Structural and Multidisciplinary Optimization (2018) 58:2387–2410
https://doi.org/10.1007/s00158-018-2096-y

RESEARCH PAPER

Semi-analytical sensitivity analysis for nonlinear transient problems

Felipe Fernandez[1] · Daniel A. Tortorelli[1,2]

Received: 30 March 2018 / Revised: 5 September 2018 / Accepted: 7 September 2018 / Published online: 28 September 2018

Abstract

Efficient analytical sensitivity computations are essential elements of gradient-based optimization schemes; unfortunately, they can be difficult to implement. This implementation issue is often resolved by adopting the semi-analytical method which exhibits the efficiency of the analytical methods and the ease of implementation of the finite difference method. However, care must be taken as semi-analytical sensitivities may exhibit errors due to truncation and round-off. Additional errors are introduced if the convergence tolerance of the primal analysis is not sufficiently small. This paper gives a general overview and some new developments of the analytical and semi-analytical sensitivity analyses for nonlinear steady-state, transient, and dynamic problems. We discuss the restrictive assumptions, accuracy, and consistency of these methods. Both adjoint and direct differentiation methods are studied. Numerical examples are provided.

Keywords Non-linear · Semi-analytical · Sensitivity analysis · Dynamic · Adjoint · Direct

1 Introduction

Both, analyses and design sensitivity analyses, are crucial in gradient-based optimization wherein analyses are performed to predict the performance of proposed designs, while design sensitivity analyses are performed to quantify the performance changes with respect to design changes. Since the optimization is iterative and because it relies on the accurate values of the gradients, efficient and accurate sensitivity analyses are essential. The finite difference sensitivity method requires one re-analysis to compute the sensitivities of the performance functions with respect to each design variable, so this method is extremely inefficient especially when the primal analysis is time-consuming. On the other hand, analytical direct differentiation and adjoint sensitivity analyses are very efficient. Unfortunately, this efficiency requires the analytical evaluation of various derivatives which may be difficult to compute since they require detailed knowledge of the analysis program. Indeed, analytical sensitivities require the differentiation of specific element formulations and material models with respect to a variety of design variables (Cheng and Olhoff 1993; Kiendl et al. 2014). To alleviate these implementation issues, the semi-analytical method approximates these derivatives with finite differences; as such little knowledge of the analysis program is required. However, care must be exercised as the accuracy of the semi-analytical method depends on the finite difference perturbation size. For a thorough review of sensitivity analyses and the semi-analytical method see Haftka and Adelman (1989), Tortorelli and Michaleris (1994), Gunzburger (2003), van Keulen et al. (2005), Haftka and Gürdal (2012).

Much work has been focused on the semi-analytical method for linear static structural problems (Gallagher and Zienkiewicz 1973; Botkin 1982; Camarda and Adelman 1984; Esping 1984; Cheng and Liu 1987; Barthelemy et al. 1988; Pedersen et al. 1989; Barthelemy and Haftka 1990; Haftka and Adelman 1989; Fenyes and Lust 1991; Olhoff and Rasmussen 1991; Bestle and Seybold 1992), especially its application to shape sensitivity analysis.

Our response functions are integrals over the domain. In shape sensitivity analysis, the design variables include

Responsible Editor: Hai Huang

This work was partially performed under the auspices of the US Department of Energy by Lawrence Livermore Laboratory under contract DE-AC52-07NA27344, cf. ref number LLNLCONF-717640.

✉ Felipe Fernandez
lfferna2@illinois.edu

Daniel A. Tortorelli
tortorelli2@llnl.gov

[1] Department of Mechanical Sciences and Engineering, University of Illinois at Urbana-Champaign, Champaign, IL, USA

[2] Center for Design and Optimization, Lawrence Livermore National Laboratory, Livermore, CA, USA

geometric parameters that define this domain. Thus, analytical shape sensitivity analyses require the use of the material derivative from continuum mechanics and such computations can be onerous. For this reason, the semi-analytical method of shape sensitivity analyses may be preferable for its ease of implementation; moreover, it is fully reliable for most problems in which the structural displacement field entails small rigid-body rotations relative to deformations of the finite elements (Olhoff et al. 1993). However, large errors attributed to rigid body rotations of the finite elements have been found in shape sensitivities computed with the semi-analytical method (Barthelemy et al. 1988; Cheng et al. 1989; Pedersen et al. 1989; Fenyes and Lust 1991; Olhoff and Rasmussen 1991; Cheng and Olhoff 1993).

Different approaches have been suggested to improve the accuracy of the semi-analytical method. For example, improved accuracy is obtained by using the second-order central differences scheme, instead of first-order accurate forward differences (Barthelemy et al. 1988; Cheng et al. 1989; Haftka and Adelman 1989; Pedersen et al. 1989; Fenyes and Lust 1991). This method requires an additional computational cost and unfortunately does not completely eliminate the errors caused by large rigid body motions in shape sensitivity analysis. To circumvent this, the *natural approach* retains consistency conditions for rigid body modes and their derivatives (Mlejnek 1992). Alternatively, the analytical derivatives of the element rigid body modes are incorporated in the *refined* semi-analytical design sensitivities approach to alleviate inaccuracies (Van Keulen and De Boer 1998). Utilizing specific characteristics of the element stiffness matrices to compute correction factors, the so-called *exact* semi-analytical eliminates truncation error (Olhoff et al. 1993). A proposed *improved* semi-analytical method obtains better accuracy by using the von Neumann series (Oral 1996).

Kiendl et al. (2014) use the isogeometric finite element in which non-rational uniform B-splines (NURBS) are used to parameterize both the finite element response and the domain geometry. A multilevel approach allows for a more coarse, i.e., smooth, design parameterization versus the finite element response. The semi-analytical method is combined with a sensitivity weighting scheme to compute the design updates for their optimization example problems.

Semi-analytical methods have been applied for nonlinear static structures. Haftka (1993) and Mróz and Haftka (1994) use it to compute sensitivities of limit loads and show that the semi-analytical method is equivalent to the overall finite difference method when a single Newton iteration is used. A more thorough formulation of the refined semi-analytical method was presented for linear, linearized buckling, geometrically nonlinear and limit point analyses in de Boer and van Keulen (2000). The exact semi-analytical method has also been extended to geometric nonlinearities in Wang et al. (2015). Curiously, this formulation uses the secant stiffness matrix and incorporates correction terms to eliminate truncation errors.

The refined semi-analytical approach was also extended to obtain second-order derivatives (de Boer et al. 2002). The higher-order semi-analytical derivatives studied by Bernard et al. (1993) use cubic polynomials to develop surrogate models of the mass and stiffness matrices so that higher-order derivatives can be easily computed.

Sensitivity analysis for transient problems have been extensively studied (Adelman and Haftka 1986; Haug 1987; Haftka and Gürdal 2012). These studies included nonlinearities (Ray et al. 1978; Michaleris et al. 1994; Kreissl et al. 2011; Deng et al. 2011), and shape sensitivities (Meric 1988; Tortorelli et al. 1991). The semi-analytical method has been applied for linear transient structural problems using a reduced order modal model (Camarda and Adelman 1984; Greene and Haftka 1991; Hooijkamp and van Keulen 2018). As such, these methods are restricted to linear systems.

The semi-analytical method has been applied to transient heat conduction problems (Gu and Grandhi 1998), including nonlinear behaviour (Gu et al. 2002), and nonlinear coupled with structural dynamics (Chen et al. 2003). It is unclear how their use of the Precise Time Integration scheme (Zhong and Williams 1994) which is limited to time varying linear systems affects the accuracy of their nonlinear analyses and subsequent sensitivity analysis.

Semi-analytical sensitivity analysis via direct differentiation has been applied to dynamic systems with large rotations (Brüls and Eberhard 2008), and to flexible multi-body systems (Tromme et al. 2015). In the latter, to ease the computation, the pseudo load is approximated using the perturbation of the residual. This approximation is easy to implement, since simulation codes usually have a function to compute the residual (Tromme et al. 2015). The goal of this paper is to study this formulation and extend it to the adjoint method.

In the following, we study the semi-analytical method to facilitate the sensitivity analyses for transient nonlinear systems. The transient problems are treated as general as possible. To do this, we use both an implicit-explicit time integration algorithm and the popular Newmark time stepping method. Additionally, we use a general formulation, so the methods can be applied to any type of transient problems (e.g., thermal, structural, multi-body, etc.). We systematically develop direct and adjoint sensitivity analysis approaches. Furthermore, for transient and dynamic problems, we study the adjoint differentiate-then-discretize and the adjoint discretize-then-differentiate approaches. The adjoint semi-analytical sensitivity analysis approaches require restrictive assumptions. In particular, we show that the adjoint differentiate-then-discretize method exhibits consistency error and requires some terms to be constant in order to reuse the tangent stiffness matrix from the primal analysis. We also show that the semi-analytical adjoint differentiate-then-discretize method, for nonlinear transient and nonlinear dynamic systems is limited to systems with symmetric stiffness and damping matrices. Fortunately, we show that by using an implicit time integration, the discretize-then-differentiate adjoint method can accommodate asymmetric stiffness matrices.

The major contributions of this paper are (1) an overview of analytical sensitivity analysis for nonlinear transient problems, (2) the development of novel efficient semi-analytical formulations, (3) the identification of restrictions for semi-analytical adjoint methods, and (4) a discussion of the consistency and accuracy of the methods.

This paper gives a general overview of the finite difference method (Section 2.2) and the analytical and semi-analytical sensitivity analyses for nonlinear steady state (Section 2), transient (Section 3) and dynamic (Section 4) systems. Numerical examples are provided in Sections 3.6, and 4.6 wherein the accuracy of the methods are discussed. To quantify the accuracy, we introduce the relative percentage error between the sensitivities obtained by finite differences δF_f and the analytical sensitivities δF as

$$e_f = \left| \frac{\delta F_f - \delta F}{\delta F} \right| 100\% \,. \tag{1}$$

Similarly, we compute the relative error of the semi-analytical sensitivities δF_s with respect to the analytical sensitivities δF as

$$e_s = \left| \frac{\delta F_s - \delta F}{\delta F} \right| 100\% \,. \tag{2}$$

2 Steady-state nonlinear problems

After finite element discretization, the steady-state nonlinear problem is expressed in terms of the residual function $\mathbf{R}$ via the equation

$$\mathbf{R}(\mathbf{U}) = \mathbf{0} \,, \tag{3}$$

where $\mathbf{U}$ is the response vector, e.g., displacement. This nonlinear problem is solved using the iterative Newton-Raphson method. If the residual of the current iterate $\mathbf{U}_j$ is not a solution, $\mathbf{R}(\mathbf{U}_j) \neq \mathbf{0}$, then the next iterate $\mathbf{U}_{j+1} = \mathbf{U}_j + \Delta\mathbf{U}_j$ is computed by equating the first order Taylor series expansion of $\mathbf{R}$ about $\mathbf{U}_{j+1}$ to zero, i.e.,

$$\mathbf{R}(\mathbf{U}_{j+1}) = \mathbf{R}(\mathbf{U}_j + \Delta\mathbf{U}_j) \approx \mathbf{R}(\mathbf{U}_j) + \mathbf{K}\,\Delta\mathbf{U}_j = \mathbf{0} \,, \tag{4}$$

where $\mathbf{K} = \partial\mathbf{R}/\partial\mathbf{U}$ is the tangent matrix. The incremental response update $\Delta\mathbf{U}_j$ is obtained by solving the linear equation

$$\mathbf{K}(\mathbf{U}_j)\Delta\mathbf{U}_j = -\mathbf{R}_j(\mathbf{U}_j) \,, \tag{5}$$

whereafter the next iterate

$$\mathbf{U}_{j+1} = \mathbf{U}_j + \Delta\mathbf{U}_j \,, \tag{6}$$

is computed. The steps of evaluating the residual $\mathbf{R}$ and updating the response $\mathbf{U}$ are repeated until the solution converges to a within user specified tolerance, i.e., until $|\mathbf{R}(\mathbf{U})| \leq \epsilon_R$.

2.1 Sensitivity analysis of steady-state nonlinear systems

For the sensitivity analysis we treat the residual $\mathbf{R}$ and the response $\mathbf{U}$ as functions of the n_d vector of design variables $\mathbf{d} = [d_1, d_2, .., d_{n_d}]^\top$, i.e., we now have express (3) as

$$\mathbf{R}(\mathbf{U}(\mathbf{d}), \mathbf{d}) = \mathbf{0} \,. \tag{7}$$

After completing the primal analysis of (3), we can evaluate any number of response functions F. For our purposes, the response function depends on the response $\mathbf{U}(\mathbf{d})$ to the problem in (7) whereby we express

$$F(\mathbf{d}) = G(\mathbf{U}(\mathbf{d}), \mathbf{d}) \,. \tag{8}$$

Using the chain rule, the derivative of the response functional of (8) with respect to each d_i is

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}G}{\mathrm{D}d_i} = \frac{\partial G}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} + \frac{\partial G}{\partial d_i} \,, \tag{9}$$

where $\partial\mathbf{U}/\partial d_i$ is implicitly defined through (7).

2.2 Finite difference method

The forward finite difference method approximates the derivatives of a response function F using a truncated Taylor series expansion

$$\frac{\mathrm{D}F(\mathbf{d})}{\mathrm{D}d_i} \approx \frac{F(\mathbf{d} + \epsilon\,\mathbf{e}_i) - F(\mathbf{d})}{\epsilon} \,, \tag{10}$$

where $\mathbf{e}_i = [0, 0, .., 1, ..., 0, 0]^\top$ is the unit vector of component i, and ϵ the perturbation. The approximation $\mathrm{D}F(\mathbf{d})/\mathrm{D}d_i\,\epsilon \approx F(\mathbf{d} + \epsilon\,\mathbf{e}_i) - F(\mathbf{d})$ exhibits truncation error $o(\epsilon)$, where o is a function defined such that $o(\epsilon)$ tends to zero faster than ϵ, i.e., $\lim_{\epsilon\to 0} o(\epsilon)/\epsilon = 0$. To reduce the truncation error $o(\epsilon)$ it is desirable to choose a small ϵ, however, numerical round-off error will erode the accuracy of the approximation if ϵ is too small.

Since the response function depends on the response $\mathbf{U}(\mathbf{d})$, the approximation of (10) is expressed by

$$\frac{\mathrm{D}F(\mathbf{d})}{\mathrm{D}d_i} \approx \frac{G(\mathbf{U}(\mathbf{d} + \epsilon\,\mathbf{e}_i), \mathbf{d} + \epsilon\,\mathbf{e}_i) - G(\mathbf{U}(\mathbf{d}), \mathbf{d})}{\epsilon} \,. \tag{11}$$

As seen above, the response $\mathbf{U}(\mathbf{d} + \epsilon\, \mathbf{e}_i)$ must be calculated for each design variable d_i; this is easily obtained by modifying the finite element model, but computationally inefficient because it requires n_d additional simulations to compute the $\mathbf{U}(\mathbf{d} + \epsilon\, \mathbf{e}_i)$. Note that second-order accurate approximations which are accurate to $o(\epsilon^2)$ can be obtained by central differences, but this requires two re-analyses for $\mathbf{U}(\mathbf{d} \pm \epsilon\, \mathbf{e}_i)$ which is even more costly. As seen here, the finite difference method is easy to implement, computationally inefficient, and subjected to truncation and round-off errors.

2.3 Direct differentiation for steady-state nonlinear systems

In the direct differentiation approach, the implicit derivative $\partial\mathbf{U}/\partial d_i$, i.e., pseudo response, is obtained by differentiating (7) respect to d_i, which after some rearranging defines the so-called pseudo problem

$$\mathbf{K}\,\frac{\partial\mathbf{U}}{\partial d_i} = -\frac{\partial\mathbf{R}}{\partial d_i}\,, \tag{12}$$

where $-\partial\mathbf{R}/\partial d_i$ is the pseudo load. Notice that the tangent operator $\mathbf{K}$ from the primal analysis appears in the pseudo problem; moreover, it is already factored, assuming the use of direct solvers in the primal analysis. Thus, the evaluation of the implicit derivative $\partial\mathbf{U}/\partial d_i$ only requires the formation of the pseudo load vector $-\partial\mathbf{R}/\partial d_i$ and a back substitution. Once the implicit derivative $\partial\mathbf{U}/\partial d_i$ is obtained, (9) is evaluated to obtain the sensitivities for any number of functions F. As seen here, the direct method is computationally efficient because it solves one pseudo problem using the previously factored tangent matrix for each design variable regardless of the number of response functions. In addition, the computed sensitivities are numerically exact.

In the semi-analytical formulation, the derivatives $\partial\mathbf{R}/\partial d_i$ and $\mathrm{D}G/\mathrm{D}d_i$ of (12) and (9) are approximated to within $o(\epsilon)$ via finite differences

$$\begin{aligned}\frac{\partial\mathbf{R}(\mathbf{U}(\mathbf{d}),\mathbf{d})}{\partial d_i} &\approx \frac{1}{\epsilon}\Big(\mathbf{R}(\mathbf{U}(\mathbf{d}),\mathbf{d}+\epsilon\,\mathbf{e}_i) - \cancelto{\mathbf{0}}{\mathbf{R}(\mathbf{U}(\mathbf{d}),\mathbf{d})}\Big)\\ &\approx \frac{1}{\epsilon}\mathbf{R}(\mathbf{U}(\mathbf{d}),\mathbf{d}+\epsilon\,\mathbf{e}_i)\,,\end{aligned} \tag{13}$$

$$\begin{aligned}\frac{\mathrm{D}G(\mathbf{U}(\mathbf{d}),\mathbf{d})}{\mathrm{D}d_i} \approx \frac{1}{\epsilon}\Bigg(G\left(\mathbf{U}(\mathbf{d}) + \epsilon\,\frac{\partial\mathbf{U}(\mathbf{d})}{\partial d_i}, \mathbf{d}+\epsilon\,\mathbf{e}_i\right)\\ - G(\mathbf{U}(\mathbf{d}),\mathbf{d})\Bigg)\,.\end{aligned} \tag{14}$$

In (13), we assume the residual $\mathbf{R}(\mathbf{U}(\mathbf{d}),\mathbf{d}) = \mathbf{0}$; however, we solve the primal analysis until the solution converges to a user defined tolerance, i.e., $|\mathbf{R}(\mathbf{U}(\mathbf{d}),\mathbf{d})| \le \epsilon_R$. This tolerance imposes a new source of error in addition to the truncation and round-off errors.

Since the function G is known, the derivatives $\partial G/\partial d_i$ and $\partial G/\partial\mathbf{U}$ in the sensitivity $\mathrm{D}F/\mathrm{D}d_i$ can be computed exactly as in (9) or approximated as in (14). We assume the former.

The approximations in (13) and (14) are easy to implement because they only require the generation of the $\mathbf{d} + \epsilon\,\mathbf{e}_i$ followed by the evaluations of the perturbed residual $\mathbf{R}(\mathbf{U}(\mathbf{d}), \mathbf{d} + \epsilon\,\mathbf{e}_i)$ and response function $G\,(\mathbf{U}(\mathbf{d}) + \epsilon\,\partial\mathbf{U}/\partial d_i, \mathbf{d} + \epsilon\,\mathbf{e}_i)$ which are readily computed by the subroutines that are used to compute $\mathbf{R}(\mathbf{U}(\mathbf{d}), \mathbf{d})$ and $G(\mathbf{U}(\mathbf{d}), \mathbf{d})$. Thusly, the semi-analytical method shares the simplicity of the finite difference method and the efficiency of the analytical methods. It is noted, however, that tolerance ϵ_R, truncation and round-off errors may pollute the results. In most cases, a design perturbation will not affect all of the element internal force vectors. As such, we only need to evaluate the elemental residual $\mathbf{R}(\mathbf{U}(\mathbf{d}), \mathbf{d} + \epsilon\,\mathbf{e}_i)$ of the affected elements. An extreme case of this occurs in topology optimization where each volume fraction design variable only affects a single element. Less extreme cases occur in shape optimization where each dimensional change may only affect a subset of the element boundary elements.

2.4 Adjoint method for steady-state nonlinear systems

In the adjoint method, the derivative $\partial\mathbf{U}/\partial d_i$ is annihilated. This formulation uses the identity

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\partial G}{\partial\mathbf{U}}\frac{\partial\mathbf{U}}{\partial d_i} + \frac{\partial G}{\partial d_i} + \mathbf{\Lambda}^\top\left(\mathbf{K}\frac{\partial\mathbf{U}}{\partial d_i} + \frac{\partial\mathbf{R}}{\partial d_i}\right)\,, \tag{15}$$

which follows from (9) and (12). In the above, $\mathbf{\Lambda}$ is the arbitrary adjoint vector. Rearranging (15) yields

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \left(\frac{\partial G}{\partial\mathbf{U}} + \mathbf{\Lambda}^\top\mathbf{K}\right)\frac{\partial\mathbf{U}}{\partial d_i} + \frac{\partial G}{\partial d_i} + \mathbf{\Lambda}^\top\frac{\partial\mathbf{R}}{\partial d_i}\,, \tag{16}$$

from which we identify the adjoint problem that we solve for the heretofore arbitrary $\mathbf{\Lambda}$, i.e.,

$$\mathbf{K}^\top\mathbf{\Lambda} = -\frac{\partial G}{\partial\mathbf{U}}^\top\,. \tag{17}$$

In this way, the term containing $\partial\mathbf{U}/\partial d_i$ is annihilated from (16) reducing the sensitivity to

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\partial G}{\partial d_i} + \mathbf{\Lambda}^\top\frac{\partial\mathbf{R}}{\partial d_i}\,. \tag{18}$$

The adjoint method requires the solution of one adjoint problem (cf. (17)) for each response function F regardless of the number of design variables. And like the direct method, the adjoint problem utilizes the tangent matrix from the primal analysis, so it is also computationally efficient and numerically exact. Furthermore, the tangent stiffness matrix may be already factored, if a direct solver is used in the primal analysis.

