Max-Planck-Institut
für Mathematik
in den Naturwissenschaften
Leipzig

Wave Propagation Problems treated with Convolution Quadrature and BEM

(revised version: October 2010)

by

Lehel Banjai, and Martin Schanz

Preprint no.: 60 2010

Wave Propagation Problems treated with Convolution Quadrature and BEM

Lehel Banjai and Martin Schanz

1 Introduction – State of the Art

The Boundary Element Method (BEM) in time domain is especially important for treating wave propagation problems in semi-infinite or infinite domains. In this application the main advantage of this method becomes obvious, i.e., its ability to model the radiation condition correctly. Certainly this is not the only advantage of a time domain BEM but very often the main motivation as, e.g., in earthquake engineering or scattering problems. The mathematical background of time-dependent boundary integral equations is summarized by Costabel [27].

Scattering problems have been treated very early with integral equations where some solution techniques may be seen as a BEM in time domain, e.g., [37]. For elastodynamics the first boundary integral formulation was published by Cruse and Rizzo [28]. However, this formulation performs in Laplace domain with a subsequent inverse transformation to time domain to achieve results for the transient behavior. The corresponding formulation in Fourier domain, i.e., frequency domain, was presented by Domínguez [35]. The first boundary element formulation directly in the time domain was developed by Mansur for the scalar wave equation and for elastodynamics with zero initial conditions [56]. The extension of this formulation to non-zero initial conditions was presented by Antes [5]. A completely different approach to handle dynamic problems utilizing static fundamental solutions is the so-called dual reciprocity BEM. This method was introduced by Nardini and Brebbia [62] and details may be found in the monograph of Partridge et al [64]. A very detailed review of elastodynamic boundary element formulations and a list of ap-

Lehel Banjai
Max-Planck Institute for Mathematics in the Sciences, Inselstrasse 22, 04103 Leipzig, Germany, e-mail: banjai@mis.mpg.de

Martin Schanz
Graz University of Technology, Institute of Applied Mechanics, Technikerstr. 4, 8010 Graz, Austria, e-mail: m.schanz@tugraz.at

plications can be found in two articles of Beskos [15, 16]. Fast formulations for elastodynamics based on a plane wave expansion has been published by Otani et al [63] and Takahashi et al [80].

An important area of applications of time (and frequency) domain boundary integral equations is electrodynamics. Variational methods initiated for acoustics [9] have been extended to electromagnetism in [81, 83, 7, 66] and also to FEM-BEM coupling in the time domain [8]. Collocation methods are also here of great importance in applications [34]. There has been a very important development in fast methods for electrodynamics [77, 85, 25] where fast multipole methods for high-frequency problems [25, 24] have been extended to the time-domain. All these methods have been known to experience stability problems in longer time computations [32, 33, 76], but various remedies have over the years been developed [32, 33, 31, 30, 76]. In particular, as in the frequency domain case, the combined integral equations give rise to more stable methods [76].

The above listed methodologies to treat time dependent problems with the BEM can be split in two main groups: direct computation in time domain or inverse transformation combined with computation in Laplace domain. Not only due to the dependency of numerical inverse transformations on some sophisticated parameter, but also due to physical reasons it is more natural to work in the real time domain and observe the phenomenon as it evolves. But, as all time-stepping procedures, such a formulation requires an adequate choice of the time step size. An improperly chosen time step size leads to instabilities or numerical damping. An improved and stable version of the underlying integral equation has been published by Bamberger and Ha-Duong [9] and Aimi and Diligenti [3]. Both rely on an energy principle and require two temporal integrations. The instabilities of the usual time-stepping algorithm have been analysed by Birgisson et al [19]. Four procedures to improve the stability of the classical dynamic time-stepping BE formulation can be quoted: the first employs modified numerical time marching procedures, e.g., [6] for acoustics, [65] for elastodynamics; the second employs a modified fundamental solution, e.g., [67] for elastodynamics; the third employs an additional integral equation for velocities [57]; and the last uses weighting methods, e.g., [87] for elastodynamics and [88] for acoustics.

Beside these improved approaches there exist the possibility to solve the convolution integral in the boundary integral equation with the so-called Convolution Quadrature Method (CQM) proposed by Lubich [51, 52]. Applications to hyperbolic and parabolic integral equations can be found in [55, 53]. The CQM utilizes the Laplace domain fundamental solution and results not only in a more stable time stepping procedure but also damping effects in case of visco- or poroelasticity can be taken into account (see [73, 74, 71]). The motivation to use the CQM in these engineering applications is that only the Laplace domain fundamental solutions are required. This fact is also used for BE formulations in cracked anisotropic elastic [89] or piezoelectric materials [39]. Another aspect is the better stability behavior compared with the above mentioned formulation. For acoustics this may be found in [1, 2] and in elastodynamics in [72]. Recently work has begun in investigating CQM for electromagnetism [83]. In the framework of fast BE formulations the

CQM is used in a Panel-clustering formulation for the Helmholtz equation by Hackbusch et al [46]. Recently, some newer mathematical aspects of the CQM have been published by Lubich [54]. Further, interest in high order Runge-Kutta based CQM has lately increased due to its good performance in applications, see [10] for numerical experiments in acoustics and [12, 14, 22] for convergence results.

In this paper, both, the linear multistep and Runge-Kutta based CQM is described together with most recent theoretical results on convergence, the application to various linear hyperbolic problems is explained, and the paper ends with a numerical experiment for an elastodynamic problem. Important for the paper at hand are different approaches to the implementation of CQM. The originally proposed construction of convolution weights by fast Fourier transform (FFT) [52] is described, also the recent decoupling approach promoted in [13], and the recursive method of [10], a modification of [48].

Throughout this paper, vectors and tensors are denoted by bold symbols and matrices by sans serif and upright symbols. The Laplace transform of a function $f(t)$ is denoted by $\hat{f}(s)$ with the complex Laplace parameter $s \in \mathbb{H}$ and $\mathbb{H} = \{s \in \mathbb{C} | \Re s > 0\}$.

2 Time Dependent Boundary Integral Equations

In this work linear hyperbolic differential equations are considered. The most simple equation is the scalar wave equation. However, vectorial problems will also be tackled and, hence, the basic equations are described for the simplest vectorial problem, for elastodynamics.

2.1 Governing Equations

Describing with $\mathbf{x}$ and t the position in the three-dimensional Euclidean space $\mathbb{R}^3$ and the time point from the interval $(0,\infty)$ the hyperbolic initial value problem for the displacement field $\mathbf{u}(\mathbf{x},t)$ is

$$\begin{aligned} c_1^2 \nabla\nabla\cdot\mathbf{u}(\mathbf{x},t) - c_2^2 \nabla\times\nabla\times\mathbf{u}(\mathbf{x},t) &= \frac{\partial^2 \mathbf{u}}{\partial t^2}(\mathbf{x},t) && (\mathbf{x},t) \in \Omega\times(0,\infty) \\ \mathbf{u}(\mathbf{y},t) &= \mathbf{g}_D(\mathbf{y},t) && (\mathbf{y},t) \in \Gamma_D\times(0,\infty) \\ \mathbf{t}(\mathbf{y},t) &= \mathbf{g}_N(\mathbf{y},t) && (\mathbf{y},t) \in \Gamma_N\times(0,\infty) \\ \mathbf{u}(\mathbf{x},0) = \frac{\partial \mathbf{u}}{\partial t}(\mathbf{x},0) &= \mathbf{0} && (\mathbf{x},t) \in \Omega\times(0)\,. \end{aligned} \tag{1}$$

The material properties of the solid are represented by the wave speeds

$$c_1 = \sqrt{\frac{K+\frac{4}{3}G}{\rho}} \qquad c_2 = \sqrt{\frac{G}{\rho}}\,,$$

with the material data compression modulus K, shear modulus G, and the mass density ρ. The first statement in (1) requires the fulfillment of the partial differential equation in the spatial domain Ω for all times $0 < t < \infty$. This spatial domain Ω has the boundary Γ which is subdivided into two disjoint sets Γ_D and Γ_N at which boundary conditions are prescribed. The Dirichlet boundary condition is the second statement of (1) and assigns a given datum $\mathbf{g}_D$ to the displacement $\mathbf{u}$ on the part Γ_D of the boundary. Similarly, the Neumann boundary condition is the third statement in which the datum $\mathbf{g}_N$ is assigned to the surface traction $\mathbf{t}$, which is defined by

$$\begin{aligned}\mathbf{t}(\mathbf{y},t) =& (\mathscr{T}\mathbf{u})(\mathbf{y},t)\\ =& \lim_{\Omega\ni\mathbf{x}\to\mathbf{y}\in\Gamma}[\left(G\left(\nabla\mathbf{u}+(\nabla\mathbf{u})^T\right)+\left(K-\frac{2}{3}G\right)\nabla\cdot\mathbf{u}\,\mathbf{I}\right)(\mathbf{x},t)\cdot\mathbf{n}(\mathbf{y})] \qquad (2)\\ =& \lim_{\Omega\ni\mathbf{x}\to\mathbf{y}\in\Gamma}[\boldsymbol{\sigma}(\mathbf{x},t)\cdot\mathbf{n}(\mathbf{y})]\,.\end{aligned}$$

In (2), $\boldsymbol{\sigma}$ is the stress tensor depending on the displacement field $\mathbf{u}$ according to the linear strain-displacement relationship and Hooke's law. For later purposes the traction operator $\mathscr{T}$ is defined, which maps the displacement field $\mathbf{u}$ to the surface traction $\mathbf{t}$. The boundary conditions have to hold for all times and may be also prescribed in each direction by different types, e.g., roller bearings. Finally, in the last statement of (1) the condition of a quiescent past is given which implies homogeneous initial conditions.

Beside the elastodynamic problem, a number of other wave propagation problems describing different physical phenomena can be treated similarly. The respective governing differential equations are listed next.

2.1.1 Acoustics – scalar wave equation

The hyperbolic differential equation for waves traveling in a non-viscous fluid is

$$c^2\nabla^2 p\,(\mathbf{x},t) = \frac{\partial^2 p}{\partial t^2}\,(\mathbf{x},t) \quad (\mathbf{x},t)\in\Omega\times(0,\infty)\,, \qquad (3)$$

with boundary conditions defined analogously to (1) and also vanishing initial conditions. The wave velocity is defined by

$$c = \sqrt{\frac{K}{\rho}}$$

with the compressibility K of the fluid. The traction operator (2) degenerates to the normal derivative to define the normal flux

$$q_n(\mathbf{y},t) = (\mathscr{T}p)(\mathbf{y},t) = \lim_{\Omega \ni \mathbf{x} \to \mathbf{y} \in \Gamma} [\nabla p(\mathbf{x},t) \cdot \mathbf{n}(\mathbf{y})] \,.$$

2.1.2 Viscoelastodynamics

This extension of the elastodynamic case to materials with damping can easily performed with the elastic-viscoelastic correspondence principle [26]. This principle says that in Laplace domain the material data has simply to be exchanged by the viscoelastic material data which are dependent on the Laplace variable s, i.e., they are time dependent. Consequently, the governing differential equation is the Laplace transform of (1) to Laplace domain

$$c_{1v}^2(s)\,\nabla\nabla\cdot\hat{\mathbf{u}}(\mathbf{x},s) - c_{2v}^2(s)\,\nabla\times\nabla\times\hat{\mathbf{u}}(\mathbf{x},s) = s^2\hat{\mathbf{u}}(\mathbf{x},s) \quad (\mathbf{x},s)\in\Omega\times\mathbb{H}\,, \tag{4}$$

with the viscoelastic wave speeds

$$c_{1v}(s) = \sqrt{\frac{\hat{K}(s)+\frac{4}{3}\hat{G}(s)}{\rho}} \qquad c_{2v} = \sqrt{\frac{\hat{G}(s)}{\rho}}\,. \tag{5}$$

The material data $\hat{K}(s)$ and $\hat{G}(s)$ can for most materials given as rational functions of s, e.g., for the simplest causal model - the three parameter model - it holds

$$\hat{K}(s) = K\frac{1+q^H s}{1+p^H s} \qquad \hat{G}(s) = G\frac{1+q^D s}{1+p^D s}\,, \tag{6}$$

with the compression modulus K and the shear modulus G from elasticity. The parameters q^H, q^D, p^H, and p^D are further material data. More details on viscoelastic constitutive equations may be found in [26] and their implementation in BEM in [72, 40].

The traction operator is defined as in elastodynamics where Hooke's law has now the material data from (6), i.e., the constitutive equation in time domain is a convolution integral. This and also the structure of (4) shows that a formulation of the problem in time domain yields an integro-differential equation.

2.1.3 Poroelastodynamics

The wave propagation in saturated two-phase media as, e.g., soil is governed by a coupled set of differential equations for the solid displacements $\mathbf{u}$ and the pore pressure p. Beside mixture theory based approaches (see, e.g., the Theory of Porous Media [20] or the simple mixture theory [86]), Biot's theory is widely used in practice and will also be used here. The basic formulation for wave propagation problems can be found in the two papers [17, 18]. The set of governing equations in Laplace domain is

$$
\begin{aligned}
G\nabla^2\hat{\mathbf{u}}(\mathbf{x},s)+\left(K+\frac{1}{3}G\right)\nabla\nabla\cdot\hat{\mathbf{u}}(\mathbf{x},s)-(\alpha-\beta(s))\nabla\hat{p}=&s^2\left(\rho-\beta(s)\rho_f\right)\hat{\mathbf{u}}(\mathbf{x},s)\\
\frac{\beta(s)}{s\rho_f}\nabla^2\hat{p}(\mathbf{x},s)-\frac{\phi^2 s}{R}\hat{p}(\mathbf{x},s)-(\alpha-\beta(s))s\nabla\cdot\hat{\mathbf{u}}(\mathbf{x},s)=&0\,,\\
&(\mathbf{x},s)\in\Omega\times\mathbb{H}
\end{aligned}
\tag{7}
$$

with the bulk material data shear modulus G and compression modulus K, Biot's coefficients α and R, and the porosity ϕ. The bulk density is denoted by $\rho=(1-\phi)\rho_s+\phi\rho_f$, composed by the partial densities of the solid ρ_s and the fluid ρ_f. The complex valued parameter $\beta(s)$ is an abbreviation and defined as

$$
\beta(s)=\frac{\kappa\rho_f\phi^2 s^2}{\phi^2 s+s^2\kappa\left(\rho_a+\phi\rho_f\right)}
$$

with the permeability κ and the apparent mass density ρ_a. As in viscoelasticity, this set of governing equations can not be formulated as a pure differential equation in time domain because the coefficients depend on s. The wave velocities, due to the incorporated friction between the solid and the fluid, are time dependent. The respective wave numbers, defined as usual $\lambda=\frac{s}{c}$, are

$$
\begin{aligned}
\lambda_{1,2}^2=&\frac{s^2}{2}\left[\frac{\phi^2\rho_f}{\beta(s)R}+\frac{\rho-\beta(s)\rho_f}{K+\frac{4}{3}G}+\frac{\rho_f(\alpha-\beta(s))^2}{\beta(s)\left(K+\frac{4}{3}G\right)}\right.\\
&\left.\pm\sqrt{\left(\frac{\phi^2\rho_f}{\beta(s)R}+\frac{\rho-\beta(s)\rho_f}{K+\frac{4}{3}G}+\frac{\rho_f(\alpha-\beta(s))^2}{\beta(s)\left(K+\frac{4}{3}G\right)}\right)^2-4\frac{\phi^2\rho_f\left(\rho-\beta(s)\rho_f\right)}{\beta(s)R\left(K+\frac{4}{3}G\right)}}\right],\\
\lambda_3^2=&\frac{s^2\left(\rho-\beta(s)\rho_f\right)}{G}\,.
\end{aligned}
$$

Compared to the above given models in poroelasticity three waves, a fast and slow compressional wave and a shear wave, exist.

