Sensitivity computations in higher order continuation methods

Isabelle Charpentier, Komlanvi Lampoh

Sensitivity computations in higher order method continuation methods

Isabelle Charpentier[a], Komlanvi Lampoh[b]

[a]*Icube UMR 7357, 300 bd Sébastien Brant, BP 10413, F-67412 Illkirch Cedex, France*
[b]*Centre des Matériaux, Mines ParisTech, 10 rue Henri Desbruères, BP 87, F-91003 EVRY Cedex, France*

Abstract

Sensitivity analysis is a key tool in the study of the relationships between the input parameters of a model and the output solution. Although sensitivity analysis is extensively addressed in the literature, little attention has been brought to the methodological aspects of the sensitivity of nonlinear parametric solutions computed through a continuation technique.

This paper proposes four combinations of sensitivity analysis with continuation and homotopy methods, including sensitivity analysis along solution branches or at a particular point. Theoretical aspects are discussed in the higher order continuation framework Diamant. The sensitivity methods are applied to a thermal ignition problem and some free vibration problems. Remarkable eigenvalue maps are produced for the complex nonlinear eigenvalue problems.

Keywords: continuation, homotopy, sensitivity, automatic differentiation, Diamant, complex nonlinear eigenvalue problem
2010 MSC: 35B60, 35C20, 65D25, 68W30, 90C31

1. Introduction

Modeling in engineering frequently results in nonlinear parameterized problems, the solutions of which depend on a set of modeling parameters p. These may define material properties, a behavior law, a geometry, initial or boundary conditions. Under-determined nonlinear parametric problems include a particular parameter λ to be varied. Their solutions form branches in bifurcation diagrams [28, 35]. Under analyticity assumptions, the solutions of nonlinear problems may be computed through Taylor-based numerical methods and higher order Automatic Differentiation (AD) [7, 23, 2, 33]. Among them, Diamant [13, 10] is targeted to the solution of nonlinear parametric problems by continuation.

In numerous scientific domains, sensitivity analysis is of prime importance in the understanding of the relationships between the model inputs (modeling parameters, initial or boundary conditions, for instance) and the solution or the branch of solutions. Usages are concerned with uncertainty measurements and optimization, for instance. Sensitivity analysis is thus extensively addressed in the literature [17, 29]. Nevertheless, to date, little attention has been brought to the methodological study of the sensitivity of nonlinear parametric solutions. In other words, the sensitivity analysis to a perturbation in the modeling parameter is often addressed qualitatively only, by plotting bifurcation diagrams for different values of the parameters.

The paper discusses several quantitative sensitivity techniques adapted to the higher order continuation and homotopy methods proposed by the Diamant framework. Given a scientific problem, the choice for one or another depends on both the continuation objectives (trajectory computations [39], path planning [4], complex nonlinear eigenvalue calculation [5]) and the usage of the sensitivity results. The implementation partially relies on Diamanlab [11] that takes advantage of the AD generality [21] and provides a graphical user interface for an interactive continuation. The sensitivity drivers are validated on a thermal ignition problem [1] and the damped beam problem described in the NLEVP benchmark [3]. A frequency-dependent damping model is then considered to propose remarkable eigenvalue maps for the complex nonlinear eigenvalue problems.

The layout of the paper is as follows. Section 3 discusses the Diamant framework through a brief presentation of both the continuation driver and the homotopy driver. It moreover provides a short introduction to AD and Diamanlab. Theoretical aspects and possible usages of sensitivity computations are described in Section 4 meanwhile numerical results are reported in Section 5. Section 6 provides conclusions.

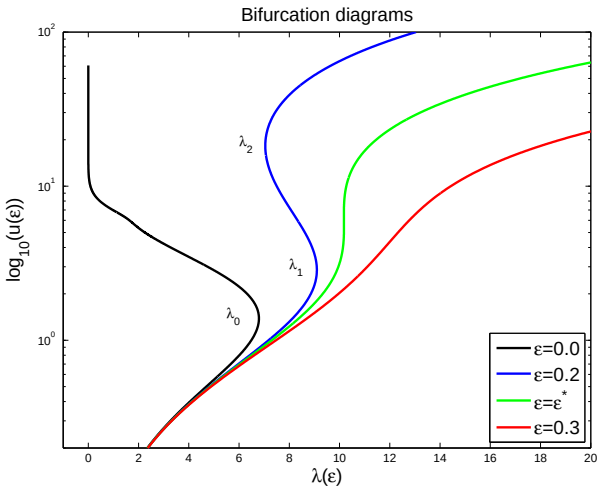

FIGURE 1: Bifurcation diagrams, $u(0.5, 0.5)$ against λ, for the thermal ignition problem.

2. Setting of the problem

The solutions of a nonlinear parametric problem form branches $(u(p), \lambda(p))$ usually represented in a projected bifurcation diagram (Fig. 1) as plots of particular state vector components $u(p)$ against the parameter $\lambda(p)$. The problem of thermal ignition [1] is chosen as an illustration. The temperature distribution in a material with exothermic reaction may be written as

$$\frac{\partial u}{\partial t} = \Delta u + \lambda \exp\left(\frac{u}{1+\epsilon u}\right), \tag{1}$$

where $u = (T - T_s)E_a/(RT_s^2)$ is a normalized temperature variable depending on the temperature T, the surface temperature T_s, the universal gas constant R and the Arrhenius activation energy E_a. In equation (1), the parameter ϵ defined as RT_s/E determines the normalized temperature behavior. The steady-state problem in the unit square domain,

$$\Delta u + \lambda \exp\left(\frac{u}{1+\epsilon u}\right) = 0 \quad \text{in } \Omega = (0,1)\times(0,1), \tag{2}$$

$$u = 0 \quad \text{on the boundary } \partial\Omega, \tag{3}$$

is discretized using the five-point stencil finite difference method for the Laplace operator. Eleven equidistributed points are used in each direction. The discrete parametric problem is solved using Taylor-based expansions with a truncation order of 15 and the Diamanlab software (paragraph 3.3.2). The initial guess (u_0, λ_0) is such that $u_0 \equiv 0$ in Ω and $\lambda_0 = 0$.

Figure 1 plots the solutions for various values of the parameter ϵ. The case $\epsilon = 0$ corresponds to the Gelfand equation and exhibits a turning point at $(u(0.5, 0.5), \lambda_0) \simeq (6.793, 1.385)$. For $0 < \epsilon < \epsilon^*$, the solution branch presents two turning points λ_1 and λ_2. From a physical point of view, the material ignition takes place when the turning point λ_1 is passed. The solution branch degenerates for $\epsilon^* \simeq 0.24174$. This value depends on the geometry of the domain [1]. Former explanations are those of a brief qualitative sensitivity analysis of the solutions of (2)–(3) performed by plotting curves for various values of the parameter ϵ. The quantitative sensitivity analysis of the solution branches requires the computation of the first order derivatives of $(u(p), \lambda(p))$ with respect to p in order to propose accurate first order approximations for the solutions.

Computing quantitative sensitivities is an important issue not only for the understanding of the relationships between the nonlinear parametric problem inputs and the output solution branches, but also in classical uses of sensitivity results such as optimization or uncertainty measurement. This paper proposes different options for an accurate calculation of the sensitivities of the solution branches computed through higher-order continuation methods. Sensitivity

branches (u^d, λ^d) computed using additional Taylor-based expansions are presented in paragraph 4.1. Then, local sensitivities (u_1^p, λ_1^p) computed at particular points are proposed in paragraph 4.2. Numerical results, including the thermal ignition problem, are reported in Section 5. Prior to that, the next section presents the Diamant framework.

