

Article

Analytical Sensitivity Analysis of Dynamic Problems with Direct Differentiation of Generalized- α Time Integration

Erich Wehrle ^{1,2,*}  and Veit Gufler ² 

¹ Collins Aerospace, Applied Research and Technology, Multidisciplinary Design Optimization Research Group, 80805 Munich, Germany

² Faculty of Engineering, Free University of Bozen-Bolzano, Universitätsplatz 1, 39100 Bozen, South Tyrol, Italy; veit.gufler@unibz.it

* Correspondence: erich.wehrle@collins.com

Abstract: In this paper, the direct differentiation of generalized- α time integration is derived, equations are introduced and results are shown. Although generalized- α time integration has found usage, the derivation and the resulting equations for the analytical sensitivity analysis via direct differentiation are missing. Thus, here, the sensitivity equations of generalized- α time integration via direct differentiation are provided. Results with generalized- α are compared with Newmark- β time integration and their sensitivities with numerical sensitivities via forward finite differencing in terms of accuracy and performance. An example is shown for each linear structural dynamics and flexible multibody dynamics.

Keywords: sensitivity analysis; time integration; structural dynamics; flexible multibody dynamics



Citation: Wehrle, E.; Gufler, V. Analytical Sensitivity Analysis of Dynamic Problems with Direct Differentiation of Generalized- α Time Integration. *Machines* **2024**, *12*, 128. <https://doi.org/10.3390/machines12020128>

Academic Editors: Jian Wu, Xiangkun He and Guangfei Xu

Received: 4 December 2023

Revised: 3 February 2024

Accepted: 5 February 2024

Published: 12 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Analytical design sensitivity analysis is developed in this paper for the generalized- α family of time integration. After deriving and introducing design sensitivity using direct differentiation, results are compared with numerical differentiation via forward differencing in terms of accuracy and computational effort.

Efficient sensitivity analysis is the centerpiece of efficient and effective gradient-based design optimization. Other uses of sensitivity analysis include uncertainty analysis and quantification, e.g., the adjoint variable method applied to the sensitivity analysis with respect to uncertain and random variables was developed in [1]. Overviews of design sensitivity analysis are provided by [2–6]. In transient problems, sensitivity analysis must be extended to time integration [7,8], which is handled here by introducing and deriving the equations for the direct differentiation of the generalized- α family of time integration.

In the dynamic systems addressed, the transient response (response in time domain) is of interest and therefore a temporal discretization is applied. The calculation of the system responses, here position, velocity and acceleration, can be carried out with explicit or implicit time integration methods. Implicit time integration allows for larger time steps than explicit methods, which is bounded by the Courant criterion to remain stable [9].

Newmark introduced an implicit integration scheme, now referred to as Newmark- β , in 1959 with two integration parameters γ and β [10]. The Hilber–Hughes–Taylor (HHT) method was introduced in 1977 to improve upon Newmark with the additional integration parameter α_f for numerical damping [11], while the Wood–Bossak–Zienkiewicz (WBZ, also known as simply Bossak or Wood) introduced the parameter α_m [12]. Generalized- α was introduced by Chung and Hulbert in 1993 [13] to generalize all these methods with the integration parameters α_m , α_f , β and γ . These methods can be generalized via a single implementation which is therefore referred to as the generalized- α family of time integrators. In the generalized- α method, for which the family is named, all parameters can be controlled by a single value, namely the spectral radius at infinity ρ_∞ ranging from zero,

being the highest amount of damping, to unity, being none [13]. It should be noted that we differentiate between the variants of the Greek letter rho: ϱ is used here for the spectral radius at infinity, while ρ is used below for density. The mapping of ϱ_∞ to α_m , α_f , β and γ is shown below. The choice of this value can affect the response for a given time step, see, e.g., [14].

Generalized- α time integration has found usage in both nonlinear structural dynamics and multibody dynamics. Multibody dynamics addresses the dynamic response with kinematic constraints. Rigid multibody dynamics considers the bodies rigid, while flexible multibody dynamics allows for the deformation in the bodies of the system. Applications are shown including both linear structural dynamics and flexible multibody dynamics.

This paper is concerned with the sensitivity analysis of dynamic problems, which is used when one or more parameters affect the system responses. The calculation of the system responses is referred to here as primal analysis so that it may be distinguished from the sensitivity analysis. This is of interest for the efficient design optimization via gradient-based optimization algorithms in addition to investigating the sensitivities themselves or for uncertainty analyses. Methods for sensitivity analysis include analytical, numerical, complex step and algorithmic approaches [5].

Analytical sensitivities can be derived either manually (by hand) or via the automatic differentiation [15,16] of the implemented code. The automatic differentiation and manual differentiation of rigid multibody systems are compared in [17]. Although prone to human error and not automatic, the former represents an exact and one of the fastest methods for sensitivity analysis [18]. Manual sensitivity analysis requires high implementation effort, but the resulting code is typically highly reusable and must therefore only be derived and implemented once. Due to its physics- and knowledge-based nature, in addition to its increased transparency, hand differentiation is preferred by the authors.

This study limits itself to analytical sensitivity analysis, which is further divided into direct differentiation (also known as forward sensitivity analysis) and the adjoint variable method (also known as backward sensitivity analysis). Generally, it can be said that direct differentiation is more efficient than the adjoint variable method if the number of design variables is less than the number of functions to differentiate, i.e., in optimization, the sum of the number of objective functions (one in case of single objective optimization) and the number of constraints. Due to its nature, the adjoint variable method must be performed after the time integration of the primal analysis and integrated backwards in time. Thus, the adjoint variable method, in contrast to direct differentiation, needs dedicated solver routines to integrate backwards in time and cannot reuse primal forward time integration [7,8]. Furthermore, for nonlinear analysis, the system matrices must be stored or recalculated when integrating backwards in time. Therefore, direct differentiation is chosen in this work. The reader is referred to the following literature for the adjoint variable method: [19–30].

A number of studies have been carried out for sensitivity analysis in transient structural dynamics and multibody dynamics. Early works on sensitivity analysis with transient structural dynamics include [31], which bridges sensitivities in static and dynamic analysis via the adjoint variable method, including maximum-value objectives and constraints. In [32], sensitivity analysis with the adjoint variable method and direct differentiation is introduced for time-averaged and time-specific constraints. The German-language monograph [33] clearly demonstrates the analytic sensitivity analysis of transient linear dynamics with Newmark- β time integration, providing all necessary equations for implementation, which also includes the sensitivity of the duration of a dynamic operation. The adjoint variable method for transient structural dynamics with Newmark- β time integration is derived by [34]. The adjoint variable method and direct differentiation of nonlinear and transient problems is handled by [35] in which the crucial aspects of the nonlinear solver for sensitivity analysis are introduced.

The sensitivity analysis of multibody systems is often limited to the governing equations and often not extended to the nonlinear solver or time integration. Exceptions to

this can be found for direct differentiation with the divide-and-conquer-based modal superposition method in [36]; for direct differentiation and the adjoint variable method with general formulations of differential–algebraic equations in [37]; for direct differentiation and the adjoint variable method of ordered differential equations in [38]; and for the adjoint variable method with the floating frame of reference formulation in [20]. The adjoint variable method of the governing equations in descriptor form and for generic optimization functions is shown by [21] for several time integration methods, including the backward difference formula, Newmark-, Adams–Bashforth–Moulton and diagonally implicit Runge–Kutta. The derivation of the adjoint variable method based on generalized- α time integration and applied to the finite-element-based formulation is provided by [39]. The transient response sensitivity analysis of localized nonlinear behavior using Newmark- β [40].

Higher-order sensitivities can also be of interest, especially in cases of a low number of degrees of freedom and design variables. Second-order sensitivities via the adjoint variable method with the Runge–Kutta–Fehlberg method are shown in [41]. Second-order sensitivities for multibody systems with the adjoint variable method and differential–algebraic equations are shown in [42]. Higher-order sensitivities are used in [43,44] to develop a surrogate model for optimization. This work is limited to first-order sensitivities, as required by gradient-based optimization, although the equations provided can be readily extended to higher-order derivatives.

The scope here is thus the investigation of the direct differentiation of the generalized- α method. A method in this regard is introduced in [45], yet the needed equations are not derived and shown. The present work builds upon and extends the works to generalized- α . The authors developed Newmark- β sensitivities in [7,46], which are in turn based on [33,34]. In this paper, Newmark- β serves as a comparison to the developed method for generalized- α .

This work first generally introduces generalized- α time integration and its equations. Afterwards, the sensitivity equations are introduced for structural dynamics and flexible multibody dynamics, respectively. A numerical example is shown for each and the results are provided for both the primal and sensitivity analyses in comparison with Newmark- β time integration. We close with a conclusion, including a discussion of this work. Furthermore, a detailed derivation of the effective system matrices is provided in the Appendices A and B.

2. Sensitivity Analysis with Generalized- α Time Integration

In this section, we introduce the equations for an acceleration-form (a -form) of generalized- α time integration using a predictor–corrector scheme and extend this for sensitivity analysis. The acceleration-form refers to the formulation, where the acceleration is the primary variable (i.e., the variable for which it is being solved), and contrasts with the displacement-form (d -form) in which the displacement is the primary variable. The predictor–corrector scheme allows for nonlinear terms in the governing equation in which the predicted values are updated in the nonlinear solver, e.g., Newton–Raphson, and then corrected.

