Computer Methods in Applied Mechanics and Engineering 452 (2026) 118763

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

Computing leaky waves in semi-analytical waveguide models by exponential residual relaxation

Hauke Gravenkamp [a,*], Bor Plestenjak [b], Daniel A. Kiefer [c]

[a] Institute of Materials, Technologies and Mechanics, Otto von Guericke University Magdeburg, 39106 Magdeburg, Germany
[b] IMFM and Faculty of Mathematics and Physics, University of Ljubljana, Jadranska 19, SI-1000 Ljubljana, Slovenia
[c] Institut Langevin, ESPCI Paris, Université PSL, CNRS, 75005 Paris, France

ARTICLE INFO

Keywords:
Guided waves
Plates
Soil dynamics
Leaky waves
Semi-analytical method
Zhang neural networks

ABSTRACT

Semi-analytical methods for modeling guided waves in structures of constant cross-section yield frequency-dependent polynomial eigenvalue problems for the wavenumbers and mode shapes. Solving these eigenvalue problems over a range of frequencies results in continuous eigencurves. Recent research has shown that eigencurves of differentiable parameter-dependent eigenvalue problems can alternatively be computed as solutions to a system of ordinary differential equations (ODEs) obtained by postulating an exponentially decaying residual of a modal solution. Starting from an approximate initial guess of the eigenvalue and eigenvector at a given frequency, the complete eigencurve is obtained using standard numerical ODE solvers. We exploit this idea to develop an efficient method for computing the dispersion curves of plate structures coupled to unbounded solid or fluid media. In these scenarios, the approach is particularly useful because the boundary conditions give rise to nonlinear terms that severely hinder the application of traditional solvers. We discuss suitable approximations of the nonlinearity for obtaining initial values, analyze computational costs and robustness of the proposed algorithm, and verify results by comparison against existing methods.

1. Introduction

We consider the problem of wave propagation along thin structures, also known as guided waves. We focus on elastic waves *(stress waves)*, while similar phenomena exist in acoustics, electromagnetism, or piezo-electric structures. Such guided elastic waves are relevant in various applications in science and engineering. Specifically, in the ultrasonic range, they are widely used in non-destructive testing (NDT) [1,2] and material characterization [3,4], while at much lower frequencies, they play a role in soil dynamics and earthquake engineering [5]. An interesting scenario is encountered when such waveguides are in contact with unbounded solid or fluid media, see Fig. 1 and similar discussions in [6]. These situations can occur, e.g., in layered soils [7] and plate structures [8,9] or pipes [10] in NDT applications.

Guided waves in linear systems give rise to a displacement field that can be written as the superposition of propagating and evanescent modes [11], each of them characterized by a mode shape and an axial wavenumber. At any given frequency, an infinite number of modes exist; however, only a finite subset propagates with sufficiently low attenuation to be practically relevant. The computation of these modes' wavenumbers over a range of frequencies yields dispersion curves, which are essential in many applications

* Corresponding author.
E-mail addresses: hauke.gravenkamp@ovgu.de (H. Gravenkamp), bor.plestenjak@fmf.uni-lj.si (B. Plestenjak), daniel.kiefer@espci.fr (D.A. Kiefer).

https://doi.org/10.1016/j.cma.2026.118763
Received 5 September 2025; Received in revised form 15 January 2026; Accepted 16 January 2026
Available online 22 January 2026

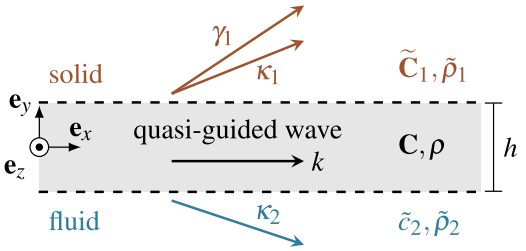

Fig. 1. Plate of thickness h, stiffness tensor $\mathbf{C}$, and mass density ρ in contact with unbounded media of different properties. For illustration, the plate is coupled both to another solid ($\widetilde{\mathbf{C}}_1, \tilde{\rho}_1$) and a fluid halfspace (wave velocity $\tilde{c}_2$, density $\tilde{\rho}_2$) at the top and bottom surface, respectively, while all combinations of fluid, solid, and free surface are generally possible. The wavenumber of a (quasi-) guided wave mode is denoted by k, while the free-field wavenumbers of longitudinal and shear waves in the unbounded media are $\kappa_{1,2}$ and γ_1, respectively.

for predicting wave propagation behavior. For some simple cases, such as homogeneous plates with traction-free or clamped surfaces, implicit closed-form expressions for dispersion relations can be derived and solved by standard root-finding algorithms [12]. More complex structures, i.e., layered plates or cylindrical waveguides, require more general formulations like the Transfer Matrix Method, Global Matrix Method, or Stiffness Matrix Method, together with mode-tracing techniques [13–15].

Over the past few decades, semi-analytical methods have emerged as the standard for most practical problems. These methods discretize the waveguide's cross-section using finite elements or similar techniques, yielding an eigenvalue formulation whose solutions are the axial wavenumbers and mode shapes. Notable representatives of such methods include the semi-analytical finite element method (SAFE) [16,17], the thin layer method (TLM) [18,19], and a variant of the scaled boundary finite element method [20], which are all widely adopted due to their robustness and ability to capture all solutions that the chosen discretization can represent. Despite their advantages, semi-analytical methods face challenges in describing waveguides that are in contact with unbounded media. A common approach is to apply the well-known concept of perfectly matched layers (PML) [21,22] or a variant called perfectly matched discrete layers (PMDL) [23]. While this technique requires discretizing parts of the unbounded domain (hence increasing matrix sizes) and additional effort for filtering spurious modes and dealing with backward propagating modes [24], it is a viable option in many scenarios, especially for three-dimensional problems, where analytical descriptions of the wave transmission into the unbounded domain are limited [25]. A simpler, yet generally more expensive approach relies on absorbing regions [26], i.e., a truncated discretization of the unbounded domain with appropriate material damping. To a lesser extent, boundary element methods have been employed to achieve the same goal [27]. Furthermore, approximate techniques using dashpots [28] have been developed to account for radiation damping at high frequencies in an efficient manner. In contrast to this, incorporating the exact boundary conditions (i.e., the actual relationship between the wavenumbers in the plate and those in the unbounded media) leads to nonlinear eigenvalue problems [6,27,29,30], which are notoriously difficult to solve [31,32]. In special cases, such as homogeneous fluid loading, the problem can be linearized and efficiently solved [33]. Iterative solutions have also been proposed for fluid loading scenarios [30]. The latter uses the solutions of the free waveguide as initial guesses and attempts to compute solutions of the nonlinear problem by inverse iteration, which is relatively efficient but not always robust in the presence of many modes with similar eigenvalues.

Recently, we demonstrated that the nonlinear eigenvalue problem arising in more general cases, such as layered plate structures coupled to fluid and/or solid halfspaces, can be stated as a multiparameter eigenvalue problem [6]. This reformulation enables the application of recently developed solution techniques [34], implemented in the Matlab toolbox MultiParEig [35]. While this approach is highly robust and capable of finding all solutions, it is computationally demanding. The cost increases rapidly with the size of the finite element matrices and the number of additional parameters, restricting its application to relatively small cases.

Hence, it is worthwhile to consider other ideas, in particular with the goal of obtaining more efficient algorithms that extend the applicability of the underlying formulation to more complex problems. In this paper, we explore an alternative approach to computing eigencurves based on a methodology that was presented by Zhang et al. [36,37] and further developed by Uhlig et al. [38,39] for the solution of different parameter-dependent matrix equations. The key idea is to transform the matrix equation (in our application, a parameter-dependent nonlinear eigenvalue problem) into an ordinary differential equation (ODE) by assuming that the solution's residual decays exponentially with respect to the free parameter. The resulting ODE can then be solved using standard numerical techniques. This concept arose in this form in the study of recurrent neural networks (see also, e.g., [40,41]) and is hence known to many as Zeroing Neural Networks [42] or Zhang Neural Networks [43], both abbreviated as ZNN. There are several somewhat similar methodologies, such as dynamic relaxation [44], artificial compressibility [45], or Baumgarte stabilization [46].

In [47], we have briefly considered this approach for computing eigencurves of the free waveguide problem, but stated that we were 'not yet sure about this approach's practical usefulness.' The free waveguide problem is generally easy to linearize and efficiently solvable by standard methods, making an alternative mode-tracing approach less relevant. However, we now recognize that this technique is particularly advantageous for leaky waveguides, as it can directly incorporate the involved nonlinearities. This allows computations to be performed on the original matrices, in contrast to the transformation into a multiparameter eigenvalue problem in [6], which ultimately requires solving a much larger linear problem. Furthermore, in the case of leaky waves, the focus is typically on a small subset of modes – specifically those that radiate energy away from the waveguide and exhibit relatively low attenuation. A core advantage of the proposed approach is that it follows individual modes during computation, unlike traditional semi-analytical solution procedures or our previous work, which compute eigenvalues at predefined frequency or wavenumber points

without tracking their evolution. This bears some conceptual resemblance to mode-tracing techniques used in the Global Matrix Method and other analytical approaches. These methods typically employ extrapolation of known solutions, which is essential for establishing sufficiently accurate starting values for root-finding algorithms. For specific problems, tangentially related continuation approaches have been used in the modeling of free elastic waveguides [48], for finding exceptional points in acoustic waveguides [49], or to trace bending modes in fluid-conveying pipes [50]. The latter problem class has also been investigated using the technique presented in the current paper [51].

This paper focuses on developing an algorithm to compute eigencurves of the nonlinear eigenvalue problem describing leaky waves in plate structures. For brevity, we will not provide details on the derivation of said eigenvalue problem and instead refer to some of the many papers that contain the complete formulation. We will pose the known problem directly in Section 2 and then explain the fundamental concept of deriving a system of ODEs from a given objective function in Section 3. We then apply this formulation to the problem at hand, i.e., the nonlinear eigenvalue problem, in Section 4 and discuss the issue of efficiently obtaining good initial values to start the mode-tracing in Section 5. Finally, we present a selection of numerical examples of increasing complexity in Section 6 before drawing conclusions from our findings in Section 7.

2. Problem statement

The semi-discrete version of the guided wave formulation, when the coupling to unbounded fluid or solid media is included, leads to a nonlinear eigenvalue problem for wavenumbers k and eigenvectors $\boldsymbol{\phi}$ with the frequency ω playing the role of the continuous parameter. The eigenvector includes the nodal displacements arising from the finite-element discretization of the plate's cross-section, as well as the amplitudes of the displacements or pressure describing the unbounded domains at the plate's surfaces. The detailed derivation of the formulation in the version used here is presented in [6] and is not repeated here. Further details on the underlying semi-analytical model for general waveguides can be found in, e.g., Gravenkamp et al. [20,30] and many others. The finite-element matrices used in the following were computed with the help of the open-source Matlab implementation *SAMWISE* [52]. Since the form of the waveguide problem discussed here involves only terms in ω^2, we introduce $\mu = \omega^2$ and write

$$\left(-k^2\mathbf{E}_0 + \mathrm{i}k\mathbf{E}_1 - \mathbf{E}_2 + \mu\mathbf{M} + \mathbf{R}(k,\mu)\right)\boldsymbol{\phi} = \mathbf{0}. \tag{1}$$

The term $\mathbf{R}(k,\mu)$ incorporates the nonlinearities that arise due to coupling to the unbounded halfspaces at the top and bottom of the plate structure; it reads

$$\mathbf{R}(k,\mu) = \sum_{j=1}^{6} b_j \xi_j \mathbf{R}_j \tag{2}$$

with

$$b_j = \begin{cases} \mathrm{i} & j \in \{1,2\} \quad \text{fluid halfspace,} \\ k & j \in \{3,4,5,6\} \quad \text{solid halfspace.} \end{cases} \tag{3}$$

Here, the symbol ξ represents vertical wavenumbers of partial waves in the unbounded domains, and i is the imaginary unit. The sparse coupling matrices $\mathbf{R}_j$ have been derived in detail previously [6]. For simplicity in notation, we always include all six terms (one/two partial waves in each adjacent fluid/solid), and for any halfspace that is not present in the current model, we set the corresponding $\mathbf{R}_j$ matrices to $\mathbf{0}$ and ξ_j to an arbitrary finite value. Hence, we introduce here the parameter b_j to distinguish the slightly different terms for solid and fluid media. This notation implies that $j \in \{1,2\}$ always correspond to variables associated with the acoustic fluids (if present), while $j \in \{3,4,5,6\}$ are reserved for solid media. The vertical wavenumber ξ_j is related to the corresponding free-field wave speed c_j (for every halfspace present in the model) via

$$\xi_j(k,\mu) = \pm\sqrt{\frac{\mu}{c_j^2} - k^2}. \tag{4}$$

Note that, in the above relationship, the vertical wavenumber depends nonlinearly on the eigenvalue k, which is the main challenge in solving this class of problems.

