Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

FRANK SCHMIDT, DAVID YEVICK[1], TILMANN FRIESE[2]

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

[1]Department of Physics, University of Waterloo Waterloo, Ontario N2L 3G1, Canada
[2]Department of Physics, University of Waterloo Waterloo, Ontario N2L 3G1, Canada

Preprint SC 99-46 (December 1999)

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

Frank Schmidt, David Yevick, Tilmann Friese

Abstract: In this paper, we generalize the nonlocal discrete transparent boundary condition introduced by Schmidt and Deuflhard [Comp. Math. Appl. **29** (1995) 53-76] and Schmidt and Yevick [J. Comput. Phys. **134** (1997) 96-107] to propagation methods based on arbitrary Padé approximations to 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.

Keywords: 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 $\mathcal{R}^2$ 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. For 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 the as fix-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

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

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 has been extensively examined in e.g. [10] and [2]. Additionally, the non-commutativity of the factors in (3) in 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].

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

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 initial conditions $u(x,0) = u_0(x)$ compactly supported on 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 insure the realization of the third point above. We will restrict our derivation to uniform step-sizes in the z-direction of propagation in order to apply the simplified shift-operator technique introduced in [9]. Similar formulas could be derived for non-equidistant step sizes using the more direct procedure of [8] or the algebraic approach [7] but the analysis as well as the resulting formulas would be far more complicated. Recently, a specialization of the above problem, namely a Padé (2,2) approximation, 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. As a result, we are able to develop efficient numerical techniques for high Padé orders. Since our numerical implementation relies on non-trivial data structures, explicit pseudo codes will as well be presented below.

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 $\tilde{u}(x,z) := u(x,z)\exp(-ik_0 z)$ by introducing a suitable reference-wavenumber k_0 such that the mean phase velocity of the wavevector components of $\tilde{u}(x,z)$ in the z-direction is effectively minimized. The resulting spectrally shifted Helmholtz equation, where we 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 $x := x \cdot k_0$

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

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

$$\sqrt{1+X^2} \simeq \frac{c_0' + c_2' X^2 + \ldots + c_{2m}' X^{2m}}{c_0 + c_2 X^2 + \ldots + c_{2n} X^{2n}}\,. \tag{6}$$

Table 1: 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

can be applied to the square-root operator. The most accurate of these are generally obtained for $m = n$ or $m = n + 1$, see [10]. 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 only 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 $c'_{2i,}\, c_{2j,}\ \ i = 0 \dots m$, $j = 0 \dots n$ can be obtained through either an explicit factorization ([3]) or by the Newman-procedure [5]. The latter technique, upon 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 1.

3 Longitudinal Discretization

The implicit midpoint discretization of (5) results in

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

Here $u_i(x)$, $0 < i \le n$ denotes $u(x, z_0 + i\Delta z)$ with z_0 the initial value of the longitudinal distance and Δz the propagation step length. This yields the discrete evolution equation

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

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

$$u_i(x) = \frac{P'(X^2)}{P(X^2)} u_{i-1}(x)\,. \tag{7}$$

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

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

The quantities $C'(X^2)$ and $C(X^2)$ are themselves polynomials defined by $C'(X^2) = c'_0 + c'_2 X^2 + \ldots + c'_{2m} X^{2m}$ and $C(X^2) = c_0 + c_2 X^2 + \ldots + c_{2n} X^{2n}$, see (6) and Tab. (1). Applying a complex root finder yields

$$P'(X^2) = c' \prod_{j=1}^{k} \left(1 - a'_j X^2\right) \text{ and } P(X^2) = c \prod_{j=1}^{k} \left(1 - a_j X^2\right) , \tag{8}$$

with c and $c'$are constants. Finally, inserting $X^2 := \left(k^2 - k_0^2\right)/k_0^2 + \partial_x^2$ leads to the desired factorization.

4 Discrete Evolution Equation

From the rational approximation (7) and the factorization (8) we obtain the longitudinally discretized form of the evolution equation

$$u_i(x) = \left(\frac{1 - a'_k \partial_x^2}{1 - a_k \partial_x^2}\right) \cdots \left(\frac{1 - a'_2 \partial_x^2}{1 - a_2 \partial_x^2}\right) \left(\frac{1 - a'_1 \partial_x^2}{1 - a_1 \partial_x^2}\right) u_{i-1}(x) , \tag{9}$$

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, order $2k$ discrete evolution problem (9) can be recast into the following system of k second-order differential equations

$$\begin{aligned}
\left(1 - a_1 \partial_x^2\right) g_i^{(1)}(x) - \left(1 - a'_1 \partial_x^2\right) u_{i-1}(x) &= 0 \\
\left(1 - a_2 \partial_x^2\right) g_i^{(2)}(x) - \left(1 - a'_2 \partial_x^2\right) g_i^{(1)}(x) &= 0 \\
&\vdots \\
\left(1 - a_{k-1} \partial_x^2\right) g_i^{(k-1)}(x) - \left(1 - a'_{k-1} \partial_x^2\right) g_i^{(k-2)}(x) &= 0 \\
\left(1 - a_k \partial_x^2\right) u_i(x) - \left(1 - a'_k \partial_x^2\right) g_i^{(k-1)}(x) &= 0 .
\end{aligned} \tag{10}$$

To write (10) as a simple matrix equation, we introduce the discrete shift-operator s with the property that $su_i(x) = u_{i-1}(x)$. Introducing the notation $\dot{g}$ for $\partial_x g$ and eliminating $u_{i-1}(x)$ in (10) in terms of $u_i(x)$ yields

$$\left(\mathbf{E} + \mathbf{A}\partial_x^2\right) \mathbf{g}_i(x) = 0 \text{ with boundary conditions } \dot{\mathbf{g}}_{i,+} = \mathbf{B}_+ \mathbf{g}_{i,+},\ \dot{\mathbf{g}}_{i,-} = \mathbf{B}_- \mathbf{g}_{i,-}, \tag{11}$$

in which the $k \times k$-matrices $\mathbf{E}$ and $\mathbf{A}$ and the k-element vectors $\mathbf{g}_i(x)$, $\mathbf{g}_{i,\pm}$ and $\dot{\mathbf{g}}_{i,\pm}$ are

