Goal-oriented space-time adaptivity for transient dynamics using a modal description of the adjoint solution

Francesc Verdugo · Núria Parés · Pedro Díez

Abstract This article presents a space-time adaptive strategy for transient elastodynamics. The method aims at computing an optimal space-time discretization such that the computed solution has an error in the quantity of interest below a user-defined tolerance. The methodology is based on a goal-oriented error estimate that requires accounting for an auxiliary adjoint problem. The major novelty of this paper is using modal analysis to obtain a proper approximation of the adjoint solution. The idea of using a modal-based description was introduced in a previous work for error estimation purposes. Here this approach is used for the first time in the context of adaptivity. With respect to the standard direct time-integration methods, the modal solution of the adjoint problem is highly competitive in terms of computational effort and memory requirements. The performance of the proposed strategy is tested in two numerical examples. The two examples are selected to be representative of different wave propagation phenomena, one being a 2D bulky continuum and the second a 2D domain representing a structural frame.

F. Verdugo
Laboratori de Càlcul Numèric (LaCàN),
Universitat Politècnica de Catalunya (UPC),
Jordi Girona 1-3 E-08034 Barcelona, Spain.
E-mail: francesc.verdugo@upc.edu

N. Parés
Laboratori de Càlcul Numèric (LaCàN),
Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona (EUETIB),
Compte d'Urgell, 187, E-08036, Barcelona, Spain.
E-mail: nuria.pares@upc.edu

P. Díez (corresponding author)
Laboratori de Càlcul Numèric (LaCàN),
Universitat Politècnica de Catalunya (UPC),
Jordi Girona 1-3 E-08034 Barcelona, Spain.
and Centre Internacional de Mètodes Numèrics en Enginyeria (CIMNE), Gran Capitán s/n, E-08034 Barcelona, Spain.
Tel.: (+34) 934017240
Fax.: (+34) 934011825
E-mail: pedro.diez@upc.edu

1 Introduction

Computing high fidelity numerical approximations requires a fine discretization and leads to a large consumption of computational resources. Adaptivity aims at providing the optimal discretization (space mesh and time grid) guaranteeing some user-prescribed accuracy at a minimum computational cost. Many adaptive techniques have been developed with application to different problem types. These tools are particularly important in wave propagation problems, e.g. linear elastodynamics, because the features of the solution concentrate at the wave fronts and therefore a fine mesh is only required at specific regions of the domain.

Over the last three decades, a vast literature has been produced on adaptivity. Among the the pioneering works, references [27,18] propose adaptive techniques for flow problems using curvature and gradient based error indicators. This type of heuristic error indicators are used to identify the parts of the solution requiring a finer mesh size. This approach is applicable to many problem types because error indicators do not rely on the problem properties, but in the geometrical features of the solution. This type of indicators detect properly the errors associated with interpolation but fail in capturing the error from other sources, e.g. pollution error.

A more reliable alternative to drive mesh adaptivity are a posteriori error estimators. They are used to effi-

ciently control the accuracy of some output of the solution by means of refining the discretization only where is needed (in the zones where the error is emanating from). The available outputs for assessing the accuracy of the approximation are global norms, e.g. the energy or L^2 norm [1,15,32], or *quantities of interest* [20,21,7, 28]. Error estimators considering quantities of interest are referred as *goal-oriented*.

Goal-oriented adaptivity is discussed in the literature for many problem types. For instance, for elliptic problems [20–22,29,19], for the convection-diffusion-reaction equation [25,26], for non-linear structural problems [16,17], for time-dependent parabolic problems [23, 24,5] and for elastodynamics (or other 2nd order hyperbolic problems) [2–4,11].

Goal-oriented adaptivity for elastodynamics is a very challenging topic and it is still ongoing research. The main difficulties are 1) solving the associated space-time adjoint solution accurately to estimate the error in the quantity of interest, 2) splitting the contributions of the space and time discretization errors and 3) transferring the solution from one mesh to another without loss of accuracy.

References [2,3,11] are among the few discussing goal-oriented adaptivity in elastodynamics. The input of the adaptive procedure is a desired error tolerance in some quantity of interest. The adjoint solution is computed with the same time-integration method as the original solution. This approach might be memory demanding because at least the original or the adjoint solution has to be stored as a whole (at each mesh node and time point) prior to evaluate the error estimate.

The adaptive strategy presented in this article is an alternative to the previous approach. Here, the adjoint problem is approximated using modal analysis, as suggested in reference [30], to preclude the costly adjoint approximation and storage. The modal-based adjoint approximation is particularly efficient for some quantities of interest. This is because the adjoint solution is stored for a few vibration modes instead that for all time steps. Moreover, the time description of the adjoint solution is known analytically once the vibration frequencies and modes are available. This simplifies the algorithmic complexity of the adaptive procedure.

The remainder of this paper is organized as follows. Section 2 presents the equations of elastodynamics. Section 3 presents the weak and discrete versions of the problem using the double field time-continuous Galerkin method. The modal-based error assessment approach is presented in section 4. Section 5 presents the space-time adaptive procedure. Finally, the methodology is illustrated in section 6 with two numerical examples. The paper is concluded with some remarks.

2 Problem statement

2.1 Governing equations

A visco-elastic body occupies an open bounded domain $\Omega \subset \mathbb{R}^d$, $d \leq 3$, with boundary $\partial\Omega$. The boundary is divided in two disjoint parts, Γ_N and Γ_D such that $\partial\Omega = \overline{\Gamma_\mathrm{N}} \cup \overline{\Gamma_\mathrm{D}}$ and the considered time interval is $I := (0, T]$. Under the assumption of small perturbations, the evolution of displacements $\mathbf{u}(\mathbf{x}, t)$ and stresses $\boldsymbol{\sigma}(\mathbf{x}, t)$, $\mathbf{x} \in \Omega$ and $t \in I$, is described by the visco-elastodynamic equations,

$$\rho(\ddot{\mathbf{u}} + a_1\dot{\mathbf{u}}) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{f} \quad \text{in } \Omega \times I, \tag{1a}$$
$$\mathbf{u} = \mathbf{0} \quad \text{on } \Gamma_\mathrm{D} \times I, \tag{1b}$$
$$\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{g} \quad \text{on } \Gamma_\mathrm{N} \times I, \tag{1c}$$
$$\mathbf{u} = \mathbf{u}_0 \quad \text{at } \Omega \times \{0\}, \tag{1d}$$
$$\dot{\mathbf{u}} = \mathbf{v}_0 \quad \text{at } \Omega \times \{0\}, \tag{1e}$$

where an upper dot indicates derivation with respect to time, that is $(\dot{\bullet}) := \frac{\mathrm{d}}{\mathrm{d}t}(\bullet)$, and $\mathbf{n}$ denotes the outward unit normal to $\partial\Omega$. The input data includes the mass density $\rho = \rho(\mathbf{x}) > 0$, the first Rayleigh coefficient $a_1 \geq 0$, the body force $\mathbf{f} = \mathbf{f}(\mathbf{x}, t)$ and the traction $\mathbf{g} = \mathbf{g}(\mathbf{x}, t)$ acting on the Neumann boundary $\Gamma_\mathrm{N} \times I$. The initial conditions for displacements and velocities are $\mathbf{u}_0 = \mathbf{u}_0(\mathbf{x})$ and $\mathbf{v}_0 = \mathbf{v}_0(\mathbf{x})$ respectively. For the sake of simplicity and without any loss of generality, Dirichlet conditions (1b) are taken as homogeneous.

The set of equations (1) is closed with the constitutive law,

$$\boldsymbol{\sigma} := \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}(\mathbf{u} + a_2\dot{\mathbf{u}}), \tag{2}$$

where the parameter $a_2 \geq 0$ is the second Rayleigh coefficient, the tensor $\boldsymbol{\mathcal{C}}$ is the standard 4th-order elastic Hooke tensor. The strains are given by the kinematic relation corresponding to small perturbations, that is $\boldsymbol{\varepsilon}(\mathbf{w}) := \frac{1}{2}(\boldsymbol{\nabla}\mathbf{w} + \boldsymbol{\nabla}^\mathrm{T}\mathbf{w})$.

2.2 Numerical approximation

In order to properly split the space and time error components, the adaptive strategy presented in this paper requires that the numerical solution under consideration fulfills the discrete version of a variational formulation. Thus, a weak residual (integrated both in space and time) associated with the numerical solution is readily introduced. The splitting procedure uses the fact that the residual vanishes for the functions in the test space, that is Galerkin orthogonality holds.

Among the possible space-time variational formulations available for transient elastodynamics, the double

field *time-continuous Galerkin method* [10,2] is the numerical solver selected. Note however that the rationale of this article can be easily extended to other space-time variational formulations, for instance, the one proposed by Johnson [14] or the one proposed by Hulbert and Hughes [12,13].

The definition of the weak form of the problem requires introducing the following functional spaces: the standard Sobolev space associated with static displacement fields

$$\boldsymbol{\mathcal{V}}_0 := \{\mathbf{w} \in [H^1(\Omega)]^d : \ \mathbf{w} = \mathbf{0} \text{ on } \Gamma_{\mathrm{D}}\} \tag{3}$$

and the Bochner space $L^2(I; \boldsymbol{\mathcal{V}}_0)$ associated with $\boldsymbol{\mathcal{V}}_0$ of square-integrable functions from I into $\boldsymbol{\mathcal{V}}_0$. With these notations, the trial space $\boldsymbol{\mathcal{W}}$ for the double field time-continuous Galerkin method is defined as

$$\boldsymbol{\mathcal{W}} := \{\mathbf{w} \in L^2(I; \boldsymbol{\mathcal{V}}_0) : \ \dot{\mathbf{w}} \in L^2(I; \boldsymbol{\mathcal{V}}_0')\}.$$

Note that, $\mathbf{w} \in \boldsymbol{\mathcal{W}}$ implies that $\mathbf{w} \in C^0(\bar{I}; [L^2(\Omega)]^d)$ and therefore functions in $\boldsymbol{\mathcal{W}}$ are continuous both in space and time, but they do not necessarily have a continuous time derivative.

The test space is associated with a partition of the time interval I defined as $\mathcal{T} := \{t_0, t_1, \dots, t_N\}$, with $0 = t_0 < t_1 < \dots < t_N = T$. The time points in $\mathcal{T}$ define the time intervals $I_n := (t_{n-1}t_n]$, $n = 1, \dots, N$. The time step length for each interval is $\Delta t_n := t_n - t_{n-1}$, $n = 1, \dots, N$ and the characteristic time step length for the partition $\mathcal{T}$ is $\Delta t := \max\limits_{1 \leq n \leq N}(\Delta t_n)$.

The test space is defined as

$$\widehat{\boldsymbol{\mathcal{W}}} := \{\mathbf{w} \in L^2(I; \boldsymbol{\mathcal{V}}_0) : \ \mathbf{w}|_{I_n} \in L^2(I_n; \boldsymbol{\mathcal{V}}_0) \text{ and}$$
$$\dot{\mathbf{w}}|_{I_n} \in L^2(I_n; \boldsymbol{\mathcal{V}}_0'), \ n = 1, \dots, N\}.$$

Functions in $\widehat{\boldsymbol{\mathcal{W}}}$ when restricted to a time interval I_n have the same regularity as functions in $\boldsymbol{\mathcal{W}}$. However, functions in $\widehat{\boldsymbol{\mathcal{W}}}$ are allowed to be discontinuous-in-time at the points in $\mathcal{T}$. This property is needed to define a time marching scheme, computing the solution successively in each time interval.

Using these notations, the space-time weak form of problem (1) reads: find $\mathbf{U} = [\mathbf{u}_u, \mathbf{u}_v] \in \boldsymbol{\mathcal{W}} \times \boldsymbol{\mathcal{W}}$ such that

$$B(\mathbf{U}, \mathbf{W}) = L(\mathbf{W}) \quad \forall \mathbf{W} := [\mathbf{w}_u, \mathbf{w}_v] \in \widehat{\boldsymbol{\mathcal{W}}} \times \widehat{\boldsymbol{\mathcal{W}}}, \tag{4}$$

where the bilinear form $B(\cdot, \cdot)$ and the linear functional $L(\cdot)$ are defined as

$$B(\mathbf{U}, \mathbf{W}) := \int_I m(\dot{\mathbf{u}}_v + a_1 \mathbf{u}_v, \mathbf{w}_v) \, \mathrm{d}t$$
$$+ \int_I a(\mathbf{u}_u + a_2 \mathbf{u}_v, \mathbf{w}_v) \, \mathrm{d}t + m(\mathbf{u}_v(0), \mathbf{w}_v(0))$$
$$+ \int_I a(\dot{\mathbf{u}}_u - \mathbf{u}_v, \mathbf{w}_u) \, \mathrm{d}t + a(\mathbf{u}_u(0), \mathbf{w}_u(0)),$$
$$L(\mathbf{W}) := \int_I l(t; \mathbf{w}_v) \, \mathrm{d}t$$
$$+ a(\mathbf{u}_0, \mathbf{w}_u(0)) + m(\mathbf{v}_0, \mathbf{w}_v(0)),$$

where

$$a(\mathbf{v}, \mathbf{w}) := \int_\Omega \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}(\mathbf{w}) \, \mathrm{d}\Omega,$$
$$m(\mathbf{v}, \mathbf{w}) := \int_\Omega \rho \mathbf{v} \cdot \mathbf{w} \, \mathrm{d}\Omega,$$
$$l(t; \mathbf{w}) := (\mathbf{f}(t), \mathbf{w}) + (\mathbf{g}(t), \mathbf{w})_{\Gamma_{\mathrm{N}}},$$

and

$$(\mathbf{v}, \mathbf{w}) := \int_\Omega \mathbf{v} \cdot \mathbf{w} \, \mathrm{d}\Omega, \quad (\mathbf{v}, \mathbf{w})_{\Gamma_{\mathrm{N}}} := \int_{\Gamma_{\mathrm{N}}} \mathbf{v} \cdot \mathbf{w} \, \mathrm{d}\Gamma.$$

The weak problem (4) is a *double field* formulation, having two unknowns, displacements $\mathbf{u}_u$ and velocities $\mathbf{u}_v$, which are a priori independent. That is, the velocity $\mathbf{u}_v$ is not strongly enforced to coincide with $\dot{\mathbf{u}}_u$. However, the relation between displacements and velocities is weakly imposed by means of the term $a(\dot{\mathbf{u}}_u - \mathbf{u}_v, \mathbf{w}_u)$.

The initial conditions (1d) and (1e) are also weakly imposed introducing the terms $a(\mathbf{u}_u(0) - \mathbf{u}_0, \mathbf{w}_u(0))$ and $m(\mathbf{u}_v(0) - \mathbf{v}_0, \mathbf{w}_v(0))$ respectively. The weak problem (4) is consistent with the original strong problem (1) in the sense that the solution $\mathbf{u}$ of problem (1) fulfills

$$B([\mathbf{u}, \dot{\mathbf{u}}], \mathbf{W}) = L(\mathbf{W}) \quad \forall \mathbf{W} \in \widehat{\boldsymbol{\mathcal{W}}} \times \widehat{\boldsymbol{\mathcal{W}}}.$$

The fully discrete version of problem (4) requires introducing a finite element partition of the domain Ω, which in the framework of mesh adaptivity is allowed to be different at each time point in $\mathcal{T}$. The finite element mesh discretizing the spatial domain Ω associated with time $t_n \in \mathcal{T}$ is denoted in the following by $\mathcal{P}_n$. The associated finite element space of continuous, elementwise polynomials of degree p is referred as $\boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n) \subset \boldsymbol{\mathcal{V}}_0$. The notation emphasizing the dependence on $\mathcal{P}_n$ highlights the fact that the finite element space depends on the computational mesh. The upper-script H stands for the characteristic element size in the mesh and it is included in the notation to indicate the discrete character of the finite element space. In the case that different values of p have to be accounted for, the notation is completed adding p as upper-script, e.g. the spaces

$\boldsymbol{\mathcal{V}}_0^{H,p}(\mathcal{P}_n)$ and $\boldsymbol{\mathcal{V}}_0^{H,p+1}(\mathcal{P}_n)$ are also used in the following.

The space meshes $\mathcal{P}_n$ are built considering a hierarchical tree-based mesh refinement strategy [6,8,31]. In this framework, the computational meshes are obtained recursively splitting the elements of an initial background mesh denoted as $\mathcal{P}_{\mathrm{bg}}$ as shown in figure 1. Thus, $\boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_{\mathrm{bg}}) \subset \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n)$ for all the spatial meshes $n = 0, \ldots, N$.
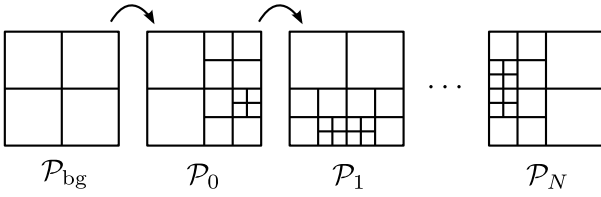

Fig. 1 A hierarchical tree-based technique is used to build the space meshes $\mathcal{P}_n$, $n = 0, \ldots, N$ from the background mesh $\mathcal{P}_{\mathrm{bg}}$.

The tree-based structure enormously facilitates the mesh refinement and unrefinement operations as well as the data transfer between different meshes. However, this approach requires dealing with a conforming approximation on an irregular spatial meshes involving *hanging* or *irregular* nodes. A constrained finite element approximation is used to enforce the continuity of the finite element solution across the edges of the mesh containing hanging nodes (introducing constraints on the local basis functions). A detailed description is given in appendix A.