3. Solution approach

3.1. General concept

The basic idea, as outlined in [37–39] and applied to the free waveguide problem in [47], is the following. Assume we want to minimize a parameterized objective function $f(y(t),t)$, i.e., solve

$$f(y(t),t) = 0 \tag{5}$$

with a given parameter $t \in [t_0, t_1]$. Thus, the goal is to find a function $y(t)$, such that the objective function vanishes for all t within the given interval. To address this problem numerically, we postulate the existence of the following derivative

$$f'(y(t),t) = -\chi f(y(t),t), \tag{6}$$

which implies an exponentially decreasing residual of (5). Here, the prime symbol denotes the total derivative with respect to t, and χ is an algorithmic constant to be chosen later. Eq. (6) represents an ordinary differential equation (ODE) that can be solved numerically, provided that y is known at some initial value, say $y(t_0) =: y_0$. Hence, instead of Eq. (5), we will solve the initial value problem:

Let $f(y(t), t)$ be a differentiable objective function and $\mathcal{I} = (t_0, t_1)$ an open interval. Find $y(t)$ such that

$$f'(y(t), t) = -\chi f(y(t), t), \quad t \in \mathcal{I} \tag{7a}$$

$$y(t_0) = y_0. \tag{7b}$$

3.2. Introductory example

Let us consider the objective function

$$f(y(t), t) = t + (t^2 + 1)y - 2 \tag{8}$$

with the derivative

$$f'(y(t), t) = 1 + 2ty + (t^2 + 1)y'. \tag{9}$$

This problem has a simple analytical solution that we pose here for verifying the results; namely, the objective function vanishes for

$$y_e(t) = \frac{2 - t}{t^2 + 1}, \tag{10}$$

i.e.,

$$f(y_e(t), t) = 0 \quad \forall\, t. \tag{11}$$

To demonstrate the numerical procedure, we substitute (8) and (9) into (6) and obtain

$$(t^2 + 1)\, y' = -\chi\big(t + (t^2 + 1)y - 2\big) - 1 - 2ty. \tag{12}$$

The ODE (12) can be integrated numerically starting from some known value of y. Here, we employ an implicit multistep solver with a variable step size as implemented in Matlab's function *ode15s*. Note that here and in the following examples, we employ the solver in its 'mass matrix form' and avoid division by the factor $t^2 + 1$. For a more efficient evaluation, we can provide the solver with the Jacobian, which is obtained as the derivative of the right-hand side of (12) with respect to y:

$$J(t) = -\chi(t^2 + 1) - 2t. \tag{13}$$

Say we want to solve Eq. (8) within an interval $t \in [-4, 4]$. As a starting value, we could, of course, use the known exact solution at a given t-value. However, to demonstrate the robustness of this approach, particularly when an exact solution is unknown, we use an approximate initial value. Noting that

$$\lim_{t \to \pm\infty} y_e(t) = 0, \tag{14}$$

we approximate

$$y_e(-4) \approx 0 \tag{15}$$

and, hence, select $t_0 = -4$, $y_0 = 0$. Furthermore, we choose $\chi = 10$ and set the relative tolerance of the numerical solver to 10^{-6}. Results of the computed solution and the corresponding residual are shown in Fig. 2. It is worth noting that, despite the large error of the initial condition (the exact value is $y_e(-4) = 6/17 \approx 0.35$), the numerical result rapidly approaches the exact solution and ultimately yields residuals below 10^{-5}. In fact, the algorithm converges even if the initial condition is off by several orders of magnitude.

3.3. Extension to coupled systems of equations

We observe that the procedure outlined above is adjusted straightforwardly to treat systems that depend on several functions of the parameter t. Namely, the minimization of a system of objective functions

$$\mathbf{f}(\mathbf{y}(t), t) =: \begin{bmatrix} f_1(y_1(t), y_2(t), \dots, y_n(t), t) \\ f_2(y_1(t), y_2(t), \dots, y_n(t), t) \\ \vdots \\ f_n(y_1(t), y_2(t), \dots, y_n(t), t) \end{bmatrix} = \mathbf{0} \tag{16}$$

is posed as the solution of the initial value problem

$$\mathbf{f}'(\mathbf{y}(t), t) = -\chi \mathbf{f}(\mathbf{y}(t), t), \quad t \in \mathcal{I} \tag{17a}$$

$$\mathbf{y}(t_0) = \mathbf{y}_0. \tag{17b}$$

Here, we have assumed that the same algorithmic constant χ is chosen for all entries of $\mathbf{f}(\mathbf{y}(t), t)$. Otherwise, χ is to be replaced by a corresponding diagonal matrix $\mathbf{X} = \mathrm{diag}(\chi_1, \chi_2, \dots, \chi_n)$.

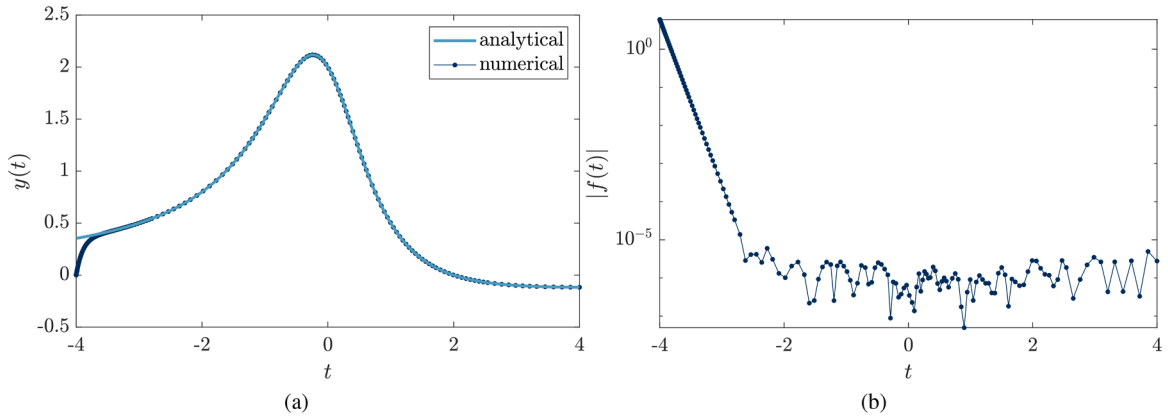

Fig. 2. Analytical and numerical solution $y(t)$ of the introductory example and residual $|f(t)|$ as functions of the parameter t.

3.4. Application to eigenvalues of matrix functions

We will now apply the same technique to compute the eigencurves of a matrix function, i.e., to solve an eigenvalue problem that depends on one parameter:

$$\mathbf{L}(k,\mu)\boldsymbol{\phi} = \mathbf{0}. \tag{18}$$

The specific case of (leaky) guided waves will be addressed in the following section. For easier comparison with the previous literature, the eigenvalue and eigenvector are denoted as $k = k(\mu)$, $\boldsymbol{\phi} = \boldsymbol{\phi}(\mu)$, and $\mu = \omega^2$ is the free parameter. As the eigenvectors are only defined up to a multiplicative constant, we need to choose a normalization in order to obtain unique results, e.g.,

$$n(\boldsymbol{\phi}) =: \boldsymbol{\phi}^{\mathrm{H}}\boldsymbol{\phi} - 1 = 0. \tag{19}$$

Hence, we define the objective function

$$\mathbf{f}(\boldsymbol{\phi}, k, \mu) = \begin{bmatrix} \mathbf{L}(k,\mu)\boldsymbol{\phi} \\ n(\boldsymbol{\phi}) \end{bmatrix} \tag{20}$$

and write the total derivative with respect to the parameter μ as

$$\mathbf{f}'(\boldsymbol{\phi}, k, \mu) = \begin{bmatrix} \mathbf{L}(k,\mu)\boldsymbol{\phi}' + \mathbf{L}'(k,\mu)\boldsymbol{\phi} \\ \boldsymbol{n}_{\boldsymbol{\phi}}(\boldsymbol{\phi})\,\boldsymbol{\phi}' \end{bmatrix}. \tag{21}$$

Remark 1. The normalization defined by Eq. (19) is not complex differentiable with respect to $\boldsymbol{\phi}$, though it is differentiable with respect to the real and imaginary parts $\Re(\boldsymbol{\phi})$, $\Im(\boldsymbol{\phi})$, and the Wirtinger derivatives exist [53], permitting a well-defined complex gradient [54,55]. It is known in the context of optimization problems that we can employ the approximation $2\boldsymbol{\phi}^{\mathrm{H}}$ (analogously to real-valued vectors) to obtain a linearization, e.g., in Newton's method [56]. Here, we will use the same concept to approximate the derivative with respect to $\boldsymbol{\phi}$ that we require in the following steps. There are other possible normalizations; in particular, a popular choice is to use $n(\boldsymbol{\phi}) = \boldsymbol{\phi}_{\mathrm{c}}^{\mathrm{H}}\boldsymbol{\phi} - 1$, with some constant vector $\boldsymbol{\phi}_{\mathrm{c}}$ (e.g., the initial value). While this approach generally works and avoids the theoretical issue of a non-differentiable normalization, we found that the one defined by Eq. (19) resulted in a slightly more robust computation in our examples. Hence, we will stick with this version and the general notation $\boldsymbol{n}_{\boldsymbol{\phi}}$, irrespective of the normalization we employ, keeping in mind that, in the case of Eq. (19), $\boldsymbol{n}_{\boldsymbol{\phi}}(\boldsymbol{\phi}) = 2\boldsymbol{\phi}^{\mathrm{H}}$ is not a true complex derivative.

