Wavelets and convolution quadrature for the efficient solution of a 2D space-time BIE for the wave equation

Wavelets and convolution quadrature for the efficient solution of a 2D space-time BIE for the wave equation *

S. Bertoluzza†, S. Falletta‡, L. Scuderi§

Abstract

We consider a wave propagation problem in 2D, reformulated in terms of a Boundary Integral Equation (BIE) in the space-time domain. For its solution, we propose a numerical scheme which is based on a second order Lubich discrete convolution quadrature formula for the discretization in time, coupled with a classical Galerkin method in space. It is known that the main advantage of the Lubich formula is the use of the FFT algorithm to retrieve the discretization in time of the integral operators with a computational complexity of order $\mathcal{R}\log\mathcal{R}$, $\mathcal{R}$ being twice the total number of time steps performed. On the other hand, the discretization in space leads in general to a quadratic complexity, hence the global working storage required is $M^2\mathcal{R}/2$, where M is the number of grid points chosen on the domain boundary.

To reduce the complexity in space, we consider here approximant functions of wavelet type. According to the properties of the wavelet basis, it turns out that the discretized integral operators have a rapid decay to zero with respect to the time, and the overwhelming majority of the associated matrix entries assume negligible values. Based on an a priori estimate of the decaying behavior in time of the matrix entries, we devise a *time downsampling strategy* that allows to compute only the elements which are significant with respect to a prescribed tolerance. Such approach allows to obtain highly sparse matrices by a downsampled FFT algorithm with a computational complexity of order $\overline{\mathcal{R}}\log\overline{\mathcal{R}}$, where $\overline{\mathcal{R}} \ll \mathcal{R}$.

KEY WORDS: wave equation; space-time boundary integral equations; multiresolution analysis; downsampling; numerical methods

1 Introduction

Let $\Omega^e = \mathbb{R}^2 \setminus \overline{\Omega^i}$ be the complement of a bounded rigid obstacle $\Omega^i \subset \mathbb{R}^2$, having a closed smooth boundary Γ. We consider the following exterior Dirichlet problem for

*This work was supported by GNCS-INDAM 2016 research program: Accoppiamento FEM-BEM non conforme mediante tecniche di decomposizione di dominio di tipo mortar, and by GNCS-INDAM 2017 research program: Nuove tecniche numeriche per la risoluzione di problemi evolutivi mediante il metodo degli elementi di contorno.

†I.M.A.T.I - CNR, Pavia , Italy. Email: silvia.bertoluzza@imati.cnr.it

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

§Dipartimento di Scienze Matematiche, Politecnico di Torino, Italy. Email: letizia.scuderi@polito.it

the wave equation:

$$\begin{cases} \frac{1}{c^2}u_{tt}(\mathbf{x},t) - \Delta u(\mathbf{x},t) &= 0 & \text{in } \Omega^e \times (0,T) \\ u(\mathbf{x},t) &= g(\mathbf{x},t) & \text{on } \Gamma \times (0,T) \\ u(\mathbf{x},0) &= 0 & \text{in } \Omega^e \\ u_t(\mathbf{x},0) &= 0 & \text{in } \Omega^e, \end{cases} \tag{1}$$

where c represents the wave propagation velocity.

It is well known that the following single-layer potential representation

$$u(\mathbf{x},t) = \int_\Gamma \int_0^t G(||\mathbf{x}-\mathbf{y}||, t-\tau)\varphi(\mathbf{y},\tau)\, d\tau\, d\Gamma_{\mathbf{y}} \quad \mathbf{x} \in \Omega^e, \quad t \in [0,T] \tag{2}$$

holds, where $G(\mathbf{x},t)$ denotes the fundamental solution

$$G(r,t) = \frac{1}{2\pi} \frac{H\left(t - \frac{r}{c}\right)}{\sqrt{t^2 - \frac{r^2}{c^2}}}, \qquad r = ||\mathbf{x}-\mathbf{y}||, \tag{3}$$

$H(\cdot)$ being the Heaviside function. The function φ in (2) is the solution of the following Time Dependent Boundary Integral Equation (TDBIE)

$$\int_\Gamma \int_0^t G(r, t-\tau)\varphi(\mathbf{y},\tau)\, d\tau d\Gamma_{\mathbf{y}} = g(\mathbf{x},t), \quad \mathbf{x} \in \Gamma, \quad t \in [0,T] \tag{4}$$

and represents the jump of the normal derivative of u along Γ.

Several numerical approaches have been proposed for solving wave equation problems by means of BIEs. In particular, we mention the pioneering work by Bamberger and Ha Duong for the scattering problems in the frequency domain [2], the Lubich convolution quadrature method [25] and the energetic approach [1] for solving time dependent boundary integral equations. In [2, 25], theoretical results on stability and convergence are proved when a Galerkin scheme in space is considered.

In our recent research we considered the Lubich convolution method for the discretization of the time integrals appearing in the definition of the single and double-layer operators; it is based on convolution quadrature formulas associated with A-stable methods for ordinary differential equations. Rather than using the explicit expression of the kernel of the integral equation, the Lubich formulas use its Laplace transform, which, in the case of the wave equation, turns out to have better regularity properties. The major advantage of this approach is that it allows to reduce the computational complexity of the time discretization to an order $\mathcal{R}\log\mathcal{R}$ ($\mathcal{R} = 2N$, where N is the total number of time steps performed), thanks to the use of a fast Fourier transform. The time scheme is generally coupled with a collocation or a Galerkin method in space, leading to a quadratic complexity M^2, where M denotes the number of grid points chosen on the domain boundary. Therefore, the overall (space-time) complexity is of order $M^2\mathcal{R}\log\mathcal{R}$. Such approach has been succesfully applied to wave propagation problems in 2D and in 3D, with Dirichlet, Neumann and mixed boundary conditions (see [12], [13], [26]). The Lubich convolution quadrature has also been used for the time approximation of non reflecting boundary conditions of exact type prescribed on artificial boundaries, in a FEM-BEM coupling method for the solution of exterior problems (see

[10] and [11] and the very recent work [9] for problems of waves scattered by moving obstacles).

The main drawback of this approach is the high memory requirement: in fact the method requires to store all the matrices involved in the final linear system, which, when standard Lagrangian basis functions are considered for the space approximation, are generally fully populated, and the required working storage is M^2N.

In three dimensions and for the Lubich-collocation approach, the computational cost and the required memory storage can be significantly reduced when the wave propagation velocity is much higher than one. Indeed, in [14], the authors showed that for high velocities only a very small number of matrices, let us say the first N_0, with $N_0 \ll N$, are significant with respect to a prescribed tolerance, and the remaining ones can be neglected without affecting the solution accuracy. Contrary to the 3D case, in 2D this property does not hold, and this makes the application of the BEMs to 2D large scale problems extremely costly. To overcome this drawback, several effective techniques have been developed over the last decades, aimed at reducing computational cost and memory storage. Among these we mention the fast multiple method ([17]), panel clustering ([18], [15]) and hierarchical matrices ([19]). However, for these methods, the FFT algorithm cannot be applied. A possible alternative is to use a wavelet type approximation in space. It is known that the wavelet approximation has the property of yielding sparse matrices when applied to a wide class of pseudo differential operators ([7]). Wavelet BEMs have already been considered, for example, in [20], [3], [21], [23] for stationary problems. In [22] the authors apply a wavelet BEM for a time dependent wave equation combined with finite differences in time. Up to our knowledge, there are not further papers dealing with wavelet approximations in time dependent wave equation problems.

In this paper we combine the good properties of a wavelet Galerkin approximation in space and those of the Lubich convolution quadrature in time, with the twofold effect of heavily sparsifying the matrices genereated by the space discretization and of considerably reducing the computational cost of the time discretization. This combination turns out to have an effect that goes well beyond the sum of the independent effects of the two techniques. Indeed, the rapid time decreasing behavior of the matrix entries allows to devise a time compression strategy consisting in downsampling the FFT with a resulting computational complexity in time of order $\overline{\mathcal{R}} \log \overline{\mathcal{R}}$, with $\overline{\mathcal{R}} \ll \mathcal{R}$. The benefits of the resulting method are, on the one hand, a consistent reduction of the memory storage and, on the other hand, a significant increase in the efficiency of the quadrature in time.

The paper is organized as follows. In Section 2 we introduce the proposed method, by describing the main steps that lead to the time convolution quadrature formula associated to the single-layer potential BIE (4) and the spatial wavelet discretization, in a rather abstract setting. Convergence estimates are derived for the full discretized scheme. In Section 2.3 we present the fundamental properties of the wavelet basis that will be used to estimate the decaying behavior in time of the matrix entries. In Section 2.3.3 we describe the time downsampling strategy that we apply in the new proposed method to retrieve final sparse matrices by maintaining the optimal time

computational complexity of the FFT algorithm. Finally, in Section 3, we apply the proposed numerical approach to several problems, to show its efficiency in terms of accuracy and computational complexity.

2 Lubich-wavelet Galerkin BEM

In this section we present the Lubich-wavelet Galerkin BEM that we apply to equation (4). It is based on a BDF2 Lubich convolution quadrature formula in time and a Galerkin method in space, which uses biorthogonal wavelet approximating basis functions. In the sequel whatever refers to this numerical approach will be labeled by using the superscript $\mathcal{W}$.

2.1 Time discretization

For the time discretization, we split the interval $[0, T]$ into N steps of equal length $\Delta_t = T/N$ and collocate equation (4) at the time instants $t_n = n\Delta_t$, $n = 0, \ldots, N$:

$$\int_\Gamma \int_0^{t_n} G(r, t_n - \tau)\varphi(\mathbf{y}, \tau)\, d\tau\, d\Gamma_\mathbf{y} = g(\mathbf{x}, t_n),\ \mathbf{x} \in \Gamma,\ n = 0, \ldots, N. \tag{5}$$

The time integrals are then discretized by means of the Lubich convolution quadrature rule associated with the BDF method of order 2 (see [24]):

$$\int_0^{t_n} G(r, t_n - \tau)\varphi(\mathbf{y}, \tau)\, d\tau \approx \sum_{j=0}^{n} \omega_{n-j}(\Delta_t; r)\varphi^j(\mathbf{y}) \quad n = 0, \ldots, N \tag{6}$$

where we have set $\varphi^j(\mathbf{y}) := \varphi(\mathbf{y}, t_j)$. The coefficients ω_n are defined by the contour integrals

$$\omega_n(\Delta_t; r) = \frac{1}{2\pi\imath} \int_{|z|=\rho} \widehat{G}\left(r, \frac{\gamma(z)}{\Delta_t}\right) z^{-(n+1)}\, dz, \tag{7}$$

where

$$\widehat{G}(r, s) = \frac{1}{2\pi} K_0\left(s\frac{r}{c}\right)$$

denotes the Laplace transform of the fundamental solution G, K_0 being the modified Bessel function of second kind and of order 0. In (7) the function γ is the characteristic quotient of the Backward Differentiation Formula of order 2 (BDF2), i.e $\gamma(z) = 3/2 - 2z + 1/2z^2$ and ρ is such that for $|z| \le \rho$ the corresponding $\gamma(z)$ lies in the domain of analyticity of $\widehat{G}$ (for details see [25]).

By introducing the polar coordinate $z = \rho e^{\imath\varphi}$, the integrals in (7) can be efficiently computed by a trapezoidal rule with $\mathcal{R}$ equal steps of length $2\pi/\mathcal{R}$:

$$\omega_n(\Delta_t; r) \approx \frac{\rho^{-n}}{\mathcal{R}} \sum_{m=0}^{\mathcal{R}-1} \widehat{G}\left(r, \frac{\gamma(\rho e^{\imath 2\pi \frac{m}{\mathcal{R}}})}{\Delta_t}\right) e^{-\imath 2\pi n \frac{m}{\mathcal{R}}}. \tag{8}$$

For the computation of the convolution coefficients in (8), a proper choice of the involved parameters, suggested by Lubich in [24], is $\mathcal{R} = 2N$ and ρ such that $\rho^N = 10^{-6}$. Indeed, these choices lead to an approximation of ω_n with a relative error of order 10^{-6}, if $\widehat{G}$ is computed with a relative accuracy bounded by 10^{-12}. We point out that these are the choices we have done in the numerical tests.

2.2 Space discretization

For the space discretization, we consider a standard Galerkin boundary element method. For simplicity, we assume that the curve Γ is given by a global parametric representation. In this case the integration over Γ is reduced to an integration over the parametrization interval. Precisely, let

$$\mathbf{y} = \boldsymbol{\eta}(\theta) = (\eta_1(\theta), \eta_2(\theta)), \quad \theta \in [0, 2\pi]$$

denote the parametrization of the curve Γ. By discretizing the parametrization interval into 2^L subintervals, we introduce a finite dimensional space $V_L \subseteq C^0(0, 2\pi)$ satisfying periodic boundary conditions. In view of the matrix compression technique that we are going to employ further on, it is convenient to denote the basis of V_L by $\{\psi_\lambda,\ \lambda \in \Lambda_L\}$, where Λ_L denotes a suitable finite index set.