The traction operator has to be seen in a generalized way and has obviously two parts. It is composed of the definition of the total stress and the flux governed by Darcy's law

$$
\begin{bmatrix}\hat{\mathbf{t}}\\ \hat{q}\end{bmatrix}(\mathbf{y},s)=(\mathscr{T}\begin{bmatrix}\hat{\mathbf{u}}\\ \hat{p}\end{bmatrix})(\mathbf{y},s)=\lim_{\Omega\ni\mathbf{x}\to\mathbf{y}\in\Gamma}\begin{bmatrix}[\hat{\sigma}-\alpha\hat{p}\mathbf{I}](\mathbf{x},s)\cdot\mathbf{n}(\mathbf{y})\\ \left[-\frac{\beta}{s\rho_f}\left(\nabla\hat{p}+\rho_f s^2\hat{\mathbf{u}}\right)\right](\mathbf{x},s)\cdot\mathbf{n}(\mathbf{y})\end{bmatrix}\,.
$$

2.1.4 Electromagnetism – Maxwell equations

The system of Maxwell equations in a homogeneous and isotropic medium is given by

$$\begin{aligned} \mu \frac{\partial \mathbf{H}}{\partial t}(\mathbf{x},t) + \nabla \times \mathbf{E}(\mathbf{x},t) &= 0 \\ \varepsilon \frac{\partial \mathbf{E}}{\partial t}(\mathbf{x},t) - \nabla \times \mathbf{H}(\mathbf{x},t) &= 0, \end{aligned} \tag{8}$$

with $\mathbf{E}$ and $\mathbf{H}$ being the electric and magnetic field, respectively, and ε and μ respectively electric permittivity and magnetic permeability. Boundary conditions are obtained by a combination of tangential traces of the two fields: $\mathbf{n}\times\mathbf{E}$ and $\mathbf{n}\times\mathbf{H}$, e.g., $\mathbf{n}\times\mathbf{E}=0$ for a perfectly conducting surface and the impedance boundary condition $\mathbf{n}\times\mathbf{H}-\alpha(\mathbf{n}\times\mathbf{E})\times\mathbf{n}=0$, $\alpha\geq 0$, for an imperfectly conducting surface [60].

The relationship to wave equations can be made more visible by rewriting the first order system (8) as a second order system. This can be done by, for example, eliminating the magnetic field $\mathbf{H}$ and thereby obtaining the equation

$$-c^2\,\nabla\times\nabla\times\mathbf{E}(\mathbf{x},t) = \frac{\partial^2\mathbf{E}}{\partial t^2}(\mathbf{x},t)\ ,$$

with the wave speed $c=\frac{1}{\sqrt{\varepsilon\mu}}$.

2.2 Integral Equations

For all of the governing equations given above, a representation formula can be derived (see, e.g., for acoustics [61], for elastodynamics [84], for viscoelastodynamics [42], for poroelastodynamics [72], and for electromagnetism [78, Chapter 25]). Representation formula for Maxwell equations does not fit the general framework of the other equations, therefore it is presented separately.

Taking $\mathbf{u}$ as representative for the unknowns in the governing equations (1), (3), (4), and (7) the representation formula is

$$\begin{aligned} \mathbf{u}(\mathbf{x},t) = &\int_0^t\int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{t}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_{\mathbf{y}}\,\mathrm{d}\,\tau - \\ &\int_0^t\int_\Gamma (\mathscr{T}_{\mathbf{y}}\mathbf{U})(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{u}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_{\mathbf{y}}\,\mathrm{d}\,\tau \qquad \mathbf{x}\in\Omega,\mathbf{y}\in\Gamma\ . \end{aligned} \tag{9}$$

The surface measure $\mathrm{d}\Gamma_{\mathbf{y}}$ carries its subscript in order to emphasize that the integration variable is $\mathbf{y}$. Similarly, $\mathscr{T}_{\mathbf{y}}$ indicates that the derivatives involved in the computation of the surface traction are taken with respect to the variable $\mathbf{y}$. The function $\mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)$ denotes the fundamental solution of the respective governing equation. In the Laplace domain, the fundamental solutions of all of the above given problems can be formulated in 3-d as

$$\hat{\mathbf{U}}(\mathbf{x}-\mathbf{y},s) = \sum_{i=1}^{w} \mathbf{A}_i(r,s)\,\frac{e^{-\lambda_i r}}{4\pi r} \qquad \text{with} \quad r = |\mathbf{x}-\mathbf{y}|\,, \tag{10}$$

using the wave number $\lambda_i = \frac{s}{c_i}$ instead of the wave velocities c_i. The upper limit w of the sum in (10) is the amount of body waves in the model. The coefficients $\mathbf{A}_i(r,s)$ are listed in the Appendix. In 2-d, the structure of the fundamental solution is the same, however, the exponential function has to be replaced by the modified Bessel functions of zero or first order. Time dependent fundamental solutions are only available for acoustics, elastodynamics, and electromagnetism, but even here, for example for elastodynamics and the dissipative wave equation in acoustics, the time domain fundamental solution can become very complex. In the following, this problem is overcome by using the CQM for time discretisation.

By means of equation (9), the unknown $\mathbf{u}$ is given at any point $\mathbf{x}$ inside the domain Ω and at any time $0 < t < \infty$, if the boundary data $\mathbf{u}(\mathbf{y},\tau)$ and $\mathbf{t}(\mathbf{y},\tau)$ are known for all points $\mathbf{y}$ of the boundary Γ and times $0 < \tau < t$. The first boundary integral equation is obtained by taking the expression (9) to the boundary. Using operator notation, this boundary integral equation reads

$$(\mathscr{V}\mathbf{t})(\mathbf{x},t) = \mathscr{C}(\mathbf{x})\mathbf{u}(\mathbf{x},t) + (\mathscr{K}\mathbf{u})(\mathbf{x},t) \qquad (\mathbf{x},t) \in \Gamma \times (0,\infty)\,. \tag{11}$$

The introduced operators are the single layer operator $\mathscr{V}$, the integral-free term $\mathscr{C}$, and the double layer operator $\mathscr{K}$ which are defined as

$$(\mathscr{V}\mathbf{t})(\mathbf{x},t) = \int_0^t \int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{t}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \tag{12a}$$

$$\mathscr{C}(\mathbf{x}) = \mathscr{I} + \lim_{\varepsilon\to 0} \int_{\partial B_\varepsilon(\mathbf{x})\cap\Omega} (\mathscr{T}_\mathbf{y}\mathbf{U}_{\text{static}})^\top(\mathbf{x}-\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y} \tag{12b}$$

$$(\mathscr{K}\mathbf{u})(\mathbf{x},t) = \lim_{\varepsilon\to 0} \int_0^t \int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} (\mathscr{T}_\mathbf{y}\mathbf{U})^\top(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{u}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau\,. \tag{12c}$$

In these expressions, $B_\varepsilon(\mathbf{x})$ denotes a ball of radius ε centered at $\mathbf{x}$ and $\partial B_\varepsilon(\mathbf{x})$ is its surface. In (12b), the integral free term is only determined by the static counterpart of each operator, i.e., the index $_{\text{static}}$ denotes the respective fundamental solution. E.g., in elastodynamics $\mathbf{U}_{\text{static}}$ is the elastostatic fundamental solution. Note that the single layer operator (12a) involves a weakly singular integral over Γ and the double layer operator (12c) has to be understood in the sense of a principal value. Further, it should be remarked that the operator notation in (12a) and (12c) includes the convolution operator in time.

Application of the traction operator $\mathscr{T}_x$ to the dynamic representation formula (9) yields the second boundary integral equation

$$(\mathscr{D}\mathbf{u})(\mathbf{x},t) = (\mathscr{I} - \mathscr{C}(\mathbf{x}))\,\mathbf{t}(\mathbf{x},t) - (\mathscr{K}'\mathbf{t})(\mathbf{x},t) \qquad \mathbf{x} \in \Gamma\,. \tag{13}$$

The newly introduced operators are the adjoint double layer operator $\mathscr{K}'$ and the hyper-singular operator $\mathscr{D}$. They are defined as

$$(\mathscr{K}'\mathbf{t})(\mathbf{x},t) = \lim_{\varepsilon\to 0} \int_0^t \int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} (\mathscr{T}_{\mathbf{x}}\mathbf{U})(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{t}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_{\mathbf{y}}\,\mathrm{d}\,\tau$$

$$(\mathscr{D}\mathbf{u})(\mathbf{x},t) = -\lim_{\varepsilon\to 0} \int_0^t \mathscr{T}_{\mathbf{x}} \int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} (\mathscr{T}_{\mathbf{y}}\mathbf{U})^\top(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{u}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_{\mathbf{y}}\,\mathrm{d}\,\tau\,.$$

The hyper-singular operator has to be understood in the sense of a finite part.

For the solution of mixed initial boundary value problems, a non-symmetric formulation by means of the first boundary integral equation (11) in combination with a collocation technique will be used. A symmetric formulation is obtained using both the first and the second boundary integral equation, (11) and (13) in combination with a Galerkin technique.

Symmetric formulation

First, the Dirichlet datum $\mathbf{u}$ and the Neumann datum $\mathbf{t}$ are decomposed into

$$\mathbf{u} = \tilde{\mathbf{u}} + \tilde{\mathbf{g}}_D \qquad \text{and} \qquad \mathbf{t} = \tilde{\mathbf{t}} + \tilde{\mathbf{g}}_N\,, \tag{15}$$

with arbitrary but fixed extensions, $\tilde{\mathbf{g}}_D$ and $\tilde{\mathbf{g}}_N$, of the given Dirichlet and Neumann data, $\mathbf{g}_D$ and $\mathbf{g}_N$. They are introduced such that

$$\begin{aligned} \tilde{\mathbf{g}}_D(\mathbf{x},t) &= \mathbf{g}_D(\mathbf{x},t)\,, \qquad (\mathbf{x},t) \in \Gamma_D \times (0,\infty) \\ \tilde{\mathbf{g}}_N(\mathbf{x},t) &= \mathbf{g}_N(\mathbf{x},t)\,, \qquad (\mathbf{x},t) \in \Gamma_N \times (0,\infty) \end{aligned}$$

holds. The extension $\tilde{\mathbf{g}}_D$ of the given Dirichlet datum has to be continuous due to regularity requirements [79].

In order to establish a symmetric formulation, the first boundary integral equation (11) is used only on the Dirichlet boundary Γ_D whereas the second one (13) is used only on the Neumann part Γ_N. Taking the prescribed boundary conditions (1) into account and inserting the decompositions (15) into both integral equations leads to the symmetric formulation for the unknowns $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{t}}$

$$\begin{aligned} \mathscr{V}\tilde{\mathbf{t}} - \mathscr{K}\tilde{\mathbf{u}} &= \mathbf{f}_D, \quad (\mathbf{x},t) \in \Gamma_D \times (0,\infty) \\ \mathscr{D}\tilde{\mathbf{u}} + \mathscr{K}'\tilde{\mathbf{t}} &= \mathbf{f}_N, \quad (\mathbf{x},t) \in \Gamma_N \times (0,\infty) \end{aligned} \tag{16}$$

with the abbreviations

$$\begin{aligned} \mathbf{f}_D &= \mathscr{C}\tilde{\mathbf{g}}_D + \mathscr{K}\tilde{\mathbf{g}}_D - \mathscr{V}\tilde{\mathbf{g}}_N \\ \mathbf{f}_N &= (\mathscr{I} - \mathscr{C})\,\tilde{\mathbf{g}}_N - \mathscr{K}'\tilde{\mathbf{g}}_N - \mathscr{D}\tilde{\mathbf{g}}_D\,. \end{aligned}$$

Representation formula for Maxwell equations

The representation formula has the following form for the electric field

$$\begin{aligned}\mathbf{E}(\mathbf{x},t) = &-\mu \int_0^t \int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\frac{\partial \mathbf{j}}{\partial t}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \\ &+\frac{1}{\varepsilon}\nabla \int_0^t \int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\partial_t^{-1}\nabla_\Gamma\cdot\mathbf{j}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \\ &-\nabla\times \int_0^t \int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{m}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau\end{aligned}$$

and the following for the magnetic field

$$\begin{aligned}\mathbf{H}(\mathbf{x},t) = &-\varepsilon \int_0^t \int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\frac{\partial \mathbf{m}}{\partial t}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \\ &+\frac{1}{\mu}\nabla \int_0^t \int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\partial_t^{-1}\nabla_\Gamma\cdot\mathbf{m}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \\ &+\nabla\times \int_0^t \int_\Gamma \mathbf{U}(\mathbf{x}-\mathbf{y},t-\tau)\mathbf{j}(\mathbf{y},\tau)\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau,\end{aligned}$$

where $\mathbf{j}=\mathbf{H}\times\mathbf{n}$ and $\mathbf{m}=\mathbf{n}\times\mathbf{E}$ are, respectively, the surface current and surface charge density. The symbol ∂_t^{-1} denotes integration on the interval $[0,t]$, this is consistent with the operational notation introduced in the next section. The fundamental solution $\hat{\mathbf{U}}(\mathbf{x},s)$ still has the form (10) and is in fact the same as the fundamental solution for the acoustic wave equation, showing the close relationship between the two sets of equations. Taking tangential traces one obtains boundary integral formulations of boundary value problems. Since the formalism using the four integral operators introduced for other governing equations does not directly translate to the Maxwell system, for further information the reader is referred to literature, see [78, Chapter 25].

3 Convolution quadrature

All of the time domain integral operators of the previous section have the form of a time convolution

$$u(t) = \int_0^t k(t-\tau)g(\tau)d\tau. \tag{17}$$

The difficulty in computing such convolutions comes from the fact that the kernel $k(t)$ is often distributional and in many cases of practical interest, e.g., viscoelasticity and poroelasticity, even not known explicitly. However, the Laplace transform of the kernel

$$K(s) = \hat{k}(s) = \mathscr{L}k(s) := \int_0^\infty k(t)e^{-st}\,\mathrm{d}t$$

is always explicitly known and simpler. For this reason it is essential to be able to compute (17) by using only the Laplace transformed kernel $K(s)$. To make this dependence on the Laplace transformed kernel explicit, operational notation, going back to Heaviside and standard in papers on convolution quadrature [53],

$$\left(K(\partial_t)g\right)(t) := \int_0^t k(t-\tau)g(\tau), \tag{18}$$

is used in this paper. The rationale behind this notation comes from identities of the type $K(\partial_t)g = g'$ for $K(s) = s$ and the composition rule $K_2K_1(\partial_t)g = K_2(\partial_t)K_1(\partial_t)g$. Convolution quadrature time discretization will be explained and convergence results given with the following assumption on the operator $K(s)$:

$$\begin{aligned} &K(s) \text{ is analytic for } \Re s > 0 \text{ and bounded as} \\ &|K(s)| \le C(\sigma_0)\frac{|s|^\mu}{(\Re s)^\nu}, \text{ for } \Re s \ge \sigma_0 > 0. \end{aligned} \tag{19}$$

To make the connection to the previous section explicit, note that in this notation the single layer operator of (12a) can be written as

$$(\mathscr{V}\mathbf{t})(\mathbf{x},t) = \left(V(\partial_t)\mathbf{t}\right)(\mathbf{x},t)$$

where V is the single layer operator in the Laplace domain:

$$(V(s)\boldsymbol{\phi})(\mathbf{x}) := \int_\Gamma \hat{\mathbf{U}}(\mathbf{x}-\mathbf{y},s)\boldsymbol{\phi}(\mathbf{y})\,\mathrm{d}s_y$$

and $\hat{\mathbf{U}}$ is the explicitly known fundamental solution in the Laplace domain, see (10).