3. Higher order continuation framework

Diamant [13, 10] is the AD-based framework derived from the asymptotic numerical method [36, 15]. Under analyticity assumptions, it automates the solution of the nonlinear parametric problems written in the generic form

$$\mathcal{R}(u(a,p),\lambda(a,p)) = 0, \tag{4}$$

$$a =< u(a,p) - u(0,p), \frac{\partial u}{\partial a}(0,p) > + (\lambda(a,p) - \lambda(0,p))\frac{\partial \lambda}{\partial a}(0,p). \tag{5}$$

In system (4)–(5), the residual vector $\mathcal{R} \in \mathbb{C}^n$ depends on the unknown state vector $u(p) \in \mathbb{C}^n$ and the scalar parameter $\lambda(p)$ to be varied. The variable p denotes the modeling parameters. Equation (5) measures the pseudo-arc length along the branch of solutions $(u(a,p), \lambda(a,p))$ issued from $(u(0,p), \lambda(0,p))$. Another path equation may be used to close the under-determined nonlinear residual problem (4). The branch $(u(a,p), \lambda(a,p))$ is approximated using the truncated Taylor series (6) in a vicinity of $a = 0$ following

$$(u(a,p),\lambda(a,p)) \simeq \Big(\sum_{k=0}^{\kappa} \frac{a^k}{k!}\frac{\partial^k u}{\partial a^k}(0,p), \sum_{k=0}^{\kappa} \frac{a^k}{k!}\frac{\partial^k \lambda}{\partial a^k}(0,p)\Big) = \Big(\sum_{k=0}^{\kappa} a^k u_k, \sum_{k=0}^{\kappa} a^k \lambda_k\Big), \tag{6}$$

where κ is the truncation order and u_k and λ_k denote the Taylor coefficients of u and λ at order k, respectively. The Taylor coefficients of $\mathcal{R}$ are denoted by $\mathcal{R}_k$ and verify

$$\mathcal{R}(u(a,p),\lambda(a,p)) \simeq \sum_{k=0}^{\kappa} a^k \mathcal{R}_k = 0. \tag{7}$$

For the sake of conciseness, the dependence of u, λ and $\mathcal{R}$ with respect to a and p is omitted in the description of the continuation driver (paragraph 3.1) and the homotopy driver (paragraph 3.2). From a theoretical and computer point of view, AD (paragraph 3.3) is a key tool for the evaluation of the Taylor coefficients.

3.1. Continuation driver

The analyticity of $\mathcal{R}$ implies that

$$\mathcal{R}_k = 0, \quad \text{for } k = 1 \text{ to } \kappa. \tag{8}$$

The truncated series (6) are introduced in the nonlinear problem (4). Using the Faà di Bruno generalized formula [30], the Taylor coefficients $\mathcal{R}_k$ of the compound function $\mathcal{R}$ are expressed as

$$\mathcal{R}_k = \{\mathcal{R}_{1|u_1=Id,\lambda_1=1}\}(u_k,\lambda_k)^T + \{\mathcal{R}_{k|u_k=0,\lambda_k=0}\} = 0, \quad \text{for } k = 1 \text{ to } \kappa, \tag{9}$$

and comprise the same Jacobian $\{\mathcal{R}_{1|u_1=Id,\lambda_1=1}\}$ over the orders. This Jacobian contains the tangent linear contributions resulting from the differentiation of $\mathcal{R}$ with respect to u and λ. The remaining terms $\{\mathcal{R}_{k|u_k=0,\lambda_k=0}\}$ contain the nonlinear contributions depending on the Taylor coefficients u_l and λ_l, for $l = 1$ to $k-1$. The resulting κ linear systems,

$$\{\mathcal{R}_{1|u_1=Id,\lambda_1=1}\}(u_k,\lambda_k)^T = -\{\mathcal{R}_{k|u_k=0,\lambda_k=0}\}, \quad \text{for } k = 1 \text{ to } \kappa, \tag{10}$$

$$a = (u_k - u_0)u_1 + (\lambda_k - \lambda_0)\lambda_1, \quad \text{for } k = 1 \text{ to } \kappa. \tag{11}$$

allow to determine a higher order approximation for the solution (u, λ) of the generic problem (4)–(5). The solution of (10)–(11) is performed in an iterative manner from $k = 1$ to κ, by alternating higher order derivative computations and linear system solutions. The continuation driver (10)–(11) is thus a high level mathematical solver. Implementation detail is reported in [11].

As discussed in previous papers, Diamant is a generic and competitive path following method that may replace the Newton-Raphson scheme in the solution of any equilibrium system of smooth equations. The step size control is based on the *a posteriori* estimation (12) of the range of validity of the series

$$a_{max} = \left(\beta \frac{|u_1|}{|u_\kappa|}\right)^{1/\kappa - 1}, \tag{12}$$

where β is a small parameter (usually $\beta = 1.10^{-6}$). Residuals are measured at the end of the computed branch of solutions to evaluate the interest of a correction. In practice, the higher order predictor Diamant is frequently run without the need for correction steps.

Best practices are concerned with nonlinear mechanical problems involving material or geometrical nonlinearities [15, 10].

3.2. Homotopy driver

A homotopy is a continuous deformation that allows for the solution of a nonlinear problem from a known solution using a continuation method [27, 35]. Among the best practices relying on homotopy, one may cite trajectory computations [39], path planning [4], and complex nonlinear eigenvalue problems [5].

In the Diamant framework, the homotopy related to the nonlinear residual problem (4) is written [5] as

$$\mathcal{R}(u(a), \lambda(a)) = \mathcal{S}(u(a), \lambda(a)) + a\mathcal{T}(u(a), \lambda(a)) = 0, \quad \text{for} \quad a \in (0, 1). \tag{13}$$

This homotopy is run from a known solution $(u(0), \lambda(0))$ of the problem

$$\mathcal{S}(u(0), \lambda(0)) = 0, \quad \text{for } a = 0, \tag{14}$$

to the desired solution $(u(1), \lambda(1))$ of the residual problem (4), for $a = 1$. The function $\mathcal{T}(u, \lambda)$ defines as

$$\mathcal{T}(u, \lambda) = \mathcal{R}(u, \lambda) - \mathcal{S}(u, \lambda), \tag{15}$$

usually contains the nonlinearities. The use of the Taylor series (6) in equation (13) yields generic linear systems

$$(\{\mathcal{S}_{1|u_1=Id,\lambda_1=1}\} + a_{(j)}\{\mathcal{T}_{1|u_1=Id,\lambda_1=1}\})(u_k, \lambda_k)^T = -\{\mathcal{S}_{k|u_k=0,\lambda_k=0}\} - a_{(j)}\{\mathcal{T}_{k|u_k=0,\lambda_k=0}\} - \mathcal{T}_{k-1}, \quad \forall\, k = 1, .., \kappa, \tag{16}$$

that are similar to the linear systems (10). In (16), the Jacobian related to the j^{th} branch is denoted by $\mathcal{J}_{(j)}$, that is

$$\mathcal{J}_{(j)} = (\{\mathcal{S}_{1|u_1=Id,\lambda_1=1}\} + a_{(j)}\{\mathcal{T}_{1|u_1=Id,\lambda_1=1}\}). \tag{17}$$

The linear systems (16) are closed with either a path equation or some additional constraint. The choice of $\mathcal{S}$ and $\mathcal{T}$ and the closure equation is problem-dependent. Other computational aspects may be automated.

Initially, this homotopy driver was developed for the solution of the generic complex nonlinear eigenvalue problem (18) arising in the free vibration modeling of viscoelastic structures [5]

$$\mathcal{R}(u, \lambda) = (K + (D_0 + \mathcal{D}(\lambda)) - \lambda M)u = 0, \tag{18}$$

assuming that the eigensolutions satisfy $u(x, t) = u(x)e^{i\omega t}$ and $\omega^2 = \lambda$. In equation (18), the stiffness matrix K and the mass matrix M are real symmetric positive definite. The damping term $D_0 + \mathcal{D}(\lambda)$ is a complex symmetric positive semidefinite matrix that may depend on λ in a nonlinear fashion. In [5], the homotopy is run from one of the eigensolutions of the real problem (19),

$$\mathcal{S}(u, \lambda) = (K + D_0 - \lambda M)u, \tag{19}$$

$$\mathcal{T}(u, \lambda) = \mathcal{D}(\lambda)u, \tag{20}$$

to the related eigensolution of the complex problem (18). The mass orthonormalization of the eigenvectors

$$\lambda_k = -\frac{u_0^T\left(\{\mathcal{S}_{k|u_k=0,\lambda_n=0}\} + a_{(j)}\{\mathcal{T}_{k|u_k=0,\lambda_k=0}\} + \mathcal{T}_{k-1}\right)}{u_0^T(\{\mathcal{S}_{1|u_1=0,\lambda_1=1}\} + a_{(j)}\{\mathcal{T}_{1|u_1=0,\lambda_1=1}\})}, \tag{21}$$

is used to deduce the closure equation.

3.3. Automatic differentiation (AD)

Given a simulation code, sensitivity computations and higher order derivatives may be implemented using Automatic Differentiation [21]. AD tools apply the classical differentiation rules such as "the derivative of a product of two functions is the derivative of the first times the second, plus the derivative of the second times the first", and so forth. On the one hand, the source transformation tools [24, 6] propose to augment the user code with statements encoding the first order derivative computation in tangent linear mode or adjoint mode. The generated code is readable, notably in the tangent linear mode. Some of these tools propose a second order differentiation too. On the other hand, the derivative computations may be hidden in an operator overloading library that includes specific data types as well as methods that encode the classical differentiation rules. The user code is linked to the library at compile time (or at interpretation time) to enable derivative computations at run time. In both cases, the derivative calculation is correct to machine precision.