We continue by isolating terms in $\mathbf{f}'(\boldsymbol{\phi}, k, \mu)$ that are multiplied by $\boldsymbol{\phi}$, $\boldsymbol{\phi}'$, or k'. To this end, the total derivative $\mathbf{L}'(k,\mu)$ is written using the partial derivatives as

$$\mathbf{L}'(k,\mu) = \frac{\mathrm{d}}{\mathrm{d}\mu}\mathbf{L}(k,\mu) = \frac{\partial}{\partial\mu}\mathbf{L}(k,\mu) + \frac{\partial}{\partial k}\mathbf{L}(k,\mu)\,\frac{\mathrm{d}}{\mathrm{d}\mu}k, \tag{22}$$

which we abbreviate as

$$\mathbf{L}'(k,\mu) =: \mathbf{L}_{\mu}(k,\mu) + \mathbf{L}_{k}(k,\mu)\,k', \tag{23}$$

such that

$$\mathbf{f}'(\boldsymbol{\phi}, k, \mu) = \begin{bmatrix} \mathbf{L}(k,\mu)\boldsymbol{\phi}' + \mathbf{L}_{k}(k,\mu)\boldsymbol{\phi}\,k' + \mathbf{L}_{\mu}(k,\mu)\boldsymbol{\phi} \\ \boldsymbol{n}_{\boldsymbol{\phi}}(\boldsymbol{\phi})\,\boldsymbol{\phi}' \end{bmatrix}. \tag{24}$$

The precise definition of these terms will be specified later for the cases of interest. Substituting Eqs. (20) and (24) into (17a), we find that the system of ODEs is of the form

$$\begin{bmatrix} \mathbf{L}(k,\mu) & \mathbf{L}_{k}(k,\mu)\boldsymbol{\phi} \\ \boldsymbol{n}_{\boldsymbol{\phi}}(\boldsymbol{\phi}) & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}' \\ k' \end{bmatrix} = -\begin{bmatrix} \chi_1\mathbf{L}(k,\mu)\boldsymbol{\phi} \\ \chi_2 n(\boldsymbol{\phi}) \end{bmatrix} - \begin{bmatrix} \mathbf{L}_{\mu}(k,\mu)\boldsymbol{\phi} \\ 0 \end{bmatrix}, \tag{25}$$

which we will abbreviate for later use as

$$\mathbf{A}(\boldsymbol{\phi}, k, \mu)\begin{bmatrix}\boldsymbol{\phi}'\\ k'\end{bmatrix} = \mathbf{b}(\boldsymbol{\phi}, k, \mu). \tag{26}$$

Note that we use two different decay parameters χ_1, χ_2 to account for a potentially different behavior of the matrix Eq. (18) and the normalization (19). The Jacobian is given as

$$\mathbf{J}(\boldsymbol{\phi}, k, \mu) = \begin{bmatrix}\partial_{\boldsymbol{\phi}}\mathbf{b} & \partial_k\mathbf{b}\end{bmatrix} = -\begin{bmatrix}\chi_1\mathbf{L}(k,\mu) + \mathbf{L}_\mu(k,\mu) & \chi_1\mathbf{L}_k(k,\mu)\boldsymbol{\phi} + \mathbf{L}_{\mu k}(k,\mu)\boldsymbol{\phi}\\ \chi_2\boldsymbol{n}_{\boldsymbol{\phi}}(\boldsymbol{\phi}) & 0\end{bmatrix}, \tag{27}$$

with

$$\mathbf{L}_{\mu k} = \frac{\partial^2}{\partial k\partial\mu}\mathbf{L}(k,\mu). \tag{28}$$

Both $\boldsymbol{\phi}$ and k are generally complex-valued. It is interesting to note that (25) can be split into real and imaginary parts in order to optimize both simultaneously. Defining $\boldsymbol{\phi} = \boldsymbol{\phi}_r + \mathrm{i}\boldsymbol{\phi}_i$ and $k = k_r + \mathrm{i}k_i$, we can rewrite (25) as

$$\begin{bmatrix}\Re(\mathbf{L}) & -\Im(\mathbf{L}) & \Re(\mathbf{L}_k\boldsymbol{\phi}) & -\Im(\mathbf{L}_k\boldsymbol{\phi})\\ \Im(\mathbf{L}) & \Re(\mathbf{L}) & \Im(\mathbf{L}_k\boldsymbol{\phi}) & \Re(\mathbf{L}_k\boldsymbol{\phi})\\ \Re(\boldsymbol{n}_{\boldsymbol{\phi}}) & -\Im(\boldsymbol{n}_{\boldsymbol{\phi}}) & 0 & 0\\ \Im(\boldsymbol{n}_{\boldsymbol{\phi}}) & \Re(\boldsymbol{n}_{\boldsymbol{\phi}}) & 0 & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{\phi}_r'\\ \boldsymbol{\phi}_i'\\ k_r'\\ k_i'\end{bmatrix} = -\begin{bmatrix}\chi_1\Re(\mathbf{L}\boldsymbol{\phi})\\ \chi_1\Im(\mathbf{L}\boldsymbol{\phi})\\ \chi_2\Re(n)\\ \chi_2\Im(n)\end{bmatrix} - \begin{bmatrix}\Re(\mathbf{L}_\mu\boldsymbol{\phi})\\ \Im(\mathbf{L}_\mu\boldsymbol{\phi})\\ 0\\ 0\end{bmatrix}. \tag{29}$$

While some implementations of solvers for systems of ODEs may recommend or even require this real-valued formulation, we will stick with the complex version in what follows.

3.5. Incorporating algebraic constraints

It is generally straightforward to formulate the approach for an arbitrary number of coupled equations, as mentioned in Section 3.3. Here, we only showcase one specific case that will be of particular importance to our current application, namely the inclusion of additional scalar equations. Assume the eigenvalue problem depends on another parameter $\xi = \xi(\mu)$, such that

$$\mathbf{L}(k,\xi,\mu)\boldsymbol{\phi} = \mathbf{0}. \tag{30}$$

In our case, the solution to (30) satisfies a scalar equation

$$g(k,\xi,\mu) = 0. \tag{31}$$

Thus, Eq. (25) is extended as

$$\begin{bmatrix}\mathbf{L}(k,\xi,\mu) & \mathbf{L}_\xi(k,\xi,\mu)\boldsymbol{\phi} & \mathbf{L}_k(k,\xi,\mu)\boldsymbol{\phi}\\ \mathbf{0} & g_\xi(k,\xi,\mu) & g_k(k,\xi,\mu)\\ \boldsymbol{n}_{\boldsymbol{\phi}}(\boldsymbol{\phi}) & 0 & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{\phi}'\\ \xi'\\ k'\end{bmatrix} = -\begin{bmatrix}\chi_1\mathbf{L}(k,\xi,\mu)\boldsymbol{\phi}\\ \chi_1 g(k,\xi,\mu)\\ \chi_2 n(\boldsymbol{\phi})\end{bmatrix} - \begin{bmatrix}\mathbf{L}_\mu(k,\xi,\mu)\boldsymbol{\phi}\\ g_\mu(k,\xi,\mu)\\ 0\end{bmatrix}, \tag{32}$$

where the matrix entries are defined via

$$\mathbf{L}'(k,\xi,\mu) =: \mathbf{L}_\mu(k,\xi,\mu) + \mathbf{L}_\xi(k,\xi,\mu)\,\xi' + \mathbf{L}_k(k,\xi,\mu)\,k', \tag{33a}$$

$$g'(k,\xi,\mu) =: g_\mu(k,\xi,\mu) + g_\xi(k,\xi,\mu)\,\xi' + g_k(k,\xi,\mu)\,k', \tag{33b}$$

and the Jacobian is extended correspondingly as

$$\mathbf{J}(\boldsymbol{\phi}, k, \xi, \mu) = \begin{bmatrix}\partial_{\boldsymbol{\phi}}\mathbf{b} & \partial_\xi\mathbf{b} & \partial_k\mathbf{b}\end{bmatrix} = -\begin{bmatrix}\chi_1\mathbf{L} + \mathbf{L}_\mu & \chi_1\mathbf{L}_\xi\boldsymbol{\phi} + \mathbf{L}_{\mu\xi}\boldsymbol{\phi} & \chi_1\mathbf{L}_k\boldsymbol{\phi} + \mathbf{L}_{\mu k}\boldsymbol{\phi}\\ \mathbf{0} & \chi_1 g_\xi + g_{\mu\xi} & \chi_1 g_k + g_{\mu k}\\ \chi_2\boldsymbol{n}_{\boldsymbol{\phi}} & 0 & 0\end{bmatrix}. \tag{34}$$

Several constraints of this form can be included analogously.

4. Specific case of (leaky or trapped) guided waves

4.1. Overview

We can now address the problem at hand, i.e., the nonlinear parameterized eigenvalue problem (1). For clarity, we substitute Eqs. (2) and (4) into (1) to appreciate the overall structure of the eigenvalue problem:

$$\Big(-k^2\mathbf{E}_0 + \mathrm{i}k\mathbf{E}_1 - \mathbf{E}_2 + \mu\mathbf{M} + \sum_{j=1}^{6}\pm b_j\sqrt{\mu/c_j^2 - k^2}\,\mathbf{R}_j\Big)\boldsymbol{\phi} = \mathbf{0}. \tag{35}$$

Note that each combination of signs for the vertical wavenumbers describes a distinct mode, and we will fix this combination of signs while tracing one individual mode. To apply the solution procedure outlined in Section 3.4, we only need to derive the terms $\mathbf{L}_\mu(k,\mu)$,

$\mathbf{L}_k(k,\mu)$, $\mathbf{L}_{\mu k}(k,\mu)$. As they are trivial to obtain, we will provide the relevant expressions without much further explanation. In the following, we will refer to this variant of the algorithm as VERSION I. We will see that this approach is remarkably robust and accurate for the most part – with one important exception: The square-root function $\sqrt{z}$ is not complex differentiable for $\Re(z) \leq 0, \Im(z) = 0$ (and not even continuous at $\Im(z) = 0$ for any $\Re(z) < 0$).[1]

While the vast majority of solutions exhibit a significant imaginary part of the wavenumber (i.e., attenuation due to leakage into the adjacent halfspace or material damping), some *trapped modes* may be present that are characterized by a vanishing imaginary part. Famously, the *quasi-Scholte modes* that exist in a solid plate immersed in an acoustic fluid propagate with a real wavenumber, say k_S, slightly larger than the free-field wavenumber in the fluid, and are thus characterized by

$$(\kappa^f)^2 - k_S^2 < 0, \quad |(\kappa^f)^2 - k_S^2| \ll 1, \quad \Im\big((\kappa^f)^2 - k_S^2\big) = 0. \tag{36}$$

Numerically, the consequence of this discontinuity in the coupling term is that the solution of the ODE is prone to strong oscillations when the argument of the square root approaches the negative real axis. The nonlinear solver essentially extrapolates based on previous solutions of the eigenvector and eigenvalue and performs iterations to minimize the residual of the nonlinear objective function. Hence, near the negative real axis, it becomes likely that, during an extrapolation or iteration, an approximation for the eigenvalue is found with the incorrect sign. Once this happens, the nonlinear iterations are unlikely to converge due to the discontinuity. Depending on the employed solver, this issue manifests either as erroneous solutions or as a failure to adjust the step size within the acceptable error tolerance.

To eliminate this issue of a discontinuous objective function, we consider an alternative variant of the proposed procedure that consists of treating the vertical wavenumbers ξ_j as additional parameters and solving the coupled system of equations

$$\Big(-k^2\mathbf{E}_0 + \mathrm{i}k\mathbf{E}_1 - \mathbf{E}_2 + \mu\mathbf{M} + \sum_{j=1}^{6} b_j\,\xi_j\mathbf{R}_j \Big)\boldsymbol{\phi} = \mathbf{0}, \tag{37a}$$

$$\xi_1^2 - \frac{\mu}{c_1^2} + k^2 = 0, \tag{37b}$$

$$\xi_2^2 - \frac{\mu}{c_2^2} + k^2 = 0, \tag{37c}$$

$$\vdots$$

Thus, for each partial wave in one of the unbounded domains (if present), we add a constraint equation in the sense of Section 3.5. Again, this formulation is straightforwardly obtained after deriving the terms in Eq. (32), which will be given in Section 4.3. We will refer to this formulation as VERSION II. Clearly, the advantage of this approach lies in the fact that the constraint equations are continuously differentiable with respect to all parameters (k, ξ_j, μ), hence, numerical instabilities near real-valued solutions are avoided. However, this version can, in some cases, cause a different problem: As we introduce a constraint on ξ_j^2 rather than ξ_j, instabilities can occur if two solutions corresponding to the branches $+\sqrt{\mu/c_j^2 - k^2}$ and $-\sqrt{\mu/c_j^2 - k^2}$ are very similar. So far, we have encountered this issue only once (Example IV in Section 6.2) in the case of rather extreme attenuation due to a small acoustic mismatch between the plate structure and the unbounded media. Nevertheless, it is worthwhile to combine both versions into a single highly robust algorithm. We propose using VERSION I as the default variant and switching to VERSION II only when the argument of the square root function for any of the partial waves gets close to the negative real axis. Specifically, we use VERSION II if $|\Im(\xi_j^2)| < 0.01|\Re(\xi_j^2)|$ or $|\xi_j^2| < 0.01$ for any of the partial waves. We note that VERSION II evokes slightly higher computational costs per evaluation due to up to four additional unknowns ξ_j. However, this difference is generally negligible, as the number of unknowns emerging from the finite-element discretization of the plate is typically significantly larger.