3.1 Linear multistep based convolution quadrature

For $\Delta t > 0$ let $t_j = j\Delta t$ be the discrete time steps at which (18) is to be computed. Convolution quadrature approximation of (18) at $t = t_n$ is given by

$$\left(K(\partial_t^{\Delta t})g\right)(t_n) := \sum_{j=0}^{n} \omega_{n-j}^{\Delta t}(K)g(t_j). \tag{20}$$

Here the convolution weights $\omega_j^{\Delta t}(K)$ are defined implicitly by

$$K\left(\frac{\gamma(\zeta)}{\Delta t}\right) = \sum_{j=0}^{\infty} \omega_j^{\Delta t}(K)\zeta^j, \tag{21}$$

where $\gamma(\zeta)$ is the quotient of the generating polynomials of a linear multistep method of order p. For hyperbolic problems only A-stable methods are admissible, the most often used methods being the backward difference formulas of order 1 (BDF1/backward Euler) and order 2 (BDF2) for which

$$\gamma(\zeta) = 1-\zeta \text{ (BDF1)}, \quad \gamma(\zeta) = \frac{3}{2} - 2\zeta + \frac{1}{2}\zeta^2, \text{ (BDF2)}.$$

An important property of convolution quadrature is that the composition rule is preserved. Namely, $K_2K_1(\partial_t^{\Delta t})g = K_2(\partial_t^{\Delta t})K_1(\partial_t^{\Delta t})g$. Further, for $K(s) = s$, $K(\partial_t^{\Delta t})g = \partial_t^{\Delta t} g$ is the linear multistep approximation of the derivative g'.

A brief motivation for the approximation (20) is in order. Making use of the extension $g(t) \equiv 0$ for $t \leq 0$, the approximation (20) can be defined for all t: $\left(K(\partial_t^{\Delta t})g\right)(t) = \sum_{j=0}^{\infty} \omega_j^{\Delta t}(K) g(t-t_j)$. Taking the Laplace transformation of this expression gives

$$\mathscr{L}\left(K(\partial_t^{\Delta t})g\right)(s) = \left(\sum_{j=0}^{\infty} \omega_j^{\Delta t}(K) e^{-s\Delta t j}\right) \mathscr{L}g(s) = K\left(\frac{\gamma\left(e^{-s\Delta t}\right)}{\Delta t}\right)\mathscr{L}g(s).$$

Since $\mathscr{L}(K(\partial_t)g)(s) = K(s)\mathscr{L}g(s)$, the convolution quadrature manifests itself through the approximation $s \approx \frac{\gamma\left(e^{-s\Delta t}\right)}{\Delta t} = s + sO((s\Delta t)^p)$, p being the order of the multistep method. The restriction to A-stable methods comes from the requirement $\Re\gamma\left(e^{-s\Delta t}\right) > 0$ for $\Re s > 0$.

Next a result on convergence of the linear multistep based convolution quadrature is given, the proof of which can be found in [53].

Theorem 1 (Lubich 1994). *Let* (19) *hold,* $g(0) = g'(0) = \cdots = g^{(m-1)}(0) = 0$ *for* m *such that* $m \geq \max(p+2+\mu, p)$*, and let* $u_n = \left(K(\partial_t^{\Delta t})g\right)(t_n)$ *be the approximation obtained by convolution quadrature* (20) *based on BDF formula of order* $p = 1,2$*. Then there exists* $\bar{t} > 0$ *such that for all* $0 < \Delta t < \bar{t}$ *and* $n = 0,1,\ldots,N = T/\Delta t$ *it holds*

$$|u_n - u(t_n)| \leq C\Delta t^p \int_0^{t_n} |g^{(m)}(\tau)|\,\mathrm{d}\,\tau.$$

The constant C *is independent of* Δt *and* N*, but depends on* T *and constant* $C(\sigma_0)$ *in* (19)*.*

The result proved in [53] covers a larger class of A-stable linear multistep methods. The statement here has been restricted to BDF methods in order to shorten the exposition. The trapezoid rule does not satisfy the assumptions of the general theory given in [53] if $\mu > 0$. Recently, in [10] the convergence of the trapezoid

rule has been proved for this case and successful numerical experiments have been performed for acoustic scattering applications.

Because of the restriction to A-stable linear multistep methods, the highest order attainable is $p = 2$. To achieve higher orders of convergence one has to turn to Runge-Kutta methods. Further reasons to prefer Runge-Kutta methods are highlighted later in the paper, see Section 4.3.

3.2 Runge-Kutta based convolution quadrature

Let a Runge-Kutta method of (classical) order p and stage order q be given by its Butcher tableau $\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$ where $A \in \mathbb{R}^{m\times m}$, $b, c \in \mathbb{R}^m$; for a detailed introduction to Runge-Kutta methods see [21, 47, 49]. A Runge-Kutta method is said to be A-stable if the stability function

$$R(z) = 1 + zb^T(I - zA)^{-1}\mathbb{1}, \qquad \mathbb{1} := (1,1,\ldots,1)^T,$$

is bounded as

$$|R(z)| \leq 1, \quad \text{for } \Re z \leq 0 \text{ and } I - zA \text{ is non-singular for all } \Re z \leq 0. \tag{22}$$

To simplify expressions assume further that $b^T A^{-1} = (0,0,\ldots,1)$, i.e., that the method is stiffly accurate [47]; this in turn implies that $c_m = 1$. A further technical assumption is needed

$$|R(\mathrm{i}y)| < 1, \qquad \text{for all } |y| > 0.$$

Radau IIA and Lobatto IIIC are examples of Runge-Kutta methods satisfying all of the above conditions.

In a Runge-Kutta method computations are done not only at the equally spaced points $t_j = j\Delta t$ but also at the stages $t_j + c_\ell \Delta t$, $\ell = 1,2,\ldots,m$. Note that $c_m = 1$ implies $t_j + c_m\Delta t = t_{j+1}$. The Runge-Kutta based convolution quadrature approximation to $u(t_n + c_\ell\Delta t)$, $\ell = 1,\ldots,m$, is then given by

$$\begin{pmatrix} u_{n1} \\ \vdots \\ u_{nm} \end{pmatrix} = \left(K(\underline{\partial}_t^{\Delta t})g\right)_n := \sum_{j=0}^{n} W_{n-j}^{\Delta t}(K) \begin{pmatrix} g(t_j + c_1\Delta t) \\ \vdots \\ g(t_j + c_m\Delta t) \end{pmatrix}.$$

Here, the matrix convolution weights $W_j^{\Delta t}(K)$ are defined implicitly through a generating function

$$K\left(\frac{\Delta(\zeta)}{\Delta t}\right) = \sum_{j=0}^{\infty} W_j^{\Delta t}(K)\zeta^j,$$

with

$$\Delta(\zeta) = A^{-1} - \zeta A^{-1}\mathbb{1}b^T A^{-1}.$$

The solution at t_{n+1} is given simply by $u_{n+1} = u_{nm} = b^T A^{-1}(u_{n\ell})_{\ell=1}^m$, i.e.,

$$u_{n+1} := b^T A^{-1} \left(K(\underline{\partial_t}^{\Delta t}) g \right)_n .$$

The composition rule still holds for the stage approximation, that is, $K_2 K_1(\underline{\partial_t}^{\Delta t}) g = K_2(\underline{\partial_t}^{\Delta t}) K_1(\underline{\partial_t}^{\Delta t}) g$. This is however not true for the approximation $b^T A^{-1} K(\underline{\partial_t}^{\Delta t}) g$, whence we refrain from using the operational quadrature notation here.

First convergence results under the assumption (19) with $\nu = 0$ have been proved in [12]. Subsequently it has been noticed that, unlike in the linear multistep case, a more favourable result can be proved if $\nu > 0$. This result has been proved in [14] and is stated next. It shows that for sufficiently smooth and compatible data an order of convergence $O(\Delta t^{q+1-\mu+\nu})$ is obtained; recall that q is the *stage* order of the Runge-Kutta method.

Theorem 2. *Assume* (19)*, with* $\nu \geq 0$*. Let* $r > \max(p+\mu+1, q+1)$ *and* $g \in C^r([0,T])$ *satisfy* $g(0) = g'(0) = \cdots = g^{(r)}(0) = 0$*. Then, under the above conditions on the Runge-Kutta method there exists* $\bar{t} \geq 0$ *such that for* $0 < \Delta t < \bar{t}$ *and* $t \in [0,T]$*,*

$$|u_n - u(t_n)| \leq C(\Delta t^p + \Delta t^{q+1-\mu+\nu}) \int_0^t |g^{(r+1)}(\tau)| d\tau.$$

The constant C is independent of Δt *and g, but does depend on the Runge-Kutta method,* $\bar{t}$*, and T.*

3.3 Implementation

The implicitly defined convolution weights $\omega_j^{\Delta t}(K)$ can be computed by numerical quadrature of the Cauchy integral formula, as proposed in [52],

$$\omega_j^{\Delta t}(K) = \frac{1}{2\pi \mathrm{i}} \oint_C K\left(\frac{\gamma(\zeta)}{\Delta t}\right) \zeta^{-j-1} \,\mathrm{d}\,\zeta \approx \frac{\mathscr{R}^{-j}}{N+1} \sum_{\ell=0}^{N} K\left(\frac{\gamma\left(\mathscr{R}\zeta_{N+1}^{-\ell}\right)}{\Delta t}\right) \zeta_{N+1}^{\ell j}, \quad (23)$$

where $\zeta_{N+1} = e^{\frac{2\pi \mathrm{i}}{N+1}}$ and $0 < \mathscr{R} < 1$. The computational cost using the fast Fourier transform (FFT) to compute the sum for all $j = 0, 1, \ldots, N$, is $O(N \log N)$ and the error is $O(\mathscr{R}^{N+1})$. Due to finite precision arithmetic the accuracy is restricted to $\sqrt{\mathrm{eps}}$, where eps is the machine precision and the parameter $\mathscr{R}$ is chosen as $\mathscr{R} = \mathrm{eps}^{\frac{1}{2(N+1)}}$; see [52].

In applications it is of interest to solve a discrete convolutional system:

$$\text{Find } u_n, \text{ such that} \qquad g_n = \sum_{j=0}^{n} \omega_{n-j}^{\Delta t}(K) u_j, \quad n = 0, 1, \ldots, N, \qquad (24)$$

or the equivalent system in the Runge-Kutta case. Due to the composition rule $K_2K_1(\partial_t^{\Delta t})g = K_2(\partial_t^{\Delta t})K_1(\partial_t^{\Delta t})g$ solving this system is equivalent to computing the convolution with the operator K^{-1}:

$$u_n = \sum_{j=0}^{n} \omega_{n-j}^{\Delta t}(K^{-1})g_j = \sum_{j=0}^{N} \omega_{n-j}^{\Delta t}(K^{-1})g_j, \quad n = 0,1,\ldots,N, \tag{25}$$

with the definition, $\omega_j = 0$ for $j < 0$, which is compatible with (21). Two approaches to implementation are presented next. The first one uses the representation (25), whereas the second uses (24), but both avoid constructing the weights $\omega_j^{\Delta t}$ explicitly. The presentation is done for the linear multistep based convolution quadrature. Modifications needed in the Runge-Kutta case are explained at the end of the subsection.

3.3.1 Solving the convolutional system by computing a discrete convolution with K^{-1}

Next an efficient method for computing (25) is presented. The method has been introduced in [13] but bears similarities with Method iii) of [52].

Substituting the approximation (23), this time with K^{-1} instead of K, into (25) and after rearranging the terms the following expression is obtained

$$u_n \approx \frac{\mathscr{R}^{-n}}{N+1} \sum_{\ell=0}^{N} K^{-1}\left(\frac{\gamma\left(\mathscr{R}\zeta_{N+1}^{-\ell}\right)}{\Delta t}\right)\left[\sum_{j=0}^{N} \mathscr{R}^j g_j \zeta_{N+1}^{-\ell j}\right]\zeta_{N+1}^{\ell n}. \tag{26}$$

The term in the square bracket is the discrete Fourier transform of the vector $(g_0, \mathscr{R}g_1, \ldots, \mathscr{R}^N g_N)^T$ and hence can be computed in $O(N\log N)$ time using FFT. The outer sum represents the inverse discrete Fourier transform also computable in $O(N\log N)$ time using FFT. Thus, the whole computation can be performed in $O(N\log N)$ time and the convolution weights need never be computed explicitly. In [13], it is shown that the error of this approximation is still $O(\mathscr{R}^{N+1})$ with the accuracy again restricted to $\sqrt{\text{eps}}$ by the finite precision arithmetic.

Since computing $K^{-1}(s)$ is usually a significantly more complex and expensive operation than the computation of $K(s)$, this method can become expensive [58]. For this reason a recursive procedure is presented in the next section that requires the inversion of $K(s)$ only at the single frequency $s = \gamma(0)/\Delta t$.

3.3.2 Solving the discrete convolutional system recursively

In [10], a modification of the recursive procedure of [48] is introduced which allows the solution of (24) without ever constructing the convolution weights. This method is presented next.

First, assume that

$$\sum_{j=0}^{n} \omega_{n-j}^{\Delta t}(K) u_j = g_n,$$

has already been solved for $n = 0, 1, \ldots, N_{1/2} < N$. Then, it remains to solve

$$\sum_{j=N_{1/2}+1}^{n} \omega_{n-j}^{\Delta t}(K) u_j = g_n - \sum_{j=0}^{N_{1/2}} \omega_{n-j}^{\Delta t}(K) u_j, \qquad n = N_{1/2}+1, \ldots, N. \tag{27}$$

Once the history $\sum_{j=0}^{N_{1/2}} \omega_{n-j}^{\Delta t}(K) u_j$ is computed, the above system can be computed recursively. The expensive part is hence the computation of the history, but it can be computed efficiently using the fast Fourier transform (FFT). In order to avoid constructing the weights $\omega_j^{\Delta t}(K)$ explicitly, a scaled FFT can be used, as explained next.