The fully discrete problem is obtained replacing in (4) the trial and test spaces $\boldsymbol{\mathcal{W}}$ and $\widehat{\boldsymbol{\mathcal{W}}}$ by their discrete counterparts. For the sake of simplicity and without loss of generality, the method is presented here for piecewise linear (in time) trial functions. Hence, the time dependence of the approximations for displacements and velocities corresponds to a linear interpolation inside the time intervals I_n (piecewise linear in I). The space dependence is inherited from the spaces $\boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n)$. The resulting discrete space-time functional spaces read

$$\boldsymbol{\mathcal{W}}_u^{H,\Delta t} := \{\mathbf{w} \in \boldsymbol{\mathcal{W}} : \ \mathbf{w}(0) = \mathbf{u}_0,$$
$$\mathbf{w}(t) = \sum_{n=0}^N \theta_n(t)\mathbf{w}(t_n),$$
$$\mathbf{w}(t_n) \in \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n), \ n = 0, \ldots, N\},$$

and

$$\boldsymbol{\mathcal{W}}_v^{H,\Delta t} := \{\mathbf{w} \in \boldsymbol{\mathcal{W}} : \ \mathbf{w}(0) = \mathbf{v}_0,$$
$$\mathbf{w}(t) = \sum_{n=0}^N \theta_n(t)\mathbf{w}(t_n),$$
$$\mathbf{w}(t_n) \in \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n), \ n = 0, \ldots, N\},$$

where $\theta_n(t)$ are the linear shape functions associated with the time grid $\mathcal{T}$. Note that functions in $\boldsymbol{\mathcal{W}}_u^{H,\Delta t}$ and $\boldsymbol{\mathcal{W}}_v^{H,\Delta t}$ are continuous piecewise polynomials fulfilling the initial conditions for displacements and velocities respectively. Functions $\mathbf{w} \in \boldsymbol{\mathcal{W}}_u^{H,\Delta t}$ are such that at the points of the time grid, $t_n \in \mathcal{T}$, they belong to one of the standard Finite Element spaces, namely $\mathbf{w}(t_n) \in \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n)$. At an intermediate time $t \in I_n$, $t \neq t_n$, function $\mathbf{w}(t)$ belongs to $\boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_{n-1}) + \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n)$, that is the space generated by the superposition of the two meshes $\mathcal{P}_n$ and $\mathcal{P}_{n+1}$, see figure 2. The same holds for functions in $\boldsymbol{\mathcal{W}}_v^{H,\Delta t}$.

The fully discrete test space $\widehat{\boldsymbol{\mathcal{W}}}^{H,\Delta t}$ is defined as

$$\widehat{\boldsymbol{\mathcal{W}}}^{H,\Delta t} := \{\mathbf{w} \in \widehat{\boldsymbol{\mathcal{W}}} : \ \mathbf{w}|_{I_n} \in \mathbb{P}^0(I_n; \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n)),$$
$$n = 1, \ldots, N\},$$

where $\mathbb{P}^0(I_n; \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n))$ denotes the space of constant functions taking values in I_n and returning a value in $\boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n)$. Functions in $\widehat{\boldsymbol{\mathcal{W}}}^{H,\Delta t}$ are continuous piecewise polynomials in space and piecewise constants in time. Function $\mathbf{w} \in \widehat{\boldsymbol{\mathcal{W}}}^{H,\Delta t}$ is such that, for a time $t \in I_n$, $\mathbf{w}(t) \in \boldsymbol{\mathcal{V}}_0^H(\mathcal{P}_n)$, see figure 2. The polynomial dependence in time of functions in $\widehat{\boldsymbol{\mathcal{W}}}^{H,\Delta t}$ is one degree lower (piecewise constants) than the polynomial dependence in time of the trial space $\boldsymbol{\mathcal{W}}_u^{H,\Delta t}$ (piecewise linear). In this case, the trial and test spaces have the same number of degrees of freedom.

Using the discrete trial and test spaces, the fully discrete problem reads: find $\widetilde{\mathbf{U}} := [\tilde{\mathbf{u}}_u, \tilde{\mathbf{u}}_v] \in \boldsymbol{\mathcal{W}}_u^{H,\Delta t} \times \boldsymbol{\mathcal{W}}_v^{H,\Delta t}$ such that

$$B(\widetilde{\mathbf{U}}, \mathbf{W}) = L(\mathbf{W}) \quad \forall \mathbf{W} \in \widehat{\boldsymbol{\mathcal{W}}}^{H,\Delta t} \times \widehat{\boldsymbol{\mathcal{W}}}^{H,\Delta t}. \tag{5}$$

Problem (5) is integrated over the whole space-time domain $\Omega \times I$. However, having selected discontinuous test functions results in a time marching scheme that solves successively N problems in the time slabs $\Omega \times I_n$, $n = 1, \ldots, N$. Note that the step by step computational methodology resembles the classical time integration methods based on finite differences (i.e. Crank Nicholson, Newmark, etc.). In fact, if the mesh does not change, then the discrete displacements and velocities $\tilde{\mathbf{u}}_u$ and $\tilde{\mathbf{u}}_v$ at times t_n, $n = 1, \ldots, N$, coincide with the
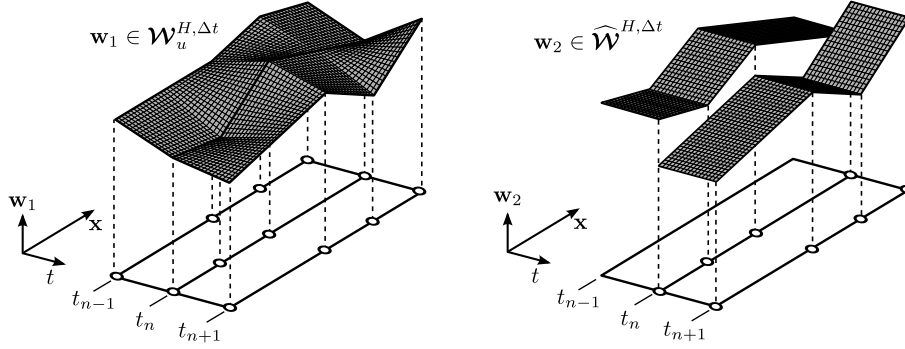
Fig. 2 Illustration of two generic functions, $\mathbf{w}_1 \in \mathcal{W}_u^{H,\Delta t}$ (left) and $\mathbf{w}_2 \in \widehat{\mathcal{W}}^{H,\Delta t}$ (right), inside the time interval $(t_{n-1}, t_{n+1}]$ when the time points t_{n-1}, t_n and t_{n+1} have different computational meshes. The active nodes in meshes $\mathcal{P}_{n-1}$, $\mathcal{P}_n$ and $\mathcal{P}_{n+1}$ are marked with circles ($\circ$) on the x-t plane.

approximation given by the Newmark method with parameters $\beta = 1/4$ and $\gamma = 1/2$, see [2] for a detailed proof. The actual resolution of problem (5) is detailed in appendix A.

2.3 Discretization error and error equation

The discretization error associated with $\widetilde{\mathbf{U}}$ is defined as

$$\mathbf{E} := \mathbf{U} - \widetilde{\mathbf{U}} = [\mathbf{e}_u, \mathbf{e}_v] = [\mathbf{u} - \tilde{\mathbf{u}}_u, \dot{\mathbf{u}} - \tilde{\mathbf{u}}_v] \in \mathcal{W} \times \mathcal{W},$$

where $\mathbf{e}_u$ and $\mathbf{e}_v$ are the errors in displacements and velocities respectively. The error $\mathbf{E}$ fulfills the following residual equation: find $\mathbf{E} = [\mathbf{e}_u, \mathbf{e}_v] \in \mathcal{W} \times \mathcal{W}$ such that

$$B(\mathbf{E}, \mathbf{W}) = R(\mathbf{W}) := L(\mathbf{W}) - B(\widetilde{\mathbf{U}}, \mathbf{W})$$
$$\forall \mathbf{W} \in \widehat{\mathcal{W}} \times \widehat{\mathcal{W}}, \quad (6)$$

which is derived replacing the exact solution $\mathbf{U}$ by $\widetilde{\mathbf{U}} + \mathbf{E}$ in (4) and using linearity of the forms $B(\cdot, \cdot)$ and $L(\cdot)$.

The residual $R(\cdot)$ fulfils the Galerkin orthogonality property

$$R(\mathbf{W}) = 0 \text{ for all } \mathbf{W} \in \widehat{\mathcal{W}}^{H,\Delta t} \times \widehat{\mathcal{W}}^{H,\Delta t}. \quad (7)$$

Although the Galerkin orthogonality property of the residual $R(\cdot)$ is not necessary to derive an error estimate for the error in the quantity of interest, it is required in the space-time adaptive strategy in order to properly split the space and time error contributions.

3 Goal-oriented modal-based error assessment

3.1 Quantity of interest and adjoint problem

The proposed a posteriori error estimation adaptive strategy aims at assessing and controlling the discretization error $\mathbf{E}$ measured using some specific quantity of interest. The quantity of interest is defined by means of a bounded lineal functional $L^{\mathcal{O}} : \mathcal{W} \times \mathcal{W} \longrightarrow \mathbb{R}$ which extracts a single representative scalar value of the whole space-time solution, namely

$$L^{\mathcal{O}}(\mathbf{W}) := L_u^{\mathcal{O}}(\mathbf{w}_u) + L_v^{\mathcal{O}}(\mathbf{w}_v), \quad (8)$$

where $L_u^{\mathcal{O}} : \mathcal{W} \longrightarrow \mathbb{R}$ and $L_v^{\mathcal{O}} : \mathcal{W} \longrightarrow \mathbb{R}$ are linear functionals representing quantities of interest for displacements and velocities respectively.

The estimation of the value $s^e := L^{\mathcal{O}}(\mathbf{E})$ requires introducing an auxiliary problem associated with the functional $L^{\mathcal{O}}(\cdot)$, usually denoted by *adjoint* or *dual* problem. The variational form of the adjoint problem reads: find $\mathbf{U}^d := [\mathbf{u}_u^d, \mathbf{u}_v^d] \in \mathcal{W} \times \mathcal{W}$ such that

$$B(\mathbf{W}, \mathbf{U}^d) = L^{\mathcal{O}}(\mathbf{W}) \quad \forall \mathbf{W} \in \widehat{\mathcal{W}} \times \widehat{\mathcal{W}}. \quad (9)$$

The adjoint solution characterizes the quantity of interest $L^{\mathcal{O}}(\cdot)$ in the sense that, if $\mathbf{U}^d$ is available, then the functional $L^{\mathcal{O}}(\cdot)$ coincides with $B(\cdot, \mathbf{U}^d)$, and in particular the computable quantity $L(\mathbf{U}^d)$ is equal to the quantity of interest $L^{\mathcal{O}}(\mathbf{U})$.

In practice, the functional $L^{\mathcal{O}}(\cdot)$ is selected with the same structure as $L(\cdot)$, namely

$$L_u^{\mathcal{O}}(\mathbf{w}_u) := a(\mathbf{u}^{\mathcal{O}}, \mathbf{w}_u(T)) \text{ and} \quad (10a)$$

$$L_v^{\mathcal{O}}(\mathbf{w}_v) := \int_0^T (\mathbf{f}^{\mathcal{O}}(t), \mathbf{w}_v(t)) \, dt$$
$$+ \int_0^T (\mathbf{g}^{\mathcal{O}}(t), \mathbf{w}_v(t))_{\Gamma_N} \, dt + m(\mathbf{v}^{\mathcal{O}}, \mathbf{w}_v(T)), \quad (10b)$$

where $\mathbf{f}^{\mathcal{O}}$, $\mathbf{g}^{\mathcal{O}}$, $\mathbf{v}^{\mathcal{O}}$ and $\mathbf{u}^{\mathcal{O}}$ are the data characterizing the quantity of interest. The functions $\mathbf{f}^{\mathcal{O}}$ and $\mathbf{g}^{\mathcal{O}}$ extract global or localized averages of velocities in Ω and Γ_N, respectively, integrated over the whole time interval $[0, T]$. The fields $\mathbf{v}^{\mathcal{O}}$ and $\mathbf{u}^{\mathcal{O}}$ play the role of weighting

functions to compute averages of velocities and strains at the final simulation time T.

For the description of $L^{\mathcal{O}}(\cdot)$ given in (10), the weak adjoint problem (9) is equivalent to the following strong equation for the adjoint displacement $\mathbf{u}^{\mathrm{d}}$,

$$\rho(\ddot{\mathbf{u}}^{\mathrm{d}} - a_1 \dot{\mathbf{u}}^{\mathrm{d}}) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^{\mathrm{d}}(\mathbf{u}^{\mathrm{d}}) = -\mathbf{f}^{\mathcal{O}} \quad \text{in } \Omega \times I, \quad (11a)$$

$$\mathbf{u}^{\mathrm{d}} = \mathbf{0} \quad \text{on } \Gamma_{\mathrm{D}} \times I, \quad (11b)$$

$$\boldsymbol{\sigma}^{\mathrm{d}}(\mathbf{u}^{\mathrm{d}}) \cdot \mathbf{n} = -\mathbf{g}^{\mathcal{O}} \quad \text{on } \Gamma_{\mathrm{N}} \times I, \quad (11c)$$

$$\mathbf{u}^{\mathrm{d}} = \mathbf{u}^{\mathcal{O}} \quad \text{at } \Omega \times \{T\}, \quad (11d)$$

$$\dot{\mathbf{u}}^{\mathrm{d}} = \mathbf{v}^{\mathcal{O}} \quad \text{at } \Omega \times \{T\}, \quad (11e)$$

with the constitutive law

$$\boldsymbol{\sigma}^{\mathrm{d}}(\mathbf{u}^{\mathrm{d}}) := \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}(\mathbf{u}^{\mathrm{d}} - a_2 \dot{\mathbf{u}}^{\mathrm{d}}). \quad (12)$$

The strong problem (11) has the same structure as the original one (1) except that the terms affected by a_1 and a_2 have opposite sign and the conditions (11d) and (11e) are stated for $t = T$ instead that for $t = 0$ (final conditions instead of initial). Thus, the adjoint problem is solvable and stable if integrated backwards in time. The change of sign in the time direction brings the adjoint problem back to the same features and properties as the direct one.

3.2 Error representation

The adjoint problem allows rewriting the error in the quantity of interest in terms of residuals, combining the original and adjoint problems. Indeed, taking $\mathbf{W} = \mathbf{U}^{\mathrm{d}}$ in the error equation (6) and using the definition of the adjoint problem, the following representation for s^{e} is found

$$s^{\mathrm{e}} := R(\mathbf{U}^{\mathrm{d}}). \quad (13)$$

This error representation is useful because states that the error in the quantity of interest can be exactly computed if the adjoint solution $\mathbf{U}^{\mathrm{d}}$ is available. Moreover, in an error estimation setup where the exact adjoint solution is not known, replacing $\mathbf{U}^{\mathrm{d}}$ by a computable approximation $\tilde{\mathbf{U}}^{\mathrm{d}}$ in (13) gives an accurate approximation of the error in the quantity of interest

$$s^{\mathrm{e}} \approx R(\tilde{\mathbf{U}}^{\mathrm{d}}) =: \tilde{s}^{\mathrm{e}}. \quad (14)$$

The scalar estimate $\tilde{s}^{\mathrm{e}}$ provides a single scalar quantity accounting both for the total error associated with the space and time discretizations and therefore, it does not directly provide enough information to adapt separately the space and time discretizations.

The error representation (13) is rewritten in such a way that the contributions of the space and time discretization errors are separated. This is achieved by introducing projection operators $\boldsymbol{\Pi}^{H}$ and $\boldsymbol{\Pi}^{\Delta t}$ associated with the space and time discretizations.

The spatial projection $\boldsymbol{\Pi}^{H}$ is defined for a function in $\mathbf{W} \in \widehat{\mathcal{W}} \times \widehat{\mathcal{W}}$ and provides a function which is discrete in space. The spatial discretization (the mesh) varies along the time but it is constant in a time interval I_n. Thus, the operator $\boldsymbol{\Pi}^{H}$ is defined for $t \in I_n$, $n = 1, \ldots, N$, as

$$[\boldsymbol{\Pi}^{H}\mathbf{W}](t) := [\boldsymbol{\pi}_n^{H}\mathbf{w}_u(t), \boldsymbol{\pi}_n^{H}\mathbf{w}_v(t)],$$

being $\boldsymbol{\pi}_n^{H}$ the standard interpolation operator from $\boldsymbol{\mathcal{V}}_0$ into $\boldsymbol{\mathcal{V}}_0^{H}(\mathcal{P}_n)$. On the other hand, the projection in time operator $\boldsymbol{\Pi}^{\Delta t}$ maps the time-dependent function $\mathbf{W} \in \widehat{\mathcal{W}} \times \widehat{\mathcal{W}}$ into a piecewise constant in time function. This projection is defined by taking the average of its displacement and velocity components inside each time interval I_n

$$[\boldsymbol{\Pi}^{\Delta t}\mathbf{W}]|_{I_n} := [\boldsymbol{\pi}_n^{\Delta t}\mathbf{w}_u, \boldsymbol{\pi}_n^{H}\mathbf{w}_v],$$

where

$$\boldsymbol{\pi}_n^{\Delta t}\mathbf{w} := \frac{1}{\mathrm{meas}(I_n)} \int_{I_n} \mathbf{w} \, \mathrm{d}t.$$

Remark 1. *Figure 3 illustrates the projection operators $\boldsymbol{\Pi}^{H}$ and $\boldsymbol{\Pi}^{\Delta t}$ using a generic function $\mathbf{W} \in \widehat{\mathcal{W}} \times \widehat{\mathcal{W}}$. Function $\boldsymbol{\Pi}^{H}\mathbf{W}$ belongs to the space $\widehat{\mathcal{W}}^{H} \times \widehat{\mathcal{W}}^{H}$, where*

$$\widehat{\mathcal{W}}^{H} := \{\mathbf{w} \in \widehat{\mathcal{W}} : \mathbf{w}|_{I_n} \in L^2(I_n; \boldsymbol{\mathcal{V}}_0^{H}(\mathcal{P}_n)), \text{ and}$$
$$\dot{\mathbf{w}}|_{I_n} \in L^2(I_n; (\boldsymbol{\mathcal{V}}_0^{H}(\mathcal{P}_n))') \ n = 1, \ldots, N\}.$$

Note that $\boldsymbol{\Pi}^{H}\mathbf{W}$ is discrete in space: for each particular time $t \in I$, function $[\boldsymbol{\Pi}^{H}\mathbf{W}](t)$ belongs to one of the discrete finite elements spaces $\boldsymbol{\mathcal{V}}_0^{H}(\mathcal{P}_n) \times \boldsymbol{\mathcal{V}}_0^{H}(\mathcal{P}_n)$. However, the time description of $\boldsymbol{\Pi}^{H}\mathbf{W}$ is infinite dimensional: for a given $\mathbf{x} \in \Omega$, $\boldsymbol{\Pi}^{H}\mathbf{W}(\mathbf{x}, \cdot) \in L^2(I) \times L^2(I)$.