In the semi-analytical formulation, the derivative $\partial \mathbf{R}/\partial d_i$ is approximated via finite differences (cf. (13)) and use (18) to obtain the sensitivities. As previously mentioned, the derivative $\partial G/\partial \mathbf{U}$ is obtained analytically using our knowledge of the function G.

3 Transient nonlinear problems

A first-order transient problem is expressed in residual form as

$$\mathbf{R}(\mathbf{U}(t,\mathbf{d}),\dot{\mathbf{U}}(t,\mathbf{d}),\mathbf{d}) = \mathbf{0}\,, \tag{19a}$$

$$\mathbf{U}(0) = \mathbf{U}^0\,, \tag{19b}$$

where we note the design dependencies as in (7), $t \in [0, t_f]$ denotes time and t_f the terminal analysis time. The response function for this system is expressed as

$$F(\mathbf{d}) = \int_0^{t_f} G(\mathbf{U}(t,\mathbf{d}),\dot{\mathbf{U}}(t,\mathbf{d}),\mathbf{d})\,\mathrm{d}t\,. \tag{20}$$

Our goal is to compute the sensitivity in an efficient, accurate and easy manner, i.e., we want to compute

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \int_0^{t_f} \left(\frac{\partial G}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} + \frac{\partial G}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}}{\partial d_i} + \frac{\partial G}{\partial d_i} \right) \mathrm{d}t\,. \tag{21}$$

For the sensitivity analysis, we can implement the direct method whereby we differentiate (19a) and (19b) to define the pseudo problem

$$\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} = -\frac{\partial \mathbf{R}}{\partial d_i}\,, \tag{22a}$$

$$\frac{\partial \mathbf{U}(0)}{\partial d_i} = \frac{\partial \mathbf{U}^0}{\partial d_i}\,, \tag{22b}$$

which we solve for $\partial \mathbf{U}/\partial d_i$ and $\partial \dot{\mathbf{U}}/\partial d_i$ and then we evaluate (21). Alternatively, we can implement the adjoint approach, whereby we utilize (22a) to write (21) as

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i} = &\int_0^{t_f} \left(\frac{\partial G}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} + \frac{\partial G}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}}{\partial d_i} + \frac{\partial G}{\partial d_i} \right) \mathrm{d}t \\ &+ \int_0^{t_f} \boldsymbol{\lambda}^{\top} \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial d_i} \right) \mathrm{d}t\,,\end{aligned} \tag{23}$$

where again $\boldsymbol{\lambda}$ is the arbitrary adjoint vector. Integrating by parts and rearranging (23) yields

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i} = &\int_0^{t_f} \left(\frac{\partial G}{\partial d_i} + \boldsymbol{\lambda}^{\top}\frac{\partial \mathbf{R}}{\partial d_i} \right) \mathrm{d}t - \left(\frac{\partial G}{\partial \dot{\mathbf{U}}} + \boldsymbol{\lambda}^{\top}\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}} \right) \frac{\partial \mathbf{U}}{\partial d_i}\bigg|_{t=0} \\ &+ \int_0^{t_f} \frac{\partial \mathbf{U}}{\partial d_i}^{\top} \left(\frac{\partial G}{\partial \mathbf{U}}^{\top} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial G^{\top}}{\partial \dot{\mathbf{U}}} \right) + \frac{\partial \mathbf{R}^{\top}}{\partial \mathbf{U}}\boldsymbol{\lambda} \right. \\ &\left. - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}^{\top}}{\partial \dot{\mathbf{U}}}\boldsymbol{\lambda} \right) \right) \mathrm{d}t + \frac{\partial \mathbf{U}}{\partial d_i}^{\top} \left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^{\top} + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{\top}\boldsymbol{\lambda} \right)\bigg|_{t=t_f}.\end{aligned} \tag{24}$$

Next, a time mapping is introduced, i.e., we define $\boldsymbol{\Lambda}$ such that

$$\boldsymbol{\Lambda}(t_f - t) = \boldsymbol{\lambda}(t)\,, \tag{25}$$

and hence

$$-\dot{\boldsymbol{\Lambda}}(t_f - t) = \dot{\boldsymbol{\lambda}}(t)\,, \tag{26}$$

substituting the above into (24) renders

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i} = &\int_0^{t_f} \left(\frac{\partial G}{\partial d_i} + \boldsymbol{\Lambda}^{\top}\frac{\partial \mathbf{R}}{\partial d_i} \right) \mathrm{d}t - \left(\frac{\partial G}{\partial \dot{\mathbf{U}}} + \boldsymbol{\Lambda}^{\top}\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}} \right) \frac{\partial \mathbf{U}}{\partial d_i}\bigg|_{t=0} \\ &+ \int_0^{t_f} \frac{\partial \mathbf{U}}{\partial d_i}^{\top} \left(\frac{\partial G}{\partial \mathbf{U}}^{\top} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial G^{\top}}{\partial \dot{\mathbf{U}}} \right) + \frac{\partial \mathbf{R}^{\top}}{\partial \mathbf{U}}\boldsymbol{\Lambda} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}^{\top}}{\partial \dot{\mathbf{U}}} \right)\boldsymbol{\Lambda} \right. \\ &\left. + \frac{\partial \mathbf{R}^{\top}}{\partial \dot{\mathbf{U}}}\dot{\boldsymbol{\Lambda}} \right) \mathrm{d}t + \frac{\partial \mathbf{U}}{\partial d_i}^{\top} \left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^{\top} + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{\top}\boldsymbol{\Lambda} \right)\bigg|_{t=t_f}.\end{aligned} \tag{27}$$

where all quantities are evaluated at time t except $\boldsymbol{\Lambda}$ which is evaluated at $t_f - t$. We can annihilate the terms containing the implicitly defined derivative $\partial \mathbf{U}/\partial d_i$ by requiring $\boldsymbol{\Lambda}$ to solve

$$\begin{aligned}&\frac{\partial \mathbf{R}^{\top}}{\partial \dot{\mathbf{U}}}\dot{\boldsymbol{\Lambda}} + \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{\top} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}^{\top}}{\partial \dot{\mathbf{U}}} \right) \right) \boldsymbol{\Lambda} \\ &= -\frac{\partial G}{\partial \mathbf{U}}^{\top} + \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial G^{\top}}{\partial \dot{\mathbf{U}}} \right)\,,\end{aligned} \tag{28a}$$

$$\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{\top}\bigg|_{t=t_f} \boldsymbol{\Lambda}(0) = -\frac{\partial G}{\partial \dot{\mathbf{U}}}^{\top}\bigg|_{t=t_f}\,. \tag{28b}$$

Using this $\boldsymbol{\Lambda}$, DF/Dd_i reduces to the known quantity

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i} = &\int_0^{t_f} \left(\frac{\partial G}{\partial d_i} + \boldsymbol{\Lambda}^{\top}\frac{\partial \mathbf{R}}{\partial d_i} \right) \mathrm{d}t \\ &- \left(\frac{\partial G}{\partial \dot{\mathbf{U}}} + \boldsymbol{\Lambda}^{\top}\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}} \right) \frac{\partial \mathbf{U}}{\partial d_i}\bigg|_{t=0}\,.\end{aligned} \tag{29}$$

where again all quantities are evaluated at time t except $\boldsymbol{\Lambda}$ which is evaluated at $t_f - t$.

3.1 Discretization

To solve the above, we discretize in time using an explicit/implicit parameter $0 \leq \alpha \leq 1$ so that

$$\mathbf{U}^n = \mathbf{U}^{n-1} + \left(\alpha \dot{\mathbf{U}}^n + (1-\alpha)\dot{\mathbf{U}}^{n-1}\right) \Delta t \,, \tag{30}$$

where $\mathbf{U}^n = \mathbf{U}(t_n)$ and $\dot{\mathbf{U}}^n = \dot{\mathbf{U}}(t_n)$.[1] We then solve (19a) at the discrete times t_n. Finally, the integrals in (20) and (21) are evaluated as

$$F = \sum_{n=0}^{N} \mu_n \, G^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d}) \,, \tag{31}$$

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \sum_{n=0}^{N} \mu_n \left(\frac{\partial G^n}{\partial \mathbf{U}} \frac{\partial \mathbf{U}^n}{\partial d_i} + \frac{\partial G^n}{\partial \mathbf{U}} \frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + \frac{\partial G^n}{\partial d_i} \right) \,, \tag{32}$$

where, e.g., $G^n = G(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d})$ and the coefficient μ_n depends on the summation scheme, e.g., for trapezoidal $2\mu_0 = \mu_1 = \mu_2 = ... = \mu_{N-1} = 2\mu_N = \Delta t$.

3.2 Primal analysis

The initial condition $\mathbf{U}^0$ is given, but $\dot{\mathbf{U}}^0$ is needed in (30) to obtain $\mathbf{U}^1$. To these ends, we use (19a), i.e., we use the Newton-Raphson method to solve

$$\mathbf{R}^0(\mathbf{U}^0, \dot{\mathbf{U}}^0, \mathbf{d}) = \mathbf{0} \,, \tag{33}$$

for $\dot{\mathbf{U}}^0$. The procedure is akin to that which we use to evaluate $\mathbf{U}$ in Section 2. Here $\mathbf{K}^0 = \partial \mathbf{R}^0 / \partial \dot{\mathbf{U}}$ is the tangent matrix. Having $\mathbf{U}^0$ and $\dot{\mathbf{U}}^0$, we compute the first term in (31), i.e., $F = \mu_0 \, G^0(\mathbf{U}^0, \dot{\mathbf{U}}^0, \mathbf{d})$.

Now we commence our analysis. At each time step t_n, we insert $\mathbf{U}^n$ of (30), in (19a) and solve the resulting equation for $\dot{\mathbf{U}}^n$. Again, we use Newton's method for this solution, cf. Section 2, where we introduce the tangent stiffness matrix $\mathbf{K}^n = \partial \mathbf{R}^n / \partial \dot{\mathbf{U}} + \alpha \Delta t \, \partial \mathbf{R}^n / \partial \mathbf{U}$. After convergence, $\mathbf{U}^n$ is updated as per (30) and F is updated as per (31), i.e.,

$$F \leftarrow F + \mu_n \, G^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d}) \,, \tag{34}$$

where the symbol ← represents the update assignment.

The time is then incremental and the process repeats itself until the terminal time t_f. A flow chart describing these computations is provided in Fig. 1 wherein multiple functions F are evaluated for $n = 1, 2, ..., N$.

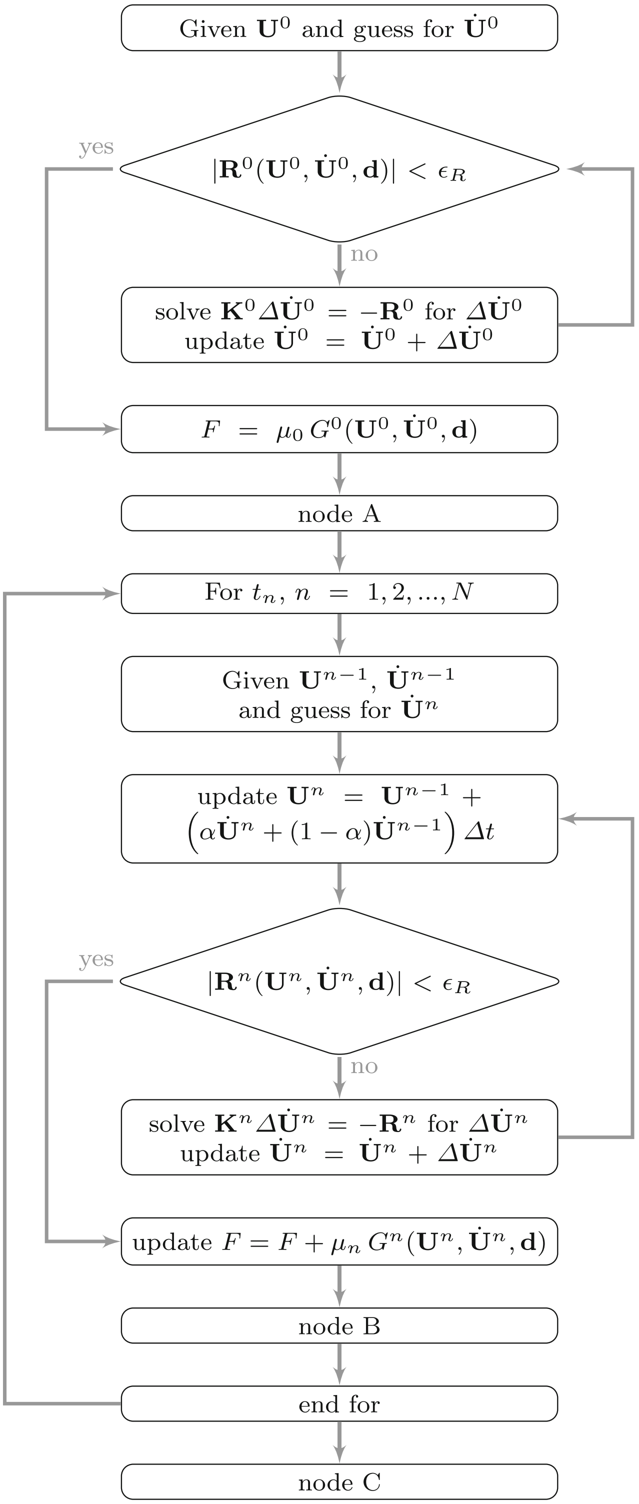
Given $\mathbf{U}^0$ and guess for $\dot{\mathbf{U}}^0$
yes
$|\mathbf{R}^0(\mathbf{U}^0, \dot{\mathbf{U}}^0, \mathbf{d})| < \epsilon_R$
no
solve $\mathbf{K}^0 \Delta\dot{\mathbf{U}}^0 = -\mathbf{R}^0$ for $\Delta\dot{\mathbf{U}}^0$
update $\dot{\mathbf{U}}^0 = \dot{\mathbf{U}}^0 + \Delta\dot{\mathbf{U}}^0$
$F = \mu_0 \, G^0(\mathbf{U}^0, \dot{\mathbf{U}}^0, \mathbf{d})$
node A
For t_n, $n = 1, 2, ..., N$
Given $\mathbf{U}^{n-1}$, $\dot{\mathbf{U}}^{n-1}$
and guess for $\dot{\mathbf{U}}^n$
update $\mathbf{U}^n = \mathbf{U}^{n-1} +$
$\left(\alpha\dot{\mathbf{U}}^n + (1-\alpha)\dot{\mathbf{U}}^{n-1}\right) \Delta t$
yes
$|\mathbf{R}^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d})| < \epsilon_R$
no
solve $\mathbf{K}^n \Delta\dot{\mathbf{U}}^n = -\mathbf{R}^n$ for $\Delta\dot{\mathbf{U}}^n$
update $\dot{\mathbf{U}}^n = \dot{\mathbf{U}}^n + \Delta\dot{\mathbf{U}}^n$
update $F = F + \mu_n \, G^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d})$
node B
end for
node C

Fig. 1 Primal analysis flowchart

[1] For $\alpha = 0$, $1/2$, or 1, we recover the forward Euler, Crank-Nicolson, and backward Euler strategies respectively.

3.3 Direct differentiation

For the direct differentiation, we discretize $\partial \mathbf{U}/\partial d_i$ like $\mathbf{U}$, i.e.,

$$\frac{\partial \mathbf{U}^n}{\partial d_i} = \frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + \left(\alpha \frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + (1-\alpha)\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i}\right)\Delta t\,. \tag{35}$$

Note that the initial condition $\partial \mathbf{U}^0/\partial d_i$ is known, but $\partial \dot{\mathbf{U}}^0/\partial d_i$ is not. So before commencing, we must obtain $\partial \dot{\mathbf{U}}^0/\partial d_i$ like we did $\dot{\mathbf{U}}^0$. To these ends, we differentiate (33) to obtain the linear equation

$$\mathbf{K}^0 \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} = -\left(\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial d_i}\right), \tag{36}$$

which we solve for $\partial \dot{\mathbf{U}}^0/\partial d_i$. Having $\partial \mathbf{U}^0/\partial d_i$ and $\partial \dot{\mathbf{U}}^0/\partial d_i$, we update DF/Dd_i as per (32), i.e.,

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_0\left(\frac{\partial G^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial G^0}{\partial \mathbf{U}}\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} + \frac{\partial G^0}{\partial d_i}\right). \tag{37}$$

Now we march in time evaluating $\partial \mathbf{U}^n/\partial d_i$ and $\partial \dot{\mathbf{U}}^n/\partial d_i$ as we did to compute $\mathbf{U}^n$ and $\dot{\mathbf{U}}^n$. Equation (22a) and (35) render the linear equation

$$\begin{aligned}\mathbf{K}^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d})\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} &= -\left(\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}}\left(\frac{\partial \mathbf{U}^{n-1}}{\partial d_i}\right.\right.\\ &\quad \left.\left. + (1-\alpha)\Delta t\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i}\right) + \frac{\partial \mathbf{R}^n}{\partial d_i}\right),\end{aligned} \tag{38}$$

which we solve for $\partial \dot{\mathbf{U}}^n/\partial d_i$. Next, we update $\partial \mathbf{U}^n/\partial d_i$ as per (35) and DF/Dd_i as per (32)

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} \leftarrow \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_n\left(\frac{\partial G^n}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^n}{\partial d_i} + \frac{\partial G^n}{\partial \mathbf{U}}\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + \frac{\partial G^n}{\partial d_i}\right). \tag{39}$$

We continue marching in this manner for all t_n. In so far as our sensitivity analysis algorithm is concerned, we insert nodes A and B from Fig. 2 into the primal analysis flowchart of Fig. 1.

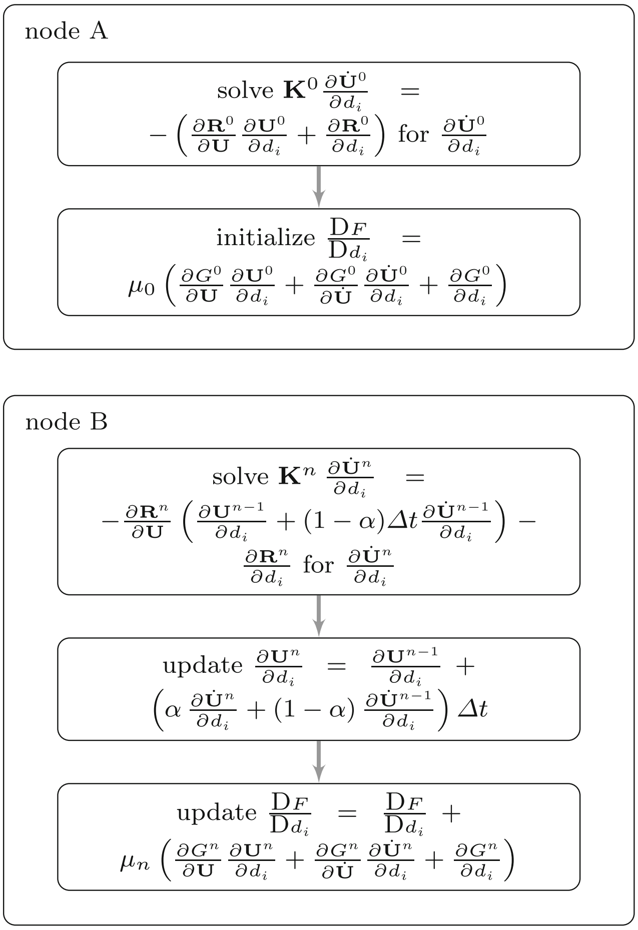
node A
solve $\mathbf{K}^0 \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} = -\left(\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial d_i}\right)$ for $\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}$
initialize $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_0\left(\frac{\partial G^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial G^0}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} + \frac{\partial G^0}{\partial d_i}\right)$
node B
solve $\mathbf{K}^n \frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} = -\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}}\left(\frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + (1-\alpha)\Delta t\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i}\right) - \frac{\partial \mathbf{R}^n}{\partial d_i}$ for $\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i}$
update $\frac{\partial \mathbf{U}^n}{\partial d_i} = \frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + \left(\alpha\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + (1-\alpha)\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i}\right)\Delta t$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_n\left(\frac{\partial G^n}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^n}{\partial d_i} + \frac{\partial G^n}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + \frac{\partial G^n}{\partial d_i}\right)$

Fig. 2 Direct differentiation nodes

For the semi-analytical method we use the approximations

$$\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial d_i} \approx \frac{1}{\epsilon}\mathbf{R}^0\left(\mathbf{U}^0 + \epsilon\frac{\partial \mathbf{U}^0}{\partial d_i}, \dot{\mathbf{U}}^0, \mathbf{d} + \epsilon\,\mathbf{e}_i\right), \tag{40}$$

$$\begin{aligned}&\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}}\left(\frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + (1-\alpha)\Delta t\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i}\right) + \frac{\partial \mathbf{R}^n}{\partial d_i} \approx\\ &\frac{1}{\epsilon}\mathbf{R}^n\left(\mathbf{U}^n + \epsilon\left(\frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + (1-\alpha)\Delta t\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i}\right),\right.\\ &\qquad\qquad\left.\dot{\mathbf{U}}^n, \mathbf{d} + \epsilon\,\mathbf{e}_i\right),\end{aligned} \tag{41}$$

$$\begin{aligned}&\frac{\partial G^n}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^n}{\partial d_i} + \frac{\partial G^n}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + \frac{\partial G^n}{\partial d_i}\\ &\approx \frac{1}{\epsilon}\left(G^n\left(\mathbf{U}^n + \epsilon\frac{\partial \mathbf{U}^n}{\partial d_i}, \dot{\mathbf{U}}^n + \epsilon\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i}, \mathbf{d} + \epsilon\,\mathbf{e}_i\right)\right.\\ &\quad \left. - G^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d})\right),\end{aligned} \tag{42}$$

in (36), (37), (38), and (39). Again, we assume the user can code $\partial G^n/\partial \mathbf{U}$, $\partial G^n/\partial \dot{\mathbf{U}}$ and $\partial G^n/\partial \mathbf{d}$, so we do not use (42). As mentioned before, semi-analytical sensitivities carry the error due to ϵ_R, truncation, and round-off.