In this work, we extend the well-established generalized- α time integration method to include design sensitivity analysis, which is defined by the differentiation with respect to the design variables (but could include other parameters, e.g., uncertain parameters). Sensitivity analysis is carried out here by directly differentiating both the equations of the time integration as well as the governing equations. We consider both linear structural dynamics as well as flexible multibody dynamics, which is intrinsically nonlinear. After introducing the general equations for generalized- α time integration and its sensitivities, the domain-specific governing equations for linear structural dynamics and flexible multibody dynamics are given. Flexible multibody dynamics is shown using the specific approach developed in [7,47,48] with an index-1 approach and the floating frame of reference formulation.

2.1. Generalized- α Time Integration

The method for primal and sensitivity analysis with generalized- α time integration is introduced in the following. This is based on Newmark’s equations for the updated position \underline{q} and velocity vectors $\underline{\dot{q}}$ with the calculated value of the acceleration vector $\underline{\ddot{q}}$,

$$\underline{q}_{n+1} = \underbrace{\underline{q}_n + \Delta t \underline{\dot{q}}_n + \left(\frac{1}{2} - \beta\right) \Delta t^2 \underline{\ddot{q}}_n}_{\text{predictor } \underline{q}_{\text{pred}}} + \underbrace{\beta \Delta t^2 \underline{\ddot{q}}_{n+1}}_{\text{corrector}} \tag{1}$$

$$\underline{\dot{q}}_{n+1} = \underbrace{\underline{\dot{q}}_n + (1 - \gamma) \Delta t \underline{\ddot{q}}_n}_{\text{predictor } \underline{\dot{q}}_{\text{pred}}} + \underbrace{\gamma \Delta t \underline{\ddot{q}}_{n+1}}_{\text{corrector}} \tag{2}$$

where γ and β are the Newmark integration parameters. Though the simulation is evaluated at the times t_n and t_{n+1} , values at intermediate points are interpolated. The position q and velocity \dot{q} use time $t_{n+1-\alpha_f}$ and the acceleration \ddot{q} uses time $t_{n+1-\alpha_m}$, giving the approximations by linear interpolation (see Figure 1),

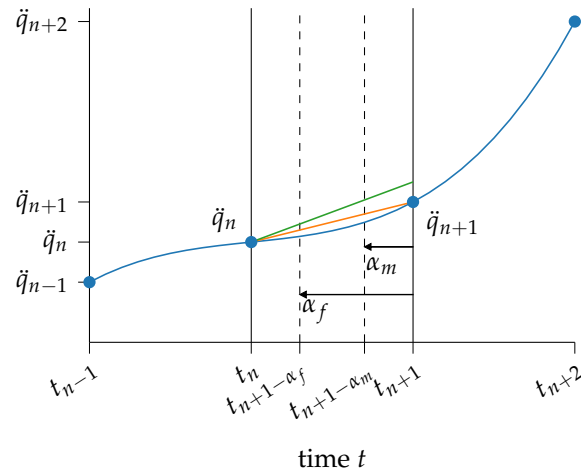


Figure 1. Time integration with — acceleration, — $\gamma = 0.5$, — $\gamma = 0.75$.

$$\underline{q}_{n+1-\alpha_f} = (1 - \alpha_f) \underline{q}_{n+1} + \alpha_f \underline{q}_n \tag{3}$$

$$\underline{\dot{q}}_{n+1-\alpha_f} = (1 - \alpha_f) \underline{\dot{q}}_{n+1} + \alpha_f \underline{\dot{q}}_n \tag{4}$$

$$\underline{\ddot{q}}_{n+1-\alpha_m} = (1 - \alpha_m) \underline{\ddot{q}}_{n+1} + \alpha_m \underline{\ddot{q}}_n \tag{5}$$

where α_m and α_f are integration parameters. A predictor–corrector scheme is used in order to accommodate the nonlinear calculation and state-dependency of the system parameters, i.e., the system matrices with respect to position and velocity. The predicted state variables with Newmark’s equations are

$$\underline{q}_{\text{pred}} = \underline{q}_n + \Delta t \underline{\dot{q}}_n + \left(\frac{1}{2} - \beta\right) \Delta t^2 \underline{\ddot{q}}_n \tag{6}$$

$$\underline{\dot{q}}_{\text{pred}} = \underline{\dot{q}}_n + (1 - \gamma) \Delta t \underline{\ddot{q}}_n \tag{7}$$

and we calculate $\underline{\dot{q}}_{n+1}$ using the effective system equations (see below for an explanation of structural dynamics and flexible multibody dynamics, respectively). The predicted values are then corrected with Newmark's equations,

$$\underline{q}_{n+1} = \underline{q}_{\text{pred}} + \beta \Delta t^2 \underline{\ddot{q}}_{n+1}, \quad (8)$$

$$\underline{\dot{q}}_{n+1} = \underline{\dot{q}}_{\text{pred}} + \gamma \Delta t \underline{\ddot{q}}_{n+1}. \quad (9)$$

For the sensitivity analysis within the generalized- α time integration, we use the direct differentiation of the approximations

$$\underline{\underline{\nabla}} \underline{q}_{n+1-\alpha_f} = (1 - \alpha_f) \underline{\underline{\nabla}} \underline{q}_{n+1} + \alpha_f \underline{\underline{\nabla}} \underline{q}_n, \quad (10)$$

$$\underline{\underline{\nabla}} \underline{\dot{q}}_{n+1-\alpha_f} = (1 - \alpha_f) \underline{\underline{\nabla}} \underline{\dot{q}}_{n+1} + \alpha_f \underline{\underline{\nabla}} \underline{\dot{q}}_n, \quad (11)$$

$$\underline{\underline{\nabla}} \underline{\ddot{q}}_{n+1-\alpha_m} = (1 - \alpha_m) \underline{\underline{\nabla}} \underline{\ddot{q}}_{n+1} + \alpha_m \underline{\underline{\nabla}} \underline{\ddot{q}}_n, \quad (12)$$

where the nabla operator ∇ denotes the design sensitivity, i.e., the total derivative with respect to the design variables, e.g., for the sensitivity scalar response with respect to the design variable vector,

$$\underline{\underline{\nabla}}(\cdot) = \frac{d(\cdot)}{dx}. \quad (13)$$

The same direct differentiation is applied to the predictors, which are then defined by

$$\underline{\underline{\nabla}} \underline{q}_{\text{pred}} = \underline{\underline{\nabla}} \underline{q}_n + \Delta t \underline{\underline{\nabla}} \underline{\dot{q}}_n + \left(\frac{1}{2} - \beta\right) \Delta t^2 \underline{\underline{\nabla}} \underline{\ddot{q}}_n, \quad (14)$$

$$\underline{\underline{\nabla}} \underline{\dot{q}}_{\text{pred}} = \underline{\underline{\nabla}} \underline{\dot{q}}_n + (1 - \gamma) \Delta t \underline{\underline{\nabla}} \underline{\ddot{q}}_n. \quad (15)$$

After calculating $\underline{\underline{\nabla}} \underline{\dot{q}}_{n+1}$ by use of the effective sensitivity system (see below for explanation of structural dynamics and flexible multibody dynamics, respectively), the following correctors are used:

$$\underline{\underline{\nabla}} \underline{q}_{n+1} = \underline{\underline{\nabla}} \underline{q}_{\text{pred}} + \beta \Delta t^2 \underline{\underline{\nabla}} \underline{\ddot{q}}_{n+1}, \quad (16)$$

$$\underline{\underline{\nabla}} \underline{\dot{q}}_{n+1} = \underline{\underline{\nabla}} \underline{\dot{q}}_{\text{pred}} + \gamma \Delta t \underline{\underline{\nabla}} \underline{\ddot{q}}_{n+1}. \quad (17)$$

For a nonlinear analysis, a Newton solver can be integrated into the solving routine. This is explained with flexible multibody systems below in Section 2.3.

The time integration constants of generalized- α are α_m , α_f , β and γ and their choice can lead to a number of established time integration methods, including the Newmark family of integrators, Hilber–Hughes–Taylor and Wood–Bossak–Zienkiewicz. Therefore, this can then be referred to as the α -family of time integrators. The equations introduced above in this section can therefore be generally used within this family. Specifically for the generalized- α time integration used here, the choice of integration constants can be tied to a single user-defined parameter, and the spectral radius at infinity ρ_∞ [13],

$$\alpha_m = \frac{2\rho_\infty - 1}{\rho_\infty + 1}, \quad (18)$$

$$\alpha_f = \frac{\rho_\infty}{\rho_\infty + 1}, \quad (19)$$

$$\gamma = \frac{1}{2} - \alpha_m + \alpha_f, \quad (20)$$

$$\beta = \frac{1}{4} \left(1 - \alpha_m + \alpha_f\right)^2. \quad (21)$$

The spectral radius at infinity is sometimes referred to as the (numerical) damping and is limited to a range of $[0, 1]$. The value $\varrho_\infty = 1$ returns the Newmark- β retaining frequencies, which are preserved, while $\varrho_\infty = 0$ annihilates all frequencies above $\frac{\Delta t}{T}$ after one time step in which T is the period of the highest frequency of interest. The results converge to those without numerical damping when the time step goes to zero. Thus, the level of numerical damping introduced to generalized- α time integration depends on both the step size as well as the spectral radius at infinity and the choice is critical for proper results. A parameter study with a fixed-time step and varying the values of the spectral radius at infinity is shown in [14]. As the present study develops a general framework for this methodology, this is not addressed here and the reader is referred to [13,49–51]. In this paper, we use a moderate value of $\varrho_\infty = 0.55$, though the sensitivity analysis methodology developed is general and independent of this parameter value.