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

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

The Dirichlet-to-Neumann operators $\mathbf{B}_\pm$ implement the desired boundary conditions and must be constructed such that the asymptotic boundary condition $\lim_{|x|\to\infty} u_i(x) = 0$ is fulfilled for all propagation steps. Thus at step $i-1$ the function $u_{i-1}(x)$ maps over $k-1$ intermediate functions $g_i^{(j)}(x)$ to the solution $u_i(x)$. We can equivalently consider this procedure as a one-step propagation (mapping) from $u_{i-1}(x)$ to a vector of functions $\mathbf{g}_i(x)$, that is,

$$u_{i-1}(x) \quad \longmapsto \quad \underbrace{g_i^{(1)}(x) \mapsto g_i^{(2)}(x) \mapsto \ldots \mapsto g_i^{(k-1)}(x) \longmapsto u_i(x)}_{\mathbf{g}_i(x) := \left(g_i^{(1)}(x), g_i^{(2)}(x), \ldots g_i^{(k-1)}(x), u_i(x)\right)} \tag{12}$$

$$u_{i-1}(x) \quad \longmapsto \quad \mathbf{g}_i(x)\,. \tag{13}$$

In the lowest order case $k = 1$ the vector $\mathbf{g}_i(x)$ consists of the single function $u_i(x)$.

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 $\pm$-subscript. As well, 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 (11) 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 (11) is

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

By construction $\mathbf{A}$ is invertible, since $a_j \neq 0$, $j = 1, \ldots, k$, cf. Sec.3. Therefore we can write equivalently

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

where the $k \times k$-matrix $\mathbf{C}^2 := -\mathbf{A}^{-1}\mathbf{E}$ and $\mathbf{I}$ is the $k \times k$-identity-matrix. If we assume for the moment that we can obtain the square roots, $\pm\mathbf{C}$, of the matrix $\mathbf{C}^2$ we have $(p^2\mathbf{I} - \mathbf{C}^2) = (p\mathbf{I} + \mathbf{C})\,(p\mathbf{I} - \mathbf{C})$, since $\mathbf{C}$ and $\mathbf{I}$ commute. Thus imposing the ansatz $\dot{\mathbf{g}}_{i,0} = \mathbf{B}\mathbf{g}_{i,0}$, see (11), and imposing the boundary operator $\mathbf{B} = -\mathbf{C}$, leads to a special solution of the matrix equation, namely

$$\widehat{\mathbf{g}}_i(p) = (p\mathbf{I} + \mathbf{C})^{-1}\, \mathbf{g}_{i,0} \quad \text{subject to boundary conditions } \dot{\mathbf{g}}_{i,0} = \mathbf{B}\mathbf{g}_{i,0}. \tag{14}$$

To derive the nonlocal boundary conditions, which is equivalent to finding the s-dependent boundary operator $\mathbf{B}$, we apply the same procedure as in [8] or [9]. That is, we construct the matrix $\mathbf{C}$ such that all poles p_j, $j = 1, \ldots, k$ of $(p\mathbf{I} + \mathbf{C})^{-1}$ are located in the right half of the complex plane, i.e. $\Re p_j > 0$, $j = 1, \ldots, k$. This ensures that the exterior solution decays appropriately as $x \to \pm\infty$.

The square root of $\mathbf{C}^2 := -\mathbf{A}^{-1}\mathbf{E}$ is obtained by decomposing the matrices $\mathbf{A}$ and $\mathbf{E}$ into two components. 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

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

Normalizing the above matrices with respect to $\mathbf{A}_0$ generates new matrices

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

From the previous definitions together with the ansatz $\mathbf{C}(s) = \mathbf{C}_0 + \mathbf{s}\mathbf{C}_1 + \mathbf{s}^2\mathbf{C}_2 + \ldots$, where $\mathbf{C}_j \in \mathcal{C}^{k\times k}$, $j \geq 0$, we observe that we must find matrices $\mathbf{C}_j$ such that

$$(\mathbf{I} + s\mathbf{A}_1)\left(\mathbf{C}_0 + \mathbf{s}\mathbf{C}_1 + \mathbf{s}^2\mathbf{C}_2 + \ldots\right)\left(\mathbf{C}_0 + \mathbf{s}\mathbf{C}_1 + \mathbf{s}^2\mathbf{C}_2 + \ldots\right) = -\mathbf{E}_0 - s\mathbf{E}_1. \tag{15}$$

As is evident by comparing coefficients, to solve (15) we must first find $\mathbf{C}_0$ such that $\mathbf{C}_0^2 = -\mathbf{E}_0$. Because the matrix $\mathbf{E}_0$ 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 $\Re(\mathbf{C})_{jj} > 0$, $j = 1, \ldots, k$, see Alg. (1). Hence all poles corresponding to $\mathbf{C}_0$, with $\mathbf{C}_0$ constructed according to Alg. 1, are located in the right half of the complex plane.

Algorithm 1 Calculate $\mathbf{C} = \sqrt{-\mathbf{E}_0}$

```
for i = 1 to k do
  c_ii = sqrt(-E_{0,ii})
  if i > 1 then
    for j = i - 1 to1 do
      c_ij = ( -E_{0,ij} - sum_{m=j+1}^{i-1} c_im c_mj ) / (c_ii + c_jj)
    end for
  end if
end for
```

$$c_{ij} = \left(-\mathbf{E}_{0,ij} - \sum_{m=j+1}^{i-1} c_{im} c_{mj}\right) / (c_{ii} + c_{jj})$$

Remark. Since $\mathbf{C}_0 = \mathbf{C}(s)$ at $s = 0$ the condition $\mathbf{B} = -\mathbf{C}_0$ supplies the desired boundary conditions for the first step – i.e. the first $k-1$ intermediate solutions $g_1^{(1)}(x), \ldots, g_1^{(k-1)}(x)$ – and the solution $u_1(x)$ after the first step. By construction, all eigenvalues of the square-root have a positive real part, so that 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 (15). The corresponding pseudo-code is given in Alg. 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 $\mathbf{C}_j$, $j = 1, \ldots, n-1$

