Exact non-reflecting boundary condition for 3D time-dependent multiple scattering-multiple source problems

Exact non-reflecting boundary condition for 3D time-dependent multiple scattering-multiple source problems[☆]

Silvia Falletta[a], Giovanni Monegato[b]

[a]*Dip. Scienze Matematiche "J.L. Lagrange" - Politecnico di Torino, e-mail: silvia.falletta@polito.it*
[b]*Dip. Scienze Matematiche "J.L. Lagrange" - Politecnico di Torino, e-mail: giovanni.monegato@polito.it*

Abstract

We consider some 3D wave equation problems defined in an unbounded domain, possibly with far field sources. For their solution, by means of standard finite element 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. The computational complexity of the discrete convolution is of order $N \log N$, being N the total number of time steps performed. That of the fully discretized NRBC is $O(N_{\mathcal{B}}^2 N \log N)$, where $N_{\mathcal{B}}$ denotes the number of mesh points taken on $\mathcal{B}$.

Besides showing a good accuracy and numerical stability, the proposed NRBC has the property of being suitable for artificial boundaries of general shapes. It also allows the treatment of far field (multiple) sources, that do not have to be necessarily included in the finite computational domain, being transparent not only for outgoing waves but also for incoming ones. This approach is in particular used to solve multiple scattering problems.

Keywords: wave equation; absorbing boundary conditions; space-time boundary integral equations; numerical methods; multiple scattering

1. Introduction

A key issue for solving PDE problems in unbounded domains is the introduction of a proper artificial boundary to delimit the computational domain of interest, hence the association with this boundary of a condition which guarantees that the solution of the initial boundary value problem defined in this (bounded) domain coincides with the corresponding restriction of the solution of the original problem. The method of Artificial (or Absorbing, or Non Reflecting) Boundary Condition (ABC or NRBC) consists of introducing an artificial boundary $\mathcal{B}$ that truncates the infinite domain and determines two distinct regions: a bounded one of interest Ω and a residual infinite domain $\mathcal{D}$. By analyzing the problem in $\mathcal{D}$, a non reflecting boundary condition on $\mathcal{B}$ is derived in order to avoid spurious reflections.

[☆]This work was supported by the Ministero dell'Istruzione, dell'Università e della Ricerca of Italy, under the research program PRIN 2012: Innovative methodologies for PDE-based numerical modelling.

Once a NRBC has been defined, this is coupled with the condition given on the boundary of the original problem physical domain and with the known initial values, to uniquely define the solution of the corresponding problem in Ω. Then, this latter can be solved, by coupling a time integrator with, for example, a finite differences or a finite elements method.

Many papers have been published on this topic, in particular in the last two decades; their number is too large to mention them. For a review, see for example [9], [10], [11], [6]. All these papers, except for [24], Sections 5.5, 5.6, [12], [16], [6], deal with the construction of NRBC with the property of absorbing only outgoing wave, not waves that are either outgoing or incoming. Therefore, known sources must necessarily be included in the computational domain. However, this can be a severe drawback when, for example, sources are far away from the physical domain. Moreover, the NRBC holds only for a single convex artificial boundary having a special shape, like a circle (sphere) or ellipse (ellipsoid). Only in the last years multiple scattering problems have been examined (see [12], [16]).

Very recently, in [6], we have proposed a global non reflecting boundary condition for the solution of two-dimensional exterior problems for the classical wave equation, which is given by a linear combination of a single and a double layer operators. It is defined by a known space-time boundary integral relationship that the problem solution and its normal derivative must satisfy at the chosen artificial boundary $\mathcal{B}$. It is of exact type, and it holds for a (smooth) curve of arbitrary shape; therefore, it can be used also in situations of multiple scattering, 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, the proposed NRBC naturally includes the effects of these data and it is automatically transparent for outgoing waves as well as for incoming ones.

For the discretization of the artificial condition, namely for the approximation of the single and double layer operators, we have proposed a numerical scheme which is based on a second order Lubich discrete convolution quadrature formula (see [18]) for the discretization of the time integral, coupled with a classical collocation method in space. The computational complexity of the discrete convolution rule is of order $N \log N$, being N the total number of time steps performed. That of the fully discretized NRBC is $O(N_{\mathcal{B}}^2 N \log N)$, where $N_{\mathcal{B}}$ denotes the number of mesh points taken on $\mathcal{B}$. When the discretization of the bounded domain Ω, where we apply the chosen finite element scheme, is refined, and the time stepsize is simultaneously reduced, the accuracy of the NRBC discretization increases.

The numerical examples presented in [6] for the two-dimensional case, show that indeed the proposed NRBC is very competitive, from both the accuracy and the computational cost points of view, with some of local type, such as the second order Engquist-Majda and Bayliss-Turkel ones.

In this new paper, we extend the results presented in [6] to the three-dimensional case, and also consider multiple scattering/multiple source problems. The computational cost of this NRBC is significantly higher than that of local types; however, we believe that its generality and new applications should justify it. In any case, a first attempt to reduce it is described in Section 3.4. Four numerical examples are presented in Section 6, while in the last section we draw some conclusions and outline some possible further improvements.

2. Exact non reflecting boundary conditions for multiple scattering

We consider the problem of a wave propagating through a homogeneous medium in three dimension and impinging upon a scatterer made up of $\boldsymbol{\kappa}$ bounded obstacles of arbitrary shape. We assume that the scatterers are impenetrable and well separated from each other. We denote by $\Omega_k^i \subset \mathbb{R}^3$, with $k = 1, \cdots, \boldsymbol{\kappa}$, $\Omega_k^i \cap \Omega_\ell^i = \emptyset$ for $k \neq \ell$, the open bounded domains with sufficiently smooth boundaries Γ_k, and by $\Omega^i = \cup_{k=1}^{\boldsymbol{\kappa}} \Omega_k^i \subset \mathbb{R}^3$ the connected open domain bounded by the union of the obstacle boundaries $\Gamma = \cup_{k=1}^{\boldsymbol{\kappa}} \Gamma_k$. Then, we set $\Omega^e = \mathbb{R}^3 \setminus \overline{\Omega^i}$, and consider the following wave propagation problem in Ω^e:

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

As often occurs in practical situations, we assume that the initial values u_0, v_0 and the source term f have local supports.

When one has to determine the solution u^e of the above problem in a bounded subregion of Ω^e, surrounding the physical obstacles Ω_k^i, it is necessary to truncate the infinite domain Ω^e by introducing an artificial smooth boundary $\mathcal{B}$. However, the idea of introducing a single artificial boundary $\mathcal{B}$ that encloses all the obstacles becomes too expensive when, for example, the scatterers are far from each other, and one has to determine the solution u^e only in a neighborhood of each scatterer. In this case, it is preferable to surround each single obstacle by an artificial boundary $\mathcal{B}_k$ and compute the problem solution in the domains of interest.

We denote by Ω_k the subdomain bounded internally by Γ_k and externally by $\mathcal{B}_k$, and by $\Omega = \cup_{k=1}^{\boldsymbol{\kappa}} \Omega_k$ the bounded computational domain of interest. Finally, we set $\mathcal{D} = \mathbb{R}^3 \setminus \overline{\Omega}$. To solve our problem in the Ω domain, we need to prescribe Non Reflecting Boundary Conditions on $\mathcal{B} = \cup_{k=1}^{\boldsymbol{\kappa}} \mathcal{B}_k$, which allow outgoing waves leave Ω_k without spurious reflections. These waves propagate to all other subdomains, and are then reflected by the other scatterers; therefore, subsequently they reenter in Ω_k.

We remark that the artificial boundary is chosen to detect the (bounded) 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 this end, we analyze the problem in $\mathcal{D}$, and we impose on $\mathcal{B}$ the integral relation that the solution u and its (outward) normal derivative $\partial_{\mathbf{n}_\mathcal{D}} u$ have to satisfy. In [6] an exact NRBC has been proposed in two dimension and for a single scatterer and a single artificial boundary. Following [5], where a BIE has been derived for the non homogeneous wave equation with non trivial initial data, the derivation of a NRBC in the case of multiple connected artificial boundaries is fairly simple. To write it in a more compact form, we

introduce the single and double layer integral operators, defined by

$$\begin{aligned}\boldsymbol{\mathcal{V}}\psi(\mathbf{x},t) &:= \int_0^t \int_{\mathcal{B}} G(\mathbf{x}-\mathbf{y},t-\tau)\psi(\mathbf{y},\tau)d\mathcal{B}_{\mathbf{y}}d\tau \\ &= \sum_{k=1}^{\boldsymbol{\kappa}} \int_0^t \int_{\mathcal{B}_k} G(\mathbf{x}-\mathbf{y},t-\tau)\psi(\mathbf{y},\tau)d\mathcal{B}_{\mathbf{y}}d\tau,\end{aligned}$$

and

$$\begin{aligned}\boldsymbol{\mathcal{K}}\varphi(\mathbf{x},t) &:= \int_0^t \int_{\mathcal{B}} \partial_{\mathbf{n}_{\mathcal{D}}} G(\mathbf{x}-\mathbf{y},t-\tau)\varphi(\mathbf{y},\tau)d\mathcal{B}_{\mathbf{y}}d\tau \\ &= \sum_{k=1}^{\boldsymbol{\kappa}} \int_0^t \int_{\mathcal{B}_k} \partial_{\mathbf{n}_{\mathcal{D}}} G(\mathbf{x}-\mathbf{y},t-\tau)\varphi(\mathbf{y},\tau)d\mathcal{B}_{\mathbf{y}}d\tau,\end{aligned}$$

respectively, where $\partial_{\mathbf{n}_{\mathcal{D}}}$ denotes the outward unit normal derivative on the boundary $\mathcal{B}$ for the problem defined in the residual domain $\mathcal{D}$, and $G(\mathbf{x},t)$ is the fundamental solution of the wave equation given in (1), that is:

$$G(\mathbf{x},t) = \frac{\delta(t-\|\mathbf{x}\|)}{4\pi\|\mathbf{x}\|}, \tag{2}$$

$\delta(\cdot)$, being the well known Dirac delta function. The NRBC on $\mathcal{B}$ is then given by:

$$\frac{1}{2}u(\mathbf{x},t) = \boldsymbol{\mathcal{V}}\partial_{\mathbf{n}_{\mathcal{D}}}u(\mathbf{x},t) - \boldsymbol{\mathcal{K}}u(\mathbf{x},t) + I_{u_0}(\mathbf{x},t) + I_{v_0}(\mathbf{x},t) + I_f(\mathbf{x},t) \quad \mathbf{x}\in\mathcal{B}, \tag{3}$$

where the "volume" terms I_{u_0}, I_{v_0} and I_f are generated by the non homogeneous initial conditions and the non trivial source, respectively. These volume terms have the following integral representations:

$$I_{u_0}(\mathbf{x},t) = \begin{cases} \dfrac{\partial}{\partial t}\displaystyle\int_{\mathrm{supp}(u_0)} u_0(y)G(\mathbf{x}-\mathbf{y},t)dy, & \text{if} \quad \mathrm{supp}(u_0)\subset\mathcal{D} \\ 0, & \text{if} \quad \mathrm{supp}(u_0)\subset\Omega \end{cases} \tag{4}$$

$$I_{v_0}(\mathbf{x},t) = \begin{cases} \displaystyle\int_{\mathrm{supp}(v_0)} v_0(y)G(\mathbf{x}-\mathbf{y},t)dy, & \text{if} \quad \mathrm{supp}(v_0)\subset\mathcal{D} \\ 0, & \text{if} \quad \mathrm{supp}(v_0)\subset\Omega \end{cases} \tag{5}$$

$$I_f(\mathbf{x},t) = \begin{cases} \displaystyle\int_0^t \int_{\mathrm{supp}(f)} f(y,\tau)G(\mathbf{x}-\mathbf{y},t-\tau)dyd\tau, & \text{if} \quad \mathrm{supp}(f)\subset\mathcal{D} \\ 0, & \text{if} \quad \mathrm{supp}(f)\subset\Omega. \end{cases} \tag{6}$$

Thus, denoting by $\partial_{\mathbf{n}} = \frac{\partial}{\partial\mathbf{n}}$ the outward unit normal derivative defined on the boundary $\mathcal{B}$, for the problem defined in 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 following form:

$$\begin{cases} u_{tt}(\mathbf{x},t) - \Delta u(\mathbf{x},t) & = \tilde{f}(\mathbf{x},t) & \text{in } \Omega \times (0,T) \\ u(\mathbf{x},t) & = g(\mathbf{x},t) & \text{in } \Gamma \times (0,T) \\ \frac{1}{2}u(\mathbf{x},t) + \boldsymbol{\mathcal{V}}\partial_{\mathbf{n}}u(\mathbf{x},t) + \boldsymbol{\mathcal{K}}u(\mathbf{x},t) & = I_{u_0}(\mathbf{x},t) + I_{v_0}(\mathbf{x},t) + I_f(\mathbf{x},t) & \text{in } \mathcal{B} \times (0,T) \\ u(\mathbf{x},0) & = \tilde{u}_0(\mathbf{x}) & \text{in } \Omega \\ u_t(\mathbf{x},0) & = \tilde{v}_0(\mathbf{x}) & \text{in } \Omega, \end{cases} \tag{7}$$

where

$$\tilde{u}_0 = \begin{cases} 0 & \text{if} \quad \text{supp}(u_0) \subset \mathcal{D}. \\ u_0 & \text{if} \quad \text{supp}(u_0) \subset \Omega \end{cases}$$
$$\tilde{v}_0 = \begin{cases} 0 & \text{if} \quad \text{supp}(v_0) \subset \mathcal{D}. \\ v_0 & \text{if} \quad \text{supp}(v_0) \subset \Omega \end{cases}$$
$$\tilde{f} = \begin{cases} 0 & \text{if} \quad \text{supp}(f) \subset \mathcal{D}. \\ f & \text{if} \quad \text{supp}(f) \subset \Omega \end{cases}$$

As far as we know, this is the first NRBC that allows to include the (locally supported) source and initial data in the non absorbing boundary condition rather than in the computational domain. All the existing NRBCs work under the assumption that the supports of the data are included in the computational domain, so that the chosen (convex) artificial boundary $\mathcal{B}$ must enclose them. Thus, the resulting method becomes too expensive when the datum supports are far from the obstacles. On the contrary, in such cases, using our NRBC we can choose artificial boundaries of general shape, even non convex, and let the supports of the source and the initial data be included in the residual infinite domain $\mathcal{D}$.