2.2. Structural Dynamics

The governing equation for linear structural dynamics is defined by the following second-order equation of motion:

$$\underline{m}\ddot{\underline{q}}(t) + \underline{d}\dot{\underline{q}}(t) + \underline{k}\underline{q}(t) - \underline{F}(t) = \underline{0}, \quad (22)$$

where \underline{m} is the mass matrix, \underline{d} is the damping matrix, \underline{k} is the stiffness matrix and \underline{F} is the external force vector. For linear structural dynamics, the system matrices are constant and independent of position and time. For the solving scheme, we add the residual force vector $\underline{R}(t)$, which is to be zero (to an allowable value) at all times, to the right-hand side. The α -family of integration methods uses intermediate-point approximations between t_n and t_{n+1} , namely $t_{n+1-\alpha_f}$ and $t_{n+1-\alpha_m}$, revealing

$$\underline{R}_{n+1-\alpha_f} = \underline{m}\ddot{\underline{q}}_{n+1-\alpha_m} + \underline{d}\dot{\underline{q}}_{n+1-\alpha_f} + \underline{k}\underline{q}_{n+1-\alpha_f} - \underline{F}_{n+1-\alpha_f}, \quad (23)$$

where the linear approximation of the intermediate external force is

$$\underline{F}_{n+1-\alpha_f} = (1 - \alpha_f)\underline{F}_{n+1} + \alpha_f\underline{F}_n. \quad (24)$$

The effective equation of motion is defined by

$$\underline{m}_{\text{eff}}\ddot{\underline{q}}_{n+1} = \underline{F}_{\text{eff}}, \quad (25)$$

where

$$\underline{m}_{\text{eff}} = (1 - \alpha_m)\underline{m} + (1 - \alpha_f)\gamma\Delta t\underline{d} + (1 - \alpha_f)\beta\Delta t^2\underline{k}, \quad (26)$$

$$\begin{aligned} \underline{F}_{\text{eff}} = & (1 - \alpha_f)\underline{F}_{n+1} + \alpha_f\underline{F}_n + \\ & - \alpha_m\underline{m}\ddot{\underline{q}}_n - \underline{d}\left((1 - \alpha_f)\dot{\underline{q}}_{\text{pred}} + \alpha_f\dot{\underline{q}}_n\right) - \underline{k}\left((1 - \alpha_f)\underline{q}_{\text{pred}} + \alpha_f\underline{q}_n\right). \end{aligned} \quad (27)$$

For the derivation of the effective mass matrix $\underline{m}_{\text{eff}}$ and effective force vector $\underline{F}_{\text{eff}}$ for generalized- α method, see Appendix A.1.

The sensitivity analysis is derived using the direct differentiation which is applied to the governing Equation (23). This reveals the sensitivity of the force balance for the α -family of integration methods as

$$\begin{aligned} \underline{\nabla}\underline{R}_{n+1-\alpha_f} = & \underline{\nabla}\underline{m}\ddot{\underline{q}}_{n+1-\alpha_m} + \underline{m}\underline{\nabla}\ddot{\underline{q}}_{n+1-\alpha_m} + \underline{\nabla}\underline{d}\dot{\underline{q}}_{n+1-\alpha_f} + \underline{d}\underline{\nabla}\dot{\underline{q}}_{n+1-\alpha_f} + \\ & + \underline{\nabla}\underline{k}\underline{q}_{n+1-\alpha_f} + \underline{k}\underline{\nabla}\underline{q}_{n+1-\alpha_f} - \underline{\nabla}\underline{F}_{n+1-\alpha_f} \end{aligned} \quad (28)$$

where the intermediate external force sensitivity is found with the approximation

$$\underline{\nabla F}_{n+1-\alpha_f} = (1 - \alpha_f) \underline{\nabla F}_{n+1} + \alpha_f \underline{\nabla F}_n. \tag{29}$$

The design sensitivities of the system matrices are three-dimensional matrices and therefore efficient handling is needed.

The effective equation of motion for sensitivities is defined by the differentiation of (25),

$$\underline{m}_{\text{eff}} \underline{\nabla \ddot{q}}_{n+1} = \underline{F}_{\text{pseudo}}, \tag{30}$$

where

$$\underline{F}_{\text{pseudo}} = \underline{\nabla F}_{\text{eff}} - \underline{\nabla \underline{m}}_{\text{eff}} \underline{\ddot{q}}_{n+1}, \tag{31}$$

and fully expressed by

$$\begin{aligned} \underline{F}_{\text{pseudo}} = & (1 - \alpha_f) \underline{\nabla F}_{n+1} + \alpha_f \underline{\nabla F}_n - \underline{\nabla \underline{m}} \left((1 - \alpha_m) \underline{\ddot{q}}_{n+1} + \alpha_m \underline{\ddot{q}}_n \right) - \alpha_m \underline{m} \underline{\nabla \ddot{q}}_n + \\ & - \underline{\nabla \underline{d}} \left((1 - \alpha_f) \underline{\dot{q}}_{n+1} + \alpha_f \underline{\dot{q}}_n \right) - \underline{d} \left((1 - \alpha_f) \underline{\nabla \dot{q}}_{\text{pred}} + \alpha_f \underline{\nabla \dot{q}}_n \right) + \\ & - \underline{\nabla \underline{k}} \left((1 - \alpha_f) \underline{q}_{n+1} + \alpha_f \underline{q}_n \right) - \underline{k} \left((1 - \alpha_f) \underline{\nabla q}_{\text{pred}} + \alpha_f \underline{\nabla q}_n \right). \end{aligned} \tag{32}$$

For the derivation of the effective mass matrix $\underline{m}_{\text{eff}}$ and effective pseudo force $\underline{F}_{\text{pseudo}}$ for the generalized- α method, see Appendices A.1 and A.2.

2.3. Flexible Multibody Dynamics

Flexible multibody dynamics is the analysis of mechanical systems consisting of flexible bodies constrained by kinematic joints modeled with differential–algebraic equations. This work builds upon the previous work of analytical sensitivity analysis with multibody dynamics, including [37,52,53]. In addition to the kinematic constraints with respect to structural dynamics, multibody dynamics also must integrate a nonlinear solver in order to consider the dependencies of the system matrices and vectors on the state variables. This is carried out with Newton steps.

The flexible multibody system is modeled with the floating frame of the reference formulation [54–56]. The generic index-3 differential–algebraic equation for a flexible multibody system with holonomic constraints is written as follows:

$$\underline{m}(\underline{q}, t) \underline{\ddot{q}}(t) + \underline{d}(\underline{q}, t) \underline{\dot{q}}(t) + \underline{k}(\underline{q}, t) \underline{q}(t) + \underline{J}_{\Phi}^T(\underline{q}, t) \underline{\lambda}(t) = \underline{F}_{\text{ext}}(\underline{q}, \underline{\dot{q}}, t) + \underline{F}_v(\underline{q}, \underline{\dot{q}}, t), \tag{33}$$

$$\underline{\Phi}(\underline{q}, t) = \underline{0}, \tag{34}$$

where \underline{q} , $\underline{\dot{q}}$ and $\underline{\ddot{q}}$ are the generalized position, velocity and acceleration vectors; $\underline{\lambda}$ is the vector of Lagrange multipliers of kinematic constraints; the matrices \underline{m} , \underline{d} , and \underline{k} are the mass, damping and stiffness matrices; \underline{J}_{Φ} is the Jacobian matrix of kinematic constraints; $\underline{F}_{\text{ext}}$ is the vector of external forces; \underline{F}_v is the quadratic velocity vector; and $\underline{\Phi}$ is the vector of kinematic constraints at the position level.

Numerical problems related to the solution of index-3 differential–algebraic equations [57] can be avoided by a number of methods including index reduction, coordinate reduction, augmented Lagrangian formulation and preconditioning [58–60]. Although generalized- α time integration is suitable for directly solving index-3 or index-2 differential–algebraic equations, in this work, we address the problem using index reduction leading

to an index-1 differential–algebraic equation in which the kinematic constraint function is differentiated twice with respect to time,

$$\underline{m}\ddot{\underline{q}} + \underline{d}\dot{\underline{q}} + \underline{k}\underline{q} + \underline{J}_{\Phi}^T \underline{\lambda} = \underline{F}_{\text{ext}} + \underline{F}_v, \quad (35)$$

$$\underline{J}_{\Phi} \ddot{\underline{q}} = \underline{F}_c, \quad (36)$$

where \underline{F}_c is the right-hand side of acceleration constraints. As index-1 differential–algebraic equations are susceptible to drift in the kinematic constraints, Baumgarte stabilization [61] is used. Kinematic constraint drift is stabilized in this method by adding damping and restoring terms to the acceleration constraint equations that are proportional to the velocity and position constraints. Details of its implementation, including the sensitivity analysis, are described in [14,62].