3.4 Adjoint method using differentiate-then-discretize

In the differentiate-then-discretize approach, one obtains the adjoint problem (cf. (28a) and (28b)) and the sensitivity (cf. (29)) at the continuous time level. Now we use numerical time integration to compute

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \sum_{n=0}^{N} \mu_{N-n} \left(\frac{\partial G^{N-n}}{\partial d_i} + \mathbf{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial d_i} \right) - \left(\frac{\partial G^0}{\partial \mathbf{U}} + \mathbf{\Lambda}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \right) \frac{\partial \mathbf{U}^0}{\partial d_i} . \tag{43}$$

Before we evaluate the above, we must solve the adjoint problem of (28a) and (28b). To do this, we discretize the adjoint variable $\mathbf{\Lambda}$ like $\mathbf{U}$, i.e.,

$$\mathbf{\Lambda}^n = \mathbf{\Lambda}^{n-1} + \left(\alpha\, \dot{\mathbf{\Lambda}}^n + (1-\alpha)\, \dot{\mathbf{\Lambda}}^{n-1} \right) \Delta t\, , \tag{44}$$

To reuse $\mathbf{K}^n$ like the direct method, we restrict our adjoint discussion to those $\mathbf{R}$ such that

$$\frac{\mathrm{d}}{\mathrm{d}t} \left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}} \right) = \mathbf{0} . \tag{45}$$

Notably $\partial \mathbf{R}/\partial \dot{\mathbf{U}}$ is typically interpreted as a mass matrix, so the mass matrix must be constant which is fairly common.

Referring to (28b), we initially solve the adjoint problem

$$\frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^{\top} \mathbf{\Lambda}^0 = -\frac{\partial G^N}{\partial \dot{\mathbf{U}}}^{\top} , \tag{46}$$

for $\mathbf{\Lambda}^0$ and then solve (28a) with $\mathbf{\Lambda}^0$ to evaluate $\dot{\mathbf{\Lambda}}^0$, i.e.,

$$\frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^{\top} \dot{\mathbf{\Lambda}}^0 = -\frac{\partial \mathbf{R}^N}{\partial \mathbf{U}}^{\top} \mathbf{\Lambda}^0 - \frac{\partial G^N}{\partial \mathbf{U}}^{\top} + \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^N \right)^{\top} + \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}}^2} \dot{\mathbf{U}}^N \right)^{\top} . \tag{47}$$

Note that (46) and (47) do not use the tangent stiffness matrix from the primal problem. Next, we compute

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_N \left(\frac{\partial G^N}{\partial d_i} + \mathbf{\Lambda}^{0\top} \frac{\partial \mathbf{R}^N}{\partial d_i} \right) , \tag{48}$$

cf. (29), (31), and (32). Time marching now commences for the remaining time steps, i.e., for $n = 1, 2, ..., N-1$ we evaluate $\dot{\mathbf{\Lambda}}^0$ by solving

$$\mathbf{K}^{N-n\top} \dot{\mathbf{\Lambda}}^n = -\frac{\partial G^{N-n}}{\partial \mathbf{U}}^{\top} + \left(\frac{\partial^2 G^{N-n}}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^{N-n} \right)^{\top} + \left(\frac{\partial^2 G^{N-n}}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^{N-n} \right)^{\top} - \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^{\top} \left(\mathbf{\Lambda}^{n-1} + (1-\alpha)\, \Delta t\, \dot{\mathbf{\Lambda}}^{n-1} \right) , \tag{49}$$

where $\mathbf{K}^{N-n}$ is the tangent stiffness matrix of the primal problem. Then, we compute $\mathbf{\Lambda}^n$ as per (44) and update

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} \leftarrow \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_{N-n} \left(\frac{\partial G^{N-n}}{\partial d_i} + \mathbf{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial d_i} \right) . \tag{50}$$

Finally, we solve

$$\left(\frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} + \alpha \Delta t \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}} \right)^{\top} \dot{\mathbf{\Lambda}}^N = -\frac{\partial G^0}{\partial \mathbf{U}}^{\top} + \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^0 \right)^{\top} + \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^0 \right)^{\top} - \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}^{\top} \left(\mathbf{\Lambda}^{N-1} + (1-\alpha)\, \Delta t\, \dot{\mathbf{\Lambda}}^{N-1} \right) , \tag{51}$$

for $\dot{\mathbf{\Lambda}}^N$, we evaluate $\mathbf{\Lambda}^N$ with (44) and update

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} \leftarrow \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_0 \frac{\partial G^0}{\partial d_i} - \frac{\partial G^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} + \mathbf{\Lambda}^{N\top} \left(\mu_0 \frac{\partial \mathbf{R}^0}{\partial d_i} - \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}} \frac{\partial \mathbf{U}^0}{\partial d_i} \right) , \tag{52}$$

As in (47) and (51) does not use the tangent stiffness matrix from the primal analysis.

The second derivatives $\ddot{\mathbf{U}}^n$ in (47), (49), and (51) can be computed using the known first derivatives $\ldots \dot{\mathbf{U}}^{n-1}, \dot{\mathbf{U}}^n, \dot{\mathbf{U}}^{n+1}, \ldots$ and Δt, and a second order forward difference for $\dot{\mathbf{U}}^0$, backward differences for $\dot{\mathbf{U}}^N$, and central differences for any other $\ddot{\mathbf{U}}^n$ (cf. Figure 3). The adjoint sensitivity analysis is executed after the primal analysis is concluded. Thus, we describe this algorithm by inserting node C of Fig. 3 into the flowchart of Fig. 1.

In the semi-analytical, we consider a further restriction that $\partial \mathbf{R}/\partial \mathbf{U}$ is symmetric, so the term in the adjoint load of (47) can be approximated as

$$\frac{\partial \mathbf{R}^N}{\partial \mathbf{U}}^{\top} \mathbf{\Lambda}^0 \approx \frac{1}{\epsilon} \mathbf{R}^N \left(\mathbf{U}^N + \epsilon \mathbf{\Lambda}^0, \dot{\mathbf{U}}^N, \mathbf{d} \right) , \tag{53}$$

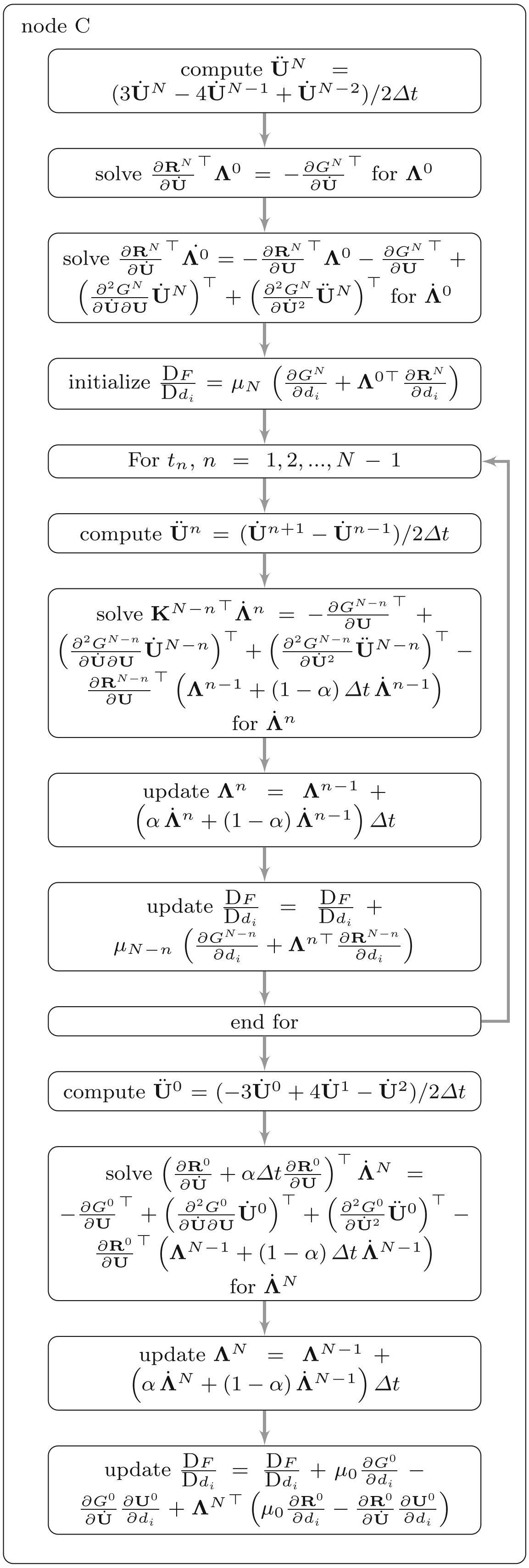
node C
compute $\ddot{\mathbf{U}}^N = (3\dot{\mathbf{U}}^N - 4\dot{\mathbf{U}}^{N-1} + \dot{\mathbf{U}}^{N-2})/2\Delta t$
solve $\frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^\top \mathbf{\Lambda}^0 = -\frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top$ for $\mathbf{\Lambda}^0$
solve $\frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^\top \dot{\mathbf{\Lambda}}^0 = -\frac{\partial \mathbf{R}^N}{\partial \mathbf{U}}^\top \mathbf{\Lambda}^0 - \frac{\partial G^N}{\partial \mathbf{U}}^\top + \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^N\right)^\top + \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^N\right)^\top$ for $\dot{\mathbf{\Lambda}}^0$
initialize $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_N \left(\frac{\partial G^N}{\partial d_i} + \mathbf{\Lambda}^{0\top} \frac{\partial \mathbf{R}^N}{\partial d_i}\right)$
For t_n, $n = 1, 2, ..., N - 1$
compute $\ddot{\mathbf{U}}^n = (\dot{\mathbf{U}}^{n+1} - \dot{\mathbf{U}}^{n-1})/2\Delta t$
solve $\mathbf{K}^{N-n\top} \dot{\mathbf{\Lambda}}^n = -\frac{\partial G^{N-n}}{\partial \mathbf{U}}^\top + \left(\frac{\partial^2 G^{N-n}}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^{N-n}\right)^\top + \left(\frac{\partial^2 G^{N-n}}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^{N-n}\right)^\top - \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^\top \left(\mathbf{\Lambda}^{n-1} + (1-\alpha)\, \Delta t\, \dot{\mathbf{\Lambda}}^{n-1}\right)$ for $\dot{\mathbf{\Lambda}}^n$
update $\mathbf{\Lambda}^n = \mathbf{\Lambda}^{n-1} + \left(\alpha\, \dot{\mathbf{\Lambda}}^n + (1-\alpha)\, \dot{\mathbf{\Lambda}}^{n-1}\right) \Delta t$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_{N-n} \left(\frac{\partial G^{N-n}}{\partial d_i} + \mathbf{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial d_i}\right)$
end for
compute $\ddot{\mathbf{U}}^0 = (-3\dot{\mathbf{U}}^0 + 4\dot{\mathbf{U}}^1 - \dot{\mathbf{U}}^2)/2\Delta t$
solve $\left(\frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} + \alpha \Delta t \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\right)^\top \dot{\mathbf{\Lambda}}^N = -\frac{\partial G^0}{\partial \mathbf{U}}^\top + \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^0\right)^\top + \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^0\right)^\top - \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}^\top \left(\mathbf{\Lambda}^{N-1} + (1-\alpha)\, \Delta t\, \dot{\mathbf{\Lambda}}^{N-1}\right)$ for $\dot{\mathbf{\Lambda}}^N$
update $\mathbf{\Lambda}^N = \mathbf{\Lambda}^{N-1} + \left(\alpha\, \dot{\mathbf{\Lambda}}^N + (1-\alpha)\, \dot{\mathbf{\Lambda}}^{N-1}\right) \Delta t$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_0 \frac{\partial G^0}{\partial d_i} - \frac{\partial G^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} + \mathbf{\Lambda}^{N\top} \left(\mu_0 \frac{\partial \mathbf{R}^0}{\partial d_i} - \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i}\right)$

Fig. 3 Adjoint differentiate-then-discretize node

and the term in the adjoint load of (49) can be approximated as

$$\frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^\top \left(\mathbf{\Lambda}^{n-1} + (1-\alpha)\, \Delta t\, \dot{\mathbf{\Lambda}}^{n-1}\right) \approx \frac{1}{\epsilon} \mathbf{R}^{N-n} \left(\mathbf{U}^{N-n} + \epsilon \left(\mathbf{\Lambda}^{n-1} + (1-\alpha)\, \Delta t\, \dot{\mathbf{\Lambda}}^{n-1}\right), \dot{\mathbf{U}}^{N-n}, \mathbf{d}\right) . \quad (54)$$

In regard to $\mathrm{D}F/\mathrm{D}d_i$ of (52), we use the approximation

$$\mu_0 \frac{\partial \mathbf{R}^0}{\partial d_i} - \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} \approx \frac{1}{\epsilon} \mathbf{R} \left(\mathbf{U}^0, \dot{\mathbf{U}}^0 - \epsilon \frac{\partial \mathbf{U}^0}{\partial d_i}, \mathbf{d} + \mu_0 \epsilon\, \mathbf{e}_i\right) . \quad (55)$$

Finally, the derivative $\partial \mathbf{R}^n / \partial d_i$ in (48) and (50) is approximated as

$$\frac{\partial \mathbf{R}^n}{\partial d_i} \approx \frac{1}{\epsilon} \mathbf{R} \left(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d} + \epsilon\, \mathbf{e}_i\right) . \quad (56)$$

Of course, the semi-analytical approximations exhibit the previously discussed errors.

Again, we assume the user can code $\partial G / \partial \mathbf{U}$, etc., as these would be time consuming to compute by finite differences.

3.5 Adjoint method using discretize-then-differentiate

In this second option of the adjoint method, we use (22a) and (35) to equivalently write (32) as

$$\begin{aligned} \frac{\mathrm{D}F}{\mathrm{D}d_i} = & \sum_{n=0}^{N} \mu_n \left(\frac{\partial G^n}{\partial \mathbf{U}} \frac{\partial \mathbf{U}^n}{\partial d_i} + \frac{\partial G^n}{\partial \mathbf{U}} \frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + \frac{\partial G^n}{\partial d_i}\right) \\ & + \sum_{n=0}^{N} \mathbf{\Lambda}^{n\top} \left(\frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}^{N-n}}{\partial d_i} \right. \\ & \left. + \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}} \frac{\partial \mathbf{U}^{N-n}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial d_i}^{N-n} \right) \\ & + \sum_{n=0}^{N-1} \mathbf{\Phi}^{n\top} \left(\frac{\partial \mathbf{U}^{N-n}}{\partial d_i} - \frac{\partial \mathbf{U}^{N-n-1}}{\partial d_i} \right. \\ & \left. - \left(\alpha \frac{\partial \dot{\mathbf{U}}^{N-n}}{\partial d_i} + (1-\alpha) \frac{\partial \dot{\mathbf{U}}^{N-n-1}}{\partial d_i}\right) \Delta t \right) , \end{aligned} \quad (57)$$

where $\boldsymbol{\Lambda}^n$ and $\boldsymbol{\Phi}^n$ are arbitrary adjoint vectors. Rearrangement subsequently yields

$$\begin{aligned}
\frac{\mathrm{D}F}{\mathrm{D}d_i} &= \sum_{n=0}^{N}\left(\mu_{N-n}\frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top}\frac{\partial \mathbf{R}}{\partial d_i}^{N-n}\right)\\
&+\left(\mu_0\frac{\partial G^0}{\partial \mathbf{U}} + \boldsymbol{\Lambda}^{N\top}\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}} - \boldsymbol{\Phi}^{N-1\top}\right)\frac{\partial \mathbf{U}^0}{\partial d_i}\\
&+\left(\mu_0\frac{\partial G^0}{\partial \dot{\mathbf{U}}} + \boldsymbol{\Lambda}^{N\top}\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}} - (1-\alpha)\Delta t\,\boldsymbol{\Phi}^{N-1\top}\right)\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}\\
&+\sum_{n=1}^{N-1}\left(\mu_{N-n}\frac{\partial G^{N-n}}{\partial \mathbf{U}} + \boldsymbol{\Lambda}^{n\top}\frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{N-n}\right.\\
&\left.\quad-\boldsymbol{\Phi}^{n-1\top} + \boldsymbol{\Phi}^{n\top}\right)\frac{\partial \mathbf{U}^{N-n}}{\partial d_i}\\
&+\sum_{n=1}^{N-1}\left(\mu_{N-n}\frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}} + \boldsymbol{\Lambda}^{n\top}\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{N-n}\right.\\
&\left.\quad-(1-\alpha)\Delta t\,\boldsymbol{\Phi}^{n-1\top} - \alpha\Delta t\,\boldsymbol{\Phi}^{n\top}\right)\frac{\partial \dot{\mathbf{U}}^{N-n}}{\partial d_i}\\
&+\left(\mu_N\frac{\partial G^N}{\partial \mathbf{U}} + \boldsymbol{\Lambda}^{0\top}\frac{\partial \mathbf{R}}{\partial \mathbf{U}}^N + \boldsymbol{\Phi}^{0\top}\right)\frac{\partial \mathbf{U}^N}{\partial d_i}\\
&+\left(\mu_N\frac{\partial G^N}{\partial \dot{\mathbf{U}}} + \boldsymbol{\Lambda}^{0\top}\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^N - \alpha\Delta t\,\boldsymbol{\Phi}^{0\top}\right)\frac{\partial \dot{\mathbf{U}}^N}{\partial d_i}\,. \qquad (58)
\end{aligned}$$

To annihilate the implicitly defined derivatives $\partial\mathbf{U}^N/\partial d_i$ and $\partial\dot{\mathbf{U}}^N/\partial d_i$, we first solve the adjoint problem

$$\mathbf{K}^{N\top}\boldsymbol{\Lambda}^0 = -\mu_N\alpha\,\Delta t\,\frac{\partial G^N}{\partial \mathbf{U}}^{\top} - \mu_N\frac{\partial G^N}{\partial \dot{\mathbf{U}}}^{\top}\,, \qquad (59)$$

for $\boldsymbol{\Lambda}^0$ and evaluate $\boldsymbol{\Phi}^0$ from either of the following expressions

$$\begin{aligned}
\boldsymbol{\Phi}^0 &= -\mu_N\frac{\partial G^N}{\partial \mathbf{U}}^{\top} - \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{N\top}\boldsymbol{\Lambda}^0 &(60)\\
&= \frac{1}{\alpha\Delta t}\left(\mu_N\frac{\partial G^N}{\partial \dot{\mathbf{U}}}^{\top} + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{N\top}\boldsymbol{\Lambda}^0\right). &(61)
\end{aligned}$$

Note that for an explicit method, i.e., $\alpha = 0$, we must use (60) to evaluate $\boldsymbol{\Phi}^0$. We next evaluate

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_N\frac{\partial G^N}{\partial d_i} + \boldsymbol{\Lambda}^{0\top}\frac{\partial \mathbf{R}}{\partial d_i}^N\,. \qquad (62)$$

To annihilate $\partial\mathbf{U}^n/\partial d_i$ and $\partial\dot{\mathbf{U}}^n/\partial d_i$, we march in time computing $\boldsymbol{\Lambda}^n$ from

$$\begin{aligned}
\mathbf{K}^{N-n\top}\boldsymbol{\Lambda}^n &= -\mu_{N-n}\,\alpha\,\Delta t\,\frac{\partial G^{N-n}}{\partial \mathbf{U}}^{\top}\\
&\quad-\mu_{N-n}\frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}}^{\top} + \Delta t\,\boldsymbol{\Phi}^{n-1}\,, \qquad (63)
\end{aligned}$$

and updating $\boldsymbol{\Phi}^n$ from either of the following equations

$$\begin{aligned}
\boldsymbol{\Phi}^n &= \boldsymbol{\Phi}^{n-1} - \mu_{N-n}\frac{\partial G^{N-n}}{\partial \mathbf{U}}^{\top} - \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{N-n\top}\boldsymbol{\Lambda}^n &(64)\\
&= -\frac{1-\alpha}{\alpha}\boldsymbol{\Phi}^{n-1}\\
&\quad+\frac{1}{\alpha\Delta t}\left(\mu_{N-n}\frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}}^{\top} + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{N-n\top}\boldsymbol{\Lambda}^n\right). &(65)
\end{aligned}$$

Again (65) is restricted to the $\alpha \neq 0$ case. Due to the different $\boldsymbol{\Phi}$ updates, we define option 1 if we choose to use (60) and (64), and option 2 if we use (61) and (65). After each of these t^n computations, we update

$$\frac{\mathrm{D}F}{\mathrm{D}d_i} \leftarrow \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_{N-n}\frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top}\frac{\partial \mathbf{R}}{\partial d_i}^{N-n}\,. \qquad (66)$$

Finally, to annihilate $\partial\dot{\mathbf{U}}^0/\partial d_i$, we solve the linear problem

$$\mathbf{K}^{0\top}\boldsymbol{\Lambda}^N = -\mu_0\frac{\partial G^0}{\partial \mathbf{U}} + (1-\alpha)\Delta t\,\boldsymbol{\Phi}^{N-1}\,, \qquad (67)$$

for $\boldsymbol{\Lambda}^N$ and update

$$\begin{aligned}
\frac{\mathrm{D}F}{\mathrm{D}d_i} \leftarrow \frac{\mathrm{D}F}{\mathrm{D}d_i} &+ \mu_0\frac{\partial G^0}{\partial d_i} + \mu_0\frac{\partial G^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} - \boldsymbol{\Phi}^{N-1\top}\frac{\partial \mathbf{U}^0}{\partial d_i}\\
&+\boldsymbol{\Lambda}^{N\top}\left(\frac{\partial \mathbf{R}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i}\right). \qquad (68)
\end{aligned}$$

All of the computations in (59)–(67) are performed after the primal analysis is terminated, thus we insert node C from Fig. 4 into the flowchart of Fig. 1.