To discretize the NRBC, i.e., the single and double layer operators, we propose a numerical scheme which is based on a discrete convolution quadrature formula, for the time integral approximation, and a classical collocation method for the space integral discretization. As remarked in [6], if N denotes the number of time steps to be performed, the proposed NRBC discretization requires $O(N \log N)$ operations to compute, for each given collocation (space) point, the associated temporal convolution at all chosen instants.

We set $u(t)(\mathbf{x}) = u(\mathbf{x},t)$ and 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 $\mathcal{B}$. Following [6], but in a more general setting here, for any given $t \geq 0$ we also introduce the functional spaces

$$X_k = X_k(t) = \{u_k(t) \in H^1(\Omega_k) : u_k(t)|_{\Gamma_k} = g(t)|_{\Gamma_k}\}, \qquad X = X(t) = \prod_{k=1}^{\boldsymbol{\kappa}} X_k$$

and

$$X_{k,0} = \{u_k(t) \in H^1(\Omega_k) : u_k(t)|_{\Gamma_k} = 0\}, \qquad X_0 = \prod_{k=1}^{\boldsymbol{\kappa}} X_{k,0}.$$

Similarly, we set

$$H^{-1/2}(\mathcal{B}) = \prod_{k=1}^{\boldsymbol{\kappa}} H^{-1/2}(\mathcal{B}_k).$$

Then, the problem defined in the domain of interest Ω takes the following form:
given $\tilde{f} \in L^2(\Omega \times (0,T)), \tilde{u}_0 \in X, \tilde{v}_0 \in L^2(\Omega)$, *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*

$$\begin{cases} \frac{d^2}{dt^2}(u(t),w)_\Omega + a(u(t),w) - (\lambda(t),w)_\mathcal{B} &= (\tilde{f}(t),w)_\Omega, & \forall w \in X_0 \\ \frac{1}{2}u(\mathbf{x},t) + \boldsymbol{\mathcal{V}}\lambda(\mathbf{x},t) + \boldsymbol{\mathcal{K}}u(\mathbf{x},t) &= I_{u_0}(\mathbf{x},t) + I_{v_0}(\mathbf{x},t) + I_f(\mathbf{x},t) & \text{on } \mathcal{B} \\ u(0) &= \tilde{u}_0 \quad \text{in } \Omega \\ \frac{du}{dt}(0) &= \tilde{v}_0 \quad \text{in } \Omega. \end{cases} \tag{8}$$

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

$$a(v,w) = \int_\Omega \nabla v \cdot \nabla w = \sum_{k=1}^{\boldsymbol{\kappa}} \int_{\Omega_k} \nabla v_k \cdot \nabla w_k,$$

and, $(v,w)_D = \int_D vw = \sum_{k=1}^{\boldsymbol{\kappa}} \int_{D_k} v_k w_k$, *where, depending on its occurrence,* D *is either* Ω *or* $\mathcal{B}$.

3. Discretization of the NRBC

3.1. Approximation in time

We approximate the single and double layer operators appearing in the NRNC by combining a second order (time) convolution quadrature formula of Lubich (see [18]) with a classical space collocation method. For its 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 $t_n = n\Delta_t$, $n = 0,\ldots,N$:

$$\frac{1}{2}u(\mathbf{x},t_n) + (\boldsymbol{\mathcal{V}}\lambda)(\mathbf{x},t_n) + (\boldsymbol{\mathcal{K}}u)(\mathbf{x},t_n) \quad = I_{u_0}(\mathbf{x},t_n) + I_{v_0}(\mathbf{x},t_n) + I_f(\mathbf{x},t_n) \tag{9}$$

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 second order Backward Differentiation Method (BDF) for ordinary differential equations (see [5]). We obtain:

$$(\boldsymbol{\mathcal{V}}\lambda)(\mathbf{x},t_n) \;\approx\; \sum_{j=0}^{n}\sum_{k=1}^{\boldsymbol{\kappa}} \int_{\mathcal{B}_k} \omega_{n-j}^{\boldsymbol{\mathcal{V}}}(\Delta_t; ||\mathbf{x}-\mathbf{y}||)\lambda(\mathbf{y},t_j)\, d\mathcal{B}_\mathbf{y}, \quad n = 0,\ldots,N \tag{10}$$

$$(\boldsymbol{\mathcal{K}}u)(\mathbf{x},t_n) \;\approx\; \sum_{j=0}^{n}\sum_{k=1}^{\boldsymbol{\kappa}} \int_{\mathcal{B}_k} \omega_{n-j}^{\boldsymbol{\mathcal{K}}}(\Delta_t; ||\mathbf{x}-\mathbf{y}||)u(\mathbf{y},t_j)\, d\mathcal{B}_\mathbf{y}, \quad n = 0,\ldots,N \tag{11}$$

whose coefficients $\omega_n^\mathcal{J}, \mathcal{J} = \boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{K}}$, 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 $K^{\boldsymbol{\mathcal{V}}} = \widehat{G}$ is the Laplace transform of the kernel G appearing in the definition of the single layer operator $\boldsymbol{\mathcal{V}}$, and $K^{\boldsymbol{\mathcal{K}}} = \widehat{\partial G/\partial \mathbf{n}}$ is the Laplace transform of the kernel $\partial G/\partial \mathbf{n}$ appearing in the definition of the double layer operator $\boldsymbol{\mathcal{K}}$.

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

$$\begin{aligned} K^{\boldsymbol{\mathcal{V}}}(r,s) &= \frac{1}{4\pi r} e^{-rs}, \\ K^{\boldsymbol{\mathcal{K}}}(r,s) &= -\frac{1}{4\pi r} e^{-rs} \left(\frac{1}{r} + s\right) \frac{\partial r}{\partial \mathbf{n}}. \end{aligned} \tag{12}$$

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

$$\omega_n^{\mathcal{J}}(\Delta_t; r) \approx \frac{\rho^{-n}}{L} \sum_{l=0}^{L-1} K^{\mathcal{J}} \left(r, \frac{\gamma(\rho \exp(il2\pi/L))}{\Delta_t}\right) \exp(-inl2\pi/L). \tag{13}$$

We choose $L = 2N$ and $\rho^N = \sqrt{\varepsilon}$, since Lubich in ([18]) 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 $\mathbf{x} \in \mathcal{B}$, all the $\omega_n^{\mathcal{J}}$ can be computed simultaneously by the FFT, with $O(N \log N)$ flops. Note that when we choose $L > N$, as in our case, the required $\omega_n^{\mathcal{J}}$ are given by the first N components of the coefficient vector determined by the FFT.

3.2. Approximation in space

For the space discretization, each surface $\mathcal{B}_k$, of the global artificial boundary $\mathcal{B}$, is approximated by a continuous piecewise triangular surface $\mathcal{B}_{k,\Delta}$, interpolating $\mathcal{B}_k$ at the triangle vertices $\{\mathbf{x}_{k,i}, i = 1, \ldots, M_k\}$. We denote by $\Delta_{x,k}$ the mesh size of $\mathcal{B}_{k,\Delta}$, which is given by the maximum triangle diameter.

Further, we set $u_{\mathcal{B}_k}(\mathbf{x},t) = u(\mathbf{x},t)_{|_{\mathcal{B}_k}}$ and $\lambda_{\mathcal{B}_k}(\mathbf{x},t) = \lambda(\mathbf{x},t)_{|_{\mathcal{B}_k}}$, $k = 1, \cdots, \boldsymbol{\kappa}$. At each time instant t_j, the unknown function $u_{\mathcal{B}_k}(\cdot, t_j)$ and its normal derivative $\lambda_{\mathcal{B}_k}(\cdot, t_j)$ on $\mathcal{B}_k$ are approximated by

$$u_{\Delta_{x,k}}(\mathbf{x}, t_j) := \sum_{i=1}^{M_k} u_{k,i}^j b_{k,i}(\mathbf{x}), \quad \mathbf{x} \in \mathcal{B}_{k,\Delta} \tag{14}$$

and

$$\lambda_{\Delta_{x,k}}(\mathbf{x}, t_j) := \sum_{i=1}^{M_k} \lambda_{k,i}^j b_{k,i}(\mathbf{x}), \quad \mathbf{x} \in \mathcal{B}_{k,\Delta} \tag{15}$$

respectively, where $u_{k,i}^j \approx u_{\mathcal{B}_k}(\mathbf{x}_{k,i}, t_j)$, $\lambda_{k,i}^j \approx \lambda_{\mathcal{B}_k}(\mathbf{x}_{k,i}, t_j)$, and $\{b_{k,i}\}_{i=1}^{M_k}$ are the classical continuous piecewise linear basis functions associated with the given triangulation.

3.3. Time-space discretization

After having introduced the above time and space discretizations, we collocate the resulting discretized BIE at the (collocation) points $\mathbf{x}_{k,h}$, $h = 1, \ldots, M_k$, for each $k = 1, \cdots, \boldsymbol{\kappa}$. To write the final system of equations in vectorial notation, we define the matrices

$$(\mathbf{V}_{n-j}^{k,\ell})_{hi} = \int_{\mathcal{B}_\ell} \omega_{n-j}^{\boldsymbol{\mathcal{V}}}(\Delta_t; \|\mathbf{x}_{k,h} - \mathbf{y}\|) b_{\ell,i}(\mathbf{y}) d\mathcal{B}_y, \quad i = 1, \cdots, M_\ell \tag{16}$$

$$(\mathbf{K}_{n-j}^{k,\ell})_{hi} = \int_{\mathcal{B}_\ell} \omega_{n-j}^{\boldsymbol{\mathcal{K}}}(\Delta_t; \|\mathbf{x}_{k,h} - \mathbf{y}\|) b_{\ell,i}(\mathbf{y}) d\mathcal{B}_y, \quad i = 1, \cdots, M_\ell \tag{17}$$

$\ell = 1, \ldots, \boldsymbol{\kappa}$, and the vectors

$$\begin{aligned}
\mathbf{I}_{u_0}^{k,n} &= [I_{u_0}(\mathbf{x}_{k,1}, t_n), I_{u_0}(\mathbf{x}_{k,2}, t_n), \cdots, I_{u_0}(\mathbf{x}_{k,M_k}, t_n)]^T \\
\mathbf{I}_{v_0}^{k,n} &= [I_{v_0}(\mathbf{x}_{k,1}, t_n), I_{v_0}(\mathbf{x}_{k,2}, t_n), \cdots, I_{v_0}(\mathbf{x}_{k,M_k}, t_n)]^T \\
\mathbf{I}_{f}^{k,n} &= [I_{f}(\mathbf{x}_{k,1}, t_n), I_{f}(\mathbf{x}_{k,2}, t_n), \cdots, I_{f}(\mathbf{x}_{k,M_k}, t_n)]^T .
\end{aligned}$$

Then, we introduce the unknown vectors $\mathbf{u}_{\mathcal{B}_\ell}^j = \left[u_{\ell,1}^j, \ldots, u_{\ell,M_\ell}^j\right]^T$ and $\boldsymbol{\lambda}_{\mathcal{B}_\ell}^j = \left[\lambda_{\ell,1}^j, \ldots, \lambda_{\ell,M_\ell}^j\right]^T$, for $\ell = 1, \ldots, \boldsymbol{\kappa}$ and $j = 0, \ldots, n$, and obtain the following system of equations (see [6] for the case of a single scatterer):

$$\begin{aligned}
\left(\frac{1}{2}\mathbf{I}_k + \mathbf{K}_0^{k,k}\right)\mathbf{u}_{\mathcal{B}_k}^n &+ \sum_{\substack{\ell=1 \\ \ell\neq k}}^{\boldsymbol{\kappa}} \mathbf{K}_0^{k,\ell}\mathbf{u}_{\mathcal{B}_\ell}^n + \sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n-1} \mathbf{K}_{n-j}^{k,\ell}\mathbf{u}_{\mathcal{B}_\ell}^j + \sum_{\ell=1}^{\boldsymbol{\kappa}} \mathbf{V}_0^{k,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^n + \sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n-1} \mathbf{V}_{n-j}^{k,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^j \\
&= \mathbf{I}_{u_0}^{k,n} + \mathbf{I}_{v_0}^{k,n} + \mathbf{I}_f^{k,n}, \qquad k = 1, \ldots, \boldsymbol{\kappa};\ n = 0, \ldots, N
\end{aligned} \tag{18}$$

where the matrix $\mathbf{I}_k$ denotes the identity matrix of order M_k.