Define

$$\tilde{g}_n := \sum_{j=0}^{N_{1/2}} \omega_{n-j}^{\Delta t}(K) u_j = \mathscr{R}^{-n} \sum_{j=0}^{N_{1/2}} \mathscr{R}^{n-j} \omega_{n-j}^{\Delta t}(K) \mathscr{R}^j u_j, \qquad n = N_{1/2}+1, \ldots, N,$$

for a fixed $0 < \mathscr{R} < 1$. Let $\tilde{\mathbf{g}}_{\mathscr{R}}$ be the vector obtained by a matrix-vector multiplication of the circulant matrix, whose first column is given by

$$\mathbf{c}_{\mathscr{R}}(K) := (\omega_0^{\Delta t}(K), \mathscr{R}\omega_1^{\Delta t}(K), \cdots, \mathscr{R}^N \omega_N^{\Delta t}(K))^T,$$

with the vector

$$\mathbf{u}_{\mathscr{R}} := (u_0, \mathscr{R}u_1, \cdots, \mathscr{R}^{N_{1/2}} u_{N_{1/2}}, 0, \cdots, 0)^T.$$

It is not difficult to check that

$$\mathscr{R}^{-n} (\tilde{\mathbf{g}}_{\mathscr{R}})_n = \tilde{g}_n, \qquad \text{for } n = N_{1/2}+1, \ldots, N;$$

here it is implicitly assumed that the numbering of elements in a vector begin with 0. Therefore, if $\tilde{\mathbf{g}}_{\mathscr{R}}$ can be computed efficiently and without explicitly constructing the convolution weights, then so can the history required for (27). Since circulant matrices are diagonalized by the discrete Fourier transform, in the following denoted by $\mathscr{F}_{N+1}$, it holds

$$\tilde{\mathbf{g}}_{\mathscr{R}} = \mathscr{F}_{N+1}^{-1} \operatorname{diag}(\mathscr{F}_{N+1} \mathbf{c}_{\mathscr{R}}(K)) \mathscr{F}_{N+1} \mathbf{u}_{\mathscr{R}}. \tag{28}$$

The definition of $\mathscr{F}_{N+1}$ that will be used in the following is

$$(\mathscr{F}_{N+1}\mathbf{u})_\ell = \sum_{j=0}^{N} u_j \zeta_{N+1}^{-\ell j}, \qquad \text{with } \zeta_{N+1} = e^{\frac{2\pi \mathrm{i}}{N+1}}.$$

The definition of convolution weights (21) then gives

$$(\mathscr{F}_{N+1}\mathbf{c}_{\mathscr{R}}(K))_\ell = \sum_{j=0}^{N} \mathscr{R}^j \omega_j^{\Delta t} \zeta_{N+1}^{-\ell j} = K\left(\frac{\gamma\left(\mathscr{R}\zeta_{N+1}^{-\ell}\right)}{\Delta t}\right) - \sum_{j=N+1}^{\infty} \mathscr{R}^j \omega_j^{\Delta t} \zeta_{N+1}^{-\ell j}$$
$$= K\left(\frac{\gamma\left(\mathscr{R}\zeta_{N+1}^{-\ell}\right)}{\Delta t}\right) - \sum_{k=1}^{\infty} \mathscr{R}^{k(N+1)} \left[\sum_{j=0}^{N} \mathscr{R}^j \omega_{j+k(N+1)}^{\Delta t}(K) \zeta_{N+1}^{-\ell j}\right].$$

Since the term in square brackets is again a discrete Fourier transform, considering (28) and applying $\mathscr{F}_{N+1}^{-1}$ to both sides in the above equation gives

$$\tilde{\mathbf{g}}_{\mathscr{R}} = \mathscr{F}_{N+1}^{-1} \operatorname{diag}\left[K\left(\frac{\gamma(\mathscr{R})}{\Delta t}\right), \cdots, K\left(\frac{\gamma\left(\mathscr{R}\zeta_{N+1}^{-N}\right)}{\Delta t}\right)\right] \mathscr{F}_{N+1}\mathbf{u}_{\mathscr{R}}$$
$$- \sum_{k=1}^{\infty} \mathscr{R}^{k(N+1)} \operatorname{diag}\left[\omega_{k(N+1)}^{\Delta t}, \mathscr{R}\omega_{1+k(N+1)}^{\Delta t}, \cdots, \mathscr{R}^N \omega_{N+k(N+1)}^{\Delta t}\right] \mathscr{F}_{N+1}\mathbf{u}_{\mathscr{R}}.$$

Scaling both sides with $R^{-1} := \operatorname{diag}(1, \mathscr{R}^{-1}, \ldots, \mathscr{R}^{-N})$ finally gives

$$\tilde{\mathbf{g}} = R^{-1}\mathscr{F}_{N+1}^{-1} \operatorname{diag}\left[K\left(\frac{\gamma(\mathscr{R})}{\Delta t}\right), \cdots, K\left(\frac{\gamma\left(\mathscr{R}\zeta_{N+1}^{-N}\right)}{\Delta t}\right)\right] \mathscr{F}_{N+1}\mathbf{u}_{\mathscr{R}} \qquad (29)$$
$$- \sum_{k=1}^{\infty} \mathscr{R}^{k(N+1)} \operatorname{diag}\left[\omega_{k(N+1)}^{\Delta t}(K), \cdots, \omega_{N+k(N+1)}^{\Delta t}(K)\right] \mathscr{F}_{N+1}\mathbf{u}_{\mathscr{R}}.$$

Therefore, the vector $\tilde{\mathbf{g}}$, containing the update due to the history, can be computed to an accuracy $O(\mathscr{R}^{N+1})$ by using only evaluations of the Laplace domain operator $K(s)$. Further, the computational cost is only $O(N\log N)$. The error is however restricted by finite precision eps of computations of $K(s)$ and the FFT. Therefore, the total error for computation of the correction $\tilde{g}_n$ is $\mathscr{R}^{N+1} + \mathscr{R}^{-n}\text{eps}$, $n = N_{1/2}, \ldots, N$. Hence, the best accuracy $\sqrt{\text{eps}}$ is obtained with the choice $\mathscr{R} = \text{eps}^{1/2N}$.

This procedure can be continued recursively. Thereby no convolution weights $\omega_j^{\Delta t}(K)$ need to be computed except for the first one

$$\omega_0^{\Delta t}(K) = K\left(\frac{\gamma(0)}{\Delta t}\right).$$

It is also the only operator that needs to be inverted if the recursion is performed until a 1×1 system is reached. In practice it is more common to stop the recursion once a small sized system is reached and then solve the small system using the method of Section 3.3.1. See Algorithm *SolveCQ* for the structure of such an approach. In order to solve (24) the algorithm is called with arguments *SolveCQ*(0, N, g, u, J), where a constant J defines the size of the "small" system. The cost of such a recursive procedure is $O(N\log^2 N)$ [48] since there are $\log N$ levels in the recursion and at each level an FFT is computed.

Algorithm *SolveCQ*(N_0, N_1, g, u, J)
($*$ Solves convolutional system $\sum_{j=N_0}^{n} \omega_{n-j}^{\Delta t}(K) u_j = g_n, \quad n = N_0, \ldots, N_1$ $*$)

1. **if** $N_1 - N_0 \geq J$
2. **then** $N_{1/2} = \lceil (N_1 - N_0)/2 \rceil$
3. *SolveCQ*$(N_0, N_{1/2}, g, u, J)$
4. update right-hand side

$$g_n = g_n - \sum_{j=N_0}^{N_{1/2}} \omega_{n-j}^{\Delta t}(K) u_j, \quad n = N_{1/2}+1, \ldots, N_1$$

using (29).

5. *SolveCQ*$(N_{1/2}+1, N_1, g, u, J)$
6. **else** compute

$$u_n = \sum_{j=N_0}^{n} \omega_{n-j}^{\Delta t}(K^{-1}) g_j, \quad n = N_0, \ldots, N_1,$$

using (26).

Remark 1 (Parallelization). Algorithm *SolveCQ*, see lines 6. and 4., can easily gain from the availability of a parallel architecture. In applications, the expensive part of the computation is the evaluation of the operator $K(s)$. Due to the diagonalization of the (block) circulant matrices, in both methods, this part of the computation is trivially parallel and, therefore, appropriate also for distributed memory parallel architecture.

3.3.3 A few remarks regarding the implementation of Runge-Kutta based convolution quadrature

The same procedure as explained above for the linear multistep case can also be used to implement the Runge-Kutta based convolution quadrature. Again, it is only necessary to be able to evaluate operators

$$K\left(\frac{\Delta\left(\mathscr{R}\zeta_{N+1}^{-\ell}\right)}{\Delta t}\right) \quad \text{and} \quad K\left(\frac{\Delta(0)}{\Delta t}\right) = K\left(\frac{A^{-1}}{\Delta t}\right).$$

If $\Delta(\mathscr{R}\zeta_{N+1}^{-\ell})$ has a full basis of eigenvectors, i.e., if there exist invertible matrix X and diagonal matrix $\Lambda = \operatorname{diag}(\lambda_1, \lambda_2, \cdots, \lambda_m)^T$ such that $\Delta(\mathscr{R}\zeta_{N+1}^{-\ell}) = X\Lambda X^{-1}$, then the matrix valued operator is easily computed by

$$K\left(\frac{\Delta\left(\mathscr{R}\zeta_{N+1}^{-\ell}\right)}{\Delta t}\right) = X \operatorname{diag}(K(\lambda_1/\Delta t), \cdots, K(\lambda_m/\Delta t)) X^{-1}.$$

In [10], it has been shown that there is only a single value of $\mathscr{R}\zeta_{N+1}^{-\ell}$, respectively two such values, for which $\Delta\left(\mathscr{R}\zeta_{N+1}^{-\ell}\right)$ is not diagonalizable in the case of the 2-stage Radau IIA method, and respectively, the 3-stage Radau IIA method. These particular values are very unlikely to be hit during a computation, still the condition number of the basis of eigenvectors X should, as a precaution, be examined.

4 Convolution quadrature applied to hyperbolic initial value problems

In the notation of Section 3, the time domain integral operators $\mathscr{V},\mathscr{K},\mathscr{K}'$, and $\mathscr{D}$ can be written as $V(\partial_t),K(\partial_t),K'(\partial_t)$, and $D(\partial_t)$ where V,K,K', and D are the corresponding Laplace domain operators

$$
\begin{aligned}
(V\mathbf{t})(\mathbf{x},s) &= \int_\Gamma \hat{\mathbf{U}}(\mathbf{x}-\mathbf{y},s)\mathbf{t}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \\
(K\mathbf{u})(\mathbf{x},s) &= \lim_{\varepsilon\to 0}\int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} (\mathscr{T}_\mathbf{y}\hat{\mathbf{U}})^\top(\mathbf{x}-\mathbf{y},s)\mathbf{u}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y}, \\
(K'\mathbf{t})(\mathbf{x},s) &= \lim_{\varepsilon\to 0}\int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} (\mathscr{T}_\mathbf{x}\hat{\mathbf{U}})(\mathbf{x}-\mathbf{y},s)\mathbf{t}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \\
(D\mathbf{u})(\mathbf{x},s) &= -\lim_{\varepsilon\to 0}\mathscr{T}_\mathbf{x}\int_{\Gamma\setminus B_\varepsilon(\mathbf{x})} (\mathscr{T}_\mathbf{y}\hat{\mathbf{U}})^\top(\mathbf{x}-\mathbf{y},s)\mathbf{u}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y}, \quad \text{for } \mathbf{x}\in\Gamma.
\end{aligned}
$$

Once the Cauchy data are computed, the representation formula (9) can be used to evaluate the solution inside the domain Ω. The single and double layer operators used in the representation formula are denoted by $\widetilde{V}$ and $\widetilde{K}$, i.e.,

$$
\begin{aligned}
(\widetilde{V}\mathbf{t})(\mathbf{x},s) &= \int_\Gamma \hat{\mathbf{U}}(\mathbf{x}-\mathbf{y},s)\mathbf{t}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\,\tau \\
(\widetilde{K}\mathbf{u})(\mathbf{x},s) &= \int_\Gamma (\mathscr{T}_\mathbf{y}\hat{\mathbf{U}})^\top(\mathbf{x}-\mathbf{y},s)\mathbf{u}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y}, \quad \text{for } \mathbf{x}\in\Omega.
\end{aligned}
$$

The linear multistep method based convolution quadrature of the symmetric formulation (16) is given by

$$
\begin{aligned}
(V(\partial_t^{\Delta t})\tilde{\mathbf{t}})(\mathbf{x},t_n) - (K(\partial_t^{\Delta t})\tilde{\mathbf{u}})(\mathbf{x},t_n) &= \mathbf{f}_D(\mathbf{x},t_n), \quad \mathbf{x}\in\Gamma_D \\
(D(\partial_t^{\Delta t})\tilde{\mathbf{u}})(\mathbf{x},t_n) + (K'(\partial_t^{\Delta t})\tilde{\mathbf{t}})(\mathbf{x},t_n) &= \mathbf{f}_N(\mathbf{x},t_n), \quad \mathbf{x}\in\Gamma_N
\end{aligned}
\tag{32}
$$

for $n=0,1,\ldots,N$ and with the abbreviations

$$\mathbf{f}_D = \mathscr{C}\tilde{\mathbf{g}}_D + K(\partial_t^{\Delta t})\tilde{\mathbf{g}}_D - V(\partial_t^{\Delta t})\tilde{\mathbf{g}}_N,$$
$$\mathbf{f}_N = (\mathscr{I} - \mathscr{C})\,\tilde{\mathbf{g}}_N - K'(\partial_t^{\Delta t})\tilde{\mathbf{g}}_N - D(\partial_t^{\Delta t})\tilde{\mathbf{g}}_D\,.$$

Once (32) is solved for the boundary data, the solution $\mathbf{u}$ inside the domain Ω is obtained by discretizing the representation formula as

$$\mathbf{u}_n(\mathbf{x}) = (\widetilde{V}(\partial_t^{\Delta t})\mathbf{t})(\mathbf{x}, t_n) - (\widetilde{K}(\partial_t^{\Delta t})\mathbf{u})(\mathbf{x}, t_n), \qquad \mathbf{x} \in \Omega\,. \tag{33}$$

For the Runge-Kutta based convolution quadrature, $\partial_t^{\Delta t}$ is replaced by $\underline{\partial_t}^{\Delta t}$.

4.1 Bounds in the Laplace domain

In order to be able to apply Theorem 1 and Theorem 2 to show convergence and stability of the semi-discretized symmetric formulation (32), estimates in the Laplace domain of the form (19) are needed. That is, considering the symmetric formulation in Laplace domain

$$\begin{aligned}(V(s)\hat{\mathbf{t}})(\mathbf{x}, s) - (K(s)\hat{\mathbf{u}})(\mathbf{x}, s) &= \hat{\mathbf{f}}_D(\mathbf{x}, s), \quad \mathbf{x} \in \Gamma_D \\ (D(s)\hat{\mathbf{u}})(\mathbf{x}, s) + (K'(s)\hat{\mathbf{t}})(\mathbf{x}, s) &= \hat{\mathbf{f}}_N(\mathbf{x}, s), \quad \mathbf{x} \in \Gamma_N\end{aligned} \tag{34}$$

with

$$\hat{\mathbf{f}}_D = \mathscr{C}\hat{\mathbf{g}}_D + K(s)\hat{\mathbf{g}}_D - V(s)\hat{\mathbf{g}}_N$$
$$\hat{\mathbf{f}}_N = (\mathscr{I} - \mathscr{C})\,\hat{\mathbf{g}}_N - K'(s)\hat{\mathbf{g}}_N - D(s)\hat{\mathbf{g}}_D,$$

an s-dependent bound in an appropriate norm of the solution operator

$$T(s) : (\hat{\mathbf{g}}_D, \hat{\mathbf{g}}_N) \mapsto (\hat{\mathbf{u}}, \hat{\mathbf{t}})$$

is needed. Since the kernel functions of the integral operators involved in (34) are analytic in the right half complex plane as functions of s, so are the integral operators themselves, and consequently, if it exists, the solution operator T also. If the problem is well-posed, the solution operator must be polynomially bounded in appropriate norms, but determining the degree of such a polynomial bound, μ in (19), and ν in (19), is in general difficult and for the symmetric formulation only known in the acoustic case, see [50], with the results extendible to the elastic case. For the linear multistep based convolution quadrature, see Theorem 1, the value of μ gives the smoothness of the data required for optimal convergence rate to be reached. For the Runge-Kutta method, see Theorem 2, this constant, in fact $\mu - \nu$, influences the optimal convergence rate however smooth the data may be. Accordingly, for the Runge-Kutta method it is of an extra importance to know this constant.