AD is undoubtedly the more practicable approach for the higher order differentiation, combining generality and ease of use in efficient tools [18, 34, 14, 2, 14]. The higher order AD relies on operator overloading as the vehicle of attaching recurrent derivative computations to the arithmetic operators and intrinsic functions provided by the programming language. A comprehensive overview of the techniques, tools and usage is available on the AD community's website `www.autodiff.org`. Main applications are sensitivity analysis, gradient-based optimization and nonlinear problem solution.

3.3.1. AD usages in the Diamant continuation framework

Devoted to the generic solution of nonlinear parametric problems, the Diamant drivers rely on the recurrence formulas (10) and (16), respectively. Each of these sequences of linear systems involves a Jacobian and higher order derivatives. The Jacobian calculation is a well-known issue in AD. It may be achieved using a tangent linear code together with either the canonical basis for small systems, or a sparse Jacobian evaluation for larger ones. Within Diamant, the Jacobian and the higher order terms are computed using an AD operator overloading library. Consequently, Diamant inherits from AD's generality and ease of use.

It is well-known that a particular differentiation is mandatory for an efficient AD of high level mathematical operations such as linear solvers, fast Fourier/wavelet transforms and nonlinear solvers. This also applies to the Diamant continuation drivers.

3.3.2. Diamanlab

Diamanlab [11] is an object-oriented software in Matlab targeted to the interactive solution of differential algebraic equation systems. It implements the continuation driver (10). Solutions branches of the nonlinear problem under study are computed as a set of Taylor-based solutions stored in checkpoints [20, 8]. Each checkpoint contains the Taylor series information (starting point $(u(0), \lambda(0))$, series and range of validity) and an oriented tangent vector that indicates the continuation direction. Room may be added to collect complementary information such as sensitivities. During the continuation process, the end point of the last computed branch becomes the next checkpoint unless the user interacts with the plotted bifurcation diagram to set another point and to continue.

From a developer point of view, the core Diamanlab classes are concerned with the handling of the generic nonlinear problem $\mathcal{R}$, the higher-order AD, the checkpoint management and the interactive continuation (graphical user interface). The higher order differentiation methods are gathered in the *@Taylor* class. The interested reader is referred to [11] for detail. Diamanlab offers generality and ease of use at the user level too. Any analytical nonlinear satisfying (4) may be solved deriving a user class from the generic system class. For the sake of reproducibility, Diamanlab includes a benchmark of classical nonlinear problems. These may serve as a basis for the implementation of a new user problem.

The sensitivity driver (23)–(24) presented in paragraph 4.1 has been implemented introducing new classes based on the existing Diamanlab classes. The sensitivity driver benefits from the graphical user interface. For now, the homotopy sensitivity driver makes use of the *@Taylor* class only.

4. Sensitivity computations and usage

Sensitivity analysis and continuation methods are two of the major pillars in computational sciences and engineering. Designing AD-based numerical methods for accurate quantitative sensitivity computations is then a natural

objective. Several options exist. First, sensitivity branches (u^d, λ^d) may be approximated using series expanded in the path variable. Second, local sensitivities denoted by (u_1^p, λ_1^p) may be computed at particular points. These local sensitivities do not depend on the path variable. Both methods are discussed for the continuation and the homotopy drivers, together with their benefits and usages.

4.1. Sensitivity branches

The Diamant continuation drivers (10) and (16) are differentiated with respect to the modeling parameter p in order to compute sensitivities as solution branches (u^d, λ^d). Such a process involves the Taylor coefficients (u_k^d, λ_k^d),

$$u_k^d = \frac{\partial u_k}{\partial p} = \Big(\frac{\partial^{k+1} u}{\partial p \partial a^k}\Big)(0,p) \quad \text{and} \quad \lambda_k^d = \frac{\partial \lambda_k}{\partial p} = \Big(\frac{\partial^{k+1} \lambda^d}{\partial p \partial a^k}\Big)(0,p), \quad \text{for } k = 1,..,\kappa, \tag{22}$$

that are mixed derivatives of first order in p and of higher order in a.

4.1.1. Continuation

The tangent linear differentiation of (10) with respect to p yields the κ linear systems [31]

$$\{\mathcal{R}_{1|u_1=Id,\lambda_1=1}\}(u_k^d, \lambda_k^d)^T = -\{\mathcal{R}^d_{k|u_k=0,\lambda_k=0}\} - \{\mathcal{R}^d_{1|u_1=Id,\lambda_1=1}\}(u_k, \lambda_k)^T, \quad \forall\, k = 1,..,\kappa, \tag{23}$$

$$a^d = (u_k^d - u_0^d)u_1 + (u_k - u_0)u_1^d + (\lambda_k^d - \lambda_0^d)\lambda_1 + (\lambda_k - \lambda_0)\lambda_1^d, \quad \forall\, k = 1,..,\kappa. \tag{24}$$

These systems involve the Jacobian exhibited in the continuation driver and may be solved at almost no extra cost. The sensitivity branches (u^d, λ^d) are obtained from a differentiation of Eq. (6), that is

$$(u^d(a,a^d), \lambda^d(a,a^d)) = \Big(\sum_{k=0}^{\kappa} a^k u_k^d + k a^{k-1} a^d u_k, \sum_{k=0}^{\kappa} a^k \lambda_k^d + k a^{k-1} a^d \lambda_k\Big). \tag{25}$$

From a computer point of view, mixed derivatives [12, 9, 31, 38] may be computed using a source transformation AD tool onto the code implementing $\mathcal{R}$ (the actual user code) to generate a tangent linear code $\mathcal{R}^d$. The sensitivity driver implemented using Diamanlab solves equations (10)–(11) as well as (23)–(24). The higher-order AD of $\mathcal{R}^d$ is performed at interpretation time by defining u^d and λ^d as *@Taylor* objects. The higher order tensor algorithm [19], implemented within Rapsodia [14], is less appropriate for the evaluation of such mixed derivatives because the Diamant drivers alternate derivative computations with linear system solutions.

The sensitivity driver is new to Diamanlab. It is used in paragraph 5.1 to plot branches of sensitivities for the thermal ignition problem. Some sensitivity results obtained for different viscoelastic sandwich beams and plates are reported in [31].

4.1.2. Homotopy continuation

The homotopy formulation (16) is differentiated with respect to the modeling parameter p following the differentiation process presented in paragraph 4.1.1. The differentiation of (16) and (17) yields

$$\mathcal{J}^d_{(j)} = \{\mathcal{S}^d_{1|u_1=Id,\lambda_1=1}\} + a_{(j)}\{\mathcal{T}^d_{1|u_1=Id,\lambda_1=1}\} + a^d_{(j)}\{\mathcal{T}_{1|u_1=Id,\lambda_1=1}\}, \tag{26}$$

$$\mathcal{J}_{(j)}(u_k^d, \lambda_k^d)^T = -\{\mathcal{S}^d_{k|u_k=0,\lambda_k=0}\} - a_{(j)}\{\mathcal{T}^d_{k|u_k=0,\lambda_k=0}\} - a^d_{(j)}\{\mathcal{T}_{k|u_k=0,\lambda_k=0}\} - \mathcal{T}^d_{k-1} - \mathcal{J}^d_{(j)}(u_k, \lambda_k)^T, \quad \forall\, k = 1,..,\kappa. \tag{27}$$

The closure equation is to be differentiated too.

The sensitivity of solution branches is of special interest when the trajectory computation is the main objective of the homotopy [25, 39], including the evolution of the complex eigenvalues with respect to a particular modeling parameter as proposed in paragraph 5.2.2.

4.2. Local sensitivities

Sensitivities along branches may have limited interest when the goal of the homotopy is the solution at the end point only. Likewise, the sensitivity of solution branches computed and plotted using an interactive continuation software may be questionable when the user operates a “jump” from one point of the bifurcation diagram to another, for instance. The sensitivity drivers should thus allow for a local sensitivity analysis, at either a user-defined checkpoint or at the end point of the homotopy method. In both cases, the differentiation stages in a and p are decoupled and there are no more mixed derivatives to compute.

4.2.1. Continuation

Let $(u(p), \lambda(p))$ be a known solution point. An approximated solution of (4) may be computed for a small perturbation of p in the direction of the unit vector p_1 as

$$(u(p + \alpha p_1), \lambda(p + \alpha p_1)) = (u(p) + \alpha u_1^p(p)), \lambda(p) + \lambda_1^p(p)), \tag{28}$$

where α is a small real parameter and the sensitivity $(u_1^p(p), \lambda_1^p(p))$ is the first order derivative of $(u(p), \lambda(p))$ computed with respect to p. From a numerical point of view, one may use either the tangent linear code $\mathcal{R}^d$, or the code $\mathcal{R}$ overloaded for a differentiation with respect to p.