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

$$\begin{aligned}
\left(\frac{1}{2}\mathbf{I}_k + \mathbf{K}_0^{k,k}\right)\mathbf{u}_{\mathcal{B}_k}^n &+ \sum_{\substack{\ell=1 \\ \ell\neq k}}^{\boldsymbol{\kappa}} \mathbf{K}_0^{k,\ell}\mathbf{u}_{\mathcal{B}_\ell}^n + \sum_{\ell=1}^{\boldsymbol{\kappa}} \mathbf{V}_0^{k,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^n \\
&= -\sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n-1} \mathbf{K}_{n-j}^{k,\ell}\mathbf{u}_{\mathcal{B}_\ell}^j - \sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n-1} \mathbf{V}_{n-j}^{k,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^j + \mathbf{I}_{u_0}^{k,n} + \mathbf{I}_{v_0}^{k,n} + \mathbf{I}_f^{k,n}
\end{aligned} \tag{19}$$

for each $k = 1, \ldots, \boldsymbol{\kappa}$.

Remark 3.1. *As described in [5], 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 (16), (17), the ω kernel by its discretization (13), and exchanging the integration symbol with that of the quadrature sum (for details see [5]).*

The evaluation of the above volume integrals $\mathbf{I}_{u_0}^{k,n}$, $\mathbf{I}_{v_0}^{k,n}$, $\mathbf{I}_f^{k,n}$ has been discussed in [20], where an efficient numerical approach has been proposed for the 2D case and for compactly supported data (see also [5]). In the forthcoming numerical tests, we will consider sources concentrated at a point. This choice extremely simplifies the evaluation of the volume term appearing in the NRBC equation.

3.4. NRBC computational cost and memory space

Let assume that the chosen Finite Element (FE) grid is regular. For notational simplicity, we denote by N_h the number of its interior points, and by $N_{\mathcal{B}}$ that of the points lying on the boundary $\mathcal{B}_k$; then, we have $N_{\mathcal{B}} = O(N_h^{2/3})$ as $N_h \to \infty$, i.e., as $\Delta_x \to 0$. Furthermore, as we have already tested in the 2D case, the robustness of the proposed NRBC discretization, and its higher accuracy with respect to that of the associated FEM, may allow a decoupling of the NRBC grid from that of the FEM. That is, one might construct the discretization of the ABC on a subset of the boundary nodes defined by the FE grid. The coupling of the two grids can be performed by a local linear interpolation process.

We further recall that all sums

$$\sum_{j=0}^{n-1} \mathbf{K}_{n-j}^{k,\ell} \mathbf{u}_{\mathcal{B}_\ell}^j, \quad \sum_{j=0}^{n-1} \mathbf{V}_{n-j}^{k,\ell} \boldsymbol{\lambda}_{\mathcal{B}_\ell}^j, \qquad n = 1, \ldots, N \tag{20}$$

are simultaneously computed by applying a FFT-based algorithm (see [3], Sect. 8.3.1), with a computational cost of $O(N_{\mathcal{B}}^2 N \log N) = O(N_h^{4/3} N \log N)$ flops.

The working space at a first glace appears to be that of $2N_{\mathcal{B}}^2 N$ real numbers; that is, that due to the construction of the above matrices $\mathbf{K}_m, \mathbf{V}_m, m = 0, \ldots, N$. However, this can be significantly reduced. Indeed, as mentioned in Remark 3.1, for each row index, the corresponding row elements of all matrices $\mathbf{K}_m, \mathbf{V}_m, m = 0, \ldots, N$, are simultaneously computed by means of the FFT algorithm. However, before computing, for all matrices, the elements of all rows having the next row index, we set equal to zero those whose size is less than a threshold value; for example, 10^{-5} or 10^{-8}, depending of the final accuracy we want to achieve. Then, we will store, and later use, only the remaining "non zero" row elements. At the end, the total number of elements of each couple of matrices $\mathbf{K}_m, \mathbf{V}_m$ that need to be store is only a fraction of $2N_{\mathcal{B}}^2$. Also the corresponding matrix-vector products will have a computational cost much lower than $N_{\mathcal{B}}^2$.

Some theoretical results, which partially allows us to understand the behavior of the matrices $\mathbf{V}_m$, are reported below. Unfortunately, we cannot derive similar ones for the $\mathbf{K}_m$ matrices, since an explicit representation for the ω-coefficients which define them is not known. However, the numerical testing we have performed seem to confirm that, as in the 2D case, the behaviors of both matrices are very similar.

Since for the ω-coefficients associated with the $\boldsymbol{\mathcal{V}}$ operator, an explicit analytic representation is known, to verify if the matrices $\mathbf{V}_m$ have some properties, which may be useful to reduce the computational cost of the discretized NRBC, we have obtained the following bounds.

Lemma 3.2. *Let $\Delta_t = T/N$, with $T > 0$ fixed and N arbitrary positive integer. For the ω-coefficients associated with the operator $\boldsymbol{\mathcal{V}}$ the following bounds hold.*

(i) For all integers $1 \le n \le N$ and reals $r > 0$,

$$r|\omega_n(\Delta_t; r)| < 0.05462 \times n^{-\frac{1}{4}} \left(\frac{rN}{nT} e^{-\frac{rN}{nT}+1} \right)^{n/2}. \tag{21}$$

(ii) Let n be a fixed positive integer. For $0 < r_0 \leq r \leq r_1$, with r_0, r_1 arbitrary real numbers, we have:

$$|\omega_n(\Delta_t; r)| \leq C_n \left(N e^{-\alpha_n N}\right)^{n/2}, \tag{22}$$

where

$$C_n = \frac{0.05462}{r_0} n^{-1/4} \left(\frac{r_1 e}{nT}\right)^{n/2}, \qquad \alpha_n = \frac{r_0}{nT}.$$

Proof. In the case of the operator $\boldsymbol{\mathcal{V}}$, the following explicit representation for the ω coefficients has been derived in [17] (see also [19]):

$$\omega_n(\Delta_t; r) = \frac{1}{4\pi r n!} e^{-\frac{3r}{2\Delta_t}} \left(\frac{r}{2\Delta_t}\right)^{n/2} H_n\left(\sqrt{\frac{2r}{\Delta_t}}\right), \; n \geq 0, \tag{23}$$

where $r = \|\mathbf{x} - \mathbf{y}\|, \mathbf{x}, \mathbf{y} \in \mathcal{B}$, and $H_n(x) = 2^n x^n + \ldots$ is the n-degree Hermite orthogonal polynomial.

First we consider the case $n = 1 \ldots, N$, with $N \to \infty$, that is, $\Delta_t \to 0$. Using the well-known Stirling's formula for the factorial:

$$n! = \sqrt{2\pi} n^{n+\frac{1}{2}} e^{-n+\frac{\theta}{12n}}, \quad 0 < \theta < 1$$

and the bound (see [1], (22.14.17))

$$|H_n(x)| < 1.0865 \times 2^{\frac{n}{2}} e^{\frac{x^2}{2}} \sqrt{n!}$$

from representation (23) we obtain:

$$r|\omega_n(\Delta_t; r)| < \frac{1.0865}{2^{9/4}\pi^{5/4}} n^{-\frac{n}{2}-\frac{1}{4}} e^{\frac{n}{2}} e^{-\frac{1}{2}\frac{rN}{T}} \left(\frac{rN}{T}\right)^{n/2},$$

from which (21) follows.

Then, we consider the case n fixed and $\Delta_t \to 0$. We also assume $0 < r_0 \leq r \leq r_1$, with r_0, r_1 given real numbers. In this case, bound (22) follows immediately from (21). □

Starting from (21) above, and noting that the function $y = xe^{-x+1}$ is always positive in $(0, \infty)$, is increasing from 0 to its maximum value 1 in the interval $[0, 1]$, and exponentially decaying to zero in $[1, \infty)$, a straightforward calculation then gives the following results.

Corollary 3.3. *Let assume $T > kd_B$, where $0 < r \leq d_B$, d_B denoting the diameter of the artificial boundary $\mathcal{B}$. Then, for all integers n, $N/k \leq n \leq N$, with $k \geq 1$ fixed, we have:*

$$r|\omega_n(\Delta_t; r)| < 0.05462 \times n^{-1/4} \left(\frac{kd_B}{T} e^{-\frac{kd_B}{T}+1}\right)^{\frac{n}{2}}, \tag{24}$$

where $0 < \frac{kd_B}{T} e^{-\frac{kd_B}{T}+1} < 1$.

Note that if in bounds (24) we set $x = kd_B/T$, for $0 \leq x \leq 1/4$, i.e. $T \geq 4kd_B$, we have, for example, $y < 0.53$, hence $0.53^{32} < 1.51E-9$ and $0.53^{64} < 2.26E-18$. The smaller is x the smaller is y, hence the faster is the convergence to zero of the associated powers. In particular, for all $r > 0$, the right hand side of (24) tends to zero exponentially, as $N_T \to \infty$. In general, as long as x is away from the abscissa $\bar{x} = 1$, the corresponding value of $y^{n/2}$ decay exponentially to zero, as $N \to \infty$.

Remark 3.4. *From bound (24) it also follows that the larger is T with respect to the diameter of the artificial boundary $\mathcal{B}$, the larger is the value of k that can be taken. Thus for all integers N sufficiently large, all the matrices V_n, $N/k \leq n \leq N$, can be neglected.*

In the following three sets of figures, we consider an artificial boundary given by a sphere of radius $R = 0.1$, final time instants $T = 0.5, 1, 10$, respectively, and time step T/N. In particular, on the left hand side figures, we report the maximum absolute value of the matrix elements of the matrices $\mathbf{V}_n$ and $\mathbf{K}_n$, for $n = 0, \ldots, N$. On the right hand side figures, we plot, for each n, the number of each matrix element whose size is larger or equal than the threshold value $1E-8$.

Figure 1: Left plot: behavior of the maximum absolute value of the elements of the matrix V_n (Single Layer) versus time. Right plot: number of the non zero entries of the matrix V_n after threshold. $R = 0.1, T = 0.5$

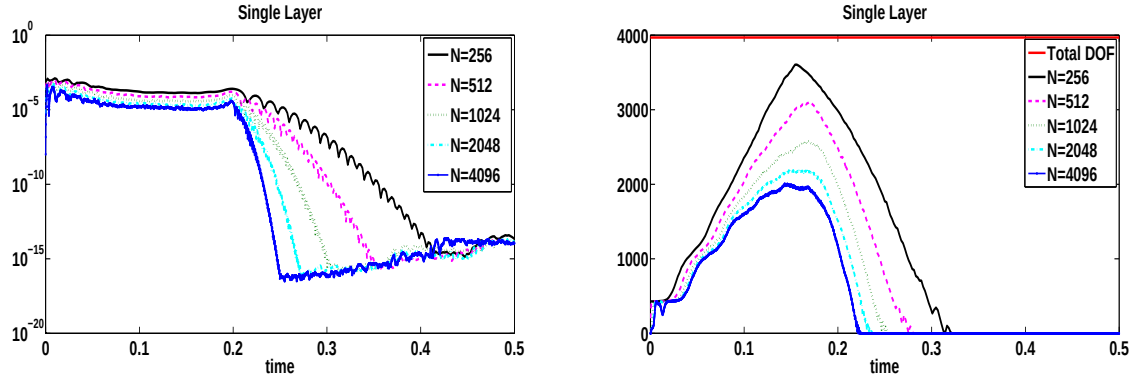


Figure 2: Left plot: behavior of the maximum absolute value of the elements of the matrix K_n (Double Layer) versus time. Right plot: number of the non zero entries of the matrix K_n after threshold. $R = 0.1, T = 0.5$

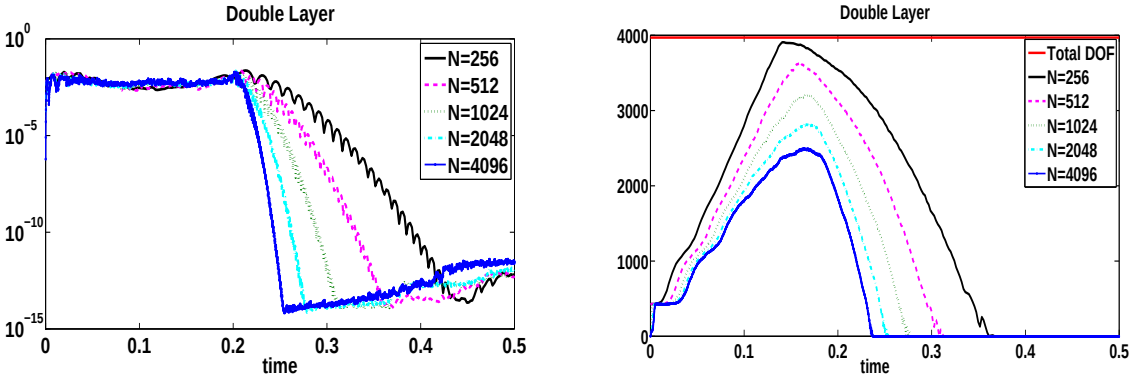


Figure 3: Left plot: behavior of the maximum absolute value of the elements of the matrix V_n (Single Layer) versus time. Right plot: number of the non zero entries of the matrix V_n after threshold. $R = 0.1, T = 1$