The sensitivities using the differentiate-then-discretize and discretize-then-differentiate adjoint approaches are different because the discretization and differentiation steps do not commute. As seen shortly, the latter approach yields more accurate results. However, for large number of time steps, the time discretization error shrinks and the methods converge.

For the semi-analytical, if we use option 1, we again require $\partial\mathbf{R}^n/\partial\mathbf{U}$ to be symmetric and we approximate the adjoint load terms of (60) and (64) as

$$\frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{N-n\top}\boldsymbol{\Lambda}^n \approx \frac{1}{\epsilon}\mathbf{R}\left(\mathbf{U}^{N-n} + \epsilon\boldsymbol{\Lambda}^n, \dot{\mathbf{U}}^{N-n}, \mathbf{d}\right), \qquad (69)$$

Fortunately, we have option 2 to approximate $\boldsymbol{\Phi}^n$ if $\partial\mathbf{R}^n/\partial\mathbf{U}$ is asymmetric and we cannot use (69). We consider the restriction for which $\partial\mathbf{R}/\partial\dot{\mathbf{U}}$ is symmetric, which is common, and $\alpha \neq 0$. In this case, the adjoint load terms of (61) and (65) are approximated as

$$\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{N-n\top}\boldsymbol{\Lambda}^n \approx \frac{1}{\epsilon}\mathbf{R}\left(\mathbf{U}^{N-n}, \dot{\mathbf{U}}^{N-n} + \epsilon\boldsymbol{\Lambda}^n, \mathbf{d}\right), \qquad (70)$$

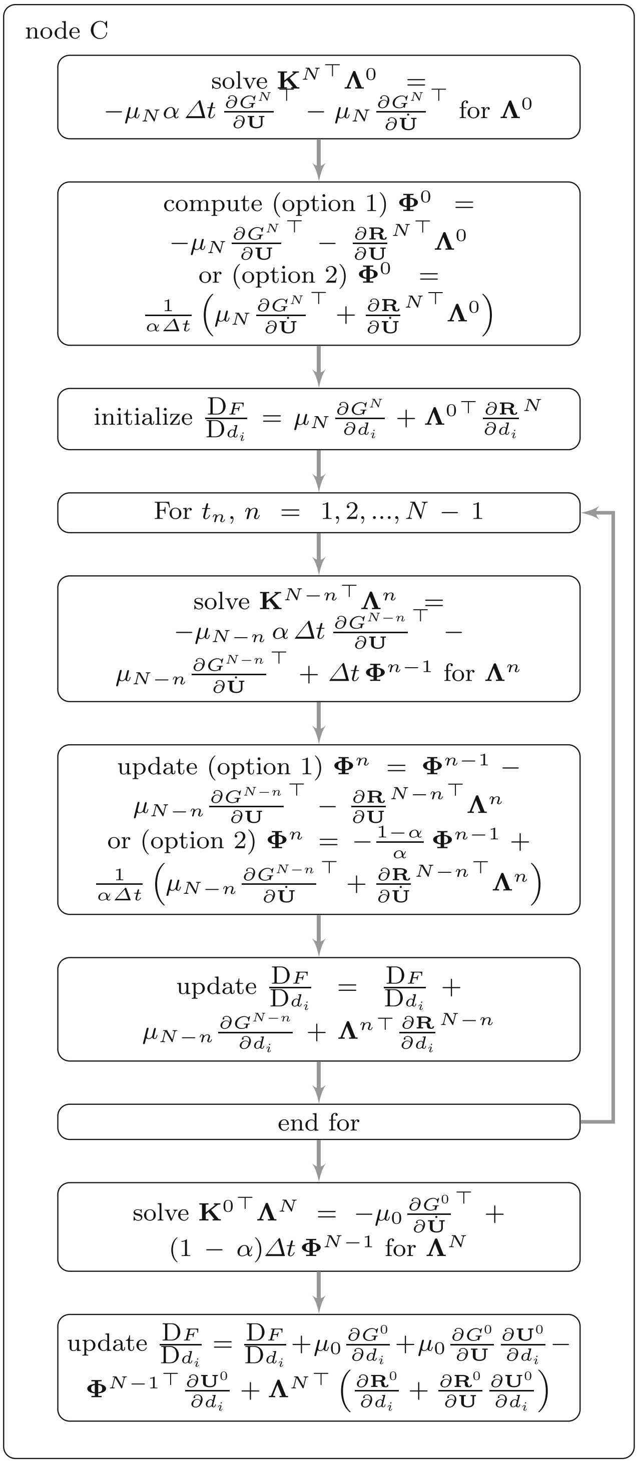
node C
solve $\mathbf{K}^{N\top}\boldsymbol{\Lambda}^0 = -\mu_N\alpha\,\Delta t\,\frac{\partial G^N}{\partial \mathbf{U}}^\top - \mu_N\frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top$ for $\boldsymbol{\Lambda}^0$
compute (option 1) $\boldsymbol{\Phi}^0 = -\mu_N\frac{\partial G^N}{\partial \mathbf{U}}^\top - \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{N\top}\boldsymbol{\Lambda}^0$
or (option 2) $\boldsymbol{\Phi}^0 = \frac{1}{\alpha\Delta t}\left(\mu_N\frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{N\top}\boldsymbol{\Lambda}^0\right)$
initialize $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_N\frac{\partial G^N}{\partial d_i} + \boldsymbol{\Lambda}^{0\top}\frac{\partial \mathbf{R}}{\partial d_i}^N$
For t_n, $n = 1, 2, ..., N-1$
solve $\mathbf{K}^{N-n\top}\boldsymbol{\Lambda}^n = -\mu_{N-n}\,\alpha\,\Delta t\,\frac{\partial G^{N-n}}{\partial \mathbf{U}}^\top - \mu_{N-n}\frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}}^\top + \Delta t\,\boldsymbol{\Phi}^{n-1}$ for $\boldsymbol{\Lambda}^n$
update (option 1) $\boldsymbol{\Phi}^n = \boldsymbol{\Phi}^{n-1} - \mu_{N-n}\frac{\partial G^{N-n}}{\partial \mathbf{U}}^\top - \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{N-n\top}\boldsymbol{\Lambda}^n$
or (option 2) $\boldsymbol{\Phi}^n = -\frac{1-\alpha}{\alpha}\boldsymbol{\Phi}^{n-1} + \frac{1}{\alpha\Delta t}\left(\mu_{N-n}\frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}}^\top + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^{N-n\top}\boldsymbol{\Lambda}^n\right)$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_{N-n}\frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top}\frac{\partial \mathbf{R}}{\partial d_i}^{N-n}$
end for
solve $\mathbf{K}^{0\top}\boldsymbol{\Lambda}^N = -\mu_0\frac{\partial G^0}{\partial \dot{\mathbf{U}}}^\top + (1-\alpha)\Delta t\,\boldsymbol{\Phi}^{N-1}$ for $\boldsymbol{\Lambda}^N$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_0\frac{\partial G^0}{\partial d_i} + \mu_0\frac{\partial G^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} - \boldsymbol{\Phi}^{N-1\top}\frac{\partial \mathbf{U}^0}{\partial d_i} + \boldsymbol{\Lambda}^{N\top}\left(\frac{\partial \mathbf{R}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i}\right)$

Fig. 4 Adjoint discretize-then-differentiate node

and in $\mathrm{D}F/\mathrm{D}d_i$ of (68) we approximate the sum

$$\frac{\partial \mathbf{R}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} \approx \frac{1}{\epsilon}\mathbf{R}\left(\mathbf{U}^0 + \epsilon\frac{\partial \mathbf{U}^0}{\partial d_i}, \dot{\mathbf{U}}^0, \mathbf{d} + \epsilon\,\mathbf{e}_i\right). \tag{71}$$

The derivatives $\partial \mathbf{R}^n/\partial d_i$ of (62), (66), and (68) are approximated via finite differences using (56). Again, these semi-analytical approximations are susceptible to the previously discussed errors.

3.6 Transient example

Consider the transient heat conduction problem of a straight one-dimensional fin with constant cross-sectional area (Kramer and Stockman 1963) expressed in non-dimensional form as

$$\begin{aligned}
\dot{\theta} - \frac{\mathrm{d}}{\mathrm{d}x}\left[k(\theta)\frac{\mathrm{d}\theta}{\mathrm{d}x}\right] + M^2\theta^{p+1} &= 0, \quad \text{in } 0 < x < 1\,,\\
\frac{\mathrm{d}\theta}{\mathrm{d}x} &= 0, \quad \text{at } x = 0,\ t > 0\,,\\
\theta &= 1, \quad \text{at } x = 1,\ t > 0\,,\\
\theta &= 1, \quad \text{at } t = 0\,,
\end{aligned} \tag{72}$$

where x, t, and θ are the non-dimensional position, time, and temperature respectively. $k(\theta) = 1 + \xi\theta$ is the non-dimensional thermal conductivity, ξ and $M = 1$ are fin parameters, and the exponent $p = 1/3$ models the removal of heat by turbulent natural convection along the fin. The bar is discretized by 5 equal length linear finite elements and the time domain [0, 1] is discretized into N equal time steps. The Newton-Raphson tolerance is $\epsilon_R = |\mathbf{R}| < 10^{-14}$. The various sensitivity methods are illustrated for the following response function

$$F = \int_0^2\int_0^1 \left(\zeta\theta^2(x,t) + (1-\zeta)\dot{\theta}^2(x,t)\right)\mathrm{d}x\,\mathrm{d}t \tag{73}$$

where the integral is approximated by using the trapezoidal rule in time and the element wise 2-point Gaussian quadrature in space. We use $\zeta = 0.5$ and compute the sensitivities with respect to the parameter $d = M$. The perturbation $\epsilon = 10^{-6}$ is used in the finite difference and semi-analytical approaches, unless otherwise stated.

3.6.1 Symmetric $\partial \mathbf{R}/\partial \mathbf{U}$ and $\partial \mathbf{R}/\partial \dot{\mathbf{U}}$

We first consider the linear thermal conductivity case, i.e., $\xi = 0$, for which $\partial \mathbf{R}/\partial \mathbf{U}$ and $\partial \mathbf{R}/\partial \dot{\mathbf{U}}$ are symmetric. The computations performed with the various methods yield similar results, cf. Table 1. For $N = 100$ and $\alpha = 0$, the explicit integration scheme is not stable. Also, for $\alpha = 0$, we cannot use the semi-analytical adjoint discretize-then-differentiate option 2, cf. (61) and (65).

To examine the consistency of the methods, we show the error e_f, cf. (1), for different perturbation sizes ϵ for the $N = 1000$ and $\alpha = 0.5$ case, cf. Figure 5. As the perturbation ϵ decreases, the sensitivities obtained by finite differences converge to those obtained analytically.

Table 1 Sensitivities for the symmetric problem

Method	$N = 100$	$N = 1000$
$\alpha = 0$		
Direct	–	−0.018588335152
Semi-a. direct	–	−0.018588317345
Adj. diff.-then-disc.	–	−0.018449255359
Semi-a. adj. diff.-then-disc.	–	−0.018449264996
Adj. disc.-then-diff. opt. 1	–	−0.018588335152
Semi-a. adj. disc.-then-diff. opt. 1	–	−0.018588346428
Adj. disc.-then-diff. opt. 2	–	–
Semi-a. adj. disc.-then-diff. opt. 2	–	–
Finite differences	–	−0.018588118600
$\alpha = 0.5$		
Direct	−0.017592906261	−0.018015117466
Semi-a. direct	−0.017592887840	−0.018015099351
Adj. diff.-then-disc.	−0.017654283299	−0.018015355049
Semi-a. adj. diff.-then-disc.	−0.017654292528	−0.018015364471
Adj. disc.-then-diff. opt. 1	−0.017592906261	−0.018015117466
Semi-a. adj. disc.-then-diff. opt. 1	−0.017592915281	−0.018015131326
Adj. disc.-then-diff. opt. 2	−0.017592906261	−0.018015117466
Semi-a. adj. disc.-then-diff. opt. 2	−0.017592914483	−0.018015125166
Finite differences	−0.017592690027	−0.018014899961
$\alpha = 1$		
Direct	−0.011964307217	−0.017441917048
Semi-a. direct	−0.011964285704	−0.017441898622
Adj. diff.-then-disc.	−0.013595022204	−0.017585996028
Semi-a. adj. diff.-then-disc.	−0.013595029346	−0.017586005227
Adj. disc.-then-diff. opt. 1	−0.011964307217	−0.017441917048
Semi-a. adj. disc.-then-diff. opt. 1	−0.011964319986	−0.017441927075
Adj. disc.-then-diff. opt. 2	−0.011964307217	−0.017441917048
Semi-a. adj. disc.-then-diff. opt. 2	−0.011964305753	−0.017441924379
Finite differences	−0.011964084834	−0.017441698807

However, the finite difference sensitivities erode for small perturbations due to round-off error.

We also show the error e_f for different time discretizations N for the $\epsilon = 10^{-6}$ and $\alpha = 0.5$ case, cf. Figure 6. The errors of the sensitivities obtained by the different methods show no dependency on N, with the exception of the adjoint differentiate-then-discretize scheme. As expected, this sensitivity has a consistency error that decreases as the number of time steps increases (Gunzburger 2003; Jensen et al. 2014).

To examine the accuracy of the semi-analytical sensitivities, we compare them to their respective analytical sensitivities via the error e_s of (2) for different perturbation sizes ϵ and the $N = 1000$ and $\alpha = 0.5$ case, cf. Figure 7. As expected, the error is smaller as the perturbation size decreases until round-off error pollutes the computations.

In Fig. 8, we show the error e_s for different time discretization N using the $\epsilon = 10^{-6}$ and $\alpha = 0.5$ case.

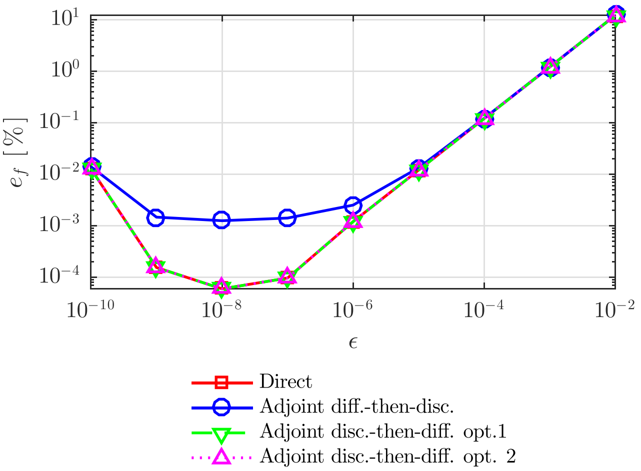

Fig. 5 Relative percentage error of the sensitivities obtained by the analytical methods with respect to finite differences for the symmetric problem for $\alpha = 0.5$ and $N = 1000$

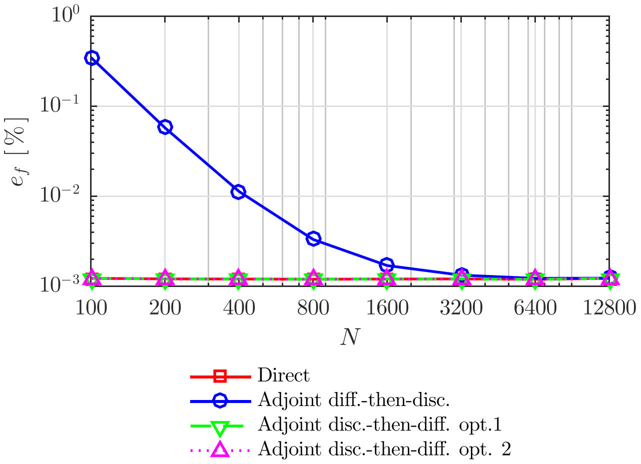

Fig. 6 Relative percentage error of the sensitivities obtained by the analytical methods with respect to finite differences for the symmetric problem for $\alpha = 0.5$ and $\epsilon = 10^{-6}$

The errors of the semi-analytical sensitivities show no dependency on N because the semi-analytical approximations are independent of the time discretization, i.e., the error is solely due to the perturbation size ϵ.

3.6.2 Asymmetric $\partial \mathbf{R}/\partial \mathbf{U}$

We consider the nonlinear thermal conductivity case where $\xi = 0.5$ for which only $\partial \mathbf{R}/\partial \dot{\mathbf{U}}$ is symmetric and $\partial \mathbf{R}/\partial \mathbf{U}$ is not. The computations performed with the various methods yield similar results, cf. Table 2, with the exception of the semi-analytical adjoint differentiate-then-discretize and semi-analytical adjoint discretize-then-differentiate option 1 schemes, which exhibit errors of approximately 0.1% with respect to their analytical counter parts. We attribute this error to the asymmetric $\partial \mathbf{R}/\partial \mathbf{U}$. Again for $N = 100$, the explicit $\alpha = 0$ scheme is not stable.

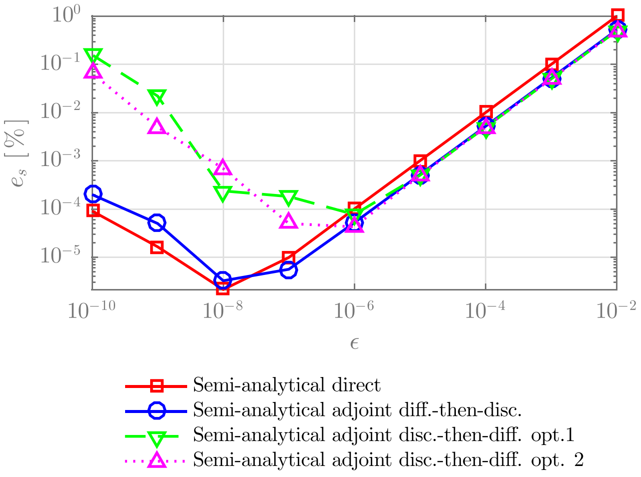

Fig. 7 Relative percentage error of the semi-analytical sensitivities of the symmetric problem for $\alpha = 0.5$ and $N = 1000$

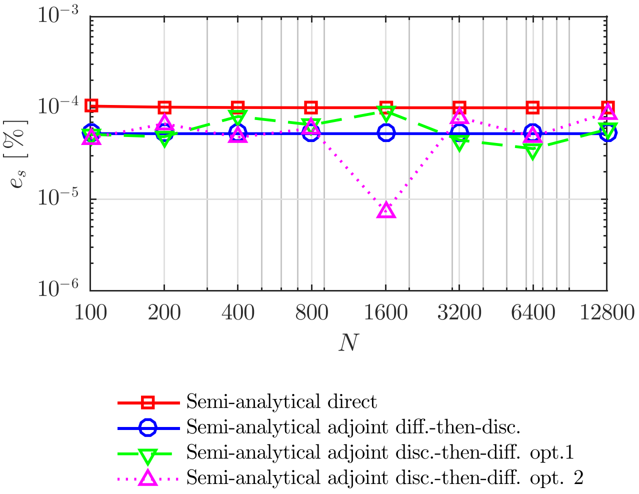

Fig. 8 Relative percentage error of the semi-analytical sensitivities of the symmetric problem for $\alpha = 0.5$ and $\epsilon = 10^{-6}$

To examine the consistency of the methods, we show the error e_f for different perturbation sizes and time steps in Figs. 9 and 10. Again as shown in the previous example, the adjoint method differentiate-then-discretize has a consistency error that decreases as the number of time steps increases.

Now we examine the accuracy of the semi-analytical sensitivities, computing the error e_s for different perturbation sizes ϵ with $N = 1000$ and $\alpha = 0.5$, cf. Figure 11. Since $\partial \mathbf{R}/\partial \mathbf{U}$ is not symmetric, (54) and (69) do not hold, resulting in appreciable error in both the semi-analytical adjoint differentiate-then-discretize and the semi-analytical adjoint discretize-then-differentiate option 1 schemes. The other semi-analytical methods do not exhibit this error. Again as the perturbation size decreases, the error lessens until round-off error pollutes the computations.

In Fig. 12, we show the error e_s for different time discretization N using the $\epsilon = 10^{-6}$ and $\alpha = 0.5$ case. The error for the semi-analytical adjoint differentiate-then-discretize and the semi-analytical adjoint discretize-then-differentiate option 1 schemes is evident.

4 Nonlinear dynamic problems

A nonlinear dynamic problem can be expressed through a residual as

$$\mathbf{R}(\mathbf{U}(t,\mathbf{d}), \dot{\mathbf{U}}(t,\mathbf{d}), \ddot{\mathbf{U}}(t,\mathbf{d}), \mathbf{d}) = \mathbf{0}\,, \tag{74a}$$

$$\dot{\mathbf{U}}(0) = \dot{\mathbf{U}}^0\,, \tag{74b}$$

$$\mathbf{U}(0) = \mathbf{U}^0\,, \tag{74c}$$

where we note the design dependencies as in (7). The response function for this system is again expressed by (20) and its sensitivity computed by (21).