Bounds for various formulations with explicit dependance on s, have so far been computed for acoustics and electromagnetism. For the acoustic case, in the pioneering work of Bamberger and Ha-Duong [9] estimates

$$\|V(s)\|_{H^{-1/2}(\Gamma)\to H^{1/2}(\Gamma)} \le C\frac{|s|}{\Re s} \quad \text{and} \quad \|V^{-1}(s)\|_{H^{1/2}(\Gamma)\to H^{-1/2}(\Gamma)} \le C\frac{|s|^2}{\Re s}, \quad (35)$$

have been proved. Therefore, according to Theorem 2, the expected rate of convergence to the exact densities is $O(\Delta t^q)$, i.e., stage order q. More favourable bounds have been shown in [14] for the operator $\widetilde{V}(s)$, these imply that the rate of convergence to $\mathbf{u}(\mathbf{x},t)$ for a fixed $\mathbf{x} \in \Omega$ is $O(\Delta t^p)$, that is the full (classical) order of the Runge-Kutta method; this result is likely to extend to all of the other wave propagation problems.

In the recent work by Laliena and Sayas [50], various formulations, the symmetric coupling, FEM-BEM coupling, transmission problems, etc., have also been investigated in the acoustics case. As stated by the authors of [50] all these results are extendible to the elastic case. The bound obtained in [50] for the solution operator of the symmetric formulation is

$$\|T(s)\| \le C\frac{|s|^{5/2}}{\Re s}.$$

For the electric field integral equation (EFIE) formulation of the problem of scattering of electromagnetic waves by a perfect conductor, the corresponding bound has been given in [66, 81].

Note that for the analysis of the fully discretized problem, i.e., discretized both in time *and* space, bounds of the type (35) are needed also for the spatially discretized integral operators [53].

4.2 Properties of convolution weights

It is instructive to investigate the shape of the convolution weights for the various boundary integral operators. In this section, the single layer operator convolution weights $\omega_j^{\Delta t}(V)$ and $W_j^{\Delta t}(V)$ for the acoustic and viscoelastodynamic equations are investigated.

These have the form

$$\omega_j^{\Delta t}(V)\mathbf{t} = \int_\Gamma \widetilde{\omega}_j^{\Delta t}(\mathbf{x}-\mathbf{y})\mathbf{t}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y} \qquad \text{and} \qquad W_j^{\Delta t}(V)\mathbf{t} = \int_\Gamma \widetilde{W}_j^{\Delta t}(\mathbf{x}-\mathbf{y})\mathbf{t}(\mathbf{y})\,\mathrm{d}\Gamma_\mathbf{y},$$

the kernels being given by generating functions

$$\hat{\mathbf{U}}(\mathbf{z},\gamma(\zeta)/\Delta t) = \sum_{j=0}^{\infty} \widetilde{\omega}_j^{\Delta t}(\mathbf{z})\zeta^j \qquad \text{and} \qquad \hat{\mathbf{U}}(\mathbf{z},\Delta(\zeta)/\Delta t) = \sum_{j=0}^{\infty} \widetilde{W}_j^{\Delta t}(\mathbf{z})\zeta^j.$$

For the backward Euler method and the acoustic wave equation, the kernels $\widetilde{\omega}_j^{\Delta t}(V)$ can be given explicitly:

$$\widetilde{\omega}_j^{\Delta t}(\mathbf{z}) = \frac{e^{-\frac{|\mathbf{z}|}{c\Delta t}}}{4\pi|\mathbf{z}|}\left(\frac{|\mathbf{z}|}{c\Delta t}\right)^j \frac{1}{j!}, \qquad \text{BDF1 for the wave equation.}$$

From this formula and Stirling's approximation of $j!$ it is not difficult to see that $\widetilde{\omega}_j^{\Delta t}(\mathbf{z})$ is close to zero except for $|\mathbf{z}|/c \approx j\Delta t$. This is not surprising since the kernel function in this case approximates, in a certain sense, the Dirac delta distribution $\frac{\delta(t_j-|\mathbf{z}|/c)}{4\pi|\mathbf{z}|}$. Explicit formulas for $\widetilde{\omega}_j^{\Delta t}(\mathbf{z})$ in the case of BDF2 can be given in terms of Hermite polynomials [46]. The width of the intervals to which $|\mathbf{z}|$ needs to belong to in order that $|\widetilde{\omega}_j^{\Delta t}(\mathbf{z})| > \varepsilon$ for some $\varepsilon > 0$ have been investigated in [46]. For Runge-Kutta methods such estimates do not exist as yet, but numerical experiments [10], suggest that the width of this band is considerably smaller for high-order Runge-Kutta methods.

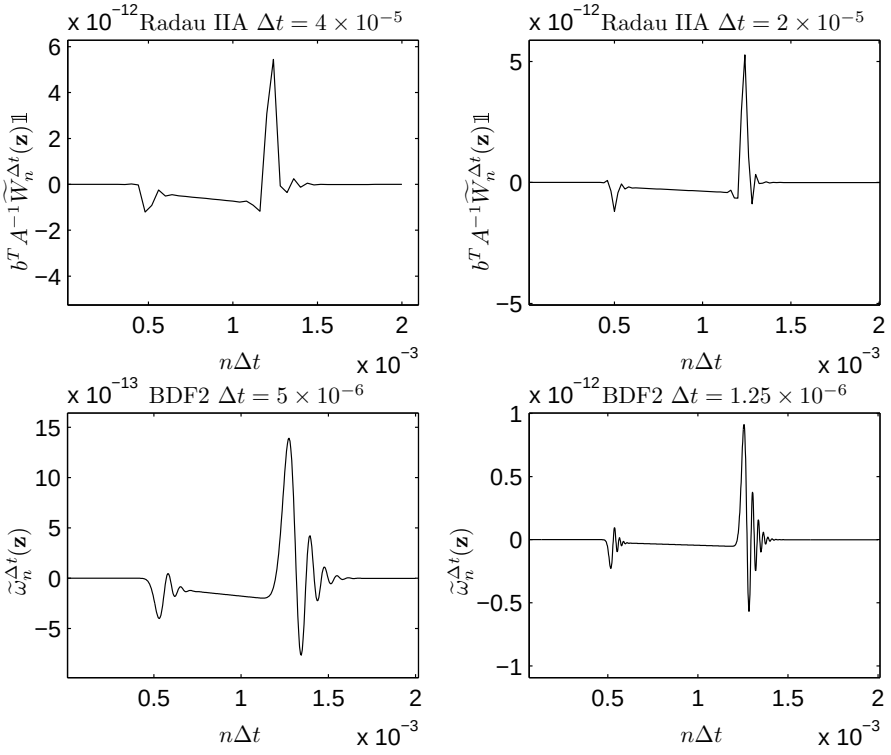


Fig. 1: Plots of the $i = 1$ and $j = 2$ entry of the tensors $\widetilde{\omega}_n^{\Delta t}(\mathbf{z})$ and $b^T A^{-1}\widetilde{W}_n^{\Delta t}(\mathbf{z})\mathbb{1} = \sum_{l=1}^m \left(\widetilde{W}_n^{\Delta t}(\mathbf{z})\right)_{ml}$ with $\mathbf{z} = (1,1,0)^T$ for 3-stage Radau IIA and BDF2 methods.

Because of the increased complexity of viscoelastodynamics compared to acoustics it is particularly of interest to investigate the shape of the kernel functions in this case. In Figure 1, the shapes are compared for different choices of Δt and the

underlying linear multistep or Runge-Kutta method. For the Runge-Kutta method the sum of the last row of $\widetilde{W}_j^{\Delta t}(\mathbf{z}) \in \mathbb{R}^{m\times m}$ is plotted; in fact each component has a similar shape. For this plot, the measured material data of a Perspex (PMMA) are used, i.e., the material constants in (5) and (6), are set with

$$K = 6.2\times 10^9\,\mathrm{N/m^2}, \quad G = 1.33\times 10^9\,\mathrm{N/m^2}, \quad \rho = 1184\,\mathrm{kg/m^3}$$
$$q^H = q^D = 0.0023\,\mathrm{1/s}, \quad p^H = p^D = 0.002\,\mathrm{1/s}.$$

The two waves, with different speeds of propagation, can nicely be seen in these plots. It is also seen that for the Runge-Kutta method the fronts are much better localised and with less non-physical oscillation for considerably larger Δt than for the BDF2 kernels. This suggests that the higher order brings also qualitative advantages, that is, that the results should be closer to the physical reality earlier as Δt is decreased. This observation is explained more thoroughly in the next section.

4.3 Dissipation and dispersion

It is often possible to say more about the numerical solution of a problem than just the assymptotic convergence order. Certain qualitative properties of the numerical solution can be quantified by the notions of numerical dissipation and dispersion, see [82].

An important fact, in this respect, is that the convolution quadrature of the time-domain boundary integral equation is equivalent to a boundary integral formulation of the semi-discretization of the underlying partial differential equation. Namely, the solution of the semi-discrete problem (32) and (33) satisfies the linear multistep, respectively Runge-Kutta discretization, of the underlying partial differential equation (1). For example in the case of viscoelastodynamics, see (4), the solution $\mathbf{u}_n$, $n = 0,1,\ldots,N$, of (32) will satisfy the semi-discrete PDE

$$c_{1v}^2(\partial_t^{\Delta t})\Delta\mathbf{u} - c_{2v}^2(\partial_t^{\Delta t})\nabla\times\nabla\times\mathbf{u} = (\partial_t^{\Delta t})^2\mathbf{u} \tag{36}$$

whereas in the case of the acoustic wave equation the solution u_n, $n = 0,1,\ldots,N$, of (32) satisfies the semi-discrete PDE

$$c^2\Delta u = (\partial_t^{\Delta t})^2 u \tag{37}$$

on the domain Ω. For the method of proof of this fact, see [53, Theorem 5.2] and the introduction of [10]. To perform dispersion and dissipation analysis, one assumes (36) or (37) to hold in the whole space $\mathbb{R}^3$ and investigates the shape of plane wave solutions. Such analysis is classical, but has first been performed in the context of convolution quadrature in [23].

For simplicity only the scalar wave equation (37) is investigated here. The non-discretized wave equation (3) admits plane-wave solutions of the form $e^{\mathrm{i}(\frac{\xi}{c}\cdot\mathbf{x}+\omega t)}$

with $\omega^2 = |\xi|^2$. Semi-discrete equation (37) also admits plane wave solutions $u_n(\mathbf{x}) = e^{\mathrm{i}(\frac{\xi}{c}.\mathbf{x}+\omega_{\Delta t}t_n)}$, but, the relationship between $\omega_{\Delta t}$ and ξ is considerably more involved and constitutes the dissipation and dispersion analysis. For linear multistep methods the relationship is given by

$$|\xi|^2 = \left(\frac{\gamma\left(e^{-\mathrm{i}\omega_{\Delta t}\Delta t}\right)}{\mathrm{i}\Delta t}\right)^2. \tag{38}$$

In the case of backward Euler discretization, i.e., $\gamma(\zeta) = 1-\zeta$, solving this equation for $\omega_{\Delta t}$ the following relationship is obtained

$$\omega_{\Delta t} = \pm|\xi| + \frac{\mathrm{i}}{2}\Delta t|\xi|^2 \mp \frac{1}{3}\Delta t^2|\xi|^3\cdots, \qquad \text{Backward Euler.}$$

This shows that plane waves satisfying the semi-discrete wave equation discretized by first order BDF method are of size $O(e^{-\frac{1}{2}\Delta t|\xi|^2})$, i.e., the solutions are significantly damped unless $\Delta t|\xi|^2 \ll 1$. This is a much stronger condition than the sampling condition of a few degrees of freedom per wavelength, i.e., $\Delta t|\xi| \ll 1$. In general, it is seen from (38) and the approximation property $\gamma(e^{-z}) = z + O(z^{p+1})$, that for a p-th order linear multistep based discretization to give an accurate result, Δt must satisfy the condition $\Delta t^p|\xi|^{p+1} \ll 1$. Since the order of A-stable multistep methods is restricted to $p \leq 2$, this condition on Δt is always significantly more stringent than the sampling condition.

For Runge-Kutta methods consider the plane wave $u_{n\ell} = e^{\mathrm{i}(\frac{\xi}{c}.\mathbf{x}+\omega_{\Delta t}t_n+\omega_{\ell,\Delta t}c_\ell\Delta t)}$. Since $c_m = 1$, $\omega_{m,\Delta t} = \omega_{\Delta t}$ must hold, but in general it is not possible to require $\omega_{\ell,\Delta t} = \omega_{\Delta t}$ for all ℓ. For the analysis the following result proved in [10] will be used

Lemma 1. *Let* (22) *hold,* $|\zeta| \neq 1$*, and* λ *be an eigenvalue of* $\Delta(\zeta)$*, but not of* A^{-1}*. Then* $R(\lambda) = \zeta^{-1}$*.*

A similar calculation as for the linear multistep methods gives the relationship

$$|\xi|^2\begin{pmatrix} u_{n1} \\ \vdots \\ u_{nm}\end{pmatrix} = \left(\frac{\Delta\left(e^{-\mathrm{i}\omega_{\Delta t}\Delta t}\right)}{\mathrm{i}\Delta t}\right)^2\begin{pmatrix} u_{n1} \\ \vdots \\ u_{nm}\end{pmatrix}. \tag{39}$$

A solution $\omega_{l,\Delta t}$ of the following equation also satisfies (39)

$$|\xi|\begin{pmatrix} u_{n1} \\ \vdots \\ u_{nm}\end{pmatrix} = \left(\frac{\Delta\left(e^{-\mathrm{i}\omega_{\Delta t}\Delta t}\right)}{\mathrm{i}\Delta t}\right)\begin{pmatrix} u_{n1} \\ \vdots \\ u_{nm}\end{pmatrix}.$$

Therefore, $\mathrm{i}\Delta t|\xi|$ is an eigenvalue of $\Delta\left(e^{-\mathrm{i}\omega_{\Delta t}\Delta t}\right)$ and for small enough $\Delta t|\xi|$ cannot be an eigenvalue of A^{-1}. Consequently, due to Lemma 1,

$$R(\mathrm{i}\Delta t|\xi|) = e^{\mathrm{i}\omega_{\Delta t}\Delta t}.$$

Recalling the approximation property of the stability function $R(z) = e^z + O(z^{p+1})$. it is seen that

$$\omega_{\Delta t} = |\xi| + |\xi| O(|\xi\Delta t|^p). \tag{40}$$

Since for the 2-stage Radau IIA method $p = 3$ and for the 3-stage method $p = 5$, it is seen from the last equation that these methods are significantly less dissipative and dispersive than the A-stable linear multistep formulas. Furthermore, the constant implicit in (40) is very favourable in the case of Radau IIA methods, it is $C = 1/216$ for the 2-stage and $C = 1/7200$ for the 3-stage method.

5 Space discretization

Space discretization, in the context of convolution quadratures, poses no extra difficulty compared to the space discretization of boundary integral operators of elliptic, in particular Helmholtz, problems. It is merely necessary to replace the Laplace domain integral operators in (32) by their discretized counterparts.

