An exact NRBC for 2D wave equation problems in unbounded domains

An exact NRBC for 2D wave equation problems in unbounded domains. *

S. Falletta†, G. Monegato‡

Abstract

We consider some 2D wave equation problems defined in an unbounded domain, possibly with far field sources. For their solution, by means of standard finite element or finite difference methods, we propose a non reflecting boundary condition (NRBC) on the chosen artificial boundary $\mathcal{B}$, which is based on a known space-time integral equation defining a relationship between the solution of the differential problem and its normal derivative on $\mathcal{B}$. Such a NRBC is exact, non local both in space and time. We discretize it by using a fast convolution quadrature technique in time and a collocation method in space. Besides showing a good accuracy and numerical stability, the proposed NRBC has the property of being suitable for artificial boundaries of general shapes; moreover, from the computational point of view, it is competitive with well known existing NRBCs of local type. It also allows the treatment of far field sources, that do not have to be necessarily included in the finite computational domain, being transparent for both incoming and outgoing waves.

KEY WORDS: wave equation; absorbing boundary conditions; space-time boundary integral equations; numerical methods

1 Introduction

Infinite or unbounded domains are often encountered in mathematical models associated with acoustic, aerodynamic, geophysical, electromagnetic and many other problems. Typically, the phenomenon of interest is local but embedded in a vast surrounding medium. Although the exterior region may not be truly unbounded, the boundary effects are often negligible, so that one further simplifies the problem by replacing the vast exterior by an infinite medium. Mathematical models of natural phenomena usually consist of partial differential equations, and many standard numerical methods, such as finite differences and finite elements, can be used to solve them. These can even handle complex geometries, inhomogeneous media and nonlinearity. However, they require a finite computational domain with prescribed boundary conditions. A

*This work was supported by the Ministero dell'Istruzione, dell'Università e della Ricerca of Italy, under the research program PRIN09: Boundary element methods for time dependent problems.

†Dipartimento di Scienze Matematiche, Politecnico di Torino, Italy. Email: silvia.falletta@polito.it

‡Dipartimento di Scienze Matematiche, Politecnico di Torino, Italy. Email: giovanni.monegato@polito.it

key issue is therefore the choice of a bounded computational domain, where one is interested in studying the behaviour of the solution, and the introduction of boundary conditions which guarantee that the solution of the initial boundary value problem inside the finite computational domain coincides with the restriction to the computational domain of the solution of the original problem, which is defined in the infinite region. The method of artificial (or absorbing) boundary condition (ABC) consists of introducing an artificial boundary $\mathcal{B}$ that truncates the infinite domain and determines two distinct regions: a bounded domain of interest Ω and a residual infinite domain $\mathcal{D}$. By analyzing the problem in $\mathcal{D}$, a non reflecting boundary condition (NRBC) on $\mathcal{B}$ is derived in order to avoid spurious reflections. Once the NRBC is given, it is used to solve the problem in Ω by using a numerical method such as, for example, finite differences, finite elements, finite volumes or spectral methods.

The NRBCs are usually divided into two main categories: exact, that is *non local*, both in space and time, and approximate, i.e., *local* both in space and time.

In the context of exact NRBC, we mention the Dirichlet to Neumann (DtN) method proposed by Givoli [12, 13] for elliptic problems, where the exact artificial boundary condition is based on the Dirichlet-to-Neumann (DtN) map associated with the operator and geometry under consideration. Such a map has been derived for various elliptic linear differential equations ([15, 16, 11, 10, 18, 36, 37, 4, 31]) and has also been applied to time-dependent wave problems ([12, 14, 19]). It is limited to circular and spherical artificial boundaries, and its main drawback stands in the fact that the boundary conditions also depends on the behaviour of the solution in the residual infinite domain $\mathcal{D}$; thus it requires further efforts to somehow truncate and approximate $\mathcal{D}$. In [28, 1, 32], the authors proposed fast, low-memory implementations of exact non local conditions, but limited to rectangular or circular (spherical) geometries which, especially in the three dimensional case, could be excessively large with respect to the domain of interest.

The earlier approximate NRBCs, still widely used, are those proposed by Engquist and Majda [7]. By using the Laplace-Fourier transform in time and in the plane tangential to the artificial boundary, they derived exact boundary conditions in terms of a pseudo-differential operator, which is then localized through a Taylor approximation up to the second order, and a Padé approximation for higher orders ([5]). Higdon [24, 25] derived a NRBC which is exact for any linear combination of plane waves whose angles of incidence are fixed a priori. In both cases, the high-order derivatives involved in these boundary conditions greatly complicate their use in any numerical scheme. As a result, first and second order boundary conditions were commonly used in practice. Later, high order local non reflecting boundary conditions for the wave equation, which do not involve derivatives of order greater than 2, were proposed by Collino [6], Givoli and Neta [17] and Hagstrom and Warburton [23]. In all these three cases, the methods proposed require a straight-edge boundary, and special treatment of the corner effects. When $\mathcal{B}$ is a disk/sphere, Hagstrom and Hariharan [22] derived a new formulation of the classical Bayliss and Turkel [3] NRBC of arbitrarily high order, without using derivatives of order greater than 2. For a review of these methods, see [14].

Many more papers have been published on this topic, in particular in the last decade; but their number is too large to mention them. All these papers deal with the construction of NRBCs with the property of absorbing only outgoing wave, not

incoming ones. Therefore, sources must necessarily be included in the computational domain, and this can be a severe drawback.

In the present work, we consider the (non homogeneous) exterior Dirichlet problem for the classical 2D wave equation, and propose for its solution a NRBC which arises from a known space-time boundary integral relationship that the problem solution and its normal derivative ought to satisfy at the chosen artificial boundary $\mathcal{B}$. It holds for a (smooth) curve of arbitrary shape; therefore, it can be used also in situations of multiple scattering (see [33], [34]), and even in more general ones. Moreover, it allows the problem to have non trivial data, whose (local) supports do not have necessarily to be included in the Ω domain, as it is usually done, in particular when they are away from the domain of interest. In such a case, our NRBC will naturally include the effects of these data and will be transparent for both outgoing and incoming waves. Such an ABC condition is of exact type and it is given by a linear combination of a single and a double layer operators, which are well known in Boundary Integral Equation (BIE) formulations of PDE problems.

For the discretization of the artificial condition, namely for the approximation of the single and double layer operators, we propose a numerical scheme which is based on a second order Lubich discrete convolution quadrature formula for the discretization of the time integral, coupled with a classical collocation method in space. When the discretization of the bounded domain Ω, where we apply the chosen finite element or finite difference scheme, is refined, the accuracy of the BIE discretization will also increase. Although we have not a proof for the consistency of this latter discretization, all the numerical tests we have performed seem to confirm this property.

If N denotes the number of time steps to be performed, the proposed discretized NRBC requires $O(N \log N)$ operations to compute, for each given collocation (space) point, the associated temporal convolution at all chosen instants. Moreover, when the artificial boundary is a circle, the space non locality is comparable, from the computational point of view, to a locality property. Indeed, in such a case, if M denotes the number of (equidistant) collocation points on the boundary, the computational cost of the construction of the proposed NRBC turns out to be $O(MN \log N)$, which is almost that of a NRBC of local type, both in space and time. For an artificial boundary of general shape, taking advantage of "sparsity" of the involved (approximated) matrices, the overall CPU time can be reduced significantly.

The final numerical scheme is then obtained by coupling such non reflecting condition with a second order finite element or finite difference method. The numerical examples that we present in the final section show that indeed the proposed NRBC is very competitive, from both the accuracy and the computational cost points of view.

The paper is organized as follows. In Section 2 we introduce the proposed space-time artificial boundary condition, and present three possible ways of representing the restriction of the original PDE problem to the bounded region of interest. In Section 3, first we discretize the chosen formulation, to determine the solution of the original problem in the (bounded) domain of interest, by means of a (space) finite element method, combined with a classical one-step time integration scheme. Then, we apply the proposed numerical approach to some test problems. Several remarks are made.

In Section 4, we replace the (space) finite element discretization by a classical finite difference scheme, and apply the resulting numerical method to a test problem. In this case, we compare the CPU time required by our approach with those required by the

corresponding schemes associated with the (first and second order) Engquist-Majda and Sommerfeld ABCs.

Finally, in Section 5 we make some further comments and draw some conclusions.

2 The model problem

Let $\Omega^i \subset \mathbb{R}^2$ be an open bounded domain with a sufficiently smooth boundary Γ; define $\Omega^e = \mathbb{R}^2 \setminus \bar{\Omega}^i$. We consider the following wave propagation problem in Ω^e:

$$\begin{cases} u^e_{tt}(\mathbf{x},t) - \Delta u^e(\mathbf{x},t) &= f(\mathbf{x},t) \quad \text{in } \Omega^e \times (0,T) \\ u(\mathbf{x},t) &= g(\mathbf{x},t) \quad \text{in } \Gamma \times (0,T) \\ u^e(\mathbf{x},0) &= u_0(\mathbf{x}) \quad \text{in } \Omega^e \\ u^e_t(\mathbf{x},0) &= v_0(\mathbf{x}) \quad \text{in } \Omega^e. \end{cases} \tag{1}$$

As often occurs in practical applications, we assume that the initial condition u_0, the initial velocity v_0 and the source term f are either trivial or have a local support. Since in general one has to determine the solution u^e of the above problem in a bounded subregion of Ω^e, surrounding the physical domain Ω^i, we truncate the infinite domain Ω^e by introducing an artificial smooth boundary $\mathcal{B}$. This boundary divides Ω^e into two sub-domains: a finite computational domain Ω, which is bounded internally by Γ and externally by $\mathcal{B}$, and an infinite residual domain $\mathcal{D}$.

We remark that we suppose to choose the artificial boundary in such a way that it detects the region where one has to compute the problem solution. This region does not necessarily have to contain the supports of the source term and of the initial data. Thus, in general, the support of a datum will be either in the (bounded) region of interest Ω, or in the residual domain $\mathcal{D}$. In the latter case it will be taken into account by a corresponding term of the artificial boundary condition formulation.

To obtain a well posed problem in Ω, we need to impose a proper boundary condition on $\mathcal{B}$. To do so, we analyze the problem in $\mathcal{D}$, and we impose on $\mathcal{B}$ the integral relation that the solution u and its normal derivative, with respect the outward normal, have to satisfy. Thus, denoting by $\partial_{\mathbf{n}_\mathcal{D}} = \frac{\partial}{\partial \mathbf{n}_\mathcal{D}}$ the outward (boundary) unit normal derivative for the problem exterior to the domain $\mathcal{D}$, we introduce the single and double layer integral operators, defined by

$$\mathcal{V}\lambda(\mathbf{x},t) := \int_0^t \int_\mathcal{B} G(\mathbf{x}-\mathbf{y}, t-\tau)\lambda(\mathbf{y},\tau) d\mathcal{B}_\mathbf{y} d\tau,$$

and

$$\mathcal{K}\varphi(\mathbf{x},t) = \int_0^t \int_\mathcal{B} \partial_{\mathbf{n}_\mathcal{D}} G(\mathbf{x}-\mathbf{y}, t-\tau)\varphi(\mathbf{y},\tau) d\mathcal{B}_\mathbf{y} d\tau,$$

respectively, where $G(\mathbf{x},t)$ denotes the wave equation fundamental solution in $\mathbb{R}^2$

$$G(\mathbf{x},t) = \frac{1}{2\pi} \frac{H(t-\|\mathbf{x}\|)}{\sqrt{t^2 - \|\mathbf{x}\|^2}},$$

$\delta(\cdot), H(\cdot)$ being the well known Dirac delta and Heaviside functions. Moreover, recalling

the representations (see [9]):

$$I_{u_0}(\mathbf{x},t) = \frac{\partial}{\partial t}\int_{\mathcal{D}} u_0(y)G(\mathbf{x}-\mathbf{y},t)dy, \tag{2}$$

$$I_{v_0}(\mathbf{x},t) = \int_{\mathcal{D}} v_0(y)G(\mathbf{x}-\mathbf{y},t)dy, \tag{3}$$

$$I_f(\mathbf{x},t) = \int_0^t \int_{\mathcal{D}} f(y,\tau)G(\mathbf{x}-\mathbf{y},t-\tau)dyd\tau, \tag{4}$$

for the possible "volume" terms generated by the non trivial source and the non homogeneous initial conditions, following the theory of the standard boundary integral equations, we obtain (for more details see [9]) the (boundary) relationship

$$\frac{1}{2}u(\mathbf{x},t) = \mathcal{V}\partial_{\mathbf{n}_\mathcal{D}}u(\mathbf{x},t) - \mathcal{K}u(\mathbf{x},t) + I_{u_0}(\mathbf{x},t) + I_{v_0}(\mathbf{x},t) + I_f(\mathbf{x},t) \quad \mathbf{x}\in\mathcal{B}. \tag{5}$$

Contrarily to what happens in the BEM method, equation (5) has not to be solved (neither u nor $\partial_{\mathbf{n}_\mathcal{D}}u$ are given at $\mathcal{B}$), but it represents the natural relation that u and its normal derivative ought to satisfy at each point of the artificial boundary and at each time t. Relation (5) represents an exact NRBC, non local both in time and in space.

The mapping properties of the operators $\mathcal{V},\mathcal{K}$ have been studied; see, for example, [30], [26]. The notation and the mathematics required to derive them is fairly heavy. Therefore, we simply outline a few of the main ingredients and results. To this end, we denote (see [26]) by $\mathcal{A}(\mu,X,Y)$, X,Y being two Hilbert spaces, the set of analytic functions $F:\mathbb{C}_+\to\mathcal{L}(X,Y)$ for which there exists a real number μ such that for all $\sigma>0$ we have $\|F(s)\|\le C_0|s|^\mu$, for some $C_0=C_0(\sigma)$ and all $s: Re(s)>\sigma$. We also denote by $\widehat{F}$ the Laplace transform of F.