$\mathbf{Z} := -\left(\mathbf{E}_1 + \mathbf{A}_1 \mathbf{C}_0^2\right)$
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

$$\begin{aligned}
\dot{\mathbf{g}}_{i,0} &= \mathbf{B}(s)\mathbf{g}_{i,0} \\
&= \left(\mathbf{B}_0 + s\mathbf{B}_1 + \ldots + s^{i-1}\mathbf{B}_{i-1}\right)\mathbf{g}_{i,0} \\
&= \mathbf{B}_0\mathbf{g}_{i,0} + \mathbf{B}_1\mathbf{g}_{i-1,0} + \ldots + \mathbf{B}_{i-1}\mathbf{g}_{1,0}\,.
\end{aligned} \tag{16}$$

Eq. (16) 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 $a_1, \ldots, a_k$.. As 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 lower to larger step numbers and introduce a composite boundary vector $\mathbf{g}_0 = \left(\mathbf{g}_{1,0}^T, \ldots \mathbf{g}_{i-1,0}^T, \mathbf{g}_{i,0}^T\right)^T$. Similarly, we generate the composite boundary matrices

$$
\begin{aligned}
\mathbf{B} &= \left(\mathbf{B}_{i-1}, \mathbf{B}_{i-2}, \ldots \mathbf{B}_0\right), \quad 1 \le i \le n & (17) \\
\mathbf{C} &= -\mathbf{B}. & (18)
\end{aligned}
$$

in place of the boundary operator $\mathbf{B}(s)$, after which the normal derivative $\dot{\mathbf{g}}_{i,0}$ from system (16) 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 Alg. 5.

6 Finite-element Discretization

From the representation (10), we will now generate a finite-element discretization on the interior domain. For illustrative purposes, consider the first equation of the system (10) 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 finally performing a partial integration yields

$$
\begin{aligned}
&\left(v, g_i^{(1)}\right) + \left(\partial_x v, a_1 \partial_x g_i^{(1)}\right) - \left(a_1 \partial_x g_i^{(1)}\right)\Big|_{x-}^{x+} = \qquad (19) \\
&\qquad (v, u_{i-1}) + \left(\partial_x v, a'_1 \partial_x u_{i-1}\right) - \left(a'_1 \partial_x u_{i-1}\right)\Big|_{x-}^{x+} .
\end{aligned}
$$

The variational problem corresponding to this equation is therefore to find a function $g_i^{(1)} \in H^1(\Omega)$ such that (19) holds for any $v \in H^1(\Omega_{\mathrm{i}})$. The other equations of the system (10) 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 $V_{\mathrm{h}} \subset H^1(\Omega)$. 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 $v_h \in V_h$

$$
\begin{aligned}
&\left(v_h, g_{h,i}^{(1)}\right) + \left(\partial_x v_h, a_1 \partial_x g_{h,i}^{(1)}\right) - \left(a_1 \partial_x g_{h,i}^{(1)}\right)\Big|_{x-}^{x+} = \qquad (20) \\
&\qquad (v_h, u_{h,i-1}) + \left(\partial_x v_h, a'_1 \partial_x u_{h,i-1}\right) - \left(a'_1 \partial_x u_{h,i-1}\right)\Big|_{x-}^{x+} .
\end{aligned}
$$

A compact notation for (20) results if we define the vectors $\mathbf{b}_i^{(1)}, \mathbf{b}_i'^{(1)} \in \mathcal{C}^N$ with $N = \dim V_h$ by

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

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 $\mathbf{g}_i^{(1)} = \left(g_{h,i,1}^{(1)}, \dots, g_{h,i,N}^{(1)}\right)^T \in \mathcal{C}^N$ and $\mathbf{u}_{i-1} = (u_{h,i-1,1}, \dots, u_{h,i-1,N})^T \in \mathcal{C}^N$, that are the discrete counterparts of the continuous functions $g_i^{(1)}(x)$, $u_i(x)$ we have

$$(\mathbf{M} + \mathbf{A}_1)\, \mathbf{g}_i^{(1)} - \mathbf{b}_i^{(1)} = (\mathbf{M} + \mathbf{A}_1')\, \mathbf{u}_{i-1} - \mathbf{b}_i'^{(1)}. \tag{21}$$

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 (10), we generate the following block matrix equation in terms of the matrices and vectors introduced in the preceding paragraph,

$$\begin{pmatrix} \mathbf{M}+\mathbf{A}_1 & & & \\ & \mathbf{M}+\mathbf{A}_2 & & \\ & & \ddots & \\ & & & \mathbf{M}+\mathbf{A}_k \end{pmatrix} \begin{pmatrix} \mathbf{g}_i^{(1)} \\ \mathbf{g}_i^{(2)} \\ \vdots \\ \mathbf{u}_i \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(1)} \\ \mathbf{b}_i^{(2)} \\ \vdots \\ \mathbf{b}_i^{(k)} \end{pmatrix} =$$
$$\begin{pmatrix} \mathbf{M}+\mathbf{A}_1' & & & \\ & \mathbf{M}+\mathbf{A}_2' & & \\ & & \ddots & \\ & & & \mathbf{M}+\mathbf{A}_k' \end{pmatrix} \begin{pmatrix} \mathbf{u}_{i-1} \\ \mathbf{g}_i^{(1)} \\ \vdots \\ \mathbf{g}_i^{(k-1)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i'^{(1)} \\ \mathbf{b}_i'^{(2)} \\ \vdots \\ \mathbf{b}_i'^{(k)} \end{pmatrix}. \tag{22}$$

To solve the system (22), the vectors $\mathbf{b}_i^{(j)}$, $\mathbf{b}_i'^{(j)}$, $j = 1, \dots k$ must be constructed in accordance with the boundary conditions. The relationship between the discretized evolution equation (22) and the boundary condition (16) is determined by first decomposing the boundary condition at each boundary according to