At a given point, local sensitivities do not depend on the path variable a and the path equation (5) cannot be used to close the system. As in classical parameterized problems, the local sensitivity of u may be computed for a given λ. For the sake of completeness, it is important to mention that higher-order local approximations may be obtained following the asymptotic analysis proposed in [23].

4.2.2. Homotopy

The differentiation of the homotopy and the propagation of the derivatives along the path are not mandatory when one is interested in the sensitivity of the end point solution. The particular case of the nonlinear eigenvalue problem with an orthonormality closure constraint (21) is discussed in paragraph 5.2. This local computation requires one extra linear system solution at the end point, but it leaves the Taylor series capabilities available for an optimization method [16, 23, 22] based on a higher order differentiation of the eigenvariables u and λ with respect to the modeling parameter p. From a numerical point of view, the paper focuses on sensitivity results only.

5. Numerical results and discussion

The sensitivity drivers are evaluated on a thermal ignition problem [1] for the continuation one, and on the damped beam problem [26] described in the NLEVP benchmark [3] for the homotopy driver. A frequency-dependent damping model [37, 5] is considered to emphasize the abilities of the Diamant driver and to discuss approximated eigenvalue maps.

5.1. Thermal ignition problem

Branch and checkpoint sensitivity results are proposed for the steady-state problem presented in Section 2. The code related to the discrete thermal ignition problem is denoted by $\mathcal{R}$.

The sensitivity $(u^d, \lambda^d) = \left(\dfrac{\partial u}{\partial \epsilon}, \dfrac{\partial \lambda}{\partial \epsilon}\right)$ of the solution (u, λ) of the thermal ignition model (2)–(3) satisfies

$$\Delta u^d + \left(\lambda^d + \lambda \frac{u^d - u^2 \epsilon^d}{(1 + \epsilon u)^2}\right) \exp\left(\frac{u}{1 + \epsilon u}\right) = 0 \quad \text{in } \Omega = (0, 1) \times (0, 1), \tag{29}$$

$$u^d = 0 \quad \text{on the boundary } \partial\Omega. \tag{30}$$

Equations (29)–(30) are discretized with the five-point stencil finite difference scheme used for the equations (2)–(3) and implemented in the tangent linear code denoted by $\mathcal{R}^d$. This contains the statements of $\mathcal{R}$ as well. In Matlab, the code $\mathcal{R}^d$ is overloaded at interpretation time for the higher AD. The sensitivity driver solves the equations (10)–(11) and (23)–(24) to compute the Taylor coefficients and the branches of sensitivities following (25). For the sake of reproducibility, the sensitivity driver and the user class related to the thermal ignition problem may be obtained from the corresponding author.

The correctness of the sensitivity process is checked with a classical Taylor test. This compares the sensitivity result (u^d, λ^d) obtained with the sensitivity driver at point ϵ in the direction ϵ^d to first order finite difference approximations computed using the continuation driver as follows

$$r_\alpha(u) = \frac{|u(\epsilon + \alpha\epsilon^d) - u(\epsilon)|}{\alpha |u^d(\epsilon, \epsilon^d)|} \quad \text{and} \quad r_\alpha = (\lambda) \frac{|\lambda(\epsilon + \alpha\epsilon^d) - u(\epsilon)|}{\alpha |\lambda^d(\epsilon, \epsilon^d)|}, \quad \text{for small } \alpha \in \boldsymbol{R}. \tag{31}$$

α	$r_\alpha(u)$	$r_\alpha(\lambda)$
1.E-3	1.13957691646213	0.99771091315504
1.E-4	1.00808650725132	0.99988326634830
1.E-5	1.00075453289025	0.99998938980647
1.E-6	1.00007491725486	0.99999894943411
1.E-7	1.00000748384382	0.99999989502055
1.E-8	1.00000072256463	0.99999997858741
1.E-9	0.99999752919078	1.00000005788443
1.E-10	0.99999122129183	0.99999883793030
1.E-11	1.00003853053399	0.99999639802205

TABLE 1: Taylor test on the sensitivity of $(u(0.5, 0.5), \lambda)$ with respect to $(\epsilon, \epsilon^d) = (0.2, 0.1)$ at the end point of the 5^{th} branch.

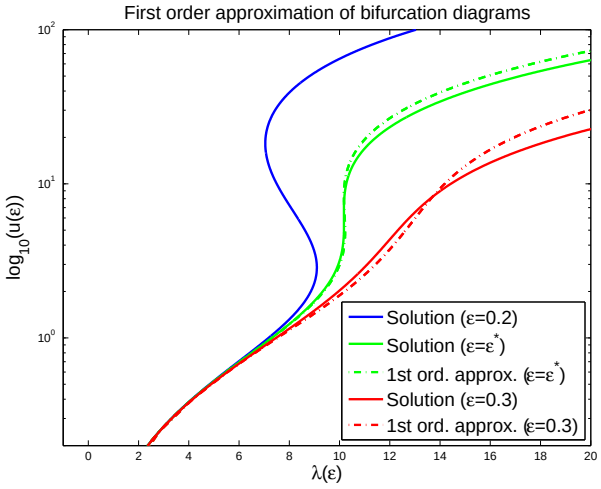

FIGURE 2: Bifurcation diagrams, $u(0.5, 0.5)$ against λ, and their first order approximations for the thermal ignition problem. The sensitivity is computed at $\epsilon = 0.2$. Color online.

Ratios computed at the end point of the fifth branch using $(\epsilon, \epsilon^d) = (0.2, 0.1)$ are reported in Table 1. Theoretically, these ratio should tend linearly toward 1 as α tends to 0 since small α are necessary to minimize the numerical errors coming from the finite difference approximation. However, the subtraction of too close floating-point numbers yields a cancellation error that dominates the truncation error coming from the finite difference computation. Taylor tests presenting such a behavior indicate that the implemented sensitivity computation is correct.

Bifurcation diagrams and their first order approximations are displayed in Fig. 2. Solid lines plot solutions (u, λ) computed for various values of the modeling parameter ϵ (blue for $\epsilon = 0.2$, green for $\epsilon = \epsilon^*$ and red for $\epsilon = 0.3$) using the continuation driver. The solution (u^d, λ^d) is calculated using the sensitivity driver with the parameter $\epsilon = 0.2$ and the perturbation $\epsilon^d = 0.1$. The first order approximated solutions are plotted using dashed lines with same colors. The agreement between the thermal ignition problem solutions and their approximations using the sensitivity computations is good. In particular, the turning point positions are well-reproduced for this highly nonlinear problem (Figs. 2 and 3.a).

In Fig. 3.a, the sensitivity (u^d, λ^d) evaluated along the path for $(\epsilon, \epsilon^d) = (0.2, 0.02)$ is represented as blue dotted lines linking the solution computed at $\epsilon = 0.2$ (blue curve) to the first order approximation $(u + u^d, \lambda + \lambda^d)$ (magenta dashed curve). The solution computed at $\epsilon = 0.22$ is plotted using a magenta solid line. Checkpoints are indicated using circles with same colors. This picture shows that the proposed continuation driver requires very few branches computations, about twenty, each one involving a Jacobian computation. The first order approximation of the solution

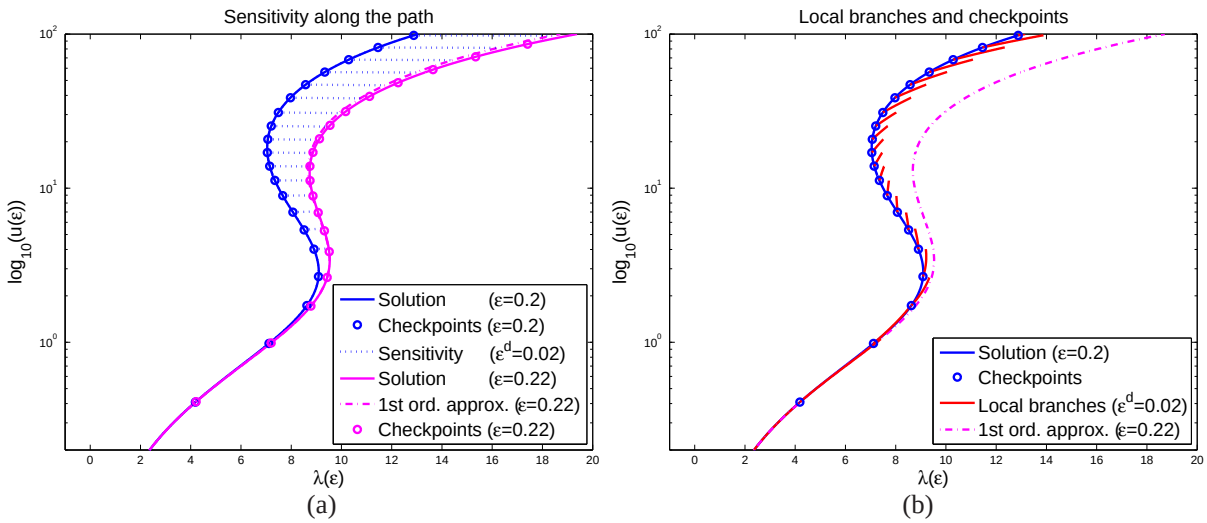

FIGURE 3: Sensitivities of the solution computed at $\epsilon = 0.2$. (a) Along the path, $\epsilon^d = 0.1\epsilon$. (b) : Local branches at checkpoints. Color online.