On the other hand, the function $\boldsymbol{\Pi}^{\Delta t}\mathbf{W}$ belong to $\widehat{\mathcal{W}}^{\Delta t} \times \widehat{\mathcal{W}}^{\Delta t}$, where

$$\widehat{\mathcal{W}}^{\Delta t} := \{\mathbf{w} \in \widehat{\mathcal{W}} : \mathbf{w}|_{I_n} \in \mathbb{P}^0(I_n; \boldsymbol{\mathcal{V}}_0),$$
$$n = 1, \ldots, N\}.$$

Note that $\boldsymbol{\Pi}^{\Delta t}\mathbf{W}$ is piecewise constant in time, but its spatial description is infinite dimensional, namely $\boldsymbol{\Pi}^{\Delta t}\mathbf{W}(\cdot, t) \in \boldsymbol{\mathcal{V}}_0 \times \boldsymbol{\mathcal{V}}_0$.

Once the space and time projections are introduced, the space and time errors are separated adding the value $R(\boldsymbol{\Pi}^{H}\mathbf{U}^{\mathrm{d}}) - R(\boldsymbol{\Pi}^{H}\mathbf{U}^{\mathrm{d}}) + R(\boldsymbol{\Pi}^{H}\boldsymbol{\Pi}^{\Delta t}\mathbf{U}^{\mathrm{d}})$ in the right hand side of (13) (the latter term vanishes due to the
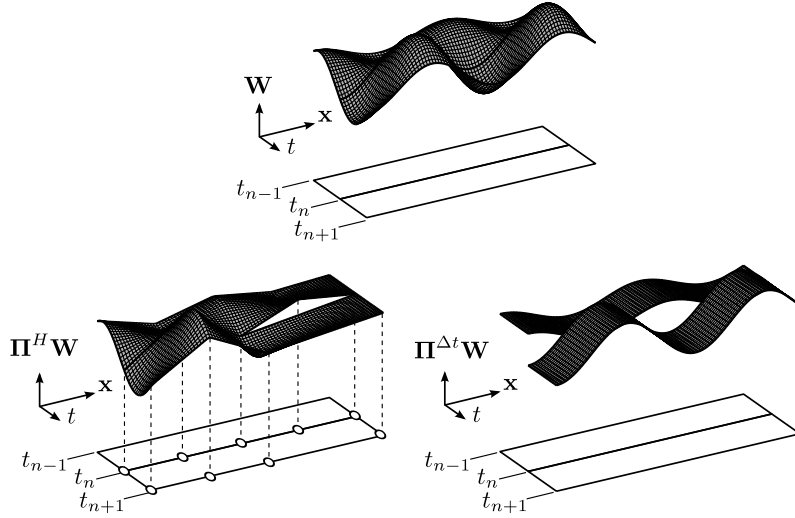
Fig. 3 Illustration of the projection operators $\boldsymbol{\Pi}^H$ and $\boldsymbol{\Pi}^{\Delta t}$. The figure displays (one field of) the original function $\mathbf{W} \in \widehat{\mathcal{W}} \times \widehat{\mathcal{W}}$ inside the time intervals $I_n = (t_{n-1}, t_n]$ and $I_{n+1} = (t_n, t_{n+1}]$ (top) along with its projections in space and time $\boldsymbol{\Pi}^H \mathbf{W} \in \widehat{\mathcal{W}}^H \times \widehat{\mathcal{W}}^H$ (left) and $\boldsymbol{\Pi}^{\Delta t}\mathbf{W} \in \widehat{\mathcal{W}}^{\Delta t} \times \widehat{\mathcal{W}}^{\Delta t}$ (right).

Galerkin orthogonality property because $\boldsymbol{\Pi}^H \boldsymbol{\Pi}^{\Delta t}\mathbf{U}^{\mathrm{d}} \in \widehat{\mathcal{W}}^{H,\Delta t} \times \widehat{\mathcal{W}}^{H,\Delta t}$). That is,

$$s^{\mathrm{e}} = \underbrace{R(\mathbf{U}^{\mathrm{d}} - \boldsymbol{\Pi}^H \mathbf{U}^{\mathrm{d}})}_{=: \, s_{\mathrm{s}}^{\mathrm{e}}} + \underbrace{R(\boldsymbol{\Pi}^H(\mathbf{U}^{\mathrm{d}} - \boldsymbol{\Pi}^{\Delta t}\mathbf{U}^{\mathrm{d}}))}_{=: \, s_{\mathrm{t}}^{\mathrm{e}}}. \quad (15)$$

The terms $s_{\mathrm{s}}^{\mathrm{e}}$ and $s_{\mathrm{t}}^{\mathrm{e}}$ are associated with the space and time discretization errors respectively. Note that $s_{\mathrm{s}}^{\mathrm{e}}$ tends to zero as the space discretization is refined because $\boldsymbol{\Pi}^H \mathbf{U}^{\mathrm{d}}$ tends to $\mathbf{U}^{\mathrm{d}}$. Similarly, $s_{\mathrm{t}}^{\mathrm{e}}$ tends to zero with Δt because $\boldsymbol{\Pi}^{\Delta t}\mathbf{U}^{\mathrm{d}}$ tends to $\mathbf{U}^{\mathrm{d}}$. The space and time error components $s_{\mathrm{s}}^{\mathrm{e}}$ and $s_{\mathrm{t}}^{\mathrm{e}}$ are used as refinement indicators because they can be reduced independently by respectively enriching the space and time discretizations.

The space and time splitting is straightforwardly transformed to the estimated version of the error $\tilde{s}^{\mathrm{e}}$, replacing $\mathbf{U}^{\mathrm{d}}$ by the computable approximation $\tilde{\mathbf{U}}^{\mathrm{d}}$ in equation (15), namely

$$\tilde{s}^{\mathrm{e}} = \tilde{s}_{\mathrm{s}}^{\mathrm{e}} + \tilde{s}_{\mathrm{t}}^{\mathrm{e}}, \quad (16)$$

where $\tilde{s}_{\mathrm{s}}^{\mathrm{e}} := R(\tilde{\mathbf{U}}^{\mathrm{d}} - \boldsymbol{\Pi}^H \tilde{\mathbf{U}}^{\mathrm{d}})$ and $\tilde{s}_{\mathrm{t}}^{\mathrm{e}} := R(\boldsymbol{\Pi}^H \tilde{\mathbf{U}}^{\mathrm{d}} - \boldsymbol{\Pi}^H \boldsymbol{\Pi}^{\Delta t}\tilde{\mathbf{U}}^{\mathrm{d}})$ are the computable space and time error contributions.

3.3 Modal-based adjoint approximation

The error estimate $\tilde{s}^{\mathrm{e}}$ is computable once the adjoint approximation $\tilde{\mathbf{U}}^{\mathrm{d}}$ is available. Typically, the adjoint approximation is computed using the same code used for the original problem (1), i.e. using direct time-integration methods, see reference [2]. An alternative approach proposed in [30] considers modal analysis to compute the adjoint approximation. The modal-based strategy is particularly well suited for some particular quantities of interest and allows effectively computing and storing the adjoint problem. In that case, the adjoint solution is stored for each vibration mode instead of for each time step.

Modal analysis requires introducing the semidiscrete equation (discrete in space but exact in time) associated with the adjoint problem (11). Consequently, a discrete version of the functional space $\mathcal{V}_0$ is required. The semidiscrete problem is defined using the finite element space $\mathcal{V}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}})$, that stands for the finite element space associated with the mesh $\mathcal{P}_{\mathrm{bg}}$ of degree of interpolation $p + 1$ (a p-refined version of $\mathcal{V}_0^H(\mathcal{P}_{\mathrm{bg}})$). Having a $p + 1$ degree approximation of the adjoint solution, $\tilde{\mathbf{U}}^{\mathrm{d}}$, precludes the Galerkin orthogonality effect and the corresponding underestimation of the error, see [30]. Recall that, along the adaptive process, the background mesh is used as the base to build up all the adapted meshes by local refinement. Thus, the representation of $\tilde{\mathbf{U}}^{\mathrm{d}}$ in the adapted mesh is simplified if $\tilde{\mathbf{U}}^{\mathrm{d}}$ is in $\mathcal{V}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}})$.

With these definitions, the semidiscrete problem reads: find $\mathbf{u}^{\mathrm{d},H,p+1}(t) \in \mathcal{V}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}})$ verifying the final conditions $\mathbf{u}^{\mathrm{d},H,p+1}(T) = \mathbf{u}^{\mathcal{O}}$ and $\dot{\mathbf{u}}^{\mathrm{d},H,p+1}(T) = \mathbf{v}^{\mathcal{O}}$ and such that for all $t \in I$

$$m(\ddot{\mathbf{u}}^{\mathrm{d},H,p+1}(t) - a_1\dot{\mathbf{u}}^{\mathrm{d},H,p+1}, \mathbf{w})$$
$$+ a(\mathbf{u}^{\mathrm{d},H,p+1}(t) - a_2\dot{\mathbf{u}}^{\mathrm{d},H,p+1}(t), \mathbf{w}) =$$
$$-l^{\mathcal{O}}(t; \mathbf{w}) \quad \forall \mathbf{w} \in \mathcal{V}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}}), \quad (17)$$

where $l^{\mathcal{O}}(t; \mathbf{w}) := (\mathbf{f}^{\mathcal{O}}(t), \mathbf{w}) + (\mathbf{g}^{\mathcal{O}}(t), \mathbf{w})_{\Gamma_{\mathrm{N}}}$.

Equation (17) leads to an algebraic system of second order ordinary differential equations which is conveniently rewritten using the eigenvalues and eigenfunctions of the problem: find $(\tilde{\omega}, \tilde{\mathbf{q}}) \in \mathbb{R} \times \boldsymbol{\mathcal{V}}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}})$ such that

$$a(\tilde{\mathbf{q}}, \mathbf{w}) = \tilde{\omega}^2 m(\tilde{\mathbf{q}}, \mathbf{w}) \quad \forall \mathbf{w} \in \boldsymbol{\mathcal{V}}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}}). \tag{18}$$

Note that the number of eigenpairs solution of this problem is the number of degrees of freedom in the finite element space $\boldsymbol{\mathcal{V}}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}})$, denoted by N_{dof}. The eigenpairs are sorted from low to high frequencies, namely $\tilde{\omega}_1 \leq \tilde{\omega}_2 \cdots \leq \tilde{\omega}_{N_{\mathrm{dof}}}$, and the eigenfunctions are normalized to be orthonormal with respect the mass product, i.e.

$$m(\tilde{\mathbf{q}}_i, \tilde{\mathbf{q}}_j) = \delta_{ij}, \quad 1 \leq i, j \leq N_{\mathrm{dof}}. \tag{19}$$

The semidiscrete approximation $\mathbf{u}^{\mathrm{d},H,p+1}$ is expressed as a linear combination of the eigenfunctions $\tilde{\mathbf{q}}_i$

$$\mathbf{u}^{\mathrm{d},H,p+1}(\mathbf{x}, t) = \sum_{i=1}^{N_{\mathrm{dof}}} \tilde{\mathbf{q}}_i(\mathbf{x}) \tilde{y}_i(t). \tag{20}$$

Thus, the new unknowns of the problem are the time-dependent coefficients $\tilde{y}_i(t)$, $i = 1, \ldots, N_{\mathrm{dof}}$. The representation in terms of the unknowns $\tilde{y}_i(t)$ given in (20) allows uncoupling the system (17) into a set of ordinary differential equations, namely

$$\ddot{\tilde{y}}_i - [a_1 + a_2(\tilde{\omega}_i)^2]\dot{\tilde{y}}_i + (\tilde{\omega}_i)^2 \tilde{y}_i = \tilde{l}_i, \tag{21a}$$

$$y_i^H(T) = \tilde{u}_i, \tag{21b}$$

$$\dot{y}_i^H(T) = \tilde{v}_i, \tag{21c}$$

where the r.h.s. terms l_i, u_i and v_i are computed using the data characterizing the quantity of interest (10) and the eigenfunction $\tilde{\mathbf{q}}_i$

$$\tilde{l}_i(t) := (\mathbf{f}^{\mathcal{O}}(t), \tilde{\mathbf{q}}_i) + (\mathbf{g}^{\mathcal{O}}(t), \tilde{\mathbf{q}}_i)_{\Gamma_{\mathrm{N}}},$$
$$\tilde{u}_i := m(\mathbf{u}^{\mathcal{O}}, \tilde{\mathbf{q}}_i) \text{ and } \tilde{v}_i := m(\mathbf{v}^{\mathcal{O}}, \tilde{\mathbf{q}}_i). \tag{22}$$

The time dependent coefficients $\tilde{y}_i(t)$, $i = 1, \ldots, N_{\mathrm{dof}}$, may be computed analytically for many particular cases of the forcing data. The particular solution for constant-in-time data is given in [30]. Therefore the value of the adjoint solution $\mathbf{u}^{\mathrm{d},H,p+1}$ at any time $t \in I$ is easily reconstructed from the computed eigenfunctions $\tilde{\mathbf{q}}_i$ and the analyticaly computed time-dependent functions $\tilde{y}_i(t)$ using expression (20).

In practice, it is not feasible to compute all the eigenpairs $(\tilde{\omega}_i, \tilde{\mathbf{q}}_i)$, $i = 1, \ldots, N_{\mathrm{dof}}$ and consequently the modal expansion (20) has to be truncatied to the first $M \ll N_{\mathrm{dof}}$ terms, namely

$$\tilde{\mathbf{u}}^{\mathrm{d}}(\mathbf{x}, t) := \sum_{i=1}^{M} \tilde{\mathbf{q}}_i(\mathbf{x}) \tilde{y}_i(t). \tag{23}$$

The number of required vibration modes M has to be selected such that the truncated high frequency modes (for $i > M$) are negligible. That is, M is such that $\tilde{\mathbf{u}}^{\mathrm{d}}$ is a good approximation to $\mathbf{u}^{\mathrm{d},H,p+1}$. This is equivalent to assume that for $i > M$ the values of $\tilde{l}_i$, $\tilde{u}_i$ and $\tilde{v}_i$, as defined in (22), are close to zero, and consequently $\tilde{y}_i(t) \approx 0$. This is guaranteed if the data $\mathbf{f}^{\mathcal{O}}$, $\mathbf{g}^{\mathcal{O}}$, $\mathbf{u}^{\mathcal{O}}$ and $\mathbf{v}^{\mathcal{O}}$ are well captured by the expansion of the first M eigenvectors. Consequently, a quantity of interest can be easily treaded with the modal-based approach if its associated data $\mathbf{f}^{\mathcal{O}}$, $\mathbf{g}^{\mathcal{O}}$, $\mathbf{u}^{\mathcal{O}}$ and $\mathbf{v}^{\mathcal{O}}$ are well captured by the expansion of the first M eigenfunctions.

Once the computable adjoint approximation $\tilde{\mathbf{u}}^{\mathrm{d}}$ is available, the double field approximation $\tilde{\mathbf{U}}^{\mathrm{d}}$ used in the error estimate $\tilde{s}^{\mathrm{e}}$ given in (14) is readily defined as $\tilde{\mathbf{U}}^{\mathrm{d}} := [\tilde{\mathbf{u}}^{\mathrm{d}}, \dot{\tilde{\mathbf{u}}}^{\mathrm{d}}]$.

4 Space-time Adaptivity

4.1 Adaptivity framework

The space-time adaptive strategy aims at finding a time discretization $\mathcal{T}$ and a space discretization $\mathcal{P}_n$ at each time point $t_n \in \mathcal{T}$ such that 1) they keep the error s^{e} below a user-prescribed tolerance $s_{\mathrm{tol}}^{\mathrm{e}}$ and 2) they are optimal in the sense that they minimize the computational cost. In practice, the accuracy prescription is enforced for the estimated error and the property which is actually achieved is

$$|\tilde{s}^{\mathrm{e}}| \leq s_{\mathrm{tol}}^{\mathrm{e}}. \tag{24}$$

Changing the space discretization at each time step $t_n \in \mathcal{T}$ is not computationally affordable. This is because remeshing operations, matrix assembly and data transfer between different meshes are costly operations and cannot, in general, be performed at each time step. Here, an adaptive strategy organized in time-blocks, similar to the one proposed in reference [5], is adopted in order to reduce the number of mesh changes.

The blockwise adaptive strategy consist in splitting the time interval I into N^{bk} time intervals (or time blocks) The time interval I is split into N^{bk} time intervals (or time blocks)

$$I_m^{\mathrm{bk}} := \left(\frac{T}{N^{\mathrm{bk}}}(m-1), \frac{T}{N^{\mathrm{bk}}}m \right], \ m = 1, \ldots, N^{\mathrm{bk}}.$$

The blockwise adaptive strategy consists taking the same space mesh inside each time interval I_m^{bk}, this mesh is denoted as $\mathcal{P}_m^{\mathrm{bk}}$ for $m = 1, \ldots, N^{\mathrm{bk}}$, see figure 4. Note that with this definition the computational meshes $\mathcal{P}_n$ associated with the time points $t_n \in I_m^{\mathrm{bk}}$ are such that

$\mathcal{P}_n = \mathcal{P}_m^{\text{bk}}$. A generic element of the mesh $\mathcal{P}_m^{\text{bk}}$ is denoted by Ω_k^m, $k = 1, \ldots, N_m^{\text{el}}$, where N_m^{el} is the number of elements in $\mathcal{P}_m^{\text{bk}}$.

Additionally, the time step length is assumed to be constant inside the intervals I_m^{bk} and denoted by Δt_m^{bk}. Consequently, the time step length Δt_n associated with times $t_n \in I_m^{\text{bk}}$ are such that $\Delta t_n = \Delta t_m^{\text{bk}}$, see figure 4.
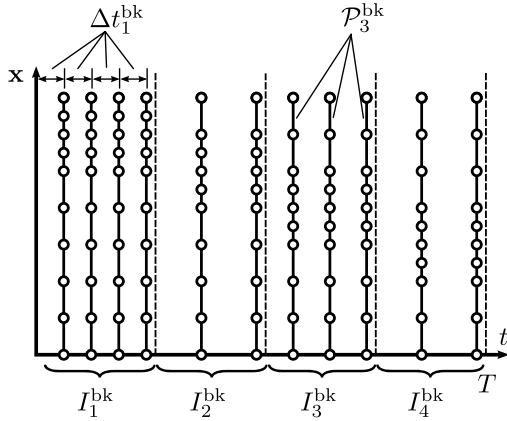
Fig. 4 The space mesh is assumed to be constant inside the time intervals I_m^{bk}. Analogously, the time step length is taken constant inside each interval I_m^{bk}.