The generalized- α time integration is based on

$$\left(\underline{m}\ddot{\underline{q}}\right)_{n+1-\alpha_m} + \left(\underline{d}\dot{\underline{q}}\right)_{n+1-\alpha_f} + \left(\underline{k}\underline{q}\right)_{n+1-\alpha_f} + \left(\underline{J}_{\Phi}^T \underline{\lambda}\right)_{n+1-\alpha_m} = \underline{F}_{\text{ext},n+1-\alpha_f} + \underline{F}_{v,n+1-\alpha_f}, \quad (37)$$

$$\left(\underline{J}_{\Phi} \ddot{\underline{q}}\right)_{n+1-\alpha_m} = \underline{F}_{c,n+1-\alpha_f}, \quad (38)$$

with the intermediate approximations of all force terms of the equations of motion. These intermediate approximations are shown in Appendix B.1. Inserting the intermediate approximations of the force terms acting on the flexible multibody system leads to the equations of motion in residual form given by

$$\begin{aligned} \underline{R}_{1,n+1-\alpha_f} = & (1 - \alpha_m)\underline{m}_{n+1}\ddot{\underline{q}}_{n+1} + \alpha_m\underline{m}_n\ddot{\underline{q}}_n + (1 - \alpha_f)\underline{d}_{n+1}\dot{\underline{q}}_{n+1} + \alpha_f\underline{d}_n\dot{\underline{q}}_n + \\ & + (1 - \alpha_f)\underline{k}_{n+1}\underline{q}_{n+1} + \alpha_f\underline{k}_n\underline{q}_n + (1 - \alpha_m)\underline{J}_{\Phi,n+1}^T \underline{\lambda}_{n+1} + \alpha_m\underline{J}_{\Phi,n}^T \underline{\lambda}_n + \\ & - (1 - \alpha_f)\underline{F}_{\text{ext},n+1} - \alpha_f\underline{F}_{\text{ext},n} - (1 - \alpha_f)\underline{F}_{v,n+1} - \alpha_f\underline{F}_{v,n}, \end{aligned} \quad (39)$$

$$\underline{R}_{2,n+1-\alpha_f} = (1 - \alpha_m)\underline{J}_{\Phi,n+1} \ddot{\underline{q}}_{n+1} + \alpha_m\underline{J}_{\Phi,n} \ddot{\underline{q}}_n - (1 - \alpha_f)\underline{F}_{c,n+1} - \alpha_f\underline{F}_{c,n}. \quad (40)$$

In the case of a nonlinear mass matrix, as is the case with the floating frame of reference formulation, this weighted formulation of the residual equations are first-order accurate. For a formulation of second-order accuracy, see, e.g., [63], where the dynamic equilibrium is enforced exactly at the time steps of the simulation. The derived time integration and its differentiation with the weighted residual formulation can be extended and applied to further formulations.

The effective equation of motion is defined by

$$\begin{bmatrix} \underline{m}_{\text{eff}} & \underline{J}_{\Phi,\text{eff}}^T \\ \underline{J}_{\Phi,\text{eff}} & \underline{0} \end{bmatrix} \begin{bmatrix} \ddot{\underline{q}}_{n+1} \\ \underline{\lambda}_{n+1} \end{bmatrix} = \begin{bmatrix} \underline{F}_{a,\text{eff}} \\ \underline{F}_{c,\text{eff}} \end{bmatrix}, \quad (41)$$

where

$$\underline{m}_{\text{eff}} = (1 - \alpha_m)\underline{m}_{n+1} + (1 - \alpha_f)\gamma\Delta t\underline{d}_{n+1} + (1 - \alpha_f)\beta\Delta t^2\underline{k}_{n+1}, \quad (42)$$

$$\underline{J}_{\Phi,\text{eff}} = (1 - \alpha_m)\underline{J}_{\Phi,n+1}, \quad (43)$$

$$\begin{aligned} \underline{F}_{a,\text{eff}} = & (1 - \alpha_f)\underline{F}_{\text{ext},n+1} + \alpha_f\underline{F}_{\text{ext},n} + (1 - \alpha_f)\underline{F}_{v,n+1} + \alpha_f\underline{F}_{v,n} + \\ & - \alpha_m\underline{m}_n\ddot{\underline{q}}_n - (1 - \alpha_f)\underline{d}_{n+1}\dot{\underline{q}}_{\text{pred}} - \alpha_f\underline{d}_n\dot{\underline{q}}_n + \\ & - (1 - \alpha_f)\underline{k}_{n+1}\underline{q}_{\text{pred}} - \alpha_f\underline{k}_n\underline{q}_n - \alpha_m\underline{J}_{\Phi,n}^T \underline{\lambda}_n, \end{aligned} \quad (44)$$

$$\underline{F}_{c,\text{eff}} = (1 - \alpha_f)\underline{F}_{c,n+1} + \alpha_f\underline{F}_{c,n} - \alpha_m\underline{J}_{\Phi,n} \ddot{\underline{q}}_n. \quad (45)$$

The derivations of the effective equation of motion and the terms $\underline{m}_{\text{eff}}$, $\underline{J}_{\Phi, \text{eff}}$, $\underline{F}_{a, \text{eff}}$ and $\underline{F}_{c, \text{eff}}$ are shown in Appendix B.1.

With flexible multibody dynamics, the system parameters (matrix- and vector-valued for $n_{\text{dof}} > 1$) including mass, damping, stiffness, constraints and forces are generally dependent on state variables. Therefore, a nonlinear solver is needed to solve the system and the Newton–Raphson method is used here of the following form:

$$\begin{bmatrix} \frac{\partial R_{1,n+1-\alpha_f}}{\partial \underline{\ddot{q}}_{n+1}} & \frac{\partial R_{1,n+1-\alpha_f}}{\partial \underline{\lambda}_{n+1}} \\ \frac{\partial R_{2,n+1-\alpha_f}}{\partial \underline{\ddot{q}}_{n+1}} & \frac{\partial R_{2,n+1-\alpha_f}}{\partial \underline{\lambda}_{n+1}} \end{bmatrix} \begin{bmatrix} \Delta \underline{\ddot{q}}_{n+1} \\ \Delta \underline{\lambda}_{n+1} \end{bmatrix} + \begin{bmatrix} \underline{R}_{1,n+1-\alpha_f} \\ \underline{R}_{2,n+1-\alpha_f} \end{bmatrix} = \underline{0}. \quad (46)$$

With the dependencies shown in (33) and (34), and the terms of the Jacobian matrix (tangent operator) of the residuals with respect to $\underline{\ddot{q}}_{n+1}$ and $\underline{\lambda}_{n+1}$ are given by

$$\begin{aligned} \frac{\partial R_{1,n+1-\alpha_f}}{\partial \underline{\ddot{q}}_{n+1}} &= (1 - \alpha_m) \left(\beta \Delta t^2 \frac{\partial \underline{m}_{n+1}}{\partial \underline{q}_{n+1}} \underline{\ddot{q}}_{n+1} + \underline{m}_{n+1} \right) + \\ &+ (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{d}_{n+1}}{\partial \underline{q}_{n+1}} \underline{\dot{q}}_{n+1} + \gamma \Delta t \underline{d}_{n+1} \right) + \\ &+ (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{k}_{n+1}}{\partial \underline{q}_{n+1}} \underline{q}_{n+1} + \beta \Delta t^2 \underline{k}_{n+1} \right) + \\ &+ (1 - \alpha_m) \beta \Delta t^2 \frac{\partial \underline{J}_{\Phi, n+1}^T}{\partial \underline{q}_{n+1}} \underline{\lambda}_{n+1} + \\ &- (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{F}_{\text{ext}, n+1}}{\partial \underline{q}_{n+1}} + \gamma \Delta t \frac{\partial \underline{F}_{\text{ext}, n+1}}{\partial \underline{\dot{q}}_{n+1}} \right) + \\ &- (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{F}_{v, n+1}}{\partial \underline{q}_{n+1}} + \gamma \Delta t \frac{\partial \underline{F}_{v, n+1}}{\partial \underline{\dot{q}}_{n+1}} \right), \end{aligned} \quad (47)$$

$$\begin{aligned} \frac{\partial R_{2,n+1-\alpha_f}}{\partial \underline{\ddot{q}}_{n+1}} &= (1 - \alpha_m) \left(\beta \Delta t^2 \frac{\partial \underline{J}_{\Phi, n+1}}{\partial \underline{q}_{n+1}} \underline{\ddot{q}}_{n+1} + \underline{J}_{\Phi, n+1} \right) + \\ &- (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{F}_{c, n+1}}{\partial \underline{q}_{n+1}} + \gamma \Delta t \frac{\partial \underline{F}_{c, n+1}}{\partial \underline{\dot{q}}_{n+1}} \right), \end{aligned} \quad (48)$$

$$\frac{\partial R_{1,n+1-\alpha_f}}{\partial \underline{\lambda}_{n+1}} = (1 - \alpha_m) \underline{J}_{\Phi, n+1}^T, \quad (49)$$

$$\frac{\partial R_{2,n+1-\alpha_f}}{\partial \underline{\lambda}_{n+1}} = \underline{0}. \quad (50)$$

The derivation of these terms is shown in Appendix B.2.