By replacing (6) in (5) and approximating the unknown φ^j by

$$\varphi^j(\boldsymbol{\eta}(\theta)) \approx \sum_{\lambda \in \Lambda_L} d_\lambda^j \psi_\lambda(\theta) =: \varphi_L^j(\theta), \tag{9}$$

we obtain the integral equations

$$\sum_{j=0}^{n} \sum_{\lambda \in \Lambda_L} d_\lambda^j \int_0^{2\pi} \omega_{n-j}(\Delta_t; r) \psi_\lambda(\theta) |\boldsymbol{\eta}'(\theta)|\, d\theta = g(\mathbf{x}, t_n), \qquad n = 0, \cdots, N. \tag{10}$$

By using test functions from the same basis, the Galerkin scheme reads: for all $n = 0, \cdots, N$, find $\varphi_L^n \in V_L$ such that

$$\begin{aligned} \sum_{j=0}^{n} \sum_{\lambda \in \Lambda_L} d_\lambda^j \int_0^{2\pi} \int_0^{2\pi} \omega_{n-j}(\Delta_t; r)\, \psi_\lambda(\theta) \psi_{\lambda'}(\sigma)\ \left|\boldsymbol{\eta}'(\theta)\right| \left|\boldsymbol{\eta}'(\sigma)\right|\, d\theta\, d\sigma \\ = \int_0^{2\pi} g(\boldsymbol{\eta}(\sigma), t_n)\, \psi_{\lambda'}(\sigma)\ \left|\boldsymbol{\eta}'(\sigma)\right|\, d\sigma \end{aligned} \tag{11}$$

for all $\lambda' \in \Lambda_L$. Denoting by $\boldsymbol{V}^{\mathcal{W},n}$ and $\boldsymbol{g}^{\mathcal{W},n}$ the matrix and the vector whose generic elements are

$$\boldsymbol{V}_{\lambda,\lambda'}^{\mathcal{W},n} := \int_0^{2\pi} \int_0^{2\pi} \omega_n(\Delta_t; r)\, \psi_\lambda(\theta) \psi_{\lambda'}(\sigma)\ \left|\boldsymbol{\eta}'(\theta)\right| \left|\boldsymbol{\eta}'(\sigma)\right|\, d\theta\, d\sigma, \tag{12}$$

and

$$\boldsymbol{g}_{\lambda'}^{\mathcal{W},n} := \int_0^{2\pi} g(\boldsymbol{\eta}(\sigma), t_n)\, \psi_{\lambda'}(\sigma)\ \left|\boldsymbol{\eta}'(\sigma)\right|\, d\sigma, \tag{13}$$

the Lubich-Galerkin method leads to a block Toeplitz lower triangular linear system of the form

$$\sum_{j=0}^{n} \boldsymbol{V}^{\mathcal{W},n-j} \boldsymbol{d}^j = \boldsymbol{g}^{\mathcal{W},n} \tag{14}$$

in the unknown vectors $\boldsymbol{d}^n = (d_\lambda^n)_{\lambda \in \Lambda_L}$, for $n = 0, \cdots, N$.

By combining (12) with (8), we obtain

$$\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'} \approx \frac{\rho^{-n}}{\mathcal{R}} \sum_{m=0}^{\mathcal{R}-1} \tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(m) e^{-\imath 2\pi n \frac{m}{\mathcal{R}}}, \tag{15}$$

where

$$\tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(m) = \int_0^{2\pi} \int_0^{2\pi} \widehat{G}\left(r, \frac{\gamma(\rho e^{\imath 2\pi \frac{m}{\mathcal{R}}})}{\Delta_t}\right) \psi_\lambda(\theta)\psi_{\lambda'}(\sigma) \left|\boldsymbol{\eta}'(\theta)\right| \left|\boldsymbol{\eta}'(\sigma)\right| \, d\theta \, d\sigma. \tag{16}$$

Denoting by

$$c^{\mathcal{W}}_{\lambda,\lambda'}(n) = \sum_{m=0}^{\mathcal{R}-1} \tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(m) e^{-\imath 2\pi n \frac{m}{\mathcal{R}}}$$

the Discrete Fourier Transform (DFT) of $\tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}$, the entries of index λ, λ' of the matrices $\boldsymbol{V}^{\mathcal{W},n}$ are then approximated by

$$\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'} \approx \frac{\rho^{-n}}{\mathcal{R}} c^{\mathcal{W}}_{\lambda,\lambda'}(n) \tag{17}$$

and can be efficiently computed, simultaneously for all the values $n = 0, \cdots, \mathcal{R}-1$, by the FFT algorithm, with a complexity of order $\mathcal{R} \log \mathcal{R}$.

The stability and the convergence of the proposed method follows from Theorem 5.4 in [25], proved by Lubich for the 3D case but, as asserted in [12], valid in 2D case as well. For the convenience of the reader, we report here the statement of the theorem.

Theorem 2.1 *Let $X_{\Delta_x} \subset L^2(\Gamma)$ be a family of finite dimensional approximation spaces of order m, that is*

$$\inf_{\psi_{\Delta_x} \in X_{\Delta_x}} \|\psi - \psi_{\Delta_x}\|_{H^{-1/2}(\Gamma)} \leq C \Delta_x^{m+1/2} \|\psi\|_{H^m(\Gamma)}, \qquad \textit{for all } \psi \in H^m(\Gamma).$$

Let the time discretization method be A-stable and of order p such that $\gamma(z)$ has no poles on the unit circle. Then, for smooth compatible data g, the fully discrete method (Galerkin in space and convolution quadrature in time) admits a unique solution $\varphi^n_{\Delta_x}$ and is unconditionally stable and convergent with optimal order:

$$\|\varphi(\cdot, t_n) - \varphi^n_{\Delta_x}\|_{H^{-1/2}(\Gamma)} = \mathcal{O}(\Delta_t^p) + \mathcal{O}(\Delta_x^{m+1/2}), \tag{18}$$

uniformly over bounded intervals.

As asserted in [12], this estimate is valid by assuming $g \in H^6((0,T), H^{1/2}(\Gamma))$ having all its derivatives up to order 4 vanishing at $t = 0$.

To apply Theorem (2.1) in our context, in the following section we introduce some fundamental properties of the wavelet basis.

2.3 Wavelets, matrix compression and time downsampling strategy

2.3.1 Wavelets

Let us now put ourselves in the wavelet framework. We assume that we are given a Multiresolution Analysis of $L^2(\mathbb{R})$ (see [8]), that is a sequence of finite dimensional subspaces V_L, satisfying the usual properties:

1. *nestedness*: $V_L \subset V_{L+1}$;
2. *scale invariance*: $f(\theta) \in V_L \Leftrightarrow f(2\theta) \in V_{L+1}$;
3. *translation invariant basis*: there exists a compactly supported function $\phi \in L^2(\mathbb{R})$ such that the set $\{\phi(\theta - n),\ n \in \mathbb{Z}\}$ forms a Riesz basis for V_0;
4. *polynomial reconstruction*: for p polynomial of degree less or equal than $m-1$, there exists a sequence b_n such that

$$p(\theta) = \sum_n b_n \phi(\theta - n).$$

It is well known that there exists a compactly supported function $\psi \in V_1$ such that the set

$$\{\phi(\theta - n),\ n \in \mathbb{Z}\} \cup_{\ell=0}^{L-1} \{2^{\ell/2}\psi(2^\ell\theta - n),\ n \in \mathbb{Z}\}$$

forms a basis for the space V_L. Moreover, given a target polynomial degree $\widetilde{m}$, the function ψ can be chosen in such a way that

$$\int_{\mathbb{R}} \psi(\theta)p(\theta)\,d\theta = 0 \quad \text{for all polynomial } p \text{ of degree lower or equal than } \widetilde{m}-1.$$

The basis functions $2^{\ell/2}\psi(2^\ell x - n)$ will, consequently, have $\widetilde{m}$ vanishing moments. Without loss of generality we can assume that $\operatorname{supp}\phi \subseteq [-R, R]$ and $\operatorname{supp}\psi \subseteq [-\tilde{R}, \tilde{R}]$ for some positive R and $\tilde{R}$.

Basis functions for $L^2(0, 2\pi)$ satifying periodic boundary conditions are easily built by setting

$$\psi_{-1,0}(\theta) = \frac{1}{\sqrt{2\pi}} \sum_{n\in\mathbb{Z}} \phi\left(\frac{\theta}{2\pi} - n\right),$$

$$\psi_{\ell,n}(\theta) = \frac{2^{\ell/2}}{\sqrt{2\pi}} \sum_{m\in\mathbb{Z}} \psi\left(2^\ell \frac{\theta}{2\pi} - 2^\ell m - n\right),\ n = 0, \ldots, 2^\ell - 1.$$

If we now introduce the multi-index $\lambda = (\ell, k) \in \Lambda_L = \{(-1, 0)\} \cup \{(\ell, k), \ell = 0, \ldots, L-1,\ k = 0, \ldots, 2^\ell - 1\}$, we can consider the finite dimensional subspace V_L of $L^2(0, 2\pi)$ generated by the basis $\{\psi_\lambda,\ \lambda \in \Lambda_L\}$.

The polynomial reconstruction property implies that the space V_L satisfies the assumption of Theorem 2.1 for $\Delta_x = 2\pi 2^{-L}$. Moreover, by using the convolution quadrature formula associated to the BDF method of order $p = 2$, for our method we derive the following estimate:

$$\|\varphi(\cdot, t_n) - \varphi_L^n\|_{H^{-1/2}(\Gamma)} = \mathcal{O}(\Delta_t^2) + \mathcal{O}(\Delta_x^{m+1/2}). \tag{19}$$

2.3.2 Matrix compression

It is well known that wavelet bases are an efficient tool for the treatment of integral operators ([6]). In particular, the use of wavelets allows to apply compression strategies which entail a high sparsification of the associated matrices. Indeed these, thanks to the localization and vanishing moments properties of the basis functions, turn out to have a relevant number of negligible entries at each time step. The truncation of the entries that are below a prescribed tolerance allows to replace the dense system of the conventional BEM into a sparse one.

In what follows, the notation $Q1 \lesssim Q2$ (resp. $Q1 \gtrsim Q2$)) means that the quantity $Q1$ is bounded from above (resp. from below) by $C \cdot Q2$, where C is a constant that, unless explicitely stated, does not depend on any relevant parameter involved in the definition of $Q1$ and $Q2$. The notation $\simeq$ means that both $Q1 \lesssim Q2$ and $Q2 \lesssim Q1$ hold.

The key result at the basis of wavelet methods for the solution of boundary integral equations is the following theorem (see [6]).

Theorem 2.2 *Assume that the operator $\mathcal{K}$ defined by*

$$\mathcal{K}u(\sigma) = \int_0^{2\pi} K(r)u(\theta) \left|\boldsymbol{\eta}'(\theta)\right| \, d\theta$$

verifies

1. *$\mathcal{K}$ is of order -1, that is*

$$\|\mathcal{K}u\|_{H^{1/2}(\Gamma)} \simeq \|u\|_{H^{-1/2}(\Gamma)}$$

2. *K is smooth except on the diagonal and we have*

$$|\partial_x^\alpha \partial_y^\beta K(x,y)| \lesssim |x-y|^{|\alpha|+|\beta|}.$$

Then, setting

$$c_{\lambda,\lambda'} = \int_0^{2\pi}\int_0^{2\pi} K(r)\psi_{\lambda'}(\theta)\psi_\lambda(\sigma) \left|\boldsymbol{\eta}'(\sigma)\right| \left|\boldsymbol{\eta}'(\theta')\right| \, d\theta \, d\sigma,$$

and denoting by Ω_λ and Ω_λ^S respectively the support and the singular support of ψ_λ (see [6] for a definition), for $\lambda = (\ell, n)$ and $\lambda' = (\ell', n')$ we have the following two bounds.

1. *Whenever $dist(\Omega_\lambda, \Omega_{\lambda'}) \geq \kappa 2^{-\min\{\ell,\ell'\}}$ it holds*

$$|c_{\lambda,\lambda'}| \lesssim \frac{2^{-(\ell+\ell')(1/2+\widetilde{m})}}{dist(\Omega_\lambda, \Omega_{\lambda'})^{2\widetilde{m}}},$$

   (the constant in the inequality depending on κ).

2. *Otherwise, assuming, to fix the ideas, that $\ell' < \ell$, it holds*

$$|c_{\lambda,\lambda'}| \lesssim \frac{2^{-\ell(1+\widetilde{m})}2^{\ell'}}{dist(\Omega_\lambda, \Omega_{\lambda'}^S)^{\widetilde{m}-1}}.$$

The aim of this paper is to combine these compression techniques with the Lubich convolution quadrature. Theorem 2.2, for the choice of $K = K^n$ with

$$K^n(r) = \sum_{m=0}^{\mathcal{R}-1} \widehat{G}\left(r, \frac{\gamma(\rho e^{\imath 2\pi \frac{m}{\mathcal{R}}})}{\Delta_t}\right) e^{-\imath 2\pi n \frac{m}{\mathcal{R}}},$$

tells us that the matrices $\boldsymbol{V}^{\mathcal{W},n}$ can be approximated by sparse matrices for each fixed n. Actually, it turns out that such matrices become sparser and sparser as n increases. Indeed, as we will see, when using wavelet bases, the sequence $\{\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}\}_{n=0}^{\mathcal{R}-1}$ (with λ, λ' fixed) of the matrix entries displays a rapid decay to zero and, from a certain point on, its elements assume negligible values. By taking advantage of this property, we will apply a numerical strategy that allows to retrieve the significant elements, up to a prescribed tolerance, by using a downsampled FFT. Such a strategy (to which we will refer as *time downsampling strategy* and that we will described in Section 2.3.3) will considerably reduce the numerical effort needed for the computation of the matrix elements (17).