Following this approach and notation, the adaptive strategy is reformulated as computing the optimal space meshes $\mathcal{P}_m^{\text{bk}}$ and time step lengths Δt_m^{bk}, for all the time intervals I_m^{bk}, $m = 1, \ldots, N^{\text{bk}}$ such that the associated numerical solution fulfills (24).

Once the adjoint solution is computed and stored in the $p + 1$ version of the background mesh (keeping the same geometry and topology but increasing the degree of polynomials from p to $p + 1$), the main stages of the adaptive procedure are summarized as follows. The numerical solution is computed sequentially starting from the first time block I_1^{bk} until the last one $I_{N^{\text{bk}}}^{\text{bk}}$. In each time slab I_m^{bk}, the numerical solution is computed and the corresponding local error contributions are estimated. The computed solution in I_m^{bk} is *accepted* or *rejected* using the information given by the local error contributions. The specific acceptability criterion is detailed later. If the solution is accepted, the loop goes to the following time interval I_{m+1}^{bk}. Else, the space or time discretization (or both) associated with the interval I_m^{bk} are adapted using the local error information and the solution is re-computed in I_m^{bk}. The process of adapting the discretization and computing the numerical solution is repeated in the interval I_m^{bk} until the solution is accepted.

The forthcoming subsections describe in detail 1) the local error contributions driving the adaptive process, 2) the criterion used to accept or reject the solution

in each interval I_m^{bk} and 3) how to adapt the space and time discretizations when required.

4.2 Local error contributions

The space and time error estimates $\tilde{s}_{\text{s}}^{\text{e}}$ and $\tilde{s}_{\text{t}}^{\text{e}}$ are decomposed into contributions associated with the time blocks I_m^{bk}, $m = 1, \ldots, N^{\text{bk}}$, namely

$$\eta_m^{\text{s}} := R_{I_m^{\text{bk}}}(\tilde{\mathbf{U}}^{\text{d}} - \boldsymbol{\Pi}^H \tilde{\mathbf{U}}^{\text{d}}), \text{ and}$$

$$\eta_m^{\text{t}} := R_{I_m^{\text{bk}}}(\boldsymbol{\Pi}^H(\tilde{\mathbf{U}}^{\text{d}} - \boldsymbol{\Pi}^{\Delta t}\tilde{\mathbf{U}}^{\text{d}}))$$

such that

$$\tilde{s}_{\text{s}}^{\text{e}} = \sum_{m=1}^{N^{\text{bk}}} \eta_m^{\text{s}} \quad \text{and} \quad \tilde{s}_{\text{t}}^{\text{e}} = \sum_{m=1}^{N^{\text{bk}}} \eta_m^{\text{t}}.$$

The local residual $R_{I_m^{\text{bk}}}(\cdot)$ is the restriction of the residual $R(\cdot)$ to the time interval I_m^{bk},

$$\begin{aligned}
R_{I_m^{\text{bk}}}(\mathbf{W}) := & \int_{I_m^{\text{bk}}} [(\mathbf{f}, \mathbf{w}_v) + (\mathbf{g}, \mathbf{w}_v)_{\Gamma_{\text{N}}}] \ \text{dt} \\
& - \int_{I_m^{\text{bk}}} m(\dot{\tilde{\mathbf{u}}}_v + a_1 \tilde{\mathbf{u}}_v, \mathbf{w}_v) \ \text{dt} \\
& + \int_{I_m^{\text{bk}}} a(\tilde{\mathbf{u}}_u + a_2 \tilde{\mathbf{u}}_v, \mathbf{w}_v) \ \text{dt} \\
& - \int_{I_m^{\text{bk}}} a(\dot{\tilde{\mathbf{u}}}_u - \tilde{\mathbf{u}}_v, \mathbf{w}_u) \ \text{dt}.
\end{aligned}$$

The indicator η_m^{t} is used to decide if the time discretization inside I_m^{bk} has to be modified. The criteria on wether the time grid has to be modified and how it has to be modified are presented in section 4.3.

The value of η_m^{s} is the indicator used to decide if the space mesh $\mathcal{P}_m^{\text{bk}}$ in the time interval I_m^{bk} has to be further adapted. Again, the detailed criteria are introduced in section 4.3. In the case the mesh is to be adapted, the required local error indicators are obtained by restricting the space integrals involved in η_m^{s} to the elements Ω_k^m. That is,

$$\eta_{m,k}^{\text{s}} := R_{\Omega_k^m \times I_m^{\text{bk}}}(\tilde{\mathbf{U}}^{\text{d}} - \boldsymbol{\Pi}^H \tilde{\mathbf{U}}^{\text{d}}),$$

where

$$\begin{aligned}
R_{\Omega_k^m \times I_m^{\text{bk}}}(\mathbf{W}) := & \int_{I_m^{\text{bk}}} [(\mathbf{f}, \mathbf{w}_v)_{\Omega_k^m} + (\mathbf{g}, \mathbf{w}_v)_{\partial\Omega_k^m \cap \Gamma_{\text{N}}}] \ \text{dt} \\
& - \int_{I_m^{\text{bk}}} m(\dot{\tilde{\mathbf{u}}}_v + a_1 \tilde{\mathbf{u}}_v, \mathbf{w}_v)_{\Omega_k^m} \ \text{dt} \\
& + \int_{I_m^{\text{bk}}} a(\tilde{\mathbf{u}}_u + a_2 \tilde{\mathbf{u}}_v, \mathbf{w}_v)_{\Omega_k^m} \ \text{dt} \\
& - \int_{I_m^{\text{bk}}} a(\dot{\tilde{\mathbf{u}}}_u - \tilde{\mathbf{u}}_v, \mathbf{w}_u)_{\Omega_k^m} \ \text{dt}.
\end{aligned}$$

Note that the error estimate $\tilde{s}^{\mathrm{e}}$ is expressed as the sum of the local error contributions defined above

$$\tilde{s}^{\mathrm{e}} = \sum_{m=1}^{N^{\mathrm{bk}}} \Big(\sum_{k=1}^{N_m^{\mathrm{el}}} \eta_{m,k}^{\mathrm{s}} \Big) + \sum_{m=1}^{N^{\mathrm{bk}}} \eta_m^{\mathrm{t}}.$$

4.3 Acceptability and remeshing criteria

Following references [2,5], the total target error $s_{\mathrm{tol}}^{\mathrm{e}}$ is split into two error targets, $\alpha_{\mathrm{s}} s_{\mathrm{tol}}^{\mathrm{e}}$ and $\alpha_{\mathrm{t}} s_{\mathrm{tol}}^{\mathrm{e}}$, associated with the space and time errors. The coefficients α_{s} and α_{t} are two user-defined positive values such that $\alpha_{\mathrm{s}} + \alpha_{\mathrm{t}} = 1$ used to balance the space and time contributions to the total error. This leaves a free parameter to be tuned by the user, who must decide the amount of the total error $s_{\mathrm{tol}}^{\mathrm{e}}$ assigned to the space and time discretizations. Discussing the optimal values for α_{s} and α_{t} is beyond the scope of this paper.

Thus, in order to achieve the accuracy prescription stated in (24), the adaptive strategy is designed aiming at finding a numerical solution such that

$$|\tilde{s}_{\mathrm{s}}^{\mathrm{e}}| \le \alpha_{\mathrm{s}} s_{\mathrm{tol}}^{\mathrm{e}} \quad \text{and} \quad |\tilde{s}_{\mathrm{t}}^{\mathrm{e}}| \le \alpha_{\mathrm{t}} s_{\mathrm{tol}}^{\mathrm{e}}. \tag{25}$$

Note that (25) guarantees that equation (24) holds, because

$$|\tilde{s}^{\mathrm{e}}| = |\tilde{s}_{\mathrm{s}}^{\mathrm{e}} + \tilde{s}_{\mathrm{t}}^{\mathrm{e}}| \le |\tilde{s}_{\mathrm{s}}^{\mathrm{e}}| + |\tilde{s}_{\mathrm{t}}^{\mathrm{e}}| \le \alpha_{\mathrm{s}} s_{\mathrm{tol}}^{\mathrm{e}} + \alpha_{\mathrm{t}} s_{\mathrm{tol}}^{\mathrm{e}} = s_{\mathrm{tol}}^{\mathrm{e}}.$$

The conditions (25) are more restrictive than (24). This is because in (24) $\tilde{s}_{\mathrm{t}}^{\mathrm{e}}$ and $\tilde{s}_{\mathrm{s}}^{\mathrm{e}}$ with different sign may cancel each other. The error compensation is not accounted for in (25) and therefore the resulting criterion is more demanding.

The error contributions are assumed to be uniformly distributed in time. That is, the space and time error tolerances, $\alpha_{\mathrm{s}} s_{\mathrm{tol}}^{\mathrm{e}}$ and $\alpha_{\mathrm{t}} s_{\mathrm{tol}}^{\mathrm{e}}$, are divided into equal contributions associated with each time block I_m^{bk}. Thus, the solution is considered to be acceptable if

$$|\eta_m^{\mathrm{s}}| \le \frac{\alpha_{\mathrm{s}} s_{\mathrm{tol}}^{\mathrm{e}}}{N^{\mathrm{bk}}}, \tag{26a}$$

$$|\eta_m^{\mathrm{t}}| \le \frac{\alpha_{\mathrm{t}} s_{\mathrm{tol}}^{\mathrm{e}}}{N^{\mathrm{bk}}}. \tag{26b}$$

If the restrictions (26) hold, then the inequalities (25) are fulfilled, because

$$|\tilde{s}_{\mathrm{s}}^{\mathrm{e}}| = \Big| \sum_{m=1}^{N^{\mathrm{bk}}} \eta_m^{\mathrm{s}} \Big| \le \sum_{m=1}^{N^{\mathrm{bk}}} |\eta_m^{\mathrm{s}}| \le \alpha_{\mathrm{s}} s_{\mathrm{tol}}^{\mathrm{e}} \text{ and}$$

$$|\tilde{s}_{\mathrm{t}}^{\mathrm{e}}| = \Big| \sum_{m=1}^{N^{\mathrm{bk}}} \eta_m^{\mathrm{t}} \Big| \le \sum_{m=1}^{N^{\mathrm{bk}}} |\eta_m^{\mathrm{t}}| \le \alpha_{\mathrm{t}} s_{\mathrm{tol}}^{\mathrm{e}}. \tag{27}$$

Similarly as when splitting the space and time contributions, criteria (26) are stronger than (25). This is

more relevant for large values of N^{bk}, because the effect of the *triangular* inequalities in the equations (27) is more important. Thus, the adapted numerical solution might be very conservative if the number of blocks N^{bk} is large.

An additional condition is added to (26) in order to allow unrefinement (mesh coarsening). Note that the conditions (26) indicate only if the solution is acceptable and, if not, if the mesh has to be refined. They do not provide a criterion to unrefine the discretization when the error indicators η_m^{s} and η_m^{t} are small enough. Following reference [2], a lower bound based acceptability criterion is added to (26):

$$\beta_{\mathrm{s}} \frac{\alpha_{\mathrm{s}} s_{\mathrm{tol}}^{\mathrm{e}}}{N^{\mathrm{bk}}} \le |\eta_m^{\mathrm{s}}|, \tag{28a}$$

$$\beta_{\mathrm{t}} \frac{\alpha_{\mathrm{t}} s_{\mathrm{tol}}^{\mathrm{e}}}{N^{\mathrm{bk}}} \le |\eta_m^{\mathrm{t}}|, \tag{28b}$$

where the coefficients β_{s} and β_{t} are two user-defined values such that $\beta_{\mathrm{s}}, \beta_{\mathrm{t}} \in [0, 1)$. If the solution does not fulfill condition (28b), then the time discretization is modified (in this case unrefined). If (28a) is violated, then the space mesh is modified and it is expected to be globally coarsened. However, the space mesh adaption is performed locally and may result in refining some parts of the domain while others are unrefined. The space remeshing criterion is described below. The coarsening criterion (28) is only checked once for each time block. This is because the need of unrefining the space or the time grid is expected to be detected with the first discretization. Moreover, checking for unrefining at each adaptive step may result in an unstable scheme.

As previously said, conditions (26) and (28) are the criteria allowing to decide if the numerical solution is accepted or rejected inside the interval I_m^{bk}. If conditions (26) and (28) hold (or only (26) after the first adaptive iteration), then the solution is accepted. Otherwise, the space and/or time discretizations are modified.

The time adaptivity is carried out, depending on the value of η_m^{t}, by either refining the discretization by halving the time step Δt_m^{bk} (if (26b) is violated) or doubling it (if (28b) is violated). If both (26b) and (28b) hold, the time discretization is unchanged.

If either (26a) or (28a) are not fulfilled, the space mesh is to be modified. Then, local criterion is required to decide which elements have to be refined or unrefined, depending on the value of the local indicators $\eta_{m,k}^{\mathrm{s}}$, $k = 1, \ldots, N_m^{\mathrm{el}}$ (for a given $m = 1, \ldots, N^{\mathrm{bk}}$). Similarly as for the time discretization, the elements to be refined are subdivided (the element size divided by two) while the elements to be coarsened are collapsed with the neighboring elements, doubling the element size. In

order to set up a space remeshing criterion, the optimal mesh is assumed to yield a uniform error distribution. Thus, the local versions (restricted to the contributions associated with element Ω_k^m) of the conditions (26a) and (28a) read

$$\gamma_m \beta_\mathrm{s} \frac{\alpha_\mathrm{s} s_\mathrm{tol}^\mathrm{e}}{N^\mathrm{bk} N_m^\mathrm{el}} \le \left| \eta_{m,k}^\mathrm{s} \right| \le \gamma_m \frac{\alpha_\mathrm{s} s_\mathrm{tol}^\mathrm{e}}{N^\mathrm{bk} N_m^\mathrm{el}}, \tag{29}$$

where

$$\gamma_m := \frac{\sum_{k=1}^{N_m^\mathrm{el}} \left| \eta_{m,k}^\mathrm{s} \right|}{\left| \sum_{k=1}^{N_m^\mathrm{el}} \eta_{m,k}^\mathrm{s} \right|} \ge 1.$$

The coefficient γ_m is introduced in order to mitigate the cancellation effect, see reference [9]. It is worth noting that introducing the factor γ_m does not introduce a distortion in the criterion: if all the local element error contributions fulfill (29), then equation (28a) holds. This is shown by noting that

$$\left| \eta_m^\mathrm{s} \right| = \left| \sum_{k=1}^{N_m^\mathrm{el}} \eta_{m,k}^\mathrm{s} \right| = \frac{1}{\gamma_m} \left(\sum_{k=1}^{N_m^\mathrm{el}} \left| \eta_{m,k}^\mathrm{s} \right| \right)$$

and therefore

$$\beta_\mathrm{s} \frac{\alpha_\mathrm{s} s_\mathrm{tol}^\mathrm{e}}{N^\mathrm{bk}} =$$
$$\frac{1}{\gamma_m} \left(\sum_{k=1}^{N_m^\mathrm{el}} \gamma_m \beta_\mathrm{s} \frac{\alpha_\mathrm{s} s_\mathrm{tol}^\mathrm{e}}{N^\mathrm{bk} N_m^\mathrm{el}} \right) \le \frac{1}{\gamma_m} \left(\sum_{k=1}^{N_m^\mathrm{el}} \left| \eta_{m,k}^\mathrm{s} \right| \right) = \left| \eta_m^\mathrm{s} \right|$$

and

$$\frac{\alpha_\mathrm{s} s_\mathrm{tol}^\mathrm{e}}{N^\mathrm{bk}} =$$
$$\frac{1}{\gamma_m} \left(\sum_{k=1}^{N_m^\mathrm{el}} \gamma_m \frac{\alpha_\mathrm{s} s_\mathrm{tol}^\mathrm{e}}{N^\mathrm{bk} N_m^\mathrm{el}} \right) \ge \frac{1}{\gamma_m} \left(\sum_{k=1}^{N_m^\mathrm{el}} \left| \eta_{m,k}^\mathrm{s} \right| \right) = \left| \eta_m^\mathrm{s} \right|.$$

The complete space-time adaptive strategy is summarized in algorithm 1.

5 Numerical Examples

5.1 Example 1: perforated plate under impulse loading

This example illustrates the performance of the proposed space-time adaptive strategy in a 2D wave propagation problem. The computational domain Ω is a perforated rectangular plate, $\Omega := (-0.5, 0.5) \times (0, 0.5) \setminus \Omega_0$ m^2, with $\Omega_0 := \{ (x, y) \in \mathbb{R}^2 : x^2 + (y - 0.25)^2 \le 0.025^2 \}$ m^2, see figure 5. The plate is clamped at the bottom side and the horizontal displacement is blocked at both

---

Data:
Problem statement: Problem geometry (Ω, Γ_N, Γ_D), final time (T), material data (E, ν, ρ), loads and initial conditions ($\mathbf{f}$, $\mathbf{g}$, $\mathbf{u}_0$, $\mathbf{v}_0$).
Problem discretization: background computational mesh ($\mathcal{P}_\mathrm{bg}$).
Error control: data defining the quantity of interest ($\mathbf{f}^\mathcal{O}$, $\mathbf{g}^\mathcal{O}$, $\mathbf{u}^\mathcal{O}$, $\mathbf{v}^\mathcal{O}$) and number of vibration modes M.
Adaptivty parameters: Number of time blocks (N^bk), prescribed error ($s_\mathrm{tol}^\mathrm{e}$), error splitting coefficients (α_s, α_t), unrefinement parameters (β_s, β_t).
Result: Numerical approximation $\tilde{\mathbf{U}}$ and error estimate $\tilde{s}^\mathrm{e}$ fulfilling $|\tilde{s}^\mathrm{e}| \le s_\mathrm{tol}^\mathrm{e}$.
`// Modal analysis`
Generate higher order space $\mathcal{V}_0^{H,p+1}(\mathcal{P}_\mathrm{bg})$;
Compute the eigenpairs ($\tilde{\omega}_i$, $\tilde{\mathbf{q}}_i$), $i = 1, \ldots, M$ in the space $\mathcal{V}_0^{H,p+1}(\mathcal{P}_\mathrm{bg})$;
`// Adjoint problem (modal solution)`
Compute the values $\tilde{l}_i$, $\tilde{u}_i$, $\tilde{v}_i$ (using $\mathbf{f}^\mathcal{O}$, $\mathbf{g}^\mathcal{O}$, $\mathbf{u}^\mathcal{O}$, $\mathbf{v}^\mathcal{O}$ and $\tilde{\mathbf{q}}_i$, $i = 1, \ldots, M$) ;
Compute the time dependent functions $\tilde{y}_i(t)$ (using $\tilde{l}_i$, $\tilde{u}_i$, $\tilde{v}_i$ and $\tilde{\omega}_i$, $i = 1, \ldots, M$) ;
`// Problem computation, error assessment and adaptivity`
Initialize discretization: $\mathcal{P}_1^\mathrm{bk} = \mathcal{P}_\mathrm{bg}$, $\Delta t_1^\mathrm{bk} = T/N^\mathrm{bk}$;
for $m = 1 \ldots N^\mathrm{bk}$ **do**
  repeat
    `// Compute solution and error estimate`
    Compute solution $\tilde{\mathbf{U}}$ in the time interval I_m^bk and the error indicators η_m^s, $\eta_{m,k}^\mathrm{s}$ and η_m^t;
    `// Mesh adaptivity`
    if _The acceptability criteria for η_m^s or η_m^t are not fulfilled_ **then**
      Refine/unrefine the spatial mesh $\mathcal{P}_m^\mathrm{bk}$ (using $\eta_{m,k}^\mathrm{s}$) and/or the time step Δt_m^bk (using η_m^t);
    end
  until _The acceptability criteria for η_m^s and η_m^t are fulfilled_ ;
  Set initial discretization for the next time interval: $\mathcal{P}_{m+1}^\mathrm{bk} = \mathcal{P}_m^\mathrm{bk}$, $\Delta t_{m+1}^\mathrm{bk} = \Delta t_m^\mathrm{bk}$;
end

Algorithm 1: Algorithm for problem approximation with error control and space-time mesh adaptivity.

