{C}^{k \times k}\,. \tag{A.6}$$

Then

PROPOSITION A.1. (i) *The product of two infinite matrices of convolution type is also of convolution type.*

If additionally $\mathbf{B}_0$ *is invertible,*

(ii) *The inverse* $\mathbf{B}^{-1}(s)$ *of the operator* $\mathbf{B}(s)$ *exists and* $\mathbf{B}^{-1}(s) \in \mathcal{L}$.

(iii) $\mathbf{B}^{-1}(s)$ *is in convolution form.*

ACKNOWLEDGMENTS

Two of us (F.S and T.F.) thank the Department of Physics of the University of Waterloo for their hospitality during a part of our collaboration. We also thank Nortel Networks, the Natural Sciences and Engineering Research Council of Canada, and the Canadian Institute for Photonic Innovations for financial support.

REFERENCES

1. A. Arnold and M. Erhardt, Discrete transparent boundary conditions for wide angle parabolic equations in underwater acoustics, *J. Comput. Phys.* **145**(2), 611 (1998).
2. A. Bamberger, B. Engquist, L. Halpern, and P. Joly, Higher order paraxial wave equation approximations in heterogeneous media, *SIAM J. Appl. Math.* **48**(1), 129 (1988).
3. A. Bamberger, B. Engquist, L. Halpern, and P. Joly, Parabolic wave equation approximations in heterogeneous media, *SIAM J. Appl. Math.* **48**(1), 99 (1988).
4. M. D. Collins, Higher-order Padé approximations for accurate and stable elastic parabolic equations with application to interface wave propagation, *J. Acoust. Soc. Am.* **89**, 1050 (1991).
5. M. D. Collins, A split-step Padé solution for the parabolic equation method, *J. Acoust. Soc. Am.* **93**, 1736 (1993).
6. T. Friese, F. Schmidt, and D. Yevick, Transparent boundary conditions for wide angle one-way helmholtz equation, *J. Comp. Phys.* **165**, 645 (2000).
7. G. R. Hadley, Multistep method for wide-angle beam propagation, *Opt. Lett.* **17**, 1743 (1992).
8. L. Halpern and L. N. Trefethen, Wide-angle one-way wave equations, *J. Acoust. Soc. Am.* **84**(4), 1397 (1988).
9. W. P. Huang and C. L. Xu, A wide-angle vector beam propagation method, *Photon. Technol. Lett.* **4**, 1118 (1992).
10. B. Mayfield, *Nonlocal Boundary Conditions for the Schrödinger Equation*, Ph.D. thesis (University of Rhode Island, Providence, RI, 1989).
11. S. W. Marcus and A. Hybrid (Finite difference-surface Green's function) Method for computing transmission losses in an inhomogeneous atmosphere over irregular terrain, *IEEE Trans. Antennas Propagation* **40**(12), 1451 (1992).
12. J. Mikusiński, *Operatinal Calculus* (Pergamon Press, London, 1959).
13. F. Schmidt, Construction of discrete transparent boundary conditions for Schrödinger-type equations, *Surv. Math. Ind.* **9**, 87 (1999).
14. F. Schmidt and P. Deuflhard, Discrete transparent boundary conditions for the numerical solution of Fresnel's equation, *Comput. Math. Applic.* **29**, 53 (1995).
15. F. Schmidt and D. Yavick, Discrete transparent boundary conditions for Schrödinger-type equations, *J. Comput. Phys.* **134**, 96 (1997).
16. L. N. Trefethen and L. Halpern, Well-posedness of one-way wave equations and absorbing boundary conditions, *J. Acoust. Soc. Am.* **47**, 421 (1986).
17. D. Yevick, J. Yu, W. Bardyszewaki, and M. Glasner, Stability issues in vector electric field propagation, *Photon. Technol. Lett.* **7**, 656 (1995).