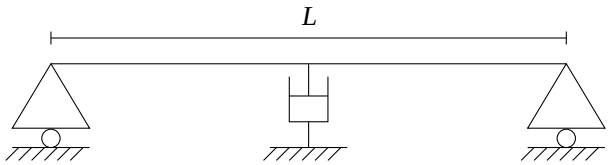

FIGURE 4: Simply supported damped beam

at $\epsilon = 0.22$ is very good. One notices that the checkpoint locations slightly differ due to the adaptive computation of the validity range a_{max}. This is not important in the proposed numerical method. Nevertheless, this may be a significant issue in the sensitivity computations performed by means of finite difference approximations.

Local sensitivity branches, denoted by $(u+u^d, \lambda+\lambda^d)$ too, are evidenced in Fig. 3.b. These are computed initializing (u_0^d, λ_0^d) to (0,0) at checkpoints to cancel the linear accumulation in the first order derivatives.

5.2. Nonlinear eigenvalue problems

Initially, the Diamant homotopy driver is designed for the solution of the complex frequency-dependent eigenvalue problems arising in the free vibration modeling [5]. For the sake of reproducibility, sensitivity computations are illustrated on the `damped_beam` problem [26] belonging to the collection of nonlinear eigenvalue problems NLEVP [3].

5.2.1. NLEVP's damped beam example

The `damped_beam` problem [26] is a quadratic eigenvalue problem (QEP) [26] arising in the free vibration analysis of the simply supported beam, Fig. 4, with a damper located in its middle. The damping coefficient d_c is constant. Assuming that the eigensolutions satisfy $u(x,t) = u(x)e^{\omega t}$, the resulting discrete `damped_beam` QEP is written as

$$Q(\omega)u = (\omega^2 M + \omega D + K)u = 0, \tag{32}$$

where the stiffness matrix K and the mass matrix M are symmetric positive definite, and the viscosity matrix D is symmetric positive semidefinite. The `damped_beam` modeling parameters are reported in Tab. 2. In NLEVP, the quadratic polynomial $Q(\omega)$ is turned into a linear one,

$$L(\omega) = \omega X + Y = 0, \tag{33}$$

Length	$L = 1\ m$	Young modulus	$E = 7 \times 10^{10}\ Pa$
Width	$l = 5 \times 10^{-2}\ m$	Density	$\rho = 0.674\ Kg.m^{-3}$
Height	$H = 5 \times 10^{-3}\ m$	Damping coefficient	$d_c = 5$

TABLE 2: Geometric and material properties of the damped beam.

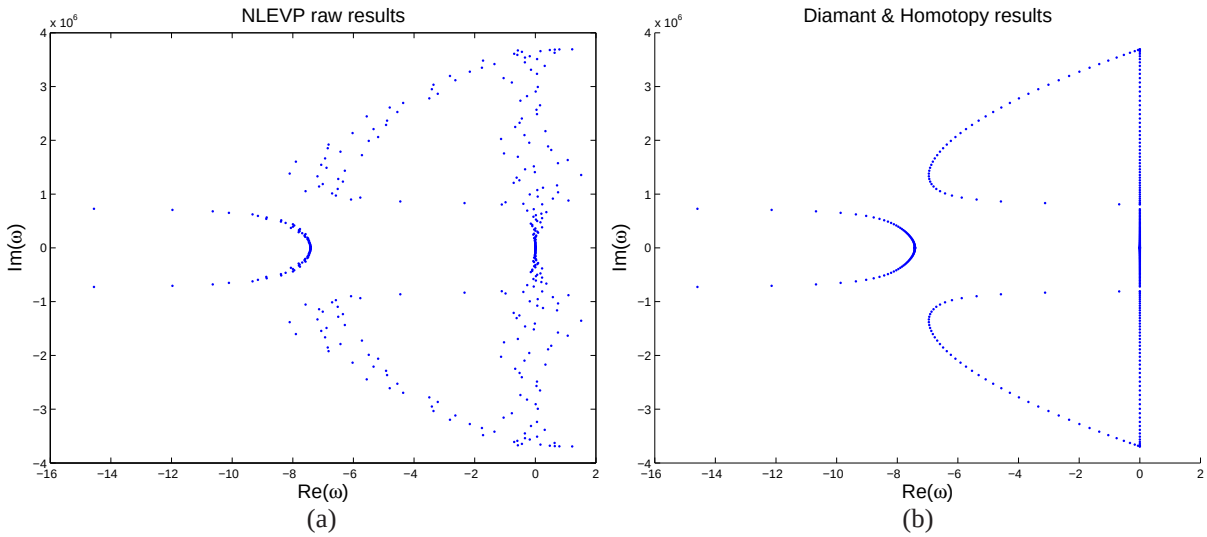

FIGURE 5: Eigensolutions of the NLEVP damped beam problem. (a) : NLEVP raw results. (b) : Homotopy results.

the dimension of which is twice those of $Q(\omega)$. The `damped_beam` raw eigenvalues (Fig. 5.a, no scaling is applied) are obtained concurrently using a QZ algorithm following

$$[\texttt{coeffs}, \texttt{fun}] = \texttt{damped\_beam(200)}; [\texttt{X}, \texttt{e}] = \texttt{polyeig(coeffs\{:\})};$$

where the cell array `coeffs={K,D,M}` contains the problem matrices.

The authors make use of this example to draw attention on important linearization aspects such as scaling, condition numbers and backward errors. Accurate results (Fig. 5.b) may be obtained either by scaling the QEP before the solution using a linearization [26], or by using the homotopy driver as proposed in the next paragraph.

5.2.2. Diamant & homotopy

Substituting ω by $i\sqrt{\lambda}$ into the QEP (32) yields the residual problem satisfying (18)

$$(K + i\sqrt{\lambda}D - \lambda M)u = 0, \tag{34}$$

where K, M and D are the `damped_beam` matrices. The problem (34) is turned into the generic complex nonlinear eigenvalue problem (18) choosing $D_0 = 0$ and $\mathcal{D}(\lambda) = i\sqrt{\lambda}D$. The complex eigensolutions of (34) are determined from the real ones in a sequential manner. A unique branch computation is necessary per eigenvalue. The eigenvalues with positive imaginary parts are computed with Diamant and plotted in Fig. 5.b. The others are reproduced by symmetry. The computed spectrum agrees with those obtained in [26] for the best linearization.

The Matlab version of the Diamant homotopy driver is obviously more time consuming than NLEVP in the solution of (32). Nevertheless, the homotopy driver has a number of precious methodological assets. First, the automation and generality of Diamant and AD allows for the solution of free vibration problems with nontrivial damping laws [5] such as a generalized Maxwell model or a Kelvin-Voigt model. Second, the Taylor-based approximations are obtained with an *a posteriori* error measured and controlled along the path. The linearization phase discussed in [26] is not necessary. Third, Taylor-based approximations provide access to the continuous variation of each of the eigenvalues

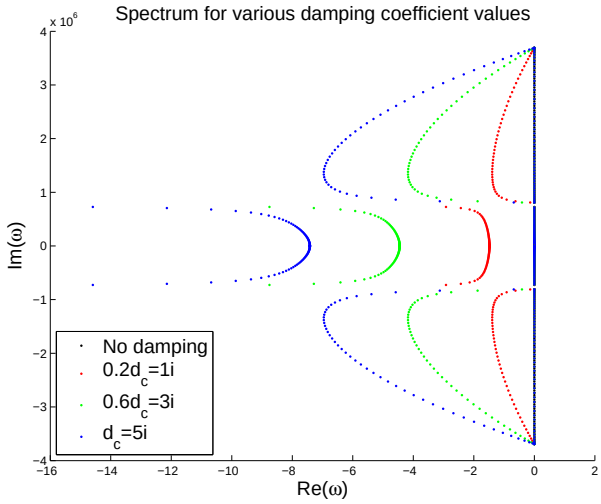

FIGURE 6: Eigensolutions for various values of the constant damping coefficient d_c.

along its homotopy path. For instance, Fig. 6 accounts for the evolution of the spectrum with respect to the damping coefficient ad_c for $a \in \{0, 0.2, 0.6, 1\}$. In other words, the homotopy intrinsically provides a sensitivity analysis with respect to the damping function since it is scaled by the path parameter a.