---

vertical sides. The plate is initially at rest, $\mathbf{u}_0 = \mathbf{v}_0 = \mathbf{0}$, and loaded with the time dependent traction

$$\mathbf{g}(t) = \begin{cases} -g(t)\mathbf{e}_2 & \text{on } \Gamma_g, \\ 0 & \text{elsewhere,} \end{cases} \tag{30}$$

where $\Gamma_g := (-0.025, 0.025) \times \{0.5\}$ m, $\mathbf{e}_2 := (0, 1)$ and $g(t)$ is the impulsive time-dependent function defined in figure 5 with parameters $g_\mathrm{max} = 30$ Pa and $t_g = 0.005$ s. No body force is acting in this example, $\mathbf{f} = \mathbf{0}$. The material properties of the plate are Young's modulus $E = 8/3$ Pa, Poisson's ratio $\rho = 1/3$, the density $\rho = 1$ kg/m^3 and the damping coefficients $a_1 = 0$ s^{-1}, $a_2 = 10^{-4}$ s. The final simulation time is $T = 0.25$ s.
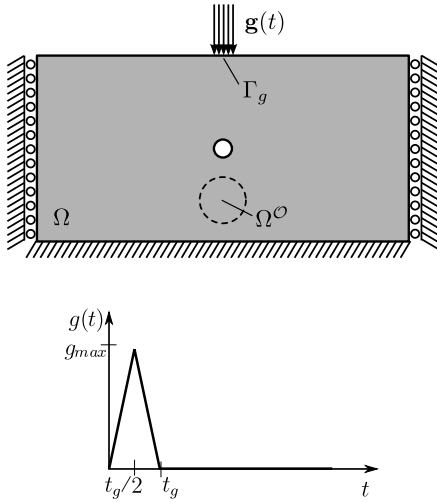
Fig. 5 Example 1: Definition of the problem geometry (top) and time-dependence of the external load (bottom).

The background mesh $\mathcal{P}_{\mathrm{bg}}$ for the quadtree remeshing strategy is plotted in figure 6. Note that only half of the domain is discretized due to the problem's symmetry by introducing proper symmetry boundary conditions. The finite element spaces $\boldsymbol{\mathcal{V}}_0^{H,1}(\mathcal{P}_n)$, $n = 1, \ldots, N$, used for solving the direct problem are build using bilinear elements (quadrilaterals with 4 nodes, i.e. $p = 1$) while the finite element space for the adjoint, $\boldsymbol{\mathcal{V}}_0^{H,2}(\mathcal{P}_n)$, is build using serendipity elements (quadrilaterals with 8 nodes, i.e. $p = 2$).
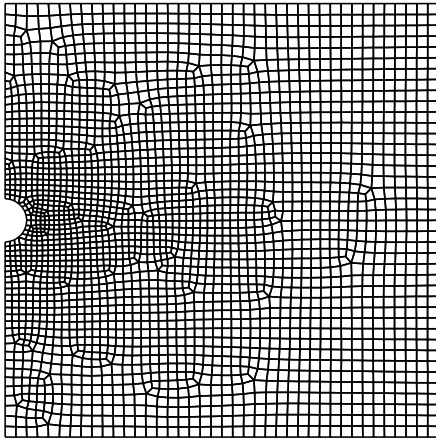
Fig. 6 Example 1: Background mesh $\mathcal{P}_{\mathrm{bg}}$ with 2452 elements for the quadtree remeshing strategy and for the adjoint problem approximation. Only half of the domain is discretized due to the problem's symmetry.

The quantity of interest considered in this example is a weighted average of the vertical velocities in the region

$$\Omega^{\mathcal{O}} := \{x, y \in \mathbb{R}^2 : \ x^2 + (y - 0.1)^2 < 0.075^2)\} \ \mathrm{m}^2,$$

see figure 5. Specifically, the quantity of interest is defined as

$$L^{\mathcal{O}}(\mathbf{W}) := m(\mathbf{v}^{\mathcal{O}}, \mathbf{w}_v(T)),$$

corresponding to $\mathbf{f}^{\mathcal{O}} = \mathbf{g}^{\mathcal{O}} = \mathbf{u}^{\mathcal{O}} = \mathbf{0}$ in (8). The weighting function $\mathbf{v}^{\mathcal{O}}$ with local support in $\Omega^{\mathcal{O}}$ is $\mathbf{v}^{\mathcal{O}} = [0, v_{\mathrm{aux}}(\sqrt{x^2 + (y - 0.1)^2})]$ for

$$v_{\mathrm{aux}}(r) = \begin{cases} \dfrac{10}{3\pi R^2 \rho} \left(2\left(\dfrac{r}{R} - 1\right)^3 + 3\left(\dfrac{r}{R} - 1\right)^2 \right) \\ \qquad\qquad\qquad\qquad \text{for } 0 \leq r \leq R, \\ 0 \quad \text{for } R < r, \end{cases}$$

$R = 0.075$ being the radius of the region of interest. Note that since the x-component of $\mathbf{v}^{\mathcal{O}}$ is zero, the quantity of interest gives an average of the vertical velocity in the region of interest $\Omega^{\mathcal{O}}$ and at time $t = T$.

The adjoint problem associated to the quantity of interest is approximated using a truncated modal based approximation where only the first 60 vibration modes are kept. This corresponds to slightly modify the quantity of interest of the problem. In the following, the function $\mathbf{v}^{\mathcal{O}}$ in the *exact* quantity of interest $L^{\mathcal{O}}(\mathbf{W}) = m(\mathbf{v}^{\mathcal{O}}, \mathbf{w}_v(T))$ is replaced by its projection onto the first $M = 60$ vibration modes $\tilde{\mathbf{q}}_i$, $i = 1, \ldots, M$, namely

$$\mathbf{v}^{\mathcal{O},M}(\mathbf{x}) := \sum_{i=1}^{M} v_i \tilde{\mathbf{q}}_i(\mathbf{x}), \quad \text{where} \quad v_i := m(\mathbf{v}^{\mathcal{O}}, \tilde{\mathbf{q}}_i).$$

Figure 7 shows that the truncated discrete approximation $\mathbf{v}^{\mathcal{O},M}$ provides a fairly good approximation of the exact weighting function $\mathbf{v}^{\mathcal{O}}$. It is worth noting that the quantity of interest is no longer strictly measuring only the vertical velocity of the solution and has no longer a local support. However, as can bee seen, the influence of the horizontal velocity and the average outside $\Omega^{\mathcal{O}}$ are small.

The exact solution $\mathbf{U}$ (and therefore the exact quantity of interest s) are unknown in this example. The exact solution is replaced here by an overkill approximation of the problem, namely $\mathbf{U}^{\mathrm{ovk}}$, computed with a finite elements mesh with $N^{\mathrm{el}} = 627712$ elements and $N = 1600$ time steps. The overkill discretization is finest discretization considered in this example. The exact value of the quantity of interest is approximated using the overkill approximation, $s \approx s^{\mathrm{ovk}} := L^{\mathcal{O}}(\mathbf{U}^{\mathrm{ovk}}) = 2.4227 \cdot 10^{-2}$ m/s.

The behavior of the adaptive strategy is first analyzed for a prescribed target error $s_{\mathrm{tol}}^{\mathrm{e}} = 5 \cdot 10^{-5}$ m/s. The user-prescribed parameters for the simulation are set to $N^{\mathrm{bk}} = 20$ for the number of space-time blocks, $\alpha_{\mathrm{s}} = 0.9$ and $\alpha_{\mathrm{t}} = 0.1$ for the coefficients used to split the total error budget into space and time and $\beta_{\mathrm{s}} = 0.5$ and $\beta_{\mathrm{t}} = 0.1$ for the lower bound factors.
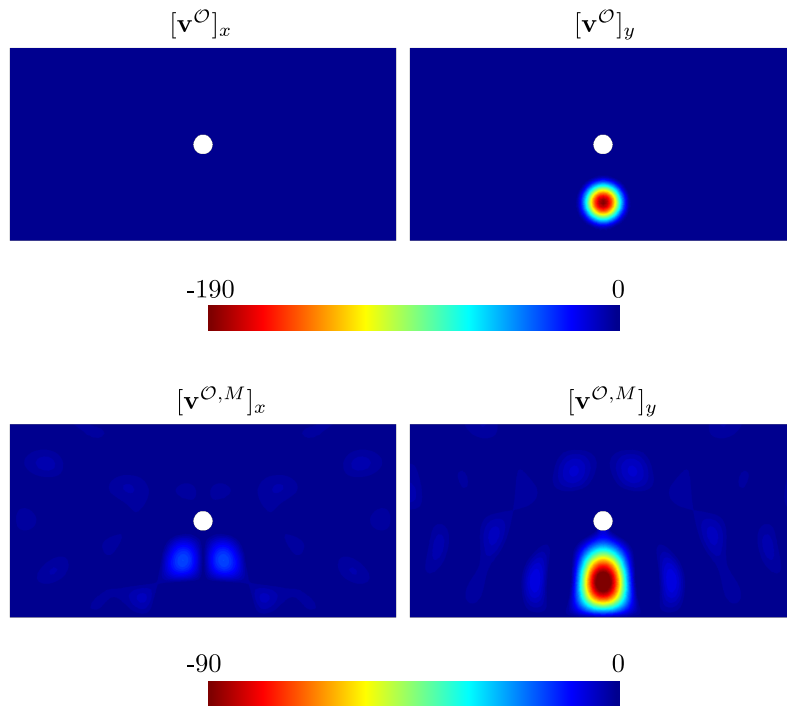
Fig. 7 Example 1: Exact (top) and truncated (bottom) weighting functions $\mathbf{v}^{\mathcal{O}}$ defining the quantity of interest $L^{\mathcal{O}}(\cdot)$.

Figure 8 shows several snapshots of an adapted numerical solution obtained with the proposed methodology. The quantity of interest associated with the numerical solution is $\tilde{s} := L^{\mathcal{O}}(\widetilde{\mathbf{U}}) = 2.4242 \cdot 10^{-2}$ m/s with an assessed error of $\tilde{s}^{\mathrm{e}} = -1.5756 \cdot 10^{-5}$ m/s. Thus the prescribed target error $s_{\mathrm{tol}}^{\mathrm{e}} = 5 \cdot 10^{-5}$ m/s is fulfilled quite sharply, that is, $|\tilde{s}^{\mathrm{e}}| \leq s_{\mathrm{tol}}^{\mathrm{e}}$, and $|\tilde{s}^{\mathrm{e}}|$ and $s_{\mathrm{tol}}^{\mathrm{e}}$ are of the same order of magnitude. Moreover, the error with respect the overkill solution, namely $s_{\mathrm{ovk}}^{\mathrm{e}} := s^{\mathrm{ovk}} - \tilde{s} = -1.5125 \cdot 10^{-5}$ m/s, is also below (in absolute value) the user-defined value $s_{\mathrm{tol}}^{\mathrm{e}}$. Note that the assessed error is a good approximation of the overkill error. That is, the effectivity of the error estimate, $\tilde{s}^{\mathrm{e}}/s_{\mathrm{ovk}}^{\mathrm{e}} = 1.041$, is fairly close to the unity.

Figure 9 shows the history of the number of elements and the time step length along the adapted computation. Note that the number of elements increases with time as the elastic waves spread along the plate and therefore a larger area has to be refined. The time step is refined only when the external load is acting at the beginning of the computation. Additionally, figure 9 also shows the number of iterations performed in each space-time block until reaching convergence. As can be seen, convergence is reach for the whole computation with at most four iterations per block.

The performance of the space-time adaptive strategy is also tested versus a uniform refinement. Three non-adapted (uniform) approximations are computed using three different space meshes and three different number of time steps N, see table 1. The initial space mesh corresponds to the background mesh showed in figure 6 which is recursively refined to obtain the other spatial meshes (each quadrilateral element is recursively subdivided into four new ones). The ratio $H/\Delta t$, or equivalently the ratio $N/(N^{\mathrm{el}})^{\frac{1}{2}}$, is kept constant in the three uniform approximations. This is to ensure that the space and time errors are reduced at the same ratio taking into account that the space discretization error scales as H^2 and the time discretization error as Δt^2, see [9] and [2].

Table 1 Example 1: Space and time discretizations for the three uniform solutions.

	N^{el}	# nodes	N
1	2452	2547	100
2	9808	9997	200
3	39232	39609	400

On the other hand, the space-time adaptive computations are performed prescribing similar total target errors as the errors obtained using uniform refinements. Specifically, four different simulations are performed setting $s_{\mathrm{tol}}^{\mathrm{e}} = 1 \cdot 10^{-3}$, $5 \cdot 10^{-4}$, $1 \cdot 10^{-4}$ and $5 \cdot 10^{-5}$ m/s combined with three different values for the
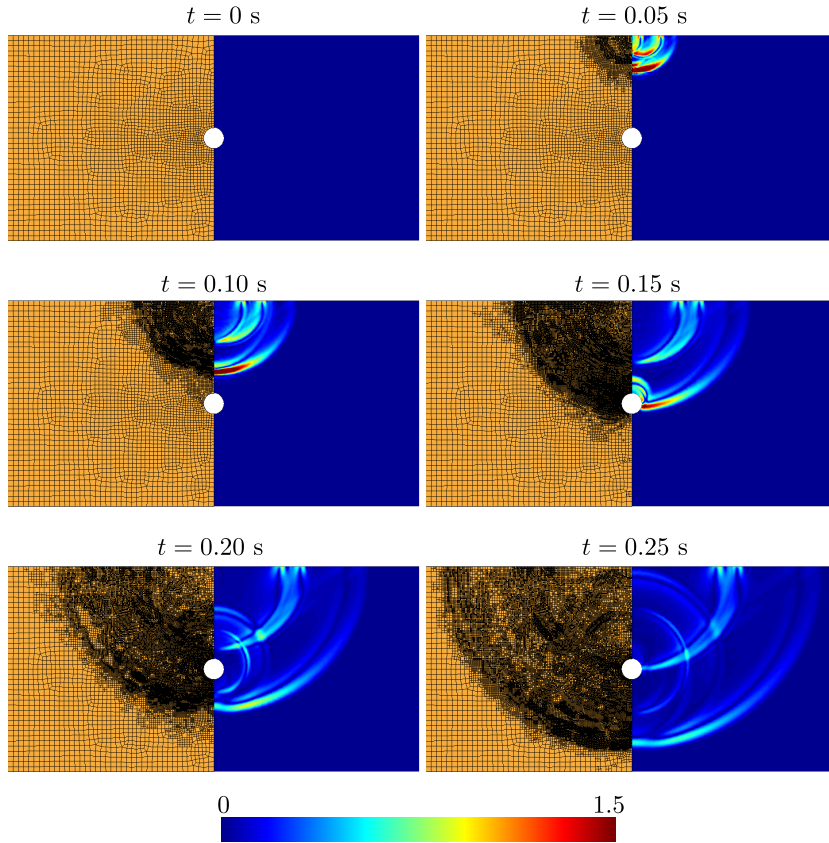
Fig. 8 Example 1: Snapshots of the computed solution (magnitude of velocities in m/s) and the computational mesh at several time points for the adapted solution verifying the prescribed target error $s_{\text{tol}}^{\text{e}} = 5 \cdot 10^{-5}$ m/s.

number of blocks, $N^{\text{bk}} = 5$, 10 and 20. The additional parameters of the adaptive procedure are $\alpha_{\text{s}} = 0.9$, $\beta_{\text{s}} = 0.5$ and $\alpha_{\text{t}} = \beta_{\text{t}} = 0.1$. The computational complexity of the simulations is measured here using the number of space-time elements (or cells), namely

$$N^{\text{cells}} := \sum_{m=1}^{N^{\text{bk}}} N_m^{\text{el}} \frac{T}{N^{\text{bk}} \Delta t_m^{\text{bk}}},$$

corresponding to sum up the number of space-time elements used inside each time interval $I_m^{\text{bk}}, m = 1, \ldots, N^{\text{bk}}$. Note that if a single space mesh is considered in the whole simulation time, then the number of space-time cells N^{cells} coincides with $N^{\text{cells}} = N^{\text{el}} N$.