In order to take full advantage of the downsampling strategy, we need to estimate *a priori*, for each couple λ, λ', the value of $\bar{n}$, such that $|\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}| \leq \varepsilon$ for $n \geq \bar{n}$, ε being a prescribed tolerance. To this aim, we need to provide a bound on how $|\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}|$ decreases as n increases. While Theorem 2.2 indicates that we can indeed hope in a compression of the matrix, it does not yield an explicit dependence on n of such a compression, which we need to choose the parameter $\bar{n}$. Due to the quite cumbersome definition of the kernel K^n, which involves a Laplace transform, combined with the BDF2 scheme and a Discrete Fourier transform, making such a dependence explicit is not immediately feasible. In order to propose a criterion for the choice of $\bar{n}$, we instead recall that, for Δ_t sufficiently small, we have (see Theorem 4.1 of [24])

$$\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'} \sim \Delta_t \int_0^{2\pi}\int_0^{2\pi} G(r, n\Delta_t)\psi_\lambda(\theta)\psi_{\lambda'}(\sigma)\ \left|\boldsymbol{\eta}'(\theta)\right|\left|\boldsymbol{\eta}'(\sigma)\right|\ d\theta\, d\sigma.$$

In order to bound the right hand side term, we recall that ψ_λ and $\psi_{\lambda'}$ are orthogonal to all polynomials of degree less or equal than $\tilde{m}-1$. For $p(x,y)$ arbitrary polynomial of degree less or equal than $\tilde{m}-1$, we then have

$$\begin{aligned}
&\left|\int_0^{2\pi}\int_0^{2\pi} G(r, n\Delta_t)\psi_\lambda(\theta)\psi_{\lambda'}(\sigma)|\ \left|\boldsymbol{\eta}'(\theta)\right|\left|\boldsymbol{\eta}'(\sigma)\right|\ d\theta\, d\sigma\right| \\
&\qquad = \left|\int_0^{2\pi}\int_0^{2\pi} \left(G(r, n\Delta_t)\left|\boldsymbol{\eta}'(\theta)\right|\left|\boldsymbol{\eta}'(\sigma)\right| - p(\theta,\sigma)\right)\psi_\lambda(\theta)\psi_{\lambda'}(\sigma)\ d\theta\, d\sigma\right| \\
&\qquad \lesssim \|G_n - p\|_{L^2(\Omega_\lambda\times\Omega_{\lambda'})},
\end{aligned}$$

where Ω_λ and $\Omega_{\lambda'}$ denote the support of, respectively, ψ_λ and $\psi_{\lambda'}$,

$$G_n(\theta,\sigma) := G(r(\theta,\sigma), n\Delta_t)\left|\boldsymbol{\eta}'(\theta)\right|\left|\boldsymbol{\eta}'(\sigma)\right|,$$

and where we used that the L^2 norm of $\psi_\lambda(\theta)\psi_{\lambda'}(\sigma)$ is 1. By the arbitrariness of p, provided $\widetilde{m} \geq 1$, we then have

$$\left|\int_0^{2\pi}\int_0^{2\pi} G(r, n\Delta_t)\psi_\lambda(\theta)\psi_{\lambda'}(\sigma)\ \left|\boldsymbol{\eta}'(\theta)\right|\left|\boldsymbol{\eta}'(\sigma)\right|\ d\theta\, d\sigma\right| \lesssim \inf_{p\in\mathbb{P}_{\tilde{m}-1}(\Omega_\lambda\times\Omega_{\lambda'})} \|G_n - p\|_{L^2(\Omega_\lambda\times\Omega_{\lambda'})}.$$

We bound the last term by a standard argument (see [4]) as

$$\begin{aligned}\|G_n - p\|_{L^2(\Omega_\lambda \times \Omega_{\lambda'})} &\lesssim 2^{-\widetilde{m}\min\{\ell,\ell'\}} |G_n|_{\widetilde{m},\Omega_\lambda\times\Omega_{\lambda'}} \\ &\lesssim 2^{-\widetilde{m}\min\{\ell,\ell'\}} 2^{-(\ell+\ell')/2} \max_{(\theta,\sigma)\in\Omega_\lambda\times\Omega_{\lambda'}} \max_{|\alpha|=\widetilde{m}} |D^\alpha G_n(\theta,\sigma)|, \quad (20)\end{aligned}$$

finally yielding

$$|\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}| \lesssim \Delta_t 2^{-\widetilde{m}\min\{\ell,\ell'\}} 2^{-(\ell+\ell')/2} \max_{(\theta,\sigma)\in\Omega_\lambda\times\Omega_{\lambda'}} \max_{|\alpha|=\widetilde{m}} |D^\alpha G_n(\theta,\sigma)|. \quad (21)$$

Given λ, λ' and a threshold ε, the idea is now to look for $\bar{n}$ such that for $n > \bar{n}$ the value of the right hand side of (21) is less or equal than ε. The solution will of course depend on the shape of the curve Γ via the derivatives of the distance d with respect to θ and σ.

In practice, assuming that the parametrization of the curve Γ is chosen in such a way that $|\eta'(\theta)| = 1$ for all $\theta \in [0, 2\pi]$, the computation of $\bar{n}$ can be performed by observing that, for $r = r(\theta,\sigma) < c\,n\Delta_t$, it holds:

$$\frac{\partial^2 G_n}{\partial\theta^2}(\theta,\sigma) = \frac{1}{2\pi}\left[\left(\frac{c^{-2}}{((n\Delta_t)^2 - c^{-2}r^2)^{3/2}} + \frac{3c^{-4}r^2}{((n\Delta_t)^2 - c^{-2}r^2)^{5/2}}\right) r_\theta^2 + \frac{c^{-2}r}{((n\Delta_t)^2 - c^{-2}r^2)^{3/2}} r_{\theta\theta}\right].$$

The computation of the derivatives $\dfrac{\partial^2 G_n}{\partial\sigma^2}$ and $\dfrac{\partial^2 G_n}{\partial\theta\partial\sigma}$ yields to an analogous expression. Therefore, setting

$$M_{\lambda,\lambda'} := \max_{(\theta,\sigma)\in\Omega_\lambda\times\Omega_{\lambda'}} r(\theta,\sigma),$$

and

$$D_{1,\Gamma} := \max_{|\alpha|=1} \max_{(\theta,\sigma)\in\Gamma\times\Gamma} |D^\alpha r(\theta,\sigma)|, \qquad D_{2,\Gamma} := \max_{|\alpha|=2} \max_{(\theta,\sigma)\in\Gamma\times\Gamma} |D^\alpha r(\theta,\sigma)|,$$

we get that the final bound for the behaviour of matrix entries reads

$$\begin{aligned}|\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}| \;\lesssim\; & \frac{1}{2\pi}\Delta_t 2^{-\widetilde{m}\min\{\ell,\ell'\}} 2^{-(\ell+\ell')/2} \left[\left(\frac{c^{-2}}{((n\Delta_t)^2 - c^{-2}M^2_{\lambda,\lambda'})^{3/2}}\right.\right. \\ & \left.\left. + \frac{3c^{-4}M^2_{\lambda,\lambda'}}{((n\Delta_t)^2 - c^{-2}M^2_{\lambda,\lambda'})^{5/2}}\right) D^2_{1,\Gamma} + \frac{c^{-2}M_{\lambda,\lambda'}}{((n\Delta_t)^2 - c^{-2}M^2_{\lambda,\lambda'})^{3/2}} D_{2,\Gamma}\right] \quad (22)\end{aligned}$$

The bound in (22) can not be analytically solved with respect to n. However, for a fixed time discretization, once the quantities involved therein are evaluated, an upper bound for $\bar{n}$ can be numerically computed by seeking the biggest time instant $t_{\bar{n}}$ beyond which the quantity (22) is smaller than ε. Just to make an example, let us consider the case of Γ being a circle of radius R. In this case, it is not difficult to obtain that the terms $D_{1,\Gamma}$ and $D_{2,\Gamma}$ can be bounded by R. Therefore, an estimate for $\bar{n}$ follows by computing the quantity to the right hand side of the following estimate

$$\begin{aligned}|\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}| \;\lesssim\; & \frac{1}{2\pi}\Delta_t 2^{-\widetilde{m}\min\{\ell,\ell'\}} 2^{-(\ell+\ell')/2} \left[\left(\frac{c^{-2}(R^2 + M_{\lambda,\lambda'}R)}{((n\Delta_t)^2 - c^{-2}M^2_{\lambda,\lambda'})^{3/2}}\right.\right. \quad (23) \\ & \left.\left. + \frac{3c^{-4}M^2_{\lambda,\lambda'}R^2}{((n\Delta_t)^2 - c^{-2}M^2_{\lambda,\lambda'})^{5/2}}\right)\right].\end{aligned}$$

We finally remark that estimate (23) will be used in the numerical tests of Section 3.

2.3.3 Time downsampling strategy

From estimate (22) we deduce that, for fixed λ, λ', the matrix elements have a rapid decay to zero for increasing values of the time instants $t_n = n\Delta_t$. Moreover, this behavior is influenced both by ℓ and ℓ' and by the speed c of the wave propagation. Indeed, as we will notice in the forthcoming numerical tests, for a fixed tolerance ε, the higher the values of the wavelet basis levels ℓ and ℓ' and of the velocity are, the smaller the value of $\bar{n}$ is. The decaying behavior to zero of $\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}$ allows us to use the time downsampling procedure to avoid the useless computation of such negligible terms, leading both to computational and memory saving advantages.

In order to describe the time downsampling procedure, we start by recalling the relation between the FFT and its Inverse Discrete Fourier Transform (IDFT). In particular, the coefficients in (16) can be retrieved by:

$$\tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(n) = \frac{1}{\mathcal{R}} \sum_{m=0}^{\mathcal{R}-1} c^{\mathcal{W}}_{\lambda,\lambda'}(m) e^{\imath 2\pi m \frac{n}{\mathcal{R}}}, \qquad n = 0, \cdots, \mathcal{R}-1. \tag{24}$$

Let us denote by $\overline{\mathcal{R}}$, with $\mathcal{R} = Q\overline{\mathcal{R}}$ the smallest integer, submultiple of $\mathcal{R}$, such that $|\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}| \leq \varepsilon$ for $n \geq \overline{\mathcal{R}}$. We remark that $\overline{\mathcal{R}} = \overline{\mathcal{R}}(\lambda, \lambda')$ depends on the pair of wavelet multi-indices λ, λ' but, for simplicity of notation, such dependency will be omitted in the forthcoming notation. According to (17), we have $c^{\mathcal{W}}_{\lambda,\lambda'}(n) \approx \rho^n \mathcal{R} \boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}$, and hence (since $\rho < 1$) the coefficients $|c^{\mathcal{W}}_{\lambda,\lambda'}(n)| \leq \mathcal{R}\varepsilon =: \tilde{\varepsilon}$, for $n \geq \overline{\mathcal{R}}$.

We rewrite (24) as

$$\begin{aligned} \tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(n) &= \frac{1}{\mathcal{R}} \sum_{m=0}^{\overline{\mathcal{R}}-1} c^{\mathcal{W}}_{\lambda,\lambda'}(m) e^{\imath 2\pi m \frac{n}{\mathcal{R}}} + \frac{1}{\mathcal{R}} \sum_{m=\overline{\mathcal{R}}}^{\mathcal{R}-1} c^{\mathcal{W}}_{\lambda,\lambda'}(m) e^{\imath 2\pi m \frac{n}{\mathcal{R}}} \\ &=: \tilde{c}^{\mathcal{W},1}_{\lambda,\lambda'}(n) + \tilde{c}^{\mathcal{W},2}_{\lambda,\lambda'}(n), \end{aligned}$$

with $|\tilde{c}^{\mathcal{W},2}_{\lambda,\lambda'}(n)| \leq (\mathcal{R} - \overline{\mathcal{R}})/\mathcal{R}\, \tilde{\varepsilon} \leq \tilde{\varepsilon}$, for $n = 0, \cdots, \mathcal{R}-1$.

By neglecting this second small term and evaluating $\tilde{c}^{\mathcal{W},1}_{\lambda,\lambda'}(n)$ at $n = pQ$, with $p = 0, \cdots, \overline{\mathcal{R}} - 1$, we get

$$\tilde{c}^{\mathcal{W},1}_{\lambda,\lambda'}(pQ) = \frac{1}{Q\overline{\mathcal{R}}} \sum_{m=0}^{\overline{\mathcal{R}}-1} c^{\mathcal{W}}_{\lambda,\lambda'}(m) e^{\imath 2\pi m \frac{p}{\overline{\mathcal{R}}}}$$

from which, applying the DFT, we deduce

$$c^{\mathcal{W}}_{\lambda,\lambda'}(n) = Q \sum_{p=0}^{\overline{\mathcal{R}}-1} \tilde{c}^{\mathcal{W},1}_{\lambda,\lambda'}(pQ) e^{-\imath 2\pi n \frac{p}{\overline{\mathcal{R}}}}, \qquad n = 0, \cdots, \overline{\mathcal{R}}-1. \tag{25}$$

Therefore, recalling that $\tilde{c}^{\mathcal{W},1}_{\lambda,\lambda'} \approx \tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}$ (up to a term smaller than $\tilde{\varepsilon}$), according to equation (25), to get the significant coefficients $c^{\mathcal{W}}_{\lambda,\lambda'}(n)$ for $n = 0, \cdots, \overline{\mathcal{R}}$, it is sufficient to compute the coefficients $\tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(pQ)$ only at the $\overline{\mathcal{R}}$ downsampled values pQ, $0 \leq p \leq \overline{\mathcal{R}} - 1$.