5.1 Galerkin and collocation in space

When using Galerkin discretization in space, finite element bases on boundaries Γ_D and Γ_N are used to construct the approximation spaces

$$X_D = \mathrm{Span}\{\varphi_1, \varphi_2, \ldots, \varphi_{M_1} \,|\, \varphi_j \equiv 0 \text{ on } \Gamma_N\},$$
$$X_N = \mathrm{Span}\{\psi_1, \psi_2, \ldots, \psi_{M_2} \,|\, \psi_j \equiv 0 \text{ on } \Gamma_D\}.$$

The unknowns $(\tilde{\mathbf{u}})_n$ and $(\tilde{\mathbf{t}})_n$ at time $t = t_n$ are approximated by a linear combination of functions in X_D and X_N:

$$(\tilde{\mathbf{u}}_h)_n = \sum_{\ell=1}^{M_1} \alpha_\ell^{(n)} \varphi_\ell \quad \text{and} \quad (\tilde{\mathbf{t}}_h)_n = \sum_{k=1}^{M_2} \beta_k^{(n)} \psi_k, \quad n = 0, 1, \ldots, N. \tag{41}$$

Inserting this ansatz into (32) and testing by functions from X_D and X_N gives the fully discrete system

$$\int_\Gamma V(\partial_t^{\Delta t})\tilde{\mathbf{t}}_h(\mathbf{x}, t_n)\psi_k(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{x} - \int_\Gamma K(\partial_t^{\Delta t})\tilde{\mathbf{u}}_h(\mathbf{x}, t_n)\psi_k(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{x} = \int_\Gamma \mathbf{f}_D(\mathbf{x}, t_n)\psi_k(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{x},$$
$$\int_\Gamma D(\partial_t^{\Delta t})\tilde{\mathbf{u}}_h(\mathbf{x}, t_n)\varphi_\ell(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{x} + \int_\Gamma K'(\partial_t^{\Delta t})\tilde{\mathbf{t}}_h(\mathbf{x}, t_n)\varphi_\ell(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{x} = \int_\Gamma \mathbf{f}_N(\mathbf{x}, t_n)\varphi_\ell(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{x},$$

for $n = 0, 1, \ldots, N$, $\ell = 1, 2, \ldots, M_1$, and $k = 1, 2, \ldots, M_2$.

When solving this convolutional linear system of equations using the techniques of Section 3.3, quadrature required to implement these equations can be done solely in Laplace domain. More specifically, the Galerkin discretization of operators $V(s_\ell)$, $K(s_\ell)$, $K'(s_\ell)$, and $D(s_\ell)$ for all the frequencies s_ℓ occurring in the algorithms described in Section 3.3 are needed. For example, Galerkin discretization of the single layer potential requires the computation of the following integrals

$$\int_\Gamma (V(s_\ell)\psi_j)(\mathbf{x})\psi_k(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{x} = \int_\Gamma \int_\Gamma \hat{\mathbf{U}}(\mathbf{x}-\mathbf{y}, s_\ell)\psi_j(\mathbf{y})\psi_k(\mathbf{x})\,\mathrm{d}\Gamma_\mathbf{y}\,\mathrm{d}\Gamma_\mathbf{x}.$$

Numerical quadrature routines for kernels $\hat{\mathbf{U}}(\mathbf{x}-\mathbf{y}, s_\ell)$ have been extensively investigated and are readily available, see for example [36, 45, 75]. In fact, one of the main advantages of convolution quadrature lies in the fact that numerical quadrature of the difficult/unknown distributional kernel function is not necessary.

It has to be mentioned that the right-hand sides $\mathbf{f}_D$ and $\mathbf{f}_N$ are not immediately available, but have to be first computed by applying time-domain integral operators to the data $\tilde{\mathbf{g}}_D$ and $\tilde{\mathbf{g}}_N$. This is usually done by first projecting the data onto boundary element bases defined on Γ; note that since it is not necessarily true that $\tilde{\mathbf{g}}_D \equiv 0$ on Γ_N and $\tilde{\mathbf{g}}_N \equiv 0$ on Γ_D it is not possible here to re-use spaces X_D and X_N.

To avoid double integration in space, it is of interest to use collocation in space instead of Galerkin discretization. Here the unknown functions are again approximated by a linear combination of basis functions as in (41) and this approximation is substituted in (32). To arrive at a system of linear equations, the resulting equations are evaluated at *collocation* points on the boundary.

Stability and convergence analysis of the fully discrete symmetric system has not yet appeared in literature in any of the applications covered in this paper. The linear multistep convolution quadrature with Galerkin discretization in space for the indirect boundary integral formulation of the Dirichlet problem of acoustics has been fully analysed in [53].

5.2 Fast data-sparse methods in frequency domain

Using Algorithm *SolveCQ* to solve the fully discrete system it is necessary to discretize operators $V(s_\ell)$, $K(s_\ell)$, $K'(s_\ell)$, and $D(s_\ell)$. Galerkin or collocation discretizations of such operators result in dense $M_j \times M_k$ matrices, $j,k = 1,2$. Therefore, direct computation and storage of such matrices has cost $O(M^2)$ with $M = \max(M_1, M_2)$. Fortunately, so called data sparse techniques have been developed in the past couple of decades that can in almost linear cost, i.e., $O(M \log^a M)$ for some $a > 0$, compute approximations of these matrices. Two main classes of such data sparse methods are hierarchical matrices ($\mathscr{H}$-matrices) [44, 43] and the fast multipole methods (FMM) [68, 24].

The difficulty of computing a data sparse representation of space discretizations of integral operators is directly related to the wavenumbers s_ℓ. The kernel functions have the form

$$\hat{\mathbf{U}}(\mathbf{x},s) = \sum_{i=1}^{w} \mathbf{A}^{(i)}(r,s) \frac{e^{-\frac{s}{c_i(s)}r}}{4\pi r}, \qquad r = |\mathbf{x}|,$$

with $c_i(s) \to const > 0$ for $|s| \to \infty$, and hence if $|\Im s_\ell| \gg 1$ the kernel is highly oscillatory and consequently difficult to discretize efficiently, on the other hand if $\Re s \gg 1$ the operator is practically diagonal and easy to efficiently discretize.

The evaluation of integral operators at different wavenumbers occurs in two places in Algorithm *SolveCQ*: in line 6. where a discrete convolutional system of size J is solved by solving a decoupled set of linear systems in Laplace domain, and in line 4. where a matrix-vector product with discretized integral operators in Laplace domain needs to be computed. In [10], it is shown that if J is chosen as a constant independent of Δt the frequencies arising in solving the small system 6 all satisfy $|\Im s|/\Re s \leq const$. This in turn implies that the integral operators in Laplace domain can be approximated by an $\mathscr{H}$-matrix with computational and storage complexity $O(M \log M)$. Furthermore, an (approximate) LU-decomposition in $\mathscr{H}$-matrix format can be computed in $O(N \log^2 N)$ time, which can be used as a very good preconditioner for solving the linear systems by an iterative method, such as GMRES.

Wavenumbers occurring in the update, line 4., can have $|\Im s| \sim \Delta t^{-1}$. If $\Delta t / c_i$, with c_i the speed of the wave, is much smaller than the size of the computational domain Ω, high-frequency problems occur for which $\mathscr{H}$-matrices lose their efficiency [58]. Fortunately, the highly-oscillatory operators need not be inverted, but only a single matrix-vector product needs to be computed. This is an ideal task for the so called fast multipole methods. Here the advantage of the recursive procedure from Section 3.3.2 can best be seen.

Many fast-multipole like methods for high-frequency Helmholtz integral operators have been developed since the early 1990s [4, 11, 29, 69, 70]. These have dealt with cases of purely real and purely imaginary wavenumbers. They can be adapted to the present case of the whole range of complex frequencies, still, to do this optimally more work is needed.

6 Numerical example

In this section, the solution procedure of section 3.3.1 is tested for elastodynamics with different Runge-Kutta and multistep methods. In order to show the validity of the results only benchmark examples, whose analytical solutions are known, are treated. All computations were performed by using the HyENA C++ library for the numerical solution of partial differential equations using the boundary element method [59]. For the Fourier like transformations the FFTW routines [38] are taken.

A 3-d rod of size $\ell_1 = 3.0\,\mathrm{m}$ and $\ell_2 = \ell_3 = 1.0\,\mathrm{m}$, as depicted in Figure 2, is considered. It is fixed on one end and the other end is excited by a pressure jump $t_1 = -1.0H(t)\,\mathrm{N/m^2}$. $H(t)$ denotes the unit step function. The material parameters of steel ($\rho = 7850\,\mathrm{kg/m^3}$, $G = 1.055 \times 10^{11}\,\mathrm{N/m^2}$, $K = 7.03 \times 10^{10}\,\mathrm{N/m^2}$) are taken. Poisson ratio is chosen to be zero, such that the results can be compared with the analytical solution of longitudinal waves in a 1-d elastodynamic rod (see [41]). The

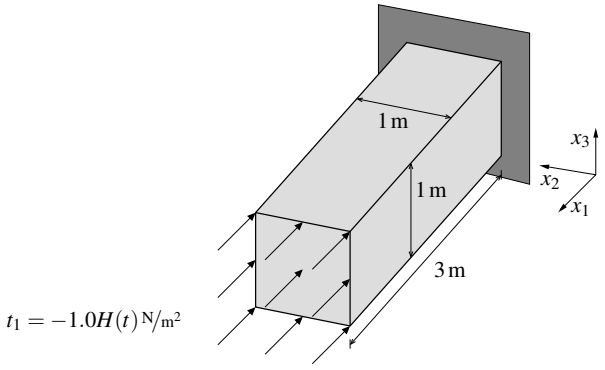


Fig. 2: System and boundary conditions

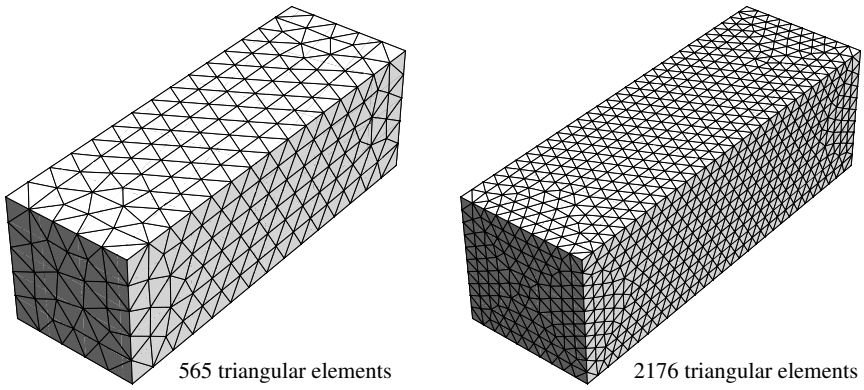
rod shown in Figure 2 is discretised with two different meshes, the coarse with 565 triangular boundary elements of uniform mesh size $h = 0.2\,\mathrm{m}$ and the fine with 2176 triangular boundary elements of uniform mesh size $h = 0.1\,\mathrm{m}$. Both are depicted in Figure 3. The displacements and tractions are approximated by piecewise constant and continuous linear polynomials, respectively. In order to compare different time discretizations the dimensionless value

$$\beta = \frac{c_1 \Delta t}{h}$$

is introduced. This value depends on the velocity of the compression wave c_1, the time step size Δt, and the average mesh size h. For the Runge-Kutta methods the time step size Δt is taken that of the stages and not of one step to have a fair comparison with the multistep method.

In the following, results are presented to show the influence of the different time discretisations, i.e., the chosen methods are BDF2, Radau IIA (2-stage), and Radau IIA (3-stage). It is studied how these different methods work in relation to the spatial discretisation and the time step size.

First, the displacement in the middle of the top and the tractions in the middle of the bottom of the bar are displayed in Figure 4 versus time for the different Runge-Kutta methods listed above and the BDF2. A collocation technique with $\beta = 0.3$ and the fine mesh is used. The displacement results are more or less equal and coincide well with the analytical solution. The traction solution is overall good as well. The differences between the Runge-Kutta methods and the BDF2 are visible in the



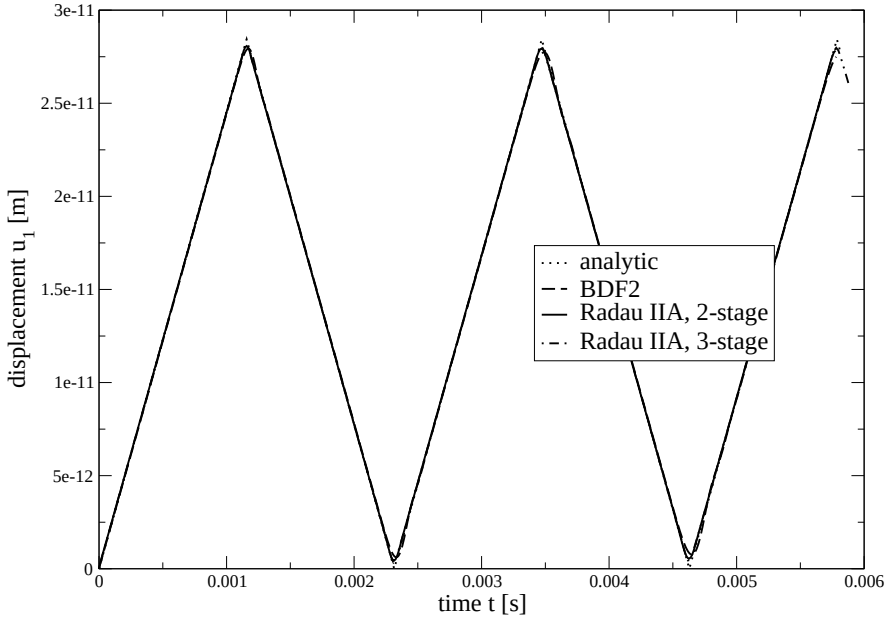
(a) coarse mesh: $h = 0.2m$ (b) fine mesh: $h = 0.1m$

Fig. 3: Uniform meshes used for the calculations

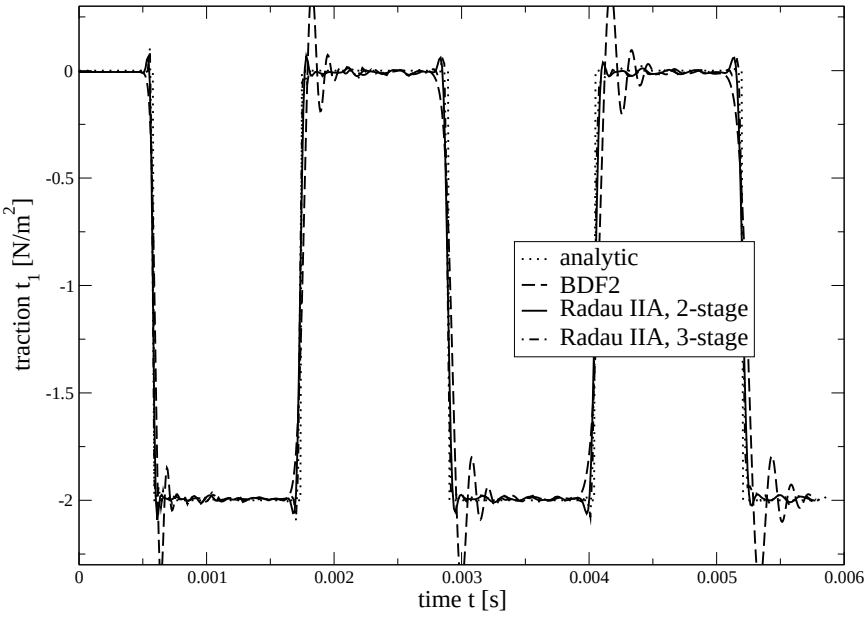
oscillations at the jumps. There, the Runge-Kutta methods show less pronounced effects and as well a better representation of the straight lines. This is in accordance with the observations made for the integration weights in section 4 (see Figure 1). The Runge-Kutta methods have represented the wave fronts much sharper than the BDF2. Hence, here the oscillations must be smaller. Nevertheless, also the results for the BDF2 are good. As the different displacement results are nearly not distinguishable, in the following only traction results will be presented.