The direct differentiation of the equations of motion (39) and (40) leads to the governing equation of the sensitivity analysis,

$$\begin{aligned} \underline{\underline{\nabla R}}_{1,n+1-\alpha_f} = & (1 - \alpha_m) \underline{\underline{\nabla m}}_{n+1} \underline{\underline{\ddot{q}}}_{n+1} + \alpha_m \underline{\underline{\nabla m}}_n \underline{\underline{\ddot{q}}}_n + \\ & + (1 - \alpha_m) \underline{\underline{m}}_{n+1} \underline{\underline{\nabla \ddot{q}}}_{n+1} + \alpha_m \underline{\underline{m}}_n \underline{\underline{\nabla \ddot{q}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{\nabla d}}_{n+1} \underline{\underline{\dot{q}}}_{n+1} + \alpha_f \underline{\underline{\nabla d}}_n \underline{\underline{\dot{q}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{d}}_{n+1} \underline{\underline{\nabla \dot{q}}}_{n+1} + \alpha_f \underline{\underline{d}}_n \underline{\underline{\nabla \dot{q}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{\nabla k}}_{n+1} \underline{\underline{q}}_{n+1} + \alpha_f \underline{\underline{\nabla k}}_n \underline{\underline{q}}_n + \\ & + (1 - \alpha_f) \underline{\underline{k}}_{n+1} \underline{\underline{\nabla q}}_{n+1} + \alpha_f \underline{\underline{k}}_n \underline{\underline{\nabla q}}_n + \\ & + (1 - \alpha_m) \underline{\underline{\nabla J}}_{\Phi,n+1}^T \underline{\underline{\lambda}}_{n+1} + \alpha_m \underline{\underline{\nabla J}}_{\Phi,n}^T \underline{\underline{\lambda}}_n + \\ & + (1 - \alpha_m) \underline{\underline{J}}_{\Phi,n+1}^T \underline{\underline{\nabla \lambda}}_{n+1} + \alpha_m \underline{\underline{J}}_{\Phi,n}^T \underline{\underline{\nabla \lambda}}_n + \\ & - (1 - \alpha_f) \underline{\underline{\nabla F}}_{\text{ext},n+1} - \alpha_f \underline{\underline{\nabla F}}_{\text{ext},n} - (1 - \alpha_f) \underline{\underline{\nabla F}}_{v,n+1} - \alpha_f \underline{\underline{\nabla F}}_{v,n}, \end{aligned} \quad (51)$$

$$\begin{aligned} \underline{\underline{\nabla R}}_{2,n+1-\alpha_f} = & (1 - \alpha_m) \underline{\underline{\nabla J}}_{\Phi,n+1} \underline{\underline{\ddot{q}}}_{n+1} + \alpha_m \underline{\underline{\nabla J}}_{\Phi,n} \underline{\underline{\ddot{q}}}_n + \\ & + (1 - \alpha_m) \underline{\underline{J}}_{\Phi,n+1} \underline{\underline{\nabla \ddot{q}}}_{n+1} + \alpha_m \underline{\underline{J}}_{\Phi,n} \underline{\underline{\nabla \ddot{q}}}_n + \\ & - (1 - \alpha_f) \underline{\underline{\nabla F}}_{c,n+1} - \alpha_f \underline{\underline{\nabla F}}_{c,n}. \end{aligned} \quad (52)$$

The governing sensitivity equation of the effective system is defined by the differentiation of (41),

$$\begin{bmatrix} \underline{\underline{m}}_{\text{eff}} & \underline{\underline{J}}_{\Phi,\text{eff}}^T \\ \underline{\underline{J}}_{\Phi,\text{eff}} & \underline{\underline{0}} \end{bmatrix} \begin{bmatrix} \underline{\underline{\nabla \ddot{q}}}_{n+1} \\ \underline{\underline{\nabla \lambda}}_{n+1} \end{bmatrix} = \begin{bmatrix} \underline{\underline{F}}_{a,\text{pseudo}} \\ \underline{\underline{F}}_{c,\text{pseudo}} \end{bmatrix}, \quad (53)$$

where

$$\begin{bmatrix} \underline{\underline{F}}_{a,\text{pseudo}} \\ \underline{\underline{F}}_{c,\text{pseudo}} \end{bmatrix} = \begin{bmatrix} \underline{\underline{\nabla F}}_{a,\text{eff}} \\ \underline{\underline{\nabla F}}_{c,\text{eff}} \end{bmatrix} - \begin{bmatrix} \underline{\underline{\nabla m}}_{\text{eff}} & \underline{\underline{\nabla J}}_{\Phi,\text{eff}}^T \\ \underline{\underline{\nabla J}}_{\Phi,\text{eff}} & \underline{\underline{0}} \end{bmatrix} \begin{bmatrix} \underline{\underline{\ddot{q}}}_{n+1} \\ \underline{\underline{\lambda}}_{n+1} \end{bmatrix}, \quad (54)$$

leading to

$$\begin{aligned} \underline{\underline{F}}_{a,\text{pseudo}} = & (1 - \alpha_f) \underline{\underline{\nabla F}}_{\text{ext},n+1} + \alpha_f \underline{\underline{\nabla F}}_{\text{ext},n} + (1 - \alpha_f) \underline{\underline{\nabla F}}_{v,n+1} + \alpha_f \underline{\underline{\nabla F}}_{v,n} + \\ & - (1 - \alpha_m) \underline{\underline{\nabla m}}_{n+1} \underline{\underline{\ddot{q}}}_{n+1} - \alpha_m \underline{\underline{\nabla m}}_n \underline{\underline{\ddot{q}}}_n - \alpha_m \underline{\underline{m}}_n \underline{\underline{\nabla \ddot{q}}}_n + \\ & - (1 - \alpha_f) \underline{\underline{\nabla d}}_{n+1} \underline{\underline{\dot{q}}}_{n+1} - \alpha_f \underline{\underline{\nabla d}}_n \underline{\underline{\dot{q}}}_n - (1 - \alpha_f) \underline{\underline{d}}_{n+1} \underline{\underline{\nabla \dot{q}}}_{\text{pred}} - \alpha_f \underline{\underline{d}}_n \underline{\underline{\nabla \dot{q}}}_n + \\ & - (1 - \alpha_f) \underline{\underline{\nabla k}}_{n+1} \underline{\underline{q}}_{n+1} - \alpha_f \underline{\underline{\nabla k}}_n \underline{\underline{q}}_n - (1 - \alpha_f) \underline{\underline{k}}_{n+1} \underline{\underline{\nabla q}}_{\text{pred}} - \alpha_f \underline{\underline{k}}_n \underline{\underline{\nabla q}}_n + \\ & - (1 - \alpha_m) \underline{\underline{\nabla J}}_{\Phi,n+1}^T \underline{\underline{\lambda}}_{n+1} - \alpha_m \underline{\underline{\nabla J}}_{\Phi,n}^T \underline{\underline{\lambda}}_n - \alpha_m \underline{\underline{J}}_{\Phi,n}^T \underline{\underline{\nabla \lambda}}_n, \end{aligned} \quad (55)$$

$$\begin{aligned} \underline{\underline{F}}_{c,\text{pseudo}} = & (1 - \alpha_f) \underline{\underline{\nabla F}}_{c,n+1} + \alpha_f \underline{\underline{\nabla F}}_{c,n} + \\ & - (1 - \alpha_m) \underline{\underline{\nabla J}}_{\Phi,n+1} \underline{\underline{\ddot{q}}}_{n+1} - \alpha_m \underline{\underline{\nabla J}}_{\Phi,n} \underline{\underline{\ddot{q}}}_n - \alpha_m \underline{\underline{J}}_{\Phi,n} \underline{\underline{\nabla \ddot{q}}}_n. \end{aligned} \quad (56)$$

The derivation of the effective mass matrix and the effective constraint Jacobian matrix is shown in Appendix B.1 and the derivation of the effective pseudo force terms is shown in Appendix B.3.