4.2. VERSION I

Let us now list the expressions obtained by applying the proposed approach to the two versions of the problem statement mentioned above. The first version is based on Eq. (25) with

$$\mathbf{L}(k,\mu) = -k^2\mathbf{E}_0 + \mathrm{i}k\mathbf{E}_1 - \mathbf{E}_2 + \mu\mathbf{M} + \mathbf{R}(k,\mu) \tag{38}$$

and

$$\mathbf{R}(k,\mu) = \sum_{j=1}^{6} b_j\,\xi_j\mathbf{R}_j. \tag{39}$$

Noting that

$$\frac{\partial}{\partial k}\xi_j(k,\mu) = -\frac{k}{\xi_j}, \qquad \frac{\partial}{\partial \mu}\xi_j(k,\mu) = \frac{1}{2c_j^2\xi_j}, \tag{40}$$

[1] As a multivalued function, $\sqrt{z}$ cannot be continuous in the whole complex plane. The standard convention introduces a branch cut along the negative real axis ($\Im(z) = 0$, $\Re(z) < 0$), and defines the principal value $\sqrt{z}$ with nonnegative real part. Then $\sqrt{z}$ is analytic off the branch cut, but for any real $x > 0$, $\lim_{\delta\to 0}\sqrt{-x+\mathrm{i}\delta} = \mathrm{i}\sqrt{x} \neq -\mathrm{i}\sqrt{x} = \lim_{\delta\to 0}\sqrt{-x-\mathrm{i}\delta}$, so $\sqrt{z}$ is not continuous on the negative real axis. It is possible to use other definitions that would instead lead to a discontinuity, e.g., for positive imaginary numbers, resulting in the analogous challenge for different modes.

and

$$b_{j,k} =: \frac{\mathrm{d}}{\mathrm{d}k} b_j = \begin{cases} 0 & j \in \{1,2\} \quad \text{fluid halfspace,} \\ 1 & j \in \{3,4,5,6\} \quad \text{solid halfspace,} \end{cases} \tag{41}$$

the expressions in Eq. (25) are readily obtained as

$$\mathbf{L}_k(k,\mu) = \frac{\partial}{\partial k}\mathbf{L}(k,\mu) = -2k\mathbf{E}_0 + \mathrm{i}\mathbf{E}_1 + \sum_{j=1}^{6}\left(b_{j,k}\,\xi_j - \frac{b_j k}{\xi_j}\right)\mathbf{R}_j, \tag{42}$$

$$\mathbf{L}_\mu(k,\mu) = \frac{\partial}{\partial \mu}\mathbf{L}(k,\mu) = \mathbf{M} + \sum_{j=1}^{6}\frac{b_j}{2\xi_j c_j^2}\mathbf{R}_j. \tag{43}$$

For computing the Jacobian, we additionally require

$$\mathbf{L}_{\mu k}(k,\mu) = \frac{\partial^2}{\partial k \partial \mu}\mathbf{L}(k,\mu) = \sum_{j=1}^{6}\left(\frac{b_{j,k}}{2\xi_j c_j^2} + \frac{b_j k}{2\xi_j^3 c_j^2}\right)\mathbf{R}_j. \tag{44}$$

4.3. Version II

The second variant differs from the first one in that we treat the ξ_j as additional parameters, such that

$$\mathbf{L} = \mathbf{L}(k,\boldsymbol{\xi},\mu), \quad \mathbf{R} = \mathbf{R}(k,\boldsymbol{\xi},\mu) \tag{45}$$

and include the constraint equations of the form

$$g_j(k,\xi_j,\mu) = \xi_j^2 - \frac{\mu}{c_j^2} + k^2 = 0. \tag{46}$$

The terms in Eq. (32) are thus obtained as

$$\mathbf{L}_\mu = \mathbf{M}, \tag{47}$$

$$\mathbf{L}_k(k,\boldsymbol{\xi}) = -2k\mathbf{E}_0 + \mathrm{i}\mathbf{E}_1 + \sum_{j=1}^{6} b_{j,k}\,\xi_j \mathbf{R}_j, \tag{48}$$

$$\mathbf{L}_\xi(k,\boldsymbol{\xi}) = \sum_{j=1}^{6} b_j\,\mathbf{R}_j, \tag{49}$$

$$g_{j\mu}(k,\xi_j,\mu) = -\tfrac{1}{c_j^2}, \tag{50}$$

$$g_{jk}(k,\xi_j,\mu) = 2k \tag{51}$$

$$g_{j\xi}(k,\xi_j,\mu) = 2\xi_j. \tag{52}$$

As before, it is assumed that the constraint equations are only included if the corresponding half-space is present in the model. Computing the Jacobian via Eq. (34) is particularly simple, since

$$g_{j\mu k} = g_{j\mu\xi} = 0, \tag{53}$$

$$\mathbf{L}_{\mu k} = \mathbf{L}_{\mu\xi} = \mathbf{0}. \tag{54}$$

4.4. On the choice of the decay parameters χ_1, χ_2

While the proposed approach is not overly sensitive to the choice of the algorithmic constants χ_1, χ_2, we must select appropriate values for the residual to decay over a reasonable interval of the parameter μ. Keeping in mind the fundamental idea that the objective function is assumed to decrease as $\exp(-\chi\mu)$, it is intuitive to choose these constants such that the residual becomes negligible within a small fraction of the considered interval of μ. In the context of waveguide modeling, we can relate this interval to the dimensionless frequency a_0, defined as [20]

$$a_0 = \frac{\omega h}{c_t} \tag{55}$$

with the layer thickness h and the (smallest) shear wave velocity c_t. For a multi-layered system, we use the average dimensionless frequency over all layers. A reasonable approach to choosing χ_1 is obtained by requiring that the residual reduces by a factor of $1/e$ within an interval Δa_0, significantly smaller than the largest dimensionless frequency of interest. This consideration leads, for a single layer, to

$$\chi_1 = c_{\chi_1}\frac{h^2}{c_t^2}, \quad \chi_2 = c_{\chi_2}\frac{h^2}{c_t^2} \tag{56}$$

with constants c_{χ_1}, c_{χ_2}. If the waveguide consists of n_ℓ layers with individual thicknesses h_i and $c_{t,i}$, we adapt these equations accordingly:

$$\chi_1 = \frac{c_{\chi_1}}{n_\ell} \sum_{i=1}^{n_\ell} \frac{h_i^2}{c_{t,i}^2}, \quad \chi_2 = \frac{c_{\chi_2}}{n_\ell} \sum_{i=1}^{n_\ell} \frac{h_i^2}{c_{t,i}^2}. \tag{57}$$

We may note that the decay parameters have the SI unit s^2, as we performed the differentiation of the objective function w.r.t. $\mu = \omega^2$ in Eq. (21). When computing dispersion curves, we are typically interested in dimensionless frequencies roughly up to $a_{0,\max} = 10$, including the first five to ten propagating modes [20]. Hence, we may choose $c_{\chi_1} = 100$, i.e., a decay by a factor of $1/e$ within an interval $\Delta a_0 = 0.1$ – typically about one percent of the entire frequency range of interest. In our numerical studies, we have found that the algorithm works well if we set $\chi_2 = \chi_1$. However, we required somewhat fewer steps in the ODE solver by choosing a smaller value, e.g., $\chi_2 = \chi_1/10$. This is a plausible choice, as χ_2 affects only the residual of the normalization. While the residual of the eigenvalue problem can take relatively large values whose decay is governed by χ_1, we can enforce the normalization to be exactly satisfied from the start. Hence, we do not require a rapid decrease of the residual in the normalization; in addition, the normalization does not have to be satisfied exactly as long as the error in eigenvalues and eigenvectors remains small. Our numerical examples will be performed with $c_{\chi_1} = 100, c_{\chi_2} = 10$, and we will give some numerical evidence for the suitability of this choice at the end of Section 6.2.

5. Initial values

The proposed approach – like all methods based on mode tracing – requires adequate starting values, which, in our case, can be viewed as initial values of the system of ordinary differential Eqs. (25) or (32). Hence, we need to obtain eigenvalues and eigenvectors of the parameter-dependent eigenvalue problem at some value of the parameter (here, the frequency) – at least approximately. We choose to compute starting values at the highest frequency of interest and trace the modes back towards $\omega = 0$. This is often helpful, as the modes are typically well separated at higher frequencies, and it is easier to select the relevant propagating modes. There are numerous ways in which we can attempt to obtain such starting values. The first one is obvious: We can make use of the direct numerical method presented in [6], which allows us to compute all solutions at a given frequency, except that this approach is computationally expensive and, hence, suitable for reasonably small matrix sizes. In this scenario, we would employ the expensive direct solution only once to obtain initial values and trace all required modes starting there. This strategy may still have merit – particularly when we do not require all modes, or when we want to follow the behavior of the modes along the frequency – but the benefit may not always be worth the effort of dealing with these two solution procedures. On the other hand, it is much more exciting to explore the possibilities of *approximating* solutions at a given frequency to obtain starting values. As we demonstrated in Section 3.2, the mode-tracing algorithm can converge to a branch even when the initial values are very far from a true solution. However, in the case of eigencurves (or any system of coupled equations), the challenge consists in finding *all* solutions. Thus, we need an adequate number of sufficiently good approximate solutions, such that every branch will be reached by one of the starting values. Ideally, we will achieve this by using exactly as many initial values as there are correct solutions, so that we do not waste resources finding the same branch multiple times.

For our problem at hand, we can attempt to approximate the nonlinear terms that occur in the eigenvalue problem, i.e., approximate the functions of the form $\sqrt{\frac{\omega_0^2}{c_j^2} - k^2}$ at a given frequency ω_0. One particularly simple approximation is obtained as

$$\sqrt{\frac{\omega_0^2}{c_j^2} - k^2} \approx \frac{\omega_0}{c_j} \qquad \textit{dashpot approximation}, \tag{58}$$

which implies that the wavenumber in the halfspace $\frac{\omega_0}{c_j}$ is significantly larger than that in the waveguide. This is a decent approximation at high frequencies as long as the free-field wave speeds in the waveguide are larger than those in the halfspace, which is often true in realistic scenarios. In fact, this assumption leads to roughly the same formulation as the dashpot boundary condition that has been presented in [28], except that, in the previous work, the interaction with the halfspace was considered as a von Neumann boundary condition. There, the applicability to many relevant scenarios has been discussed, as well as the limitations of this approach. To build on this idea, we propose to employ the approximation

$$\sqrt{\frac{\omega_0^2}{c_j^2} - k^2} \approx \sqrt{\frac{\omega_0^2}{c_j^2} - \bar{k}^2} \qquad \textit{mean value approximation}. \tag{59}$$

Here, we replace the unknown k by a characteristic wavenumber $\bar{k}$. This value is obtained by assuming that the real part of a mode propagating in the waveguide will typically be between zero and the largest free-field wavenumber of the waveguide's material at the given frequency

$$\bar{k} = \frac{\omega_0}{2c_{\min}}, \tag{60}$$

where $c_{\min}$ denotes the minimum wave speed in any of the materials inside the waveguide. Substituting this approximation in each of the nonlinear terms in (35) yields a simple quadratic eigenvalue problem that can be solved straightforwardly, even for large

matrices. We note in passing that this is a much better approximation on average[2] than the one originally proposed in [28]. As we will see in the numerical examples, this approximation yields excellent starting values that are suitable to trace (almost all) the modes in all examples we have tried. The only exception we have encountered is the quasi-Scholte modes at a fluid/solid interface, which are characterized by $\frac{\omega_0}{c_j^f} \approx k$ and hence are not well represented by any of the approximations above. In many applications, we may decide to ignore these modes, as they are often of little practical relevance.[3] Nevertheless, we found that we can approximate their wavenumber as

$$k \approx \frac{\omega_0}{c_j^f} \qquad \textit{quasi-Scholte approximation.} \tag{61}$$

The corresponding eigenvector is well approximated by setting only the component describing the acoustic pressure in the fluid to one and all other components to zero. In conclusion, we use the mean-value approximation and estimate one additional mode for each fluid/solid interface by the quasi-Scholte approximation. This combination was found to yield very reliable initial values such that all eigencurves are found by the mode-tracing algorithm in all examples we have studied.