It is then known ([26]) that the operators

$$\widehat{\mathcal{V}}(s)\Lambda(\mathbf{x}) := \int_{\mathcal{B}} \widehat{G}(\mathbf{x}-\mathbf{y},s)\Lambda(\mathbf{y})d\mathcal{B}_\mathbf{y}$$

and

$$\widehat{\mathcal{K}}(s)\Phi(\mathbf{x}) := \int_{\mathcal{B}} \widehat{\partial_{\mathbf{n}_\mathcal{D}}G}(\mathbf{x}-\mathbf{y},s)\Phi(\mathbf{y})d\mathcal{B}_\mathbf{y},$$

which denote the Dirichlet-to-Neumann and Neumann-to-Dirichlet operators for the equation $s^2u-\Delta u=0$ respectively, satisfy $\widehat{\mathcal{V}}\in\mathcal{A}(1,H^{-1/2}(\mathcal{B}),H^{1/2}(\mathcal{B}))$ and $\widehat{\mathcal{K}}\in\mathcal{A}(3/2,H^{1/2}(\mathcal{B}),H^{1/2}(\mathcal{B}))$. From these, we obtain [30] the following mapping properties for the operators $\mathcal{V}$ and $\mathcal{K}$:

$$\mathcal{V}: H_0^{r+1}(0,T;H^{-1/2}(\mathcal{B}))\to H_0^r(0,T;H^{1/2}(\mathcal{B})), r\ge 0 \tag{6}$$

and

$$\mathcal{K}: H_0^{r+3/2}(0,T;H^{1/2}(\mathcal{B}))\to H_0^r(0,T;H^{1/2}(\mathcal{B})), r\ge 0. \tag{7}$$

These spaces are defined as follows. Set first $H_0^r(0,T)=\{g_{|_{(0,T)}} : g\in H^r(\mathbb{R}) \text{ with } g\equiv 0 \text{ on } (-\infty,0)\}$, where H^r denotes the classical Sobolev space of order r. When r is an integer, this space consists of those functions g whose r-th distributional derivative is in $L^2(0,T)$ and which have $g(0)=\dots g^{(r-1)}(0)=0$. Then:

• $H_0^r(0,T;X)$ is the space of $H_0^r(0,T)$ functions of t, $\phi(\mathbf{x},t)$, such that, setting $\phi(\mathbf{x},t)=\phi(t)(\mathbf{x})$, we have $\phi(t)\in X$, with $\| \|\phi(t)\|_X\|_{H^r(0,T)} < \infty$.

• $H^{1/2}(\mathcal{B})$ and $H^{-1/2}(\mathcal{B})$ are the trace space on the artificial boundary, of $H^1(\Omega)$ functions, and the corresponding dual space, respectively.

Recalling the well known embedding property: $H^r(0,T)\subset C^m[0,T]$ for $r>m+1/2$, from (6) and (7) we deduce that the assumption $r>3/2$ guarantees that

$$\mathcal{V}: H_0^r(0,T;H^{-1/2}(\mathcal{B}))\to C([0,T];H^{1/2}(\mathcal{B})).$$

Furthermore, if $r>2$ we also have

$$\mathcal{K}: H_0^r(0,T;H^{1/2}(\mathcal{B}))\to C([0,T];H^{1/2}(\mathcal{B})).$$

Here and in the following, $C^m(I;X)$ ($C=C^0$) denotes the space of C^m functions of $t\in I$, such that for each value of t the corresponding function of $\mathbf{x}$ belongs to the space X.

To apply the numerical method that we will describe in Sections 3 and 4, we will however assume that the data of our PDE problem satisfy the smoothness and compatibility conditions which guarantee the existence and uniqueness of a classical solution, that is, $u\in C^2$ with respect to its variables. Under these assumptions, all terms of our NRBC (5) will be, in particular, continuous functions of $\mathbf{x}\in\mathcal{B}$.

Remark 2.1 *The above NRBC (5) is mathematically non local both in space and time. However, because of the properties of the Lubich convolution quadrature, the computational cost due to this non locality can be cut down significantly. For example, when we choose as artificial boundary a circle, the true computational cost associated with our boundary condition is essentially of local type. Also the time global dependence is reduced significantly by using the FFT algorithm, to compute simultaneously all the needed quantities. From the computational complexity point of view, it turns out to be almost local.*

These issues will be discussed, with more details, in the examples we will present in Section 3.3.

Remark 2.2 *We will suppose that the local support of each datum is included either in Ω or in $\mathcal{D}$. In the first case we will obviously have that the corresponding volume integral in (5) vanishes, and the data will be treated by the numerical scheme one chooses to solve the problem in Ω. In the second case, because of the local support, the domain of integration of the volume integrals is not the whole infinite domain $\mathcal{D}$. Moreover, because of the presence of the Heaviside function in the expression defining the kernel G, each term will require the computation of volume integrals defined on the intersection of the corresponding local support with the disk of radius t centered at the point where the NRBC is collocated.*

The determination of this intersection is particularly simple when the data support is itself a circle, or the data decays smoothly to zero at the boundary of its support, and we trivially embed this latter into a circle, that we will then consider as the data "practical" support.

We finally observe that the computation of I_f is very simple when the source f is concentrated at a point, that is, is of the type $f(\mathbf{x},t)=h(t)\delta(\mathbf{x}-x_0)$ (see Section 3.3, Example 6, for details).

From now on, to simplify the description, we assume that the local supports of u_0, v_0 and f are contained in Ω, so that $I_{u_0} = I_{v_0} = I_f = 0$. The treatment of source and initial data, whose supports are in $\mathcal{D}$, is postponed to Section 3.3. Thus, denoting by $\partial_{\mathbf{n}} = \frac{\partial}{\partial \mathbf{n}}$ the outward (boundary) unit normal derivative for the problem exterior to the domain Ω, and noting that $\partial_{\mathbf{n}} = -\partial_{\mathbf{n}_\mathcal{D}}$, the model problem (defined in the domain of interest Ω) takes the form:

$$\begin{cases} u_{tt}(\mathbf{x},t) - \Delta u(\mathbf{x},t) & = f(\mathbf{x},t) \quad \text{in } \Omega \times (0,T) \\ u(\mathbf{x},t) & = g(\mathbf{x},t) \quad \text{in } \Gamma \times (0,T) \\ \frac{1}{2}u(\mathbf{x},t) + \mathcal{V}\partial_{\mathbf{n}} u(\mathbf{x},t) + \mathcal{K}u(\mathbf{x},t) & = 0 \quad \text{in } \mathcal{B} \times (0,T) \\ u(\mathbf{x},0) & = u_0(\mathbf{x}) \quad \text{in } \Omega \\ u_t(\mathbf{x},0) & = v_0(\mathbf{x}) \quad \text{in } \Omega. \end{cases} \tag{8}$$

Remark 2.3 *Recalling some existence and uniqueness results on the solution of problem (1) in proper functional spaces (see [9]), it is not difficult to show that under the same assumptions also problem (8) has a unique solution in the corresponding spaces; this coincides with the restriction of the solution of the former problem in the Ω domain. Indeed, assuming for example that (1) has a unique solution in the space $C^1([0,T];H^1(\Omega^e))$, this trivially satisfies also the reduced problem (8). Moreover, if by contradiction this latter problem has also another (non trivial) solution in $C^1([0,T];H^1(\Omega))$, the difference between these two solutions defines a non trivial solution of the associated fully homogeneous version of problem (8). Knowing this solution, hence the corresponding Dirichlet and Neumann data on the artificial boundary $\mathcal{B}$, we can use the single-double layer potential representation to extend the latter solution into the infinite domain D, to define a non trivial $C^1([0,T];H^1(\Omega^e))$ solution of the fully homogeneous version of problem (1). Being the solution of this unique, it must necessarily be the trivial one.*

2.1 Restriction of the model problem to the domain of interest

There are mainly three approaches that can be considered to couple our NRBC with the model problem. The first one is given by (8), where all terms are continuous functions of their variables and, therefore, equalities are pointwise defined. This is the so-called **strong formulation** of the problem.

The other two are weak type. To describe them, we introduce the spaces

$$X = \{u \in H^1(\Omega), u = g \,\text{on}\, \Gamma\}$$

and

$$X_0 = \{u \in H^1(\Omega), u = 0 \,\text{on}\, \Gamma\}.$$

We also introduce the additional unknown function $\lambda(\mathbf{x},t) = \lambda(t)(\mathbf{x}) := \partial_{\mathbf{n}} u(\mathbf{x},t)$, which is defined only on the boundary $\mathcal{B}$, in general by means of a trace operator (see [38], Sect. 1.3), and set $u(t)(\mathbf{x}) = u(\mathbf{x},t)$. Then, the problem defined in the domain of interest Ω consists in finding the couple of unknown functions $u(t), \lambda(t)$ such that one of the following (alternative) formulations is satisfied.

Variational formulation 1. By considering the weak form in space of the first and third equations of Problem (8), we obtain:
given $f \in L^2(\Omega\times(0,T)), u_0 \in X, v_0 \in L^2(\Omega)$, *find* $u(t) \in C^0([0,T];X)\cap C^1([0,T];L^2(\Omega))$ *and* $\lambda(t) \in C^0([0,T];H^{-1/2}(\mathcal{B}))$ *such that*

$$\begin{cases} \frac{d^2}{dt^2}(u(t),w)_\Omega + a(u(t),w) - (\lambda(t),w)_\mathcal{B} & = (f(t),w), \quad \forall w \in X_0 \\ \frac{1}{2}(u(t),\varphi)_\mathcal{B} + ((\mathcal{V}\lambda)(t),\varphi)_\mathcal{B} + ((\mathcal{K}u)(t),\varphi)_\mathcal{B} & = 0, \quad \forall \varphi \in H^{1/2}(\mathcal{B}) \\ u(0) & = u_0 \\ \frac{du}{dt}(0) & = v_0. \end{cases} \tag{9}$$

holds in the distributional sense in $(0,T)$, *where* $a : X \times X \to \mathbb{R}$ *is the classical bilinear form*

$$a(v,w) = \int_\Omega \nabla v \cdot \nabla w,$$

and $(v,w)_S = \int_S vw$ $(S = \Omega$ *or* $\mathcal{B})$.

Variational formulation 2. Here we consider the weak form only of the wave equation, i.e.,

$$\begin{cases} \frac{d^2}{dt^2}(u(t),w)_\Omega + a(u(t),w) - (\lambda(t),w)_\mathcal{B} & = (f(t),w), & \forall w \in X_0 \\ \frac{1}{2}u(\mathbf{x},t) + \mathcal{V}\lambda(\mathbf{x},t) + \mathcal{K}u(\mathbf{x},t) & = 0 & \text{on } \mathcal{B} \\ u(0) & = u_0 & \\ \frac{du}{dt}(0) & = v_0. & \end{cases} \tag{10}$$

for $t \in (0,T]$.

To apply this formulation we ought to assume that all equalities are defined in the strong sense, and in particular that all their terms are continuous functions of the corresponding variables. This property is certainly guaranteed by the assumptions we have made in the sentence preceding Remark 2.1 above.

Formulation 1 is certainly of interest; however, its numerical solution using methods such as finite and boundary elements, probably makes it hardly competitive with existing NRBC approaches, in particular when the problem does not have far field sources. We recall that an analogous setting for an elliptic problem has been studied in [8].

From this point of view, the strong formulation and formulation 2 are certainly more appealing, although they require stronger smoothness conditions. Therefore, in the next sections we will consider only these two, and apply them to several test problems. To solve these, formulation 2 will naturally be associated with a (space) finite element method (see Sect. 3), while the space discretization of the strong formulation will be performed by means of finite differences (see Sect. 4).

3 Reduced problem discretization. The Finite Element Method

3.1 A Lubich-collocation method for the NRBC

In this paper we mainly consider smooth problems, i.e., problems defined on domains having a smooth boundary and smooth compatible data. However, as mentioned in [9], the proposed BIE discretization seems to perform well also in cases where such assumptions are not all satisfied. To define this discretization we adopt the numerical approach which combine a second order (time) convolution quadrature formula of Lubich (see [29]) with a classical space collocation method. This because, in the following, the NRBC will be combined only with second order finite element and finite difference methods. We recall however that there exist also higher order Lubich convolution quadratures (see [35], [2]), that could be combined with higher order space discretizations.

We consider the integral relation (5), and we assume that $\mathcal{B}$ is a smooth boundary, that for simplicity we assume to be defined by a (smooth) parametrization. For the time discretization, we split the interval $[0,T]$ into N steps of equal length $\Delta_t = T/N$ and collocate the equation at the discrete time levels $t_n = n\Delta_t$, $n = 0,\dots,N$:

$$\frac{1}{2}u(\mathbf{x},t_n) + (\mathcal{V}\lambda)(\mathbf{x},t_n) + (\mathcal{K}u)(\mathbf{x},t_n) \quad = 0 \tag{11}$$

After having exchanged the order of integration, the time integrals appearing in the definition of the single and double layer operators are discretized by means of the Lubich convolution quadrature formula associated with the BDF method of order $p = 2$ (see [9]). We obtain:

$$(\mathcal{V}\lambda)(\mathbf{x},t_n) \quad \approx \quad \sum_{j=0}^{n} \int_{\mathcal{B}} \omega_{n-j}^{\mathcal{V}}(\Delta_t; ||\mathbf{x}-\mathbf{y}||)\lambda(\mathbf{y},t_j)\, d\mathcal{B}_{\mathbf{y}}, \quad n = 0,\dots,N \tag{12}$$

$$(\mathcal{K}u)(\mathbf{x},t_n) \quad \approx \quad \sum_{j=0}^{n} \int_{\mathcal{B}} \omega_{n-j}^{\mathcal{K}}(\Delta_t; ||\mathbf{x}-\mathbf{y}||)u(\mathbf{y},t_j)\, d\mathcal{B}_{\mathbf{y}}, \quad n = 0,\dots,N \tag{13}$$

whose coefficients $\omega_n^{\mathcal{J}}, \mathcal{J} = \mathcal{V},\mathcal{K}$, are given by

$$\omega_n^{\mathcal{J}}(\Delta_t; ||\mathbf{x}-\mathbf{y}||) = \frac{1}{2\pi\imath}\int_{|z|=\rho} K^{\mathcal{J}}\left(||\mathbf{x}-\mathbf{y}||, \frac{\gamma(z)}{\Delta_t}\right) z^{-(n+1)}\, dz$$

where in this case $K^{\mathcal{V}} = \widehat{G}$ is the Laplace transform of the kernel G appearing in the definition of the single layer operator $\mathcal{V}$, and $K^{\mathcal{K}} = \widehat{\partial G/\partial \mathbf{n}}$ is the Laplace transform of the kernel $\partial G/\partial \mathbf{n}$ appearing in the definition of the double layer operator $\mathcal{K}$.