$\mathbf{b}_i^{(j)} = \mathbf{b}_{i,-}^{(j)} + \mathbf{b}_{i,+}^{(j)}$ with

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

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

$$\mathbf{b}_{i,\pm} = \begin{pmatrix} \left(\pm a_1 \partial_x g_{i,h}^{(1)}\right)\Big|_{x=x_\pm} \\ \left(\pm a_2 \partial_x g_{i,h}^{(2)}\right)\Big|_{x=x_\pm} \\ \vdots \\ \left(\pm a_{k-1} \partial_x g_{i,h}^{(k-1)}\right)\Big|_{x=x_\pm} \\ \left(\pm a_k \partial_x u_i\right)\big|_{x=x_\pm} \end{pmatrix} \text{ and } \mathbf{b}'_{i,\pm} = \begin{pmatrix} \pm \left(a'_1 \partial_x u_{i-1}\right)\big|_{x=x_\pm} \\ \pm \left(a'_2 \partial_x g_{i,h}^{(1)}\right)\Big|_{x=x_\pm} \\ \vdots \\ \pm \left(a'_{k-1} \partial_x g_{i,h}^{(k-2)}\right)\Big|_{x=x_\pm} \\ \pm \left(a'_k \partial_x g_{i,h}^{(k-1)}\right)\Big|_{x=x_\pm} \end{pmatrix}. \tag{23}$$

We now derive an equation relating the vectors $\mathbf{b}_{i,\pm}$ and $\mathbf{b}'_{i,\pm}$ to the boundary condition (16). Regarding first $\mathbf{b}_{i-1,+}$, we have from (23) and (16)

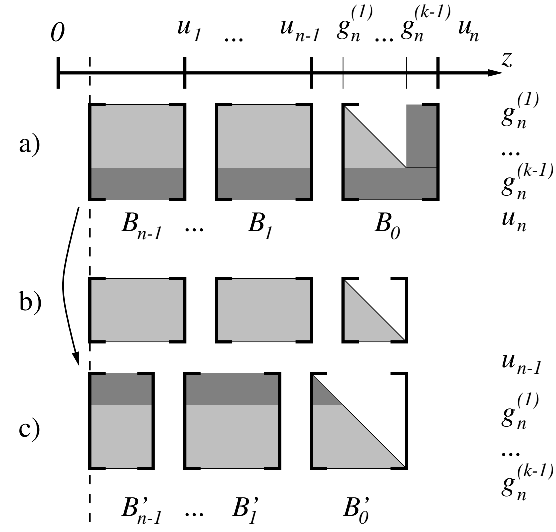
$$\begin{aligned} \mathbf{b}_{i,+} &= \operatorname{diag}(a_1, \ldots, a_k)\dot{\mathbf{g}}_{0,i} \\ &= \operatorname{diag}(a_1, \ldots, a_k)\mathbf{B}(s)\mathbf{g}_{0,i} \\ &= \operatorname{diag}(a_1, \ldots, a_k)\left(\mathbf{B}_0 + s\mathbf{B}_1 + s^2\mathbf{B}_2 + \ldots + s^{i-1}\mathbf{B}_{i-1}\right)\mathbf{g}_{0,i}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

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

$$\begin{aligned} \mathbf{b}_\pm &= \mathbf{B}_{\mathbf{d},\pm}\mathbf{g}_{0,i} + \mathbf{B}_{\mathbf{r},\pm}\mathbf{g}_i \\ \text{with } \mathbf{B}_{\mathbf{d},\pm} &:= \pm \operatorname{diag}(a_1, \ldots, a_k)\big|_{x=x_\pm} \operatorname{diag}(\mathbf{B}_{0,\pm}) \qquad (24) \\ \text{and } \mathbf{B}_{\mathbf{r},\pm}(s) &:= \pm \operatorname{diag}(a_1, \ldots, a_k)\big|_{x=x_\pm} \\ &\quad \cdot \left(\mathbf{B}_0 - \operatorname{diag}(\mathbf{B}_0) + s\mathbf{B}_1 + \ldots + s^{i-1}\mathbf{B}_{i-1}\right)\Big|_{x=x_\pm}, \qquad (25) \end{aligned}$$

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

Figure 1: Construction of the boundary operator $\mathbf{B}'$from the boundary operator $\mathbf{B}$



yields updated matrices $\mathbf{A}_j$ satisfying

$$\mathbf{A}_j = \mathbf{A}_j - \begin{pmatrix} \mathbf{B}_{\mathbf{d},-}(j,j) & & \\ & \mathbf{0} & \\ & & \mathbf{B}_{\mathbf{d},+}(j,j) \end{pmatrix}, \quad j = 1, \ldots k\,. \tag{26}$$

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

To derive a corresponding expression for $\mathbf{b}'_+$ we note that the vector $\mathbf{g}_i$ is ordered as $\left(g_i^{(1)}, g_i^{(2)}, \ldots g_i^{(k-1)}, u_i\right)^T$, which reflects the algebraic structure of the boundary condition (16). The discrete evolution system (22), however, requires the alternate ordering (23) $\left(u_{i-1}, g_i^{(1)}, g_i^{(2)}, \ldots g_i^{(k-1)}\right)^T$. Thus to derive a condition of the form

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

with an operator $\mathbf{B}'(s) := \mathbf{B}'_0 + s\mathbf{B}'_1 + \ldots + s^{i-1}\mathbf{B}'_{i-1}$, we must rearrange the columns and rows of the operator $\mathbf{B}$.

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 reduced matrix shown in part b) of Fig. 1. 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 part c) of Fig. 1. Finally, the resulting matrix is multiplied with the diagonal matrix diag $(a'_1, \dots a'_k)$ according to (23). The pseudo-code for these operations is given in Algorithm 3.

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 $\mathbf{b}' := -\left[\mathbf{C}(k, k+1 : nk), \underbrace{0, 0, \dots 0}_{k-1}\right]$

3: Compute $\mathbf{B}' \Leftarrow \text{diag}\,(a'_1, \dots a'_k) \begin{bmatrix} 0 & \mathbf{b}' \\ \mathbf{0} & \mathbf{B}' \end{bmatrix}$

Remark. To evolve the field over n propagation steps, we must first initialize the two boundary operators $\mathbf{B}_+$ and $\mathbf{B}_-$ with dimension $k \times (kn)$, 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 dimension $nk+1$ that will contain the boundary values.