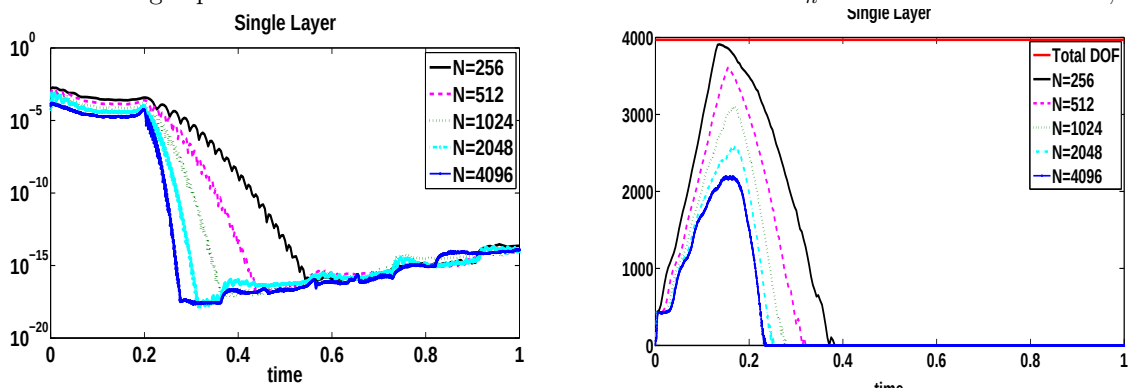


Figure 4: Left plot: behavior of the maximum absolute value of the elements of the matrix K_n (Double Layer) versus time. Right plot: number of the non zero entries of the matrix K_n after threshold. $R = 0.1, T = 1$

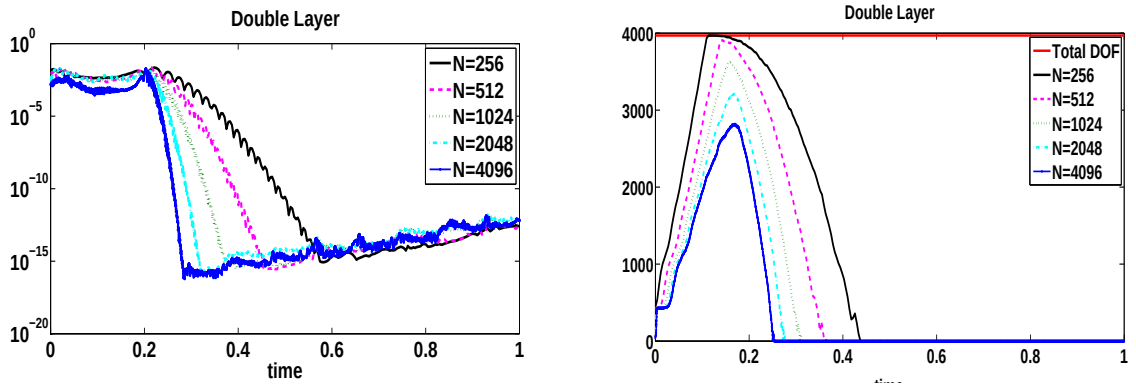
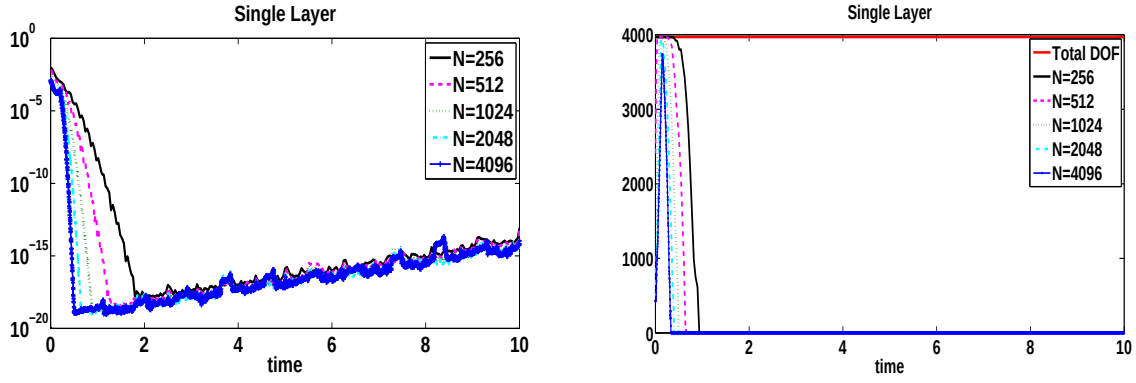


Figure 5: Left plot: behavior of the maximum absolute value of the elements of the matrix V_n (Single Layer) versus time. Right plot: number of the non zero entries of the matrix V_n after threshold. $R = 0.1, T = 10$

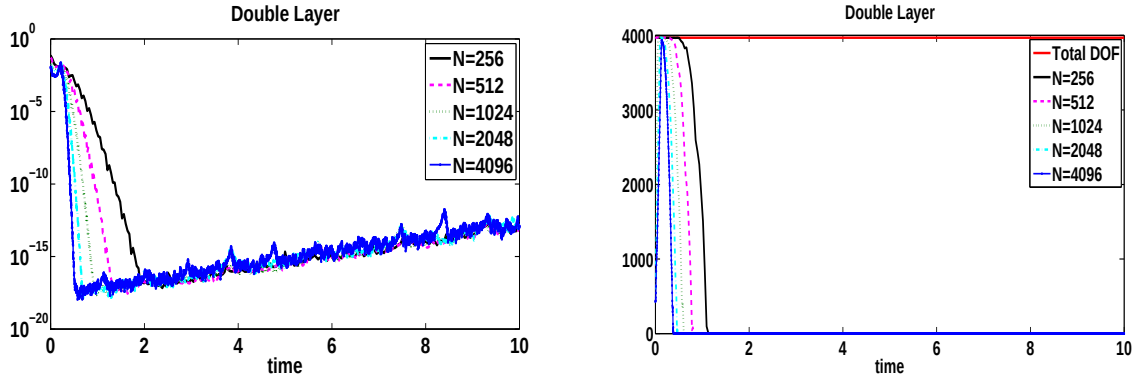


4. Discretization of the boundary value problem

4.1. Time discretization

In principle, any integration scheme can be used for the time discretization of the first equation in (8) (see for example [6], where different time schemes have been considered and

Figure 6: Left plot: behavior of the maximum absolute value of the elements of the matrix K_n (Double Layer) versus time. Right plot: number of the non zero entries of the matrix K_n after threshold. $R = 0.1, T = 10$



compared). For simplicity, we choose to use the Crank-Nicolson integration method, which is of second order and unconditionally stable. Thus, denoting by $v := \frac{\partial u}{\partial t}$ and by $u^n = u^n(x)$, $v^n = v^n(x)$, $\lambda^n = \lambda^n(x)$ and $\tilde{f}^n = \tilde{f}^n(x)$ the approximations of $u(x,t_n)$, $v(x,t_n)$, $\lambda(x,t_n)$ and $\tilde{f}(x,t_n)$, respectively, and applying the Crank-Nicolson discretization, a straightforward calculation leads to the following time-marching scheme (for more details see [6])

$$\begin{cases} (u^{n+1},w)_\Omega + \frac{\Delta_t^2}{4}a(u^{n+1},w) - \frac{\Delta_t^2}{4}(\lambda^{n+1},w)_\mathcal{B} &= (u^n,w)_\Omega - \frac{\Delta_t^2}{4}a(u^n,w) + \frac{\Delta_t^2}{4}(\lambda^n,w)_\mathcal{B} \\ & + \Delta_t(v^n,w)_\Omega + \frac{\Delta_t^2}{4}(\tilde{f}^{n+1}+\tilde{f}^n,w)_\Omega,\ \forall\, w \in X_0 \\ v^{n+1} = \frac{2}{\Delta_t}(u^{n+1}-u^n) - v^n. \end{cases} \tag{25}$$

4.2. Space discretization

At each time t_n, to compute the unknowns u^n and λ^n we will use a finite element method. To this end, we consider a finite decomposition of each computational domain Ω_k (defined in Section 2) into tetrahedra and we denote by $\Omega_{k,\Delta} = \cup_{\mathcal{T}\in\mathcal{T}_{k,h}}\mathcal{T}$ the finite polyhedral domain, whose mesh size is bounded by h and whose inner and outer boundaries are denoted by $\Gamma_{k,\Delta}$ and $\mathcal{B}_{k,\Delta}$, respectively. Then, we associate with this decomposition the functional spaces

$$X_{k,h} = \{w_{k,h} \in C^0(\Omega_k) : w_{k,h|_\mathcal{T}} \in \mathbb{P}^1(\mathcal{T}), \mathcal{T} \in \mathcal{T}_{k,h}, w_{k,h|_{\Gamma_{k,\Delta}}} = g|_{\Gamma_{k,\Delta}}\} \subset H^1(\Omega_k),$$

$$X_{k,h,0} = \{w_{k,h} \in C^0(\Omega_k) : w_{k,h|_\mathcal{T}} \in \mathbb{P}^1(\mathcal{T}), \mathcal{T} \in \mathcal{T}_{k,h}, w_{k,h|_{\Gamma_{k,\Delta}}} = 0\}, \subset H_0^1(\Omega_k)$$

of (piecewise) linear conforming finite elements in the domain Ω_k associated with the mesh $\mathcal{T}_{k,h}$. We also introduce the space $W_{k,h}$ of (continuous) functions defined on the boundary $\mathcal{B}_k$ by the finite element basis $\{b_{k,i}(x)\}_{i=1}^{M_k}$ (see (14), (15)). Finally, we set

$$X_h = \prod_{k=1}^{\kappa} X_{k,h}, \quad X_{h,0} = \prod_{k=1}^{\kappa} X_{k,h,0}, \quad W_h = \prod_{k=1}^{\kappa} W_{k,h}.$$

The Galerkin formulation of (25) then 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 $w_h \in X_{h,0}$ we have:

$$(u_h^{n+1}, w_h)_\Omega + \frac{\Delta_t^2}{4} a(u_h^{n+1}, w_h) - \frac{\Delta_t^2}{4}(\lambda_h^{n+1}, w_h)_\mathcal{B} = (u_h^n, w_h)_\Omega - \frac{\Delta_t^2}{4} a(u_h^n, w_h) + \frac{\Delta_t^2}{4}(\lambda_h^n, w_h)_\mathcal{B} + \Delta_t (v_h^n, w_h)_\Omega + \frac{\Delta_t^2}{4}(\tilde{f}^{n+1} + \tilde{f}^n, w_h)_\Omega \tag{26}$$

For every $k = 1, \ldots, \boldsymbol{\kappa}$, let $\{N_i^{\Omega_k}\}_{i \in \mathcal{S}_k}$ denote the set of finite element basis functions defined on the decomposition $\mathcal{T}_{k,h}$, where: $\mathcal{S}_k = \mathcal{S}_{I_k} \cup \mathcal{S}_{\mathcal{B}_k}$, $\mathcal{S}_{I_k}$ is the set of the internal mesh nodes of the polyhedron $\Omega_{k,\Delta}$ and $\mathcal{S}_{\mathcal{B}_k}$ is the set of the mesh nodes lying on the artificial boundary $\mathcal{B}_k$. Note that $b_{k,i} = N_i^{\Omega_k}|_{\mathcal{B}_k}, i \in \mathcal{S}_{\mathcal{B}_k}$. By properly reordering the unknown coefficients of $u_{k,h}^n := u_{h|_{\Omega_k}}^n$, we obtain the (unknown) vector $\mathbf{u}_k^n = [\mathbf{u}_{I_k}^n, \mathbf{u}_{\mathcal{B}_k}^n]^T$, whose two components $\mathbf{u}_{I_k}^n$ and $\mathbf{u}_{\mathcal{B}_k}^n$ represent the unknown values associated with the internal nodes of $\Omega_{k,\Delta}$ and with those on the boundary $\mathcal{B}_k$, respectively. Similarly for the vector $\mathbf{v}_k^n$, containing the unknown coefficients of $v_{k,h}^n$. Finally, we denote by $\boldsymbol{\lambda}_{\mathcal{B}_k}^n$ the unknown vector whose components are the coefficients of the approximant $\lambda_{\triangle_{x,k}}(\mathbf{x}_k, t_n)$ defined in (15).