Using (25) and (15), the computation of the significant entries of the matrices is therefore given by the following formula:

$$\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'} \approx \frac{\rho^{-n}}{\mathcal{R}} Q \sum_{p=0}^{\overline{\mathcal{R}}-1} \tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(pQ) e^{-\imath 2\pi n \frac{p}{\overline{\mathcal{R}}}}, \qquad n = 0, \cdots, \overline{\mathcal{R}} - 1. \tag{26}$$

It is worthwhile noting that the complexity of the downsampled FFT algorithm turns out to be of order $\overline{\mathcal{R}} \log \overline{\mathcal{R}}$. As we will see in the numerical tests, for many couples λ, λ' it results $\overline{\mathcal{R}} \ll \mathcal{R}$, and this allows a consistent computational cost reduction and, consequently, a high memory saving.

We remark that, when in (16) piece-wise linear nodal approximating basis functions are considered instead of the wavelet basis functions ψ_λ and $\psi_{\lambda'}$, a downsampling strategy could in principle be applied as well. However, in this case, it turns out that the associated coefficients obtained by the FFT are not negligible, and the cut off procedure is not effective (see Section 3.1).

In the next section we present some numerical examples to test the Lubich-wavelet Galerkin approach presented in Sections 2.1 and 2.2, combined with the time downsampling strategy described in Section 2.3. The goal of the numerical experiments is twofold: a) to investigate the sharpness of the estimate (22); b) to apply the new proposed method to problems of type (1) and to test the efficiency of the downsampling procedure in this context.

3 Numerical results

We start by detailing the choice of the wavelet basis that we use in the implementation for the spatial approximation. We consider the biorthogonal *bior2.2* compactly supported wavelet functions (see [5]), for which the analytic expression of the functions ϕ and ψ is given by

$$\phi(t) = \begin{cases} 1 - |t|, & |t| \leq 1 \\ 0, & \text{else} \end{cases}$$

and

$$\psi(t) = \frac{\sqrt{2}}{4}\phi(2t+1) + \frac{\sqrt{2}}{2}\phi(2t) - \frac{3\sqrt{2}}{2}\phi(2t-1) + \frac{\sqrt{2}}{2}\phi(2t-2) + \frac{\sqrt{2}}{4}\phi(2t-3).$$

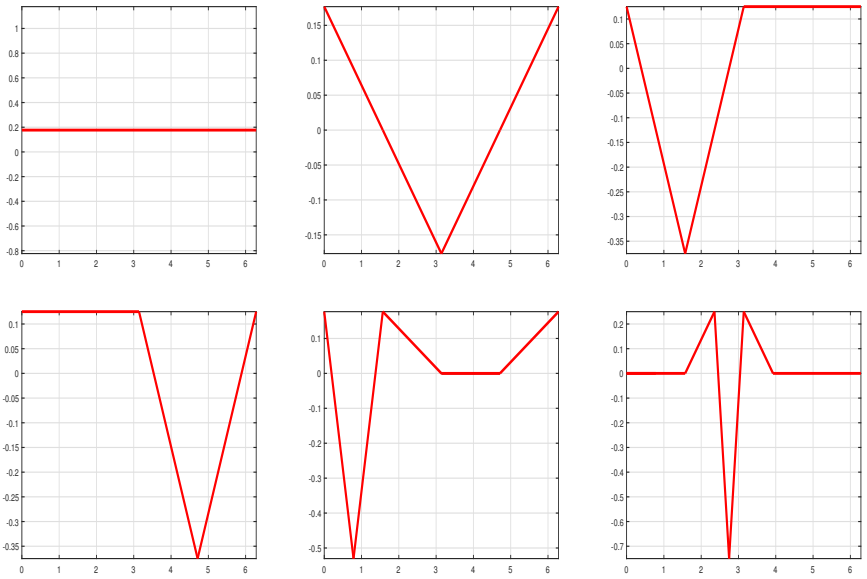
In Figure 1 we show the some basis functions for some levels ℓ. We point out that their piece-wise linear behavior will be taken into account in the numerical computation of the matrix entries.

In what follows, although not explicitly stated, the unit measures for the physical quantities are: meter (m), second (s) and m/s.

3.1 Efficiency of the downsampling strategy

To show the efficiency of the downsampling strategy we consider here, as a benchmark example, Problem (1) where Γ is the circumference of radius $R = 1$ and the final time

Figure 1: scaling function $\psi_{-1,0}$ and wavelet basis functions $\psi_{0,0}$, $\psi_{1,0}$, $\psi_{1,1}$, $\psi_{2,0}$ and $\psi_{3,3}$ for the choice $L = 5$.



$T = 10$. The discretization parameters are $L = 8$ and $N = 256$, and the threshold $\varepsilon = 1.0e - 10$.

In what follows, we refer to the approach where piece-wise linear nodal basis functions are used as to the standard approach, and we will label it by the superscript $\mathcal{S}$. The associated coefficients, corresponding to formula (16), will be denoted by $\tilde{c}^{\mathcal{S}}_{i,j}$ and the matrix entries obtained by applying the FFT will be denoted by $\boldsymbol{V}^{\mathcal{S},n}_{i,j}$.

In Figures 2 and 3 we compare the cut off effects for two choices of the velocity: $c = 1$ and $c = 343$ (speed of propagation of acoustic waves in air). In each figure, the left plots correspond to the behaviour in time of the coefficients $\tilde{c}^{\mathcal{S}}_{i,j}$ (standard approach, top row) and $\tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}$ (wavelet approach, middle and bottom rows). The right plots represent the behavior in time of the matrix entries $\boldsymbol{V}^{\mathcal{S},n}_{i,j}$ (standard approach, top row) and $\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}$ (wavelet approach, middle and bottom rows) computed by the FFT algorithm. The graphs in solid lines refer to the application of the FFT algorithm to the whole set of time instants, while the bullets refer to the FFT algorithm applied to the downsampled time instants.

As we can see, in the wavelet approach, the downsampling strategy reveals to be crucial since it significantly reduces the computation of the coefficients $\tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}$ at only $\overline{\mathcal{R}} \ll \mathcal{R} = 512$ values.

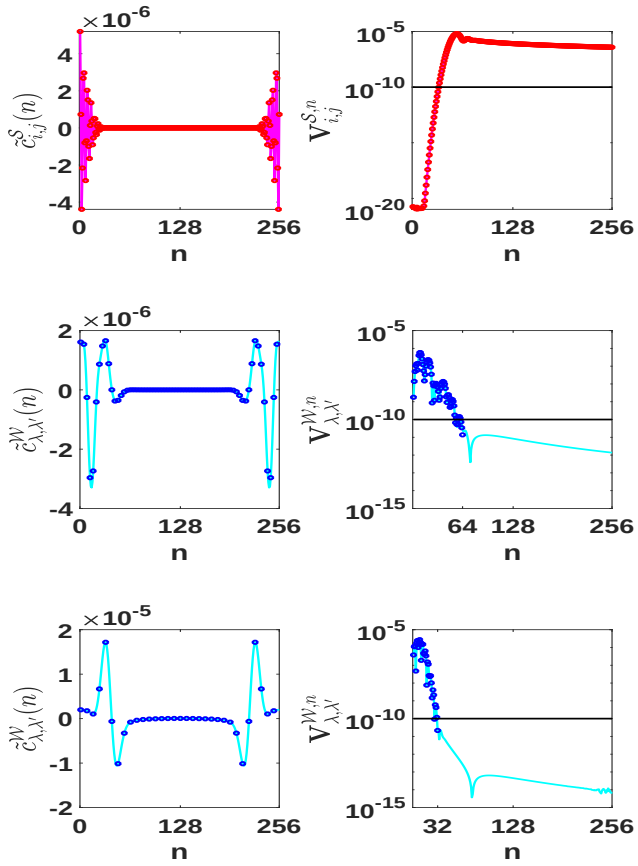
For example, for the fixed values $\lambda = (3, 2)$ and $\lambda' = (7, 14)$, it results $\overline{\mathcal{R}} = 64$ for $c = 1$ and $\overline{\mathcal{R}} = 4$ for the higher velocity $c = 343$.

For the wavelets associated to the higher levels, $\overline{\mathcal{R}}$ is even smaller, as expected. For example, for $\lambda = (7, 1)$ and $\lambda' = (7, 124)$ it results $\overline{\mathcal{R}} = 4$ for $c = 1$, and $\overline{\mathcal{R}} = 1$ for $c = 343$.

On the contrary, in the standard approach, the downsampling strategy is not effec-

tive because the matrix entries $\boldsymbol{V}_{i,j}^{\mathcal{S},n}$ are all greater than the chosen threshold ε when n increases.

Figure 2: coefficients $\tilde{c}_{i,j}^{\mathcal{S}}(n)$ and corresponding matrix entries $\boldsymbol{V}_{i,j}^{\mathcal{S},n}$ with $(i,j) = (9, 141)$ (top). Coefficients $\tilde{c}_{\lambda,\lambda'}^{\mathcal{W}}(n)$ and corresponding matrix entries $\boldsymbol{V}_{\lambda,\lambda'}^{\mathcal{W},n}$ with $\lambda = (3,2), \lambda' = (7,14)$ (middle) and with $\lambda = (7,1), \lambda' = (7,124)$ (bottom). $c = 1\, m/s$.

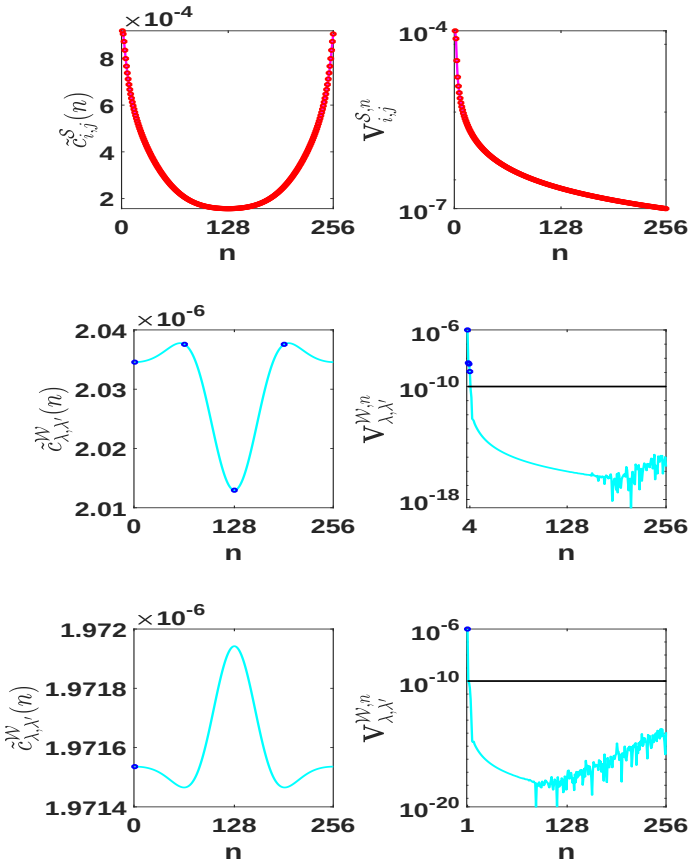


Remark 3.1 *It is worth noting that, in order to apply the downsampled FFT algorithm, $\overline{\mathcal{R}}$ must be a submultiple of $\mathcal{R}$ and, for this reason, some values $\boldsymbol{V}_{\lambda,\lambda'}^{\mathcal{W},n}$ are computed even if they are below the threshold ε (see for example Figure 2, middle and bottom plots on the right). To reduce further the storage, these coefficients will be dropped successively by applying an a posteriori cut off strategy, based on the same ε.*

The above presented benchmark example gives an idea of the computational cost and memory saving for the computation of the quantities $\tilde{c}_{\lambda,\lambda'}^{\mathcal{W}}$ for some fixed values of λ, λ'. In the forthcoming numerical results, we will show the global memory saving by computing the quantity mem(%) defined in (30) that takes into account the saving for all the indices λ, λ'.

We also highlight that the time downsampling strategy produces matrices $\boldsymbol{V}^{\mathcal{W},n}$ of the final linear system (14) that, for each fixed value of n, turn out to be sparse.