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 Alg. 4. The structure and

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. (18)}
Compute $\mathbf{A}_j$, $j = 1, \dots, k$ and $\mathbf{M}$ {acc. to Eq.(20) and Eq. (21)}
Compute $\mathbf{B}'_\pm$ {acc. to Alg. 3}
Compute $\mathbf{B_d}$ {acc. to Eq. (24)}
Update $\mathbf{A}_j$, $\quad j = 1 : k$ using $\mathbf{B_d}$ {acc. to Eq. (26) }
Compute $\mathbf{B}_\pm := -\text{diag}(a_1, \dots, a_k)_\pm [\mathbf{C}_{n-1}, \mathbf{C}_{n-2}, \dots \mathbf{C}_0 - \text{diag}(\mathbf{C}_0)]_\pm$ {acc. to Eq. (25)}
$\mathbf{B}_\pm \Leftarrow [\mathbf{0}, \mathbf{B}_\pm(:, 1 : nk-1)]$
$\mathbf{B}_\pm := [\mathbf{B}_{n-1}, \mathbf{B}_{n-2}, \dots \mathbf{B}_0] \Leftarrow \mathbf{B}_\pm - \mathbf{B}'_\pm$

the numerical details of the resulting propagation algorithm is finally described in Alg. 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}
    $\mathbf{b} = \left(\mathbf{M} + \mathbf{A}'_j\right)\mathbf{g}$
    $c_- = \mathbf{B}_-(j, (n-i)k + 1 : (n-1)k + j)\mathbf{g}_-$
    $c_+ = \mathbf{B}_+(j, (n-i)k + 1 : (n-1)k + j)\mathbf{g}_+$
    $\mathbf{b} \Leftarrow \mathbf{b} + \begin{pmatrix} c_- \\ \mathbf{0} \\ c_+ \end{pmatrix}$
    Compute $\mathbf{g}$ from $\left(\mathbf{M} + \mathbf{A}_j\right)\mathbf{g} = \mathbf{b}$
    $\mathbf{g}_- \Leftarrow (\mathbf{g}_-,\ \mathbf{g}(1))^T, \mathbf{g}_+ \Leftarrow (\mathbf{g}_+, \mathbf{g}(N))^T$ {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 through a natural extension of our earlier analysis for the Schrödinger-type Padé-(2,0) approximant [9]. Assume that the algorithm is implemented with exact arithmetic, and consider the representation of the exterior solution given by (14), namely

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

in which by construction (Alg. 1) $\Re\left(\mathrm{diag}(\mathbf{C}_0)\right) > 0$. Further, in accordance with the algebraic properties of the shift operator, [6], we define $\mathbf{g}_{i,0} = 0$ for all $i \leq 0$. Introducing the matrix $\Delta = s\,(p\mathbf{I} + \mathbf{C}_0)^{-1}\left(\mathbf{C}_1 + s\mathbf{C}_2 \ldots + s^{i-2}\mathbf{C}_{i-1}\right)$ we can represent $\widehat{\mathbf{g}}_i(p)$ by

$$\begin{aligned}
\widehat{\mathbf{g}}_i(p) &= \left((p\mathbf{I} + \mathbf{C}_0)(\mathbf{I} + \Delta)\right)^{-1}\mathbf{g}_{i,0} \\
&= (\mathbf{I} + \Delta)^{-1}(p\mathbf{I} + \mathbf{C}_0)^{-1}\mathbf{g}_{i,0} \\
&= \left(\mathbf{I} - \Delta + \Delta^2 - \ldots\right)(p\mathbf{I} + \mathbf{C}_0)^{-1}\mathbf{g}_{i,0}\,.
\end{aligned} \tag{27}$$

This expansion around $s = 0$, which results from the Neumann series expansion, is a convergent representation of our solution [6]. Since the inverse Laplace transform of the scalar quantity $\widehat{g}(p) = (p + c_0)^{-1}$ is $g(x) = \exp(-c_0 x)$, we conclude that $\lim_{x\to\infty} g_i(x) = 0$ for $\Re(c_0) > 0$, which is our desired asymptotic boundary condition. This asymptotic property is further valid for any term of the form $(p + c_0)^{-j}$, $j \geq 1$ and consequently the above manipulations can be directly generalized to the vectorial representation (27). As well, since the Laplace variable p

occurs only within the factor $(p\mathbf{I}+\mathbf{C}_0)^{-1}$, while the full solution (27) is a linear combination of powers of such factors, the vector of solutions $\mathbf{g}_i(x)$ approaches 0 at infinity.

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

We consider first the discrete variational equation, (20), where we abbreviate $g^{(1)}_{h,i}$ by g and $u_{h,i-1}$ by u and a_1, a'_1 by a, a'. We must compute function g from its predecessor u. Accordingly, (20) reads

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

If we regard the special choice $v_h = g$ and $v_h = u.$, we obtain the equation system

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

Provided that a' and a are complex conjugates, which is the essential requirement to insure stability, the sum of both equation yields

$$(g,g) - 2\Re\,(a\partial_x g)|_{x_-}^{x_+} = (u,u) - 2\Re\,(a'\partial_x u)|_{x_-}^{x_+}.$$

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

$$(g,g)_{\mathrm{out},+} - 2\Re\,(a\partial_x g)|_{x_+}^{\infty} = (u,u)_{\mathrm{out},+} - 2\Re\,(a'\partial_x u)|_{x_+}^{\infty},$$

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

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

Since the global L^2-norm is thus conserved for every intermediate step, it is conserved for the entire discrete evolution. This establishes as well 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 described by

$$u(x,0) = u_0(x) = \text{const}\ \exp\left(-(x/10)^2\right)\exp\left(ik_0 x\sin\phi\right)$$

propagating in air. 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 [4], the computational domains are either $\Omega = (-50, 50) \times (0, 400)\mu m^2$ or $\Omega = (-50, 50) \times (0, 100)\mu m^2$ while the transverse step sizes Δx vary between $0.01\mu m \leq \Delta x \leq 0.2\mu m$. We consider first the intrinsic error associated with applying the implicit mid-point rule to plane wave solutions of the Helmholtz equation. Recall that for the exact one-way Helmholtz propagator

$$u(\Delta z) = u(0) \exp\left(-i\Delta z k_0 \sqrt{1 - \sin^2\phi}\right) ,$$

for which the implicit mid-point rule yields

$$u_{\mathrm{IMR}}(\Delta z) = u_{\mathrm{IMR}}(0) \frac{1 - i\Delta z k_0/2\left(1 - \sqrt{1 - \sin^2\phi}\right)}{1 + i\Delta z k_0/2\left(1 - \sqrt{1 - \sin^2\phi}\right)} .$$

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 $\log(u(\Delta z)) - \log(u_{\mathrm{IMR}}(\Delta z))$ which we compare in Fig. 2 to that obtained by instead 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 a Padé (8,8) approximant as evident from Fig. 3.