The Newton–Raphson step for the sensitivity analysis is given by

$$\begin{bmatrix} \frac{\partial \nabla R_{1,n+1-\alpha_f}}{\partial \nabla \ddot{q}_{n+1}} & \frac{\partial \nabla R_{1,n+1-\alpha_f}}{\partial \nabla \lambda_{n+1}} \\ \frac{\partial \nabla R_{2,n+1-\alpha_f}}{\partial \nabla \ddot{q}_{n+1}} & \frac{\partial \nabla R_{2,n+1-\alpha_f}}{\partial \nabla \lambda_{n+1}} \end{bmatrix} \begin{bmatrix} \Delta \nabla \ddot{q}_{n+1} \\ \Delta \nabla \lambda_{n+1} \end{bmatrix} + \begin{bmatrix} \nabla R_{1,n+1-\alpha_f} \\ \nabla R_{2,n+1-\alpha_f} \end{bmatrix} = \underline{0}. \tag{57}$$

As shown in [7], the Jacobian of the residual sensitivity with respect to $\nabla \ddot{q}_{n+1}$ and $\nabla \lambda_{n+1}$ reduces to

$$\begin{bmatrix} \frac{\partial \nabla R_{1,n+1-\alpha_f}}{\partial \nabla \ddot{q}_{n+1}} & \frac{\partial \nabla R_{1,n+1-\alpha_f}}{\partial \nabla \lambda_{n+1}} \\ \frac{\partial \nabla R_{2,n+1-\alpha_f}}{\partial \nabla \ddot{q}_{n+1}} & \frac{\partial \nabla R_{2,n+1-\alpha_f}}{\partial \nabla \lambda_{n+1}} \end{bmatrix} = \begin{bmatrix} \frac{\partial R_{1,n+1-\alpha_f}}{\partial \ddot{q}_{n+1}} & \frac{\partial R_{1,n+1-\alpha_f}}{\partial \lambda_{n+1}} \\ \frac{\partial R_{2,n+1-\alpha_f}}{\partial \ddot{q}_{n+1}} & \frac{\partial R_{2,n+1-\alpha_f}}{\partial \lambda_{n+1}} \end{bmatrix} \underline{e}_f \tag{58}$$

which reduces the dimensions of the coefficient matrix from $(n_q + n_\Phi) \times n_x \times (n_q + n_\Phi) \times n_x$ to $(n_q + n_\Phi) \times (n_q + n_\Phi)$. The solution of the resulting equation,

$$\begin{bmatrix} \frac{\partial R_{1,n+1-\alpha_f}}{\partial \ddot{q}_{n+1}} & \frac{\partial R_{1,n+1-\alpha_f}}{\partial \lambda_{n+1}} \\ \frac{\partial R_{2,n+1-\alpha_f}}{\partial \ddot{q}_{n+1}} & \frac{\partial R_{2,n+1-\alpha_f}}{\partial \lambda_{n+1}} \end{bmatrix} \begin{bmatrix} \Delta \nabla \ddot{q}_{n+1} \\ \Delta \nabla \lambda_{n+1} \end{bmatrix} + \begin{bmatrix} \nabla R_{1,n+1-\alpha_f} \\ \nabla R_{2,n+1-\alpha_f} \end{bmatrix} = \underline{0}, \tag{59}$$

is found using the same solving scheme for the nonlinear solver of the primal analysis (46). Furthermore, both the primal analysis (46) and sensitivity analysis (59) have the same coefficient matrix, which also allows one to reuse the coefficient matrix of the primal analysis in the sensitivity analysis without the need for further evaluations.

3. Numerical Results

In this section, numerical results are shown comparing the introduced method for the sensitivity analysis for generalized- α time integration (primal analysis introduced in Section 2) with that of Newmark- β . The methods are compared with applications to both structural dynamics as well as flexible multibody dynamics.

3.1. Structural Dynamics

The linear structural dynamic problem used here as a demonstration example is based on [64], which has found usage as a benchmark in [65–71]. This unitless problem is schematically shown in Figure 2. This problem is defined by

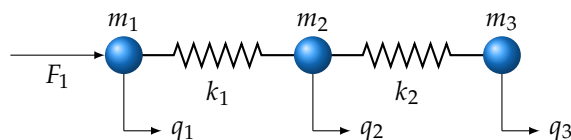


Figure 2. Structural dynamics example.

$$\underbrace{\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix}}_{\underline{m}} \underbrace{\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix}}_{\underline{\ddot{q}}} + \underbrace{\begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix}}_{\underline{k}} \underbrace{\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}}_{\underline{q}} - \underbrace{\begin{bmatrix} F_1 \\ 0 \\ 0 \end{bmatrix}}_{\underline{F}} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}}_{\underline{R}}, \tag{60}$$

with the prescribed displacement,

$$q_1 = \sin \omega t, \tag{61}$$

and the unitless values

$$\begin{aligned}
 m_1 &= 0, \\
 m_2 &= 1, \\
 m_3 &= 1, \\
 k_1 &= 10^7, \\
 k_2 &= 1, \\
 \omega &= 1.2.
 \end{aligned}$$

The masses m_2 and m_3 as well as the stiffnesses k_1 and k_2 are considered design variables with respect to which the design sensitivities are calculated. The problem consists of stiff (k_1) and flexible (k_2) parts that are typical of complex, practical structural–mechanical systems [64].

Considering the prescribed displacement, this problem can be reduced to two degrees of freedom,

$$\underbrace{\begin{bmatrix} m_2 & 0 \\ 0 & m_3 \end{bmatrix}}_{\underline{m}} \underbrace{\begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix}}_{\underline{\ddot{q}}} + \underbrace{\begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix}}_{\underline{k}} \underbrace{\begin{bmatrix} q_2 \\ q_3 \end{bmatrix}}_{\underline{q}} - \underbrace{\begin{bmatrix} k_1 q_1 \\ 0 \end{bmatrix}}_{\underline{F}} = \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{\underline{R}}, \tag{62}$$

which we implement in our in-house, Python-based solver framework SIMULI in the time $t = [0, 10]$ with the initial conditions

$$\begin{aligned}
 q_2 &= 0, \\
 q_3 &= 0, \\
 \dot{q}_2 &= 0, \\
 \dot{q}_3 &= 0.
 \end{aligned}$$

As reported in [64], unitless time steps of $\Delta t = 0.2618$ are used, giving the following values for the ratio step size to the fundamental periods of the system:

$$\begin{aligned}
 \frac{\Delta t}{T_F} &= 0.05, \\
 \frac{\Delta t}{T_1} &= 0.041667, \\
 \frac{\Delta t}{T_2} &= 131.76,
 \end{aligned}$$

where T_F is the fundamental period of the forcing function and $T_{1,2}$, for the first and second modes, $T_1 = 6.283$ and $T_2 = 0.001987$.

The results for both Newmark- β and generalized- α time integration methods are shown in Figure 3 and compared with the exact solution. The exact solution is calculated via mode superposition per [72]. The numerical damping parameter of generalized- α time integration is set as $\varrho_\infty = 0.55$. As expected with generalized- α time integration, the unphysical, high-frequent oscillatory behavior of the response can be avoided. As such, the beneficial effects of numerical damping with generalized- α time integration can be clearly seen in the velocity and acceleration of node 2, while maintaining the integrity of the other responses.

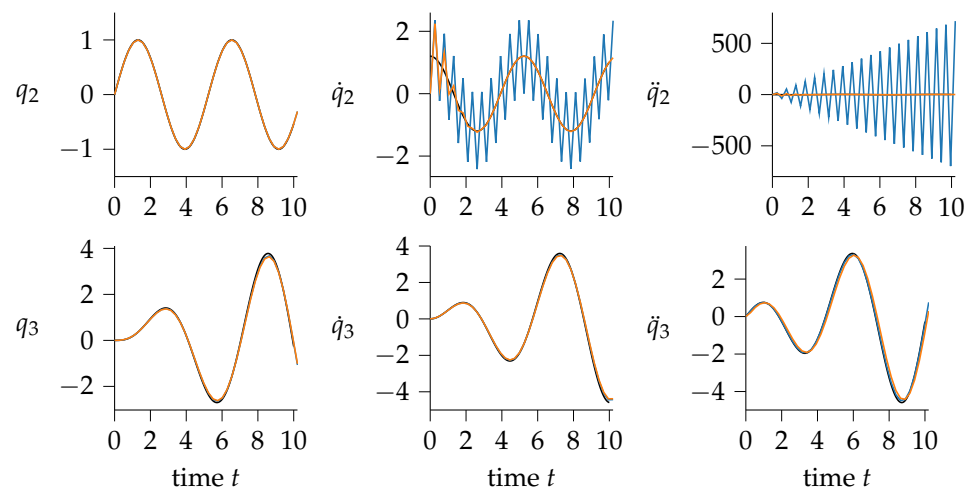


Figure 3. Responses of the structural dynamic example (unitless) calculated with —Newmark- β and —generalized- α compared with —exact solution (undermost line).

The sensitivity analysis developed above is applied and these results can be seen in Figures 4–6. Analytical sensitivity analysis with Newmark- β and generalized- α time integration methods are also compared with their numerical sensitivity implementations via finite differencing using a relative perturbation of 10^{-6} .

In cases where the Newmark- β time integration shows instability or noise, the numerical damping allows for usable sensitivities. In some cases, noise—albeit at a much lower level—persists in the first iterations before being damped out—after approximately 2.5 s (or approximately 10 iterations). Attention is drawn to those sensitivities susceptible to stability issues, namely

- sensitivities of position 2 with respect to stiffness 1 and mass 2 (Figure 4);
- sensitivities of velocity 2 with respect to stiffness 1 and mass 2 (Figure 5);
- sensitivities of acceleration 2 with respect to stiffness 1, stiffness 2 and mass 2 (Figure 6);
- sensitivities of acceleration 3 with respect to stiffness 1 and mass 2 (Figure 6).

Generalized- α time integration is not afflicted by this instability, resulting in sensitivities that can be successfully used in gradient-based algorithms.

Furthermore, generalized- α time integration eliminates the noise in those sensitivities present when using Newmark- β . This is seen with the following sensitivity responses:

- sensitivities of velocity 2 with respect to stiffness 2 (Figure 5);
- sensitivities of velocity 3 with respect to stiffness 1 and mass 2 (Figure 5).