The function $\gamma(z) = 3/2 - 2z + 1/2z^2$ is the so called characteristic quotient of the BDF method of order 2. The parameter ρ is such that for $|z| \le \rho$ the corresponding $\gamma(z)$ lies in the domain of analyticity of $K^{\mathcal{J}}$.

The Laplace transforms $K^{\mathcal{J}}$ can be computed by using some well known properties of the modified Bessel functions (see formulas 8.486(11,16,17) in [20]). In particular,

we have that

$$K^{\mathcal{V}}(r,s) = \frac{1}{2\pi} K_0(rs),$$
$$K^{\mathcal{K}}(r,s) = -\frac{1}{2\pi} s K_1(rs) \frac{\partial r}{\partial \mathbf{n}}, \tag{14}$$

where $K_0(z)$ and $K_1(z)$ are the second kind modified Bessel function of order 0 and 1, respectively.

By introducing the polar coordinate $z = \rho e^{i\varphi}$, the above integrals can be efficiently computed by the trapezoidal rule with L equal steps of length $2\pi/L$:

$$\omega_n^{\mathcal{J}}(\Delta_t; r) \approx \frac{\rho^{-n}}{L} \sum_{l=0}^{L-1} K^{\mathcal{J}} \left(r, \frac{\gamma(\rho \exp(il2\pi/L)}{\Delta_t} \right) \exp(-inl2\pi/L). \tag{15}$$

In this latter we choose $L = 2N$ and $\rho^N = \sqrt{\varepsilon}$, since Lubich in ([29]) has shown that this choice leads to an approximation of ω_n with relative error of size $\sqrt{\varepsilon}$, if $K^{\mathcal{J}}$ is computed with a relative accuracy bounded by ε. The choice of ε suggested by Lubich is 10^{-10}. According to the previous statement, this should give a relative accuracy of order 10^{-5}, which is sufficient for the tests we have performed and that we will present in the examples that will follow. For each given $\mathbf{x} \in B$, all the $\omega_n^{\mathcal{J}}$ can be computed simultaneously by the FFT, with $O(N \log N)$ flops.

For the space discretization, first we introduce a parametrization of the curve $\mathcal{B}$, $\mathbf{x} = \boldsymbol{\psi}(x) = (\psi_1(x), \psi_2(x))$ and $\mathbf{y} = \boldsymbol{\psi}(y) = (\psi_1(y), \psi_2(y))$ with $x, y \in [a, b]$. Notice that this requirement is not a restriction. Indeed, since the contour $\mathcal{B}$ can be arbitrarily chosen, we can always define a smooth parametric curve having the desired shape, by taking, for example, a (smooth) parametric cubic spline associated with a chosen set of points in $\mathbb{R}^2$.

Then, we approximate the unknown function u_{Δ_t} and the unknown normal derivative λ_{Δ_t} by continuous piecewise linear functions, associated with a uniform partition $\{x_h\}_{h=1}^{M+1}$ of the parametrization interval $[a, b]$. These are written in the form

$$u_{\Delta_t}(\boldsymbol{\psi}(x), t_j) \approx \sum_{k=1}^{M+1} u_k^j N_k(x)$$

and

$$\lambda_{\Delta_t}(\boldsymbol{\psi}(x), t_j) \approx \sum_{k=1}^{M+1} \lambda_k^j N_k(x)$$

where $\{N_k(x)\}$ are the classical Lagrangian basis functions of local degree 1.

Since the role of the NRBC is to define on $\mathcal{B}$ a relationship between the (outgoing) wave and its normal derivative, which prevents the raising of (spurious) incoming waves, the more accurate is the discretized relationship the more transparent this will be. To this end, having chosen a continuous piecewise linear approximant for u_{Δ_t}, we use an approximant of the same type also for λ_{Δ_t} (and not, as usual when solving a BIE, a piecewise constant function).

Taking into account that the curve $\mathcal{B}$ is closed, we set $u_1^j = u_{M+1}^j$ and $\lambda_1^j = \lambda_{M+1}^j$. Finally, by collocating the fully discretized equation at the points $\xi_h = x_h$,

$h = 1, \ldots, M$, we obtain the following linear system:

$$\left(\frac{1}{2}\mathbf{I} + \mathbf{K}_0\right)\mathbf{u}^n + \sum_{j=0}^{n-1}\mathbf{K}_{n-j}\mathbf{u}^j + \mathbf{V}_0\boldsymbol{\lambda}^n + \sum_{j=0}^{n-1}\mathbf{V}_{n-j}\boldsymbol{\lambda}^j = 0, \quad n = 1, \ldots, N \tag{16}$$

in the unknown vectors $\mathbf{u}^j = (u_1^j, \ldots, u_M^j)$, $j = 0, \ldots, n$, and $\boldsymbol{\lambda}^j = (\lambda_1^j, \ldots, \lambda_M^j)$, $j = 0, \ldots, n$. The symbol $\mathbf{I}$ denotes the identity matrix of order M, while the matrices $\mathbf{V}$ and $\mathbf{K}$ are given by

$$(\mathbf{V}_{n-j})_{hi} = \int_{\mathcal{B}} \omega_{n-j}^{\mathcal{V}}(\Delta_t; \|\mathbf{x}_h - \mathbf{y}\|) N_i(\mathbf{y}) d\mathcal{B}_{\mathbf{y}}, \tag{17}$$

and

$$(\mathbf{K}_{n-j})_{hi} = \int_{\mathcal{B}} \omega_{n-j}^{\mathcal{K}}(\Delta_t; \|\mathbf{x}_h - \mathbf{y}\|) N_i(\mathbf{y}) d\mathcal{B}_{\mathbf{y}}. \tag{18}$$

From the computational point of view, supposing to know $\mathbf{u}^j$ and $\boldsymbol{\lambda}^j$ at the time steps $j = 0, \ldots, n-1$, the absorbing condition at time t_n is given by

$$\left(\frac{1}{2}\mathbf{I} + \mathbf{K}_0\right)\mathbf{u}^n + \mathbf{V}_0\boldsymbol{\lambda}^n = -\sum_{j=0}^{n-1}\mathbf{K}_{n-j}\mathbf{u}^j - \sum_{j=0}^{n-1}\mathbf{V}_{n-j}\boldsymbol{\lambda}^j, \quad n = 1, \ldots, N. \tag{19}$$

Remark 3.1 *As described in [9], for each row index, the corresponding row elements of all the above matrices can be computed simultaneously by means of the FFT algorithm, after replacing, in the representations (17), (18), the ω kernel by its discretization (15), and exchanging the integration symbol with that of the quadrature sum. For the evaluation of the integrals see [9].*

3.2 The complete discrete scheme

In order to derive the complete numerical method we propose to solve (10), we start by describing the time discretization of its first equation. We choose the Crank-Nicolson scheme, of second order and unconditionally stable, which is well suited even for long time intervals, although other methods can be considered as well. To do so, we introduce the variable $v := \frac{\partial u}{\partial t}$ and we rewrite (10) as follows:

$$\begin{cases} \frac{d}{dt}(v(t), w) + a(u(t), w) - b(\lambda(t), w) & = (f(t), w), \quad \forall\, w \in X_0 \\ \frac{\partial u}{\partial t}(\mathbf{x}, t) & = v(\mathbf{x}, t), \quad \forall\, \mathbf{x} \in \Omega \\ \frac{1}{2}u(\mathbf{x}, t) + \mathcal{V}\lambda(\mathbf{x}, t) + \mathcal{K}u(\mathbf{x}, t) & = 0 \quad \text{in } \mathcal{B} \times (0, T] \\ u(\mathbf{x}, 0) & = u_0(\mathbf{x}), \quad \forall\, \mathbf{x} \in \Omega \\ v(\mathbf{x}, 0) & = v_0(\mathbf{x}), \quad \forall\, \mathbf{x} \in \Omega. \end{cases} \tag{20}$$

for all $t \in (0, T]$. Denoting by U^n, V^n, Λ_n and F^n the approximations of u, v, λ and f at time t_n, and applying the Crank-Nicolson discretization to the first two equations in (20), we have

$$\begin{array}{ll} \left(\frac{V^{n+1} - V^n}{\Delta_t}, w\right) + a\left(\frac{U^{n+1} + U^n}{2}, w\right) - b\left(\frac{\Lambda^{n+1} + \Lambda^n}{2}, w\right) & = \left(\frac{F^{n+1} + F^n}{2}, w\right), \quad \forall\, w \in X_0 \\ \frac{V^{n+1} + V^n}{2} & = \frac{U^{n+1} - U^n}{\Delta_t}, \quad \forall\, \mathbf{x} \in \Omega. \end{array}$$

From the second relation we get $V^{n+1} = \frac{2}{\Delta_t}(U^{n+1} - U^n) - V^n$, which, inserted in the first relation, leads to

$$(U^{n+1}, w) + \frac{\Delta_t^2}{4} a(U^{n+1}, w) - \frac{\Delta_t^2}{4} b(\Lambda^{n+1}, w) = (U^n, w) - \frac{\Delta_t^2}{4} a(U^n, w) + \frac{\Delta_t^2}{4} b(\Lambda^n, w) + \Delta_t (V^n, w) + \frac{\Delta_t^2}{4}(F^{n+1} + F^n, w), \; \forall\, w \in X_0$$

For the space finite element discretization, we define a regular triangular mesh $\mathcal{T}_h$ on Ω, with mesh size bounded by h. Let

$$X_h = \{w_h \in C^0(\Omega) : w_{h_{|K}} \in \mathbb{P}^1(K), K \in \mathcal{T}_h, w_{h_{|\Gamma}} = g\} \subset H^1,$$

$$X_{h,0} = \{w_h \in C^0(\Omega) : w_{h_{|K}} \in \mathbb{P}^1(K), K \in \mathcal{T}_h, w_{h_{|\Gamma}} = 0\}, \subset H_0^1$$

be the spaces of linear conforming finite elements in the domain Ω associated with the mesh T_h. Let W_h be the trace space of finite element basis functions of X_h restricted to $\mathcal{B}$. The Galerkin formulation of the above equation reads: for each $n = 0, \cdots, N-1$, find $(U_h^{n+1}, \Lambda_h^{n+1}) \in X_h \times W_h$ such that, for all $w_h \in X_{h,0}$

$$\begin{aligned}(U_h^{n+1}, w_h) + \frac{\Delta_t^2}{4} a(U_h^{n+1}, w_h) \;&-\; \frac{\Delta_t^2}{4} b(\Lambda_h^{n+1}, w_h) = (U_h^n, w_h) - \frac{\Delta_t^2}{4} a(U_h^n, w_h) \\ &+\; \frac{\Delta_t^2}{4} b(\Lambda_h^n, w_h) + \Delta_t (V_h^n, w_h) + \frac{\Delta_t^2}{4}(F^{n+1} + F^n, w_h)\end{aligned} \tag{21}$$

Let $\{N_i\}_{i \in \mathcal{S}}$ denote the set of finite element basis functions defined on the triangulation $\mathcal{T}_h$, $\mathcal{S}$ being the set of the total number of degrees of freedom (DOF) in Ω. The set $\mathcal{S}$ can be naturally split as follows: $\mathcal{S} = \mathcal{S}_I \cup \mathcal{S}_\mathcal{B}$, where $\mathcal{S}_I$ is the set of the internal DOF and $\mathcal{S}_\mathcal{B}$ is the set of the DOF associated to the artificial boundary $\mathcal{B}$. By properly reordering the unknowns, we obtain the unknown vector $\mathbf{U}^n = [\mathbf{U}_I^n, \mathbf{U}_\mathcal{B}^n]^T$, whose two components $\mathbf{U}_I^n$ and $\mathbf{U}_\mathcal{B}^n$ represent the unknown values associated with the internal DOF and with the DOF associated to the artificial boundary $\mathcal{B}$, respectively. Similarly for $\mathbf{V}_n$. Therefore, the matrix form of (21) is given by

$$\begin{aligned}\Big(\mathbf{M} + \frac{\Delta_t^2}{4}\mathbf{A}\Big)\mathbf{U}^{n+1} - \frac{\Delta_t^2}{4}\mathbf{Q}\mathbf{\Lambda}^{n+1} \;&=\; \Big(\mathbf{M} - \frac{\Delta_t^2}{4}\mathbf{A}\Big)\mathbf{U}^n + \frac{\Delta_t^2}{4}\mathbf{Q}\mathbf{\Lambda}^n + \Delta_t \mathbf{M}\mathbf{V}_n \\ &+ \frac{\Delta_t^2}{4}\Big(\mathbf{F}^{n+1} + \mathbf{F}^n\Big)\end{aligned} \tag{22}$$

where

$$\mathbf{A} = \begin{bmatrix} A_{II} & A_{I\mathcal{B}} \\ A_{\mathcal{B}I} & A_{\mathcal{B}\mathcal{B}} \end{bmatrix}, \qquad \mathbf{M} = \begin{bmatrix} M_{II} & M_{I\mathcal{B}} \\ M_{\mathcal{B}I} & M_{\mathcal{B}\mathcal{B}} \end{bmatrix}, \qquad \mathbf{Q} = \begin{bmatrix} Q_{I\mathcal{B}} \\ Q_{\mathcal{B}\mathcal{B}} \end{bmatrix}.$$

The matrix elements

$$M_{ij} = \int_\Omega N_i N_j, \quad A_{ij} = \int_\Omega \nabla N_i \cdot \nabla N_j, \quad i, j \in \mathcal{S}$$

are those of the mass and stiffness matrices, respectively, while those of Q are given by

$$Q_{ij} = \int_{\mathcal{B}} N_i N_j,\ i \in \mathcal{S}, j \in \mathcal{S}^{\mathcal{B}}.$$

Equation (22) is finally coupled with the discretized NRBC equation

$$\left(\frac{1}{2}\mathbf{I} + \mathbf{K}_0\right) \mathbf{U}_{\mathcal{B}}^{n+1} + \mathbf{V}_0 \mathbf{\Lambda}^{n+1} = -\sum_{j=0}^{n} \mathbf{K}_{n+1-j} \mathbf{U}_{\mathcal{B}}^{j} - \sum_{j=0}^{n} \mathbf{V}_{n+1-j} \mathbf{\Lambda}^{j}. \tag{23}$$

With abuse of notation, the shape functions $\{N_i\}$, used to define the matrices $\mathbf{V}_j$ and $\mathbf{K}_j$, denote the trace on $\mathcal{B}$ of the basis functions associated with the triangulation $\mathcal{T}_h$.