The next study shows the influence of the mesh size where the traction results using a Radau IIA (2-stage) are compared. In Figure 5, the results are displayed versus time for both discretisations of Figure 3 and for a collocation (denoted by 'collo') and a symmetric Galerkin BEM (denoted by 'SGBEM'). As expected the finer mesh yields better results. The difference between collocation and the SGBEM is not observable. Similar plots can be made with the other time discretisations, which yield qualitatively the same. One difference can be observed. The 3-stage Radau IIA method tends to instabilities for the chosen $\beta = 0.3$.

The sensitivity on the times step size is studied in Figure 6. The traction results are computed with the finer mesh for all three multistep methods for different β-values. For $\beta = 0.1$ the 3-stage Radau IIA method shows clearly an instability. These results are truncated after $t \approx 0.0033\,\mathrm{s}$, not to destroy the whole picture. With a coarser mesh also the other methods would show instabilities. Overall, the numerical tests confirm that a finer mesh moves the instabilities to smaller values of β. Comparing to the mathematics in section 3 this behavior is not obvious. But, it must be remarked that all proofs require some smoothness of the given data which is in the example by the Heaviside function clearly violated. However, for engineering



(a) Displacements at the top



(b) Tractions at the bottom

Fig. 4: Results for different Runge-Kutta methods and the BDF2 versus time

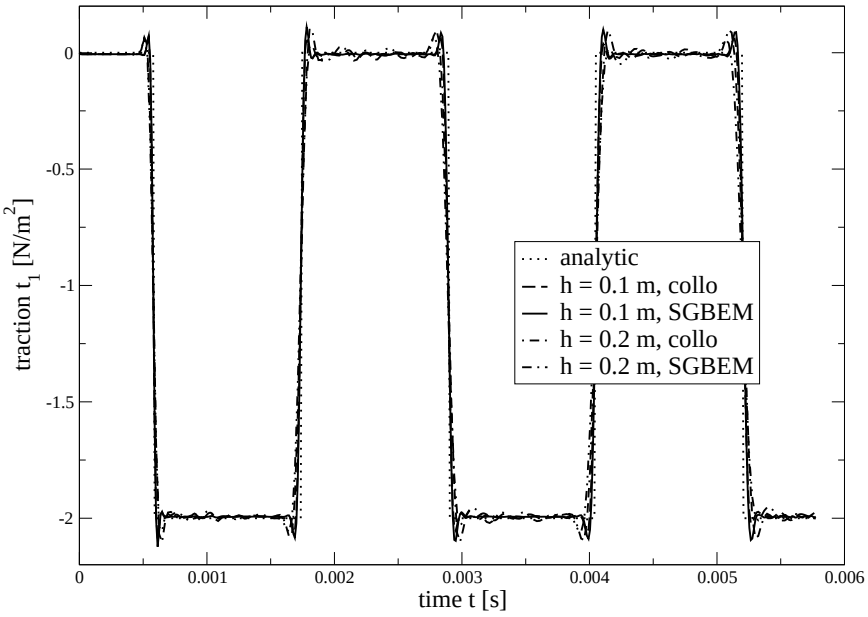


Fig. 5: Influence of mesh size using a Radau IIA (2-stage) method

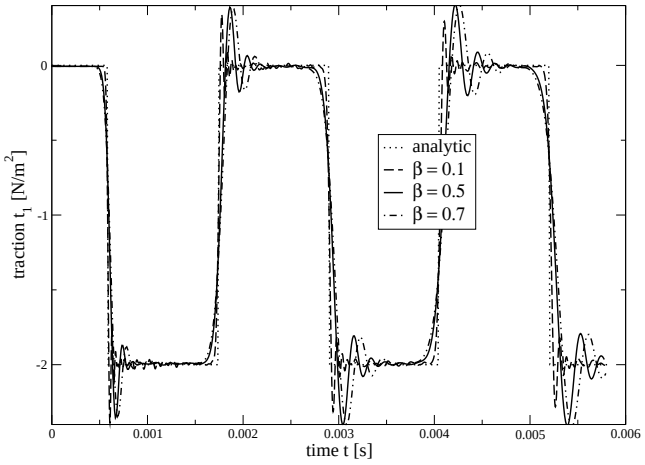
applications such loadings are necessary and, therefore, the numerical tests has been made with this right hand side.

The last study concerns the long time behavior, because a lot of time domain BE formulations suffer from either strong numerical damping or instabilities in the long time range. The proposed method shows a very nice behavior as presented in Figure 7. The collocation and the SGBEM results are given for both meshes using a 3-stage Radau IIA method. Nearly no numerical damping is observed and no instabilities. The time step size is chosen according to $\beta = 0.5$. The other Runge-Kutta or multistep methods produce comparable results. Hence, it can be concluded that the long time behavior is satisfactory.

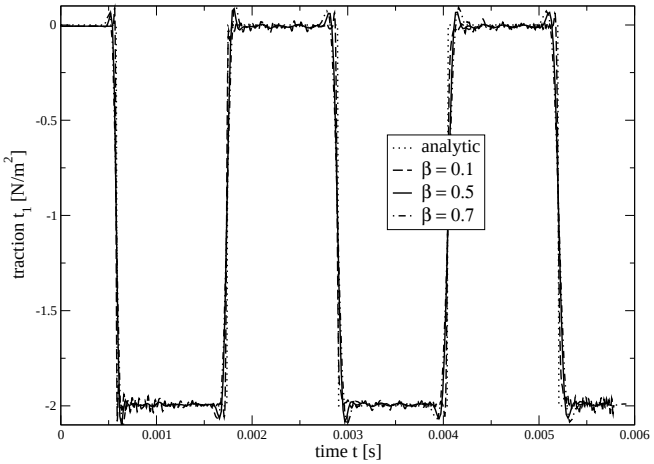
Overall, the presented results show that the method is robust with respect to the time and the spatial discretisation if the mesh is sufficiently fine and the time step size not too small.

Appendix

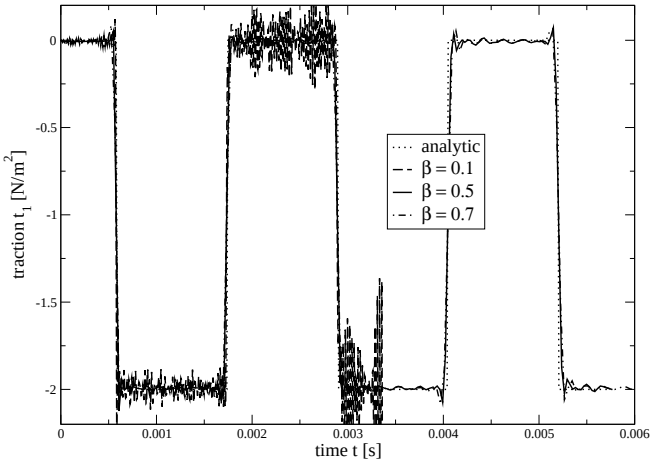
The general form of the fundamental solutions for the operators given in section 2 can be found in (10). For better readability it is recalled



(a) BDF2



(b) Radau IIA (2-stage)



(c) Radau IIA (3-stage)

Fig. 6: Influence of time step size for the different multistep methods

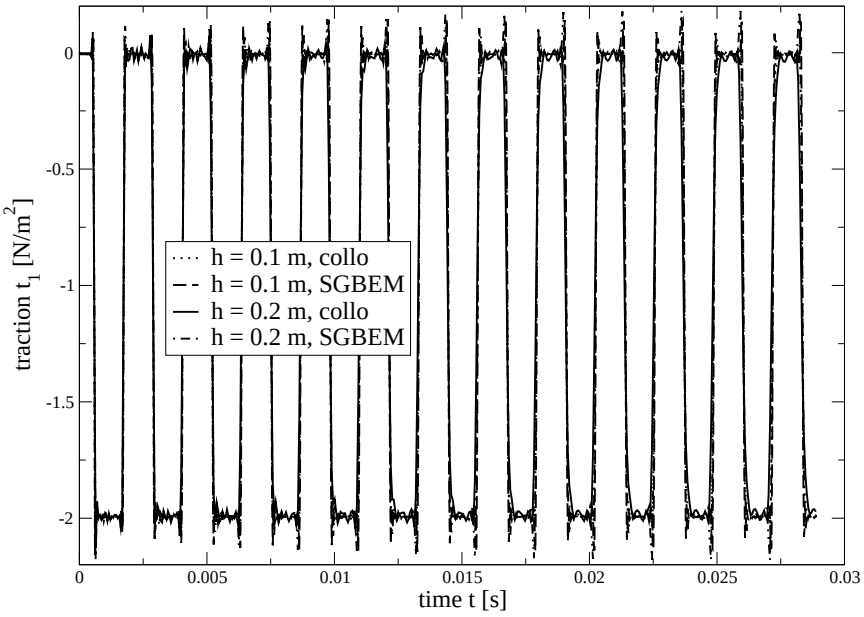


Fig. 7: Long time behavior using a Radau IIA (3-stage) method

$$\hat{\mathbf{U}}(\mathbf{x}-\mathbf{y},s) = \sum_{i=1}^{w} \mathbf{A}^{(i)}(r,s)\,\frac{e^{-\lambda_i r}}{4\pi r} \qquad \text{with} \quad r = |\mathbf{x}-\mathbf{y}| \,.$$

In the following, the coefficients $\mathbf{A}_i(r,s)$ are listed. For the vectorial problems the fundamental solutions are tensors. For them the indical notation is used with the notation $r_{,i} = \frac{x_i - y_i}{r}$ for the directional derivative and δ_{ij} for the Kronecker delta.

Acoustics

The respective equations are presented in section 2.1.1. In (3), the homogeneous form of the differential equation is given. For the definition of the fundamental solution a source of Dirac type has to be added. As in acoustics only one compressional wave appears and the sum in (10) has only one term, i.e., $w = 1$ holds. Further, it is a scalar problem, hence, the tensor of fundamental solutions degenerates to a scalar value. The coefficient is

$$A^{(1)} = 1 \qquad \text{with} \quad \lambda_1 = \frac{s}{c} = s\sqrt{\frac{\rho}{K}} \,.$$

Visco- and elastodynamics

The governing equations for viscoelasticity are given in section 2.1.2 as an extension of the elastodynamic case (1). Only the wave velocities have to be replaced by (5). The excitation in the definition of the fundamental solutions is a force of Dirac type. Two waves, the compression and the shear wave, exist and, therefore, the sum in (10) has two terms, i.e., $w = 2$ holds. The coefficients are

$$A_{ij}^{(1)} = \frac{1}{\rho s^2}\left\{\frac{3r_{,i}r_{,j}-\delta_{ij}}{r^2}\left(\lambda_1 r+1\right)+\lambda_1^2 r_{,i}r_{,j}\right\}$$
$$A_{ij}^{(2)} = \frac{1}{\rho s^2}\left\{\frac{3r_{,i}r_{,j}-\delta_{ij}}{r^2}\left(\lambda_2 r+1\right)+\lambda_2^2 r_{,i}r_{,j}\right\}$$

with the complex wave numbers

$$\lambda_1 = \frac{s}{c_1} \qquad \lambda_2 = \frac{s}{c_2} \qquad \text{in elastodynamics and}$$
$$\lambda_1 = \frac{s}{c_{1v}} \qquad \lambda_2 = \frac{s}{c_{2v}} \qquad \text{in viscoelastodynamics.}$$

Poroelastodynamics

The governing equations of proelastodynamics (7) is a coupled set of differential equations for the unkowns solid displacement $\mathbf{u}$ and pore pressure p. Consequently, the fundamental solution is a matrix

$$\hat{\mathbf{G}} = \begin{pmatrix} \hat{U}_{ij}^s & \hat{U}_i^f \\ \hat{P}_j^s & \hat{P}^f \end{pmatrix} \qquad \text{with} \quad \hat{U}_i^f = s\hat{P}_i^s \ .$$

The single entries are composed as given in (10) and have either three waves, i.e., $w = 3$ or only two compressional waves, i.e., $w = 2$. The respective coefficients of the sum are for the solid displacements due to a bulk body forc of Dirac type in the solid, i.e., $\hat{U}_{ij}^s$

$$A_{ij}^{(1)} = \frac{1}{\left(\rho-\beta(s)\rho_f\right)s^2} R_1 \frac{\lambda_4^2-\lambda_2^2}{\lambda_1^2-\lambda_2^2}$$
$$A_{ij}^{(2)} = \frac{-1}{\left(\rho-\beta(s)\rho_f\right)s^2} R_2 \frac{\lambda_4^2-\lambda_1^2}{\lambda_1^2-\lambda_2^2}$$
$$A_{ij}^{(3)} = \frac{1}{\left(\rho-\beta(s)\rho_f\right)s^2}\left(\delta_{ij}\lambda_3^2 - R_3\right)$$

with $R_k = \left(3r_{,i}r_{,j}-\delta_{ij}\right)/r^2 + \lambda_k\left(3r_{,i}r_{,j}-\delta_{ij}\right)/r + \lambda_k^2 r_{,i}r_{,j}$ and
$\lambda_4^2 = s^2\left(\rho-\beta(s)\rho_f\right)/\left(K+4/3G\right)$. The pressure caused by the same load is, i.e., $\hat{P}_j^s$

$$A_{ij}^{(1)} = \frac{(\alpha-\beta(s))\,s\rho_f r_{,j}}{\beta(s)\left(K+\frac{4}{3}G\right)\left(\lambda_1^2-\lambda_2^2\right)}\left(\lambda_1+\frac{1}{r}\right)$$
$$A_{ij}^{(2)} = \frac{-(\alpha-\beta(s))\,s\rho_f r_{,j}}{\beta(s)\left(K+\frac{4}{3}G\right)\left(\lambda_1^2-\lambda_2^2\right)}\left(\lambda_2+\frac{1}{r}\right)\,.$$

The remaining one is the pressure due to a source of Dirac type in the fluid, i.e., $\hat{P}^f$

$$A_{ij}^{(1)} = \frac{s\rho_f}{\beta(s)}\frac{\lambda_1^2-\lambda_4^2}{\lambda_1^2-\lambda_2^2} \qquad A_{ij}^{(2)} = \frac{-s\rho_f}{\beta(s)}\frac{\lambda_2^2-\lambda_4^2}{\lambda_1^2-\lambda_2^2}\,.$$

Electromagnetism

The fundamental solution is the same as for the acoustic wave equation, i.e.,

$$A^{(1)} = 1 \qquad \text{with} \quad \lambda_1 = \frac{s}{c} = s\sqrt{\varepsilon\mu}\,.$$