Next we display in Fig. 4 the spectral norms of the matrices $\mathbf{B}_i$, for the boundary conditions associated with both the (2,0) and (8,8) Padé approximants as a function of the number of propagation steps. Here the matrices $\mathbf{B}_i$ are defined and computed as in Alg. 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. The transverse grid spacing is here $\Delta x = 0.2\mu m$ while the computational domain is $\Omega = (-50, 50) \times (0, 400)\mu m^2$. 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. Further, we extended the numerical experiment by adding a second Gaussian beam, which propagates upward with an angle of $\pi/4$with respect to the z-axis. Fig. 7 demonstrates that in fact the wide-angle propagation can be realized with our numerical scheme. To examine the influence of the discretization error with respect to the transverse step length Δx, we now repeat our previous $(8, 8)$ Padé simulation, Fig. 5 with $\Delta x = 0.01\mu m$, cf. Fig.8. The boundary reflection, which

Figure 2: The phase error associated with the exact implicit mid-point discretization (solid line) compared to that of the corresponding Padé (2,0) approximant (dashed line).

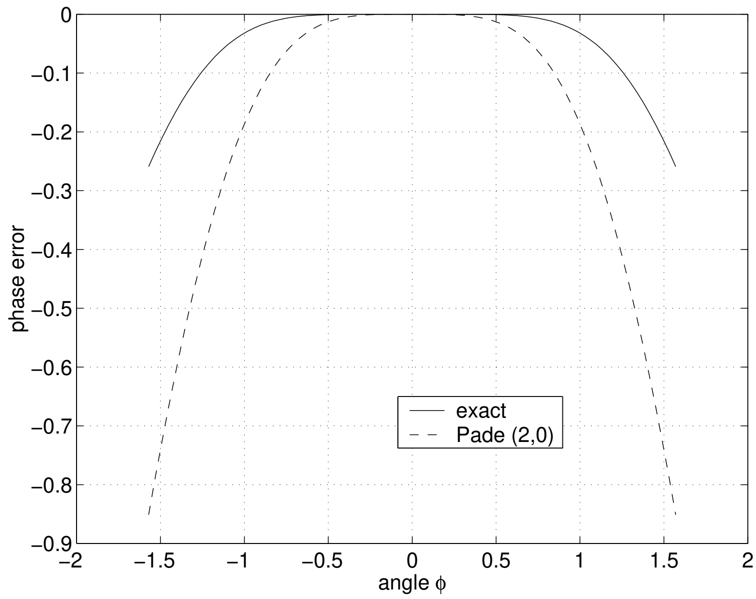


Figure 3: As in Fig. 2, but for a (8,8) Padé approximant.

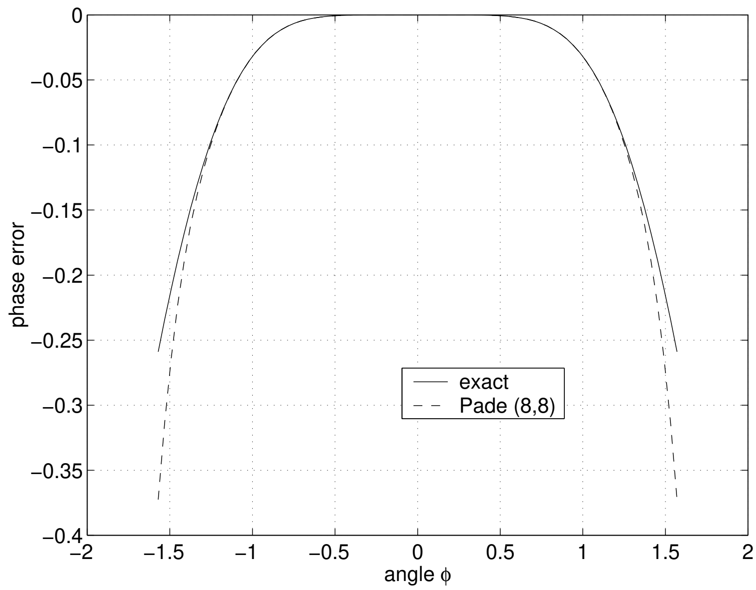


Figure 4: Spectral norm of the boundary matrices $\mathbf{B}_i$ as a function of the number of propagation steps.

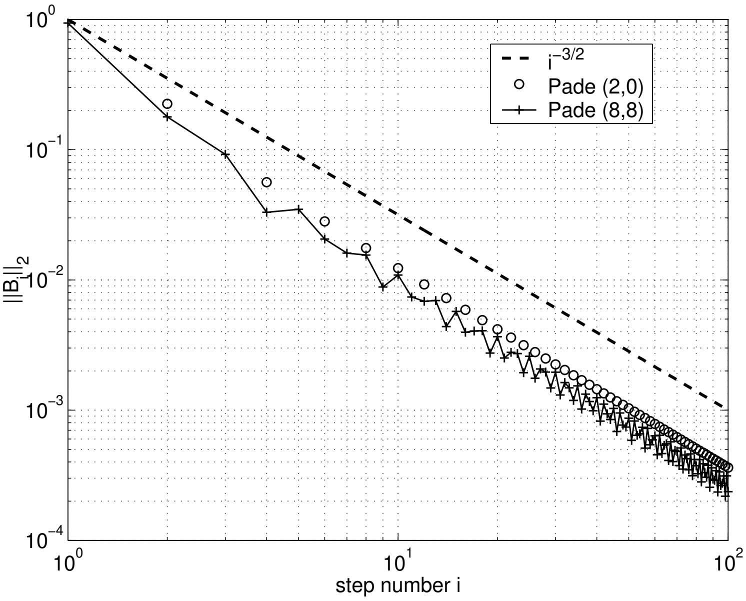


Figure 5: Gaussian beam propagation calculated with a (8,8) Padé propagator and $\Delta x = 0.2\mu m$. The dashed line represents the exact propagation angle $\theta = \pi/4$. The reflected field vanishes as $\Delta x \to 0$.