Figure 3: coefficients $\tilde{c}^{\mathcal{S}}_{i,j}(n)$ and corresponding matrix entries $\boldsymbol{V}^{\mathcal{S},n}_{i,j}$ with $(i,j) = (9, 141)$ (top). Coefficients $\tilde{c}^{\mathcal{W}}_{\lambda,\lambda'}(n)$ and corresponding matrix entries $\boldsymbol{V}^{\mathcal{W},n}_{\lambda,\lambda'}$ with $\lambda = (3,2), \lambda' = (7,14)$ (middle) and with $\lambda = (7,1), \lambda' = (7,124)$ (bottom). $c = 343\, m/s$.



Therefore, the resolution of (14),

$$\boldsymbol{V}^{\mathcal{W},0}\boldsymbol{d}^n = \boldsymbol{g}^{\mathcal{W},n} - \sum_{j=0}^{n-1} \boldsymbol{V}^{\mathcal{W},n-j}\boldsymbol{d}^j, \quad n = 0, \cdots, N,$$

takes advantage of the fast sparse matrix-vector product, in particular for what concerns the computation of the right hand side term $\sum_{j=0}^{n-1} \boldsymbol{V}^{n-j,\mathcal{W}}\boldsymbol{d}^j$ for each $n = 0, \cdots, N$.

3.2 The global algorithm

In the forthcoming numerical tests, we will compare the approach $\mathcal{W}$ with the standard Lubich-Galerkin BEM based on piecewise linear approximating functions in space (approach $\mathcal{S}$). By virtue of the considerations made in Section 2.3, in this case we will not apply the downsampling strategy, but only the a posteriori cut off of the matrix entries below the chosen threshold ε.

Finally, we will also consider the Lubich-wavelet Galerkin approach where the matrices of the Toeplitz block linear system are obtained by using the fast wavelet transform (FWT); this latter approach will be detailed later and will be labeled by the capital letter $\mathcal{T}$. As we will explain, in this case too, we will apply only the a posteriori cut off procedure.

We remark that the implementation at the current stage of development does not employ optimized libraries or parallelization, and is based on a standard (i.e., sequential) Matlab code.

For each approach we will compute the density function $\boldsymbol{\varphi}$, solution of the BIE (4), and the associated single layer potential $\mathbf{u}(\mathbf{x})$, solution of Problem (1) and defined by (2), in a chosen point $\mathbf{x} \in \Omega^e$.

We will compare the following algorithms:

Alg. $\mathcal{S}$) It consists of the following two main steps:

- $\mathcal{S}1$: the matrix elements $\boldsymbol{V}_{i,j}^{\mathcal{S},n}$ are computed by using a $\nu \times \nu$-point Gauss-Legendre quadrature rule in space applied to each interval of regularity of the piece-wise linear basis functions, and by performing the complete FFT algorithm in time.
- $\mathcal{S}2$: subsequently, the a posteriori cutting is applied to set equal to zero the matrix entries such that $|\boldsymbol{V}_{i,j}^{\mathcal{S},n}| \leq \varepsilon$ for $n = 0, \cdots, N$.

We denote by $\boldsymbol{\varphi}_{M,N}^{\mathcal{S}}$ and $\boldsymbol{u}_{M,N}^{\mathcal{S}}$ the corresponding approximate solutions, obtained by using $M = 2^L$ subintervals for the discretization of the parametrization interval $[0, 2\pi]$ and N subintervals for the time interval $[0, T]$.

Alg. $\mathcal{T}$) It consists of the following two main steps:

- $\mathcal{T}1$: the matrices $\boldsymbol{V}^{\mathcal{T},n}$ are retrieved by applying the fast wavelet transform to the matrices $\boldsymbol{V}^{\mathcal{S},n}$. To this aim we use the Matlab function `wavedec`, which returns the wavelet decompositions of the piecewise linear basis functions at the level L associated to the *bior2.2* wavelets. Storing these latter as columns of a square matrix $\boldsymbol{W}$, we compute $\boldsymbol{V}^{\mathcal{T},n}$ as

$$\boldsymbol{V}^{\mathcal{T},n} = \boldsymbol{W}^{-T} \boldsymbol{V}^{\mathcal{S},n} \boldsymbol{W}^{-1}, \quad n = 0, \cdots, N. \tag{27}$$

- $\mathcal{T}2$: then, we apply the a posteriori cutting of the matrix entries such that $|\boldsymbol{V}_{\lambda,\lambda'}^{\mathcal{T},n}| \leq \varepsilon$, for $n = 0, \cdots, N$.

The corresponding approximate solutions are denoted by $\boldsymbol{\varphi}_{M,N}^{\mathcal{T}}$ and $\boldsymbol{u}_{M,N}^{\mathcal{T}}$.

Alg. $\mathcal{W}$) It consists of the following two main steps:

- $\mathcal{W}1$: for a fixed threshold ε and for $\overline{\mathcal{R}}$ obtained according to the estimate (22), such that $\mathcal{R} = Q\overline{\mathcal{R}}$ and $|\boldsymbol{V}_{\lambda,\lambda'}^{\mathcal{W},n}| \leq \varepsilon$ for all $n \geq \overline{\mathcal{R}}$, the matrix elements $\boldsymbol{V}_{\lambda,\lambda'}^{\mathcal{W},n}$ are computed by using a $\nu \times \nu$-point Gauss-Legendre quadrature rule in space applied to each interval of regularity of the wavelet basis functions, with ν properly chosen. In time, we apply the downsampled FFT to the set of coefficients $\tilde{c}_{\lambda,\lambda'}^{\mathcal{W}}(pQ)$, with $p = 0, \cdots, \overline{\mathcal{R}}$.

$\mathcal{W}2$: subsequently, the elements computed by the downsampled FFT that are below the threshold ε, are further cut by an a posteriori cut off strategy based on the same ε (see Remark 3.1). The remaining entries are those actually stored.

We will denote by $\boldsymbol{\varphi}^{\mathcal{W}}_{M,N}$ and $\boldsymbol{u}^{\mathcal{W}}_{M,N}$ the corresponding approximate solutions.

We point out that, the a posteriori cutting $\mathcal{S}2$ applied to the approach $\mathcal{S}$ allows a low memory saving, in agreement with the remarks done in Section 2.2 about the time behavior of the matrix entries $\boldsymbol{V}^{\mathcal{S},n}_{i,j}$. On the contrary, as expected, the a posteriori cutting $\mathcal{T}2$ applied to the approach $\mathcal{T}$ allows to retrieve sparse matrices as well, as those obtained by the combination of $\mathcal{W}1$ and $\mathcal{W}2$ associated to the approach $\mathcal{W}$. However, the cutting procedure in $\mathcal{T}$ can be performed only after that the matrices $\boldsymbol{V}^{\mathcal{S},n}$ have been computed and stored for all the time steps $n = 0, \cdots, N$. Therefore the approach $\mathcal{T}$ can not be taken into account for the memory saving, and is basically considered for comparison. This drawback will be highlighted in Example 6, in which the approaches $\mathcal{S}$ and $\mathcal{T}$ can not be applied because of an out of memory, while the approach $\mathcal{W}$ can be performed.

In the following examples, we take as "exact" reference solutions $\boldsymbol{\varphi}^{\rm ex}_{M_{\rm e},N_{\rm e}}$ and $\boldsymbol{u}^{\rm ex}_{M_{\rm e},N_{\rm e}}$ the ones obtained by the standard approach $\mathcal{S}$ that uses piece-wise basis functions for the spatial approximation, and without applying any cutting. The discrete parameters $M_{\rm e}$ and $N_{\rm e}$ are reference discretization values properly chosen according to the numerical example to which they refer.

To test the accuracy and the efficiency of the above described approaches $* = \{\mathcal{S}, \mathcal{W}, \mathcal{T}\}$, we will consider:

1) in the plots, for a graphic comparison, the absolute errors

$$\mathrm{E}^*_{\varphi}(\mathbf{x},t) = \left|\boldsymbol{\varphi}^{\rm ex}_{M_{\rm e},N_{\rm e}}(\mathbf{x},t) - \boldsymbol{\varphi}^*_{M,N}(\mathbf{x},t)\right|$$

and

$$\mathrm{E}^*_{u}(\mathbf{x},t) = \left|\boldsymbol{u}^{\rm ex}_{M_{\rm e},N_{\rm e}}(\mathbf{x},t) - \boldsymbol{u}^*_{M,N}(\mathbf{x},t)\right|;$$

associated to $\boldsymbol{\varphi}^*_{M,N}$ and $\boldsymbol{u}^*_{M,N}$, respectively;

2) in the tables, the maximum in time of the absolute errors

$$\mathrm{E}^*_{L^2,\varphi} = \max_{t\in[0,T]} \|\boldsymbol{\varphi}^{\rm ex}_{M_{\rm e},N_{\rm e}}(\cdot,t) - \boldsymbol{\varphi}^*_{M,N}(\cdot,t)\|_{L^2(\Gamma)} \tag{28}$$

and

$$\mathrm{E}^*_{\max,u}(\mathbf{x}) = \max_{t\in[0,T]} \left|\boldsymbol{u}^{\rm ex}_{M_{\rm e},N_{\rm e}}(\mathbf{x},t) - \boldsymbol{u}^*_{M,N}(\mathbf{x},t)\right|; \tag{29}$$

3) the percentage of the memory saving

$$\mathrm{mem}^*(\%) = \left(1 - \frac{\mathrm{nz}^*}{M^2N}\right)\cdot 100, \tag{30}$$

M^2N being the total number of matrix elements of the standard approach, and nz^* the number of the elements that are stored after the cutting.

For simplicity, the domain Ω^i that we consider in all the forthcoming examples is a disc of radius $R = 1$. By varying the velocity of the wave propagation, we can observe the dependency on c of the sparsity of the discrete integral operators.

Test 1: sparsity pattern of the matrices $\boldsymbol{V}^{*,n}$

As remarked at the end of Section 2.3, the downsampling strategy applied in the wavelet approach produces matrices $\boldsymbol{V}^{\mathcal{W},n}$ that, for fixed values of n, turn out to be sparse. In this test, in order to show the sparsity pattern of such matrices, we consider Problem (1) with $T = 10$, and for two choices of the velocity: $c = 1$ and $c = 343$. We split the parametrization interval of the unit circumference by using $M = 2^L$ subintervals, with $L = 8$, and the time interval into $N = 256$ subintervals. We fix the threshold parameter $\varepsilon = 1.0e-08$. In Figures 4 and 5 we present the structure of the compressed matrices $\boldsymbol{V}^{\mathcal{S},n}$ (first column), $\boldsymbol{V}^{\mathcal{T},n}$ (second column) and $\boldsymbol{V}^{\mathcal{W},n}$ (third, fourth and fifth columns), at the time instants t_0 (top row), $t_{N/2}$ (middle row) and t_N (bottom row), for $c = 1$ and $c = 343$, respectively. The sparsity pattern of the matrices $\boldsymbol{V}^{\mathcal{S},n}$ and $\boldsymbol{V}^{\mathcal{T},n}$ results from the application of the a posteriori cutting $\mathcal{S}2$ and $\mathcal{T}2$, respectively. For what concerns the matrices $\boldsymbol{V}^{\mathcal{W},n}$, to validate the effectiveness of the a priori estimate (22), we show the sparsity patterns at two different stages: the number of the effectively computed entries of the matrices obtained by applying only the a priori compression strategy $\mathcal{W}1$ (fourth column); the number of the effectively stored entries of the matrices obtained by applying further the compression strategy $\mathcal{W}2$ (fifth column). For the sake of comparison, we also show the sparsity pattern of the matrices $\boldsymbol{V}^{\mathcal{W},n}$ obtained by using a compression strategy based on an optimal truncation value $\overline{\mathcal{R}}$ "a posteriori", after computing all the entries (third column).

As we can see in Figure 4 (velocity $c = 1$), at the first time instant the compressed matrix generated by the approach $\mathcal{S}$ is sparse, while for the other instants it is fully populated; indeed, for $t_{N/2}$ and t_N the number nz of non-zero entries coincides with $M^2 = (2^L)^2 = 65536$. Consequently, the total memory saving in this case is low and turns out to be only $\text{mem}^{\mathcal{S}} \approx 9\%$. On the contrary, the compressed matrices in the wavelet case are all highly sparse; they have the typical *finger structure* at the first time instant and are even sparser at the subsequent instants. In particular, at $t_N = 10$, only 62 entries out of 65536 are stored. In this case, it turns out that $\text{mem}^{\mathcal{W}} = \text{mem}^{\mathcal{T}} \approx 91\%$.

In Figure 5 (velocity $c = 343$), the effectiveness of the matrix compression for the approaches $\mathcal{W}$ and $\mathcal{T}$ with respect to the approach $\mathcal{S}$ is even more evident, as expected, because of the higher value of the velocity. In fact, in the latter case, all the entries of the matrices $\boldsymbol{V}^{\mathcal{S},n}$, for all the time instants, are computed and stored, with a consequent null memory saving ($\text{mem}^{\mathcal{S}} = 0\%$). For higher velocity the sparsification in the wavelet approach is on the contrary stronger, especially when t increases. Incidentally, we point out that for $t_{N/2} = 5$ and $t_N = 10$, only a single entry of the matrices $\boldsymbol{V}^{\mathcal{W},n}$ and $\boldsymbol{V}^{\mathcal{T},n}$, out of 65536, is computed and stored. In this case, it turns out that $\text{mem}^{\mathcal{W}} = \text{mem}^{\mathcal{T}} \approx 99.8\%$.