The analytical and numerical results are, in most plots, nearly exact. There are cases, however, where the negative consequences of the numerical instability with Newmark- β are increased. The reader's attention is especially called to

- sensitivity of velocity 2 with respect to stiffness 2 (Figure 5);
- sensitivity of velocity 2 with respect to mass 3 (Figure 5);
- sensitivity of acceleration 2 with respect to stiffness 2 (Figure 6);
- sensitivity of acceleration 2 with respect to mass 3 (Figure 6).

It can be seen that both analytical and numerical sensitivities with generalized- α time integration are not negatively affected by numerical instabilities, as is the case with Newmark- β .

The number of time steps needed for the damping to take effect can be reduced by reducing the numerical damping parameter ρ_∞ towards zero (i.e., more damping), though also affecting the physical response to a greater degree. As time continues, the instability grows, with Newmark- β time integration and therefore a clear advantage can be seen with generalized- α . It can be clearly seen how gradient-based solvers would struggle to use the gradient information provided with Newmark- β time integration.

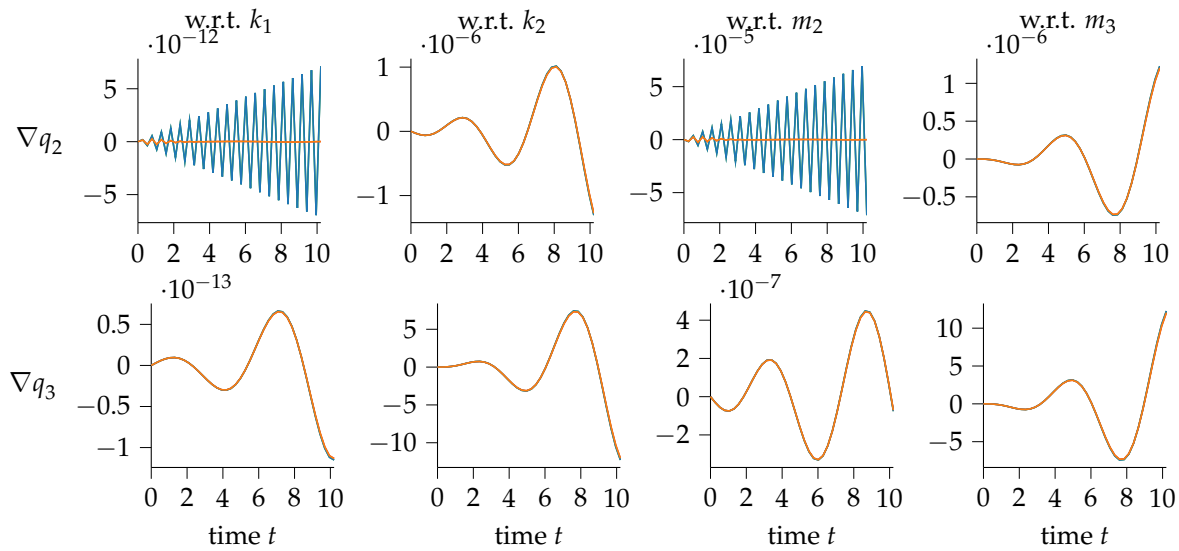


Figure 4. Position sensitivities ∇q of the structural dynamic example with respect to k_1, k_2, m_2 and m_3 (unitless), analytically calculated with —Newmark- β and —generalized- α compared with those calculated numerically via finite differencing with —Newmark- β and —generalized- α (undermost lines). Note: no visible difference between finite differencing and analytical sensitivities.

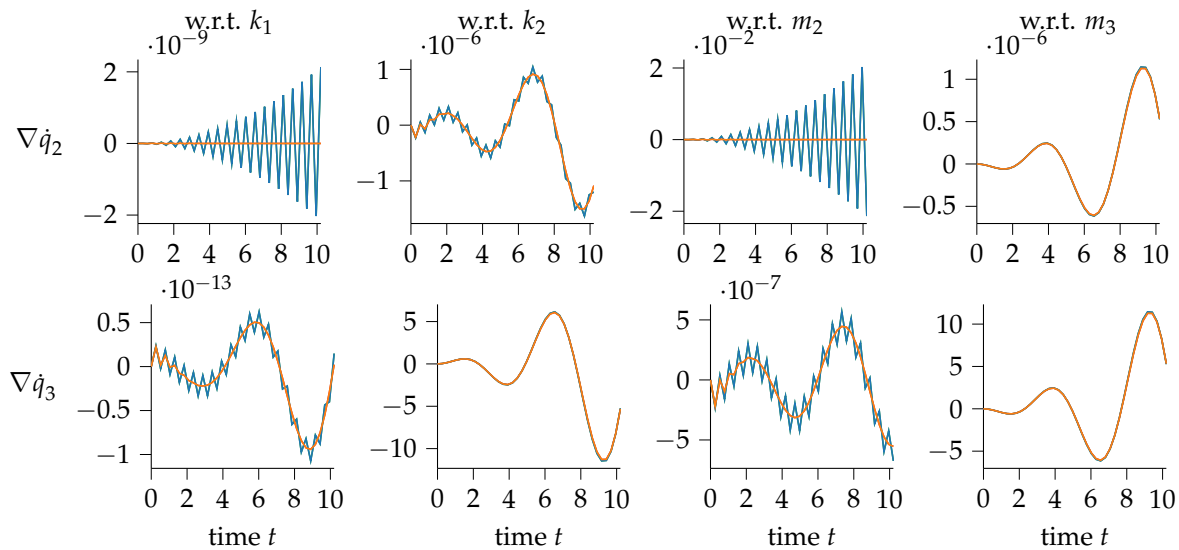


Figure 5. Velocity sensitivities $\nabla \dot{q}$ of the structural dynamic example with respect to k_1, k_2, m_2 and m_3 (unitless), analytically calculated with —Newmark- β and —generalized- α compared with those calculated numerically via finite differencing with —Newmark- β and —generalized- α (undermost lines). Note: no visible difference between finite differencing and analytical sensitivities.

For our modest case of four design variables, we show an acceleration of two, where the speedup S is defined by the ratio of computational time for finite differencing t_{FD} to that of analytical sensitivities t_{ana} ,

$$S = \frac{t_{FD}}{t_{ana}}. \tag{63}$$

The computational time required for analytical sensitivity analysis represents approximately the ideal computational effort twice that of the primal analysis, i.e., analysis without sensitivity analysis. Computation reduction is the same for both Newmark- β and generalized- α time integration methods, as can be seen in Table 1.

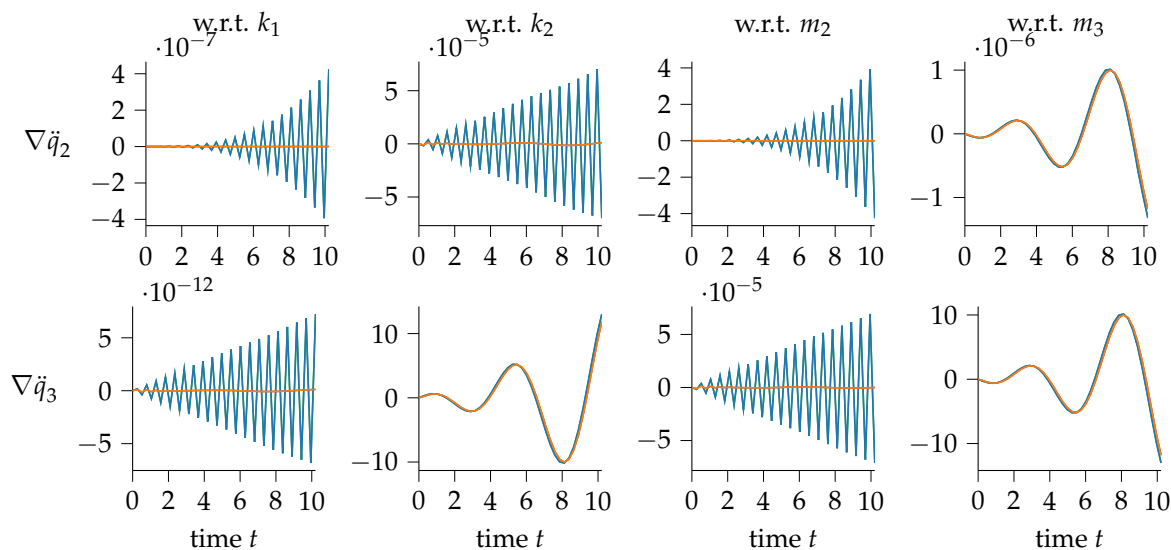


Figure 6. Acceleration sensitivities $\nabla \ddot{q}$ of the structural dynamic example with respect to k_1 , k_2 , m_2 and m_3 (unitless) analytically calculated with —Newmark- β and —generalized- α compared with those numerically calculated via finite differencing with —Newmark- β and —generalized- α (undermost lines). Note: no visible difference between finite differencing and analytical sensitivities.

Table 1. Computational effort for the structural dynamics example.