Figure 10 shows the convergence of the estimates. The estimates obtained for the uniform refinement meet the expected a-priori convergence rate of $-2/3$. This expected convergence rate is obtained considering the a-priori estimates of the error $s^{\text{e}} \propto H^2 + (\Delta t)^2$, the relation $N^{\text{cells}} \propto (H^2 \Delta t)^{-1}$ and noting that if the ratio $H/\Delta t$ is constant, then H and Δt can be written as $H = \kappa H^{\star}$ and $\Delta t = \kappa \Delta t^{\star}$, where $H^{\star}$ and $\Delta t^{\star}$ are the element and step length of the coarsest uniform discretization and κ is a refinement factor. It is then straightfor-

ward that, $s^{\text{e}} \propto (N^{\text{cells}})^{-2/3}$ since $(H^2 \Delta t)^{2/3} = C(H^2 + (\Delta t)^2) \propto s^{\text{e}}$ for $C = ((H^*)^2 \Delta t^*)^{2/3})/((H^*)^2 + (\Delta t^*)^2)$. From figure 10 and table 2 it can be seen that besides converging at the correct convergence rate, the estimates are really accurate since their effectivities are very close to 1.

As expected, the use of an adaptive refinement strategy leads to better approximations for the quantity of interest with less computational cost. The adapted solutions have a lower error than the uniform approximations for the same number of space-time cells.

5.2 Example 2: 2D structure

Consider the structure given in figure 11. The structure is initially at rest ($\mathbf{u}_0 = \mathbf{v}_0 = \mathbf{0}$), clamped at the supports and subjected to the time-dependent traction

$$\mathbf{g} = \begin{cases} g(t)\mathbf{e}_1 & \text{on } \Gamma_g, \\ 0 & \text{elsewhere.} \end{cases}$$

The set Γ_g is the region of the Neumann boundary where the load is applied, vector $\mathbf{e}_1 := (1, 0)$ is the first cartesian unit vector and function $g(t)$ describes

Table 2 Example 1: Performance of the estimate for both the uniform and adaptive strategies. The overkill value of the quantity of interest is $s^{\mathrm{ovk}} = 2.4227 \cdot 10^{-2}$ m/s obtained with $N^{\mathrm{cell}} = 1004339200$ space-time elements.

	$s_{\mathrm{tol}}^{\mathrm{e}}$ [m/s]	N^{cell}	$\tilde{s}$ [m/s]	$\tilde{s}^{\mathrm{e}}$ [m/s]	$s_{\mathrm{ovk}}^{\mathrm{e}}$ [m/s]	$\tilde{s}^{\mathrm{e}}/s_{\mathrm{ovk}}^{\mathrm{e}}$
uniform	–	245200	$2.4498\cdot10^{-2}$	$-2.7186\cdot10^{-4}$	$-2.7180\cdot10^{-4}$	1.000
	–	1961600	$2.4299\cdot10^{-2}$	$-7.1606\cdot10^{-5}$	$-7.2345\cdot10^{-5}$	0.989
	–	15692800	$2.4244\cdot10^{-2}$	$-1.7813\cdot10^{-5}$	$-1.7659\cdot10^{-5}$	1.008
$N^{\mathrm{bk}} = 5$	$1\cdot10^{-3}$	220680	$2.4498\cdot10^{-2}$	$-2.7186\cdot10^{-4}$	$-2.7180\cdot10^{-4}$	1.000
	$5\cdot10^{-4}$	499920	$2.4391\cdot10^{-2}$	$-1.6337\cdot10^{-4}$	$-1.6403\cdot10^{-4}$	0.996
	$1\cdot10^{-4}$	2211720	$2.4261\cdot10^{-2}$	$-3.3703\cdot10^{-5}$	$-3.4096\cdot10^{-5}$	0.988
	$5\cdot10^{-5}$	5511720	$2.4236\cdot10^{-2}$	$-1.0823\cdot10^{-5}$	$-8.9160\cdot10^{-6}$	1.213
$N^{\mathrm{bk}} = 10$	$1\cdot10^{-3}$	245200	$2.4498\cdot10^{-2}$	$-2.7186\cdot10^{-4}$	$-2.7180\cdot10^{-4}$	1.000
	$5\cdot10^{-4}$	391280	$2.4313\cdot10^{-2}$	$-8.6724\cdot10^{-5}$	$-8.6226\cdot10^{-5}$	1.005
	$1\cdot10^{-4}$	5158120	$2.4251\cdot10^{-2}$	$-2.4351\cdot10^{-5}$	$-2.4455\cdot10^{-5}$	0.995
	$5\cdot10^{-5}$	7074440	$2.4244\cdot10^{-2}$	$-1.5773\cdot10^{-5}$	$-1.7111\cdot10^{-5}$	0.921
$N^{\mathrm{bk}} = 20$	$1\cdot10^{-3}$	279900	$2.4439\cdot10^{-2}$	$-2.1096\cdot10^{-4}$	$-2.1219\cdot10^{-4}$	0.994
	$5\cdot10^{-4}$	462735	$2.4351\cdot10^{-2}$	$-1.2062\cdot10^{-4}$	$-1.2446\cdot10^{-4}$	0.969
	$1\cdot10^{-4}$	6732720	$2.4261\cdot10^{-2}$	$-3.6194\cdot10^{-5}$	$-3.4268\cdot10^{-5}$	1.056
	$5\cdot10^{-5}$	9080750	$2.4242\cdot10^{-2}$	$-1.5756\cdot10^{-5}$	$-1.5125\cdot10^{-5}$	1.041
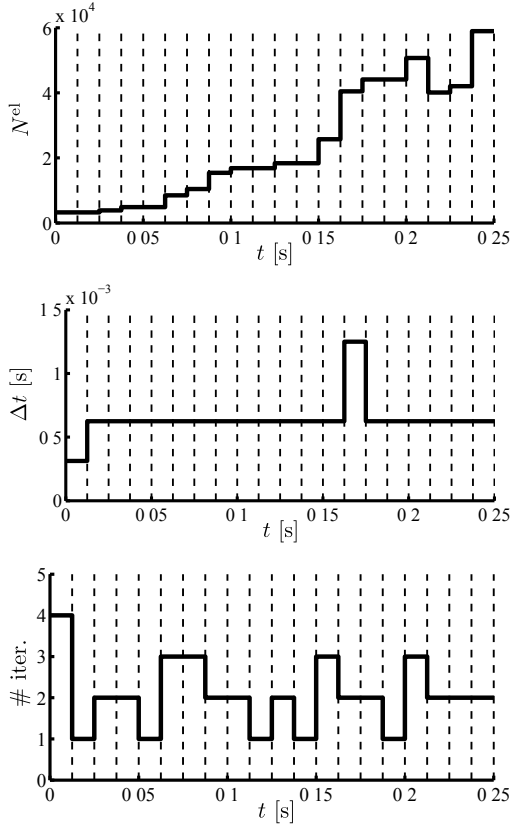
Fig. 9 Example 1: History of the number of elements (top) and of the time step (middle). Number of iterations to achieve convergence in each block (bottom).

Fig. 10 Example 1: Error convergence for the adapted and uniform computations. The adapted solutions are obtained using three different values of the number of time blocks N^{bk}.

This example focusses in the quantity of interest

$$L^{\mathcal{O}}(\mathbf{W}) := \frac{1}{\mathrm{meas}(\Gamma_g)}(\mathbf{e}_1, \mathbf{w}_u(T))_{\Gamma_g}, \tag{31}$$

which is the average of the final displacement in the boundary Γ_g where the external load is applied. Note that this quantity is not accounted in the generic quantity of interest given in equation (10). Consequently, quantity (31) is rewritten as

$$L^{\mathcal{O}}(\mathbf{W}) = a(\mathbf{u}^{\mathcal{O}}, \mathbf{w}_u(T)),$$

the time evolution of $\mathbf{g}$ given in figure 11. The traction $\mathbf{g}$ is the only external loading in this example (that is $\mathbf{f} = \mathbf{0}$). Other material and geometric parameters univocally defining the problem are reported in table 3.
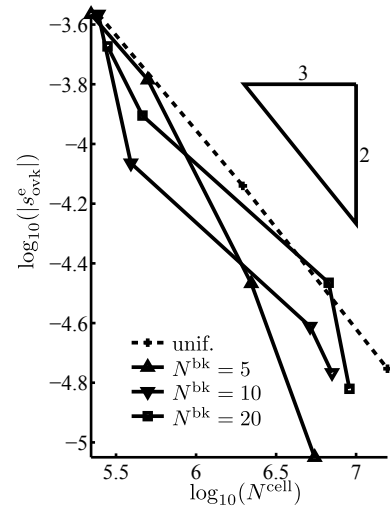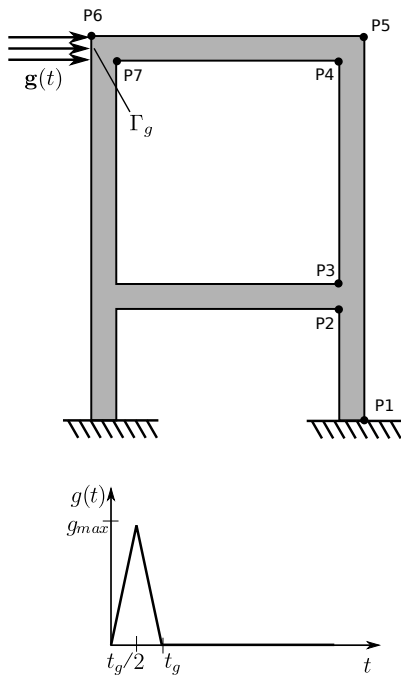
Fig. 11 Example 2: Problem statement (top) and time dependent loading at Γ_g (bottom).

Table 3 Example 2: Problem parameterization

Geometry (data in m)	Physical properties
$P_1 := (0.55, 0.00)$	$E = 2 \cdot 10^{11}$ Pa
$P_2 := (0.45, 0.45)$	$\nu = 0.2$
$P_3 := (0.45, 0.55)$	$\rho = 8 \cdot 10^3$ kg/m^3
$P_4 := (0.45, 1.45)$	$a_1 = 0$ s^{-1}
$P_5 := (0.55, 1.55)$	$a_2 = 1 \cdot 10^{-5}$ s
$P_6 := (-0.55, 1.55)$	$T = 2 \cdot 10^{-3}$ s
$P_7 := (-0.45, 1.45)$	
$\Gamma_g := \{-0.55\} \times (1.45, 1.55)$	

External load
$g_{\max} = 10^8$ Pa
$t_g = 2 \cdot 10^{-4}$ s

where $\mathbf{u}^{\mathcal{O}}$ is the *exact* solution of the static linear elasticity problem: find $\mathbf{u}^{\mathcal{O}} \in \boldsymbol{\mathcal{V}}_0$ such that

$$a(\mathbf{u}^{\mathcal{O}}, \mathbf{w}) = \frac{1}{\text{meas}(\Gamma_g)}(\mathbf{e}_1, \mathbf{w})_{\Gamma_g}, \quad \forall \mathbf{w} \in \boldsymbol{\mathcal{V}}_0. \quad (32)$$

After this reformulation, the quantity of interest is a particular case of the ones included in (10) and therefore the associated adjoint problem has the same structure as the original one. In particular, the function $\mathbf{u}^{\mathcal{O}}$ is the final displacement condition for the adjoint problem. The other forcing data for the adjoint are zero in this case, namely $\mathbf{v}^{\mathcal{O}} = \mathbf{f}^{\mathcal{O}} = \mathbf{g}^{\mathcal{O}} = \mathbf{0}$. Note that function $\mathbf{u}^{\mathcal{O}}$ is solution of an infinite dimensional
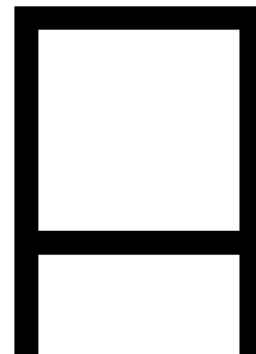
problem and therefore it is unknown. In this example, the unknown function $\mathbf{u}^{\mathcal{O}}$ is replaced by the computable one $\tilde{\mathbf{u}}^{\mathcal{O}}$ obtained by solving problem (32) in the discrete space $\boldsymbol{\mathcal{V}}_0^{H,p+1}(\mathcal{P}_{\text{bg}})$ associated with the background mesh of the adaptive process. Three different background meshes are used in this example, see figure 12.
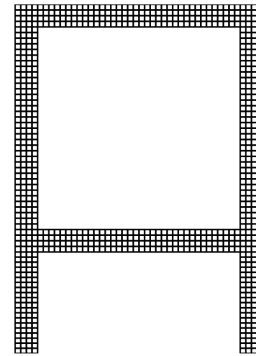
Background mesh 1
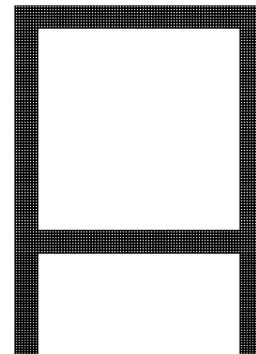
Background mesh 2

Background mesh 3

Fig. 12 Example 2: Background meshes used in this example. The number of elements in each of them is 800, 3200 and 12800 respectively.

The quantity of interest (31) is well suited for the modal based approach becasue the weighting function $\tilde{\mathbf{u}}^{\mathcal{O}}$ is well captured by the expansion of few eigenvectors. This ensures that the adjoint solution is also prop-

erly represented using few vibration modes. The projection of $\tilde{\mathbf{u}}^{\mathcal{O}}$ into the expansion of the first M eigenvectors is defined as

$$\tilde{\mathbf{u}}^{\mathcal{O},M} := \sum_{i=1}^{M} \tilde{\mathbf{u}}_i \tilde{\mathbf{q}}_i,$$

where $\tilde{u}_i := m(\tilde{\mathbf{u}}^{\mathcal{O}}, \tilde{\mathbf{q}}_i)$, $i = 1, \ldots, M$. Thus, the relative error in the projection is

$$\varepsilon^M := \frac{\|\tilde{\mathbf{u}}^{\mathcal{O}} - \tilde{\mathbf{u}}^{\mathcal{O},M}\|_m}{\|\tilde{\mathbf{u}}^{\mathcal{O}}\|_m},$$

where $\|\cdot\|_m := (m(\cdot,\cdot))^{1/2}$. Figure 13 shows the error ε^M as a function of the number of eigenvectors M. Note that the error ε^M rapidly decreases as M increases. The number of eigenvectors considered in this example is $M = 60$ and the associated projection error is $\varepsilon^{60} = 5.94 \cdot 10^{-5}$.
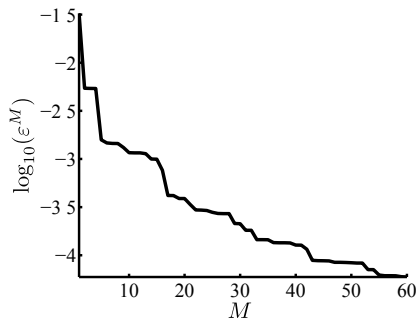
Fig. 13 Example 2: Error in projecting the weighting function $\tilde{\mathbf{u}}$ into the expansion of the first M eigenvectors $\tilde{\mathbf{q}}_1, \ldots, \tilde{\mathbf{q}}_M$. The eigenvectors $\tilde{\mathbf{q}}_i$ and the weighting function $\tilde{\mathbf{u}}^{\mathcal{O}}$ are computed in the space $\mathcal{V}_0^{H,p+1}(\mathcal{P}_{\mathrm{bg}})$ associated with the background mesh number 2 plotted in figure 12.

The exact value of the quantity of interest is unknown in this example. An overkill approximation of the quantity of interest, $s^{\mathrm{ovk}} := 1.2086 \cdot 10^{-3}$ m, is computed using a finite element mesh of $N^{\mathrm{el}} = 204800$ elements and $N = 6400$ time steps. This discretization is the richest one considered in this example.

Figures 14 and 15 show snapshots of the computed solution and the computational mesh at several time points. This particular solution is obtained using the background mesh number 2, taking $N^{\mathrm{bk}} = 10$ time blocks and prescribing the error to the value $s^{\mathrm{e}}_{\mathrm{tol}} = 5 \cdot 10^{-6}$ m. The coefficients used to split the total error budget into space and time are $\alpha_{\mathrm{s}} = 0.9$ and $\alpha_{\mathrm{t}} = 0.1$ and the unrefinement factors are taken as $\beta_{\mathrm{s}} = 0.5$ and $\beta_{\mathrm{t}} = 0.1$. The computed quantity of interest is $\tilde{s} = 1.2069 \cdot 10^{-3}$ m and the assessed error is $\tilde{s}^{\mathrm{e}} = 8.8942 \cdot 10^{-7}$ m. Note that the restriction $|\tilde{s}^{\mathrm{e}}| \leq s^{\mathrm{e}}_{\mathrm{user}}$ is also fulfilled in this example. Moreover, the error with

respect the overkill solution, $s^{\mathrm{e}}_{\mathrm{ovk}} = 1.7516 \cdot 10^{-6}$ m, is also below the user-defined value $s^{\mathrm{e}}_{\mathrm{tol}}$.

Figure 16 shows the history of number of elements in the computational mesh and the time step length for this particular computation. Note that the number of mesh elements increases in time because the stress waves spread in the structure. Note also that the time step length is smaller at the beginning of the computation due to the effect of the external load acting at the initial simulation time. Figure 16 also shows the number of iterations until achieve convergence in each time block. Note that the number of iterations is always equal or less than four.
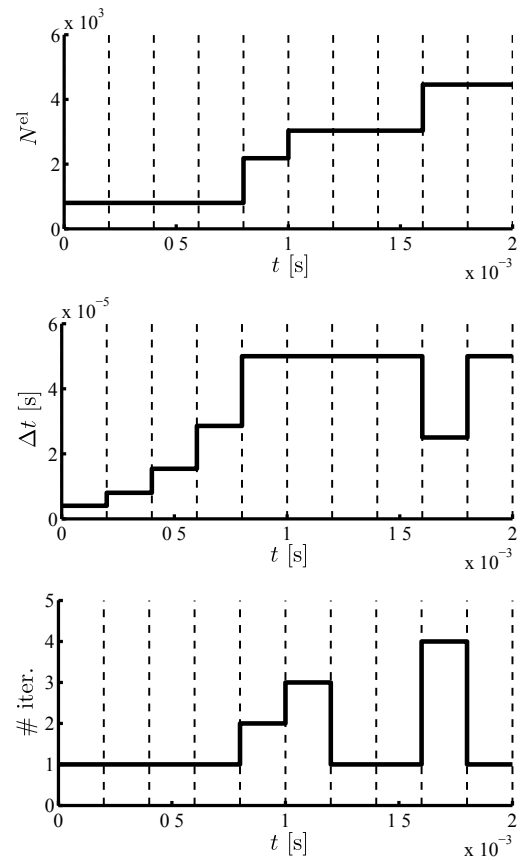
Fig. 16 Example 2: Evolution along the adaptive process of the number of elements (top) and the time step (center). Number of remeshing iterations to achieve convergence in each block (bottom).