In both cases, the comparison between the 3rd and 4th columns shows the good performance of the a priori cutting strategy.

Figure 4: Test 1. Sparsity pattern of $\boldsymbol{V}^{\mathcal{S},n}$ (left column), $\boldsymbol{V}^{\mathcal{T},n}$ (middle column) and $\boldsymbol{V}^{\mathcal{W},n}$ (right column) at t_n, for $n = 0, 128, 256$. $c = 1$.

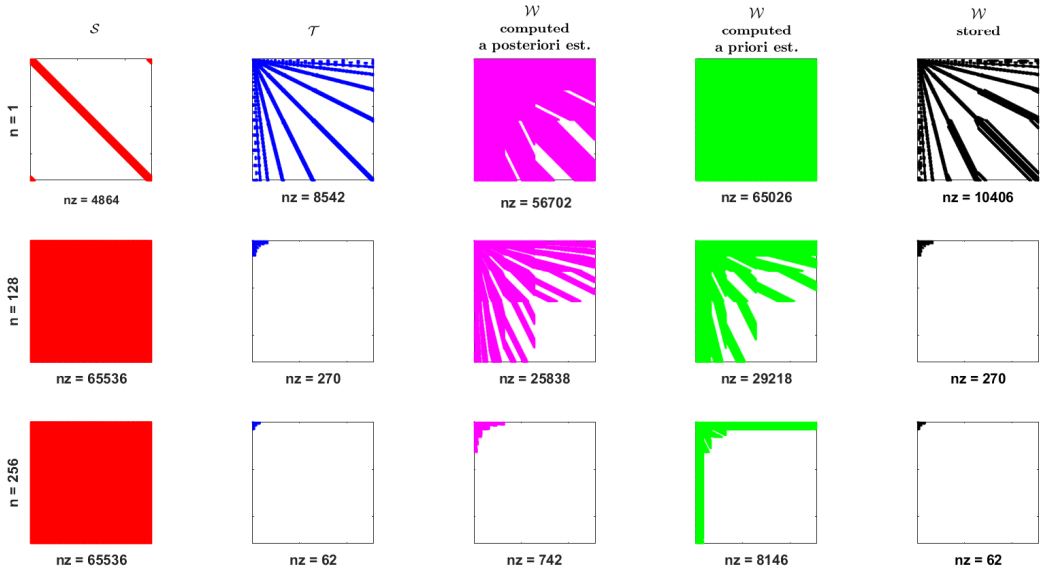


Figure 5: Test 1. Sparsity pattern of $\boldsymbol{V}^{\mathcal{S},n}$ (left column), $\boldsymbol{V}^{\mathcal{T},n}$ (middle column) and $\boldsymbol{V}^{\mathcal{W},n}$ (right column) at t_n, for $n = 0, 128, 256$. $c = 343$.

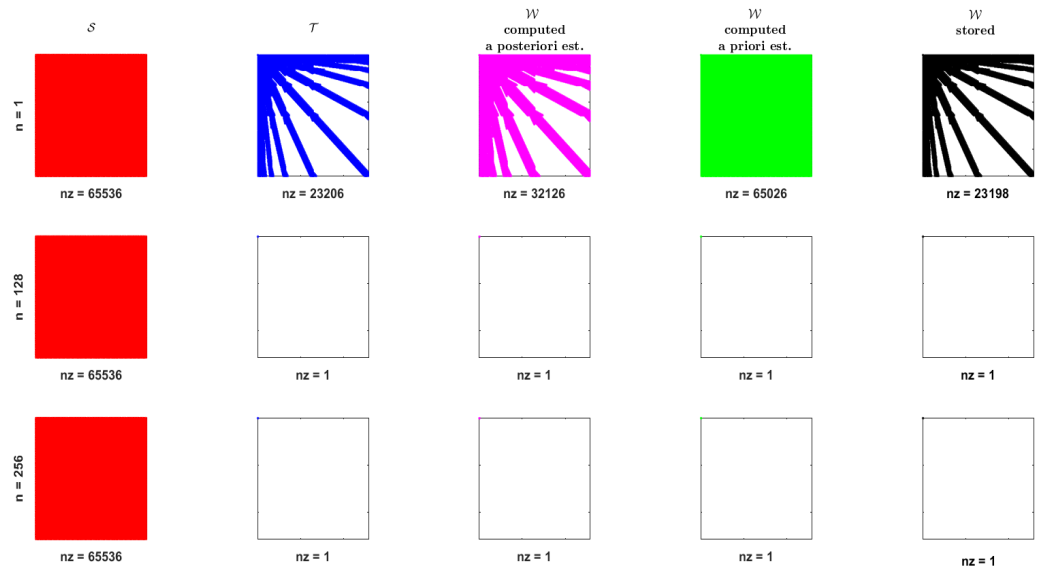
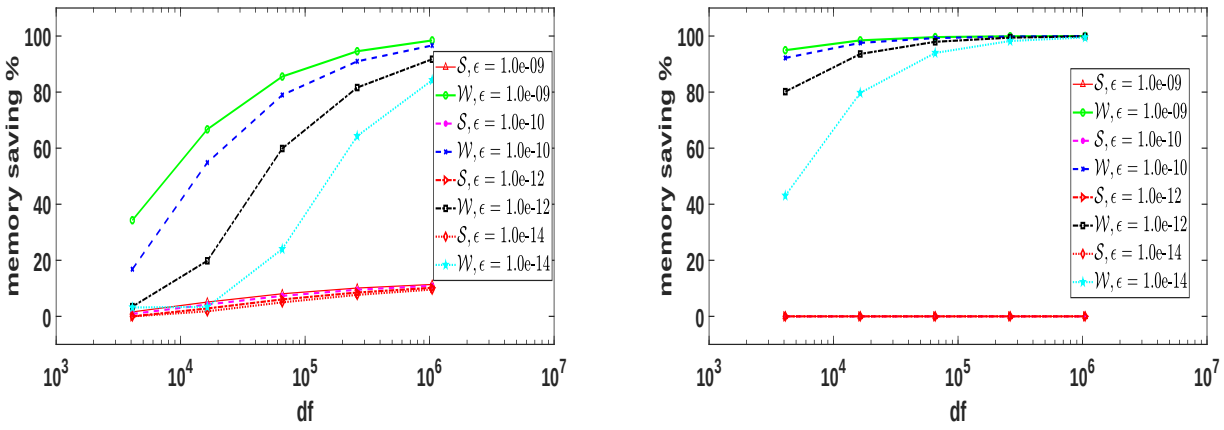


Figure 6: Test 2. Comparison of the memory saving for $\mathcal{S}$ and $\mathcal{W}$, for $T = 10$ and increasing values of df, and different thresholds ε. $c = 1$ (left plot) and $c = 343$ (right plot).



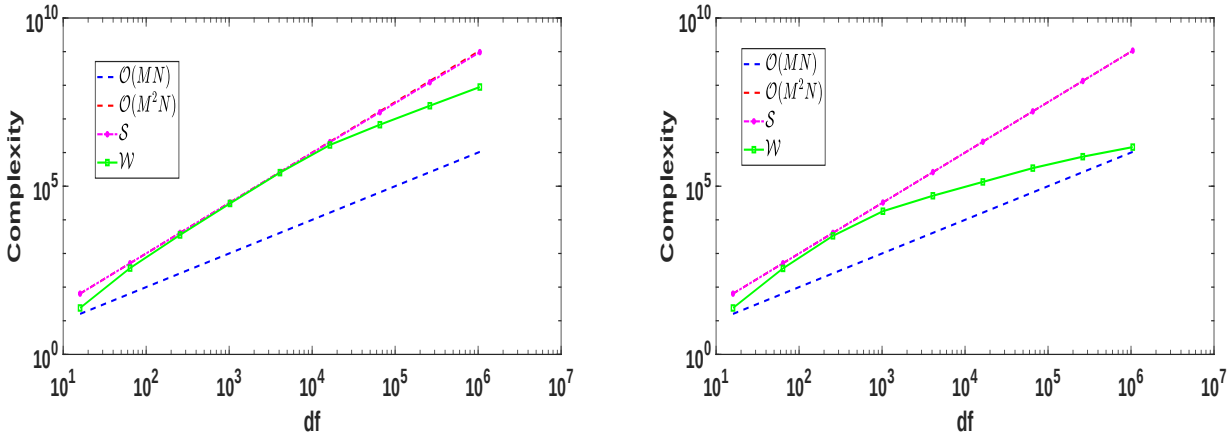
Test 2: storage requirement

To give an idea of the behavior of the storage requirement, with respect to the space and time refinements, we compare the computed storage requirements with that needed by the full matrix representation (which is $O(M^2)$ for each $n = 0, \ldots, N$, and $O(NM^2)$ globally). In Figure 6, we illustrate the behavior of the memory saving $\text{mem}^{\mathcal{S}}$ and $\text{mem}^{\mathcal{W}}$ for increasing values of M and N (in particular doubling each of them), that is for increasing number of degrees of freedom (df) and for the choices of the threshold $\varepsilon = 1.0e-09, 1.0e-10, 1.0e-12, 1.0e-14$. As we can see, the behavior of the quantity $\text{mem}^{\mathcal{S}}$ confirms that the approach $\mathcal{S}$ does not benefit from the cutting strategy, independently of ε and for any choice of the velocity c. On the other hand, the behavior of the quantity $\text{mem}^{\mathcal{W}}$ clearly illustrates that the cutting strategy is effective for the approach $\mathcal{W}$ and that the memory storage depends on the choice of ε for small numbers of df, but is asymptotically independent of ε, for any choice of the velocity c.

Figure 7, in which we consider a fix threshold $\varepsilon = 1.0e-12$, shows the global complexity $O(NM^2)$ of the approach $\mathcal{S}$ for both choices of the velocities $c = 1$ (left plot) and $c = 343$ (right plot). On the contrary, we observe a linear growth $O(MN)$ of the storage requirements for the approach $\mathcal{W}$ for the velocity $c = 1$ (right plot) and less than linear for the higher velocity $c = 343$ (right plot).

In the following examples, we apply the matrix compression strategy to solve some problems of type (1), by exployting the efficiency of the sparsification not only in terms of memory saving, but also in terms of accuracy of the approximate solutions $\boldsymbol{\varphi}^*$ and $\boldsymbol{u}^*$. In all the examples, for simplicity, the potential $\boldsymbol{u}^*$ will be evaluated at the external point $\mathbf{x} = (2, 0)$.

Figure 7: Test 2. Comparison of the complexity for $\mathcal{S}$ and $\mathcal{W}$, for $T = 10$ and increasing values of df and $\varepsilon = 1.0e-12$. $c = 1$ (left plot) and $c = 343$ (right plot).



Example 1

We consider Problem (1) with

$$g(\mathbf{x}, t) = t^4 e^{-2t} \cos(x_1^2 + 2x_2^2)$$

and $T = 10$.

In Tables 1 and 2 we report the quantities (28) and (29) for the approaches $\mathcal{S}$ and $\mathcal{T}$, and the corresponding estimated order of convergence (EOC), for the velocities $c = 1$ and $c = 343$, respectively. In the last two columns of the tables we compare the memory saving of the two approaches, having chosen $\varepsilon = 1.0e-10$.

According to the choice of the wavelet basis, for which we recall that $m = \widetilde{m} = 2$, estimate (19) in our case reads

$$\|\boldsymbol{\varphi}(\cdot, t_n) - \boldsymbol{\varphi}_L^n\|_{H^{-1/2}(\Gamma)} = \mathcal{O}(\Delta_t^2) + \mathcal{O}(\Delta_x^{5/2}),$$

uniformly with respect to $t_n \in [0, T]$. In the numerical tests, we have computed a discrete L^2-norm of the errors produced by the proposed numerical method. In this case, taking into account a known inverse-type inequality for negative Sobolev norms (see, for example, [16]), we obtain the following error estimate

$$\|\boldsymbol{\varphi}(\cdot, t_n) - \boldsymbol{\varphi}_L^n\|_{L^2(\Gamma)} = \mathcal{O}(\Delta_t^2) + \mathcal{O}(\Delta_x^2),$$

which is confirmed by the numerical results.

The reference solutions $\boldsymbol{\varphi}^{\mathrm{ex}}_{M_\mathrm{e},N_\mathrm{e}}$ and $\boldsymbol{u}^{\mathrm{ex}}_{M_\mathrm{e},N_\mathrm{e}}$ have been obtained by choosing the discretization parameters $M_\mathrm{e} = 2^9$ and $N_\mathrm{e} = 2048$. As it can be seen, in spite of the high compression, the accuracy of the solutions obtained by the approach $\mathcal{T}$ is the same of the one of the approach $\mathcal{S}$, for which the compression is very low and, in some cases, even null.

We remark that the approach $\mathcal{W}$ has produced the same values (at least up to the second significant digit) for all the quantities reported in the two tables, with the

exception of the values $M = 256$ and $N = 1024$ of the last row, for which the error is slightly larger. This mismatch is due to the spatial numerical quadrature employed for the computation of the matrix entries. Indeed, for the approach $\mathcal{W}$, an "ad hoc" quadrature strategy is needed, and this will be the object of our study in a forthcoming research.