Table 2 Sensitivities for the asymmetric problem

Method	$N = 100$	$N = 1000$
$\alpha = 0$		
Direct	–	−0.036458929390
Semi-a. direct	–	−0.036458918886
Adj. diff.-then-disc.	–	−0.036366841077
Semi-a. adj. diff.-then-disc.	–	−0.036712508113
Adj. disc.-then-diff. opt. 1	–	−0.036458929390
Semi-a. adj. disc.-then-diff. opt. 1	–	−0.036805361657
Adj. disc.-then-diff. opt. 2	–	–
Semi-a. adj. disc.-then-diff. opt. 2	–	–
Finite differences	–	−0.036458757458
$\alpha = 0.5$		
Direct	−0.035207467583	−0.035893035577
Semi-a. direct	−0.035207456378	−0.035893024774
Adj. diff.-then-disc.	−0.035306636741	−0.035893239979
Semi-a. adj. diff.-then-disc.	−0.035646027889	−0.036238627524
Adj. disc.-then-diff. opt. 1	−0.035207467583	−0.035893035577
Semi-a. adj. disc.-then-diff. opt. 1	−0.035548997747	−0.036238714626
Adj. disc.-then-diff. opt. 2	−0.035207467583	−0.035893035577
Semi-a. adj. disc.-then-diff. opt. 2	−0.035207482293	−0.035893051987
Finite differences	−0.035207295634	−0.035892861794
$\alpha = 1$		
Direct	−0.029694346673	−0.035327122553
Semi-a. direct	−0.029694332634	−0.035327111453
Adj. diff.-then-disc.	−0.030954122631	−0.035426388647
Semi-a. adj. diff.-then-disc.	−0.031289705532	−0.035771465710
Adj. disc.-then-diff. opt. 1	−0.029694346673	−0.035327122553
Semi-a. adj. disc.-then-diff. opt. 1	−0.030028066780	−0.035672041022
Adj. disc.-then-diff. opt. 2	−0.029694346673	−0.035327122553
Semi-a. adj. disc.-then-diff. opt. 2	−0.029694357644	−0.035327141665
Finite differences	−0.029694167625	−0.035326948922

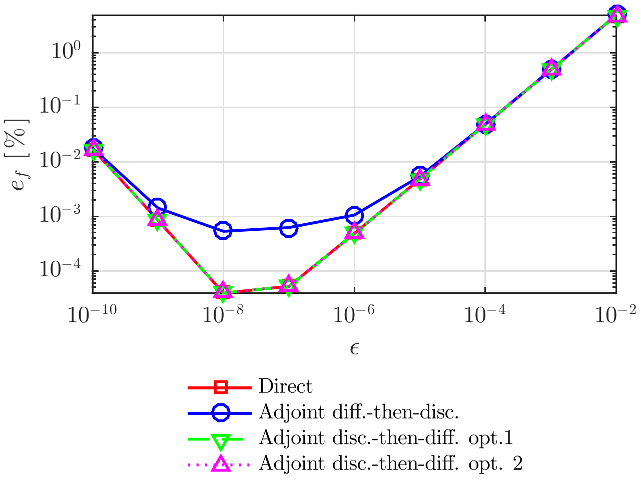

Fig. 9 Relative percentage error of the sensitivities obtained by the analytical methods with respect to finite differences for the asymmetric problem for $\alpha = 0.5$ and $N = 1000$

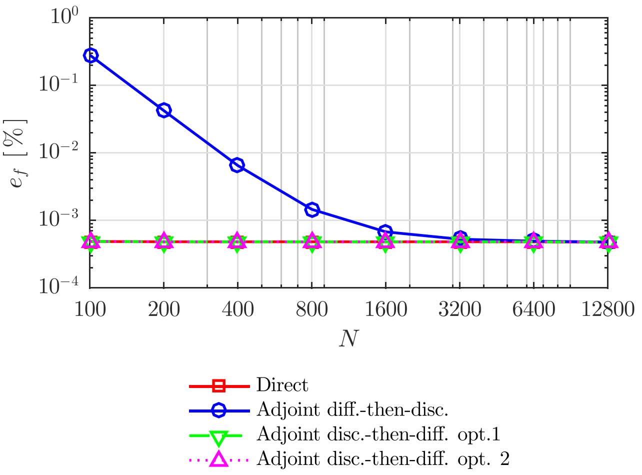

Fig. 10 Relative percentage error of the sensitivities obtained by the analytical methods with respect to finite differences for the asymmetric problem for $\alpha = 0.5$ and $\epsilon = 10^{-6}$

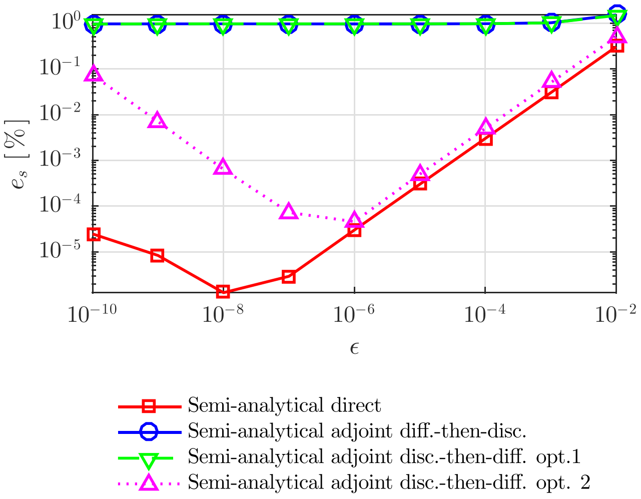

Fig. 11 Relative percentage error of the semi-analytical sensitivities of the asymmetric problem for $\alpha = 0.5$ and $N = 1000$

For the sensitivity analysis, we implement the direct method by differentiating (74a), (74b), and (74c)

$$\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}\frac{\partial \ddot{\mathbf{U}}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} = -\frac{\partial \mathbf{R}}{\partial d_i}\,, \tag{75a}$$

$$\frac{\partial \dot{\mathbf{U}}(0)}{\partial d_i} = \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}\,, \tag{75b}$$

$$\frac{\partial \mathbf{U}(0)}{\partial d_i} = \frac{\partial \mathbf{U}^0}{\partial d_i}\,, \tag{75c}$$

and solve the resulting pseudo problem for $\partial \ddot{\mathbf{U}}/\partial d_i$, $\partial \dot{\mathbf{U}}/\partial d_i$, and $\partial \mathbf{U}/\partial d_i$ whereupon we evaluate (21).

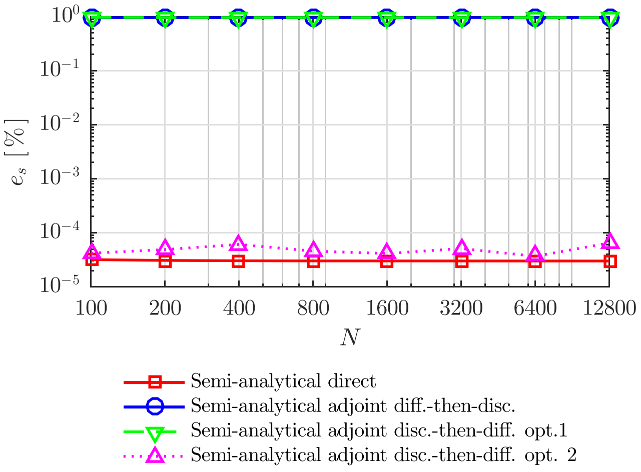

Fig. 12 Relative percentage error of the semi-analytical sensitivities of the asymmetric problem for $\alpha = 0.5$ and $\epsilon = 10^{-6}$

Alternatively, we can implement the adjoint method whereby we insert (75a) into (21) to obtain the equivalent sensitivity

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i} &= \int_0^{t_f}\left(\frac{\partial G}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} + \frac{\partial G}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}}{\partial d_i} + \frac{\partial G}{\partial d_i}\right)\mathrm{d}t \\ &\quad + \int_0^{t_f}\boldsymbol{\lambda}^\top\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}\frac{\partial \ddot{\mathbf{U}}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial \mathbf{U}}\frac{\partial \mathbf{U}}{\partial d_i} + \frac{\partial \mathbf{R}}{\partial d_i}\right)\mathrm{d}t\,.\end{aligned} \tag{76}$$

Where again $\boldsymbol{\lambda}$ is the arbitrary adjoint vector. Integrating by parts and rearranging (76) yields

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i} =& \int_0^{t_f}\left(\frac{\partial G}{\partial d_i} + \boldsymbol{\lambda}^\top\frac{\partial \mathbf{R}}{\partial d_i}\right)\mathrm{d}t \\ & - \frac{\partial \mathbf{U}^0}{\partial d_i}^\top\left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^\top + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^\top\boldsymbol{\lambda} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\boldsymbol{\lambda}\right)\right)\Bigg|_{t=0} \\ & - \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}^\top\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\boldsymbol{\lambda}\right)\Bigg|_{t=0} + \int_0^{t_f}\frac{\partial \mathbf{U}}{\partial d_i}^\top\left(\frac{\partial G}{\partial \mathbf{U}}^\top - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^\top\right)\right. \\ & \left. + \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^\top\boldsymbol{\lambda} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^\top\boldsymbol{\lambda}\right) + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\boldsymbol{\lambda}\right)\right)\mathrm{d}t \\ & + \frac{\partial \mathbf{U}}{\partial d_i}^\top\left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^\top + \frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^\top\boldsymbol{\lambda} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\boldsymbol{\lambda}\right)\right)\Bigg|_{t=t_f} \\ & + \frac{\partial \dot{\mathbf{U}}}{\partial d_i}^\top\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\boldsymbol{\lambda}\right)\Bigg|_{t=t_f}\,.\end{aligned} \tag{77}$$

Next, we introduce the time mapping of (25) and substitute it into the above (77) to obtain

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i} =& \int_0^{t_f}\left(\frac{\partial G}{\partial d_i} + \boldsymbol{\Lambda}^\top\frac{\partial \mathbf{R}}{\partial d_i}\right)\mathrm{d}t \\ & - \frac{\partial \mathbf{U}^0}{\partial d_i}^\top\left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^\top + \left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^\top - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\right)\right)\boldsymbol{\Lambda} + \frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\dot{\boldsymbol{\Lambda}}\right)\Bigg|_{t=0} \\ & - \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}^\top\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\boldsymbol{\Lambda}\right)\Bigg|_{t=0} + \int_0^{t_f}\frac{\partial \mathbf{U}}{\partial d_i}^\top\left(\frac{\partial G}{\partial \mathbf{U}}^\top - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^\top\right)\right. \\ & + \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}^\top - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^\top\right) + \frac{\mathrm{d}^2}{\mathrm{d}t^2}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\right)\right)\boldsymbol{\Lambda} \\ & \left. + \left(\frac{\partial \mathbf{R}}{\dot{\mathbf{U}}}^\top - 2\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\right)\right)\dot{\boldsymbol{\Lambda}} + \frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\ddot{\boldsymbol{\Lambda}}\right)\mathrm{d}t \\ & + \frac{\partial \mathbf{U}}{\partial d_i}^\top\left(\frac{\partial G}{\partial \dot{\mathbf{U}}}^\top + \left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}^\top - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\right)\right)\boldsymbol{\Lambda} + \frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\dot{\boldsymbol{\Lambda}}\right)\Bigg|_{t=t_f} \\ & + \frac{\partial \dot{\mathbf{U}}}{\partial d_i}^\top\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}^\top\boldsymbol{\Lambda}\right)\Bigg|_{t=t_f}\,.\end{aligned} \tag{78}$$

where all quantities are evaluated at time t except for $\boldsymbol{\Lambda}$ which is evaluated at $t_f - t$. We annihilate the terms containing the implicitly defined derivative $\partial\mathbf{U}/\partial d_i$ by requiring $\boldsymbol{\Lambda}$ to solve

$$\begin{aligned}&\frac{\partial\mathbf{R}}{\partial\ddot{\mathbf{U}}}^{\top}\ddot{\boldsymbol{\Lambda}}+\left(\frac{\partial\mathbf{R}}{\partial\dot{\mathbf{U}}}^{\top}-2\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\mathbf{R}}{\partial\ddot{\mathbf{U}}}^{\top}\right)\right)\dot{\boldsymbol{\Lambda}}\\&+\left(\frac{\partial\mathbf{R}}{\partial\mathbf{U}}^{\top}-\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\mathbf{R}}{\partial\dot{\mathbf{U}}}^{\top}\right)+\frac{\mathrm{d}^2}{\mathrm{d}t^2}\left(\frac{\partial\mathbf{R}}{\partial\ddot{\mathbf{U}}}^{\top}\right)\right)\boldsymbol{\Lambda}\\&=-\frac{\partial G}{\partial\mathbf{U}}^{\top}+\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial G}{\partial\dot{\mathbf{U}}}^{\top}\right),\end{aligned}\tag{79a}$$

$$\left.\frac{\partial\mathbf{R}}{\partial\ddot{\mathbf{U}}}^{\top}\right|_{t=t_f}\dot{\boldsymbol{\Lambda}}(0)=-\left.\frac{\partial G}{\partial\dot{\mathbf{U}}}^{\top}\right|_{t=t_f},\tag{79b}$$

$$\boldsymbol{\Lambda}(0)=\mathbf{0}\,.\tag{79c}$$

Using this $\boldsymbol{\Lambda}$, the sensitivity reduces to

$$\begin{aligned}\frac{\mathrm{D}F}{\mathrm{D}d_i}=&\int_0^{t_f}\left(\frac{\partial G}{\partial d_i}+\boldsymbol{\Lambda}^{\top}\frac{\partial\mathbf{R}}{\partial d_i}\right)\mathrm{d}t\\&-\frac{\partial\mathbf{U}^0}{\partial d_i}^{\top}\left(\frac{\partial G}{\partial\dot{\mathbf{U}}}^{\top}+\left(\frac{\partial\mathbf{R}}{\partial\dot{\mathbf{U}}}^{\top}-\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\mathbf{R}}{\partial\ddot{\mathbf{U}}}^{\top}\right)\right)\boldsymbol{\Lambda}\right.\\&\left.\left.+\frac{\partial\mathbf{R}}{\partial\ddot{\mathbf{U}}}^{\top}\dot{\boldsymbol{\Lambda}}\right)\right|_{t=0}-\frac{\partial\dot{\mathbf{U}}^0}{\partial d_i}^{\top}\left.\left(\frac{\partial\mathbf{R}}{\partial\ddot{\mathbf{U}}}^{\top}\boldsymbol{\Lambda}\right)\right|_{t=0}.\end{aligned}\tag{80}$$

where again all quantities are evaluated at time t except for $\boldsymbol{\Lambda}$ which is evaluated at $t_f - t$.

4.1 Discretization

To solve the above, we discretize in time using the Newmark method so that

$$\dot{\mathbf{U}}^n=\dot{\mathbf{U}}^{n-1}+(1-\gamma)\Delta t\ddot{\mathbf{U}}^{n-1}+\gamma\Delta t\ddot{\mathbf{U}}^n\,,\tag{81}$$

$$\begin{aligned}\mathbf{U}^n=&\,\mathbf{U}^{n-1}+\Delta t\dot{\mathbf{U}}^{n-1}\\&+\left(\frac{1}{2}-\beta\right)\Delta t^2\ddot{\mathbf{U}}^{n-1}+\beta\Delta t^2\ddot{\mathbf{U}}^n\,,\end{aligned}\tag{82}$$

where $\mathbf{U}^n=\mathbf{U}(t_n)$, $\dot{\mathbf{U}}^n=\dot{\mathbf{U}}(t_n)$ and $\ddot{\mathbf{U}}^n=\ddot{\mathbf{U}}(t_n)$. To simplify the ensuing developments, we define coefficients $a=(1-\gamma)\Delta t$, $b=\gamma\Delta t$, $c=\Delta t$, $d=(1/2-\beta)\Delta t^2$, and $e=\beta\Delta t^2$.

4.2 Primal analysis

In the primal analysis, we are given the initial condition $\mathbf{U}^0$ and $\dot{\mathbf{U}}^0$, so first we use (74a) and solve

$$\mathbf{R}^0(\mathbf{U}^0,\dot{\mathbf{U}}^0,\ddot{\mathbf{U}}^0,\mathbf{d})=\mathbf{0}\,,\tag{83}$$

for $\ddot{\mathbf{U}}^0$ by Newton-Raphson. The updates $\Delta\ddot{\mathbf{U}}^0$ for $\ddot{\mathbf{U}}^0$ are obtained by solving

$$\mathbf{K}^0(\mathbf{U}^0,\dot{\mathbf{U}}^0,\ddot{\mathbf{U}}^0,\mathbf{d})\Delta\ddot{\mathbf{U}}^0=-\mathbf{R}^0(\mathbf{U}^0,\dot{\mathbf{U}}^0,\ddot{\mathbf{U}}^0,\mathbf{d})\,,\tag{84}$$

where $\mathbf{K}^0=\partial\mathbf{R}^0/\partial\ddot{\mathbf{U}}$ is the tangent matrix. We continue updating until convergence.

Having $\mathbf{U}^0$, $\dot{\mathbf{U}}^0$ and $\ddot{\mathbf{U}}^0$, we compute the first term in (31), i.e., $F=\mu_0\,G^0(\mathbf{U}^0,\dot{\mathbf{U}}^0,\mathbf{d})$.

Now we commence our analysis. At each time step t_n, we replace $\dot{\mathbf{U}}^n$ and $\mathbf{U}^n$ with the right-hand side (RHS) of (81) and (82), solve (74a) for $\ddot{\mathbf{U}}^n$ and then evaluate $\dot{\mathbf{U}}^n$ and $\mathbf{U}^n$ from (81) and (82). Newton's method is also used for these solves, whereupon we calculate the update $\Delta\ddot{\mathbf{U}}^n$ from the linear equation

$$\mathbf{K}^n(\mathbf{U}^n,\dot{\mathbf{U}}^n,\ddot{\mathbf{U}}^n,\mathbf{d})\Delta\ddot{\mathbf{U}}^n=-\mathbf{R}^n(\mathbf{U}^n,\dot{\mathbf{U}}^n,\ddot{\mathbf{U}}^n,\mathbf{d})\,,\tag{85}$$

where $\mathbf{K}^n=\partial\mathbf{R}^n/\partial\ddot{\mathbf{U}}+b\,\partial\mathbf{R}^n/\partial\dot{\mathbf{U}}+e\,\partial\mathbf{R}^n/\partial\mathbf{U}$ is the tangent stiffness matrix. After convergence, we update F as per (34). A flowchart of these computations appears in Fig. 13.

4.3 Direct differentiation

For the direct differentiation sensitivity analysis, we discretize $\partial\mathbf{U}/\partial d_i$ like $\mathbf{U}$, i.e.,

$$\frac{\partial\dot{\mathbf{U}}^n}{\partial d_i}=\frac{\partial\dot{\mathbf{U}}^{n-1}}{\partial d_i}+a\frac{\partial\ddot{\mathbf{U}}^{n-1}}{\partial d_i}+b\frac{\partial\ddot{\mathbf{U}}^n}{\partial d_i}\,,\tag{86}$$

$$\frac{\partial\mathbf{U}^n}{\partial d_i}=\frac{\partial\mathbf{U}^{n-1}}{\partial d_i}+c\frac{\partial\dot{\mathbf{U}}^{n-1}}{\partial d_i}+d\frac{\partial\ddot{\mathbf{U}}^{n-1}}{\partial d_i}+e\frac{\partial\ddot{\mathbf{U}}^n}{\partial d_i}\,.\tag{87}$$

Note that the initial condition $\partial\mathbf{U}^0/\partial d_i$ and $\partial\dot{\mathbf{U}}^0/\partial d_i$ are known, but $\partial\ddot{\mathbf{U}}^0/\partial d_i$ is not. So before commencing, we must obtain $\partial\ddot{\mathbf{U}}^0/\partial d_i$ like we did $\ddot{\mathbf{U}}^0$. To these ends, we differentiate (83) to obtain the linear equation

$$\mathbf{K}^0\frac{\partial\ddot{\mathbf{U}}^0}{\partial d_i}=-\left(\frac{\partial\mathbf{R}^0}{\partial\mathbf{U}}\frac{\partial\mathbf{U}^0}{\partial d_i}+\frac{\partial\mathbf{R}^0}{\partial\mathbf{U}}\frac{\partial\dot{\mathbf{U}}^0}{\partial d_i}+\frac{\partial\mathbf{R}^0}{\partial d_i}\right),\tag{88}$$

which we solve for $\partial\ddot{\mathbf{U}}^0/\partial d_i$. Having $\partial\dot{\mathbf{U}}^0/\partial d_i$ and $\partial\mathbf{U}^0/\partial d_i$ we update $\mathrm{D}F/\mathrm{D}d_i$ as per (37). Now we march in time evaluating $\partial\mathbf{U}^n/\partial d_i$, $\partial\dot{\mathbf{U}}^n/\partial d_i$ and $\partial\ddot{\mathbf{U}}^n/\partial d_i$ as we did to compute $\mathbf{U}^n$, $\dot{\mathbf{U}}^n$, and $\ddot{\mathbf{U}}^n$. From (75a), (86), and (87) we formulate the linear equation

$$\begin{aligned}\mathbf{K}^n\frac{\partial\ddot{\mathbf{U}}^n}{\partial d_i}=&-\frac{\partial\mathbf{R}^n}{\partial\mathbf{U}}\left(\frac{\partial\mathbf{U}^{n-1}}{\partial d_i}+c\frac{\partial\dot{\mathbf{U}}^{n-1}}{\partial d_i}+d\frac{\partial\ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right)\\&-\frac{\partial\mathbf{R}^n}{\partial\dot{\mathbf{U}}}\left(\frac{\partial\dot{\mathbf{U}}^{n-1}}{\partial d_i}+a\frac{\partial\ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right)-\frac{\partial\mathbf{R}^n}{\partial d_i}\,.\end{aligned}\tag{89}$$

We solve the above (89) for $\partial\ddot{\mathbf{U}}^n/\partial d_i$ and update $\partial\dot{\mathbf{U}}^n/\partial d_i$ and $\partial\mathbf{U}^n/\partial d_i$ via (86) and (87) and $\mathrm{D}F/\mathrm{D}d_i$ via (39). We continue marching in this manner for all t_n. In so far as our