The performance of the adaptive strategy is compared with respect to uniform mesh refinement. To this end, the uniform refined computations are obtained using the meshes plotted in figure 12 and three different number of time steps N, see table 4. Note that the ratio $H/\Delta t$ is also kept constant in this example to ensure that the space and time errors are reduced at the same
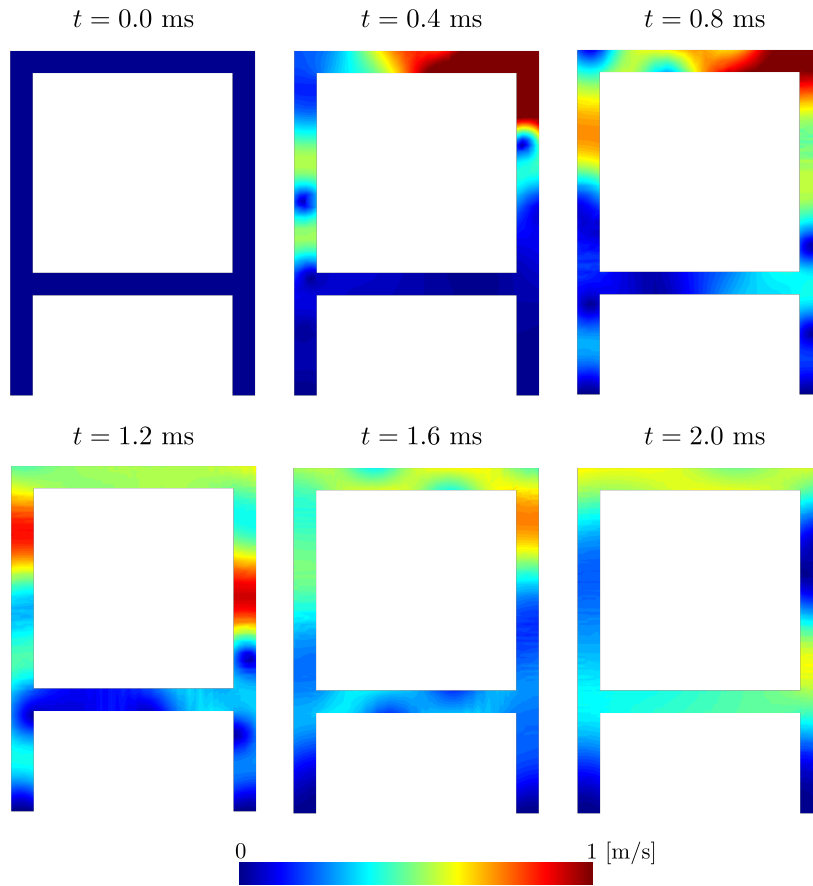
$t = 0.0$ ms $t = 0.4$ ms $t = 0.8$ ms

$t = 1.2$ ms $t = 1.6$ ms $t = 2.0$ ms

0   1 [m/s]

Fig. 14 Example 2: Snapshots of the computed solution (magnitude of velocities in m/s) at several time points.

rate. On the other hand, the adapted solutions are obtained using $N^{\mathrm{bk}} = 10$ and four different values of the prescriber error, $s^{\mathrm{e}}_{\mathrm{tol}} = 5 \cdot 10^{-5}$, $1 \cdot 10^{-5}$, $5 \cdot 10^{-6}$ and $1 \cdot 10^{-6}$ m. The dependence of the results on the chosen background mesh is studied by computing the adaptive solutions using the three background meshes plotted in figure 12. Twelve adaptive solutions are computed all together (one for each value of the prescribed error and one for each background mesh).

Table 4 Example 2: Space and time discretizations for the three uniform solutions.

	N^{el}	# nodes	N
1	200	300	200
2	800	1000	400
3	3200	3600	800
4	12800	13600	1600

Table 5 and figure 17 and give the results for the adaptive and non-adaptive solutions. The convergence curves in figure 17 shows that the adapted solutions achieve a smaller error than the non-adapted solutions

for the same number of space-time elements N^{cells}. The effectivity of the error estimate, namely $\tilde{s}^{\mathrm{e}}/s^{\mathrm{e}}$, is also shown in figure 17. Note that the computed effectivity (i.e. the quality of the error estimate) is better the finer is the background mesh. This is because the adjoint problem and the extractor $\tilde{\mathbf{u}}^{\mathcal{O}}$ are computed using the background mesh. Thus, the finer the background mesh, the better the quality of the adjoint and, therefore, the better the quality of the error estimate. Note that the computed effectivities in this example are slightly worse than the ones obtained in the first numerical example. Even though, the adaptive computations give more accurate results than the non-adapted solutions for the same number of space-time elements.

6 Closure

This article presents a goal-oriented space-time adaptive methodology for linear elastodynamics. The strategy aims at computing an optimal space-time discretization such that the numerical solution has an error in the quantity of interest below some user-defined tolerance. The space-time adaptation is driven by a goal-oriented
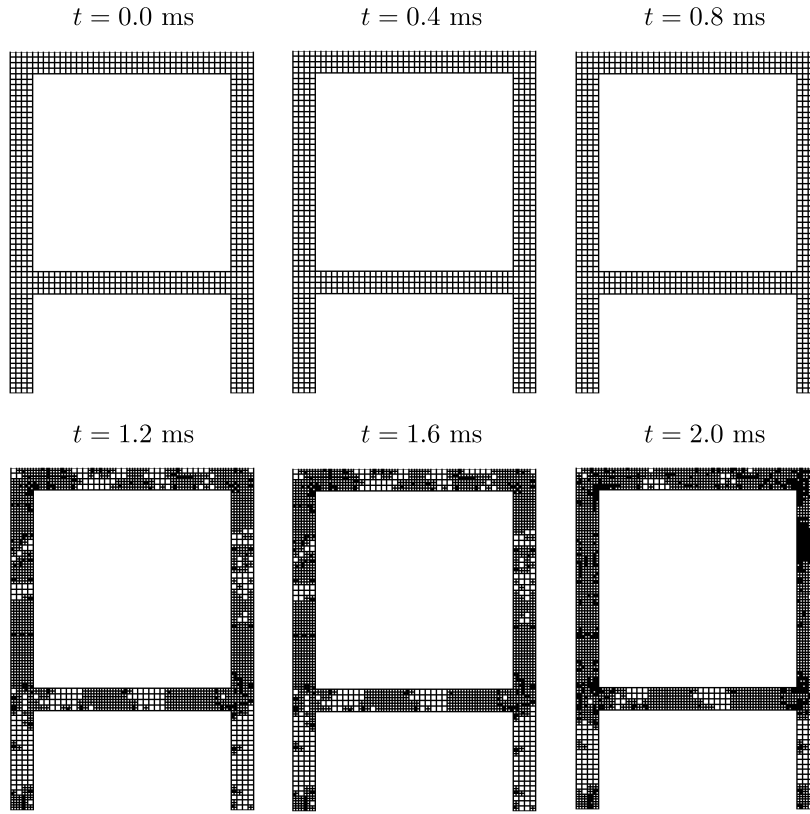
$t = 0.0$ ms $\qquad$ $t = 0.4$ ms $\qquad$ $t = 0.8$ ms

$t = 1.2$ ms $\qquad$ $t = 1.6$ ms $\qquad$ $t = 2.0$ ms

Fig. 15 Example 2: Snapshots of the computational mesh at several time points.

Table 5 Example 2: Performance of the estimate for both the uniform and adaptive strategies (for four different background meshes). The overkill value for the quantity of interest $s^{\text{ovk}} = 1.2086 \cdot 10^{-3}$ m is obtained using a uniform spatial mesh of $N^{\text{cell}} = 1310720000$ space-time elements.

|	$s^{\text{e}}_{\text{tol}}$ [m]	N^{cell}	$\tilde{s}$ [m]	$\tilde{s}^{\text{e}}$ [m]	$s^{\text{e}}_{\text{ovk}}$ [m]	$\tilde{s}^{\text{e}}/s^{\text{e}}_{\text{ovk}}$
uniform	–	40000	$1.1914 \cdot 10^{-3}$	$1.4558 \cdot 10^{-5}$	$1.7262 \cdot 10^{-5}$	0.843
	–	320000	$1.2037 \cdot 10^{-3}$	$3.6897 \cdot 10^{-6}$	$4.8973 \cdot 10^{-6}$	0.753
	–	2560000	$1.2068 \cdot 10^{-3}$	$1.3679 \cdot 10^{-6}$	$1.8601 \cdot 10^{-6}$	0.735
	–	20480000	$1.2079 \cdot 10^{-3}$	$5.6641 \cdot 10^{-7}$	$7.0564 \cdot 10^{-7}$	0.802
bg. mesh 1	$5 \cdot 10^{-5}$	28696	$1.2048 \cdot 10^{-3}$	$1.4506 \cdot 10^{-6}$	$3.8845 \cdot 10^{-6}$	0.373
	$1 \cdot 10^{-5}$	86181	$1.2068 \cdot 10^{-3}$	$3.0110 \cdot 10^{-7}$	$1.8337 \cdot 10^{-6}$	0.164
	$5 \cdot 10^{-6}$	153536	$1.2067 \cdot 10^{-3}$	$4.2316 \cdot 10^{-7}$	$1.9152 \cdot 10^{-6}$	0.220
	$1 \cdot 10^{-6}$	239936	$1.2067 \cdot 10^{-3}$	$5.8772 \cdot 10^{-7}$	$1.9421 \cdot 10^{-6}$	0.302
bg. mesh 2	$5 \cdot 10^{-5}$	90400	$1.2025 \cdot 10^{-3}$	$4.9028 \cdot 10^{-6}$	$6.1079 \cdot 10^{-6}$	0.802
	$1 \cdot 10^{-5}$	113004	$1.2066 \cdot 10^{-3}$	$1.0330 \cdot 10^{-6}$	$2.0820 \cdot 10^{-6}$	0.496
	$5 \cdot 10^{-6}$	174672	$1.2069 \cdot 10^{-3}$	$8.8942 \cdot 10^{-7}$	$1.7516 \cdot 10^{-6}$	0.507
	$1 \cdot 10^{-6}$	1212673	$1.2079 \cdot 10^{-3}$	$1.5956 \cdot 10^{-7}$	$7.4439 \cdot 10^{-7}$	0.214
bg. mesh 3	$5 \cdot 10^{-5}$	368000	$1.2056 \cdot 10^{-3}$	$2.5760 \cdot 10^{-6}$	$3.0675 \cdot 10^{-6}$	0.839
	$1 \cdot 10^{-5}$	380800	$1.2063 \cdot 10^{-3}$	$1.8447 \cdot 10^{-6}$	$2.3364 \cdot 10^{-6}$	0.789
	$5 \cdot 10^{-6}$	435724	$1.2071 \cdot 10^{-3}$	$1.3130 \cdot 10^{-6}$	$1.5426 \cdot 10^{-6}$	0.851
	$1 \cdot 10^{-6}$	3024438	$1.2083 \cdot 10^{-3}$	$1.5152 \cdot 10^{-7}$	$3.7024 \cdot 10^{-7}$	0.409
bg. mesh 4	$5 \cdot 10^{-5}$	1472000	$1.2065 \cdot 10^{-3}$	$1.9816 \cdot 10^{-6}$	$2.1207 \cdot 10^{-6}$	0.934
	$1 \cdot 10^{-5}$	1523200	$1.2073 \cdot 10^{-3}$	$1.1698 \cdot 10^{-6}$	$1.3089 \cdot 10^{-6}$	0.893
	$5 \cdot 10^{-6}$	1676800	$1.2077 \cdot 10^{-3}$	$7.8304 \cdot 10^{-7}$	$9.2219 \cdot 10^{-7}$	0.849
	$1 \cdot 10^{-6}$	4461564	$1.2084 \cdot 10^{-3}$	$1.8781 \cdot 10^{-7}$	$2.2230 \cdot 10^{-7}$	0.844
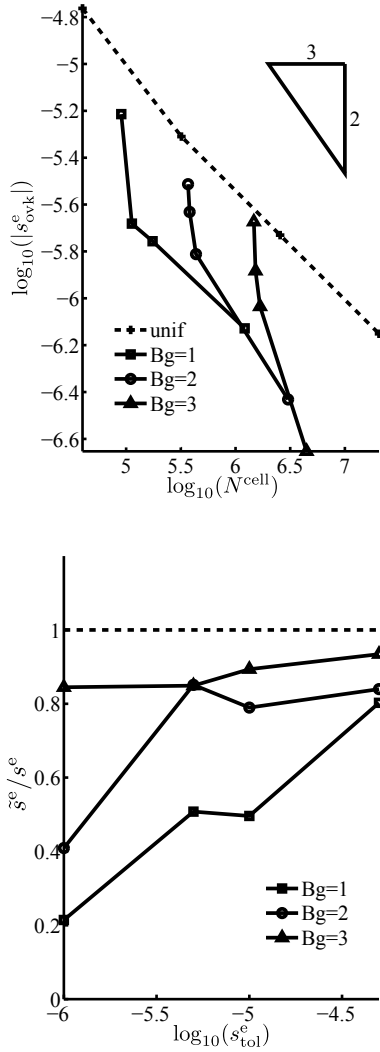
Fig. 17 Example 2: Error convergence for the adapted and uniform computations (top) and computed effectivity of the error estiamte (bottom). The adapted solutions are obtained using three background meshes.

error estimate that requires approximating an auxiliary adjoint problem.

The major novelty of this work is computing the adjoint solution with modal analysis instead of the standard direct time-integration methods. The modal-based approach is particularly efficient for some quantities of interest, because it allows to efficiently compute and store the adjoint solution.

The numerical examples show that the proposed strategy furnishes adapted solutions fulfilling the user-defined error tolerance. That is, both the assessed and computed errors are below the user-defined error value. Moreover, the discretizations obtained with the proposed adaptive strategy are more efficient than the ones obtained with a uniform refinement of all mesh elements

and time steps. The adaptive discretizations provide more accurate results than uniform remeshing, for the same number of space-time elements.

The proposed error estimate accounts for both the space and time discretization errors. The global error estimate is split into two contributions corresponding to the space and time errors using the Galerkin orthogonality property of the residual. This applies for space-time finite elements like time-continuous Galerkin methods. The extension of the approach to tackle other time-integration schemes, e.g. the ones based on finite differences and/or explicit methods with lumped mass matrix, requires further investigation.

Acknowledgment

A Linear system to be solved at each time step

This appendix details how the time-continuous Galerkin approximation is computed when the space mesh changes between times slabs.

Recall that the numerical approximation $\widetilde{\mathbf{U}}$ solution of the discrete problem (5) is computed sequentially starting from the first time slab I_1 until the last one I_N. Specifically, assuming that the solution at the time-slab I_{n-1} is known, the approximation $\widetilde{\mathbf{U}}$ restricted to the slab I_n is found solving the problem: find $\widetilde{\mathbf{U}}|_{I_n} \in \mathcal{W}_u^{H,\Delta t}|_{I_n} \times \mathcal{W}_v^{H,\Delta t}|_{I_n}$ such that

$$\int_{I_n} \left[m(\dot{\tilde{\mathbf{u}}}_v + a_1 \tilde{\mathbf{u}}_v, \mathbf{w}_v) + a(\tilde{\mathbf{u}}_u + a_2 \tilde{\mathbf{u}}_v, \mathbf{w}_v) \right] \, \mathrm{d}t$$
$$= \int_{I_n} l(t; \mathbf{w}_v) \, \mathrm{d}t, \quad \forall \mathbf{w}_v \in \mathcal{V}_0^H(\mathcal{P}_n), \quad (33\mathrm{a})$$

$$\int_{I_n} a(\dot{\tilde{\mathbf{u}}}_u - \tilde{\mathbf{u}}_v, \mathbf{w}_u) \, \mathrm{d}t = 0, \quad \forall \mathbf{w}_u \in \mathcal{V}_0^H(\mathcal{P}_n), \quad (33\mathrm{b})$$

$$\widetilde{\mathbf{U}}(t_{n-1}^+) = \widetilde{\mathbf{U}}(t_{n-1}), \quad (33\mathrm{c})$$

where, for $n > 1$, $\widetilde{\mathbf{U}}(t_{n-1})$ is the solution at the end of the previous interval I_{n-1} and, for $n = 1$, $\widetilde{\mathbf{U}}(t_{n-1} = t_0)$ is defined using the initial conditions, $\widetilde{\mathbf{U}}(t_0) = [\mathbf{u}_0, \mathbf{v}_0]$.