Table 1: Example 1. Errors, convergence orders and memory savings for $\mathcal{S}$ and $\mathcal{T}$. c=1.

M	N	$\mathrm{E}^{\mathcal{S}}_{L^2,\varphi}$	EOC	$\mathrm{E}^{\mathcal{T}}_{L^2,\varphi}$	EOC	$\mathrm{E}^{\mathcal{S}}_{\max,u}$	EOC	$\mathrm{E}^{\mathcal{T}}_{\max,u}$	EOC	$\mathrm{mem}^{\mathcal{S}}$	$\mathrm{mem}^{\mathcal{T}}$
4	16	$7.90e-01$		$7.90e-01$		$1.76e-02$		$1.76e-02$		0%	63%
			1.7		1.7		0.9		0.9		
8	32	$2.38e-01$		$2.38e-01$		$9.16e-03$		$9.16e-03$		0%	28%
			2.0		2.0		1.4		1.4		
16	64	$5.88e-02$		$5.88e-02$		$3.55e-03$		$3.55e-03$		0%	13%
			0.9		0.9		1.8		1.8		
32	128	$3.07e-02$		$3.07e-02$		$1.00e-03$		$1.00e-03$		3%	7%
			1.5		1.5		2.0		2.0		
64	256	$1.10e-02$		$1.10e-02$		$2.55e-04$		$2.55e-04$		6%	34%
			2.1		2.1		1.9		1.9		
128	512	$2.66e-03$		$2.66e-03$		$6.89e-05$		$6.89e-05$		9%	70%
			2.5		2.5		2.3		2.3		
256	1024	$4.69e-04$		$4.83e-04$		$1.38e-05$		$1.38e-05$		10%	87%

Table 2: Example 1. Errors, convergence orders and memory savings for $\mathcal{S}$ and $\mathcal{T}$. c=343.

M	N	$\mathrm{E}^{\mathcal{S}}_{L^2,\varphi}$	EOC	$\mathrm{E}^{\mathcal{T}}_{L^2,\varphi}$	EOC	$\mathrm{E}^{\mathcal{S}}_{\max,u}$	EOC	$\mathrm{E}^{\mathcal{T}}_{\max,u}$	EOC	$\mathrm{mem}^{\mathcal{S}}$	$\mathrm{mem}^{\mathcal{T}}$
4	16	$4.57e-01$		$4.57e-01$		$6.83e-05$		$6.83e-05$		0%	63%
			1.0		1.0		-0.6		-0.6		
8	32	$2.28e-01$		$2.28e-01$		$1.07e-04$		$1.07e-04$		0%	39%
			2.7		2.7		4.2		4.2		
16	64	$3.49e-02$		$3.49e-02$		$5.97e-06$		$5.97e-06$		0%	70%
			2.5		2.5		1.9		1.9		
32	128	$6.01e-03$		$6.01e-03$		$1.60e-06$		$1.60e-06$		0%	90%
			2.5		2.5		2.0		2.0		
64	256	$1.04e-03$		$1.04e-03$		$4.03e-07$		$4.03e-07$		0%	97%
			2.6		2.6		2.0		2.0		
128	512	$1.75e-04$		$1.75e-04$		$9.69e-08$		$9.69e-08$		0%	99%
			2.8		2.8		2.3		2.3		
256	1024	$2.48e-05$		$2.53e-05$		$2.00e-08$		$2.00e-08$		0%	99.7%

Example 2

We consider Problem (1) with $c = 1$,

$$g(\mathbf{x}, t) = t^4 e^{-2t}$$

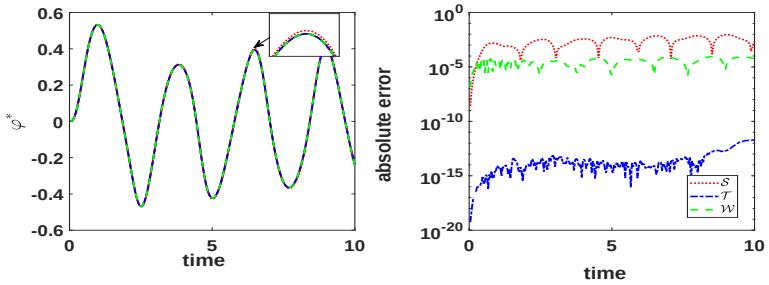
and $T = 10$. Since the resulting solution does not depend on the variable $\mathbf{x}$, it is sufficient to consider the low level of spatial discretization $L = 3$. We discretize the time interval $[0, T]$ in $N = 256$ subintervals and we choose the coarse threshold parameter $\varepsilon = 1.0e - 04$.

In the left plots of Figures 8 and 9 we show the behavior in time of the approximate solutions $\boldsymbol{\varphi}^{\mathcal{W}}_{M,N}$ and $\boldsymbol{u}^{\mathcal{W}}_{M,N}$ (green dashed line), $\boldsymbol{\varphi}^{\mathcal{S}}_{M,N}$ and $\boldsymbol{u}^{\mathcal{S}}_{M,N}$ (red dotted line), $\boldsymbol{\varphi}^{\mathcal{T}}_{M,N}$ and $\boldsymbol{u}^{\mathcal{T}}_{M,N}$ (blue dotted-dashed line). In the right plots, we report the corresponding absolute errors. The reference solutions $\boldsymbol{\varphi}^{\text{ex}}_{M_\text{e},N_\text{e}}$ and $\boldsymbol{u}^{\text{ex}}_{M_\text{e},N_\text{e}}$ are obtained by choosing the discretization parameters $M_\text{e} = M = 2^3$ and $N_\text{e} = N$. We remark that the memory saving quantities are $\text{mem}^{\mathcal{T}} = \text{mem}^{\mathcal{W}} \approx 77\%$ and $\text{mem}^{\mathcal{S}} \approx 6\%$.

As we can see, the worst solution is the one associated to the approach $\mathcal{S}$, in spite of the fact that the cutting strategy has maintained almost all the entries with a resulting low memory saving. It is worth noting that, for this approach, the threshold ε reveals to be too coarse.

On the contrary, the wavelet approach allows to obtain an accurate solution with a high memory saving. We point out that the discrepancy in the accuracy between the approaches $\mathcal{W}$ and $\mathcal{T}$ is essentially due to the numerical computation of the spatial integrals involved in the expression of the matrix entries, as already remarked in the previous example.

Figure 8: Example 2. Approximate density functions $\boldsymbol{\varphi}^*(\mathbf{x}, t)$, $\mathbf{x} = (-1, 0)$, $* = \mathcal{S}, \mathcal{T}, \mathcal{W}$ (left) and corresponding absolute errors (right).



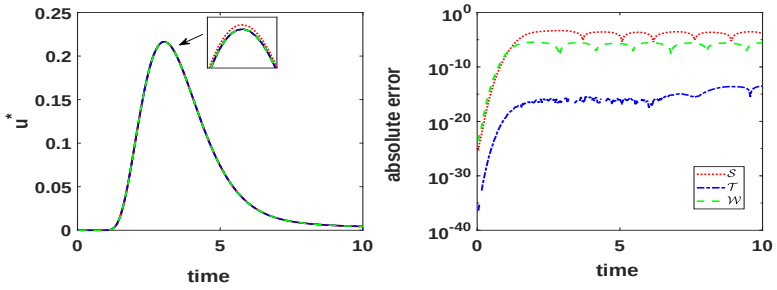
Example 3

We consider here Problem (1) with $c = 343$,

$$g(\mathbf{x}, t) = t^4 e^{-t} \cos(x_1^2 + 4x_2^2)$$

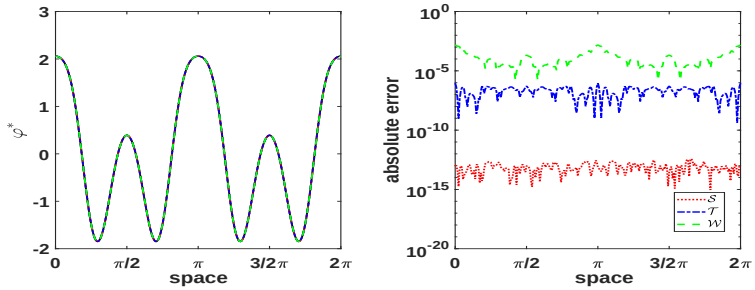
and $T = 10$. The discretization parameters are $L = 8$ and $N = 256$, and the threshold is $\varepsilon = 1.0e - 12$. The reference solutions $\boldsymbol{\varphi}^{\text{ex}}_{M_\text{e},N_\text{e}}$ and $\boldsymbol{u}^{\text{ex}}_{M_\text{e},N_\text{e}}$ are obtained by choosing

Figure 9: Example 2. Approximate potential functions $\mathbf{u}^*(\mathbf{x},t)$, $\mathbf{x}=(2,0)$, $*=\mathcal{S},\mathcal{T},\mathcal{W}$ (left) and corresponding absolute errors (right).



the discretization parameters $M_{\rm e}=M=2^8$ and $N_{\rm e}=N$. With these choices, it results that almost the whole storage of the matrices $\boldsymbol{V}^{\mathcal{S},n}$ associated to the standard approach is necessary, so that the memory saving turns out to be $\text{mem}^{\mathcal{S}}\approx 0.01\%$; this justifies the fact that the errors associated to the density function and to the potential solution are of the order of the machine precision. On the other hand, the total memory saving for the approaches $\mathcal{W}$ and $\mathcal{T}$ is $\text{mem}^{\mathcal{W}}=\text{mem}^{\mathcal{T}}\approx 98\%$. The maximum value of the corresponding errors is approximately $1.0e-03$ for $\boldsymbol{\varphi}^{\mathcal{W}}$, $1.0e-06$ for $\boldsymbol{\varphi}^{\mathcal{T}}$, $1.0e-07$ for $\boldsymbol{u}^{\mathcal{W}}$ and $1.0e-10$ for $\boldsymbol{u}^{\mathcal{T}}$.

Figure 10: Example 3. Approximate density functions $\boldsymbol{\varphi}^*(\mathbf{x},t)$, $\mathbf{x}=(-1,0)$, $*=\mathcal{S},\mathcal{T},\mathcal{W}$ (left) and corresponding absolute errors (right).



Example 4

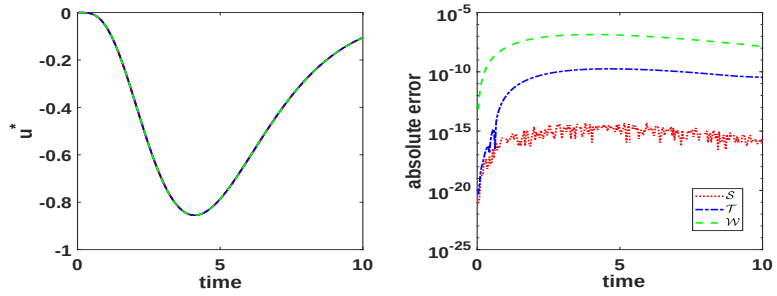
We consider the scattered field of a plane incident wave packet, impinging upon the unit disc with velocity $c=343$. The incident wave

$$u_{inc}(\mathbf{x},t)=e^{-2(x_1-50+ct)^2}+e^{-2(x_1-60+ct)^2}+e^{-2(x_1-70+ct)^2},$$

$\mathbf{x}=(x_1,x_2)$, consists in the sum of three successive waves spaced at regular intervals.

The time interval of interest $[0,0.4]$ is subdivided into $N=2048$ subintervals; the parametrization interval of the boundary Γ is discretized by choosing $L=5$. We remark that the chosen fine time discretization is necessary to reproduce the highly oscillating

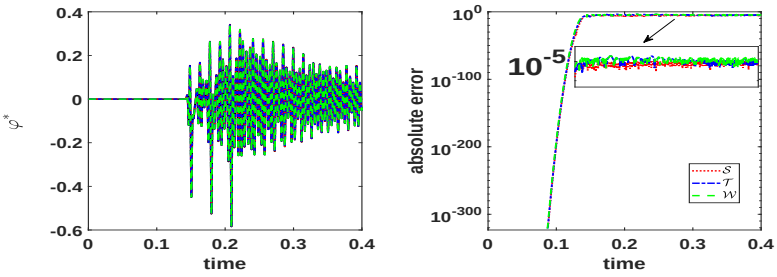
Figure 11: Example 3. Approximate potential functions $\boldsymbol{u}^*(\mathbf{x},t)$, $\mathbf{x}=(2,0)$, $*=\mathcal{S},\mathcal{T},\mathcal{W}$ (left) and corresponding absolute errors (right).



behavior of the density function (see Figure 12) and the very narrowed and picked behavior of the scattered field u_{scatt}, solution of Problem (1) with $g(\mathbf{x},t)=-u_{inc}(\mathbf{x},t)$ (see Figure 13).