Remark 3.2 *When the boundary $\mathcal{B}$ is that of a circle, and the basis $\{N_i\}$ is associated with a uniform partition of it, then all the matrices $\mathbf{V}_j$ and $\mathbf{K}_j$, $j = 0, \cdots, N$ have a Toeplitz structure. Therefore we only need to construct and store the first row of each matrix. This property reduces significantly the computational cost of this ABC, as confirmed by the examples we have examined.*

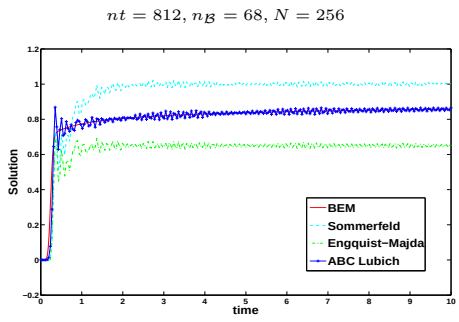
When we have to deal with a general boundary, or with a nonuniform partition, then the above property does not hold. Nevertheless, because of the behaviour of the coefficients $\omega_j^{\mathcal{J}}$ in the Lubich convolution quadrature formula, the matrices $\mathbf{V}_j$ and $\mathbf{K}_j$, $j = 0, \cdots, N$ can be approximated by corresponding very sparse matrices. This phenomenon is even more relevant in the 3D case (see for example [21]), where the corresponding Lubich coefficients $\omega_n(\Delta_t; r)$ represent a smooth approximation of delta Dirac functions (see [35], Figures 4 and 12). Thus, also in the case of a boundary $\mathcal{B}$ with no special properties, the computational cost of our NRBC can be drastically cut down. Of course, also in the Toeplitz case described above, the representative rows can be replaced by their sparse version, thus reducing further the computational cost and the storage.

Unfortunately, for the above numerical scheme, till now we have not been able to derive stability and convergence results. This appears to be a challenging and very hard task. We do not even have a proof for the consistency of the proposed NRBC discretization. This is due to the lack of estimates for the behaviors of the ω_n coefficients of the Lubich convolution quadrature, when the kernel of this latter is that of the wave equation. Because of this lack of theoretical results, we have performed an intensive numerical testing on these properties. In the next section we present a sample of the results we have obtained, to show the very good absorbing behavior of the proposed numerical scheme.

3.3 Numerical results

In this section, we present some examples of the numerical testing we have performed by using the approach discussed in the previous section. To measure the accuracy of the approximations we construct, we take a reference "exact" solution which is obtained by applying the Lubich-collocation method described in [9] with a very fine discretization. Once the density function is retrieved, the solution at any point in the infinite domain Ω^e is obtained by computing the associated potential (see [9] for details). Since in the first three examples the chosen artificial boundary $\mathcal{B}$ is a circle, we compare our

Figure 1: Example 1. Solution at a point $\mathbf{x} \in \mathcal{B}$ obtained with the Crank-Nicolson scheme.



solution with the classical (and cheap) first order Engquist-Majda NRBC (see [7]) and with the Sommerfeld NRBC (see [12]). In the last three examples we compare the solution we obtain using our NRBC with the "exact" solution defined above.

Example 1. As a first example, we apply our numerical scheme to the homogeneous case of Problem (1): the source f and the initial data u_0 and v_0 are zero throughout the infinite exterior domain Ω^e. The boundary Γ is the circle of radius $r = 0.25$, where we prescribe the Dirichlet condition $g(\mathbf{x}, t) = 1$ for all $t \geq 0$. We choose a circular artificial boundary with radius $R = 0.5$, so that Ω is the annulus bounded internally by Γ and externally by $\mathcal{B}$. Clearly, the solution of this problem is a radial function.

Notice that the Dirichlet datum does not satisfy the required compatibility condition. Nevertheless, since this is a typical test which has been considered in several papers on the topic (see [12]), to check the performance of our approach, we have applied it to this problem.

To apply the chosen NRBCs, we first rewrite the problem in polar coordinates. Then, for the space discretization we choose an unstructured triangular mesh of n_t triangles, having $n_{\mathcal{B}}$ equally spaced points on the boundary $\mathcal{B}$. Since $u(\mathbf{x}, 0) = 0$, while $g(\mathbf{x}, 0^+) = 1$, the use of the Crank-Nicolson scheme for the time integration generates, as shown in Figure 1, some unpleasant oscillations. To avoid them we use the (first order) implicit Euler scheme.

In Figure 2 we have plotted the behaviour of the approximate solution we have obtained at a point belonging to the boundary $\mathcal{B}$, $t \in [0, 10]$, for different choices of the time steps. It is compared with the exact reference solution, the first order Engquist-Majda and the Sommerfeld NRBC. It can be noticed that our NRBC converges to the exact solution as the time step decreases, while the Engquist-Majda and the Sommerfeld NRBC, as it is well known, give rise to approximations which underestimate and overestimate it, respectively.

In the next figures, whenever the exact NRBC (that will be denoted by "ABC Lubich") perfectly matches the exact reference solution, we use the same graphical sign for the two curves in order to avoid misreading.

Figure 2: Example 1. Solution obtained with the implicit Euler scheme (left column), and corresponding errors (right column).

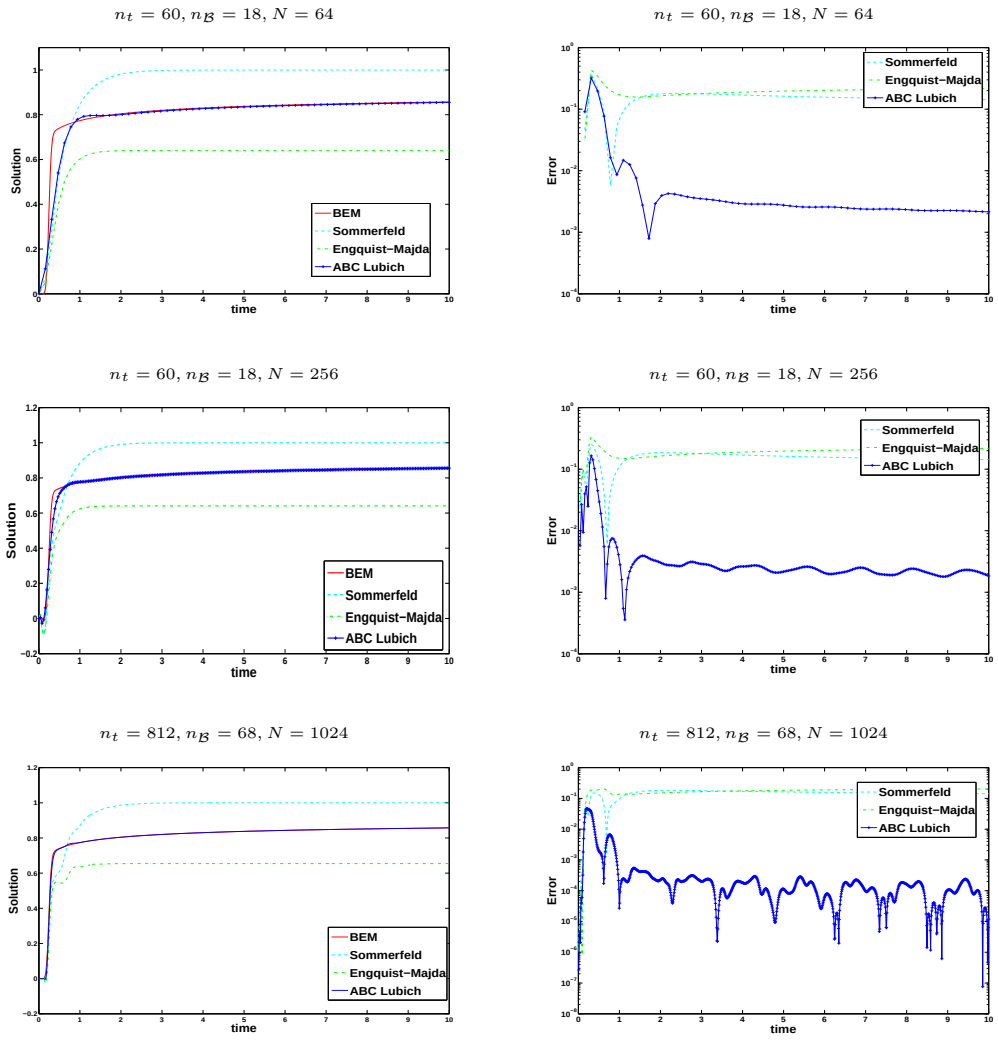
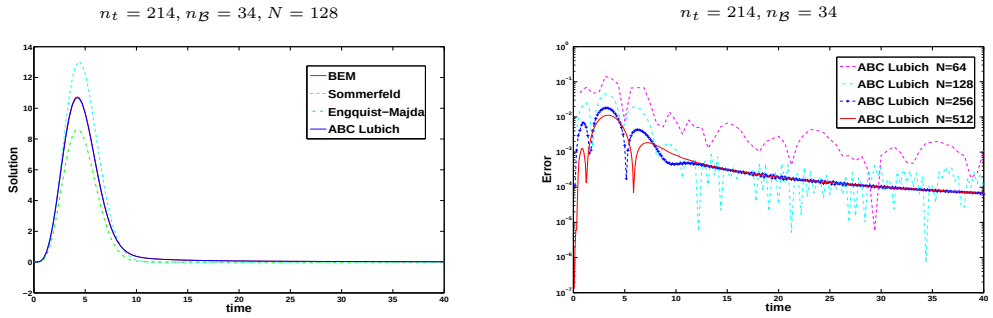


Figure 3: Example 2. Solution and corresponding errors.



Example 2. In the same setting of Example 1, we choose $g(\mathbf{x}, t) = t^3 \mathrm{e}^{-0.05(x_1^2+x_2^2-\sqrt{2}t)^2}$. With this choice, the data compatibility conditions are satisfied and the Crank-Nicolson scheme is applied. Notice that also in this case the solution is a radial function.

In Figure 3 we plot the behaviour of the solution at a mesh point $\mathbf{x} \in \mathcal{B}$, in the time interval $[0, 40]$. In particular, in the left plot we compare the approximants produced by the different NRBCs we have considered, taking $n_t = 214$ triangles in Ω and $N = 128$, with the exact solution. On the right plot we show the behaviour of the associated error, for some choices of the time discretization step.

Example 3. In this example we consider a non homogeneous problem, having zero source and zero initial velocity, but $u_0 \neq 0$, and $g = 0$ on Γ. The initial data is $u_0(x_1, x_2) = \mathrm{e}^{-5((x_1-5)^2+x_2^2)}$. Although u_0 does not have a local support (and thus contradicts one of our assumptions), it decays exponentially fast away from its center $\mathbf{x} = (5, 0)$, in such a way that from the computational point of view it can be regarded as compact and supported in a disk with radius smaller than 3 (at distance 2.7 from its center it assumes approximately values of the order 10^{-16}). The boundary Γ is a circle of radius $r_0 = 2$, and the artificial boundary is a circle of radius $R = 8$, so that the support of u_0 is included in Ω (see Figure 4, left plot). The disk bounded by Γ represents a soft obstacle that acts as a reflecting body. With these choices, the data compatibility conditions are satisfied and the Crank-Nicolson scheme can be applied.

In Figure 4, right plot, we show the behaviour of the solution at $\mathbf{x} = (8, 0)$ and for $t \in [0, 20]$. We note that the solution is zero until the initial data reaches the artificial boundary around $t = 2$. Approximatively at the same time, u_0 also reaches the reflecting boundary Γ and is reflected, so that around $t = 9$ we see another outgoing wave at the artificial boundary $\mathcal{B}$. After that time, the wave is completely out of the annulus, as the exact solution and the approximate solution with the exact NRBC show, while the other approximated NRBCs show spurious waves at $t = 14$. This phenomenon is due to the fact that at $t = 2$ the artificial boundary is not completely transparent for the Enquist-Majda and the Sommerfeld NRBC. Indeed these have generated incoming waves that, after reaching Γ again, have been reflected back. In Figure (5) we show some snapshots of the solution obtained with our exact NRBC at different time steps.