6. Numerical examples

In this section, we demonstrate the applicability of the proposed approach to problems of different complexity. We begin with a minimal example that can easily be reproduced and verified against an analytical solution. In the ensuing, we present four examples of increasing computational demand, involving both homogeneous and layered structures coupled to fluid or solid unbounded media.

6.1. Minimal example

We begin with a simple case that we discussed in a different context in [47], namely, an eigenvalue problem of the form (1) with

$$\mathbf{M} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad \mathbf{E}_0 = \frac{1}{3}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad \mathbf{E}_1 = \mathbf{0}, \quad \mathbf{E}_2 = \frac{3}{2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{R} = \mathbf{0}. \tag{62}$$

This rather academic example can be obtained by considering a homogeneous plate (thickness $h = 2$, mass density $\rho = 3$, shear modulus $G = 1$, Poisson's ratio $\nu = 0.25$, plane strain) with horizontal displacements fixed and vertical displacements approximated by only one linear finite element. The four eigencurves are calculated analytically as

$$k_{1,2} = \pm\sqrt{3}\omega, \qquad k_{3,4} = \pm\sqrt{3\omega^2 - 9}. \tag{63}$$

Note that, at $\omega = \sqrt{3}$, the solutions $k_{3,4}$ coincide, and both eigencurves are not continuously differentiable, which becomes a challenge for the adaptive ODE solver employed here. To circumvent this issue, we regularize the eigenvalue problem by modifying the coefficient matrices as

$$\hat{\mathbf{E}}_i = \mathbf{E}_i(1 - \mathrm{i}\delta). \tag{64}$$

This is equivalent to changing the shear modulus to $G = 1 - \mathrm{i}\delta$, which can be interpreted as including a small amount of material damping with a constant damping coefficient δ. To obtain numerically stable computations, it is sufficient to use $\delta = 10^{-12}$, which is the value chosen for the presented results. We calculate initial conditions by solving the eigenvalue problem at $\omega_0 = 4$ and employ the mode-tracing algorithm described in Section 4.2 with a decay parameter $\chi = 10$ to trace the modes towards $\omega = 0$. The multistep solver implemented in Matlab's function *ode15s* is employed for the solution of the ODE. Results are presented in Fig. 3, showing the real and imaginary parts of each mode i, as well as the error defined as

$$\mathrm{error}_i = \frac{|k_{i,\mathrm{exact}} - k_{i,\mathrm{numerical}}|}{\max(|k_{i,\mathrm{exact}}|)}. \tag{65}$$

The subscript i indicates that this error is computed separately for each mode, based on the difference between the numerical and the exact solution at the points resulting from the adaptive stepping procedure. The error is normalized by the maximum absolute value of the eigenvalue of the respective mode within the frequency range of interest. The error is primarily influenced by the relative tolerance chosen in the adaptive multistep solver, here 10^{-6}. Except for a small region around $\omega = \sqrt{3}$, where the error in modes 3 and 4 increases to around 10^{-3} due to the mentioned non-differentiability of two eigencurves, it remains below the requested tolerance. Note that, due to the problem's symmetry, the errors of modes 1,2 and those of modes 3,4 are indistinguishable in the figure.

[2] For most modes, $\bar{k}$ is a better approximation than assuming $k = 0$ in the nonlinear terms, especially at relatively large frequencies as all waveguide modes tend towards a free-field solution. However, modes close to their cut-off frequency exhibit a small wavenumber; hence, there will always be a few modes for which this approximation is worse than the simple dashpot.

[3] An important exception can be found in the design of surface acoustic wave (SAW) filters.

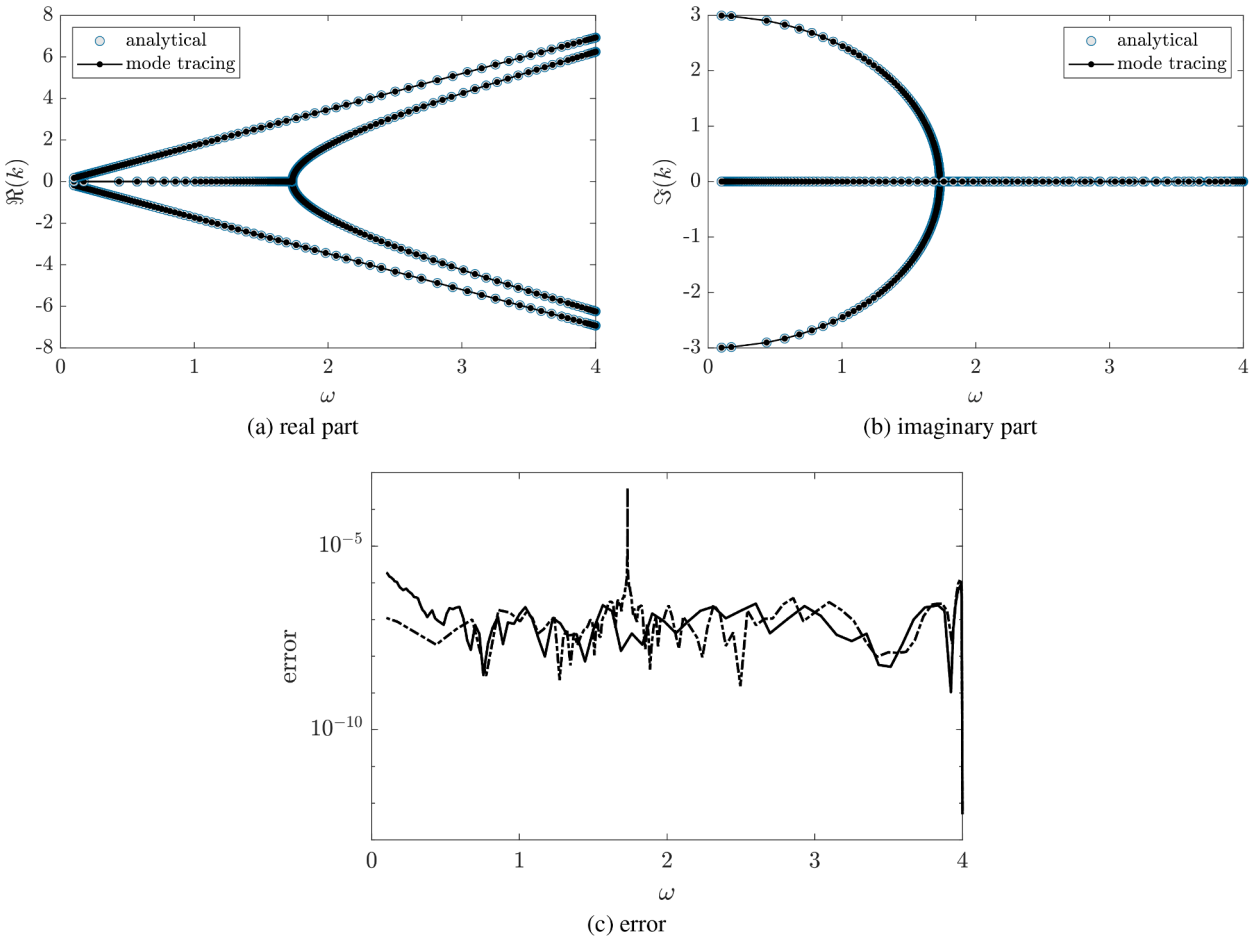

Fig. 3. Minimal example involving matrices of size 2×2. The real (a) and imaginary (b) parts of the eigencurves in terms of the eigenvalue k are compared visually with the analytical solution. The relative error of each eigenvalue within the chosen frequency range is presented in (c).

6.2. Leaky waves

Finally, we employ the proposed approach to compute dispersion curves of leaky guided waves. The examples are summarized in Table 1. For better reproducibility and conciseness, models I–IV are the same as those described in detail in a recent publication [6]. These test cases had been designed to cover a wide range of different scenarios, and the results are thoroughly verified against other techniques, namely, a linearization in the special case of symmetric fluid coupling [33] and the Global Matrix Method implemented in the commercial software *disperse* [57] for more complex cases. Hence, we can be brief in discussing these examples, as our focus here is solely on a different numerical approach. For details on the physical behavior and the peculiarities of each setup, we refer to the previous work [6]. In examples I, II, and IV, the plate is a homogeneous layer of either brass or titanium, whereas in the third scenario, it consists of three layers (titanium-brass-titanium). Each of those layers has a thickness of 1 mm and is discretized by one finite element of a polynomial order p_e adequate for the chosen frequency range up to $f_{\max}$. Details on how to select the element order for such waveguide models can be found in [58]. In each scenario, the plate is coupled to one or two halfspaces at its bottom and/or top surface as described by the column *halfspace*. The examples increase in difficulty from the prevalent and comparatively straightforward case of a plate immersed in a single fluid to the rather extreme scenario in which a metal plate is coupled to two different solid halfspaces with a slight acoustic mismatch at the interfaces – and, hence, extremely large attenuation of the leaky guided wave modes. The elastic constants of each material are listed in Table A.1 in Appendix A. In the first example, we include a small amount of material damping with $\delta = 0.001$ according to Eq. (64) to allow for a smoother transition from leaky to trapped modes.