From the definition of the discrete spaces $\mathcal{W}_u^{H,\Delta t}$ and $\mathcal{W}_v^{H,\Delta t}$, the numerical displacements and velocities $\tilde{\mathbf{u}}_u$ and $\tilde{\mathbf{u}}_v$ inside the interval I_n are expressed as a combination of the values at times t_{n-1} and t_n, namely

$$\tilde{\mathbf{u}}_u|_{I_n} = \tilde{\mathbf{u}}_u(t_{n-1})\theta_{n-1}(t) + \tilde{\mathbf{u}}_u(t_n)\theta_n(t), \quad (34\mathrm{a})$$
$$\tilde{\mathbf{u}}_v|_{I_n} = \tilde{\mathbf{u}}_v(t_{n-1})\theta_{n-1}(t) + \tilde{\mathbf{u}}_v(t_n)\theta_n(t). \quad (34\mathrm{b})$$

Thus, using the initial conditions for the interval (33c), the values $\tilde{\mathbf{u}}_u(t_{n-1})$ and $\tilde{\mathbf{u}}_v(t_{n-1}) \in \mathcal{V}_0^H(\mathcal{P}_{n-1})$ are known and the only unknowns to be determined are $\tilde{\mathbf{u}}_u(t_n)$ and $\tilde{\mathbf{u}}_v(t_n) \in$

$\mathcal{V}_0^H(\mathcal{P}_n)$. These unknowns are found inserting the representation (34) in equation (33) and noting that the following properties of the time-shape functions hold,

$$\int_{I_n} \theta_{n-1}(t) \, \mathrm{d}t = \int_{I_n} \theta_n(t) \, \mathrm{d}t = \frac{\Delta t_n}{2} \quad \text{and}$$

$$-\int_{I_n} \dot{\theta}_{n-1}(t) \, \mathrm{d}t = \int_{I_n} \dot{\theta}_n(t) \, \mathrm{d}t = 1.$$

Specifically, $[\tilde{\mathbf{u}}_u(t_n), \tilde{\mathbf{u}}_v(t_n)] \in \mathcal{V}_0^H(\mathcal{P}_n) \times \mathcal{V}_0^H(\mathcal{P}_n)$ is such that

$$m(\tilde{\mathbf{u}}_v(t_n), \mathbf{w}_v) + \frac{\Delta t_n}{2} c(\tilde{\mathbf{u}}_v(t_n), \mathbf{w}_v) + \frac{\Delta t_n}{2} a(\tilde{\mathbf{u}}_u(t_n), \mathbf{w}_v)$$
$$= l_{v,n}(\mathbf{w}_v), \ \forall \mathbf{w}_v \in \mathcal{V}_0^H(\mathcal{P}_n), \quad (35\text{a})$$

and

$$a(\tilde{\mathbf{u}}_u(t_n), \mathbf{w}_u) - \frac{\Delta t_n}{2} a(\tilde{\mathbf{u}}_v(t_n), \mathbf{w}_u)$$
$$= l_{u,n}(\mathbf{w}_u), \ \forall \mathbf{w}_u \in \mathcal{V}_0^H(\mathcal{P}_n), \quad (35\text{b})$$

where

$$l_{v,n}(\mathbf{w}) := \int_{I_n} l(t; \mathbf{w}) \, \mathrm{d}t + m(\tilde{\mathbf{u}}_v(t_{n-1}), \mathbf{w})$$
$$- \frac{\Delta t_n}{2} c(\tilde{\mathbf{u}}_v(t_{n-1}), \mathbf{w}) - \frac{\Delta t_n}{2} a(\tilde{\mathbf{u}}_u(t_{n-1}), \mathbf{w}),$$

$$l_{u,n}(\mathbf{w}) := a(\tilde{\mathbf{u}}_u(t_{n-1}), \mathbf{w}) + \frac{\Delta t_n}{2} a(\tilde{\mathbf{u}}_v(t_{n-1}), \mathbf{w}),$$

$$c(\mathbf{v}, \mathbf{w}) := m(a_1 \mathbf{v}, \mathbf{w}) + a(a_2 \mathbf{v}, \mathbf{w}).$$

Note that since the values $\tilde{\mathbf{u}}_u(t_{n-1})$ and $\tilde{\mathbf{u}}_v(t_{n-1})$ are known, the terms associated with this values are placed in the right hand side of the equations.

The computation of the terms appearing in the left hand side of (35) entails no difficulty since all the spatial functions belong to $\mathcal{V}_0^H(\mathcal{P}_n)$. On the contrary, if different spatial computational meshes are used at times t_{n-1} and t_n, the computation of the nodal force vectors associated with $l_{u,n}(\cdot)$ and $l_{v,n}(\cdot)$ involves computing mass and energy products of functions defined in the mesh at time t_{n-1} and functions defined in the mesh at time t_n, e.g. $m(\tilde{\mathbf{u}}_v(t_{n-1}), \mathbf{w}_v)$.

The use of different spatial meshes is efficiently handled by solving the discrete problem (35) using the auxiliary *union* mesh $\mathcal{P}_{n-1,n}$ containing in each zone of the domain the finer elements either in $\mathcal{P}_{n-1}$ or $\mathcal{P}_n$, see figure 18, namely

$$\mathcal{P}_{n-1,n} := \{\omega = \triangle \cap \triangle' \text{ for } \triangle \in \mathcal{P}_{n-1}, \ \triangle' \in \mathcal{P}_n\}.$$

Note that, any function belonging either to $\mathcal{V}_0^H(\mathcal{P}_{n-1})$ or $\mathcal{V}_0^H(\mathcal{P}_n)$ can be represented in the finite element space associated to $\mathcal{P}_{n-1,n}$, namely $\mathcal{V}_0^H(\mathcal{P}_{n-1,n})$, without lose of information. Thus, the products involving functions in different meshes are efficiently computed after projecting the functions in the space $\mathcal{V}_0^H(\mathcal{P}_{n-1,n})$. However, discretizing problem (35) using the mesh $\mathcal{P}_{n-1,n}$ requires introducing additional constrains to enforce that the computed fields $\tilde{\mathbf{u}}_u(t_n)$ and $\tilde{\mathbf{u}}_v(t_n)$ belong to $\mathcal{V}_0^H(\mathcal{P}_n)$ and not to $\mathcal{V}_0^H(\mathcal{P}_{n-1,n})$. That is, problem (35) leads to the following system of equations when discretized in the auxiliary finite element mesh $\mathcal{P}_{n-1,n}$:
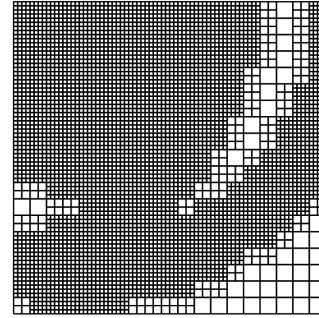
$$\begin{bmatrix} \mathbf{K}_n & -\frac{\Delta t_n}{2} \mathbf{K}_n & \mathbf{A}_n^{\mathrm{T}} & \mathbf{0} \\ \frac{\Delta t_n}{2} \mathbf{K}_n & \mathbf{M}_n + \frac{\Delta t_n}{2} \mathbf{C}_n & \mathbf{0} & \mathbf{A}_n^{\mathrm{T}} \\ \mathbf{A}_n & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_n & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{u,n} \\ \mathbf{U}_{v,n} \\ \boldsymbol{\lambda}_{u,n} \\ \boldsymbol{\lambda}_{v,n} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_{u,n} \\ \mathbf{F}_{v,n} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (36)$$
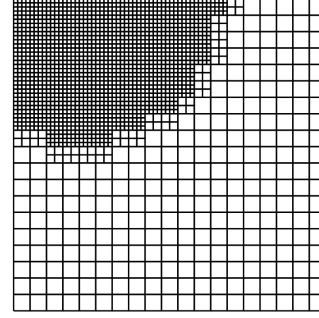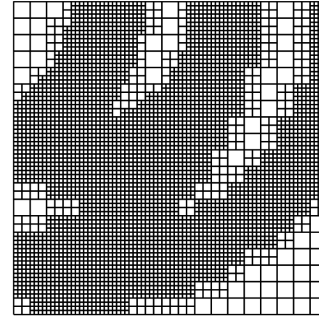
Fig. 18 Illustration of the computational meshes $\mathcal{P}_{n-1}$, $\mathcal{P}_n$ and their *union* $\mathcal{P}_{n-1,n}$.

where

$$\mathbf{F}_{u,n} := \mathbf{K}_n \mathbf{U}_{u,n-1} + \frac{\Delta t_n}{2} \mathbf{K}_n \mathbf{U}_{v,n-1},$$

$$\mathbf{F}_{v,n} := (\mathbf{M}_n - \frac{\Delta t_n}{2} \mathbf{C}_n) \mathbf{U}_{v,n-1}$$
$$- \frac{\Delta t_n}{2} \mathbf{K}_n \mathbf{U}_{u,n-1} + \int_{I_n} \mathbf{F}(t) \, \mathrm{d}t,$$

and $\mathbf{C}_n := a_1 \mathbf{M}_n + a_2 \mathbf{K}_n$. The matrices $\mathbf{M}_n$ and $\mathbf{K}_n$ and the vector $\mathbf{F}(t)$ are the discrete counterparts of the bilinear forms $m(\cdot, \cdot)$ and $a(\cdot, \cdot)$ and the linear form $l(t; \cdot)$ in the space $\mathcal{V}_0^H(\mathcal{P}_{n-1,n})$ and the vectors $\mathbf{U}_{u,n}$, $\mathbf{U}_{v,n}$, $\mathbf{U}_{u,n-1}$ and $\mathbf{U}_{v,n-1}$ contain the degrees of freedom of functions $\tilde{\mathbf{u}}_u(t_n)$, $\tilde{\mathbf{u}}_v(t_n)$, $\tilde{\mathbf{u}}_u(t_{n-1})$ and $\tilde{\mathbf{u}}_v(t_{n-1})$ expressed in the discrete space $\mathcal{V}_0^H(\mathcal{P}_{n-1,n})$. Note that the linear constrains $\mathbf{A}_n \mathbf{U}_{u,n} = \mathbf{0}$ and $\mathbf{A}_n \mathbf{U}_{v,n} = \mathbf{0}$ are introduced in order to ensure that the computed fields $\tilde{\mathbf{u}}_u(t_n)$ and $\tilde{\mathbf{u}}_u(t_n)$ belong to $\mathcal{V}_0^H(\mathcal{P}_n)$ and also to impose continuity of the solution at the hang-

ing nodes, see figure 19. The vectors $\boldsymbol{\lambda}_{u,n}$ and $\boldsymbol{\lambda}_{v,n}$ are the associated Lagrange multipliers.

□ Hanging nodes
○ Disappearing nodes

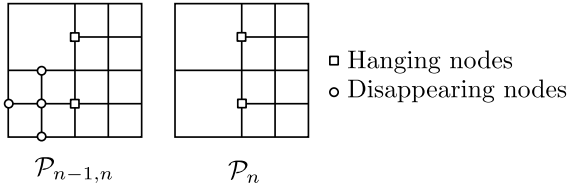$\mathcal{P}_{n-1,n}$ $\mathcal{P}_n$

Fig. 19 The numerical solution is constrained at the nodes of the mesh $\mathcal{P}_{n-1,n}$ corresponding to hanging nodes in the mesh $\mathcal{P}_n$ and also at the nodes of $\mathcal{P}_{n-1,n}$ which disappear in mesh $\mathcal{P}_n$.

Note that system (36) is at the first sight of double size than the one associated with the Newmark method. However, system (36) can be rewritten in a more convenient way by subtracting to the second row of the matrix in (36) the first row multiplied by $\frac{\Delta t_n}{2}$. That is,

$$\begin{bmatrix} \mathbf{K}_n & -\frac{\Delta t_n}{2}\mathbf{K}_n & \mathbf{A}_n^{\mathrm{T}} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_n + \frac{\Delta t_n}{2}\mathbf{C}_n + \frac{\Delta t_n^2}{4}\mathbf{K}_n & -\frac{\Delta t_n}{2}\mathbf{A}_n^{\mathrm{T}} & \mathbf{A}_n^{\mathrm{T}} \\ \mathbf{A}_n & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_n & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{u,n} \\ \mathbf{U}_{v,n} \\ \boldsymbol{\lambda}_{u,n} \\ \boldsymbol{\lambda}_{v,n} \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{F}_{u,n} \\ \mathbf{F}_{v,n} - \frac{\Delta t_n}{2}\mathbf{F}_{u,n} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}.$$

This reformulation allows to compute the velocities separately from the displacements solving a system of the same size as the usual system arising in the Newmark method, namely,

$$\begin{bmatrix} \mathbf{M}_n + \frac{\Delta t_n}{2}\mathbf{C}_n + \frac{\Delta t_n^2}{4}\mathbf{K}_n & \mathbf{A}_n^{\mathrm{T}} \\ \mathbf{A}_n & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{v,n} \\ \boldsymbol{\lambda}_n^* \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{F}_{v,n} - \frac{\Delta t_n}{2}\mathbf{F}_{u,n} \\ \mathbf{0} \end{bmatrix},$$

with $\boldsymbol{\lambda}_n^* := \boldsymbol{\lambda}_{v,n} - \frac{\Delta t_n}{2}\boldsymbol{\lambda}_{u,n}$. Once the velocities $\mathbf{U}_{v,n}$ are known, the displacements are obtained solving the system

$$\begin{bmatrix} \mathbf{K}_n & \mathbf{A}_n^{\mathrm{T}} \\ \mathbf{A}_n & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{u,n} \\ \boldsymbol{\lambda}_{u,n} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_{u,n} + \frac{\Delta t_n}{2}\mathbf{K}_n\mathbf{U}_{v,n} \\ \mathbf{0} \end{bmatrix}. \qquad (37)$$

References

1. Babuŝka, I., Rheinboldt, W.C.: Error estimates for adaptive finite element computations. SIAM J. Numer. Anal. **18**, 736–754 (1978)
2. Bangerth, W., Geiger, M., Rannacher, R.: Adaptive Galerkin finite element methods for the wave equation. Computational Methods in Applied Mathematics **1**, 3–48 (2010)
3. Bangerth, W., Rannacher, R.: Finite element approximation of the acoustic wave equation: error control and mesh adaptation. East–West Journal of Numerical Mathematics **7**, 263–282 (1999)
4. Bangerth, W., Rannacher, R.: Adaptive finite element techniques for the acoustic wave equation. Journal of Computational Acoustics **9**, 575–591 (2001)
5. Carey, V., Estep, D., Johansson, A., Larson, M., Tavener, S.: Blockwise adaptivity for time dependent problems based on coarse scale adjoint solutions. SIAM J. Sci. Comput. **32**, 2121–2145 (2010)
6. Casadei, F., Díez, P., Verdugo, F.: An algorithm for mesh refinement and un-refinement in fast transient dynamics. International Journal of Computational Methods **10**, 1–31 (2013)
7. Cirak, F., Ramm, E.: A posteriori error estimation and adaptivity for linear elasticity using the reciprocal theorem. Comput. Methods Appl. Mech. Engrg. **156**, 351–362 (1998)
8. Demkowicz, L., Oden, J.T., Rachowicz, W., Hardy, O.: Toward a universal h-p adaptive finite element strategy, part 1. Constrained approximation and data structure. Comput. Methods Appl. Mech. Engrg. **77**, 79–112 (1989)
9. Díez, P., Calderón, G.: Remeshing criteria and proper error representations for goal oriented h-adaptivity. Comput. Methods Appl. Mech. Engrg. **196**, 719–733 (2007)
10. Eriksson, K., Estep, D., Hansbo, P., Johnson, C.: Computational Differential Equations. Studentlitteratur (1996)
11. Fuentes, D., Littlefield, D., Oden, J.T., Prudhomme, S.: Extensions of goal-oriented error estimation methods to simulation of highly-nonlinear response of shock-loaded elastomer-reinforcedstructures. Comput. Methods Appl. Mech. Engrg. **195**, 4659–4680 (2006)
12. Hughes, T.J.R., Hulbert, G.M.: Space-time finite element methods for elastodynamics: Formulations and error estimates. Comput. Methods Appl. Mech. Engrg. **66**, 339–363 (1988)
13. Hulbert, G.M., Hughes, T.J.R.: Space-time finite element methods for second-order hypeerbolic equations. Comput. Methods Appl. Mech. Engrg. **84**, 327–348 (1990)
14. Johnson, C.: Discontinuous galerkin finite element methods for second order hyperbolic problems. Comput. Methods Appl. Mech. Engrg. **107**, 117–129 (1993)
15. Ladevèze, P., Leguillon, D.: Error estimate procedure in the finite element method. SIAM J. on Numerical Analysis **20**, 485–509 (1983)
16. Larsson, F., Hansbo, P., Runesson, K.: Strategies for computing goal-oriented a posteriori error measures in non-linear elasticity. Int. J. Numer. Meth. Engrg. **55**, 879–894 (2002)
17. Meyer, A.: Error estimators and the adaptive finite element method on large strain deformation problems. Mathematical Methods in the Applied Sciences **32**, 2148–2159 (2009)
18. Nithiarasu, P., Zienkiewicz, O.C.: Adaptive mesh generation for fluid mechanics problems. Int. J. Numer. Meth. Engrg. **47**, 629–662 (2000)
19. Oden, J.T., Prudhomme, S.: Goal-oriented error estimation and adaptivity for the finite element method. Computers and Math. with Appl. **41**, 735–765 (2001)
20. Paraschivoiu, M., Peraire, J., Patera, A.T.: A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations. Comput. Methods Appl. Mech. Engrg. **150**, 289–321 (1997)
21. Parés, N., Bonet, J., Huerta, A., Peraire, J.: The computation of bounds for linear-functional outputs of weak solutions to the two-dimensional elasicity equations. Comput. Methods Appl. Mech. Engrg. **195**, 406–429 (2006)
22. Parés, N., Díez, P., Huerta, A.: A subdomain-based flux-free a posteriori error estimators. Comput. Methods Appl. Mech. Engrg. **195**, 297–323 (2006)
23. Parés, N., Díez, P., Huerta, A.: Bounds of functional outputs for parabolic problems. Part I: Exact bounds of the discontinuous galerkin time discretization. Comput. Methods Appl. Mech. Engrg. **197**, 1641–1660 (2008)

24. Parés, N., Díez, P., Huerta, A.: Bounds of functional outputs for parabolic problems. Part II: Bounds of the exact solution. Comput. Methods Appl. Mech. Engrg. **197**, 1661–1679 (2008)
25. Parés, N., Díez, P., Huerta, A.: Exact bounds of the advection-diffusion-reaction equation using flux-free error estimates. SIAM J. Sci. Comput. **31**, 3064–3089 (2009)
26. Parés, N., Díez, P., Huerta, A.: Computable exact bounds for linear outputs from stabilized solutions of the advection-diffusion-reaction equation. Int. J. Numer. Meth. Engng. **93**, 483–509 (2013)
27. Peraire, J., Vahdati, M., Morgan, K., Zienkiewicz, O.C.: Adaptive remeshing for flow computations. J. Comp. Physics **72**, 449–466 (1987)
28. Prudhomme, S., Oden, J.T.: On goal–oriented error estimation for elliptic problems: application to the control of pointwise errors. Comput. Methods Appl. Mech. Engrg. **176**, 313–331 (1999)
29. Rannacher, R., Stuttmeier, F.T.: A feed-back approach to error control in finite element methods: application to linear elasticity. Comput. Mech. **19**, 434–446 (1997)
30. Verdugo, F., Parés, N., Díez, P.: Modal based goal-oriented error assessment for timeline-dependent quantities in transient dynamics. Int. J. Numer. Meth. Engng. **95**, 685–720 (2013)
31. Yerry, M.A., Shephard, M.S.: A modified quadtree approach to finite element mesh generation. IEEE Computer graphics and Applications **3**, 34–46 (1983)
32. Zienkiewicz, O.C., Zhu, J.Z.: A simple error estimator and adaptative procedure for practical engineering analysis. Int. J. Numer. Meth. Engrg. **24**, 337–357 (1987)