For illustration purposes, the damping is now achieved considering the frequency-dependent rheological model determined for the viscoelastic material 3M ISD112 at a temperature of 27° [37],

$$G(\omega) = G_0\Big(1 + \sum_{j=1}^{3} \frac{\Delta_j \omega}{\omega - i\Omega_j}\Big), \tag{35}$$

where G_0 is the shear modulus of the delayed elasticity and $\omega = \sqrt{\lambda}$. Parameters (Δ_j, Ω_j) are reported in Tab. 3. The damping is still applied at the middle of the beam using the matrix $\tilde{D} = D/d_c$. As in [5], the real constant part D_0 and frequency-dependent part $\mathcal{D}(\omega)$ of the damping function (including the height of $H = 5 \times 10^{-3}$ m),

$$D_0 + \mathcal{D}(\omega) = HG(\omega)\tilde{D}, \tag{36}$$

are split into the homotopy functions $\mathcal{S}$ and $\mathcal{T}$ as follows

$$\mathcal{S}(u, \lambda) = (K + HG_0\tilde{D} - \lambda M)u, \quad \text{and} \quad \mathcal{T}(u, \lambda) = HG_0\Big(\sum_{j=1}^{3} \frac{\Delta_j \omega}{\omega - i\Omega_j}\Big)\tilde{D}u. \tag{37}$$

Nonlinear mechanics results obtained at homotopy end points are discussed in [5] for viscoelastic sandwich beams and plates. The splitting (38) may also be used,

$$\mathcal{S}(u, \lambda) = (K - \lambda M)u, \quad \text{and} \quad \mathcal{T}(u, \lambda) = HG_0\Big(1 + \sum_{j=1}^{3} \frac{\Delta_j \omega}{\omega - i\Omega_j}\Big)\tilde{D}u. \tag{38}$$

The choice between these two splittings may be of particular interest when an electro-rheological law [40] or a magneto-rheological law [32] is used. The difference between these models and viscoelastic ones is in the parameter E_* that represents an applied electrical or magnetic field to be varied. For instance, the shear modulus used in [40] reads as

$$G(E_*) = m_g E_* + b_g, \tag{39}$$

j	Δ_j	$\Omega_j(rad/s)$
1	0.746	468.7
2	3.265	4742.4
3	43.284	71532.5

TABLE 3: Parameters for the 3M ISD112 model at a temperature of 27°C [37].

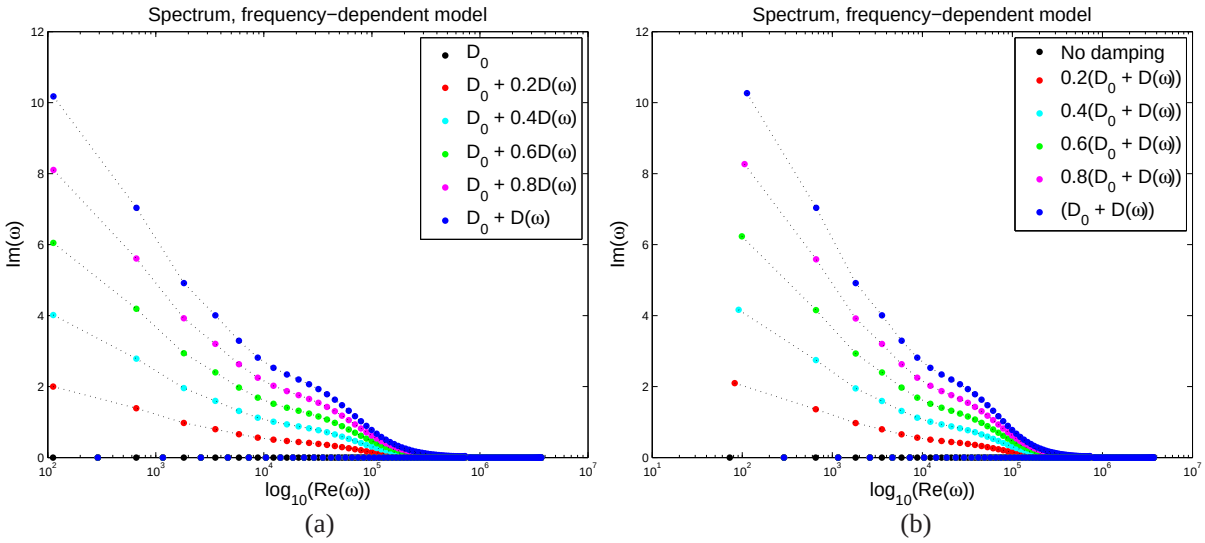

FIGURE 7: Eigensolutions for various scaling of the damping model (36). (a) : Splitting (37). (b) : Splitting (38)

where the parameters m_g and b_g are obtained from experiments. It does not depends on ω. A homotopy based on the splitting (37), with $\mathcal{T} = m_g E_*$, enables for the observation of the evolution of the eigensolutions with respect to the applied electrical field. Work is in progress to consider such a law in an actual mechanical framework. Meanwhile, Fig. 7 plots the eigenvalue evolution for the beam damped with the 3M ISD112 function considering the splittings (37) and (38), respectively. A unique branch computation is necessary per eigenvalue, except for the first eigenvalue computed with the splitting (38).

The half spectrums (positive imaginary parts) corresponding to damping function $(D_0+\mathcal{D}(\omega))$ for various $a \in (0,1)$ are plotted in Figs. 7.a and 7.b for the splitting (37) and (38), respectively. The choice for one or another splitting influences the construction of the real eigenvalue problem and determines the "slope" between the real eigenvalue (starting point of the homotopy) and the related complex eigenvalue (end point of the homotopy). Intermediate results are different along the path since the path parameter a acts as a scaling on $\mathcal{T}$ only. As expected, the first eigenvalue is the more sensitive to the splitting choice and both splittings produce equivalent eigensolutions at the end points. The first five complex eigenvalues computed at the end point of the homotopy are reported in Tab. 5 for the two splittings.

5.2.3. Sensitivities

The few statements of the `damped_beam` code are differentiated (manually) with respect to modeling parameters to get the tangent linear matrices K^d, D^d and M^d. Taylor tests, see Tab. 4, are verified in a systematic manner. Figure 8 presents sensitivity results for a perturbation of the density coefficient ρ and the two splittings. As expected, the sensitivity on ω is negative because a denser beam vibrates slower and has lower frequencies. Computations are performed using $\rho = 0.674 Kg.m^{-3}$ and $\rho^d = 0.1 Kg.m^{-3}$. In Figure 9.a, the eigenvalues (color circles, blue for $a = 1$, magenta for $a = 0.8$, green for $a = 0.6$, cyan for $a = 0.4$, red for $a = 0.2$ and black for $a = 0$) and their related sensitivity are added to propose a first order approximation (black crosses) of the first nine eigenvalues computed with $\rho = 0.774 Kg.m^{-3}$ (with squares and same colors). For the two splittings, the good agreement between squares and crosses is confirmed by relative errors (about 0.8%) presented in Tab. 5 on the first order approximations using the

α	$r_\alpha(\lambda)$ (nlevp, $d_c = 5i$)	$r_\alpha(\lambda)$ (ISD112, $H = 5.10^{-3}m$)
1.E-2	0.9889713342713	0.98895850930239
1.E-3	0.9989467373101	0.99887758359406
1.E-4	1.0011346796444	1.00115275243155
1.E-5	1.0145292086048	1.01473807301412
1.E-6	1.0772568674870	1.07828544963728
1.E-7	2.6064844849179	2.61097302946291

TABLE 4: Taylor test on the sensitivity of the first complex eigenvalue λ (at homotopy end point) with respect to a perturbation of the density $(\rho, \rho^d) = (0.2, 0.1)$.