In addition to the examples already validated in the previous publication, we include one particularly challenging case in example V. It consists of an eight-ply fiber composite material, namely T800-913, in a $[0/90]_{2s}$-layup. The same composite (but with traction-free surfaces) is discussed in [59]; further details can be found there, and the material parameters are listed in Appendix A. Wave behavior in such anisotropic composites is particularly interesting, as the wave velocities vary strongly between the propagation directions. In our example, the composite is coupled to water and steel, respectively, at the two surfaces, and we include 1 % hysteretic

Table 1
Summary of the numerical examples discussed in Section 6.2, indicating the materials of each halfspace and layer, the element orders p_e, number of degrees of freedom n_{dof} (size of finite-element matrices), maximum frequency f_{max}, maximum attenuation η_{max}, and the CPU times for the proposed approach ('tracing') and the direct solution based on multiparameter eigenvalue problems ('MultiParEig').

	material						CPU time	
	halfspace	layer	p_e	n_{dof}	f_{max}	η_{max}	tracing	MultiParEig
I	water / water	brass	9	22	4 MHz	2100 dB/m	0.7 s	2.2 s
II	— / Teflon	brass	13	45	7 MHz	7000 dB/m	0.8 s	41 s
III	Teflon / oil	Ti / brass / Ti	6 / 8 / 6	45	3 MHz	2000 dB/m	0.6 s	68 s
IV	Teflon / brass	Ti	13	32	10 MHz	30000 dB/m	0.5 s	288 s
V	steel / water	T800-913 $[0/90]_{2s}$	3	79	5 MHz	50000 dB/m	5.2 s	965 s

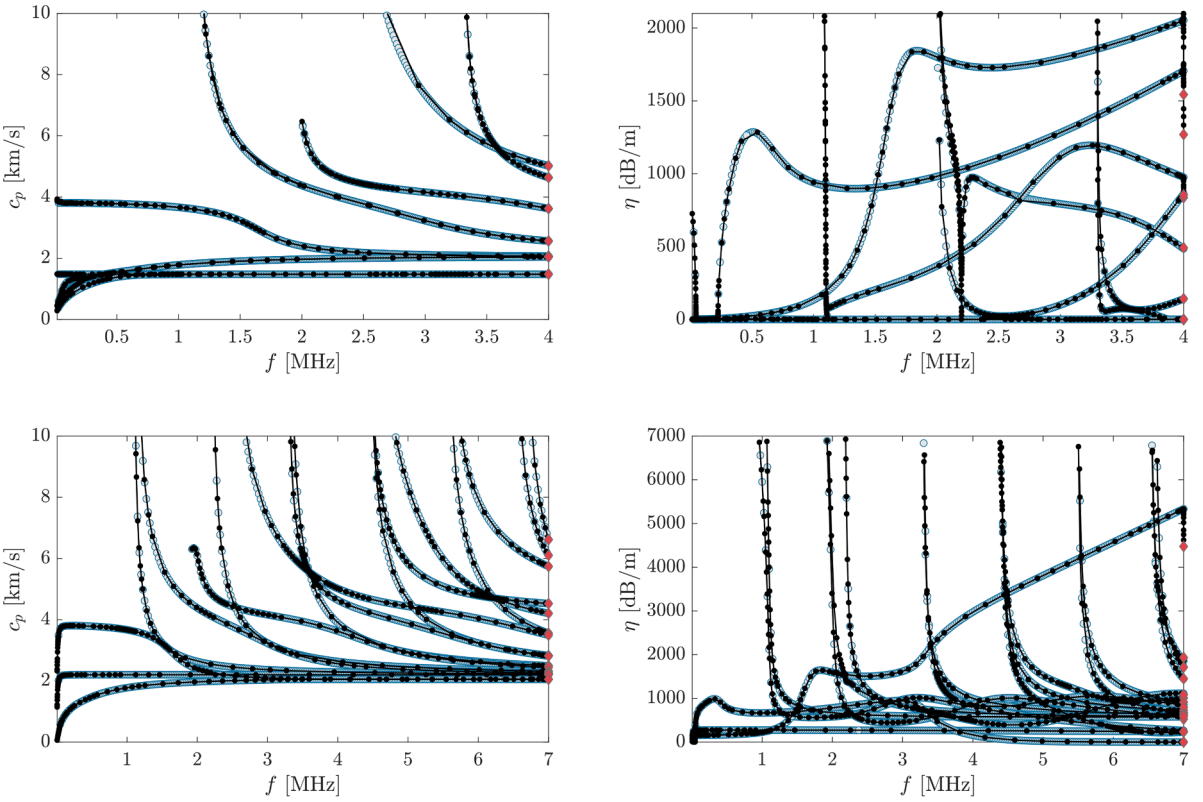

Fig. 4. Dispersion curves computed using the proposed mode tracing approach and direct computation for examples I (top) and II (bottom) as defined in Table 1. Results are presented in terms of phase velocity c_p and attenuation η as functions of the temporal frequency f.

damping. In this example, the total thickness of the composite is 1 mm, and each layer is discretized by one finite element of cubic order, leading to a total of 79 degrees of freedom.

In all examples, we compute initial conditions at the largest frequency of interest f_{max} using the approximation detailed in Section 5 and trace each mode towards $f = 0$. Note that the mode tracing is formulated in terms of the squared circular frequency μ, which is converted into the temporal frequency f for the plots only. For conciseness and comparability, we present those modes that are characterized by all partial wave vectors in the halfspaces pointing away from the plate. For a discussion on incoming and radiating waves and methods for their distinction, we refer again to [6]. Furthermore, we remove the strongly attenuated (*non-propagating*) modes, i.e., solutions with an attenuation above the chosen value of η_{max} listed in Table 1. The dispersion curves in terms of phase velocities and attenuation are presented in Fig. 4 for examples I, II, and in Fig. 5 for examples III–V. The results of the proposed approach are compared with those obtained in [6] by direct solution of the multiparameter eigenvalue problem. In all cases, both computations yield the same dispersion curves with no noticeable deviations in the presented figures. Note that the approximation described in Section 5 generally produces very good initial values at high frequencies, with discrepancies mainly visible in the attenuation of strongly attenuated modes. However, even in the cases where the initial values are relatively poor approximations

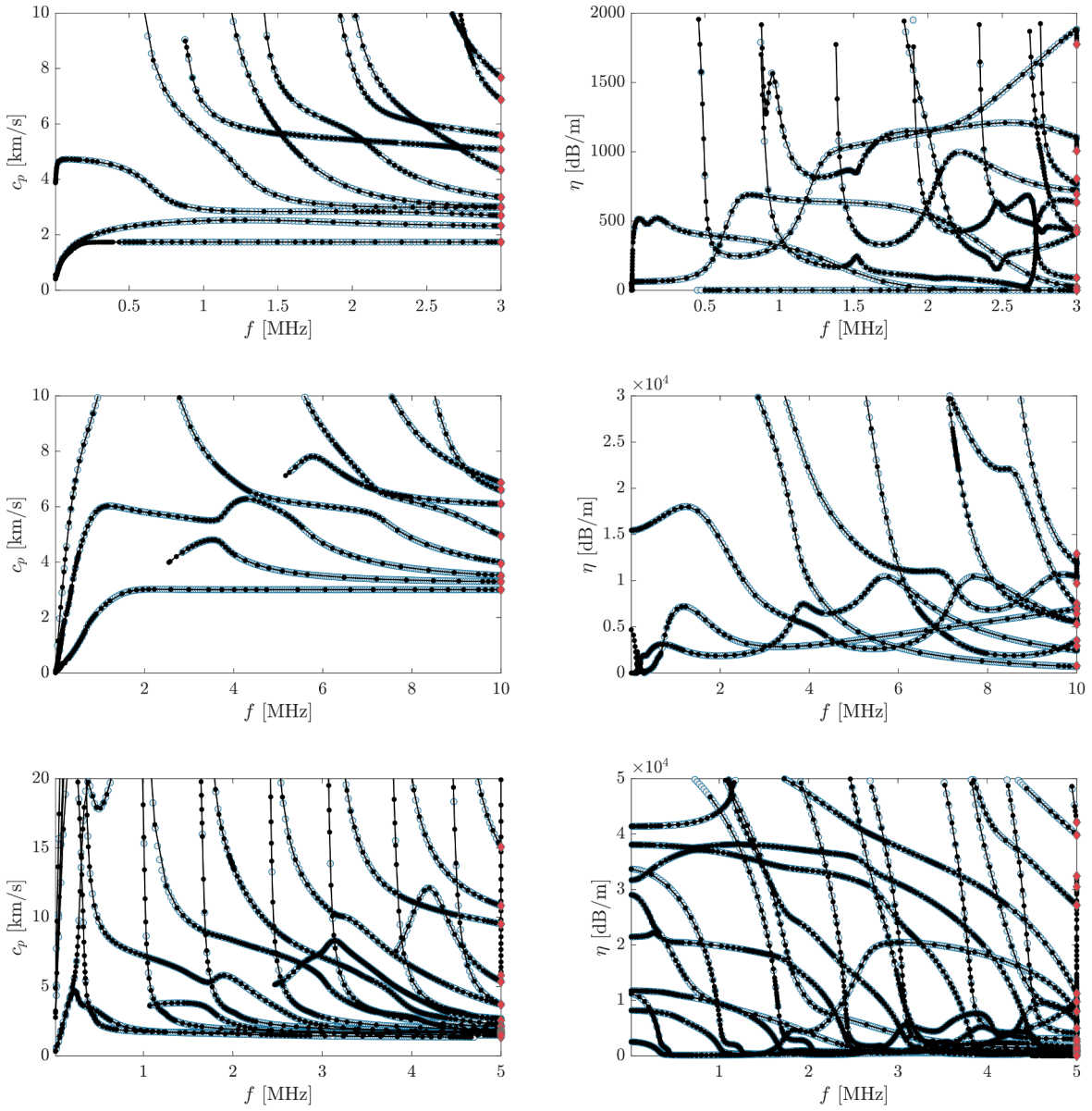

Fig. 5. Dispersion curves computed using the proposed mode tracing approach and direct computation for examples III (top), IV (middle), and V (bottom) as defined in Appendix A. Results are presented in terms of phase velocity c_p and attenuation η as functions of the temporal frequency f.

(see the attenuation in example I), the mode tracing reaches highly accurate solutions within a few steps. We may also highlight that the quasi-Scholte modes have been successfully computed. Another interesting detail is that the mode-tracing approach is capable of following eigencurves that exhibit rapid changes, which could be missed by a constant frequency step; see, e.g., the almost vertical lines in the attenuation of example I, Fig. 4.

For evaluating the accuracy of the presented results, we computed the residual r_i of the objective function (20) for each individual solution $(\boldsymbol{\phi}_i, k_i, \mu_i)$

$$r_i = |\mathbf{r}_i|, \qquad \mathbf{r}_i = \begin{bmatrix} \frac{\mathbf{L}(k_i,\mu_i)\boldsymbol{\phi}_i}{|\boldsymbol{\phi}_i|} \\ n(\boldsymbol{\phi}_i) \end{bmatrix}, \tag{66}$$

where $|\bullet|$ denotes the Euclidean vector norm. The residual depends on the requested relative tolerance '*RelTol*' of the ODE solver. Hence, in Fig. 6, we present the residual for three different values of the relative tolerance. For brevity, these results are given only for example III, while the other examples yield similar values. As the residual varies slightly between the individual modes, the figure

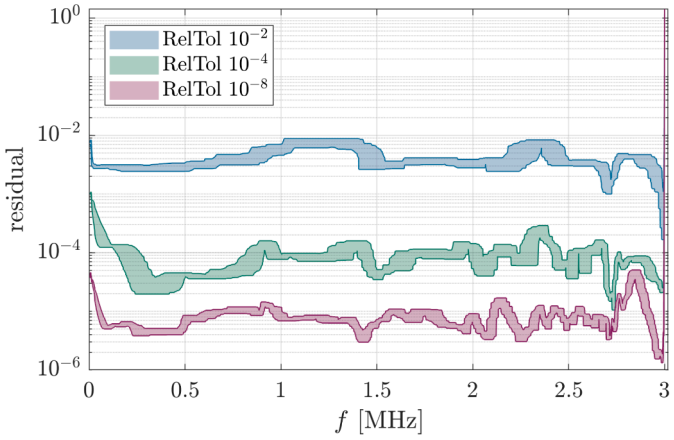

Fig. 6. Residuals of the nonlinear eigenvalue problem for example III. Three different values of the targeted relative residual in the ODE solver have been used. For each value, we show the range between the smallest and largest residual of all computed modes depending on the frequency f. Other examples behave similarly.

shows the range between the smallest and largest residual as a function of frequency. It was observed that reducing the relative tolerance below 10^{-6} does not improve the results any further, as a systematic error remains due to approximating the eigenvalue problem by the ODE (25). However, the results in Figs. 4 and 5 have been computed using a relative tolerance of 10^{-2}, already leading to sufficiently accurate results for practical applications.