To rewrite (26) in the matrix form, we consider the system of equations associated with a single computational domain $\Omega_{k,\Delta}$, $k = 1, \ldots, \boldsymbol{\kappa}$, which is given by

$$\left(\mathbf{M}^k + \frac{\Delta_t^2}{4}\mathbf{A}^k\right)\mathbf{u}_k^{n+1} - \frac{\Delta_t^2}{4}\mathbf{Q}^k\boldsymbol{\lambda}_{\mathcal{B}_k}^{n+1} = \left(\mathbf{M}^k - \frac{\Delta_t^2}{4}\mathbf{A}^k\right)\mathbf{u}_k^n + \frac{\Delta_t^2}{4}\mathbf{Q}^k\boldsymbol{\lambda}_{\mathcal{B}_k}^n + \Delta_t\mathbf{M}^k\mathbf{v}_k^n + \frac{\Delta_t^2}{4}\left(\tilde{\mathbf{f}}_k^{n+1} + \tilde{\mathbf{f}}_k^n\right) \tag{27}$$

where

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

The matrix elements

$$(\mathbf{M}^k)_{ij} = \int_{\Omega_k} N_i^{\Omega_k} N_j^{\Omega_k}, \quad (\mathbf{A}^k)_{ij} = \int_{\Omega_k} \nabla N_i^{\Omega_k} \cdot \nabla N_j^{\Omega_k}, \quad i, j \in \mathcal{S}_k$$

are those of the mass and stiffness matrices, respectively, while those of $\mathbf{Q}^k$ are given by

$$(\mathbf{Q}^k)_{ij} = \int_{\mathcal{B}_k} b_{k,i} b_{k,j}, \quad i \in \mathcal{S}_k,\ j \in \mathcal{S}^{\mathcal{B}_k}.$$

Equation (27) is finally coupled with

$$\mathbf{v}_k^{n+1} = \frac{2}{\Delta_t}(\mathbf{u}_k^{n+1} - \mathbf{u}_k^n) - \mathbf{v}_k^n \tag{28}$$

and with the discretized NRBC equation

$$\left(\frac{1}{2}\mathbf{I}_k + \mathbf{K}_0^{k,k}\right)\mathbf{u}_{\mathcal{B}_k}^{n+1} \quad + \quad \sum_{\substack{\ell=1\\ \ell\neq k}}^{\boldsymbol{\kappa}} \mathbf{K}_0^{k,\ell}\mathbf{u}_{\mathcal{B}_\ell}^{n+1} + \sum_{\ell=1}^{\boldsymbol{\kappa}} \mathbf{V}_0^{k,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^{n+1}$$

$$= -\sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n} \mathbf{K}_{n+1-j}^{k,\ell}\mathbf{u}_{\mathcal{B}_\ell}^{j} - \sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n} \mathbf{V}_{n+1-j}^{k,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^{j} + \mathbf{I}_{u_0}^{k,n+1} + \mathbf{I}_{v_0}^{k,n+1} + \mathbf{I}_f^{k,n+1}.$$

Having set $\mu = \frac{\Delta_t^2}{4}$, and letting $k = 1, \ldots, \boldsymbol{\kappa}$, we obtain a final linear system $\mathbb{A}\mathbb{X} = \mathbb{B}$ having the following block structure:

$$\mathbb{A} = \begin{pmatrix} \begin{bmatrix} \mathbf{M}^{\mathbf{1}} + \mu\mathbf{A}^{\mathbf{1}} & -\mu\mathbf{Q}^1 \\ \frac{1}{2}\mathbf{I}_1 + \mathbf{K}_0^{1,1} & \mathbf{V}_0^{1,1} \end{bmatrix} & \begin{bmatrix} \mathbf{O} & \mathbf{O} \\ \mathbf{K}_0^{1,2} & \mathbf{V}_0^{1,2} \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{O} & \mathbf{O} \\ \mathbf{K}_0^{1,\boldsymbol{\kappa}} & \mathbf{V}_0^{1,\boldsymbol{\kappa}} \end{bmatrix} \\ \begin{bmatrix} \mathbf{O} & \mathbf{O} \\ \mathbf{K}_0^{2,1} & \mathbf{V}_0^{2,1} \end{bmatrix} & \begin{bmatrix} \mathbf{M}^{\mathbf{2}} + \mu\mathbf{A}^{\mathbf{2}} & -\mu\mathbf{Q}^2 \\ \frac{1}{2}\mathbf{I}_2 + \mathbf{K}_0^{2,2} & \mathbf{V}_0^{2,2} \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{O} & \mathbf{O} \\ \mathbf{K}_0^{2,\boldsymbol{\kappa}} & \mathbf{V}_0^{2,\boldsymbol{\kappa}} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} \mathbf{O} & \mathbf{O} \\ \mathbf{K}_0^{\boldsymbol{\kappa},1} & \mathbf{V}_0^{\boldsymbol{\kappa},1} \end{bmatrix} & \cdots & & \begin{bmatrix} \mathbf{M}^{\boldsymbol{\kappa}} + \mu\mathbf{A}^{\boldsymbol{\kappa}} & -\mu\mathbf{Q}^{\boldsymbol{\kappa}} \\ \frac{1}{2}\mathbf{I}_{\boldsymbol{\kappa}} + \mathbf{K}_0^{\boldsymbol{\kappa},\boldsymbol{\kappa}} & \mathbf{V}_0^{\boldsymbol{\kappa},\boldsymbol{\kappa}} \end{bmatrix} \end{pmatrix}$$

$$\mathbb{X} = \begin{pmatrix} \begin{bmatrix} \mathbf{u}_1^{n+1} \\ \boldsymbol{\lambda}_{\mathcal{B}_1}^{n+1} \end{bmatrix} \\ \begin{bmatrix} \mathbf{u}_2^{n+1} \\ \boldsymbol{\lambda}_{\mathcal{B}_2}^{n+1} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \mathbf{u}_{\boldsymbol{\kappa}}^{n+1} \\ \boldsymbol{\lambda}_{\mathcal{B}_{\boldsymbol{\kappa}}}^{n+1} \end{bmatrix} \end{pmatrix}$$

$$\mathbb{B} = \begin{pmatrix} \begin{bmatrix} \left(\mathbf{M}^1 - \mu\mathbf{A}^1\right)\mathbf{u}_1^n + \mu\mathbf{Q}^1\boldsymbol{\lambda}_1^n + \Delta_t\mathbf{M}^1\mathbf{v}_1^n + \mu\left(\tilde{\mathbf{f}}_1^{n+1} + \tilde{\mathbf{f}}_1^n\right) \\ -\sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n}\mathbf{K}_{n+1-j}^{1,\ell}\mathbf{u}_{\mathcal{B}_\ell}^j - \sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n}\mathbf{V}_{n+1-j}^{1,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^j + \mathbf{I}_{u_0}^{1,n+1} + \mathbf{I}_{v_0}^{1,n+1} + \mathbf{I}_f^{1,n+1} \end{bmatrix} \\ \begin{bmatrix} \left(\mathbf{M}^2 - \mu\mathbf{A}^2\right)\mathbf{u}_2^n + \mu\mathbf{Q}^2\boldsymbol{\lambda}_2^n + \Delta_t\mathbf{M}^2\mathbf{v}_2^n + \mu\left(\tilde{\mathbf{f}}_2^{n+1} + \tilde{\mathbf{f}}_2^n\right) \\ -\sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n}\mathbf{K}_{n+1-j}^{2,\ell}\mathbf{u}_{\mathcal{B}_\ell}^j - \sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n}\mathbf{V}_{n+1-j}^{2,\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^j + \mathbf{I}_{u_0}^{2,n+1} + \mathbf{I}_{v_0}^{2,n+1} + \mathbf{I}_f^{2,n+1} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \left(\mathbf{M}^{\boldsymbol{\kappa}} - \mu\mathbf{A}^{\boldsymbol{\kappa}}\right)\mathbf{u}_{\boldsymbol{\kappa}}^n + \mu\mathbf{Q}^{\boldsymbol{\kappa}}\boldsymbol{\lambda}_{\boldsymbol{\kappa}}^n + \Delta_t\mathbf{M}^{\boldsymbol{\kappa}}\mathbf{v}_{\boldsymbol{\kappa}}^n + \mu\left(\tilde{\mathbf{f}}_{\boldsymbol{\kappa}}^{n+1} + \tilde{\mathbf{f}}_{\boldsymbol{\kappa}}^n\right) \\ -\sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n}\mathbf{K}_{n+1-j}^{\boldsymbol{\kappa},\ell}\mathbf{u}_{\mathcal{B}_\ell}^j - \sum_{\ell=1}^{\boldsymbol{\kappa}}\sum_{j=0}^{n}\mathbf{V}_{n+1-j}^{\boldsymbol{\kappa},\ell}\boldsymbol{\lambda}_{\mathcal{B}_\ell}^j + \mathbf{I}_{u_0}^{\boldsymbol{\kappa},n+1} + \mathbf{I}_{v_0}^{\boldsymbol{\kappa},n+1} + \mathbf{I}_f^{\boldsymbol{\kappa},n+1} \end{bmatrix} \end{pmatrix}$$

Remark 4.1. *We have tested the stability and convergence of the proposed method. From the results we have obtained, it appears that unconditional stability and convergence is guaranteed, in the given time interval* $[0,T]$*, as long as all the integrals required by the (discretized) NRBC are exactly evaluated. When these integrals are evaluated with a given accuracy* ϵ*, then, for any chosen space discretization parameter* Δ_x*, there exists a time step barrier* $\Delta_0 = \Delta_0(\epsilon)$*, with* $\Delta_0 \to 0$ *as* $\epsilon \to 0$*, such that for* $\Delta_t < \Delta_0$ *instabilities arise before reaching the final time* T*.*

For example, if in the first example of Section 6, where $T = 10$*, we choose* $N = 32, 64, 128, 256, 512, 1024$ *and consider the first two levels of space discretization, no instability appears when we perform the required integration by using a* ν*-point Gauss-Legendre rule, with* $\nu = 2, 2, 2, 8, 12, 20$ *in the case of the first level of discretization, and* $\nu = 2, 2, 2, 2, 8, 12$ *in the case of the second level. In both cases, for example, if we take* $\nu = 4$ *when* $N = 512, 1024$*, instabilities appear before reaching the endpoint* $T = 10$*. Thus, the efficient evaluation of the above mentioned integrals is a crucial point for the success of the proposed NRBC. Note however that, as* Δ_x *decreases, the accuracy given by the chosen* ν*-point rule increases, hence the value of* Δ_0 *decreases.*

5. An exact solution for a wave equation problem in the exterior of the unit sphere

To test the convergence of the proposed numerical scheme, and in particular the effectiveness of the proposed NRBC, it is important to have a reference solution at hand. We start recalling that the following single-layer potential representation associated to (1),

$$u(\mathbf{x},t) = \int_0^t \int_\Gamma G(||\mathbf{x}-\mathbf{y}||, t-\tau)\varphi(\mathbf{y},\tau)\, d\Gamma_{\mathbf{y}}\, d\tau + I_{u_0}(\mathbf{x},t) + I_{v_0}(\mathbf{x},t) + I_f(\mathbf{x},t) \quad \mathbf{x} \in \Omega^e \quad (29)$$

holds (see [5]). We remark that the density function φ in (29) is the solution of the following Time Dependent Boundary Integral Equation (TDBIE)

$$\int_\Gamma \int_0^t G(r, t-\tau)\varphi(\mathbf{y},\tau)\, d\tau d\Gamma_{\mathbf{y}} = g(\mathbf{x},t) - I_{u_0}(\mathbf{x},t) - I_{v_0}(\mathbf{x},t) - I_f(\mathbf{x},t), \quad \mathbf{x} \in \Gamma, \quad (30)$$

and represents the jump of the normal derivative of u along Γ. Once the density function is known, the solution u at any point in the infinite domain Ω^e is defined by computing the associated potential (29). For a generic problem, in order to obtain a reference solution, we will first solve (30) and then compute (29) by using a very fine space/time refinement.