Figure 4: Example 3. Domain triangulation and solution at $(8, 0)$.

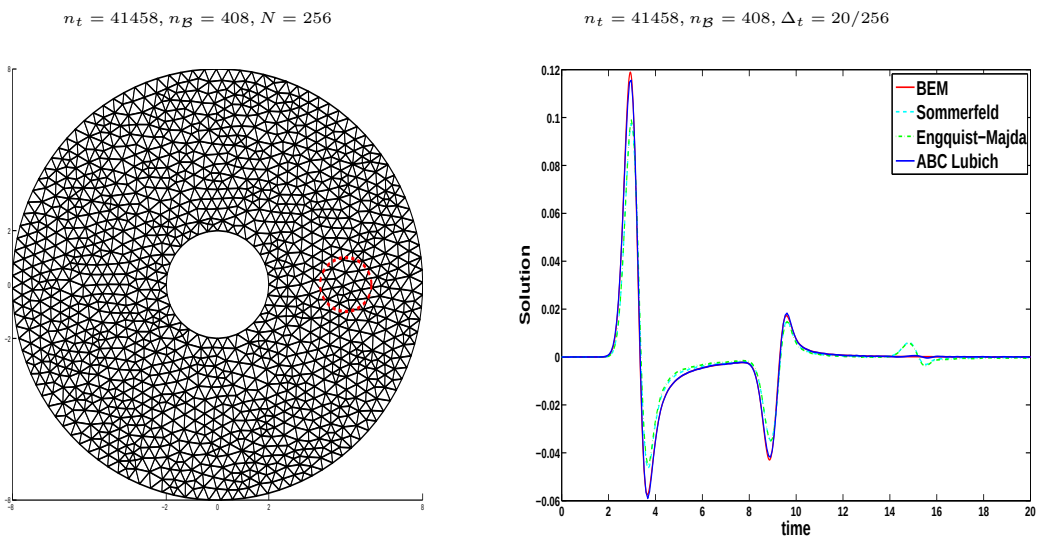


Figure 5: Example 3. Snapshots of the solution at different times.

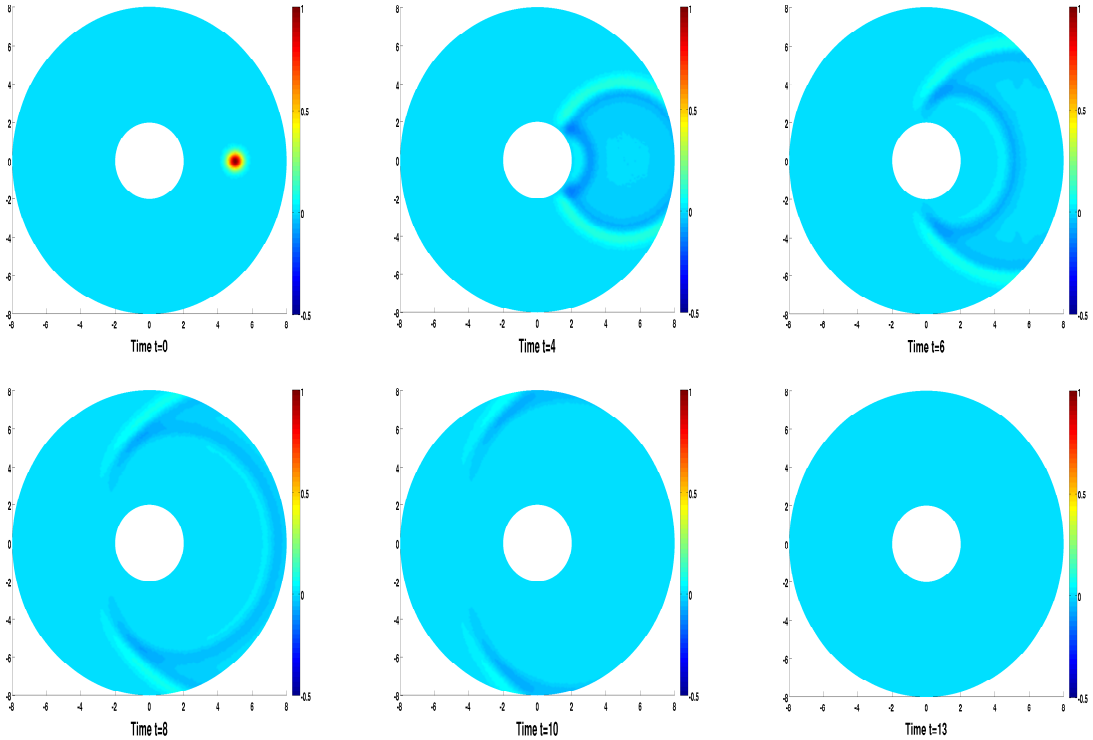


Figure 6: Example 4. Left figure: $n_t = 784$ triangles for the mesh of Ω and the mesh point $\mathbf{x} \approx (4, 0) \in \mathcal{B}$ where the solution is evaluated (bullet); right figure: the behaviours of the exact and approximate solutions at $\mathbf{x}$.

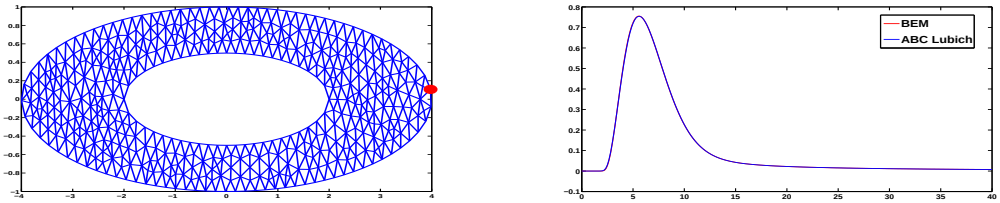
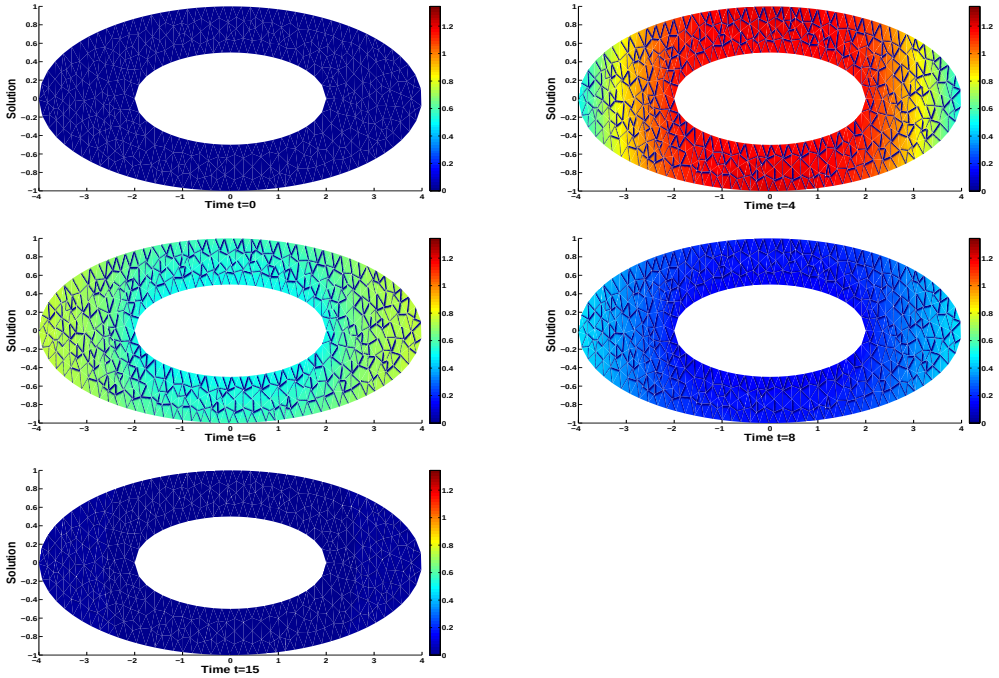


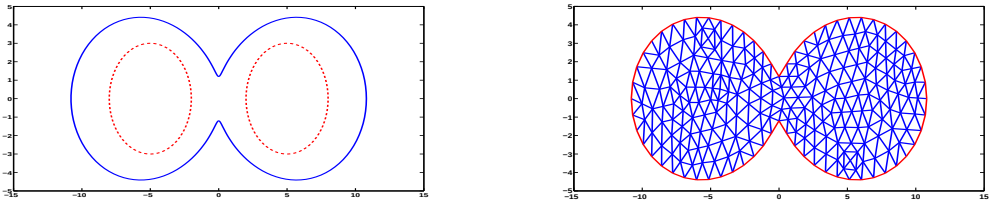
Figure 7: Example 4. Snapshots of the solution at different times.



Example 4. For some geometries of the physical domain boundary Γ, or of the domain of interest Ω, the choice of a circular artificial boundary $\mathcal{B}$ can be wasteful both from the computational and space memory point of view. In this example, we choose Γ as the ellipse centered at the origin having equation $x_1^2/a^2 + x_2^2/b^2 = 1$, with $a = 2$ and $b = 0.5$. A natural choice of $\mathcal{B}$ could be, for example, the ellipse $x_1^2/a_1^2 + x_2^2/b_1^2 = 1$, where for example we choose $a_1 = 4$ and $b_1 = 1$. We consider the homogeneous problem, with Dirichlet data $g(\mathbf{x}, t) = t^3 \mathrm{e}^{-t}$. In Figure 6 we compare the solution obtained with the exact NRBC and the exact solution at a mesh point $\mathbf{x} \approx (4, 0)$, for $t \in [0, 40]$, taking $n_t = 784$ triangles and $N = 256$. In Figure 7 we show the snapshots of the solution at different times.

Notice that in this case, due to the domain symmetries, only the first $\lceil M/4 \rceil$ rows of the matrices $\mathbf{V}_n, \mathbf{K}_n$ need to be explicitly constructed.

Figure 8: Example 5. Left figure: the nut shape artificial boundary (solid line) and the local support of the initial data u_0 (dashed line). Right figure: a nut shaped domain triangulation.



Example 5. To show the feasibility in the choice of the geometry of the artificial boundary, we apply the proposed scheme to the following problem: we consider the wave equation in the whole $\mathbb{R}^2$, with zero source f and zero initial velocity v_0. The initial data $u_0(x_1, x_2) = \mathrm{e}^{-5((x_1-5)^2+x_2^2)} + \mathrm{e}^{-5((x_1+5)^2+x_2^2)}$ is a function with two humps, one centered at $(5, 0)$ and one centered at $(-5, 0)$, both having a "practical" local support included in a circle of radius smaller than 3 and height equals to 1 (notice that u_0 is not compactly supported, but the same considerations we made in Example 3 apply here too).

Supposing we are interested in knowing the solution in a neighborhood of the center of the humps, we choose as artificial boundary a nut shape curve whose parametric equation is given by

$$\begin{aligned} x_1 &= \rho\cos(\theta), \\ x_2 &= \rho\sin(\theta), \end{aligned}$$

where $\rho = c(1 + e\cos(n\theta))$, with $c = 6, e = 0.8, n = 2$ (see Figure 8).

In Figure 9 we show the snapshots of the solution at different times, obtained with a triangulation of $n_t = 14337$ triangles, and $N = 300$ subdivisions of the time interval $[0, 30]$.

Example 6. In the final example, we aim at simulating those situations where one is interested in knowing the solution at points that are away from sources. For simplicity we suppose $u_0 = 0$, $v_0 = 0$, and $f \neq 0$. In this case, the artificial boundary $\mathcal{B}$ is chosen in such a way that the source f is locally supported in the residual domain $\mathcal{D}$, while it is zero in Ω. Therefore, the artificial boundary condition reads:

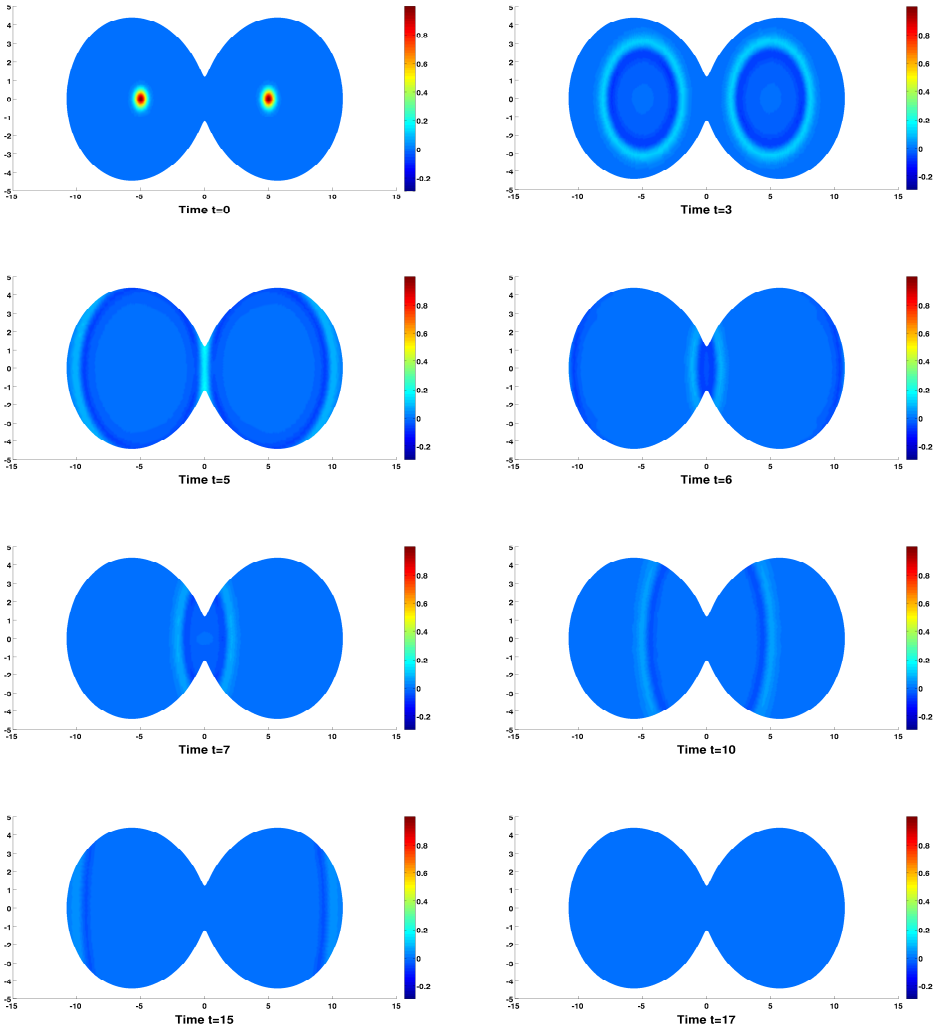
$$\frac{1}{2}u(\mathbf{x}, t) + \mathcal{V}\lambda(\mathbf{x}, t) + \mathcal{K}u(\mathbf{x}, t) = I_f(\mathbf{x}, t) \quad \text{in} \quad \mathcal{B} \times (0, T].$$

In particular, we consider a source concentrated at a point $\mathbf{x}_0$: $f(\mathbf{x}, t) = h(t)\delta(\mathbf{x} - \mathbf{x}_0)$, where $h(t)$ is a given smooth function. With this choice, the volume integral I_f (see (4)) has the following simpler form:

$$I_f(\mathbf{x}, t) = \int_0^t h(\tau)G(\mathbf{x} - \mathbf{x}_0, t - \tau)d\tau.$$

For the computation of the volume integral I_f, we apply the Lubich convolution technique. It is beyond the scope of the paper to enter into the details of the numerical

Figure 9: Example 5. Snapshots of the solution at different times.



evaluation of the volume terms, even for more general sources; this will be the subject of a future work.