References

[1] Abreu AI, Carrer JAM, Mansur WJ (2003) Scalar wave propagation in 2D: a BEM formulation based on the operational quadrature method. Eng Anal Bound Elem 27:101–105

[2] Abreu AI, Mansur WJ, Carrer JAM (2006) Initial conditions contribution in a BEM formulation based on the convolution quadrature method. Int J Numer Methods Engrg 67:417–434

[3] Aimi A, Diligenti M (2008) A new space-time energetic formulation for wave propagation analysis in layered media by BEMs. Int J Numer Methods Engrg 75(9):1102–1132

[4] Amini S, Profit A (2003) Multi-level fast multipole solution of the scattering problem. Engineering Analysis with Boundary Elements 27(5):547–564

[5] Antes H (1985) A boundary element procedure for transient wave propagations in two-dimensional isotropic elastic media. Finite Elements in Analysis and Design 1:313–322

[6] Antes H, Jäger M (1995) On stability and efficiency of 3d acoustic BE procedures for moving noise sources. In: Atluri S, Yagawa G, Cruse T (eds) Computational Mechanics, Theory and Applications, Springer-Verlag, Heidelberg, vol 2, pp 3056–3061

[7] Bachelot A, Lange V (1995) Time dependent integral method for Maxwell's system. In: Mathematical and numerical aspects of wave propagation (Mandelieu-La Napoule, 1995), SIAM, Philadelphia, PA, pp 151–159

[8] Bachelot A, Bounhoure L, Pujols A (2001) Couplage éléments finis–potentiels retardés pour la diffraction électromagnétique par un obstacle hétérogène. Numer Math 89(2):257–306
[9] Bamberger A, Ha-Duong T (1986) Formulation variationelle espace-temps pour le calcul par potentiel retardé d'une onde acoustique. Math Meth Appl Sci 8:405–435 and 598–608
[10] Banjai L (2009) Multistep and multistage convolution quadrature for the wave equation: Algorithms and experiments. to appear in SIAM J Sci Comput
[11] Banjai L, Hackbusch W (2008) Hierarchical matrix techniques for low and high frequency Helmholtz equation. IMA J Numer Anal 28(1):46–79
[12] Banjai L, Lubich C (2010) An error analysis of Runge-Kutta convolution quadrature. submitted
[13] Banjai L, Sauter S (2009) Rapid solution of the wave equation in unbounded domains. SIAM J Numer Anal 47(1):227–249
[14] Banjai L, Lubich C, Melenk JM (2010) submitted
[15] Beskos DE (1987) Boundary element methods in dynamic analysis. AMR 40(1):1–23
[16] Beskos DE (1997) Boundary element methods in dynamic analysis: Part II (1986-1996). AMR 50(3):149–197
[17] Biot MA (1956) Theory of propagation of elastic waves in a fluid-saturated porous solid.I. Low-frequency range. J Acoust Soc Am 28(2):168–178
[18] Biot MA (1956) Theory of propagation of elastic waves in a fluid-saturated porous solid.II. Higher frequency range. J Acoust Soc Am 28(2):179–191
[19] Birgisson B, Siebrits E, Peirce AP (1999) Elastodynamic direct boundary element methods with enhanced numerical stability properties. Int J Numer Methods Engrg 46:871–888
[20] de Boer R (2000) Theory of Porous Media. Springer-Verlag, Berlin
[21] Butcher JC (1987) The numerical analysis of ordinary differential equations. A Wiley-Interscience Publication, John Wiley & Sons Ltd., Chichester, runge-Kutta and general linear methods
[22] Calvo MP, Cuesta E, Palencia C (2007) Runge-Kutta convolution quadrature methods for well-posed equations with memory. Numer Math 107(4):589–614
[23] Chen1 Q, Monk P, Wang X, Weile D (2010) Analysis of convolution quadrature applied to the time-domain electric field integral equation. submitted
[24] Cheng H, Crutchfield WY, Gimbutas Z, Greengard LF, Ethridge JF, Huang J, Rokhlin V, Yarvin N, Zhao J (2006) A wideband fast multipole method for the Helmholtz equation in three dimensions. J Comput Phys 216(1):300–325
[25] Chew WC, Jin JM, Michielssen E, Song JM (2001) Fast and Efficient Algorithms in Computational Electromagnetics. Artech House, Boston, London
[26] Christensen RM (1971) Theory of Viscoelasticity. Academic Press, New York
[27] Costabel M (2005) Time-dependent problems with the boundary integral equation method. In: Stein E, de Borst R, Hughes TJR (eds) Encyclopedia of Computational Mechanics, vol 1, Fundamentals, John Wiley & Sons, New York, Chichester, Weinheim, chap 25

[28] Cruse TA, Rizzo FJ (1968) A direct formulation and numerical solution of the general transient elastodynamic problem, I. Aust J Math Anal Appl 22(1):244–259
[29] Darve E, Havé P (2004) Efficient fast multipole method for low-frequency scattering. J Comput Phys 197(1):341–363
[30] Davies P (1997) Averaging techniques for time marching schemes for retarded potential integral equations. Appl Numer Math 23:291–310
[31] Davies P, Duncan D (2004) Stability and convergence of collocation schemes for retarded potential integral equations. SIAM J Numer Anal 42(3):1167–1188
[32] Davies PJ (1996) On the stability of time-marching schemes for the general surface electric-field integral equation. IEEE Trans Antennas and Propagation 44(11):1467–1473
[33] Davies PJ (1998) A stability analysis of a time marching scheme for the general surface electric field integral equation. Appl Numer Math 27(1):33–57
[34] Davies PJ, Duncan DB, Zubik-Kowal B (2005) The stability of numerical approximations of the time domain current induced on thin wire and strip antennas. Appl Numer Math 55(1):48–68
[35] Domínguez J (1978) Dynamic stiffness of rectangular foundations. Report no. R78-20, Department of Civil Engineering, MIT, Cambridge MA
[36] Erichsen S, Sauter SA (1998) Efficient automatic quadrature in 3-d Galerkin BEM. Comput Methods Appl Mech Engrg 157(3-4):215–224, seventh Conference on Numerical Methods and Computational Mechanics in Science and Engineering (NMCM 96) (Miskolc)
[37] Friedman MB, Shaw R (1962) Diffraction of pulses by cylindrical obstacles of arbitrary cross section. J of Appl Mech 29(1):40–46
[38] Frigo M, Johnson SG (2005) The design and implementation of FFTW3. Proceedings of the IEEE 93(2):216–231, special issue on "Program Generation, Optimization, and Platform Adaptation"
[39] García-Sánchez F, Zhang C, Sáez A (2008) 2-d transient dynamic analysis of cracked piezoelectric solids by a time-domain BEM. Comput Methods Appl Mech Engrg 197(33-40):3108–3121
[40] Gaul L, Schanz M (1999) A comparative study of three boundary element approaches to calculate the transient response of viscoelastic solids with unbounded domains. Comput Methods Appl Mech Engrg 179(1-2):111–123
[41] Graff KF (1975) Wave Motion in Elastic Solids. Oxford University Press
[42] Gurtin ME, Sternberg E (1962) On the Linear Theory of Viscoelasticity. Arch Rational Mech Anal 11:291–356
[43] Hackbusch W (2009) Hierarchische Matrizen. Springer Berlin Heidelberg, algorithmen und Analysis
[44] Hackbusch W, Nowak ZP (1989) On the fast matrix multiplication in the boundary element method by panel clustering. Numer Math 54(4):463–491
[45] Hackbusch W, Sauter S (1993) On the Efficient Use of the Galerkin Method to Solve Fredholm Integral Equations. Applications of Mathematics 38(4-5):301–322

[46] Hackbusch W, Kress W, Sauter SA (2007) Sparse convolution quadrature for time domain boundary integral formulations of the wave equation by cutoff and panel-clustering. In: Schanz M, Steinbach O (eds) Boundary Element Analysis: Mathematical Aspects and Applications, Lecture Notes in Applied and Computational Mechanics, vol 29, Springer-Verlag, Berlin Heidelberg, pp 113–134

[47] Hairer E, Wanner G (1996) Solving ordinary differential equations. II, Springer Series in Computational Mathematics, vol 14, 2nd edn. Springer-Verlag, Berlin, stiff and differential-algebraic problems

[48] Hairer E, Lubich C, Schlichte M (1985) Fast numerical solution of nonlinear Volterra convolution equations. SIAM J Sci Stat Comput 6(3):532–541

[49] Hairer E, Nørsett SP, Wanner G (1993) Solving ordinary differential equations. I, Springer Series in Computational Mathematics, vol 8, 2nd edn. Springer-Verlag, Berlin, nonstiff problems

[50] Laliena AR, Sayas FJ (2009) Theoretical aspects of the application of convolution quadrature to scattering of acoustic waves. Numer Math 112(4):637–678

[51] Lubich C (1988) Convolution quadrature and discretized operational calculus. I. Numer Math 52(2):129–145

[52] Lubich C (1988) Convolution quadrature and discretized operational calculus. II. Numer Math 52(4):413–425

[53] Lubich C (1994) On the multistep time discretization of linear initial-boundary value problems and their boundary integral equations. Numer Math 67:365–389

[54] Lubich C (2004) Convolution quadrature revisited. BIT Num Math 44(3):503–514

[55] Lubich C, Schneider R (1992) Time discretization of parabolic boundary integral equations. Numer Math 63:455–481

[56] Mansur WJ (1983) A time-stepping technique to solve wave propagation problems using the boundary element method. Phd thesis, University of Southampton

[57] Mansur WJ, Carrer JAM, Siqueira EFN (1998) Time discontinuous linear traction approximation in time-domain BEM scalar wave propagation. Int J Numer Methods Engrg 42(4):667–683

[58] Messner M, Schanz M (2010) An accelerated symmetric time-domain boundary element formulation for elasticity. Eng Anal Bound Elem DOI 10.1016/j.enganabound.2010.06.007

[59] Messner M, Messner M, Rammerstorfer F, Urthaler P (2010) Hyperbolic and elliptic numerical analysis BEM library (HyENA). http://www.mech.tugraz.at/HyENA, [Online; accessed 22-January-2010]

[60] Monk P (2003) Finite element methods for Maxwell's equations. Numerical Mathematics and Scientific Computation, Oxford University Press, New York

[61] Morse PM, Ingard KU (1986) Theoretical Acoustics. Princeton University Press

[62] Nardini D, Brebbia CA (1982) A new approach to free vibration analysis using boundary elements. In: Brebbia CA (ed) Boundary Element Methods, Springer-Verlag, Berlin, pp 312–326
[63] Otani Y, Takahashi T, Nishimura N (2007) A fast boundary integral equation method for elastodynamics in time domain and its parallelisation. In: Schanz M, Steinbach O (eds) Boundary Element Analysis: Mathematical Aspects and Applications, Lecture Notes in Applied and Computational Mechanics, vol 29, Springer-Verlag, Berlin Heidelberg, pp 161–185
[64] Partridge PW, Brebbia CA, Wrobel LC (1992) The Dual Reciprocity Boundary Element Method. Computational Mechanics Publication, Southampton
[65] Peirce A, Siebrits E (1997) Stability analysis and design of time-stepping schemes for general elastodynamic boundary element models. Int J Numer Methods Engrg 40(2):319–342
[66] Pujols A (1991) Time dependent integral method for Maxwell equations. In: Mathematical and numerical aspects of wave propagation phenomena (Strasbourg, 1991), SIAM, Philadelphia, PA, pp 118–126
[67] Rizos DC, Karabalis DL (1994) An advanced direct time domain BEM formulation for general 3-D elastodynamic problems. Comput Mech 15:249–269
[68] Rokhlin V (1985) Rapid solution of integral equations of classical potential theory. J Comput Phys 60(2):187–207
[69] Rokhlin V (1990) Rapid solution of integral equations of scattering theory in two dimensions. J Comput Phys 86(2):414–439
[70] Rokhlin V (1993) Diagonal forms of translation operators for the Helmholtz equation in three dimensions. Appl Comput Harmon Anal 1(1):82–93
[71] Schanz M (2001) Application of 3-d Boundary Element formulation to wave propagation in poroelastic solids. Eng Anal Bound Elem 25(4-5):363–376
[72] Schanz M (2001) Wave Propagation in Viscoelastic and Poroelastic Continua: A Boundary Element Approach, Lecture Notes in Applied Mechanics, vol 2. Springer-Verlag, Berlin, Heidelberg, New York
[73] Schanz M, Antes H (1997) Application of 'operational quadrature methods' in time domain boundary element methods. Meccanica 32(3):179–186
[74] Schanz M, Antes H (1997) A new visco- and elastodynamic time domain boundary element formulation. Comput Mech 20(5):452–459
[75] Schwab C, Wendland WL (1992) On numerical cubatures of singular surface integrals in boundary element methods. Numer Math 62(3):343–369
[76] Shanker B, Ergin AA, Aygün K, Michielssen E (2000) Analysis of transient electromagnetic scattering from closed surfaces using a combined field integral equation. IEEE Trans Antennas and Propagation 48(7):1064–1074
[77] Shanker B, Ergin AA, Lu MY, Michielssen E (2003) Fast analysis of transient electromagnetic scattering phenomena using the multilevel plane wave time domain algorithm. IEEE Trans Antennas and Propagation 51:628–641
[78] Stein E, de Borst R, Hughes TJR (eds) (2004) Encyclopedia of computational mechanics. Vol. 1. John Wiley & Sons Ltd., Chichester, fundamentals
[79] Steinbach O (2008) Numerical Approximation Methods for Elliptic Boundary Value Problems, Texts in Applied Mathematics, vol 54. Springer

[80] Takahashi T, Nishimura N, Kobayashi S (2004) A fast BIEM for three-dimensional elastodynamics in time domain. Eng Anal Bound Elem 28:165–80, erratum in EABEM, 28, 165–180, 2004

[81] Terrasse I (1993) Résolution mathmatique et numérique des équations de Maxwell instationnaires par une méthode de potentiels retardés. PhD thesis, Ecole polytechnique

[82] Trefethen LN (1996) Finite difference and spectral methods for ordinary and partial differential equations. Unpublished text available at http://www.comlab.ox.ac.uk/nick.trefethen/pdetext.html

[83] Wang X, Wildman RA, Weile DS, Monk P (2008) A finite difference delay modeling approach to the discretization of the time domain integral equations of electromagnetics. IEEE Trans Antennas and Propagation 56(8, part 1):2442–2452

[84] Wheeler LT, Sternberg E (1968) Some theorems in classical elastodynamics. Arch Rational Mech Anal 31:51–90

[85] Wildman RA, Pisharody G, Weile DS, Balasubramaniam S, E M (2004) An accurate scheme for the solution of the time-domain integral equations of electromagnetics using higher order vector bases and bandlimited extrapolation. IEEE Trans Antennas and Propagation 52:2973–2984

[86] Wilmanski K (1998) A thermodynamic model of compressible porous materials with the balance equation of porosity. Transp Porous Media 32(1):21–47

[87] Yu G, Mansur WJ, Carrer JAM, Gong L (1998) Time weighting in time domain BEM. Eng Anal Bound Elem 22(3):175–181

[88] Yu G, Mansur WJ, Carrer JAM, Gong L (2000) Stability of Galerkin and Collocation time domain boundary element methods as applied to the scalar wave equation. Comput & Structures 74(4):495–506

[89] Zhang C (2000) Transient elastodynamic antiplane crack analysis in anisotropic solids. Internat J Solids Structures 37(42):6107–6130