In the special case of an homogeneous problem ($u_0 = v_0 = f = 0$) defined on the surface $\Gamma = \mathbb{S}^2$ of the unit sphere, it is possible to derive an explicit analytic representation for the solution of problem (1). In [22] the authors derive the analytic expression of the solution φ of the boundary integral equation (30). We follow here the same idea to get the solution u in the exterior of the unit sphere. For convenience of the reader, we report the main ingredients. Let $\hat{u}(\mathbf{x}, s)$ (the Laplace transform of $u(\mathbf{x}, t)$) be the solution of the wave equation reformulated in the frequency domain (Helmholtz equation). In [22] the authors choose the boundary Dirichlet datum $g(\mathbf{x}, t) = g(t)Y_n^m(\mathbf{x})$, where Y_n^m are the spherical harmonics of degree n

and order m, with $n = 0, 1, \ldots$ and $-n \leq m \leq n$. They prove that the density function $\varphi(\mathbf{x}, t) = \varphi(t) Y_n^m(\mathbf{x})$ satisfies the relation $\hat{g}(s) = \lambda_n(s)\hat{\varphi}(s)$, being λ_n the eigenvalues of the single layer potential Helmholtz operator associated to the Helmholtz equation. Following the same idea, to get the solution of the Dirichlet problem for the time dependent wave equation in the exterior of the unit sphere, we write

$$u(\mathbf{x}, t) = u(t) Y_n^m(\xi), \quad \text{with} \quad \mathbf{x} = r\xi \in \mathbb{R}^3,\ \xi \in \Gamma = \mathbb{S}^2 \tag{31}$$

where, for notational simplicity, we denote by $u(t)$ a function which depends also on r. Then, we use the property that the solution $\hat{u}$ of the associated Helmholtz problem satisfies the relationship

$$\hat{u}(s) = \hat{g}(s) \frac{h_n^{(1)}(isr)}{h_n^{(1)}(is)}, \tag{32}$$

where $h_n^{(1)}$ denote the spherical Bessel functions of third kind (see [14]). By computing the inverse Laplace transform of (32), we retrieve an analytic expression for the solution of the homogeneous wave equation. This is given by representation (31), with

$$u(t) = \mathcal{L}^{-1}\left(\hat{g} \frac{h_n^{(1)}(ir\cdot)}{h_n^{(1)}(i\cdot)} \right)(t) = g * \mathcal{L}^{-1}\left(\frac{h_n^{(1)}(ir\cdot)}{h_n^{(1)}(i\cdot)} \right)(t).$$

By using established properties of the Bessel functions (see in particular [1] (Sec. 10.1.1) and [15] (formula in Sec. 8.466)), and following [22], we easily get

$$\begin{aligned} h_n^{(1)}(s) &= \sqrt{\frac{\pi}{2s}} H_{n+\frac{1}{2}}^{(1)}(s) = \sqrt{\frac{\pi}{2s}} \sqrt{\frac{2}{\pi s}} (i)^{-(n+1)} e^{is} \sum_{k=0}^{n} (-1)^k \frac{(n+k)!}{k!(n-k)!} \frac{1}{(2is)^k} \\ &= \frac{1}{s} (i)^{-(n+1)} e^{is} \sum_{k=0}^{n} y_n\left(-\frac{1}{is}\right), \end{aligned}$$

where $y_n(s) := \sum_{k=0}^{n} (n,k) s^k$, $(n,k) := \frac{(n+k)!}{2^k k!(n-k)!}$ and $H_\nu^{(1)}$ is the first kind Hankel function of order ν. Using this latter expression we obtain:

$$\frac{h_n^{(1)}(irs)}{h_n^{(1)}(is)} = \frac{1}{r} e^{-(r-1)s} \frac{y_n(\frac{1}{rs})}{y_n(\frac{1}{s})}.$$

In the simplest case $n = 0$ we have $y_n(s) = 1$, wherefrom $\frac{h_0^{(1)}(irs)}{h_0^{(1)}(is)} = \frac{1}{r} e^{-(r-1)s}$. By using classical properties of the inverse Laplace transform, we easily get

$$\mathcal{L}^{-1}\left(\frac{h_0^{(1)}(ir\cdot)}{h_0^{(1)}(i\cdot)} \right)(t) = \frac{1}{r} H(t-(r-1)) \delta(t-(r-1)),$$

being H the Heaviside function. Thus, we obtain

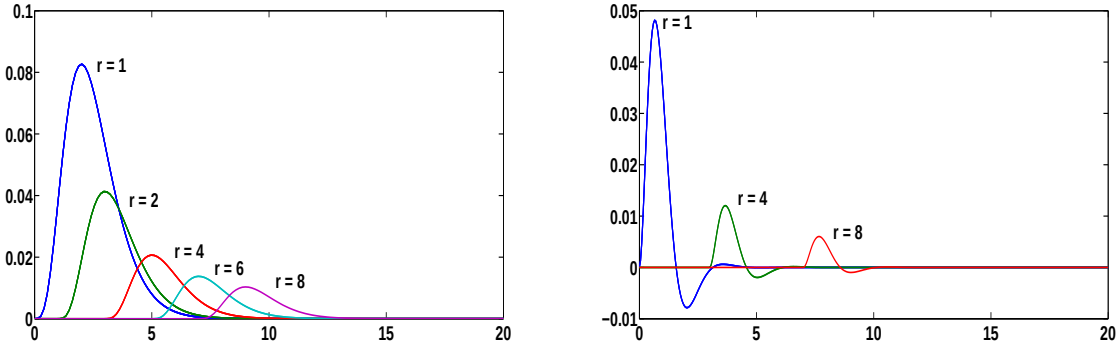
$$u(t) = \frac{1}{r}\int_0^t g(t-\tau)H(\tau-(r-1))\delta(\tau-(r-1))d\tau$$
$$= \begin{cases} 0 & \text{if} \quad t \le r-1 \\ \frac{1}{r}g(t-(r-1)) & \text{otherwise.} \end{cases} \tag{33}$$

Since $Y_0^0(\mathbf{x}) = 1/(2\sqrt{\pi})$, the solution reads

$$u(\mathbf{x},t) = \frac{1}{2\sqrt{\pi}}u(t) \tag{34}$$

with $u(t)$ defined by (33). Note that this latter is a radial function. In Figure 7 we show the behavior of $u(\mathbf{x},t)$, with respect to the time variable, corresponding to the Dirichlet data $g(\mathbf{x},t) = \frac{1}{2\sqrt{\pi}}t^4e^{-2t}$ (left plot) and $g(\mathbf{x},t) = \frac{1}{2\sqrt{\pi}}\sin(2t)te^{-2t}$ (right plot).

Figure 7: Exact solutions of the homogeneous wave equation for $\Gamma = \mathbb{S}^2$ corresponding to the choice of the Dirichlet data $g(\mathbf{x},t) = \frac{1}{2\sqrt{\pi}}t^4e^{-2t}$ (left plot) and $g(\mathbf{x},t) = \frac{1}{2\sqrt{\pi}}\sin(2t)te^{-2t}$ (right plot). Each curve corresponds to its behavior with respect to time at points having distance r from the origin.



6. Numerical results

In the numerical tests we will perform, the discretization of the three dimensional spatial domain is generated by using the Freefem++ library. In particular we use the TetGen software, which allows to generate the tetrahedral mesh of the domain Ω starting from that we define on its boundary (see [21]). The computational domains that we will consider for our numerical examples are three dimensional shells bounded internally by a surface Γ and externally by a surface $\mathcal{B}$. The model that represents the shell, which is constructed by TetGen, is a three dimensional Piecewise Linear Complex (PLC). We recall that TetGen does not generate the surface mesh of the PLC; this, together with the choice of the maximum diameter of the triangles generated to approximate the surfaces, must be given as input by the user. Moreover, TetGen does not take the curvature of the surface into account. In what follows, n_T denotes the number of tetrahedra of the decomposition of the computational domain Ω, while nt_Γ and $nt_\mathcal{B}$ denote the number of triangles belonging to the boundaries Γ and $\mathcal{B}$, respectively.

For simplicity, all the problems we will consider have trivial initial data u_0 and v_0, since the evaluation of the corresponding integrals would require a further ad hoc discussion.

Example 1. As a first example, we consider Problem (1) in the case where Ω^i consists of a single scatterer ($\boldsymbol{\kappa} = 1$). We assume that the source f is zero throughout the infinite exterior domain Ω^e. The boundary Γ is the unit sphere, where we prescribe the Dirichlet condition $g(\mathbf{x}, t) = \frac{1}{2\sqrt{\pi}} t^4 e^{-2t}$ for all $t \geq 0$. The solution of this problem is a radial function and its analytical expression is given by (34) and (33).

We choose a spherical artificial boundary $\mathcal{B}$ having radius $R = 2$, so that Ω is the shell bounded internally by Γ and externally by $\mathcal{B}$. In Figure 8 we show a section of three different refinements of the shell. Here, and in the following examples, to evaluate the integrals that appear in the elements of the matrices $\mathbf{V}_{\mathbf{m}}^{\mathbf{k},\ell}$ and $\mathbf{K}_{\mathbf{m}}^{\mathbf{k},\ell}$ (see (16) and (17)), we first map each triangle of the approximated artificial surface, where the integrand in non null, into the (standard) reference triangle; then we introduce the polar coordinates and apply a ν-points Gauss-Legendre quadrature rule, with $\nu = 4$, to each one-dimensional integral. We remark that, because of the discrepancy between the artificial boundary $\mathcal{B}_\Delta$ of the PLC Ω_Δ and that ($\mathcal{B}$) of the NRBC, and of the presence of the normal derivative in the kernel $K^{\boldsymbol{\mathcal{K}}}$ (see (12)), the solution obtained by integrating over the plain triangles of $\mathcal{B}_\Delta$ is not satisfactory for coarse spatial discretizations and presents spurious reflections that disappear with spatial refinements. Therefore, we perform the integration over the curvilinear triangles, which is simply obtained by projecting a point belonging to the plane triangle to the corresponding point of the surface (see for example Figures 14 and 15 where we compare both types of approximation).

In Figure 9 we compare the exact solution with the approximate solution we have obtained at an internal point $P \approx (1.5, 0, 0)$ (left plot) and at a point $P \approx (2, 0, 0)$ that belongs to the boundary $\mathcal{B}_\Delta$ (right plot). The approximations have been obtained by decomposing the spherical shell into $n_T = 24224$ tetrahedra and the time interval $[0, 10]$ into $N = 100$ steps. Being the exact solution known, we also compute the absolute error

$$\text{Err} := \max_{0 \leq n \leq N} \|u(\cdot, t_n) - u_{\Delta_x}(\cdot, t_n)\|_{L^2(\Omega)}. \tag{35}$$

The numerical computation of the quantity

$$\|u(\cdot, t_n) - u_{\Delta_x}(\cdot, t_n)\|^2_{L^2(\Omega)} \approx \sum_{\mathcal{T} \in \mathcal{T}_h} \int_{\mathcal{T}} |u(\mathbf{x}, t_n) - u_{\Delta_x}(\mathbf{x}, t_n)|^2 d\mathbf{x}$$

has been obtained by applying on each tetrahedron $\mathcal{T}$ a 4-point quadrature rule of order 2 (see [13] for details).

In Table 1 we report the behavior of the absolute error with respect to the space and time mesh refinement. The successive spatial refinements have been obtained by halving each time the mesh size of both the internal (h_Γ) and external ($h_\mathcal{B}$) boundaries of the shell. The associated refinements of the Ω decomposition is automatically determined by the software TetGen. The above errors are essentially due to the FE computation. Those due to the NRBC discretization appear negligible. Indeed, if we compute, for example, all the needed integrals with higher accuracy, using a 16-point Gauss-Legendre rule, we obtain the same errors reported in Table 1.

In the following three examples, since the corresponding exact solutions are not known, to measure the accuracy of the approximations we obtain, we construct a reference "exact"

Figure 8: Example 1. A section of the shell having the unit sphere as internal boundary and the sphere of radius 2 as external one: three discretizations into n_T tetrahedra.

$n_T = 315$ $n_T = 24224$ $n_T = 40904$

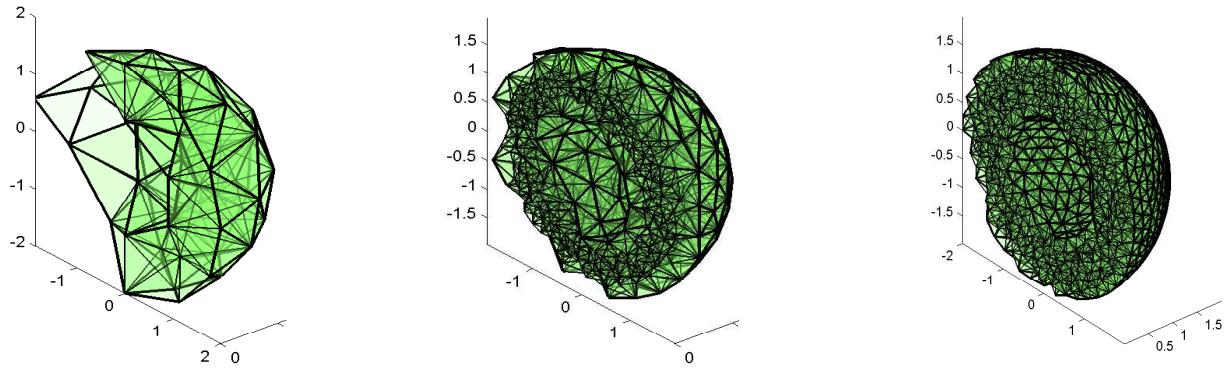


Figure 9: Example 1. Comparison between the exact solution and the approximate one at $P \approx (1.5, 0, 0)$ (left plot) and at $P \approx (2, 0, 0)$ (right plot), by using $n_T = 24224$ tetrahedra for the shell discretization and $N = 100$ time steps.