We compare the solution obtained with the above mentioned approach and the usual one, which consists of including the source into the finite computational domain Ω. We recall that, if the source is far from the area of interest, the last approach would require a much larger domain Ω, thus wasting computational time and space memory. In our test we place the source f at $\mathbf{x}_0 = (4, 0)$; Γ and $\mathcal{B}$ are the circles of radius $r_0 = 1$ and R, respectively, both centered at the origin. We choose first $R = 2$ and then $R = 5$. In the first case, f is external to the finite computational domain, while in the second case it is included in the annulus bounded by Γ and $\mathcal{B}$.

We analyze three different cases, according to the choice of the function h:

$$h(t) = \begin{cases} 10^3 t^4 \mathrm{e}^{-t} & (a) \\ 10^3 t^4 \mathrm{e}^{-t} \sin(5t) & (b) \\ \sin(5t) & (c). \end{cases}$$

In each case, we compare the solution obtained at the mesh point $\mathbf{x}_{P_1} = (1.9995, 0.0436)$ for the external source, and at $\mathbf{x}_{P_2} = (2.0758, -0.0154)$ for the internal source, with $t \in [0, T]$. Since the finite domain is different in the above mentioned cases, the mesh points where we evaluate the solution are not exactly the same, so we do not overlap the two plots, but we compare them qualitatively. In particular, we plot the solutions obtained in the case $R = 5$ (left) and $R = 2$ (right). The case (a), with $T = 20$, is shown in Figure 10, while the case (b), with $T = 40$, in Figure 11. In both cases the wave is evanescent because f represents a vanishing source. On the contrary, in the case (c), where we have taken $T = 40$, the wave has a periodic constant oscillatory behaviour (see Figure 12).

Finally, in Figure 13 we plot the approximate solution we have obtained at $\mathbf{x}_{P_1}$ when the source (c) is located at $\mathbf{x}_0 = (10, 0)$ and $R = 2$. The result obtained for $R = 12$, in which case $\mathbf{x}_0$ belongs to the (bounded) computational domain Ω, requires a much finer time integration step, hence a very high CPU time, due to the oscillating behaviour of the source. Because of this, we have omitted it.

We have also performed the same tests by replacing $\sin(5t)$ with $\sin(100t)$ in the expression of h. We have obtained very good results when $\mathbf{x}_0$ is external to Ω while, due to the very high oscillatory behaviour of the source, the case of the internal source would have required a very fine time interval partition, that we could not perform in a reasonable time.

The last tests show, in particular, that it is more efficient to include the source term f in the I_f term of our NRBC, than to have to treat it as the right hand side of the wave equation.

Figure 10: Example 6. The internal source case (left), the external source case (right) for $h(t) = 10^3 t^4 \mathrm{e}^{-t}$, $\mathbf{x}_0 = (4, 0)$.

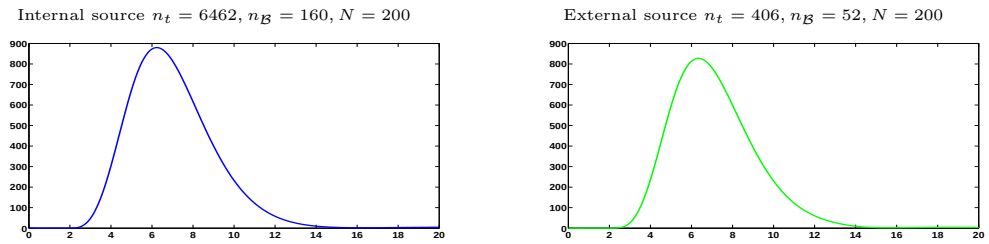


Figure 11: Example 6. The internal source case (left), the external source case (right) for $h(t) = 10^3 t^4 \mathrm{e}^{-t} \sin(5t)$, $\mathbf{x}_0 = (4, 0)$.

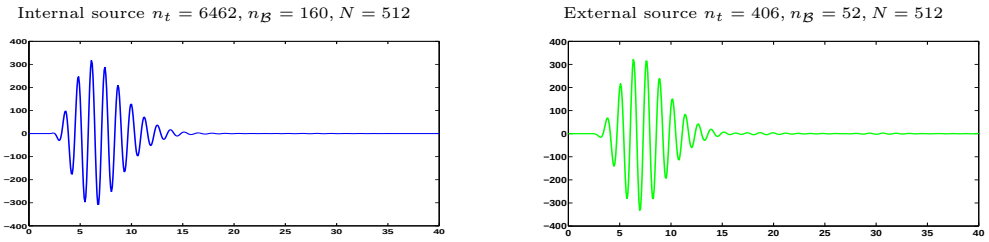


Figure 12: Example 6. The internal source case (left), the external source case (right) for $h(t) = \sin(5t)$, $\mathbf{x}_0 = (4, 0)$.

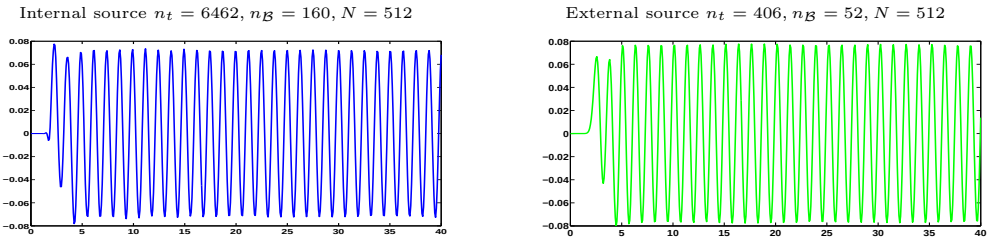
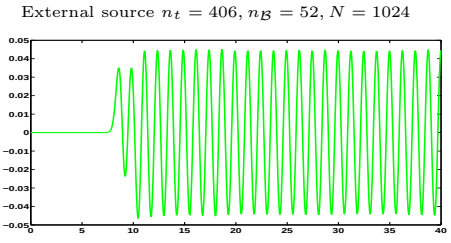


Figure 13: Example 6. The external source case for $h(t) = \sin(5t)$, $\mathbf{x}_0 = (10, 0)$.



Remark 3.3 *When we refine the discretization associated with the basic methods, chosen to solve the PDE problem on the Ω domain, to improve their accuracies the discretization of the proposed NRBC will also be (simultaneously) refined, assuming that integrals have been evaluated with sufficiently high accuracy. Thus, from this point of view, the NRBC is not an approximate one, such as those constructed in [6], [17], [22] and [23], where the truncation error of the ABC does not decrease, unless the number of terms defining the latter is increased, and the associated parameters are selected in an appropriate way, to reduce the spurious reflection below the desired level. This overhead is by no means negligible.*

3.4 Some remarks on the sparsity of the single and double layer operators

As already pointed out in the previous sections, the matrices $\mathbf{V}_j$ and $\mathbf{K}_j$, $j = 0, \cdots, N$ have many elements which are very small or even negligible. Therefore these matrices can be approximated by corresponding sparse ones. This property allows in principle to reduce the space memory storage and to speed up the computation. We show the structure of $\mathbf{V}_j$ and $\mathbf{K}_j$ in the setting of some of the numerical examples showed before. In the context of Examples 1 and 3, where $\mathcal{B}$ is the circle of radius $R = 0.5$ and $R = 8$ respectively, the choice of a uniform partition on $\mathcal{B}$ allows to take advantage of the Toeplitz structure of the matrices $\mathbf{V}_j$ and $\mathbf{K}_j$ for each time step $j = 0, \cdots, N$. Therefore, we just construct and store the first row of each matrix. In Figures (14) and (15) we show the semilogarithmic plot of the values of this row with respect to the number of nodes on $\mathcal{B}$ and for different times, for the case $R = 0.5$ and $R = 8$. In particular, we plot the matrices $\mathbf{V}_j$ and $\mathbf{K}_j$ for $j = 0, N/2, N$ corresponding to the time steps $t = 0, T/2, T$ for different choices of T. We note that, for the time step $t = 0$ both $\mathbf{V}_0$ and $\mathbf{K}_0$ have few non negligible elements; these two matrices are the only ones involved in the final linear system that needs to be solved (see equation (23)). On the contrary, the sparsity of the remaining matrices reduces for large values of j. We recall that these matrices give a contribution to the right hand side term, namely they multiply the (known) solutions at the previous time steps.

For the case of the more general nut shape of Example 5, we have performed the same analysis but, since the matrices do not satisfy the Toeplitz structure anymore, we have first computed all of them and we have then cut all the elements which are below the threshold parameter $\varepsilon = 1e-06$. The number of the remaining elements is denoted by nz. In Figures 16, 17 we show their spy plot for different times. In both cases, the solutions obtained with the choice of $\varepsilon = 1e-06$ are the same as the ones obtained by retaining all the elements of the matrices.

For simplicity, we have taken the same value of the above threshold parameter for all the $\mathbf{V}_j$ and $\mathbf{K}_j$ matrices. Further investigation for a possible more efficient choice of it is however needed. Also the use of special algorithms that take into account the sparsity property of the matrices is an argument that has not been treated in this work, but which is worthwhile to be analyzed.

In the testing we have performed, the non reflecting property of our NRBC has shown to be very robust with respect to the above element cut strategy.

Figure 14: Circular boundary $\mathcal{B}$: $R = 0.5$, $n_{\mathcal{B}} = 64$. The first row of the single layer matrices $\mathbf{V}_j$ (top row), and of the double layer matrices $\mathbf{K}_j$ (bottom row).

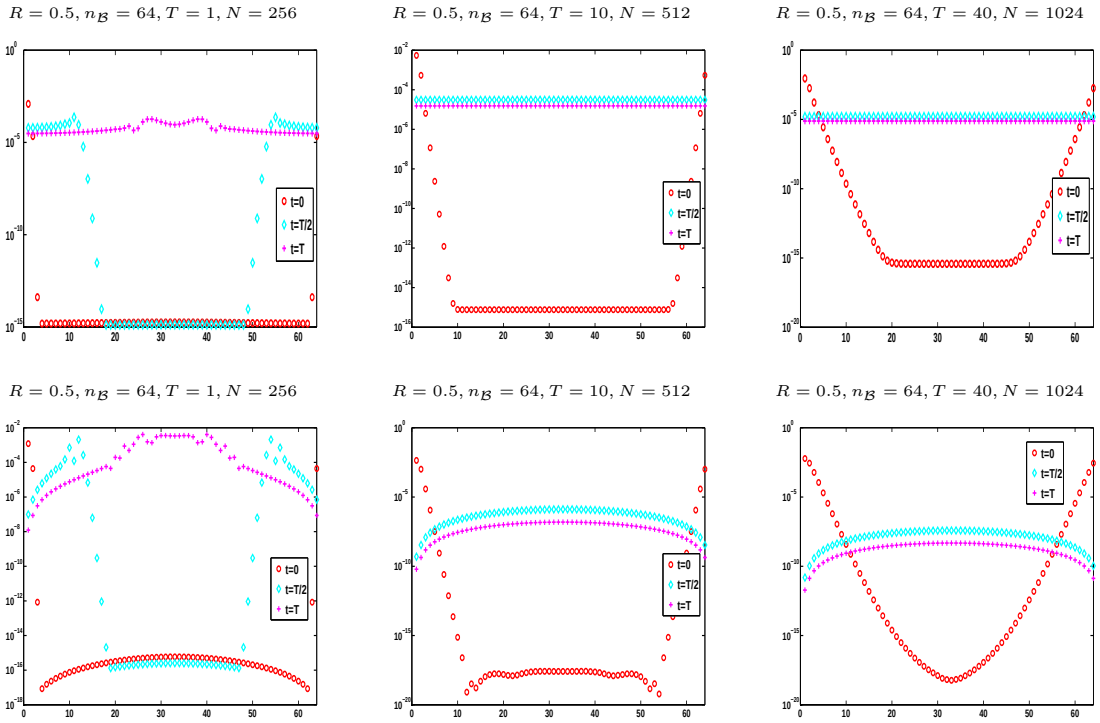


Figure 15: Circular boundary $\mathcal{B}$: $R = 8$, $n_{\mathcal{B}} = 64$. The first row of the single layer matrices $\mathbf{V}_j$ (top row), and of the double layer matrices $\mathbf{K}_j$ (bottom row).