Table 1 also lists the computational times required by both approaches for the specific examples studied here. Care must be taken when drawing general conclusions, as the computational costs depend on many factors. Obviously, the costs increase with the size of the finite-element matrices, and, especially in the case of the multiparameter eigenvalue problem, they depend drastically on the number of partial waves in the unbounded media. Furthermore, when employing the mode-tracing algorithm, the frequency steps are adjusted automatically based on the requested relative tolerance of the ODE solver (here 10^{-2}), while we choose a fixed frequency resolution for the 'MultiParEig approach.' Another important difference lies in the fact that, in the mode-tracing algorithm, we can choose to compute only specific modes (in the numerical examples, those are the weakly attenuated modes with wave vectors of partial waves in the unbounded domains pointing away from the plate). Perhaps most importantly, the implementation of the mode-tracing algorithm is not optimized for efficiency, as it involves calls to Matlab functions at every iteration in every step for every mode. In comparison, the bottleneck of the MultiParEig approach is a highly optimized eigenvalue problem solver. Despite all these difficulties in comparing the different techniques, the data serves to give the reader a rough idea of the efficiency of the mode-tracing approach, with CPU times in the order of one second for reasonable examples.

To obtain a different perspective, we evaluate the CPU times per frequency and mode, i.e., the average cost of obtaining one individual solution in the dispersion diagram. This is mainly relevant if we wish to compute all modes, including the evanescent ones. To this end, we used examples I and IV (i.e., one and four partial waves in the unbounded domains) and computed the solution for varying matrix sizes by increasing the polynomial degree of the finite-element approximation. We then divided the total computational time by the number of computed modes and frequency steps, leading to an average CPU time per solution. This value is plotted in Fig. 7 against the number of degrees of freedom for both examples and both approaches. We see that the MultiParEig approach is highly efficient in the simple case of Example I, whereas in the more complex Example IV, computational costs rapidly increase with matrix size, making mode tracing much more effective, even when computing all solutions.

Finally, we studied the effect of varying the decay parameters–specifically c_{χ_1}, c_{χ_2} according to the definition in Eq. (57) – on the solution's accuracy and efficiency. To assess the former, we computed the median of the residual as defined by Eq. (66) over all solutions. For this study, we set the relative tolerance of the ODE-solver to 10^{-6} to isolate the effect of the decay parameters from the accuracy in solving the ODE. In addition, we took note of the number of steps N_{steps} required by the solver. For brevity, we present in Fig. 8 these values only for example III, as it involves a layered plate and coupling to both a fluid and a solid halfspace; however, results for the other examples are similar. First, we observe that the proposed method is remarkably robust across a wide range of decay parameters (note the logarithmic scales of c_{χ_1}, c_{χ_2}), with a relatively minor effect on the average residual. There is, however, some trade-off between a low residual and a small number of steps. This is quite intuitive, as a small decay parameter results in a slow decay of deviations from the exact solution and, hence, to a larger residual. On the other hand, a too large decay parameter induces rapid changes in the solution in response to perturbations, which can require a finer resolution and hence a larger number of steps. We may also note that, in particular, the number of steps is less sensitive to changes in c_{χ_1} compared to c_{χ_2}. Nevertheless, this study confirms that the choice made for the numerical examples, namely, $c_{\chi_1} = 100$, $c_{\chi_2} = 10$ (marked by crosses in Fig. 8), is indeed a very suitable trade-off.

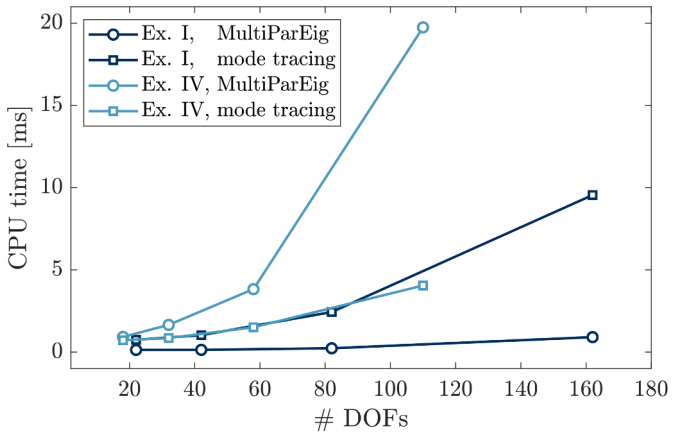

Fig. 7. Computational times per mode and frequency in examples I and IV. The number of degrees of freedom (# DOFs) is varied by increasing the element order p_e. Results are computed using the mode-tracing approach, as well as the formulation based on multiparameter eigenvalue problems (MultiParEig).

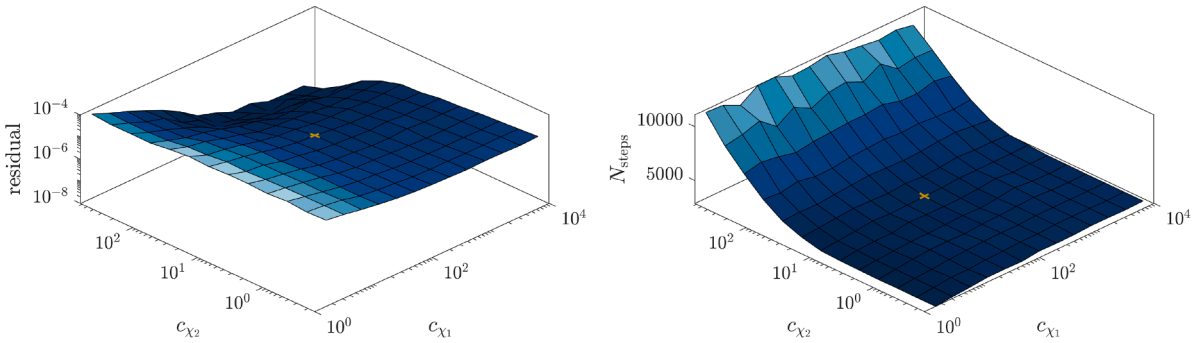

Fig. 8. Median residual and total number of steps taken by the ODE solver when varying the decay parameters c_{χ_1} and c_{χ_2} in example III.

7. Conclusions

We have seen that parameter-dependent nonlinear eigenvalue problems arising in waveguide models can be solved in a rather unconventional way by transforming them into a system of ordinary differential equations and tracing each mode along the frequency axis starting from an initial value. The advantages are that this method involves only the original finite-element matrices of the waveguide problem (in contrast to the operator determinants assembled in the solution of the multi-parameter eigenvalue problem), can exploit standard ODE solvers, and allows the computation of individual modes. Perhaps most importantly, the method is remarkably robust even when only approximate solutions are available as starting values. This property makes the application of this idea worth exploring in various other scenarios where a nonlinear eigenvalue problem is challenging to solve. In principle, this approach is applicable to *differentiable* parameter-dependent eigenvalue problems, provided sufficiently accurate initial guesses can be obtained so that the ODE solver follows the desired solution. In some cases where the eigenvalue problem is not differentiable everywhere, a suitable remedy can be obtained by introducing a small perturbation, here represented by artificial material damping. Slight modifications of the presented approach will be necessary when two or more eigencurves are identical.

CRediT authorship contribution statement

Hauke Gravenkamp: Writing – original draft, Visualization, Validation, Software, Investigation, Conceptualization; **Bor Plestenjak:** Writing – review & editing, Methodology, Conceptualization; **Daniel A. Kiefer:** Writing – review & editing, Validation, Conceptualization.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

Bor Plestenjak has been supported by the Slovenian Research and Innovation Agency (grant P1-0294). Daniel A. Kiefer has received support under the program "Investissements d'Avenir" launched by the French Government under Reference No. ANR-10-LABX-24.

Appendix A. Material parameters

The elastic constants of the isotropic materials used in the numerical examples (Section 6.2) are listed in Table A.1. In addition, example V involves the CFRP composite material T800-913 in a $[0/90]_{2s}$ layup, i.e., the plate consists of eight layers with the fiber orientations $0°/90°/0°/90°|90°/0°/90°/0°$ where $0°$ corresponds to fiber orientation along the $x-axis$ (direction of guided wave propagation), and $90°$ fiber orientation along the y-axis. For easier reference, we provide the stiffness matrix in Voigt notation for both orientations, see also [59]:

$$\mathbf{C}_{0°} = \begin{bmatrix} 154 & 3.7 & 3.7 & 0 & 0 & 0 \\ 3.7 & 9.5 & 5.2 & 0 & 0 & 0 \\ 3.7 & 5.2 & 9.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4.2 \end{bmatrix}, \quad \mathbf{C}_{90°} = \begin{bmatrix} 9.5 & 3.7 & 5.2 & 0 & 0 & 0 \\ 3.7 & 154 & 3.7 & 0 & 0 & 0 \\ 5.2 & 3.7 & 9.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4.2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2.15 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4.2 \end{bmatrix} \tag{A.1}$$

In our example V, we considered 1 % material damping, which was incorporated by replacing the above stiffness matrices by $\mathbf{C}_{0°}(1 - 0.01\,\mathrm{i})$ and $\mathbf{C}_{90°}(1 - 0.01\,\mathrm{i})$, respectively.

Table A.1
Overview of material parameters used in the numerical experiments.

	density ρ	wave speeds c_ℓ, c_t		Lamé parameters λ, G	
brass	8.40 g/cm^3	4.40 km/s	2.20 km/s	81.312 GPa	40.656 GPa
Teflon	2.20 g/cm^3	1.35 km/s	0.55 km/s	2.679 GPa	0.666 GPa
titanium	4.46 g/cm^3	6.06 km/s	3.23 km/s	70.726 GPa	46.531 GPa
steel	7.85 g/cm^3	5.92 km/s	3.22 km/s	112.403 GPa	81.395 GPa
water	1.00 g/cm^3	1.48 km/s			
oil	0.87 g/cm^3	1.74 km/s			

References

[1] M.J.S. Lowe, D.N. Alleyne, P. Cawley, Defect detection in pipes using guided waves, Ultrasonics 36 (1998) 147–154. https://doi.org/10.1016/S0041-624X(97)00038-3
[2] J. Rose, S.P. Pelts, M.J. Quarry, A comb transducer model for guided wave NDE, Ultrasonics 36 (1998) 163–169. https://doi.org/10.1016/S0041-624X(97)00042-5
[3] M. Sale, P. Rizzo, A. Marzani, Semi-analytical formulation for the guided waves-based reconstruction of elastic moduli, Mech. Syst. Signal Process. 25 (2011) 2241–2256.
[4] A. Marzani, L. De Marchi, Characterization of the elastic moduli in composite plates via dispersive guided waves data and genetic algorithms, J. Intell. Mater. Syst. Struct. 24 (17) (2012) 2135–2147. https://doi.org/10.1177/1045389X12462645
[5] E. Kausel, J.M. Roësset, Semianalytic hyperelement for layered strata, J. Eng. Mech. Div. 103 (4) (1977) 569–588. https://doi.org/10.1061/jmcea3.0002251
[6] H. Gravenkamp, B. Plestenjak, D.A. Kiefer, E. Jarlebring, Computation of leaky waves in layered structures coupled to unbounded media by exploiting multiparameter eigenvalue problems, J. Sound Vib. 596 (2025) 118716. https://doi.org/10.1016/j.jsv.2024.118716
[7] L.M. Flitman, On the motion of a rigid strip-mass lying on an elastic half-space and excited by a seismic wave, J. Appl. Math. Mech. 26 (6) (1962) 1043–1058. https://doi.org/10.1016/0021-8928(62)90194-6
[8] B.W. Drinkwater, M. Castaings, B. Hosten, The interaction of Lamb waves with solid-solid interfaces, in: Rev. Progr. Quant. Nondestr. Eval., 22, 2003, pp. 1064–1071.
[9] E. Pistone, P. Rizzo, On the use of an array of ultrasonic immersion transducers for the nondestructive testing of immersed plates, Nondestr. Test. Eval. 30 (1) (2015) 26–38. https://doi.org/10.1080/10589759.2014.979817
[10] R. Long, M. Lowe, P. Cawley, Attenuation characteristics of the fundamental modes that propagate in buried iron water pipes, Ultrasonics 41 (2003) 509–519.
[11] N. Marcuvitz, On field representations in terms of leaky modes or Eigenmodes, IRE Trans. Anntenas Propag. 4 (3) (1956) 192–194. https://doi.org/10.1109/TAP.1956.1144410
[12] H. Lamb, On waves in an elastic plate, Proc. R. Soc. London 93 (648) (1917) 114–128. https://doi.org/10.1098/rspa.1917.0008
[13] L. Knopoff, A matrix method for elastic wave problems, Bull. Seismol. Soc. Am. 54 (1) (1964) 431–438. https://doi.org/10.1785/bssa0540010431
[14] A.H. Nayfeh, The general problem of elastic wave propagation in multilayered anisotropic media, J. Acoust. Soc. Am. 89 (4) (1991) 1521–1531. https://doi.org/10.1121/1.400988
[15] M.J.S. Lowe, Matrix techniques for modeling ultrasonic waves in multilayered media, IEEE Trans. Ultrason. Ferroelectr. Freq. Control 42 (4) (1995) 525–542. https://doi.org/10.1109/58.393096
[16] R.B. Nelson, On natural vibrations and waves in laminated orthotropic plates, J. Appl. Mech. 9 (1972) 739–745.