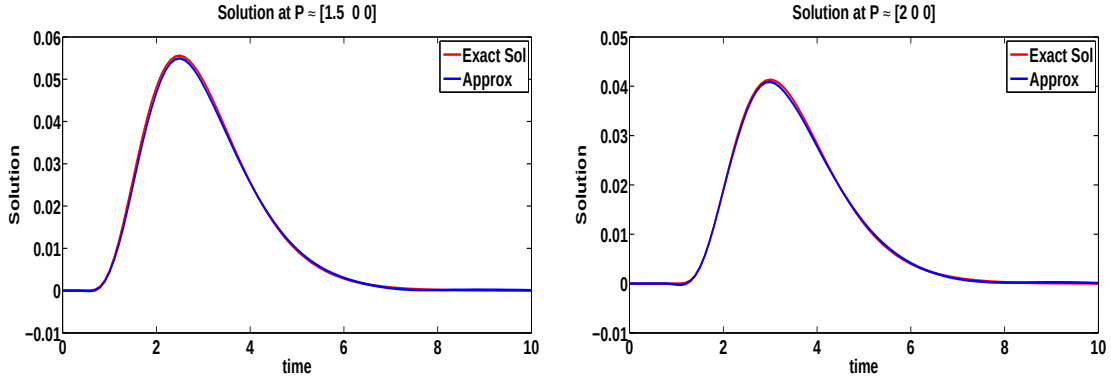


Table 1: Example 1. Behavior of the absolute error (35) with respect to the space and time mesh refinements, for $t \in [0, 10]$.

h_Γ	$h_\mathcal{B}$	nt_Γ	$nt_\mathcal{B}$	n_T	N	Err
.5	1	122	122	367	32	5.05E-02
.25	.5	440	440	27023	64	4.41E-03
.125	.25	1772	1772	44892	128	2.63E-03

solution as follows: we first compute the density function by applying the Lubich-collocation method to equation (30) on a fine space and time discretization; then, the solution at any point in the infinite domain Ω^e is retrieved by computing the associated potential (29). This

solution will be denoted by the acronym BEM.

Example 2. Case a). We consider the case of a wave generated by a single source and impinging upon a single body acting as a soft obstacle. The data of the problem are $u_0 = 0$, $v_0 = 0$, $g = 0$ and $f \neq 0$. We recall that, if a source is far from the area of interest, the existing local NRBCs would require to take a much larger domain Ω, to include the source, thus wasting computational time and space memory. This is not the case when we use our NRBC. Indeed, when a source is located in the residual domain $\mathcal{D}$, the source action is taken into account by the proposed artificial boundary condition.

To simplify the computation of the volume term I_f, 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, taking into account the presence of the delta Dirac function in the expression of the fundamental solution (2), we deduce the following simple form of the volume integral I_f defined in (6):

$$I_f(\mathbf{x}, t) = \begin{cases} \frac{h(t-\|\mathbf{x}-\mathbf{x}_0\|)}{4\pi\|\mathbf{x}-\mathbf{x}_0\|}, & \text{for all} \quad \mathbf{x} : \|\mathbf{x} - \mathbf{x}_0\| < t \\ 0 & \text{otherwise.} \end{cases} \tag{36}$$

We place the source $f(\mathbf{x}, t) = t^2 \sin(4t)e^{-t}\delta(\mathbf{x}-\mathbf{x}_0)$ at the point $\mathbf{x}_0 = (5, 0, 0)$; Γ and $\mathcal{B}$ are the surfaces of the spheres of radius $r_0 = .25$ and $R = .5$, respectively, both centered at the origin (see Figure 10 left plot). In Figure 10, right plot, we compare the approximate and the reference solution (BEM) at the point $P \approx (.5, 0, 0)$ belonging to the artificial boundary $\mathcal{B}_\Delta$. The reference solution has been obtained by a discretization of Γ into $nt_\Gamma = 122$ triangles and $N = 1E + 03$ time steps. The approximate solution has been obtained by using a decomposition of the spherical shell into $n_T = 226$ tetrahedra and $N = 500$ time steps. We note that the reference and the approximate solutions perfectly match and that the wave is null until the signal f reaches the point P at the time $t \approx 4.5$. Moreover, because of the presence of the term e^{-t} in the source f, the wave vanishes for large times.

Figure 10: Example 2. Case a). Benchmark configuration: the scatterer surrounded by the artificial boundary and the external source f (left plot). Comparison between the reference solution and the approximate one at $P \approx (.5, 0, 0)$ (right plot).

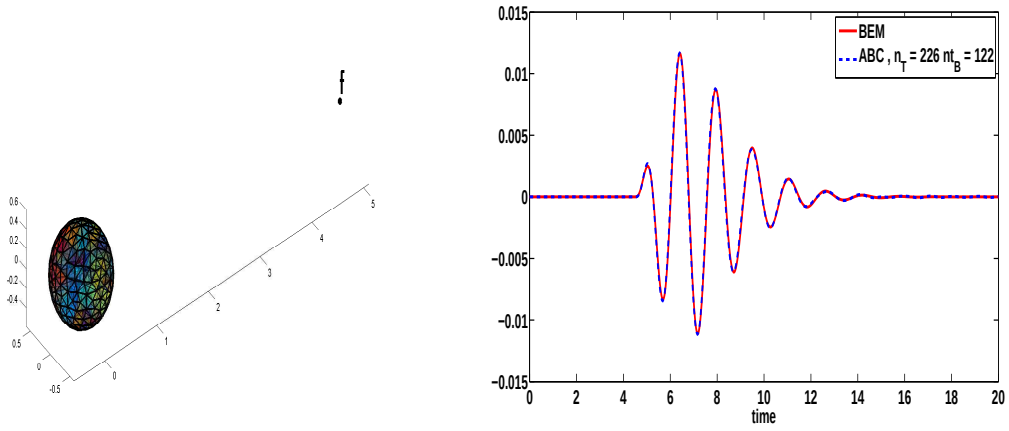
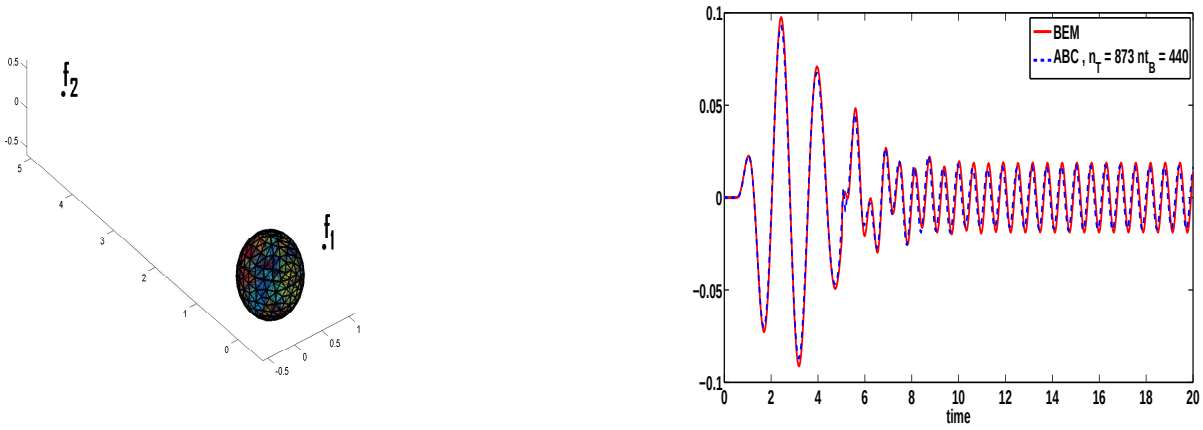


Figure 11: Example 2. Case b). Benchmark configuration: the scatterer surrounded by the artificial boundary and the external sources f_1 and f_2 (left plot). Comparison between the reference solution and the approximate one at $P \approx (.5, 0, 0)$ (right plot)



Case b). The treatment of a single source, external to the finite computational domain, can be easily extended to several sources, compactly supported and having disjoint supports. In this case, the volume term I_f that appears in the NRBC consists of the sum of several volume terms I_{f_i}, each of which is generated by the corresponding source f_i. In the next example, in the same setting of the case a), we consider the two point sources

$$\begin{aligned} f_1(\mathbf{x}, t) &= h_1(t)\delta(\mathbf{x} - \mathbf{x}_1) = t^2 \sin(4t) e^{-t} \delta(\mathbf{x} - \mathbf{x}_1) \\ f_2(\mathbf{x}, t) &= h_2(t)\delta(\mathbf{x} - \mathbf{x}_2) = \cos(10t)\delta(\mathbf{x} - \mathbf{x}_2), \end{aligned}$$

located in $\mathbf{x}_1 = (1, 0, 0)$ and $\mathbf{x}_2 = (0, 5, 0)$, respectively. It is easy to check that the volume term of the NRBC is given by $I_f = I_{f_1} + I_{f_2}$, where I_{f_i}, $i = 1, 2$ are given by (36).

In Figure 11 we show the history of the wave at the the point $P \approx (.5, 0, 0)$ belonging to the artificial boundary $\mathcal{B}_\Delta$. Also in this case the reference and the approximate solutions match. At a time $t \approx 0.5$ the effect of the first source f_1 is visible at P, and when the effect of this source vanish (because of the presence of the e^{-t} term) the oscillatory behavior of the wave is due to the persistence of the signal f_2.

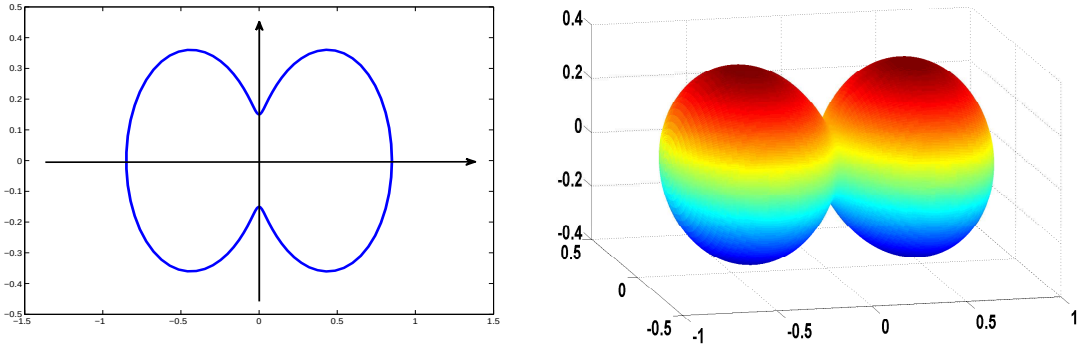
Example 3. The ductility of the artificial boundary is another important characteristic of the proposed NRBC. In this example we consider a single scatterer whose boundary Γ is the nut shape surface obtained by rotating the one dimensional curve

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

where $\rho(\theta) = c(1 + e\cos(n\theta))$, $c = 0.5$, $e = 0.7$, $n = 2$ and $\theta \in [0, 2\pi]$, along the x-axis (see Figure 12). The parametric representation of the three dimensional nut is therefore given by

$$\begin{aligned} X(\theta, \varphi) &= x(\theta) \\ Y(\theta, \varphi) &= \cos(\varphi) y(\theta) \\ Z(\theta, \varphi) &= \sin(\varphi) y(\theta). \end{aligned} \tag{37}$$

Figure 12: Example 3. The nut shape curve for the choice $c = 0.5$, $e = 0.7$ and $n = 2$ (left plot) and the corresponding three dimensional surface (right plot)



We prescribe on Γ the Dirichlet datum $g(\mathbf{x}, t) = \frac{1}{2\sqrt{\pi}} t^4 e^{-2t}$ for all $t \geq 0$. In order to study the behavior of the solution in a thin region surrounding Γ, we choose the artificial boundary $\mathcal{B}$ having the same shape of Γ, that is the nut shape surface obtained with the choice of the parameters $c = 1$, $e = 0.7$ and $n = 2$. A cross section of the resulting finite computational domain, bounded internally by Γ and externally by $\mathcal{B}$, and a decomposition of it into tetrahedra, is represented in Figure 13. In Figure 14 we plot the behavior of the BEM reference solution and of the approximate ones at a point $P \approx (1.7, 0, 0)$ of the artificial boundary $\mathcal{B}_\Delta$. The approximations are obtained with different refinements of the spatial mesh and for a fixed refinement of the time interval $[0, 10]$ into $N = 100$ steps. In particular we compare the approximation obtained when the matrix elements of the matrices $\mathbf{V_m}$ and $\mathbf{K_m}$ are computed by integrating over the plain triangles ("ABC plain" acronym for the solution, right plot) and the one obtained when the matrix elements are computed by integrating over the curvilinear triangles ("ABC curv" acronym for the solution,left plot). We note that, with the same mesh refinement, the second approach produces a more accurate approximation with respect to the first one. In Figure 15 we compare the reference and approximate solutions at the $\mathcal{B}_\Delta$ point $P \approx (0, 0.3, 0)$, which is placed in the external surrounding of the nut boundary Γ.

Figure 13: Example 3. A decomposition of the finite computational domain into tetrahedra

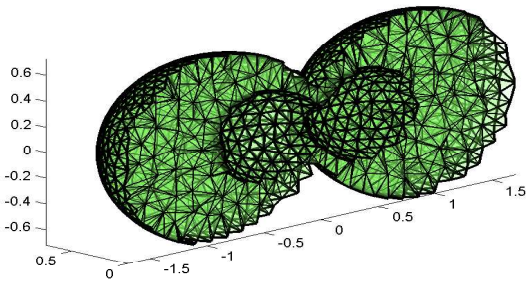


Figure 14: Example 3. Comparison between the reference solution and the approximate one at $P \approx (1.7, 0, 0)$ by curvilinear triangles (left plot) and plain triangles (right plot), $N = 100$.