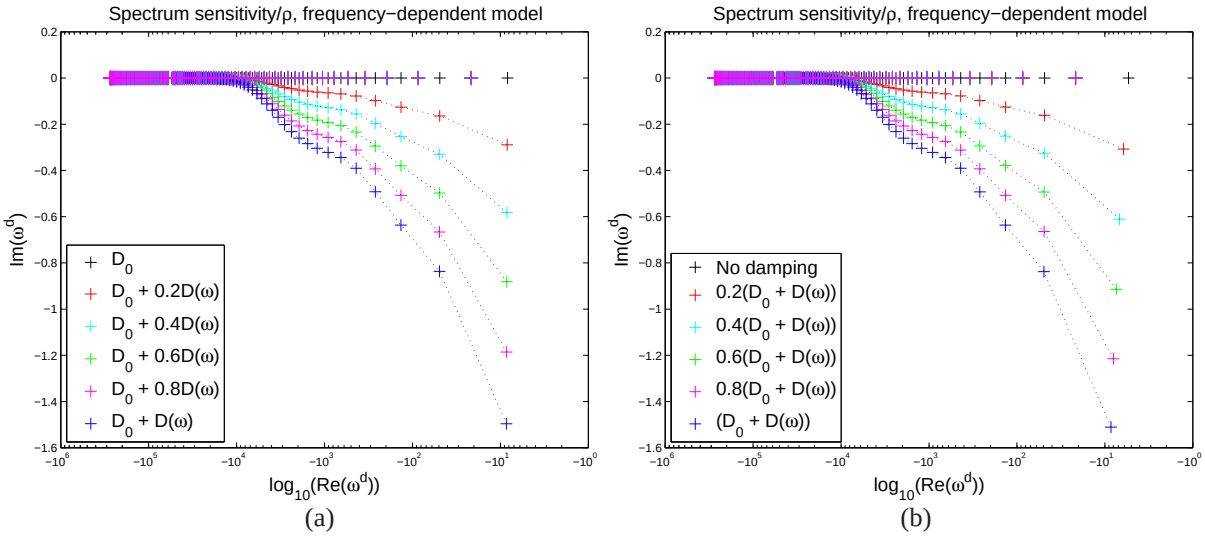

FIGURE 8: Spectrum sensitivity, with respect to ρ, for various values of the damping coefficient. (Color online).

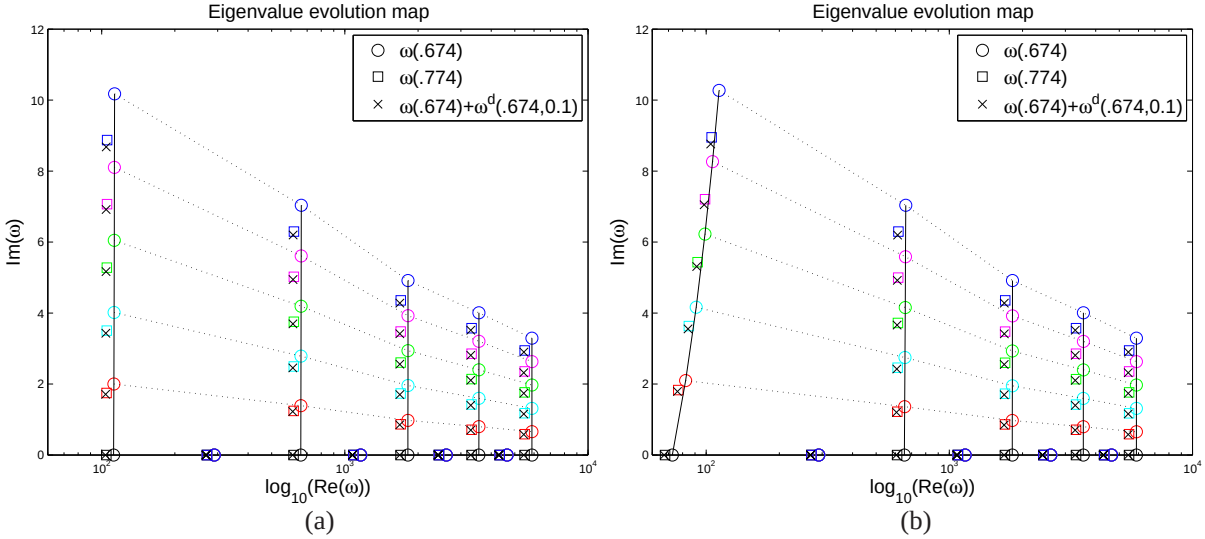

FIGURE 9: Eigenvalue maps and first order approximations for $a = 1$ (blue), $a = 0.8$ (magenta), $a = 0.6$ (green), $a = 0.4$ (cyan), $a = 0.2$ (red), $a = 0$ (black). (a) : Homotopy (37) ; (b) : Homotopy (38)

N^o	Eigenvalue ($\epsilon = 0.674$) $\omega(0.674)$	Sensitivity $\omega^d(0.674, .1)$	Eigenvalue ($\epsilon = 0.774$) $\omega(0.774)$	Path comp. Rel. err.	End-point Rel. err.
Splitting (37)					
1	1.130e+002 + 1.018e+001i	-8.539e+000 + -1.496e+000i	1.053e+002 + 8.872e+000i	8.354e-003	8.241e-003
2	2.904e+002 + 6.424e-018i	-2.154e+001 + -8.358e-019i	2.709e+002 + 4.641e-016i	7.878e-003	7.878e-003
3	6.622e+002 + 7.035e+000i	-4.932e+001 + -8.365e-001i	6.178e+002 + 6.296e+000i	7.923e-003	7.936e-003
4	1.161e+003 + 5.077e-021i	-8.616e+001 + -6.396e-022i	1.084e+003 + 2.142e-019i	7.878e-003	7.878e-003
5	1.819e+003 + 4.916e+000i	-1.351e+002 + -6.365e-001i	1.697e+003 + 4.357e+000i	7.891e-003	7.893e-003
Splitting (38)					
1	1.130e+002 + 1.027e+001i	-8.530e+000 + -1.511e+000i	1.053e+002 + 8.951e+000i	8.341e-003	8.287e-003
2	2.904e+002 + 3.445e-016i	-2.154e+001 + -4.552e-017i	2.709e+002 + 2.218e-016i	7.878e-003	7.878e-003
3	6.622e+002 + 7.035e+000i	-4.931e+001 + -8.379e-001i	6.178e+002 + 6.296e+000i	7.921e-003	7.936e-003
4	1.161e+003 + 1.444e-018i	-8.616e+001 + -1.821e-019i	1.084e+003 + 2.175e-021i	7.878e-003	7.878e-003
5	1.819e+003 + 4.916e+000i	-1.351e+002 + -6.367e-001i	1.697e+003 + 4.357e+000i	7.891e-003	7.893e-003

TABLE 5: Eigenvalues ω and sensitivities ω^d to a perturbation of the density computed at the end point of the homotopy path. Relative errors for the first five eigenvalues and the two splittings.

sensitivity ω^d computed along the path.

Sensitivities computed along the path with respect to both the homotopy parameter and one of the modeling parameter, allow for the construction of approximated eigenvalue maps. Similar maps may be constructed for the other modeling parameters. Consequently, approximated eigenvalues may be evaluated in any non trivial direction of perturbation αp_1 ($\alpha \in \boldsymbol{R}$ small, $|p_1| = 1$) from the homotopy checkpoints.

At the end point of the homotopy, see paragraph 4.2.2, the local sensitivity ω_1^p of a complex eigenvalue ω may be obtained from a differentiation with respect to p of both the mass orthonormalization formula and the generic equation (4). The tangent linear equation (40)

$$Mu_1^p = -0.5 M_1 u, \tag{40}$$

is derived the orthonormalization formula and solved from the known complex eigensolution (u, λ) to compute the sensitivity u_1^p of the eigenvector u. The sensitivity λ_1^p of the eigenvalue verifies

$$\{\mathcal{R}_{1|\lambda_1^p=1,u_1^p=0,p_1=0}\}\lambda_1^p + \{\mathcal{R}_{1|\lambda_1^p=0,u_1^p,p_1}\} = 0. \tag{41}$$

It may be computed as follows

$$\lambda_1^p = -\frac{u^T\{\mathcal{R}_{1|\lambda_1^p=0,u_1^p,p_1}\}}{u^T\{\mathcal{R}_{1|\lambda_1^p=1,u_1^p=0,p_1=0}\}}, \tag{42}$$

to deduce

$$\omega_1^p = \frac{1}{2}\frac{\lambda_1^p}{\sqrt{\lambda}}. \tag{43}$$

In equation (42), the tangent linear term $\{\mathcal{R}_{1|\lambda_1^p=1,u_1^p=0,p_1=0}\}$ is the first order derivative of $\mathcal{R}$ computed with respect to λ only. The complementary term $\{\mathcal{R}_{1|\lambda_1^p=0,u_1^p,p_1}\}$ contains the first order derivatives with respect to p and u evaluated using the direction of perturbation p_1 and the sensitivity u_1^p. Thinking in further higher order approximations, computations are performed using the code $\mathcal{R}$ overloaded for a differentiation with respect to p. The tangent linear code $\mathcal{R}^d$ may also be used. Both methods are implemented and quasi-identical results are obtained. Some relative errors measured between the perturbed eigensolutions $\omega(0.774)$ and their the first order approximation obtained with (41)–(42) are reported in Tab. 5. The different approaches proposed to compute the eigenvalue sensitivities yield very similar results at the end points.