The threshold parameter is $\varepsilon = 1.0e-08$. The reference solutions $\boldsymbol{\varphi}^{\text{ex}}_{M_\text{e},N_\text{e}}$ and $\boldsymbol{u}^{\text{ex}}_{M_\text{e},N_\text{e}}$ are obtained by choosing the discretization parameters $M_\text{e}=M=2^5$ and $N_\text{e}=N$. With these choices, it results that a high storage for the matrices $\boldsymbol{V}^{\mathcal{S},n}$ associated to the standard approach is necessary, so that the global memory saving is $\text{mem}^{\mathcal{S}}=0.4\%$. On the other hand, the total memory saving for the wavelet approach is $\text{mem}^{\mathcal{W}}=\text{mem}^{\mathcal{T}}\approx 93\%$, and the accuracy of the solutions $\boldsymbol{\varphi}^*$ and $\mathbf{u}^*$, $*=\mathcal{T},\mathcal{W}$ is preserved, as it can be seen in the plots corresponding to the error behavior. Indeed, the error is comparable with that of the approach $\mathcal{S}$.

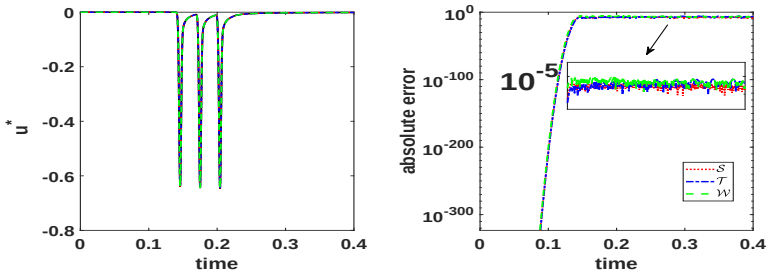
Figure 12: Example 4. Approximate density functions $\boldsymbol{\varphi}^*(\mathbf{x},t)$, $\mathbf{x}=(-1,0)$, $*=\mathcal{S},\mathcal{T},\mathcal{W}$ (left) and corresponding absolute errors (right).



Example 5

As last example, we present a case for which the approximate solution requires very fine space and time discretization parameters, that turn out to be prohibitive for the approaches $\mathcal{S}$ and $\mathcal{T}$ in terms of memory space. Indeed our PC prevented the execution of the schemes $\mathcal{S}$ and $\mathcal{T}$ because of an out of memory. Therefore, the compression strategy of the new approach revealed to be crucial for this simulation.

Figure 13: Example 4. Approximate potential functions $\boldsymbol{u}^*(\mathbf{x},t)$, $\mathbf{x}=(2,0)$, $*=\mathcal{S},\mathcal{T},\mathcal{W}$ (left) and corresponding absolute errors (right).



For this example, we consider a wave that propagates with velocity $c=343$, generated by the highly oscillating in time Dirichlet datum

$$g(\mathbf{x},t)=t^2e^{-t}\sin(512t)\cos(x_1^2+32x_2^2).$$

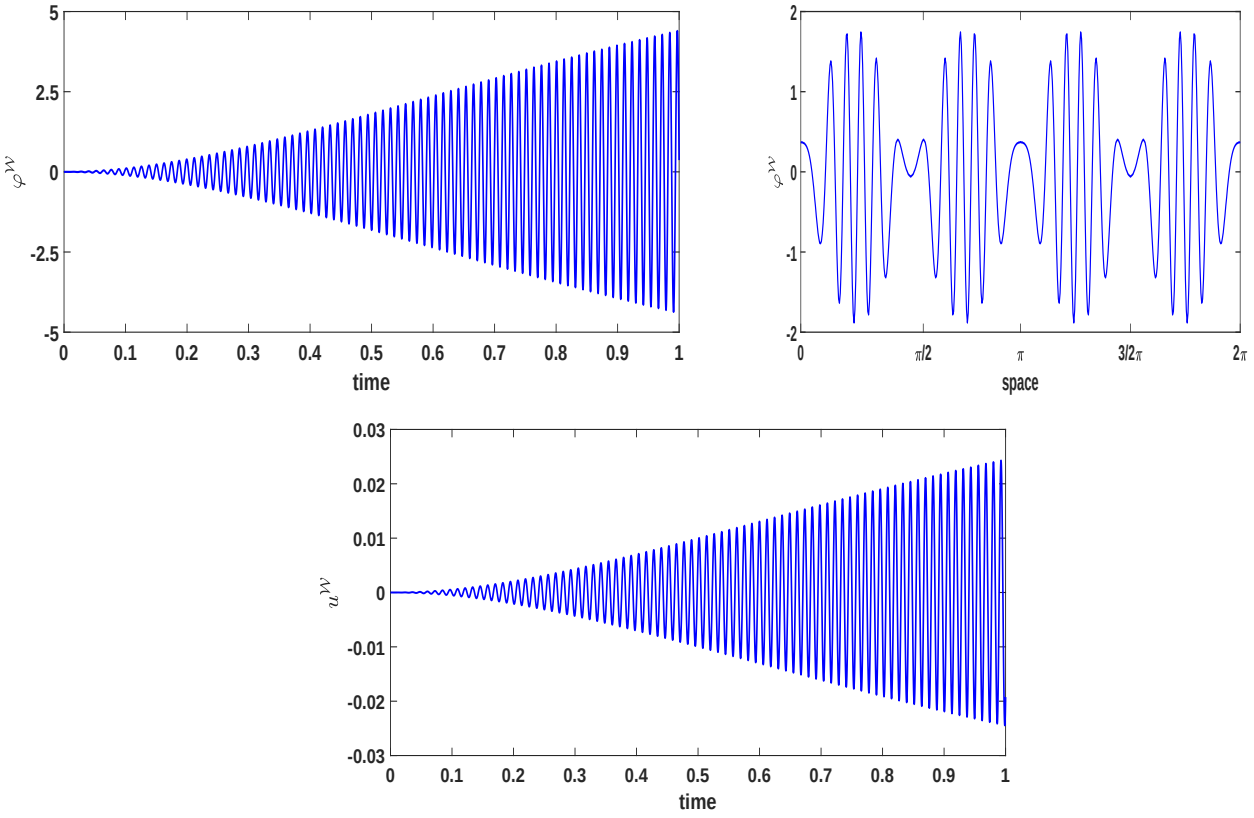
We solve the BIE in the time interval $[0,1]$ using fine spatial and temporal discretizations, obtained by choosing $L=9$ and $N=4096$. These are necessary to reproduce accurately the oscillating behavior of the density function and of the potential. The threshold parameter is $\varepsilon=1.0e-10$. In Figure 14 we show the behavior of the solutions $\boldsymbol{\varphi}^{\mathcal{W}}$ and $\boldsymbol{u}^{\mathcal{W}}$, computed with a memory saving $\text{mem}^{\mathcal{W}}\approx 99.6\%$. In particular, in the left plot we show the behavior in time of $\boldsymbol{\varphi}^{\mathcal{W}}(\mathbf{x},t)$ at $\mathbf{x}=(-1,0)$, in the middle plot the behavior of $\boldsymbol{\varphi}^{\mathcal{W}}(\mathbf{x},T)$ for $\mathbf{x}\in\Gamma$ and in the right plot the behavior in time of $\boldsymbol{u}^{\mathcal{W}}(\mathbf{x},t)$ at the external point $\mathbf{x}=(2,0)$.

4 Conclusions

We have considered a boundary integral formulation of an exterior 2D wave propagation problem. For the resolution of the corresponding BIE, we have used a wavelet Galerkin method in space coupled with a Lubich convolution quadrature in time. The coupling of the two schemes is new and, although wavelet approximations have been successfully applied to BEMs, there is not much work on their use for the resolution of time-dependent BIEs. We have devised an approach, denoted by $\mathcal{W}$, that combines the good properties of the wavelet approximation in space and those of the Lubich convolution quadrature in time. Based on an a priori estimate of the decaying behavior of the matrix entries and on a downsampled FFT, this approach offers two main advantages: a high compression of the matrices and a considerable reduction of the computational cost for the time discretization.

We have compared this approach with: the approach $\mathcal{S}$ that uses piece-wise linear spatial approximating functions and a Lubich quadrature in time, with an a posteriori cutting; the approach $\mathcal{T}$ that uses wavelet spatial approximating functions, by means of a wavelet transform, and a Lubich quadrature time, with an a posteriori cutting.

Figure 14: Example 5. Behavior of the approximate density function $\boldsymbol{\varphi}^{\mathcal{W}}(\mathbf{x}, t)$, in time for $\mathbf{x} = (-1, 0)$ (top-left) and in space for $t = T$ (top-right), and the potential solution $\boldsymbol{u}^{\mathcal{W}}(\mathbf{x}, t)$, for $\mathbf{x} = (2, 0)$ (bottom).



As shown in Section 3, the approach $\mathcal{S}$ does not permit a matrix compression strategy (independently of the wave propagation velocity) and, consequently, it does not allow neither a computational cost reduction nor a memory saving.

On the contrary, thanks to the wavelet properties, the approach $\mathcal{T}$ permits a significant compression of the matrices (depending on the wave propagation velocity) but this allows only an a posteriori cutting and a computational cost reduction in the resolution of the final linear system. Indeed, this approach can not be taken into account for the memory saving since the compression is possible only after that the matrices associated to the approach $\mathcal{S}$ have been entirely computed and stored.

Therefore, the approach $\mathcal{W}$, which allows simultaneously the reduction of the computational cost and the memory saving, turns out to be crucial when applied to large scale problems, for which the computation and the storage of all the matrices becomes prohibitive.

References

[1] A. Aimi, M. Diligenti, C. Guardasoni, I. Mazzieri, and S. Panizzi. An energy approach to space-time Galerkin BEM for wave propagation problems. *Internat. J. Numer. Methods Engrg.*, 80:1196–1240, 2009.

[2] A. Bamberger and T. Ha Duong. Formulation variationelle espace-temps pour le calcul par potentiel retardé de la diffraction d'une onde acoustique. *Math. Meth. in the Appl. Sci*, 8:405–435, 1986.

[3] G. Beylkin, R. Coifman, and V. Rokhlin. Fast wavelet transforms and numerical algorithms. I. *Comm. Pure Appl. Math.*, 44(2), 1991.

[4] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland, Amsterdam, 1978.

[5] A. Cohen, I. Daubechies, and J. Feauveau. Bi-orthogonal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, 45:485–560, 1992.

[6] W. Dahmen, H. Harbrecht, and R. Schneider. Compression techniques for boundary integral equations – Asymptotically optimal complexity estimates. *SIAM J. Numer. Anal.*, 43(6):2251–2271, 2006.

[7] W. Dahmen, S. Prössdorf, and R. Schneider. Wavelet approximation methods for pseudodifferential equations. II. Matrix compression and fast solution. 1:259–335, 1993.

[8] Ingrid Daubechies. *Ten Lectures on Wavelets*. SIAM, 2004.

[9] S. Falletta. BEM coupling with the FEM fictitious domain approach for the solution of the exterior Poisson problem and of wave scattering by rotating rigid bodies. *IMA J. Numer. Anal.*, 38(2):779–809, 2018.

[10] S. Falletta and G. Monegato. An exact non reflecting boundary condition for 2D time-dependent wave equation problems. *Wave Motion*, 51(1):168–192, 2014.

[11] S. Falletta and G. Monegato. Exact non-reflecting boundary condition for 3D time-dependent multiple scattering-multiple source problems. *Wave Motion*, 58:281–302, 2015.

[12] 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.

[13] S. Falletta, G. Monegato, and L. Scuderi. A space-time BIE method for wave equation problems: the (two-dimensional) Neumann case. *IMA J. Numer. Anal.*, 34(1):390–434, 2014.

[14] S. Falletta, G. Monegato, and L. Scuderi. On the discretization and application of two space-time boundary integral equations for 3D wave propagation problems in unbounded domains. *App. Num. Math.*, 124:22–43, 2018.

[15] S. Falletta and S. Sauter. The panel-clustering method for the wave equation in two spatial dimensions. *J. Comput. Physics*, 305:217–243, 2016.

[16] I.G. Graham, W. Hackbusch, and S. Sauter. Finite elements on degenerate meshes: inverse-type inequalities and applications. *IMA J. Numer. Anal.*, 25:379–407, 2005.

[17] L. Greengard and V. Rokhlin. A new version of the fast multipole method for the Laplace equation in three dimensions. *Acta Numerica*, 6:229–269, 1997.

[18] W. Hackbusch. *The Panel Clustering Technique for the Boundary Element Method*, volume 9/1. Springer, 1987.

[19] W. Hackbusch and Z.P. Nowak. On the fast matrix multiplication in the boundary element method by panel clustering. *Numer. Math.*, 54(4):463–491, 1989.

[20] H. Harbrecht and R. Schneider. *Wavelet Galerkin schemes for 2D-BEM*, volume 121 of *Oper. Theory Adv. Appl.*, pages 221–260. Birkhuser, Basel, 2001.

[21] G. C. Hsiao and A. Rathsfeld. Wavelet collocation methods for a first kind boundary integral equation in acoustic scattering. *Adv. Comput. Math.*, 17(4):281–308, 2002.

[22] K. Koro and K. Abe. Application of Haar wavelets to time-domain BEM for the transient scalar wave equation. *IOP Conference Series: Materials Science and Engineering*, 10(1):1–10, 2010.

[23] C. Lage and C. Schwab. Wavelet Galerkin algorithms for boundary integral equations. *SIAM J. Sci. Comput.*, 20(6):2195–2222, 1999.

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

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

[26] G. Monegato and L. Scuderi. A space-time BIE method for 2D mixed wave equation problems. *Appl. Math. Comp.*, 259:1046–1070, 2015.