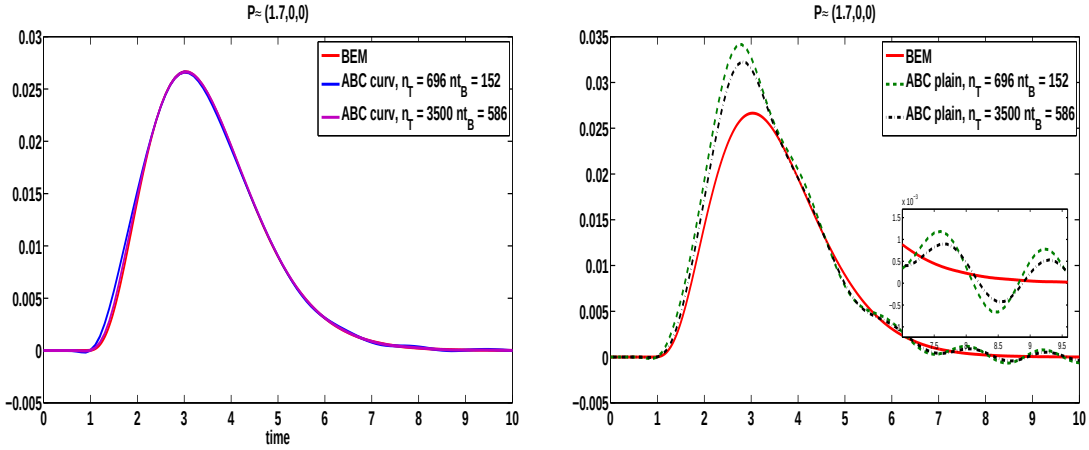
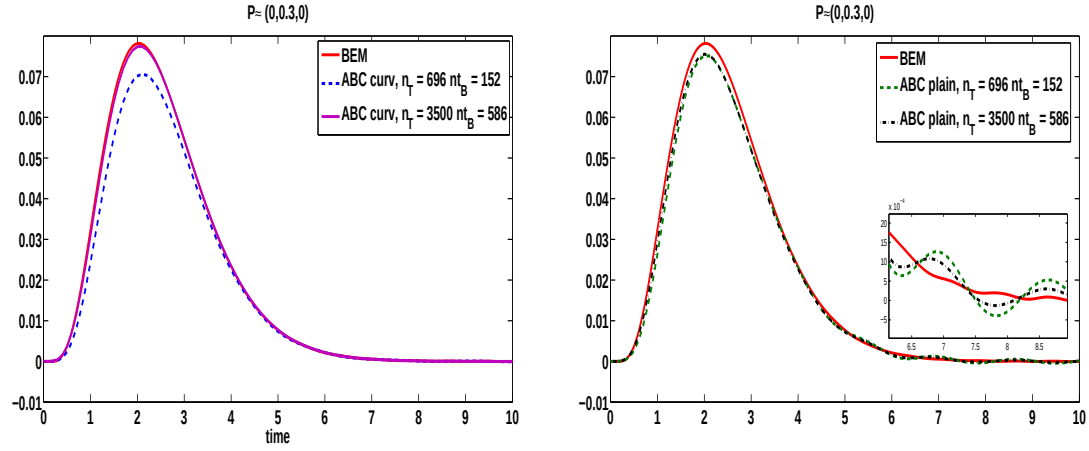


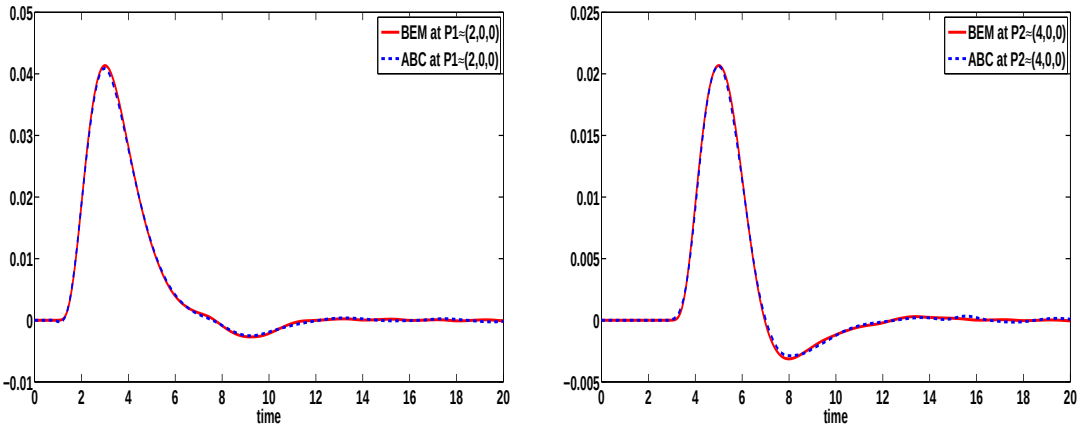
Figure 15: Example 3. Comparison between the reference solution and the approximate one at $P \approx (0, 0.3, 0)$ by curvilinear triangles (left plot) and plain triangles (right plot), $N = 100$.



Example 4. In this last example, we consider a multiple scattering problem. In particular we consider two disjoint spherical bodies whose boundaries Γ_1 and Γ_2 are spherical surfaces both of radius $r = 1$, centered at the origin and at $C = (6, 0, 0)$, respectively. As in Example 1, we assume that the source f and the initial data u_0 and v_0 are zero throughout the infinite exterior domain Ω^e. For all $t \geq 0$, we prescribe the Dirichlet condition $g(\mathbf{x}, t) = \frac{1}{2\sqrt{\pi}} t^4 e^{-2t}$ on Γ_1, and set $g(\mathbf{x}, t) = 0$ on Γ_2. Therefore, the second body acts as a soft reflecting obstacle. We choose the artificial boundary $\mathcal{B}_1$ as the spherical surface centered at the origin and with radius $R_1 = 2$, and the artificial boundary $\mathcal{B}_2$ as the spherical surface centered at C and with radius $R_2 = 2$.

In Figure 16 we compare the BEM reference solution and the corresponding approximate one at the points $P_1 \approx (2, 0, 0), P_1 \in \mathcal{B}_{1,\Delta}$ and $P_2 \approx (4, 0, 0), P_2 \in \mathcal{B}_{2,\Delta}$. Each reference solution has been obtained by a discretization of both surfaces into $nt_{\mathcal{B}_1} = nt_{\mathcal{B}_2} = 122$ triangles, and by performing $N = 1E + 03$ (equal) time steps in the interval $[0, 20]$. The approximate solutions are given by a decomposition of both spherical shells into $n_T = 24224$ tetrahedra and $N = 200$. It can be noticed that the solution at P_1 coincides with the exact solution represented in Figure 9 (right plot) until the time instant $t \approx 7$, when the effect of the second obstacle comes into play.

Figure 16: Example 4. Comparison between the reference solution and the approximate one at $P \approx (2, 0, 0)$ (left plot) and at $P \approx (4, 0, 0)$ (right plot), corresponding to the choice $n_T = 24224$ and $N = 200$.



7. Conclusions

In [16] the authors construct a fully local ABC, to deal with 3D multi-scattering problems. As for all local conditions, sources must be included in the computational domain. Moreover, the shape of artificial boundary the authors consider is that of a spherical surface, one for each obstacle, although they remark that ellipsoidal ones could also be taken. Its computational cost is $O(N_{\mathcal{B}} N)$. We must however remark that this ABC requires, as stated by the authors, a number of auxiliary functions that must be judiciously chosen, depending on the problem or desired accuracy. Furthermore, a local spline interpolation on the obstacle boundaries is also required.

In the examples they present, the proposed ABC is coupled with a finite-difference scheme. It is however not clear how their approach should be applied in the case of obstacles having more general shapes, and with the finite difference method replaced by the FE one.

Our approach is more general. Obstacles and artificial boundaries can have any (smooth) shape. Furthermore we can even have multi sources, which do not have to be necessarily included in the computational domain. The NRBC naturally takes into account their effects. One does not have to separate incoming waves from outgoing ones. The NRBC will be transparent for each one of them. Finally, we also mention that in principle, we could even have obstacles which are rotating independently from each other. Indeed, some numerical testing we have already performed in the 2D case have given very promising results.

This generality has however a cost, both in terms of CPU and space memory, which is certainly higher than that of the above mentioned local condition. But we believe that a deeper investigation on these aspects should lead to further savings. The goal is to have a computational cost close to that of the associated FEM.

Besides the computational cost and the space memory, another possible drawback of the proposed NRBC is that the time interval of integration must be fixed in advance. Moreover, if the computation of integrals defining the NRBC is not performed with the needed accuracy, as pointed out in Remark 4.1, instabilities might arise before reaching the final instant T. But increasing the number of quadrature nodes means to increase the NRBC computational cost. Nevertheless, in our opinion there is still room for reducing these drawbacks, including the computational complexity and the working space. This includes the use of discrete convolution quadratures alternative to those of Lubich (see [4], [22], [8]), which should allow the construction of sparse $\mathbf{K}_m$ and $\mathbf{V}_m$ matrices, with the position of the non zero elements known a priori, but at the cost of loosing the FFT benefits; the use of higher order Lubich convolution rules (see for example [19], [2]); and finally the use of time integration formulas which do not require to fix a priori the final time instant T (see [23]). Finally, we recall that very recently the use of a multigrid strategy has been examined, to reduce the overall computational cost of the BIE discretization (see [7]). However, the use of these strategies for reducing the computational cost of our NRBC are still at an early stage and need further investigation.

Acknowledgements

The authors are grateful to the referees for their valuable comments.

References

[1] M. Abramowitz and I. A. Stegun, Handbook of Mathematical Functions, National Bureau of Standards, U.S. Department of Commerce, Applied Mathematics Series 55, 1967.

[2] Banjai, L. and Kachanovska, M., Sparsity of Runge-Kutta convolution weights for the three-dimensional wave equation, BIT 54(4), 2014, 901-936.

[3] D.A. Bini, B. Meini, Fast algorithms with applications to Markov chains and queueing models, in:*Fast Reliable Algorithms for Matrices with Structure* (T. Kailath, A.H. Sayed, eds.), SIAM, Philadelphia, PA, 1999, 211-243.

[4] P.J. Davis, D.B. Duncan, Convolution-in-time approximations of time domain boundary integral equations, SIAM J. Sci. Comput. 35(1), 2013, B43-B61.

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

[6] S. Falletta, G. Monegato, An exact non reflecting boundary condition for 2D time-dependent wave equation problems. Wave Motion 51 (2014), no. 1, 168-192.

[7] S. Falletta, S. Sauter, The panel-clustering method for the wave equation in two spatial dimensions, submitted.

[8] S. Falletta, L. Scuderi, A new boundary element integration strategy for retarded potential boundary integral equations, submitted.

[9] D. Givoli, Numerical Methods for Problems in Infinite Domains, Elsevier, Amsterdam, 1992.

[10] D. Givoli, Recent advances in the DtN FE method, Archives of Computational Methods in Engineering 6 (1999) 71-116.

[11] D. Givoli, High-order local non-reflecting boundary conditions: A review, Wave Motion 39 (2004) 319-326.

[12] M.J. Grote, Ch. Kirsch, Nonreflecting boundary condition for time-dependent multiple scattering, J. Comput. Phys. 221 (2007) 41-62.

[13] P. Keast, Moderate-degree tetrahedral quadrature formulas, Comp. Meth. Appl. Mech. Engrg., 55 (1986), 339–348.

[14] S.N. Chandler-Wilde, P. Monk, Wave-number-explicit bounds in time-harmonic scattering. SIAM J. Math. Anal. 39 (2008), no. 5, 1428-1455.

[15] I. Gradshteyn, I. Ryzhik, Table of Integrals, Series and Products. Academic Press, 1965.

[16] M.J. Grote, I. Sim, Local nonreflecting boundary condition for time-dependent multiple scattering, J. Comput. Phys. 230 (2011) 3135-3154.

[17] W. Hackbusch, W. Kress, S. Sauter, Sparse convolution quadrature for time domain boundary integral formulations of the wave equation, IMA J. Numer. Anal. 29 (2009) 158-179.

[18] Ch. Lubich, Convolution quadrature and discretized operational calculus. I, Num. Math. 52 (1988) 129-145.

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

[20] G. Monegato, L. Scuderi, A space-time BIE method for 2D mixed wave equation problems, submitted.

[21] F. Hecht, New development in FreeFem++, J. Numer. Math. 20 (2012), no. 3-4, 251-265.

[22] S. Sauter, A. Veit, A Galerkin method for retarded boundary integral equations with smooth and compactly supported temporal basis functions, Numer. Math., 123 (2013), no. 1, 145–176.

[23] M. Lopez-Fernandez, S. Sauter, Generalized convolution quadrature with variable time stepping IMA J. Numer. Anal. 33(4), 2013, 1156-1175.

[24] A. Taflove, S.C. Hagness, Computational Electrodynamics. The finite-difference time-domain method, Hartech House, Boston-London, 2005.