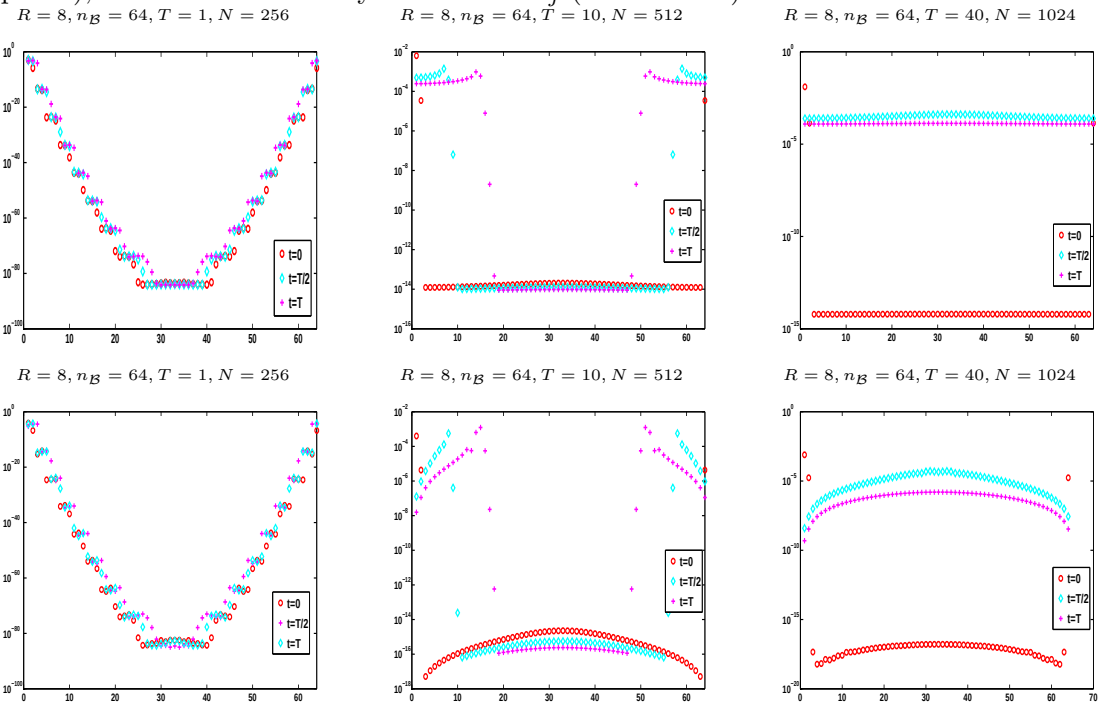


Figure 16: Nut shape boundary $\mathcal{B}$: $n_{\mathcal{B}} = 32$. The single layer matrices $\mathbf{V}_j$ (top), the double layer matrices $\mathbf{K}_j$ (bottom), $T = 1$.

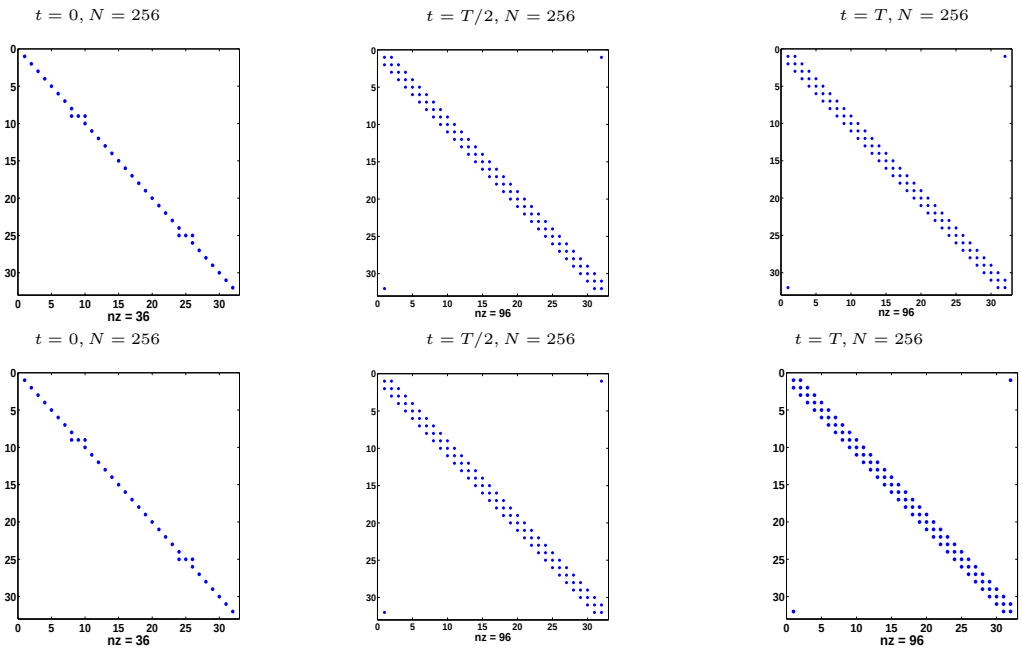
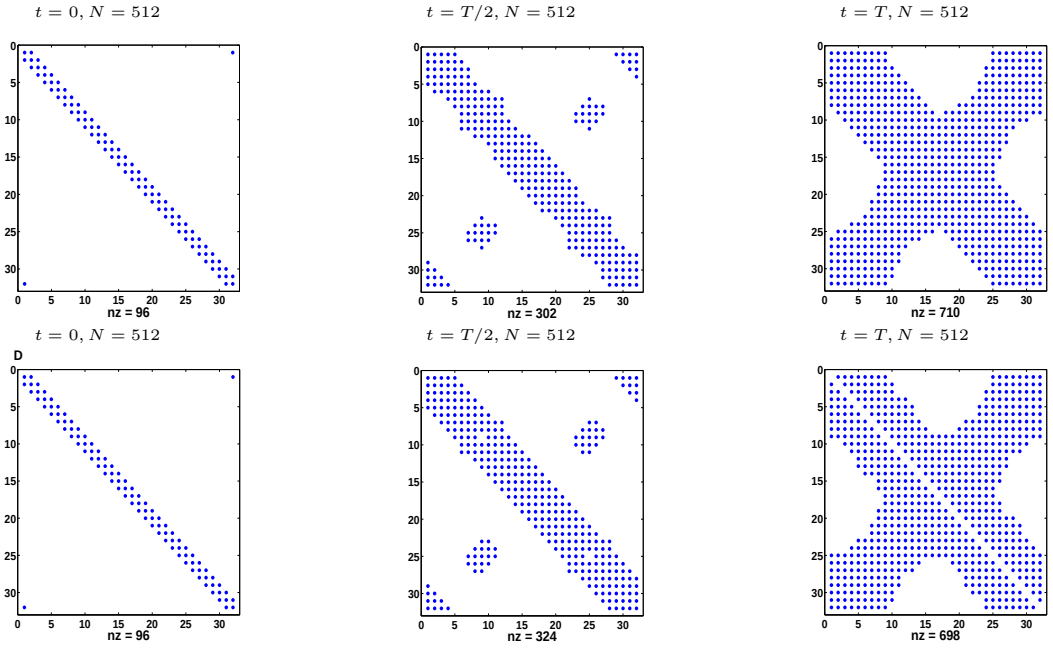


Figure 17: Nut shape boundary $\mathcal{B}$: $n_{\mathcal{B}} = 32$. The single layer matrices $\mathbf{V}_j$ (top), the double layer matrices $\mathbf{K}_j$ (bottom), $T = 10$.



4 Reduced problem discretization. The Finite Difference Method

Although our main goal is the coupling of the NRBC with finite element methods, to further test the performance of our NRBC, in this last section we apply it also to a problem where a second order finite different scheme is used for the PDE space discretization. To fix the ideas, we take Γ as the circle of radius r and $\mathcal{B}$ as the circle of radius $R > r$. By using the polar coordinates $x = \rho\cos(\theta)$, $y = \rho\sin(\theta)$, $\rho \in [r, R]$, $\theta \in [0, 2\pi]$, we transform the bounded computational domain Ω into the rectangle $[r, R] \times [0, 2\pi]$. For simplicity, we consider the homogeneous problem, with $f = 0$ and zero initial conditions, so that in polar coordinates the wave equation takes the following well known form:

$$u_{tt} - (u_{\rho\rho} + \frac{1}{\rho^2}u_{\theta\theta} + \frac{1}{\rho}u_\rho) = 0.$$

We consider a discretization of Ω into cells, obtained by a uniform partition of $[r, R]$ and $[0, 2\pi]$ into M_ρ and M_θ subintervals, respectively. Such a choice determines the set of nodes (ρ_i, θ_j), $i = 0, \cdots, M_\rho$, $j = 0, \cdots, M_\theta$, where $\rho_i = r + ih_\rho$, $h_\rho = (R - r)/M_\rho$ and $\theta_j = jh_\theta$, $h_\theta = 2\pi/M_\theta$. Then we apply the Crank-Nicolson scheme for the time integration, and the classical second order finite difference scheme for the Laplace operator. Without entering into the details of the implementation, we remark that we use the periodicity property at $\theta = 2\pi$. Moreover, since we collocate the equation also at the nodes at $\rho = R$, where the solution is not known, to maintain the second order accuracy, we introduce the fictitious nodes $(\rho_{M_\rho+1}, \theta_j)$, $j = 0, \cdots, M_\theta - 1$. The unknowns, at the time level $t_n = n\Delta_t$, are the values u_{ij}^n of the solution at the grid points (ρ_i, θ_j), $i = 1, \cdots, M_\rho$, $j = 0, \cdots, M_\theta - 1$, and the values λ_j^n of the normal derivative of u at $\rho = R$, for $j = 0, \cdots, M_\theta - 1$. The fictitious nodes are then eliminated by approximating the normal derivative λ_j^n with the symmetric finite difference formula

$$\lambda_j^n = \frac{u_{M_\rho+1,j}^n - u_{M_\rho-1,j}^n}{h_\rho} + O(h_\rho^2).$$

The final scheme is obtained by coupling the above Crank-Nicolson - finite difference method with our NRBC. The latter is approximated by the Lubich-collocation scheme described in Section 3.1.

In the numerical test we have performed below, we compare this scheme with the classical first (I) and second order (II) Engquist-Majda and with the Sommerfeld NRBC. In particular we choose $r = 0.25$, $R = 0.5$, $T = 10$ and the Dirichlet data $g(\mathbf{x}, t) = t^3 \mathrm{e}^{-1/2(x_1^2+x_2^2-\sqrt{2}t)^2}$.

We have applied the above finite difference scheme also to the previous Examples 1-3, that have been solved by means of the finite element method. The results we have obtained are comparable.

In Figure 18 we show the behavior of the solution at a mesh point $\mathbf{x}_P \in \mathcal{B}$, and we see again that our exact NRBC matches the right solution, while, as expected, the approximate NRBCs underestimate/overestimate it. As already pointed out, since the mesh on $\mathcal{B}$ is uniform, we take advantage of the Toeplitz structure of the single and double layer matrices $\mathbf{V}_j$ and $\mathbf{K}_j$ for each time step $j = 0, \cdots, N$. In this case, the computational cost for the approximation of the exact NRBC is significantly reduced.

To have an idea of the computational cost due to our NRBC, we compare, from this point of view, the latter with the second order Engquist-Majda NRBC. Thus, in Tables 1, we report the ratio R_{CPU} between the CPU time that in our testing the exact NRBC and the second order Engquist-Majda NRBC approaches has required to solve the PDE problem, for different choices of the discretization parameters M_ρ, M_θ and N. In particular, in Table 1 a) we choose $M_\rho = M_\theta = N$. Note that R_{CPU} gets closer to 2 as the values of the parameters increase, namely, when the discretizations of the domain Ω and of the time interval $[0, T]$ are (simultaneously) refined. A similar comment applies to Table 1 b), where we fix the time step size $\Delta_t = T/16$ and decrease the mesh size h of the space discretization. In this case R_{CPU} approaches 1.

In Table 1 c) we fix the finite difference mesh, by choosing for example $M_\rho = 16, M_\theta = 16$, and refine only the time step by repeatedly doubling N. The value of the ratio R_{CPU} is higher, although it increases very mildly; this is due to the overhead required for updating the right hand side of the final linear system produced by our method. We remark however that in the case of a non trivial problem, the values $M_\rho = 16, M_\theta = 16$ are fairly small, and in general, to increase the accuracy, one has to increase simultaneously all the parameters M_ρ, M_θ and N.

Figure 18: Finite differences. Solution and corresponding errors at $\mathbf{x}_P$.