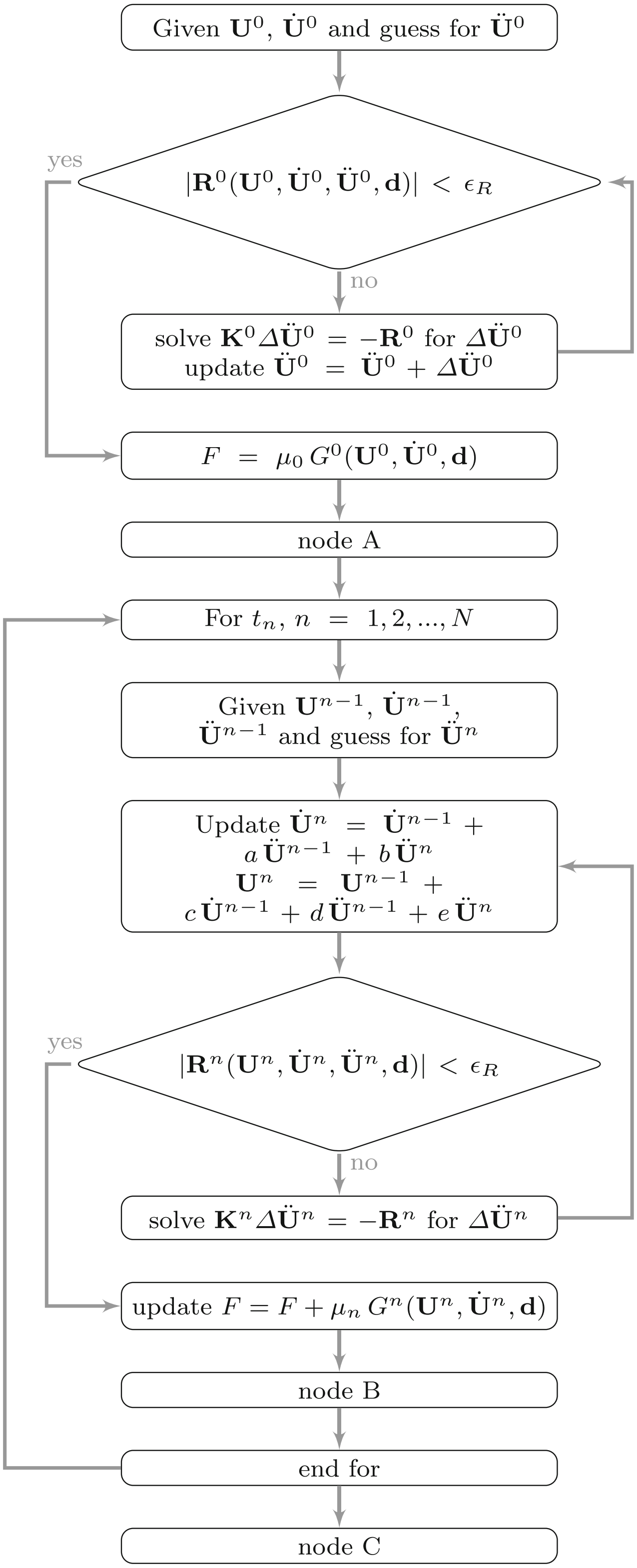
Given $\mathbf{U}^0$, $\dot{\mathbf{U}}^0$ and guess for $\ddot{\mathbf{U}}^0$
yes
$|\mathbf{R}^0(\mathbf{U}^0, \dot{\mathbf{U}}^0, \ddot{\mathbf{U}}^0, \mathbf{d})| < \epsilon_R$
no
solve $\mathbf{K}^0 \Delta\ddot{\mathbf{U}}^0 = -\mathbf{R}^0$ for $\Delta\ddot{\mathbf{U}}^0$
update $\ddot{\mathbf{U}}^0 = \ddot{\mathbf{U}}^0 + \Delta\ddot{\mathbf{U}}^0$
$F = \mu_0\, G^0(\mathbf{U}^0, \dot{\mathbf{U}}^0, \mathbf{d})$
node A
For t_n, $n = 1, 2, ..., N$
Given $\mathbf{U}^{n-1}$, $\dot{\mathbf{U}}^{n-1}$, $\ddot{\mathbf{U}}^{n-1}$ and guess for $\ddot{\mathbf{U}}^n$
Update $\dot{\mathbf{U}}^n = \dot{\mathbf{U}}^{n-1} + a\,\ddot{\mathbf{U}}^{n-1} + b\,\ddot{\mathbf{U}}^n$
$\mathbf{U}^n = \mathbf{U}^{n-1} + c\,\dot{\mathbf{U}}^{n-1} + d\,\ddot{\mathbf{U}}^{n-1} + e\,\ddot{\mathbf{U}}^n$
yes
$|\mathbf{R}^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \ddot{\mathbf{U}}^n, \mathbf{d})| < \epsilon_R$
no
solve $\mathbf{K}^n \Delta\ddot{\mathbf{U}}^n = -\mathbf{R}^n$ for $\Delta\ddot{\mathbf{U}}^n$
update $F = F + \mu_n\, G^n(\mathbf{U}^n, \dot{\mathbf{U}}^n, \mathbf{d})$
node B
end for
node C

Fig. 13 Primal analysis flowchart for dynamic problem

sensitivity analysis algorithm is concerned, we insert nodes A and B from Fig. 14 into the primal analysis flowchart of Fig. 13.

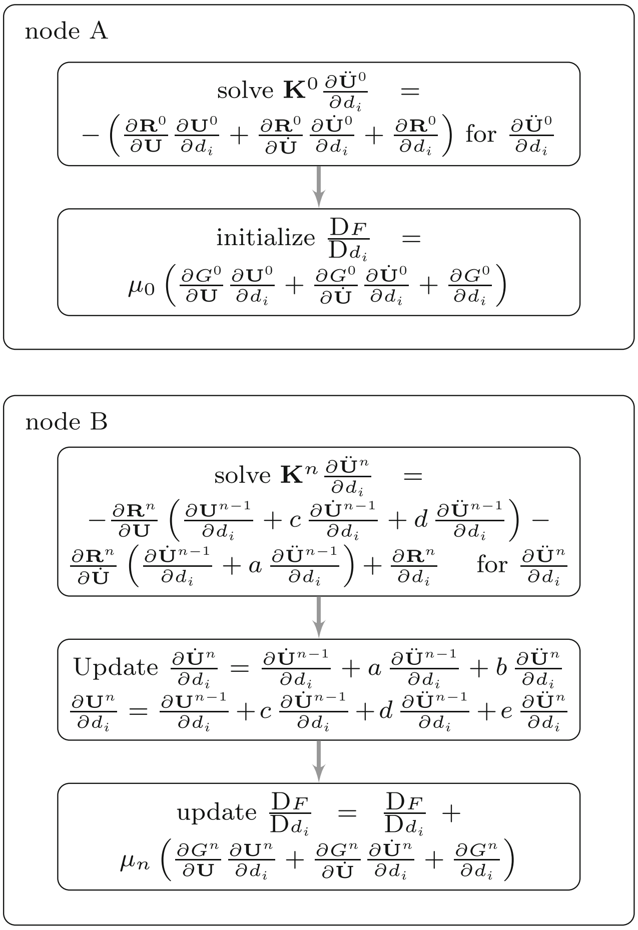
node A
solve $\mathbf{K}^0 \frac{\partial \ddot{\mathbf{U}}^0}{\partial d_i} = -\left(\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial d_i}\right)$ for $\frac{\partial \ddot{\mathbf{U}}^0}{\partial d_i}$
initialize $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_0\left(\frac{\partial G^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial G^0}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} + \frac{\partial G^0}{\partial d_i}\right)$
node B
solve $\mathbf{K}^n \frac{\partial \ddot{\mathbf{U}}^n}{\partial d_i} = -\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}}\left(\frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + c\,\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + d\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right) - \frac{\partial \mathbf{R}^n}{\partial \dot{\mathbf{U}}}\left(\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + a\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right) + \frac{\partial \mathbf{R}^n}{\partial d_i}$ for $\frac{\partial \ddot{\mathbf{U}}^n}{\partial d_i}$
Update $\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} = \frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + a\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i} + b\,\frac{\partial \ddot{\mathbf{U}}^n}{\partial d_i}$
$\frac{\partial \mathbf{U}^n}{\partial d_i} = \frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + c\,\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + d\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i} + e\,\frac{\partial \ddot{\mathbf{U}}^n}{\partial d_i}$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_n\left(\frac{\partial G^n}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^n}{\partial d_i} + \frac{\partial G^n}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + \frac{\partial G^n}{\partial d_i}\right)$

Fig. 14 Direct differentiation nodes for dynamic problem

For semi-analytical, we have the approximations

$$\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial d_i} \approx \frac{1}{\epsilon}\mathbf{R}^0\left(\mathbf{U}^0+\epsilon\frac{\partial \mathbf{U}^0}{\partial d_i}, \dot{\mathbf{U}}^0+\epsilon\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}, \ddot{\mathbf{U}}^0, \mathbf{d}+\epsilon\,\mathbf{e}_i\right), \tag{90}$$

$$\begin{aligned}&\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}}\left(\frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + c\,\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + d\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right)\\ &+\frac{\partial \mathbf{R}^n}{\partial \dot{\mathbf{U}}}\left(\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + a\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right) + \frac{\partial \mathbf{R}^n}{\partial d_i}\\ \approx\ &\frac{1}{\epsilon}\mathbf{R}^n\left(\mathbf{U}^n + \epsilon\left(\frac{\partial \mathbf{U}^{n-1}}{\partial d_i} + c\,\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + d\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right),\right.\\ &\left.\dot{\mathbf{U}}^n + \epsilon\left(\frac{\partial \dot{\mathbf{U}}^{n-1}}{\partial d_i} + a\,\frac{\partial \ddot{\mathbf{U}}^{n-1}}{\partial d_i}\right), \ddot{\mathbf{U}}^n, \mathbf{d} + \epsilon\,\mathbf{e}_i\right),\end{aligned} \tag{91}$$

which we use in (88) and (89). Again, we assume the user can code $\partial G^n/\partial \mathbf{U}$, $\partial G^n/\partial \dot{\mathbf{U}}$ and $\partial G^n/\partial d_i$.

4.4 Adjoint method using differentiate-then-discretize

In the adjoint differentiate-then-discretize approach, we discretize the adjoint problem and sensitivity of (79a) and (80). (80) is evaluated as

$$
\begin{aligned}
\frac{\mathrm{D}F}{\mathrm{D}d_i} = \sum_{n=0}^{N} \mu_{N-n} \left(\frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial d_i} \right) \\
- \frac{\partial \mathbf{U}^0}{\partial \dot{\mathbf{U}}}^{\top} \left(\frac{\partial G^0}{\partial \dot{\mathbf{U}}}^{\top} + \left(\frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}}^{\top} - \frac{\mathrm{d}}{\mathrm{d}t} \left(\frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}}^{\top} \right) \right) \boldsymbol{\Lambda}^N \right. \\
\left. + \frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}}^{\top} \dot{\boldsymbol{\Lambda}}^N \right) - \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}^{\top} \left(\frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}}^{\top} \boldsymbol{\Lambda}^N \right) .
\end{aligned}
\tag{92}
$$

To obtain $\boldsymbol{\Lambda}^n$, we solve (79a), (79b), and (79c) like we did for $\mathbf{U}$, i.e., we introduce the Newmark time stepping scheme

$$
\dot{\boldsymbol{\Lambda}}^n = \dot{\boldsymbol{\Lambda}}^{n-1} + a\, \ddot{\boldsymbol{\Lambda}}^{n-1} + b\, \ddot{\boldsymbol{\Lambda}}^n , \tag{93}
$$

$$
\boldsymbol{\Lambda}^n = \boldsymbol{\Lambda}^{n-1} + c\, \dot{\boldsymbol{\Lambda}}^{n-1} + d\, \ddot{\boldsymbol{\Lambda}}^{n-1} + e\, \ddot{\boldsymbol{\Lambda}}^n . \tag{94}
$$

To reuse $\mathbf{K}^n$ like the direct method, we restrict $\mathbf{R}$ such that

$$
\frac{\mathrm{d}}{\mathrm{d}t} \left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}} \right) = \mathbf{0} , \tag{95}
$$

$$
\frac{\mathrm{d}}{\mathrm{d}t} \left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}} \right) = \mathbf{0} . \tag{96}
$$

This means, $\partial \mathbf{R}/\partial \dot{\mathbf{U}}$ and $\partial \mathbf{R}/\partial \dot{\mathbf{U}}$ which are typically interpreted as damping and mass matrices respectively, are constant.

Noting that $\boldsymbol{\Lambda}^0 = \mathbf{0}$ from (79c), we start the algorithm by solving (79b), i.e.,

$$
\frac{\partial \mathbf{R}^N}{\partial \ddot{\mathbf{U}}}^{\top} \dot{\boldsymbol{\Lambda}}^0 = - \frac{\partial G^N}{\partial \dot{\mathbf{U}}}^{\top} , \tag{97}
$$

for $\dot{\boldsymbol{\Lambda}}^0$. Next, we obtain $\ddot{\boldsymbol{\Lambda}}^0$ from (79a), i.e.,

$$
\begin{aligned}
\frac{\partial \mathbf{R}^N}{\partial \ddot{\mathbf{U}}}^{\top} \ddot{\boldsymbol{\Lambda}}^0 = - \frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^{\top} \dot{\boldsymbol{\Lambda}}^0 - \frac{\partial G^N}{\partial \mathbf{U}}^{\top} \\
+ \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^N \right)^{\top} + \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^N \right)^{\top} .
\end{aligned}
\tag{98}
$$

Notice that (97) *and* (98) *do not use the tangent stiffness matrix of the primal analysis.* Next we initialize $\mathrm{D}F/\mathrm{D}d_i$ from (48).

The time marching now commences for the remaining in time steps t_n, i.e., for $n = 1, 2, ..., N - 1$ we solve

$$
\begin{aligned}
\mathbf{K}^{N-n\top} \ddot{\boldsymbol{\Lambda}}^n = - \frac{\partial G^{N-n}}{\mathbf{U}}^{\top} + \left(\frac{\partial^2 G^{N-n}}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^{N-n} \right)^{\top} \\
+ \left(\frac{\partial^2 G^{N-n}}{\partial \mathbf{U}^2} \ddot{\mathbf{U}}^{N-n} \right)^{\top} \\
- \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^{\top} \left(\boldsymbol{\Lambda}^{n-1} + c\, \dot{\boldsymbol{\Lambda}}^{n-1} + d\, \ddot{\boldsymbol{\Lambda}}^{n-1} \right) \\
- \frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}}^{\top} \left(\dot{\boldsymbol{\Lambda}}^{n-1} + a\, \ddot{\boldsymbol{\Lambda}}^{n-1} \right) ,
\end{aligned}
\tag{99}
$$

for $\ddot{\boldsymbol{\Lambda}}^n$. Then, we update $\boldsymbol{\Lambda}^n$ and $\dot{\boldsymbol{\Lambda}}^n$ with (93) and (94) and $\mathrm{D}F/\mathrm{D}d_i$ with (50).

Finally, we solve

$$
\begin{aligned}
\left(\frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} + b\, \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} + e\, \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}} \right)^{\top} \ddot{\boldsymbol{\Lambda}}^N = - \frac{\partial G^0}{\partial \mathbf{U}}^{\top} \\
+ \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^0 \right)^{\top} + \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^0 \right)^{\top} \\
- \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}^{\top} \left(\boldsymbol{\Lambda}^{N-1} + c\, \dot{\boldsymbol{\Lambda}}^{N-1} + d\, \ddot{\boldsymbol{\Lambda}}^{N-1} \right) \\
- \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}}^{\top} \left(\dot{\boldsymbol{\Lambda}}^{N-1} + a\, \ddot{\boldsymbol{\Lambda}}^{N-1} \right) ,
\end{aligned}
\tag{100}
$$

for $\ddot{\boldsymbol{\Lambda}}^N$, then obtain $\dot{\boldsymbol{\Lambda}}^N$ and $\boldsymbol{\Lambda}^N$ from (93) and (94), and update

$$
\begin{aligned}
\frac{\mathrm{D}F}{\mathrm{D}d_i} \leftarrow \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_0 \frac{\partial G^0}{\partial d_i} + \mu_0\, \boldsymbol{\Lambda}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \mathbf{d}} - \frac{\partial G^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} \\
- \boldsymbol{\Lambda}^{N\top} \left(\frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} \right) - \dot{\boldsymbol{\Lambda}}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} .
\end{aligned}
\tag{101}
$$

Again, we note that (100) *does not use the tangent stiffness matrix from primal problem.* This algorithm is described by inserting node C from Fig. 15 into the flowchart of Fig. 13.

For the semi-analytical, we consider the further restriction that $\partial \mathbf{R}/\partial \mathbf{U}$ and $\partial \mathbf{R}/\partial \dot{\mathbf{U}}$ are symmetric. In this way, the term in the adjoint load of (98) can be approximated as

$$
\frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^{\top} \dot{\boldsymbol{\Lambda}}^0 \approx \frac{1}{\epsilon} \mathbf{R} \left(\mathbf{U}^N , \dot{\mathbf{U}}^N + \epsilon \dot{\boldsymbol{\Lambda}}^0 , \ddot{\mathbf{U}}^N , \mathbf{d} \right) , \tag{102}
$$

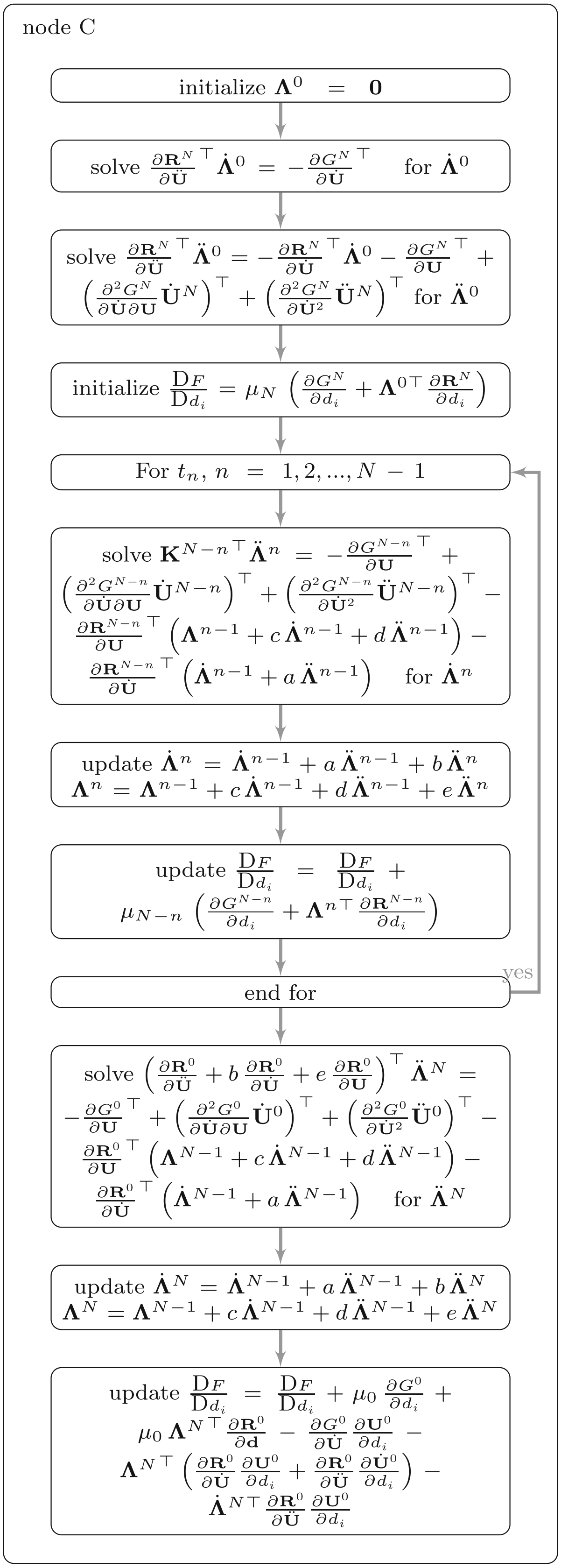
node C
initialize $\boldsymbol{\Lambda}^0 = \mathbf{0}$
solve $\frac{\partial \mathbf{R}^N}{\partial \ddot{\mathbf{U}}}^\top \dot{\boldsymbol{\Lambda}}^0 = -\frac{\partial G^N}{\partial \ddot{\mathbf{U}}}^\top$ for $\dot{\boldsymbol{\Lambda}}^0$
solve $\frac{\partial \mathbf{R}^N}{\partial \ddot{\mathbf{U}}}^\top \ddot{\boldsymbol{\Lambda}}^0 = -\frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^\top \dot{\boldsymbol{\Lambda}}^0 - \frac{\partial G^N}{\partial \mathbf{U}}^\top + \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^N\right)^\top + \left(\frac{\partial^2 G^N}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^N\right)^\top$ for $\ddot{\boldsymbol{\Lambda}}^0$
initialize $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \mu_N \left(\frac{\partial G^N}{\partial d_i} + \boldsymbol{\Lambda}^{0\top} \frac{\partial \mathbf{R}^N}{\partial d_i}\right)$
For t_n, $n = 1, 2, ..., N - 1$
solve $\mathbf{K}^{N-n\,\top} \ddot{\boldsymbol{\Lambda}}^n = -\frac{\partial G^{N-n}}{\partial \mathbf{U}}^\top + \left(\frac{\partial^2 G^{N-n}}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^{N-n}\right)^\top + \left(\frac{\partial^2 G^{N-n}}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^{N-n}\right)^\top - \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^\top \left(\boldsymbol{\Lambda}^{n-1} + c\, \dot{\boldsymbol{\Lambda}}^{n-1} + d\, \ddot{\boldsymbol{\Lambda}}^{n-1}\right) - \frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}}^\top \left(\dot{\boldsymbol{\Lambda}}^{n-1} + a\, \ddot{\boldsymbol{\Lambda}}^{n-1}\right)$ for $\dot{\boldsymbol{\Lambda}}^n$
update $\dot{\boldsymbol{\Lambda}}^n = \dot{\boldsymbol{\Lambda}}^{n-1} + a\, \ddot{\boldsymbol{\Lambda}}^{n-1} + b\, \ddot{\boldsymbol{\Lambda}}^n$ $\boldsymbol{\Lambda}^n = \boldsymbol{\Lambda}^{n-1} + c\, \dot{\boldsymbol{\Lambda}}^{n-1} + d\, \ddot{\boldsymbol{\Lambda}}^{n-1} + e\, \ddot{\boldsymbol{\Lambda}}^n$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_{N-n} \left(\frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial d_i}\right)$
yes
end for
solve $\left(\frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} + b\, \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} + e\, \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\right)^\top \ddot{\boldsymbol{\Lambda}}^N = -\frac{\partial G^0}{\partial \mathbf{U}}^\top + \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}} \partial \mathbf{U}} \dot{\mathbf{U}}^0\right)^\top + \left(\frac{\partial^2 G^0}{\partial \dot{\mathbf{U}}^2} \ddot{\mathbf{U}}^0\right)^\top - \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}^\top \left(\boldsymbol{\Lambda}^{N-1} + c\, \dot{\boldsymbol{\Lambda}}^{N-1} + d\, \ddot{\boldsymbol{\Lambda}}^{N-1}\right) - \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}}^\top \left(\dot{\boldsymbol{\Lambda}}^{N-1} + a\, \ddot{\boldsymbol{\Lambda}}^{N-1}\right)$ for $\ddot{\boldsymbol{\Lambda}}^N$
update $\dot{\boldsymbol{\Lambda}}^N = \dot{\boldsymbol{\Lambda}}^{N-1} + a\, \ddot{\boldsymbol{\Lambda}}^{N-1} + b\, \ddot{\boldsymbol{\Lambda}}^N$ $\boldsymbol{\Lambda}^N = \boldsymbol{\Lambda}^{N-1} + c\, \dot{\boldsymbol{\Lambda}}^{N-1} + d\, \ddot{\boldsymbol{\Lambda}}^{N-1} + e\, \ddot{\boldsymbol{\Lambda}}^N$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_0 \frac{\partial G^0}{\partial d_i} + \mu_0\, \boldsymbol{\Lambda}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \mathbf{d}} - \frac{\partial G^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} - \boldsymbol{\Lambda}^{N\top} \left(\frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}\right) - \dot{\boldsymbol{\Lambda}}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i}$