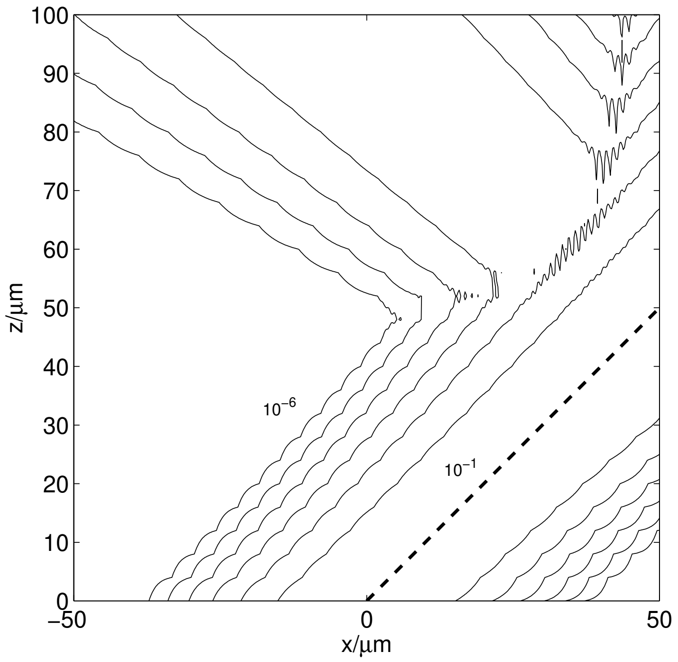


Figure 6: As in Fig. 5 except for a Padé (2,0) propagator. Note the error in the propagation angle.

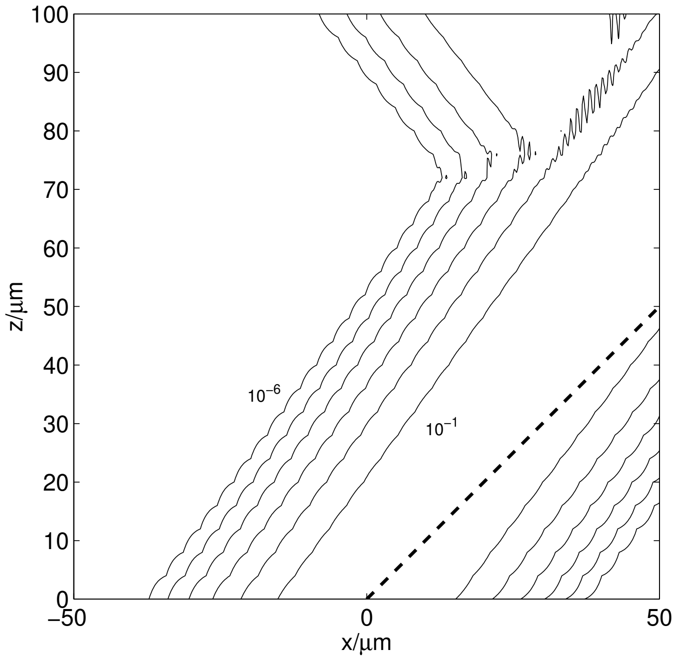


Figure 7: Propagation of two Gaussian beams at a relative angle of $\pi/2$.

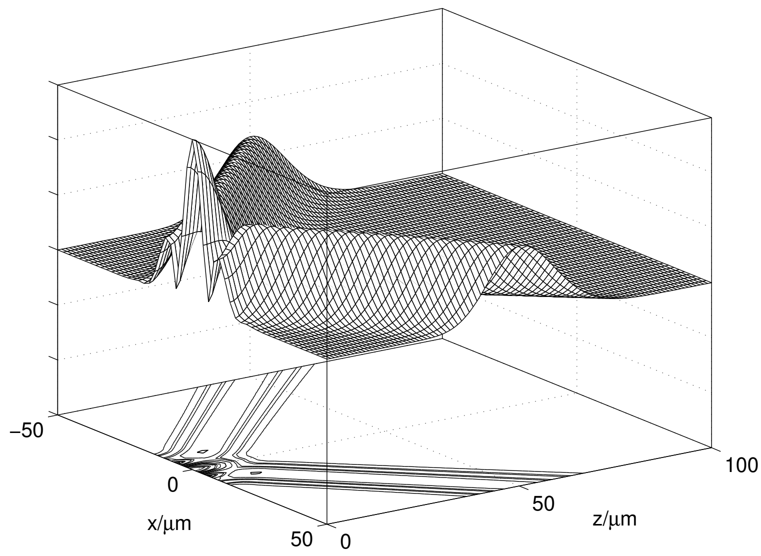
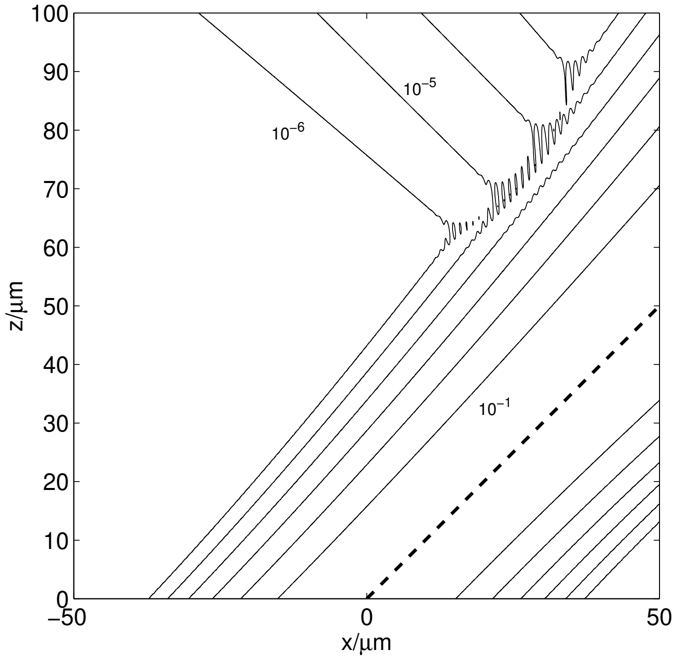