6. Conclusions

On the one hand, continuation and homotopy are classical tools for the solution of nonlinear parametric problems arising in engineering applications. On the other hand, sensitivity analysis are precious tools for model improvement,

conception or uncertainty measurement. Proposing sensitivity drivers for the continuation methods is a natural issue. This paper discusses several options for the sensitivity analysis with respect to the modeling parameters of the solution branches produced by continuation, including homotopy, together with particular applications and usages. Theoretical developments and implementation are realized in the Diamant framework that combines higher order Taylor approximations to AD, inheriting from the generality of Diamant and AD. Numerical results are presented for a thermal ignition problem and the damped beam problem issued from the NLEVP collection. In both cases, accurate sensitivities are computed.

A particular attention is brought to the complex nonlinear eigenvalue problems arising in the free vibration modeling of damped structures. A frequency-dependent rheological model is considered to emphasize the interest of sensitivity computations performed along the homotopy path. More precisely, the sensitivity driver we propose allows for the construction of approximated eigenvalue maps from both a very few Taylor series expanded in the homotopy path parameter and a sensitivity analysis performed concurrently with respect to another modeling parameter. Future work will be concerned with a higher order conception method taking advantage of eigenvalue maps.

Références

[1] N. Bazley, G. Wake, The disappearance of criticality in the theory of thermal ignition, Z. Angew. Math. Phys. 29 (1978) 971–976.

[2] M. Berz, K. Makino, K. Shamseddine, G. Hoffstätter, W. Wan, COSY INFINITY and its applications in nonlinear dynamics, in : M. Berz, C. Bischof, G. Corliss, A. Griewank (eds.), Computational Differentiation : Techniques, Applications, and Tools, SIAM, Philadelphia, PA, 1996, pp. 363–365.

[3] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder, F. Tisseur, NLEVP : A collection of nonlinear eigenvalue problems, ACM T. Math. Software 39 (2013) 7 :1–7 :28.

[4] S. Bhattacharya, V. Kumar, M. Likhachev, Search-based path planning with homotopy class constraints, in : Third Annual Symposium on Combinatorial Search, 2010.

[5] M. Bilasse, I. Charpentier, E. Daya, Y. Koutsawa, A generic approach for the solution of nonlinear residual equations. Part II : Homotopy and complex nonlinear eigenvalue method, Comput. Method. Appl. M. 198 (2009) 3999–4004.

[6] C. H. Bischof, A. Carle, G. F. Corliss, A. Griewank, P. D. Hovland, Adifor - generating derivative codes form fortran programs, Sci. Program. 1 (1992) 11–29.

[7] Y. Chang, G. F. Corliss, Solving ordinary differential equations using taylor series, ACM T. Math. Software 8 (1982) 114–144.

[8] I. Charpentier, Checkpointing schemes or adjoint codes : Application to the meteorological modelMeso-NH, SIAMJ. Sci. Comput. 22 (2001) 2135–2151.

[9] I. Charpentier, Sensitivity of solutions computed through the asymptotic numerical method, C. R. Mecanique 336 (2008) 788–793.

[10] I. Charpentier, On higher-order differentiation in nonlinear mechanics, Optim. Method. Softw. 27 (2012) 221–232.

[11] I. Charpentier, B. Cochelin, K. Lampoh, Diamanlab - An interactive Taylor-based continuation tool in MATLAB, 11 pages (Mar. 2013). URL http ://hal.archives-ouvertes.fr/hal-00853599 (17/03/2014)

[12] I. Charpentier, N. Jakse, Exact numerical derivatives of the pair-correlation function of simple liquids using the tangent linear method, J. Chem. Phys. 114 (5) (2001) 2284–2292.

[13] I. Charpentier, M. Potier-Ferry, Différentiation automatique de la méthode asymptotique numérique typée : l'approche Diamant, C. R. Mecanique 336 (2008) 336–340.

[14] I. Charpentier, J. Utke, Fast higher-order derivative tensors with rapsodia, Optim. Method. Softw. 24 (2009) 1–14.

[15] B. Cochelin, N. Damil, M. Potier-Ferry, Méthode Asymptotique Numérique, Hermes Science Publications, Paris, 2007.

[16] I. Elkhaldi, I. Charpentier, E. Daya, A gradient method for viscoelastic behaviour identification of damped sandwich structures, C. R. Mecanique 34 (2012) 619–623.

[17] H. Frey, S. Patil, Sumeet, Identification and review of sensitivity analysis methods, Risk Anal. 22 (2002) 553–578.

[18] A. Griewank, D. Juedes, J. Utke, ADOL–C, a package for the automatic differentiation of algorithms written in C/C++, ACM T. Math. Software 22 (2) (1996) 131–167.

[19] A. Griewank, J. Utke, A. Walther, Evaluating higher derivative tensors by forward propagation of univariate Taylor series, Math. Comput. 69 (2000) 1117–1130.

[20] A. Griewank, A. Walther, Treeverse : An implementation of checkpointing for the reverse or adjoint mode of computational differentiation, ACM T. Math. Software 26 (1997) 200–0.

[21] A. Griewank, A.Walther, Evaluating Derivatives : Principles and Techniques of Algorithmic Differentiation, no. 105 in Other Titles in Applied Mathematics, 2nd ed., SIAM, Philadelphia, PA, 2008.

[22] P. Guillaume, Nonlinear eigenproblems, SIAM J. Matrix Anal. A 20 (1999) 575–595.

[23] P. Guillaume, M. Masmoudi, Computation of high order derivatives in optimal shape design, Numer. Math. 67 (1994) 231–250.

[24] L. Hascoët, V. Pascual, TAPENADE 2.1 user's guide, Rapport technique 300, INRIA, Sophia Antipolis (2004).

[25] J. Hershberger, J. Snoeyink, Computing minimum length paths of a given homotopy class (extended abstract), in : Workshop on Algorithms and Data Structures, 1991.

[26] N. J. Higham, D. S. Mackey, F. Tisseur, S. D. Garvey, Scaling, sensitivity and stability in the numerical solution of quadratic eigenvalue problems, Int. J. Numer. Meth. Eng. 73 (2008) 344–360.

[27] H. Keller, Global homotopies and newton methods, in : C. de Boor, G. Golub (eds.), Recent advances in Numerical Analysis, Academic Press, New-York, 1978.

[28] H. Keller, Lectures on numerical methods in bifurcation problems, Springer-Verlag, Berlin, 1987.
[29] F. V. Keulen, R. Haftka, N. Kim, Review of options for structural design sensitivity analysis. part 1. linear systems, Comput. Method. Appl. M. 194 (2005) 3213–3243.
[30] Y. Koutsawa, I. Charpentier, E. Daya, M. Cherkaoui, A generic approach for the solution of nonlinear residual equations. Part I : the Diamant toolbox, Comput. Method. Appl. M. 198 (2008) 572–577.
[31] K. Lampoh, I. Charpentier, E. Daya, A generic approach for the solution of nonlinear residual equations. Part III : Sensitivity computations, Comput. Method. Appl. M. 200 (2011) 2983–2990.
[32] B. Nayak, S. Dwivedy, K. Murthy, Dynamic stability of magnetorheological elastomer based adaptive sandwich beam with conductive skins using FEM and the harmonic balance method, Int. J. Mech. Sci. 77 (0) (2013) 205 – 216.
[33] N. Nedialkov, J. Pryce, Solving differential-algebraic equations by taylor series (I) : Computing taylor coefficients, BIT 45 (2005) 561–591.
[34] J. Pryce, J. Reid, AD01, a Fortran 90 code for automatic differentiation, Tech. Rep. RAL-TR-1998-057, Rutherford Appleton Laboratory, Chilton, Oxfordshire, England (1998).
[35] R. Seydel, Practical Bifurcation and Stability Analysis, no. 5 in Interdisciplinary Applied Mathematics, 3rd ed., Springer, 2009.
[36] J. Thompson, A. Walker, The nonlinear perturbation analysis of discrete structural systems, Int. J. Solids Struct. 4 (1968) 757–768.
[37] M. Trindade, A. Benjeddou, R. Ohayon,Modeling of frequeney-dependent viscoelastic materials for active-passive vibration damping, J. Vib. Acoust. 122 (2) (2000) 169–174.
[38] J. Utke, B. T. Rearden, R. A. Lefebvre, Sensitivity analysis for mixed-language numerical models, Procedia Comput. Sci. 18 (0) (2013) 1794–1803.
[39] J. Verschelde, Y.Wang, Numerical homotopy algorithms for satellite trajectory control by pole placement, in : In Proceedings of MTNS 2002, Mathematical Theory of Networks and Systems, 2002. Notre Dame, 2002.
[40] J.-Y. Yeh, L.-W. Chen, Finite element dynamic analysis of orthotropic sandwich plates with an electrorheological fluid core layer, Compos. Struct. 78 (3) (2007) 368 – 376.