Fig. 15 Adjoint differentiate-then-discretize node for dynamic problem

and the terms in the adjoint load of (99) and (100) can be approximated as

$$\frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^\top \left(\boldsymbol{\Lambda}^{n-1} + c\, \dot{\boldsymbol{\Lambda}}^{n-1} + d\, \ddot{\boldsymbol{\Lambda}}^{n-1}\right) + \frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}}^\top \left(\dot{\boldsymbol{\Lambda}}^{n-1} + a\, \ddot{\boldsymbol{\Lambda}}^{n-1}\right) \approx \frac{1}{\epsilon} \mathbf{R}\left(\mathbf{U}^{N-n} + \epsilon \left(\boldsymbol{\Lambda}^{n-1} + c\, \dot{\boldsymbol{\Lambda}}^{n-1} + d\, \ddot{\boldsymbol{\Lambda}}^{n-1}\right), \dot{\mathbf{U}}^{N-n} + \epsilon \left(\dot{\boldsymbol{\Lambda}}^{n-1} + a\, \ddot{\boldsymbol{\Lambda}}^{n-1}\right), \ddot{\mathbf{U}}^{N-n}, \mathbf{d}\right). \tag{103}$$

Regarding $\mathrm{D}F/\mathrm{D}d_i$ of (48), (50), and (101), we can use the approximations

$$\frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} \approx \frac{1}{\epsilon} \mathbf{R}\left(\mathbf{U}^0, \dot{\mathbf{U}}^0 + \epsilon \frac{\partial \mathbf{U}^0}{\partial d_i}, \ddot{\mathbf{U}}^0 + \epsilon \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}, \mathbf{d}\right), \tag{104}$$

$$\frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} \frac{\partial \mathbf{U}^0}{\partial d_i} \approx \frac{1}{\epsilon} \mathbf{R}\left(\mathbf{U}^0, \dot{\mathbf{U}}^0, \ddot{\mathbf{U}}^0 + \epsilon \frac{\partial \mathbf{U}^0}{\partial d_i}, \mathbf{d}\right), \tag{105}$$

$$\frac{\partial \mathbf{R}^n}{\partial d_i} \approx \frac{1}{\epsilon} \mathbf{R}\left(\mathbf{U}^n, \dot{\mathbf{U}}^n, \ddot{\mathbf{U}}^n, \mathbf{d} + \epsilon\, \mathbf{e}_i\right). \tag{106}$$

4.5 Adjoint method using discretize-then-differentiate

In this adjoint discretize-then-differentiate method, we first discretize the primal analysis and response function in time and then we differentiate for the sensitivity analysis. Thus, we incorporate (75a), (86), and (87) into (32) to obtain the equivalent sensitivity

$$\begin{aligned}
\delta F = & \sum_{n=0}^{N} \mu_n \left(\frac{\partial G^n}{\partial \mathbf{U}} \frac{\partial \mathbf{U}^n}{\partial d_i} + \frac{\partial G^n}{\partial \dot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}^n}{\partial d_i} + \frac{\partial G^n}{\partial d_i}\right) \\
& + \sum_{n=0}^{N} \boldsymbol{\Lambda}^{n\top} \left(\frac{\partial \mathbf{R}^{N-n}}{\partial \ddot{\mathbf{U}}} \frac{\partial \ddot{\mathbf{U}}^{N-n}}{\partial d_i} + \frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}^{N-n}}{\partial d_i}\right. \\
& \left. + \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}} \frac{\partial \mathbf{U}^{N-n}}{\partial d_i} + \frac{\partial \mathbf{R}^{N-n}}{\partial d_i}\right) \\
& + \sum_{n=0}^{N-1} \boldsymbol{\Phi}^{n\top} \left(\frac{\partial \dot{\mathbf{U}}^{N-n}}{\partial d_i} - \frac{\partial \dot{\mathbf{U}}^{N-n-1}}{\partial d_i}\right. \\
& \left. - a \frac{\partial \ddot{\mathbf{U}}^{N-n}}{\partial d_i} - \frac{\partial \ddot{\mathbf{U}}^{N-n}}{\partial d_i}\right) \\
& + \sum_{n=0}^{N-1} \boldsymbol{\Psi}^{n\top} \left(\frac{\partial \mathbf{U}^{N-n}}{\partial d_i} - \frac{\partial \mathbf{U}^{N-n-1}}{\partial d_i} - c \frac{\partial \dot{\mathbf{U}}^{N-n-1}}{\partial d_i}\right. \\
& \left. - d \frac{\partial \ddot{\mathbf{U}}^{N-n-1}}{\partial d_i} - \frac{\partial \ddot{\mathbf{U}}^{N-n}}{\partial d_i}\right),
\end{aligned} \tag{107}$$

where $\boldsymbol{\Lambda}^n$, $\boldsymbol{\Phi}^n$, and $\boldsymbol{\Psi}^n$ are arbitrary adjoint vectors. Rearranging the above yields

$$\begin{aligned}
\frac{\mathrm{D}F}{\mathrm{D}d_i} &= \sum_{n=0}^{N} \left(\mu_{N-n} \frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial d_i} \right) \\
&+ \left(\mu_0 \frac{\partial G^0}{\partial \mathbf{U}} + \boldsymbol{\Lambda}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}} - \boldsymbol{\Psi}^{N-1\top} \right) \frac{\partial \mathbf{U}^0}{\partial d_i} \\
&+ \left(\mu_0 \frac{\partial G^0}{\partial \dot{\mathbf{U}}} + \boldsymbol{\Lambda}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} - \boldsymbol{\Phi}^{N-1\top} - c\, \boldsymbol{\Psi}^{N-1\top} \right) \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} \\
&+ \left(\boldsymbol{\Lambda}^{N\top} \frac{\partial \mathbf{R}^0}{\partial \ddot{\mathbf{U}}} - a\, \boldsymbol{\Phi}^{N-1\top} - d\, \boldsymbol{\Psi}^{N-1\top} \right) \frac{\partial \ddot{\mathbf{U}}^0}{\partial d_i} \\
&+ \sum_{n=1}^{N-1} \left(\mu_{N-n} \frac{\partial G^{N-n}}{\mathbf{U}} + \boldsymbol{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}} \right. \\
&+ \left. \boldsymbol{\Psi}^{n\top} - \boldsymbol{\Psi}^{n-1\top} \right) \frac{\partial \mathbf{U}^{N-n}}{\partial d_i} \\
&+ \sum_{n=1}^{N-1} \left(\mu_{N-n} \frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}^n} \partial d_i + \boldsymbol{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}} \right. \\
&+ \left. \boldsymbol{\Phi}^{n\top} - \boldsymbol{\Phi}^{n-1\top} - c\, \boldsymbol{\Psi}^{n-1\top} \right) \frac{\partial \dot{\mathbf{U}}^{N-n}}{\partial d_i} \\
&+ \sum_{n=1}^{N-1} \left(\boldsymbol{\Lambda}^{n\top} \frac{\partial \mathbf{R}^{N-n}}{\partial \ddot{\mathbf{U}}} - b\, \boldsymbol{\Phi}^{n\top} - a\, \boldsymbol{\Phi}^{n-1\top} \right. \\
&- \left. e\, \boldsymbol{\Psi}^{n\top} - d\, \boldsymbol{\Psi}^{n-1\top} \right) \frac{\partial \ddot{\mathbf{U}}^{N-n}}{\partial d_i} \\
&+ \left(\mu_N \frac{\partial G^N}{\partial \mathbf{U}} + \boldsymbol{\Lambda}^{0\top} \frac{\partial \mathbf{R}^N}{\partial \mathbf{U}} + \boldsymbol{\Psi}^{0\top} \right) \frac{\partial \mathbf{U}^N}{\partial d_i} \\
&+ \left(\mu_N \frac{\partial G^N}{\partial \dot{\mathbf{U}}} + \boldsymbol{\Lambda}^{0\top} \frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}^n} \partial d_i + \boldsymbol{\Phi}^{0\top} \right) \frac{\partial \dot{\mathbf{U}}^N}{\partial d_i} \\
&+ \left(\boldsymbol{\Lambda}^{0\top} \frac{\partial \mathbf{R}^N}{\partial \ddot{\mathbf{U}}} - b\, \boldsymbol{\Phi}^{0\top} - e\, \boldsymbol{\Psi}^{0\top} \right) \frac{\partial \ddot{\mathbf{U}}^N}{\partial d_i} .
\end{aligned} \tag{108}$$

To annihilate $\partial \ddot{\mathbf{U}}^N / \partial d_i$, $\partial \dot{\mathbf{U}}^N / \partial d_i$ and $\partial \mathbf{U}^N / \partial d_i$, we first solve the adjoint problem

$$\mathbf{K}^{N\top} \boldsymbol{\Lambda}^0 = -b\, \mu_N \frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top - e\, \mu_N \frac{\partial G^N}{\partial \mathbf{U}}^\top , \tag{109}$$

for $\boldsymbol{\Lambda}^0$, then we evaluate $\boldsymbol{\Phi}^0$ from

$$\boldsymbol{\Phi}^0 = -\mu_N \frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top - \frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^0 . \tag{110}$$

and $\boldsymbol{\Psi}^0$ from either of the following options

$$\boldsymbol{\Psi}^0 = -\mu_N \frac{\partial G^N}{\partial \mathbf{U}}^\top - \frac{\partial \mathbf{R}^N}{\partial \mathbf{U}}^\top \boldsymbol{\Lambda}^0 , \tag{111}$$

$$= \frac{1}{e} \left(\frac{\partial \mathbf{R}^N}{\partial \ddot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^0 - b\, \boldsymbol{\Phi}^0 \right) , \tag{112}$$

where (112) holds for $\beta \neq 0$. We next initialize the sensitivity from (62).

To annihilate $\partial \ddot{\mathbf{U}}^{N-n} / \partial d_i$, $\partial \dot{\mathbf{U}}^{N-n} / \partial d_i$ and $\partial \mathbf{U}^{N-n} / \partial d_i$, we march in time t_n for $n = 1, 2, ..., N-1$ by solving

$$\begin{aligned}
\mathbf{K}^{N-n\top} \boldsymbol{\Lambda}^n &= -b\, \mu_N \frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top - e\, \mu_N \frac{\partial G^N}{\partial \mathbf{U}}^\top \\
&+ \Delta t\, \boldsymbol{\Phi}^{n-1} + \left(\gamma + \frac{1}{2} \right) \Delta t^2 \boldsymbol{\Psi}^{n-1} ,
\end{aligned} \tag{113}$$

for $\boldsymbol{\Lambda}^n$, updating $\boldsymbol{\Phi}^n$ from

$$\boldsymbol{\Phi}^n = \boldsymbol{\Phi}^{n-1} + c\, \boldsymbol{\Psi}^{n-1} - \mu_{N-n} \frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}^n}^\top - \frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^n , \tag{114}$$

computing $\boldsymbol{\Psi}^n$ by either option

$$\boldsymbol{\Psi}^n = \boldsymbol{\Psi}^{n-1} - \mu_{N-n} \frac{\partial G^{N-n}}{\partial \mathbf{U}}^\top - \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^\top \boldsymbol{\Lambda}^n , \tag{115}$$

$$= \frac{1}{e} \left(-d\, \boldsymbol{\Psi}^{n-1} + \frac{\partial \mathbf{R}^{N-n}}{\partial \ddot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^n - b\, \boldsymbol{\Phi}^n - a\, \boldsymbol{\Phi}^{n-1} \right) \tag{116}$$

and updating $\mathrm{D}F / \mathrm{D}d_i$ from (66).

Finally, to annihilate $\partial \ddot{\mathbf{U}}^0 / \partial d_i$, we solve

$$\mathbf{K}^{0\top} \boldsymbol{\Lambda}^N = a\, \boldsymbol{\Phi}^{N-1} + d\, \boldsymbol{\Psi}^{N-1} , \tag{117}$$

for $\boldsymbol{\Lambda}^N$ and we update

$$\begin{aligned}
\frac{\mathrm{D}F}{\mathrm{D}d_i} &\leftarrow \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_0 \frac{\partial G^0}{\partial d_i} + \boldsymbol{\Lambda}^{N\top} \frac{\partial \mathbf{R}^0}{\partial d_i} \\
&+ \left(\mu_0 \frac{\partial G^0}{\partial \mathbf{U}} - \boldsymbol{\Psi}^{N-1\top} \right) \frac{\partial \mathbf{U}^0}{\partial d_i} \\
&+ \left(\mu_0 \frac{\partial G^0}{\partial \dot{\mathbf{U}}} - \boldsymbol{\Phi}^{N-1\top} - c\, \boldsymbol{\Psi}^{N-1\top} \right) \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} \\
&+ \boldsymbol{\Lambda}^{N\top} \left(\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}} \frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} \right) .
\end{aligned} \tag{118}$$

This algorithm is obtained by inserting node C from Fig. 16 into the primal analysis flowchart of Fig. 13.

For semi-analytical implementation, we require $\partial \mathbf{R}^n / \partial \dot{\mathbf{U}}$ to be symmetric. The adjoint load terms of (110) and (114) are thusly approximated as

$$\frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^n \approx \frac{1}{\epsilon} \mathbf{R} \left(\mathbf{U}^{N-n}, \dot{\mathbf{U}}^{N-n} + \epsilon \boldsymbol{\Lambda}^n, \ddot{\mathbf{U}}^{N-n}, \mathbf{d} \right) . \tag{119}$$

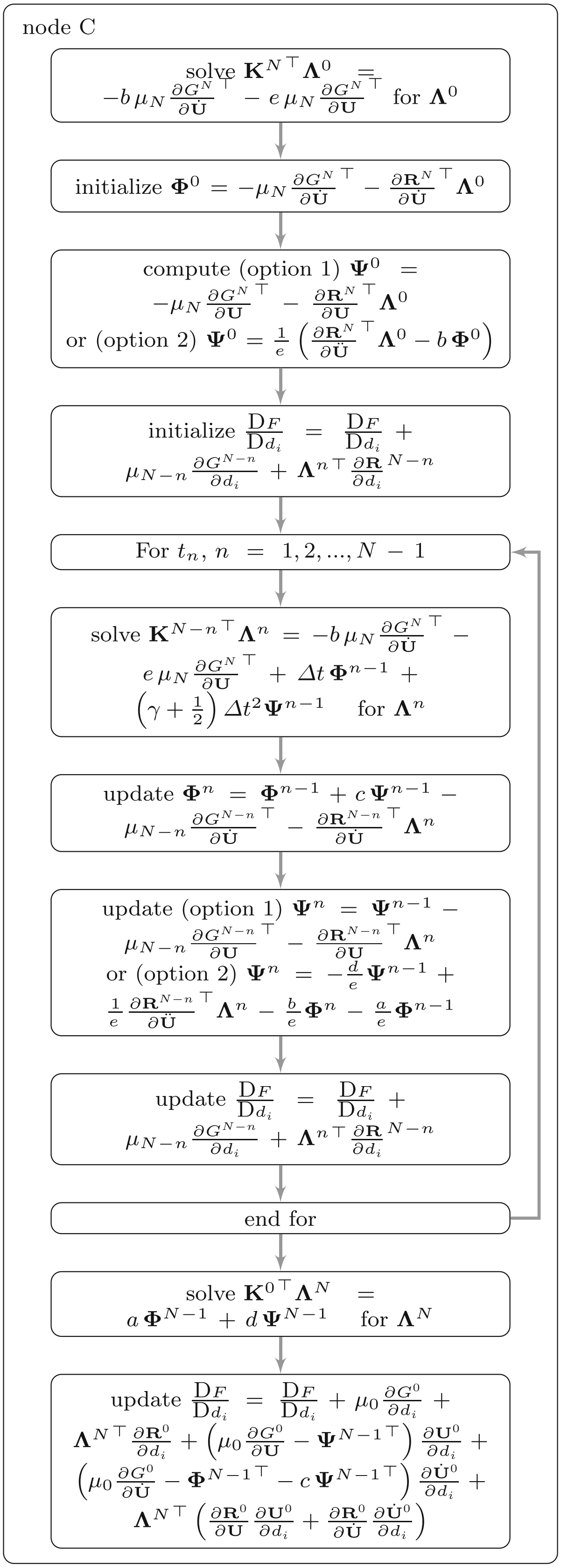
node C
solve $\mathbf{K}^{N\top}\boldsymbol{\Lambda}^0 = -b\,\mu_N \frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top - e\,\mu_N \frac{\partial G^N}{\partial \mathbf{U}}^\top$ for $\boldsymbol{\Lambda}^0$
initialize $\boldsymbol{\Phi}^0 = -\mu_N \frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top - \frac{\partial \mathbf{R}^N}{\partial \dot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^0$
compute (option 1) $\boldsymbol{\Psi}^0 = -\mu_N \frac{\partial G^N}{\partial \mathbf{U}}^\top - \frac{\partial \mathbf{R}^N}{\partial \mathbf{U}}^\top \boldsymbol{\Lambda}^0$
or (option 2) $\boldsymbol{\Psi}^0 = \frac{1}{e}\left(\frac{\partial \mathbf{R}^N}{\partial \ddot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^0 - b\,\boldsymbol{\Phi}^0\right)$
initialize $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_{N-n}\frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top}\frac{\partial \mathbf{R}}{\partial d_i}^{N-n}$
For t_n, $n = 1, 2, ..., N - 1$
solve $\mathbf{K}^{N-n\top}\boldsymbol{\Lambda}^n = -b\,\mu_N \frac{\partial G^N}{\partial \dot{\mathbf{U}}}^\top - e\,\mu_N \frac{\partial G^N}{\partial \mathbf{U}}^\top + \mathit{\Delta} t\,\boldsymbol{\Phi}^{n-1} + \left(\gamma + \frac{1}{2}\right)\mathit{\Delta} t^2 \boldsymbol{\Psi}^{n-1}$ for $\boldsymbol{\Lambda}^n$
update $\boldsymbol{\Phi}^n = \boldsymbol{\Phi}^{n-1} + c\,\boldsymbol{\Psi}^{n-1} - \mu_{N-n}\frac{\partial G^{N-n}}{\partial \dot{\mathbf{U}}}^\top - \frac{\partial \mathbf{R}^{N-n}}{\partial \dot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^n$
update (option 1) $\boldsymbol{\Psi}^n = \boldsymbol{\Psi}^{n-1} - \mu_{N-n}\frac{\partial G^{N-n}}{\partial \mathbf{U}}^\top - \frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^\top \boldsymbol{\Lambda}^n$
or (option 2) $\boldsymbol{\Psi}^n = -\frac{d}{e}\boldsymbol{\Psi}^{n-1} + \frac{1}{e}\frac{\partial \mathbf{R}^{N-n}}{\partial \ddot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^n - \frac{b}{e}\boldsymbol{\Phi}^n - \frac{a}{e}\boldsymbol{\Phi}^{n-1}$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_{N-n}\frac{\partial G^{N-n}}{\partial d_i} + \boldsymbol{\Lambda}^{n\top}\frac{\partial \mathbf{R}}{\partial d_i}^{N-n}$
end for
solve $\mathbf{K}^{0\top}\boldsymbol{\Lambda}^N = a\,\boldsymbol{\Phi}^{N-1} + d\,\boldsymbol{\Psi}^{N-1}$ for $\boldsymbol{\Lambda}^N$
update $\frac{\mathrm{D}F}{\mathrm{D}d_i} = \frac{\mathrm{D}F}{\mathrm{D}d_i} + \mu_0\frac{\partial G^0}{\partial d_i} + \boldsymbol{\Lambda}^{N\top}\frac{\partial \mathbf{R}^0}{\partial d_i} + \left(\mu_0\frac{\partial G^0}{\partial \mathbf{U}} - \boldsymbol{\Psi}^{N-1\top}\right)\frac{\partial \mathbf{U}^0}{\partial d_i} + \left(\mu_0\frac{\partial G^0}{\partial \dot{\mathbf{U}}} - \boldsymbol{\Phi}^{N-1\top} - c\,\boldsymbol{\Psi}^{N-1\top}\right)\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i} + \boldsymbol{\Lambda}^{N\top}\left(\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \dot{\mathbf{U}}}\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}\right)$

Fig. 16 Adjoint discretize-then-differentiate node for dynamic problem

The first $\boldsymbol{\Psi}^n$ option, is restricted to symmetric $\partial\mathbf{R}/\partial\mathbf{U}$. Whereby (111) and (115) are approximated as

$$\frac{\partial \mathbf{R}^{N-n}}{\partial \mathbf{U}}^\top \boldsymbol{\Lambda}^n \approx \frac{1}{\epsilon}\mathbf{R}\left(\mathbf{U}^{N-n} + \epsilon\boldsymbol{\Lambda}^n, \dot{\mathbf{U}}^{N-n}, \ddot{\mathbf{U}}, \mathbf{d}\right). \tag{120}$$

For the second $\boldsymbol{\Psi}^n$ option, considers the more common restriction for which $\partial\mathbf{R}/\partial\ddot{\mathbf{U}}$ is symmetric and $\beta \neq 0$, whence the terms in (112) and (116) are approximated as

$$\frac{\partial \mathbf{R}^{N-n}}{\partial \ddot{\mathbf{U}}}^\top \boldsymbol{\Lambda}^n \approx \frac{1}{\epsilon}\mathbf{R}\left(\mathbf{U}^{N-n}, \dot{\mathbf{U}}^{N-n}, \ddot{\mathbf{U}} + \epsilon\boldsymbol{\Lambda}^n, \mathbf{d}\right). \tag{121}$$

Finally, to compute $\mathrm{D}F/\mathrm{D}d_i$ in (118), we use the following approximation

$$\frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \mathbf{U}^0}{\partial d_i} + \frac{\partial \mathbf{R}^0}{\partial \mathbf{U}}\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}$$
$$\approx \frac{1}{\epsilon}\mathbf{R}\left(\mathbf{U}^0 + \epsilon\frac{\partial \mathbf{U}^0}{\partial d_i}, \dot{\mathbf{U}}^0 + \epsilon\frac{\partial \dot{\mathbf{U}}^0}{\partial d_i}, \ddot{\mathbf{U}}^0, \mathbf{d}\right). \tag{122}$$

The derivative $\partial\mathbf{R}^n/\partial d_i$ of (62), (66), and (118) is approximated from (106).