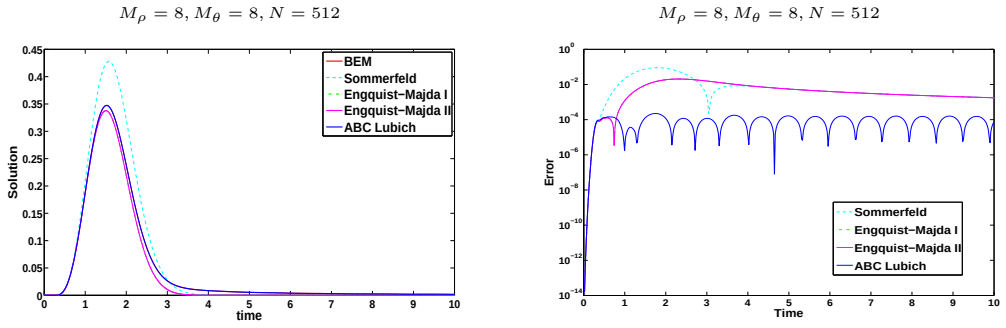
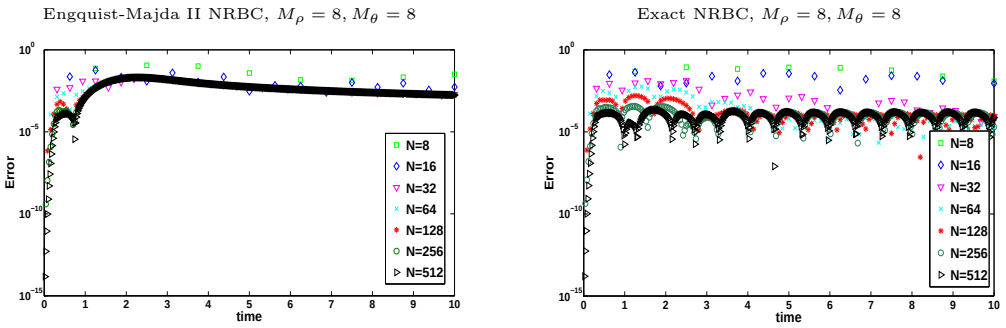


Figure 19: Finite differences. Error behavior at $\mathbf{x}_P$.



Finally, to check the order of accuracy of the proposed numerical scheme, with reference to the case associated with Table 1 a) we have also computed the corresponding space absolute error ℓ^2-norm, and the estimated order of convergence (EOC), at some

Table 1: Ratio R_{CPU} between the CPU time of the exact and the second order Engquist-Majda NRBCs.

a): $M_\rho = M_\theta = N$.

M_ρ	R_{CPU}
8	3.58
16	4.63
32	4.47
64	3.16
128	2.73
256	2.70

b): $M_\rho = M_\theta, N = 16$.

M_ρ	R_{CPU}
8	5.39
16	5.41
32	4.37
64	2.60
128	1.49
256	1.01

c): $M_\rho = M_\theta = 16$.

N	R_{CPU}
8	4.79
16	4.63
32	5.58
64	6.29
128	7.01
256	8.08

Table 2: ℓ^2-norm absolute error and convergence order for $M_\rho = M_\theta = N$, $T = 10$

M_ρ	$T/8$		$T/4$		$T/2$		T	
8	$5.22E-02$		$8.95E-02$		$1.07E-01$		$2.37E-02$	
		0.45		1.37		1.41		0.15
16	$3.83E-02$		$3.45E-02$		$4.04E-02$		$2.14E-02$	
		3.22		1.54		5.86		6.20
32	$4.12E-03$		$1.19E-02$		$6.94E-04$		$2.91E-04$	
		−0.05		2.45		3.39		3.52
64	$4.27E-03$		$2.17E-03$		$6.63E-05$		$2.53E-05$	
		1.75		2.38		3.03		2.10
128	$1.27E-03$		$4.18E-04$		$8.15E-06$		$5.91E-06$	
		2.07		2.21		0.88		3.42
256	$3.03E-04$		$9.05E-05$		$4.44E-06$		$5.51E-07$	

time instants. The results we have obtained are reported in Table 2. As reference solution, we have taken that obtained by using the space-time BEM described in [9], with a sufficiently fine discretization, to guarantee about 6 significant digits.

Although we do not have an error estimate for our (global) numerical scheme, we expect a rate of convergence of order 2. We recall that for a similar finite difference scheme for a classical Dirichlet problem, the standard error estimate is usually uniform with respect to the time variable. Therefore, it is the behavior of the maximum error that defines the convergence order of the method. The results of Table 2 seem to confirm that the introduction of our discretized NRBC, which is itself a second order scheme (see [9]), maintains the order 2.

We have also compared the CPU time required by the finite difference scheme associated with our NRBC and with the second order Engquist-Majda ABC, when the time integration is performed by using an explicit (conditionally stable) formula. We remark that in this case, we can take advantage of the explicitness of this formula, so that in our approach the only system we have to solve, at each time step, is of the following form:

$$\mathbf{V}_0 \boldsymbol{\lambda}^n = \mathbf{b}^n \tag{24}$$

with $\mathbf{b}^n$ known. Note that to reduce the computational cost of this latter, we can approximate, as suggested by Fig. 14-17, the matrix $\mathbf{V}_0$ by a very sparse version of it, which is almost banded with a very small bandwidth. When the time stepsize is small, this latter can even be almost diagonal.

In this case, besides being forced to choose the discretization parameters properly,

to guarantee the method stability, the R_{CPU} ratio is very similar to that we have when the Crank-Nicolson time integrator is used. Note that for our approach, the required CPU time can be further reduced by taking advantage of the sparsity of the approximated versions of the matrices $\mathbf{K}_j, \mathbf{V}_j$, which define the above known term $\mathbf{b}^n$.

5 Conclusions

Boundary integral equation formulations for elliptic PDE problems are nowadays a standard tool for solving some of these problems. They have also been interpreted as a relationship between the (unknown) solution and its normal derivative, which has then been used to define exact NRBC, to be associated with finite difference or finite element numerical schemes. Their analogues for time-dependent problems have not received the same attention, and their development is still at an early stage.

In these last years a few papers have appeared, to solve exterior problems for the wave equation by means of space-time BIE. A common feature of these papers is the use of special convolution quadratures developed by Ch. Lubich in the late 80's, to discretize the time integral. These are then coupled with a standard Galerkin or collocation method for the space discretization. Only for the Galerkin case theoretical results have been derived.

In this paper we have proposed to use such space-time BIE to define, as in the case of the above mentioned elliptic problems, a NRBC for the solution of 2D exterior wave equation problems, in a bounded domain of interest, by means of standard FE or FD methods.

To discretize this BIE, we have coupled a second order Lubich rule with a space collocation method defined by a continuous piecewise linear approximant. The proposed NRBC has the following main features:

- it is of exact type;

- although it is non local in time and space, from the computational complexity point of view, the proposed discretization of it is almost local with respect to both variables. This property follows from the special properties of the coefficients of the Lubich rule;

- it allows to take smooth artificial boundaries of arbitrary shape. Moreover, some numerical testing we have performed seems to confirm that its good non reflecting property is maintained even when the chosen boundary has corners.

- it is transparent for both outgoing and incoming waves, and it naturally allows the treatment of non homogeneous data. Far field sources do not have to be necessarily included in the finite computational domain;

- we have applied it to multi-scattering problems and obtained results very similar to those reported in Section 4.

- higher order Lubich convolution quadratures exists, which have already been used to solve wave equation problems (see [2]). Thus, their coupling with higher order finite element or finite difference schemes is possible. Moreover, very recently, a first attempt to construct a one-step/variable step method, obtained by coupling a Lubich quadrature with a Galerkin (space) method, has been proposed and studied (see [27]) to solve the space time BIE associated with the wave problem;

- its generalization to 3D problems is straightforward (see [9]).

Unfortunately, for the Lubich-collocation approach no theoretical results have been till now derived. This seems to be a very hard and challenging task. Therefore, nothing is theoretically known about the properties of the coupling of our NRBC with the chosen FE or FD methods. Because of this, to verify the main properties of the proposed numerical schemes, we have performed an intensive numerical testing, which seems to confirm the efficiency of the proposed approach. Further investigation is however still needed.

All the numerical computation has been performed on a PC with Intel Core2® Quad Q6600 (2.40GHz). To perform our numerical testing we have written standard (i.e., sequential) Matlab® codes.

References

[1] B. Alpert, L. Greengard, and T. Hagstrom. Rapid evaluation of nonreflecting boundary kernels for time-domain wave propagation. *SIAM J. Num. Anal*, 37:1138–1164, 2000.

[2] L. Banjai, M. Messner, and M. Schanz. Runge-kutta convolution quadrature for the boundary element method. *Comput. Methods Appl. Mech. Engrg.*, 245-246:90–101, 2012.

[3] A. Bayliss and E. Turkel. Radiation boundary conditions for wave-like equations. *Comm. Pure Appl. Math.*, 33:707–725, 1980.

[4] G. Ben-Porat and D. Givoli. Solution of unbounded domain problems using elliptic artificial boundaries. *Commun. Num. Meth. Engng*, 11:735–741, 1995.

[5] R. Clayton and B. Engquist. Absorbing boundary conditions for acoustic and elastic wave equation. *Bulletin of the Seismological Society of America*, 67(6):1529–1540, 1977.

[6] F. Collino. High order absorbing boundary conditions for wave propagation models. Straight line boundary and corner cases. In *Proc. 2nd Int. Conf. on Mathematics and Numerical Aspects of Wave Propagation*, pages 161–171. SIAM, 1993.

[7] B. Engquist and A. Majda. Absorbing boundary conditions for numerical simulation of waves. *Math. Comput.*, 31(139):629–651, 1977.

[8] S. Falletta and B.P.Lamichhane. Mortar finite elements for a heat transfer problem on sliding meshes. *Calcolo*, 46(2):131148, 2009.

[9] S. Falletta, G. Monegato, and L. Scuderi. A space-time BIE method for nonhomogeneous exterior wave equation problems. The Dirichlet case. *IMA J. Numer. Anal.*, 32(1):202–226, 2012.

[10] D. Givoli. A combined analytic-Finite Element Method for elastic shells. *Int.J. of Solids and Structures*, 26:185–198, 1990.

[11] D. Givoli. Finite element analysis of long cylindrical shells. *AIAA Journal*, 28:1331–1333, 1990.

[12] D. Givoli. *Numerical Methods for Problems in Infinite Domains*. Elsevier, 1992.

[13] D. Givoli. Recent advances in the DtN FE method. *Archives of Computational Methods in Engineering*, 6(2):71–116, 1999.

[14] D. Givoli. High-order local non-reflecting boundary conditions: A review. *Wave Motion*, 39:319–326, 2004.

[15] D. Givoli and J.B. Keller. Exact non-reflecting boundary conditions. *J. Comp.Phys.*, 82:172–192, 1989.

[16] D. Givoli and J.B. Keller. A Finite Element Method for large domains. *Comp. Meth. Appl. Mech. Engng.*, 76:41–66, 1989.

[17] D. Givoli and B. Neta. High-order non-reflecting boundary scheme for time dependent waves. *J. Comput. Phys.*, 186(1):24–46, 2003.

[18] D. Givoli and S. Vigdergauz. Artificial boundary conditions for 2D problems in geophysics. *Comput. Meth. Appl. Mech. Engng.*, 110:87–101, 1993.

[19] D. Goldman and P.E. Barbone. Dirichlet to Neumann maps for the representation of equipment with weak nonlinearities. In *ASME Noise control and acoustic division*, volume 22, pages 71–76, 1996.

[20] I.S. Gradshteyn and I.M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, 2007.

[21] W. Hackbusch, W. Kress, and S. Sauter. Sparse convolution quadrature for time domain boundary integral formulations of the wave equation. *IMA J. Numer. Anal.*, 29:158–179, 2009.

[22] T. Hagstrom and T.Warburton. A formulation of asymptotic and exact boundary conditions using local operators. *Appl. Numer. Math.*, 27(4):403–416, 1998.

[23] T. Hagstrom and T.Warburton. A new auxiliary variable formulation of high-order local radiation boundary conditions: corner compatibility conditions and extensions to first-order systems. *Wave Motion*, 39(4):327–338, 2004.

[24] R.L. Higdon. Absorbing boundary conditions for difference approximations to the multidimensional wave equation. *Math. Comp.*, 47(176):437–459, 1986.

[25] R.L. Higdon. Numerical absorbing boundary conditions for the wave equation. *Math. Comp.*, 49(179):65–90, 1987.

[26] A.R. Laliena and F-J. Sayas. Theoretical aspects of the application of convolution quadrature to scattering of acoustic waves. *Num. Math.*, 112:637–678, 2009.

[27] M. Lopez-Fernandez and S. Sauter. Generalized convolution quadratures with variable time stepping. *IMA J. Numer. Anal.*, doi: 10.1093/imanum/drs034, 2013.

[28] C. Lubich and A. Schädle. Fast convolution for non-reflecting boundary conditions. *SIAM J. Sci. Comput.*, 24:161–182, 2002.

[29] Ch. Lubich. Convolution quadrature and discretized operational calculus. I. *Num. Math.*, 52:129–145, 1988.

[30] Ch. Lubich. On the multistep time discretization of linear initial-boundary value problems and their boundary integral equations. *Num. Math.*, 67(3):365–389, 1994.

[31] M.J.Grote and J.B.Keller. On nonreflecting boundary conditions. *J. Comput. Phys.*, 122(2), 1995.

[32] M.J.Grote and J.B.Keller. Nonreflecting boundary conditions for time dependent scattering problems. *J. Comput. Phys.*, 127:52–65, 1996.

[33] M.J.Grote and Ch. Kirsch. Nonreflecting boundary condition for time-dependent multiple scattering. *J. Comput. Phys.*, 221(2):41–62, 2007.

[34] M.J.Grote and I. Sim. Local nonreflecting boundary condition for time-dependent multiple scattering. *J. Comput. Phys.*, 230:3135–3154, 2011.

[35] G. Monegato, L. Scuderi, and M.P. Stanić. Lubich convolution quadratures and their application to problems described by space-time BIEs. *Numerical Algorithms*, 56(3):405–436, 2010.

[36] I. Patlashenko and D. Givoli. Non reflecting finite element schemes for three-dimensional acoustic waves. *J. Computational Acoustics*, 5:95–115, 1997.

[37] I. Patlashenko and D. Givoli. DtN maps for unbounded wave guides. *J. Computational Physics*, 143:200–223, 1998.

[38] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer, 1994.