Figure 8: As in Fig. 5, but with $\Delta x = 0.01\mu m$.



vanishes in the $\Delta x \to 0$ limit is indeed significantly reduced. To analyze the actual dependence of the reflection from the discretization error, we display in the succeeding Fig. 9 the discrete $\mathrm{L}_2(-50, 50)$ norm as a function of propagation distance for transverse step sizes of $0.2\mu m, 0.1\mu m, 0..05\mu m, 0.025\mu m$, and $0.01\mu m$. 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 $O\left(\Delta x^2\right)$ 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 $0 \leq z \leq 400\mu m$ 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 are based on longitudinal discretization with the implicit mid-point rule. The

Figure 9: $The\mathrm{L}_2(-50,50)$ norm $||u||$ as a function of the propagation distance for varying step sizes Δx.

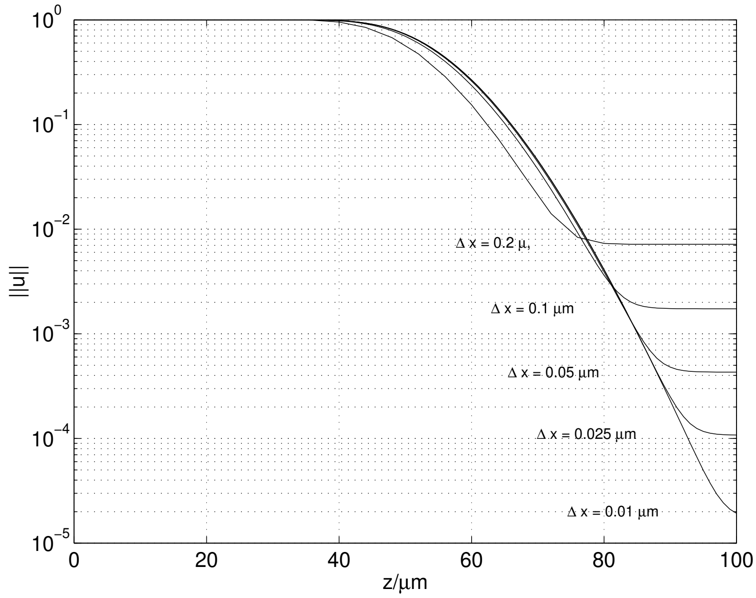
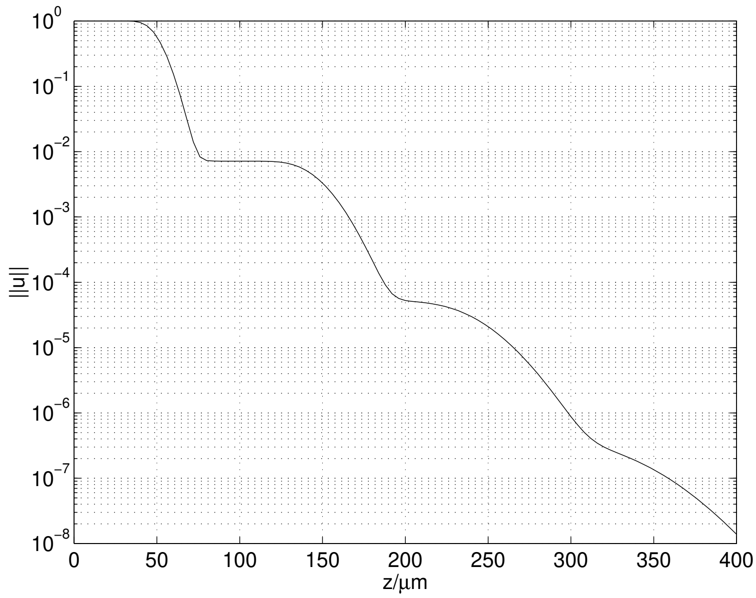


Figure 10: The discrete L_2-norm of the field within the computational window as a function of longitudinal distance for $0 \leq z \leq 400\mu m$ for the parameters of Fig. 5.



proof requires only that the rational approximations to the square-root operator obeys the condition $a_j = \bar{a}'_j$, $j = 1, \ldots, k$ and that the finite-element space V_h of the interior discrete problem is non-adaptive; 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.

Acknowledgment

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

References

[1] Anton Arnold and Matthias Erhardt. Discrete transparent boundary conditions for wide angle parabolic equations in underwater acoustics. *J. Comput. Phys.*, 145(2):611–638, 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):129–154, 1988.

[3] A. Bamberger, B. Engquist, L. Halpern, and P. Joly. Parabolic wave equation approximations in heterogeneous media. *SIAM J. Appl. Math.*, 48(1):99–128, 1988.

[4] T. Friese, F. Schmidt, and D. Yevick. Transparent boundary conditions for wide angle one-way Helmholtz equation. Preprint SC 99-45, ZIB, 1999. submitted to J. Comp. Phys.

[5] L. Halpern and L. N. Trefethen. Wide-angle one-way wave equations. *J. Acoust. Soc. Amer.*, 84(4):1397–1404, 1988.

[6] J. Mikusiński. *Operational Calculus.* Pergamon Press, London, New York, 1959.

[7] F. Schmidt. Construction of discrete transparent boundary conditions for Schrödinger-type equations. *Surv. Math. Ind.*, 9:87–100, 1999.

[8] F. Schmidt and P. Deuflhard. Discrete transparent boundary conditions for the numerical solution of Fresnel's equation. *Computers Math. Applic.*, 29:53–76, 1995.

[9] F. Schmidt and D. Yevick. Discrete transparent boundary conditions for Schrödinger-type equations. *J. Comput. Phys.*, 134:96–107, 1997.

[10] L. N. Trefethen and L. Halpern. Well-posedness of one-way wave equations and absorbing boundary conditions. *J. Acoust. Soc. Amer.*, 47:421–435, 1986.