4.6 Dynamic example

Consider a two identical masses $m_1 = m_2 = 1$ that are free to slide over a frictionless horizontal surface. The masses are connected by identical nonlinear springs and identical linear dampers as seen in Fig. 17. The internal force generated by the springs is $f_e = x + k_d\,x^3$ where x is the relative displacement of the connected nodes of the spring and the parameter $k_d = 1$ is our design variable. The dampers generate the force $f_c = k_c\,\dot{x}$, where $k_c = 0.1$. There is no external force acting in the two mass-spring-damper system but it is subjected to the initial conditions $x_1(0) = 0$, $x_2(0) = 1$, $\dot{x_1}(0) = 0$ and $\dot{x_2}(0) = 0$. The time domain is $t = [0, 10]$, the Newton-Raphson tolerance is $\epsilon_R < 10^{-15}$ and the Newmark-beta parameters are $\gamma = 1/2$ and $\beta = 1/4$.

To illustrate the various sensitivity analyses, the response function is

$$F = \int_0^{10}\left(x_1^2 + x_2^2 + \dot{x}_1^2 + \dot{x}_2^2\right)\mathrm{d}t\,, \tag{123}$$

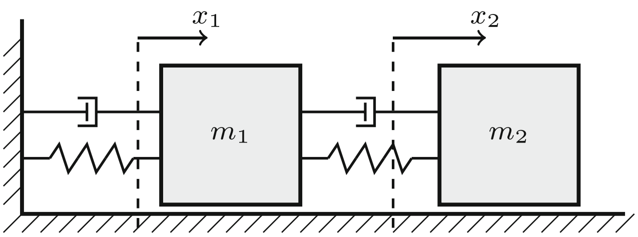

Fig. 17 Two mass-spring-damper system

Table 3 Sensitivities for the two mass-spring-damper problem with $\epsilon = 10^{-6}$.

Method	$N = 100$	$N = 1000$
Direct	0.107639247785	0.105776635574
Semi-a. direct	0.107638856238	0.105776248858
Adj. diff.-then-disc.	0.102804889175	0.105727453140
Semi-a. adj. diff.-then-disc.	0.102805080190	0.105727675445
Adj. disc.-then-diff. opt. 1	0.107639247785	0.105776635574
Semi-a. adj. disc.-then-diff. opt. 1	0.107639248264	0.105776639065
Adj. disc.-then-diff. opt. 2	0.107639247785	0.105776635578
Semi-a. adj. disc.-then-diff. opt. 2	0.107639274468	0.105776920805
Finite differences	0.107639219982	0.105776611026

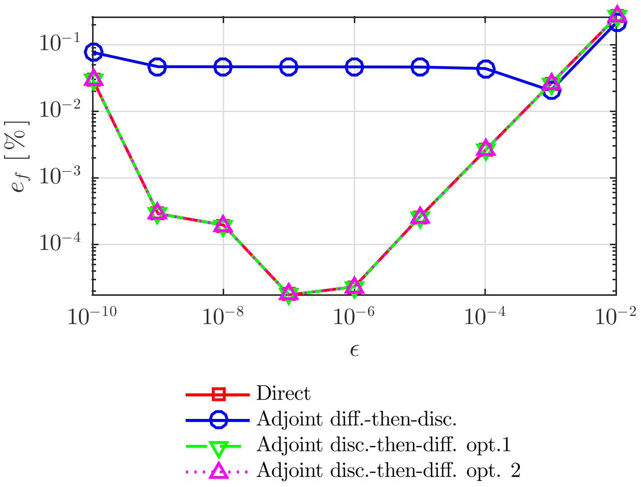

Fig. 18 Relative percentage error of the sensitivities obtained by the analytical methods with respect to finite differences for the mass-spring-damper problem $N = 1000$

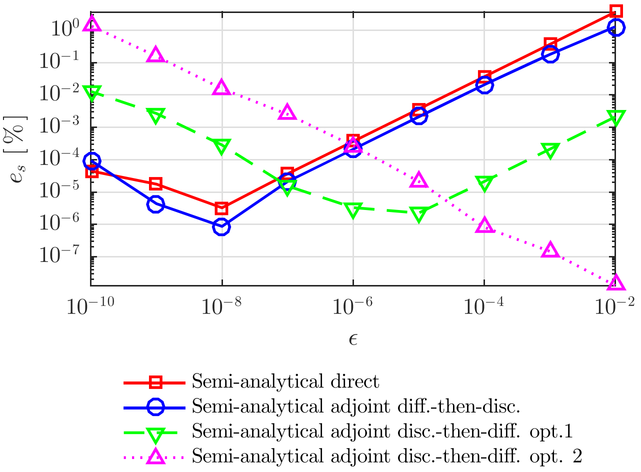

Fig. 20 Relative percentage error of the semi-analytical sensitivities for the mass-spring-damper problem for $N = 1000$

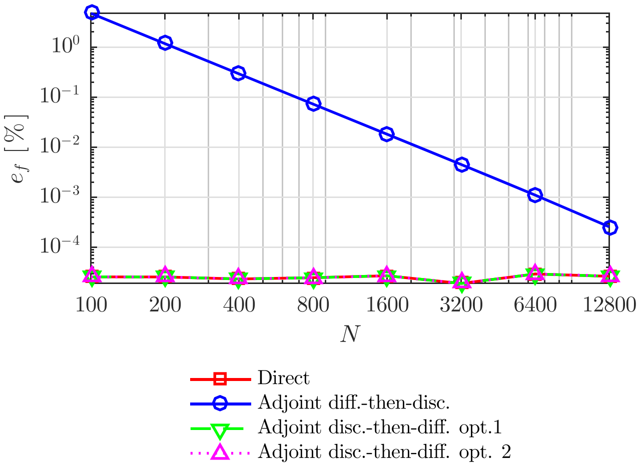

Fig. 19 Relative percentage error of the sensitivities obtained by the analytical methods with respect to finite differences for the mass-spring-damper problem $\epsilon = 10^{-6}$

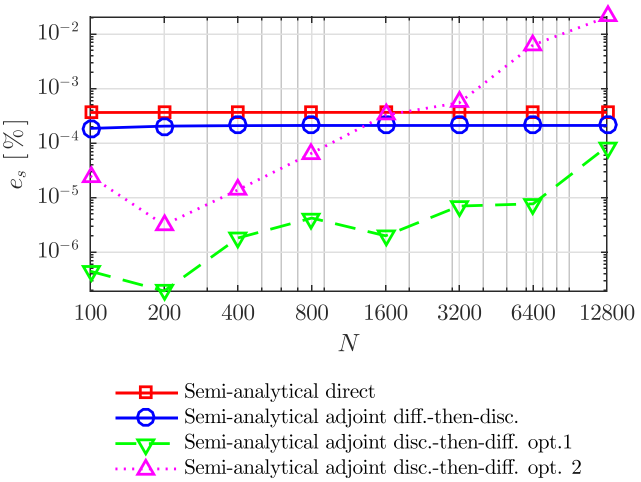

Fig. 21 Relative percentage error of the semi-analytical sensitivities for the mass-spring-damper problem for $\epsilon = 10^{-6}$

Table 4 Restrictions for semi-analytical adjoint methods for transient problems

Method	Symmetry	Additional restrictions
Semi-a. adj. diff.-then-disc.	$\partial \mathbf{R}^n/\partial \mathbf{U}$	$\frac{d}{dt}\left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}\right) = \mathbf{0}$
Semi-a. adj. disc.-then-diff. opt. 1	$\partial \mathbf{R}^n/\partial \mathbf{U}$	
Semi-a. adj. disc.-then-diff. opt. 2	$\partial \mathbf{R}^n/\partial \dot{\mathbf{U}}$	$\alpha \neq 0$

where the numerical integration is done by the trapezoidal rule. Table 3 shows the computed sensitivities values for the different methods using the perturbation size $\epsilon = 10^{-6}$. The response function converges as the number of time steps increases, thus the values of the sensitivities corresponding to $N = 100$ differ from those corresponding to $N = 1000$. For $N = 100$, the sensitivities obtained by the adjoint method differentiate-then-discretize, do not coincide with the others due to the consistency error (Gunzburger 2003; Jensen et al. 2014). However, this consistency error practically vanishes for $N = 1000$.

To examine the consistency of the methods, we show e_f for the $N = 1000$ case and different perturbation sizes, cf. Figure 18. As expected the finite differences show truncation and round off error for large and small perturbations respectively, and the adjoint differentiate-then-discretize method shows a consistency error. Figure 19 illustrates the error e_f for $\epsilon = 10^{-6}$ and different time steps, where it is seen that the consistency error of the adjoint differentiate-then-discretize method reduces as the number of time steps increases.

To examine the accuracy of the semi-analytical sensitivities, we compute the error e_s for the $N = 1000$ case, cf. Fig. 20. Again, as expected, the semi-analytical sensitivities exhibit truncation and round off error for small and large perturbation sizes respectively.

Figure 21 shows that the error e_s for $\epsilon = 10^{-6}$ is fairly independent of the time step size.

Table 5 Restrictions for semi-analytical adjoint methods for dynamic problems

Method	Symmetry	Additional restrictions
Semi-a. adj. diff.-then-disc.	$\partial \mathbf{R}^n/\partial \dot{\mathbf{U}}$, $\partial \mathbf{R}^n/\partial \mathbf{U}$	$\frac{d}{dt}\left(\frac{\partial \mathbf{R}}{\partial \dot{\mathbf{U}}}\right) = \mathbf{0}$, $\frac{d}{dt}\left(\frac{\partial \mathbf{R}}{\partial \ddot{\mathbf{U}}}\right) = \mathbf{0}$
Semi-a. adj. disc.-then-diff. opt. 1	$\partial \mathbf{R}^n/\partial \dot{\mathbf{U}}$, $\partial \mathbf{R}^n/\partial \mathbf{U}$	
Semi-a. adj. disc.-then-diff. opt. 2	$\partial \mathbf{R}^n/\partial \dot{\mathbf{U}}$, $\partial \mathbf{R}^n/\partial \ddot{\mathbf{U}}$	$\beta \neq 0$

5 Conclusions

Implementation of analytical sensitivity analyses requires detailed knowledge of the analysis program and can be error-prone and time-consuming to implement. Fortunately, these drawbacks may be reduced by adopting the semi-analytical method, where terms in the pseudo or adjoint loads and also in the sensitivities are approximated by finite differences. In this way, we are able to compute these complicated terms using subroutines that are used for the solution of the primal problem and maintain the efficiency of the analytical methods. That said, the accuracy of the semi-analytical sensitivities is susceptible to truncation, round-off errors, and additional errors if the convergence tolerance of the primal analysis is not sufficiently small.

In transient and dynamic problems, the semi-analytical sensitivity analysis approach affects both restrictive assumptions and accuracy. In particular, expressions for the adjoint differentiate-then-discretize and discretize-then-differentiate approaches differ because the differentiation and discretization steps do not commute. The differentiate-then-discretize approach requires some terms to be constant, e.g., mass matrix, in order to reuse the tangent stiffness matrix from the primal analysis; however, the first and last tangent stiffness matrices are not reused. This is not the case for the direct and the adjoint discretize-then-differentiate methods where the tangent stiffness matrix is reused for all time steps. Furthermore, the adjoint differentiate-then-discretize approach yields consistency error, albeit they reduce with the time step size.

In most cases, the semi-analytical adjoint approaches for the nonlinear transient and nonlinear dynamic systems require symmetry of $\partial \mathbf{R}^n/\partial \mathbf{U}$, $\partial \mathbf{R}^n/\partial \dot{\mathbf{U}}$, and/or $\partial \mathbf{R}^n/\partial \ddot{\mathbf{U}}$. This may be problematic, as $\partial \mathbf{R}^n/\partial \mathbf{U}$ is usually asymmetric in nonlinear problems. Fortunately, if we do not use an explicit method, the semi-analytical discretize-then-differentiate adjoint method can accommodate asymmetric $\partial \mathbf{R}^n/\partial \mathbf{U}$. A summary of these restrictions is presented in Tables 4 and 5. Example problems are provided to show the efficiency and errors associated with the various methods for nonlinear transient and nonlinear dynamic problems.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

Adelman HM, Haftka RT (1986) Sensitivity analysis of discrete structural systems. AIAA J 24(5):823–832

Barthelemy B, Haftka R (1990) Accuracy analysis of the semi-analytical method for shape sensitivity calculation. Mech Struct Mach 18(3):407–432

Barthelemy B, Chon C, Haftka R (1988) Accuracy problems associted with semi-analytical derivatives of static response. Finite Elem Anal Des 4(3):249–265

Bernard J, Kwon S, Wilson J (1993) Differentiation of mass and stiffness matrices for high order sensitivity calculations in finite element-based equilibrium problems. J Mech Des 115(4):829–832

Bestle D, Seybold J (1992) Sensitivity analysis of constrained multibody systems. Arch Appl Mech 62(3):181–190

de Boer H, van Keulen F (2000) Refined semi-analytical design sensitivities. Int J Solids Struct 37(46-47):6961–6980

de Boer H, van Keulen F, Vervenne K (2002) Refined second order semi-analytical design sensitivities. Int J Numer Methods Eng 55(9):1033–1051

Botkin M (1982) Shape optimization of plate and shell structures. AIAA J 20(2):268–273

Brüls O, Eberhard P (2008) Sensitivity analysis for dynamic mechanical systems with finite rotations. Int J Numer Methods Eng 74(13):1897–1927

Camarda C, Adelman H (1984) Static and dynamic structural-sensitivity derivative calculations in the finite-element-based engineering analysis language (eal) system. NASA TM-85743

Chen B, Gu Y, Zhang H, Zhao G (2003) Structural design optimization on thermally induced vibration. Int J Numer Methods Eng 58(8):1187–1212

Cheng G, Liu Y (1987) A new computation scheme for sensitivity analysis. Eng Optim 12(3):219–234

Cheng G, Olhoff N (1993) Rigid body motion test against error in semi-analytical sensitivity analysis. Comput Struct 46(3):515–527

Cheng G, Gu Y, Zhou Y (1989) Accuracy of semi-analytic sensitivity analysis. Finite Elem Anal Des 6(2):113–128

Deng Y, Liu Z, Zhang P, Liu Y, Wu Y (2011) Topology optimization of unsteady incompressible navier–stokes flows. J Comput Phys 230(17):6688–6708

Esping B (1984) Minimum weight design of membrane structures using eight node isoparametric elements and numerical derivatives. Comput Struct 19(4):591–604

Fenyes P, Lust R (1991) Error analysis of semianalytic displacement derivatives for shape and sizing variables. Amer Inst Aeronaut Astronaut 29(2):271–279

Gallagher R, Zienkiewicz O (1973) Optimum structural design: Theory and applications. Wiley, New York

Greene W, Haftka R (1991) Computational aspects of sensitivity calculations in linear transient structural analysis. Struct Optim 3(3):176–201

Gu Y, Grandhi R (1998) Sensitivity analysis and optimization of heat transfer and thermal-structural designs. In: 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, pp 4746

Gu Y, Chen B, Zhang H, Grandhi R (2002) A sensitivity analysis method for linear and nonlinear transient heat conduction with precise time integration. Struct Multidiscip Optim 24(1):23–37

Gunzburger MD (2003) Perspectives in flow control and optimization. Advances in Design and Control, Society for Industrial and Applied Mathematics

Haftka R (1993) Semi-analytical static nonlinear structural sensitivity analysis. AIAA J 31(7):1307–1312

Haftka R, Adelman H (1989) Recent developments in structural sensitivity analysis. Struct Optim 1(3):137–151

Haftka RT, Gürdal Z (2012) Elements of structural optimization, vol 11. Springer Science & Business Media, Berlin

Haug EJ (1987) Design sensitivity analysis of dynamic systems. In: Computer aided optimal design: Structural and Mechanical Systems. Springer, Berlin, pp 705–755

Hooijkamp EC, van Keulen F (2018) Topology optimization for linear thermo-mechanical transient problems: Modal reduction and adjoint sensitivities. Int J Numer Methods Eng 113(8):1230–1257

Jensen J, Nakshatrala P, Tortorelli D (2014) On the consistency of adjoint sensitivity analysis for structural optimization of linear dynamic problems. Struct Multidiscip Optim 49(5):831–837

van Keulen F, Haftka R, Kim R (2005) Review of options for structural design sensitivity analysis. part 1: Linear systems. Comput Methods Appl Mech Eng 194(30-33):3213–3243. Structural and Design Optimization

Kiendl J, Schmidt R, Wüchner R, Bletzinger KU (2014) Isogeometric shape optimization of shells using semi-analytical sensitivity analysis and sensitivity weighting. Comput Methods Appl Mech Eng 274:148–167

Kramer JL, Stockman NO (1963) Effect of variable thermal properties on one-dimensional heat transfer in radiating fins. Technical report, NASA TN D-1878

Kreissl S, Pingen G, Maute K (2011) Topology optimization for unsteady flow. Int J Numer Methods Eng 87(13):1229–1253

Meric R (1988) Shape design sensitivity analysis of dynamic structures. AIAA J 26(2):206–212

Michaleris P, Tortorelli DA, Vidal CA (1994) Tangent operators and design sensitivity formulations for transient non-linear coupled problems with applications to elastoplasticity. Int J Numer Methods Eng 37(14):2471–2499

Mlejnek H (1992) Accuracy of semi-analytical sensitivities and its improvement by the natural method. Struct Optim 4(2):128–131

Mróz Z, Haftka R (1994) Design sensitivity analysis of non-linear structures in regular and critical states. Int J Solids Struct 31(15):2071–2098

Olhoff N, Rasmussen J (1991) Study of inaccuracy in semi-analytical sensitivity analysis a model problem. Struct Optim 3(4):203–213

Olhoff N, Rasmussen J, Lund E (1993) A method of exact numerical differentiation for error elimination in finite-element-based semi-analytical shape sensitivity analyses. Mech Struct Mach 21(1):1–66

Oral S (1996) An improved semianalytical method for sensitivity analysis. Struct Optim 11(1-2):67–69

Pedersen P, Cheng G, Rasmussen J (1989) On accuracy problems for semi-analytical sensitivity analyses. Mech Struct Mach 17(3):373–384

Ray D, Pister KS, Polak E (1978) Sensitivity analysis for hysteretic dynamic systems: theory and applications. Comput Methods Appl Mech Eng 14(2):179–208

Tortorelli D, Michaleris P (1994) Design sensitivity analysis: Overview and review. Inverse Probl Eng 1(1):71–105

Tortorelli DA, Haber RB, Lu SCY (1991) Adjoint sensitivity analysis for nonlinear dynamic thermoelastic systems. AIAA J 29(2):253–263

Tromme E, Tortorelli D, Brüls O, Duysinx P (2015) Structural optimization of multibody system components described using level set techniques. Struct Multidiscip Optim 52(5):959–971

Van Keulen F, De Boer H (1998) Rigorous improvement of semi-analytical design sensitivities by exact differentiation of rigid body motions. Int J Numer Methods Eng 42(1):71–91

Wang W, Clausen PM, Bletzinger KU (2015) Improved semi-analytical sensitivity analysis using a secant stiffness matrix for geometric nonlinear shape optimization. Comput Struct 146:143–151

Zhong WX, Williams FW (1994) A precise time step integration method. Proc Inst Mech Eng Part C: J Mech Eng Sci 208(6):427–430