[17] J.M. Galán, R. Abascal, Numerical simulation of Lamb wave scattering in semi-infinite plates, Int. J. Numer. Methods Eng. 53 (2002) 1145–1173. https://doi.org/10.1002/nme.331
[18] E. Kausel, Wave propagation in anisotropic layered media, Int. J. Numer. Methods Eng. 23 (1986) 1567–1578. https://doi.org/10.1002/nme.1620230811
[19] E. Kausel, Accurate stresses in the thin-layer method, Int. J. Numer. Methods Eng. 61 (2004) 360–379. https://doi.org/10.1002/nme.1067
[20] H. Gravenkamp, C. Song, J. Prager, A numerical approach for the computation of dispersion relations for plate structures using the scaled boundary finite element method, J. Sound Vib. 331 (2012) 2543–2557. https://doi.org/10.1016/j.jsv.2012.01.029
[21] U. Basu, A.K. Chopra, Perfectly matched layers for transient elastodynamics of unbounded domains, Int. J. Numer. Methods Eng. 59 (2004) 1039–1074. https://doi.org/10.1002/nme.896
[22] E.A. Skelton, S.D.M. Adams, R.V. Craster, Guided elastic waves and perfectly matched layers, Wave Motion 44 (2007) 573–592. https://doi.org/10.1016/j.wavemoti.2007.03.001
[23] A. Vaziri Astaneh, M.N. Guddati, Efficient computation of dispersion curves for multilayered waveguides and half-spaces, Comput. Methods Appl. Mech. Eng. 300 (2016) 27–46. https://doi.org/10.1016/j.cma.2015.11.019
[24] A.-S. Bonnet-Ben Dhia, C. Chambeyron, G. Legendre, On the use of perfectly matched layers in the presence of long or backward propagating guided elastic waves, Wave Motion 51 (2014) 266–283. https://doi.org/10.1016/j.wavemoti.2013.08.001
[25] K.L. Nguyen, F. Treyssède, C. Hazard, Numerical modeling of three-dimensional open elastic waveguides combining semi-analytical finite element and perfectly matched layer methods, J. Sound Vib. 344 (2015) 158–178. https://doi.org/10.1016/j.jsv.2014.12.032
[26] M. Castaings, M.J.S. Lowe, Finite element model for waves guided along solid systems of arbitrary section coupled to infinite solid media, J. Acoust. Soc. Am. 123 (2) (2008) 696–708. https://doi.org/10.1121/1.2821973
[27] M. Mazzotti, I. Bartoli, A. Marzani, E. Viola, A coupled SAFE-2.5D BEM approach for the dispersion analysis of damped leaky guided waves in embedded waveguides of arbitrary cross-section, Ultrasonics 53 (7) (2013) 1227–1241. https://doi.org/10.1016/j.ultras.2013.03.003
[28] H. Gravenkamp, C. Birk, C. Song, Computation of dispersion curves for embedded waveguides using a dashpot boundary condition, J. Acoust. Soc. Am. 135 (3) (2014) 1127–1138. https://doi.org/10.1121/1.4864303
[29] T. Hayashi, D. Inoue, Calculation of leaky Lamb waves with a semi-analytical finite element method, Ultrasonics 54 (2014) 1460–1469. 24838216 https://doi.org/10.1016/j.ultras.2014.04.021
[30] H. Gravenkamp, C. Birk, C. Song, Numerical modeling of elastic waveguides coupled to infinite fluid media using exact boundary conditions, Comput. Struct. 141 (2014) 36–45. https://doi.org/10.1016/j.compstruc.2014.05.010
[31] S. Güttel, F. Tisseur, The nonlinear eigenvalue problem, Acta Numer. 26 (2017) 1–94. https://doi.org/10.1017/S0962492917000034
[32] E. Jarlebring, G. Mele, O. Runborg, The waveguide eigenvalue problem and the tensor infinite Arnoldi method, SIAM J. Sci. Comput. 39 (2017) A1062–A1088. https://doi.org/10.1137/15M10446
[33] D.A. Kiefer, M. Ponschab, S.J. Rupitsch, M. Mayle, Calculating the full leaky Lamb wave spectrum with exact fluid interaction, J. Acoust. Soc. Am. 145 (6) (2019) 3341–3350. https://doi.org/10.1121/1.5109399
[34] A. Muhič, B. Plestenjak, On the quadratic two-parameter eigenvalue problem and its linearization, Linear Algebra Appl. 432 (10) (2010) 2529–2542. https://doi.org/10.1016/j.laa.2009.12.022
[35] B. Plestenjak, MultiParEig (version 2.7.0.0), 2023, (n.d.). MATLAB Central File Exchange, www.mathworks.com/matlabcentral/fileexchange/47844-multipareig.
[36] Y. Zhang, J. Wang, Recurrent neural networks for nonlinear output regulation, Automatica 37 (2001) 1161–1173. https://doi.org/10.1016/S0005-1098(01)00092-9
[37] Y. Zhang, D. Jiang, J. Wang, A recurrent neural network for solving Sylvester equation with time-varying coefficients, IEEE Trans. Neural Netw. 13 (5) (2002). https://doi.org/10.1109/TNN.2002.1031938
[38] F. Uhlig, Y. Zhang, Time-varying matrix eigenanalyses via Zhang neural networks and look-ahead finite difference equations, Linear Algebra Appl. 580 (2019) 417–435. https://doi.org/10.1016/j.laa.2019.06.028
[39] F. Uhlig, Zhang neural networks: an introduction to predictive computations for discretized time-varying matrix problems, Numer. Math. 156 (2024) 691–739. https://doi.org/10.1007/s00211-023-01393-5
[40] A. Cichocki, R. Unbehauen, Neural networks for solving systems of linear equations. II. minimax and least absolute value problems, IEEE Trans. Circuits Syst. II Analog Digit. Signal Process. 39 (9) (1992) 619–633. https://doi.org/10.1109/82.193316
[41] J. Wang, Recurrent neural networks for solving linear matrix equations, Comput. Math. Appl. 26 (9) (1993) 23–34. https://doi.org/10.1016/0898-1221(93)90003-E
[42] L. Jin, S. Li, B. Liao, Z. Zhang, Zeroing neural networks: a survey, Neurocomputing 267 (2017) 597–604. https://doi.org/10.1016/j.neucom.2017.06.030
[43] J. Sun, S. Wang, K. Wang, Zhang neural networks for a set of linear matrix inequalities with time-varying coefficient matrix, Inf. Process. Lett. 116 (2016) 603–610. https://doi.org/10.1016/j.ipl.2016.04.014
[44] J.R.H. Otter, Computations for prestressed concrete reactor pressure vessels using dynamic relaxation, Nucl. Struct. Eng. 1 (1965) 61–75.
[45] P.A. Madsen, H.A. Schäffer, A discussion of artificial compressibility, Coastal Eng. 53 (2006) 93–98. https://doi.org/10.1016/j.coastaleng.2005.09.020
[46] J. Baumgarte, Stabilization of constraints and integrals of motion in dynamical systems, Comput. Methods Appl. Mech. Eng. 1 (1972) 1–16. https://doi.org/10.1016/0045-7825(72)90018-7
[47] H. Gravenkamp, B. Plestenjak, D.A. Kiefer, Notes on osculations and mode tracing in semi-analytical waveguide modeling, Ultrasonics 135 (2023) 107112. https://doi.org/10.1016/j.ultras.2023.107112
[48] T. Maruyama, K. Nakahata, Continuation approach combined with semi-analytical finite-element method for solving guided-wave dispersion equation, J. Nondestr. Eval. 44 (55) (2025) 1–12. https://doi.org/10.2139/ssrn.4873730
[49] B. Nennig, E. Perrey-Debain, A high order continuation method to locate exceptional points and to compute Puiseux series with applications to acoustic waveguides, J. Comput. Phys. 412 (2020) 109425. 1909.11579 https://doi.org/10.1016/j.jcp.2020.109425
[50] N. Wagner, L. Gaul, Parameter-dependent matrix eigenvalue problems and their applications in structural dynamics, in: EURODYN International Conference on Structural Dynamics, 2005, pp. 2183–2188.
[51] H. Gravenkamp, B. Plestenjak, On the direct numerical computation of Hopf bifurcations to assess the dynamic stability of fluid-conveying cantilevered pipes, Comput. Struct. 320 (2026) 108039. https://doi.org/10.1016/j.compstruc.2025.108039
[52] H. Gravenkamp, SAMWISE: Semi-Analytical Modeling of Waves in Structural Elements, 2024, (n.d.). https://doi.org/10.5281/zenodo.13830671
[53] M.F. Amin, M.I. Amin, A.Y.H. Al-Nuaimi, K. Murase, Wirtinger calculus based gradient descent and Levenberg-Marquardt learning algorithms in complex-valued neural networks, in: Neural Information Processing, Shanghai, China, 2011, pp. 550–559.
[54] D.H. Brandwood, A complex gradient operator and its application in adaptive array theory, IEE Proc. 130 (H) (1983) 11–16.
[55] A. van den Bos, Complex gradient and Hessian, IEE Proc. Vis. Image Signal Process. 141 (6) (1994) 380–382. https://doi.org/10.1049/ip-vis:19941555
[56] T. Lu, Y. Su, A Newton-type method for two-dimensional eigenvalue problems, Numer. Linear Algebra Appl. 29 (2022) e2430. https://doi.org/10.1002/nla.2430
[57] B. Pavlakovic, M.J.S. Lowe, D.N. Alleyne, DISPERSE: a general purpose program for creating dispersion curves, in: Review of Progress in Quantitative NDE, Plenum Press, 1997, pp. 185–192. https://doi.org/10.1007/978-1-4615-5947-4-24
[58] H. Gravenkamp, C. Birk, C. Song, The computation of dispersion relations for axisymmetric waveguides using the scaled boundary finite element method, Ultrasonics 54 (2014) 1373–1385. https://doi.org/10.1016/j.ultras.2014.02.004
[59] A.M.A. Huber, M.G.R. Sause, Classification of solutions for guided waves in anisotropic composites with large numbers of layers, J. Acoust. Soc. Am. 144 (6) (2018) 3236–3251. https://doi.org/10.1121/1.5082299