Time Integration	Sensitivity Analysis	Calculation Time (s) ¹	Speedup ²
Newmark- β	None	0.0555	–
Newmark- β	Finite differencing	0.2088	–
Newmark- β	Analytical	0.1023	2.0410
Generalized- α	None	0.0458	–
Generalized- α	Finite differencing	0.2122	–
Generalized- α	Analytical	0.0993	2.1370

¹ Intel Core i5-7200U CPU 2.50 GHz, ² acceleration of analytical sensitivity analysis with respect to finite differencing.

3.2. Flexible Slider–Crank Mechanism

The slider–crank mechanism, as shown in Figure 7, is modeled with flexible multibody dynamics using the floating frame of reference formulation. The mechanism consists of four bodies that are connected by three revolute joints and one prismatic joint, modeled as ideal joints without friction. The ground and the slider are considered to be rigid, while the crank and the rod are considered to be flexible and are modeled with Euler–Bernoulli beams. Figure 8 shows the flexible multibody system of the mechanism. The inertial reference frame coinciding with the reference frame of the ground is shown in blue. The body reference frames of the crank, the rod and the slider are shown in red. The undeformed bodies of the crank and the rod are shown in gray and the deformed bodies are shown in black, where the points represent the nodes of the finite elements for the geometric discretization. The coordinates that are investigated in detail for the primal analysis and the sensitivity analysis are shown in green. The dimensions of the bodies are listed in Table 2. The bodies are made of steel and Table 3 shows the material properties. A linear elastic material model with isotropic behavior is applied. In this example, the SI-derived MPa unit system is used with length in millimeters (mm), mass in tonnes (t), time in second (s), force in Newtons (N), Angle in radian (rad) and stress in megapascals (MPa).

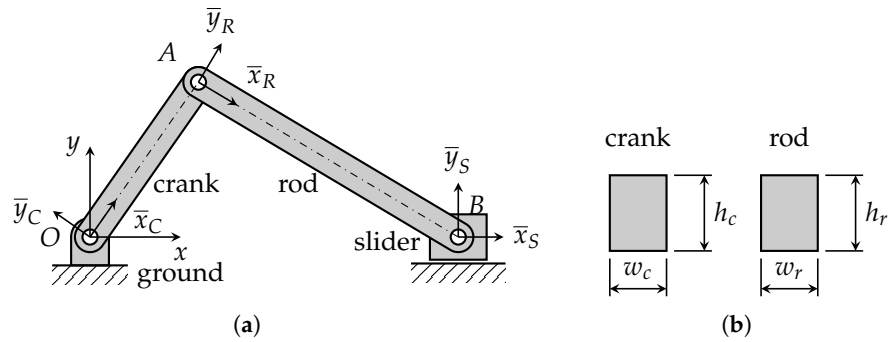


Figure 7. Slider-crank mechanism: (a) schematic diagram; and (b) cross-sections.

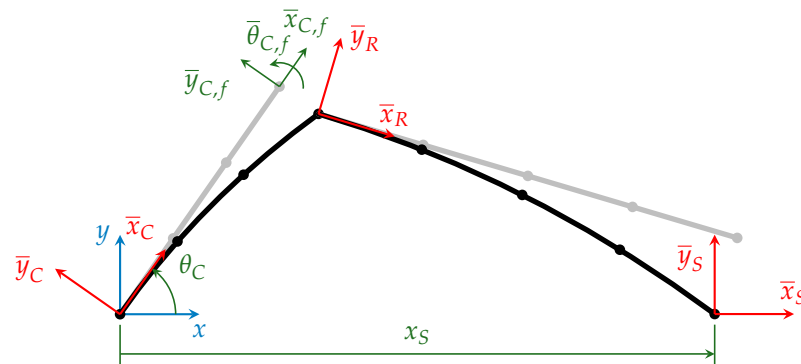


Figure 8. Flexible multibody system of the slider-crank mechanism with the undeformed bodies in gray, the deformed bodies in black, the inertial reference frame in blue, the body reference frames in red and the coordinates that are investigated in green.

Table 2. Geometry of the slider-crank mechanism.

Property	Symbol	Crank	Rod	Slider	Units
Width	w	20	20	30	mm
Height	h	30	30	30	mm
Length	l	120	180	40	mm

Table 3. Material properties.

Property	Symbol	Value	Units
Density	ρ	7.85×10^{-9}	$\frac{t}{mm^3}$
Elastic modulus	E	210 000	MPa
Poisson ratio	ν	0.3	—

The generalized coordinates of a flexible multibody system modeled with the floating frame of the reference formulation and Euler–Bernoulli beams are given by

$$\underline{q} = \left[\underline{r}^T \quad \theta \quad \underline{\bar{q}}_f^T \right]^T, \tag{64}$$

where \underline{r} is the position of the body reference frame, θ is the orientation of the body reference frame and $\underline{\bar{q}}_f$ are the nodal displacements in the body coordinates of each node on the flexible body. Since no damping is considered, the equations of motion of a flexible body are given by

$$\underline{\underline{m}} \underline{\underline{\ddot{q}}} + \underline{\underline{k}} \underline{\underline{q}} + \underline{\underline{J}}_{\Phi}^T \underline{\underline{\lambda}} = \underline{\underline{F}}_{\text{ext}} + \underline{\underline{F}}_v, \tag{65}$$

$$\underline{\underline{J}}_{\Phi} \underline{\underline{q}} = \underline{\underline{F}}_c, \tag{66}$$

where the system matrices of the flexible bodies are given by

$$\underline{m} = \begin{bmatrix} \underline{m}_{tt} & \underline{m}_{tr} & \underline{m}_{tf} \\ & \underline{m}_{rr} & \underline{m}_{rf} \\ \text{sym.} & & \underline{m}_{ff} \end{bmatrix}, \quad (67)$$

$$\underline{k} = \begin{bmatrix} 0 & 0 & 0 \\ & 0 & 0 \\ \text{sym.} & & \underline{k}_{ff} \end{bmatrix}, \quad (68)$$

and the system vectors are given by

$$\underline{F}_{\text{ext}} = \begin{bmatrix} \underline{F}_{\text{ext},t} \\ \underline{F}_{\text{ext},r} \\ \underline{F}_{\text{ext},f} \end{bmatrix}, \quad (69)$$

$$\underline{F}_v = \begin{bmatrix} \underline{F}_{v,t} \\ \underline{F}_{v,r} \\ \underline{F}_{v,f} \end{bmatrix}, \quad (70)$$

with the index denoting translational terms t , rotational terms r and flexible terms f .

For the dynamic simulation of the mechanism, the above-mentioned solver framework SIMULI is used. The simulation is performed in the time interval $t = [0 \text{ s}, 0.025 \text{ s}]$. For both generalized- α and Newmark- β time integration, a time step of $\Delta t = 1 \times 10^{-4} \text{ s}$ is chosen for comparison. The resulting values for the ratio step size to fundamental periods of the system for the first ten non-zero eigenfrequencies are found in Figure 9. The values for the ratio step size to fundamental periods range from 0.2119 s for first non-zero eigenfrequency to 3.9919 s for the tenth non-zero eigenfrequency and to 17.1157 s for the twentieth non-zero eigenfrequency (not pictured in the plot).

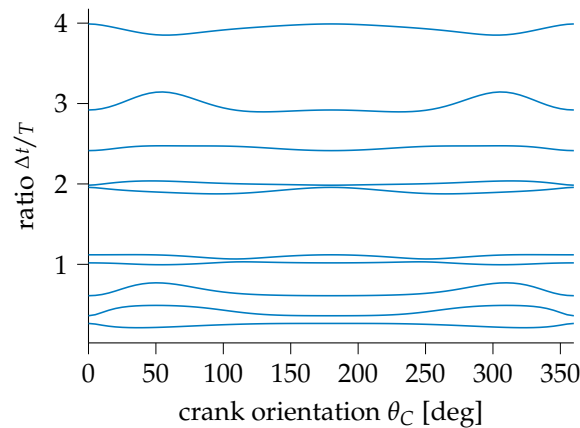


Figure 9. Ratio time step to fundamental periods for the first ten non-zero eigenfrequencies of the slider–crank mechanism.

For the initial conditions, the angular position and velocity of the crank are defined with

$$\begin{aligned} \theta_C &= 0 \text{ rad}, \\ \dot{\theta}_C &= 150\pi \frac{\text{rad}}{\text{s}}, \end{aligned}$$

and the initial conditions of the other bodies are derived through the kinematic analysis of the system. The dimensions of the cross-sections of the crank h_C , w_C and the rod h_R , w_R are used as design variables in the sensitivity analysis.

The results of the simulation of the coordinates that are highlighted in Figure 8 are shown in Figure 10. With Newmark- β , vibrations at high frequencies are observed. With generalized- α , vibrations are observed in the beginning of the simulation that are numerically damped with ongoing simulation time. In addition to the generalized coordinates, the maximum stresses on the upper and lower fiber of the beam elements of the flexible bodies including the crank and the rod are computed. The Kreisselmeier–Steinhauser function [73,74] is used to approximate the maximum stress value of the bodies at each time step with a differentiable function and the results are shown in Figure 11. With Newmark- β , the vibrating behavior of the system causes higher flexible deformation and therefore unrealistic high stress in the components. The numerical damping with generalized- α leads to lower maximum stress values and a more realistic stress distribution over time.

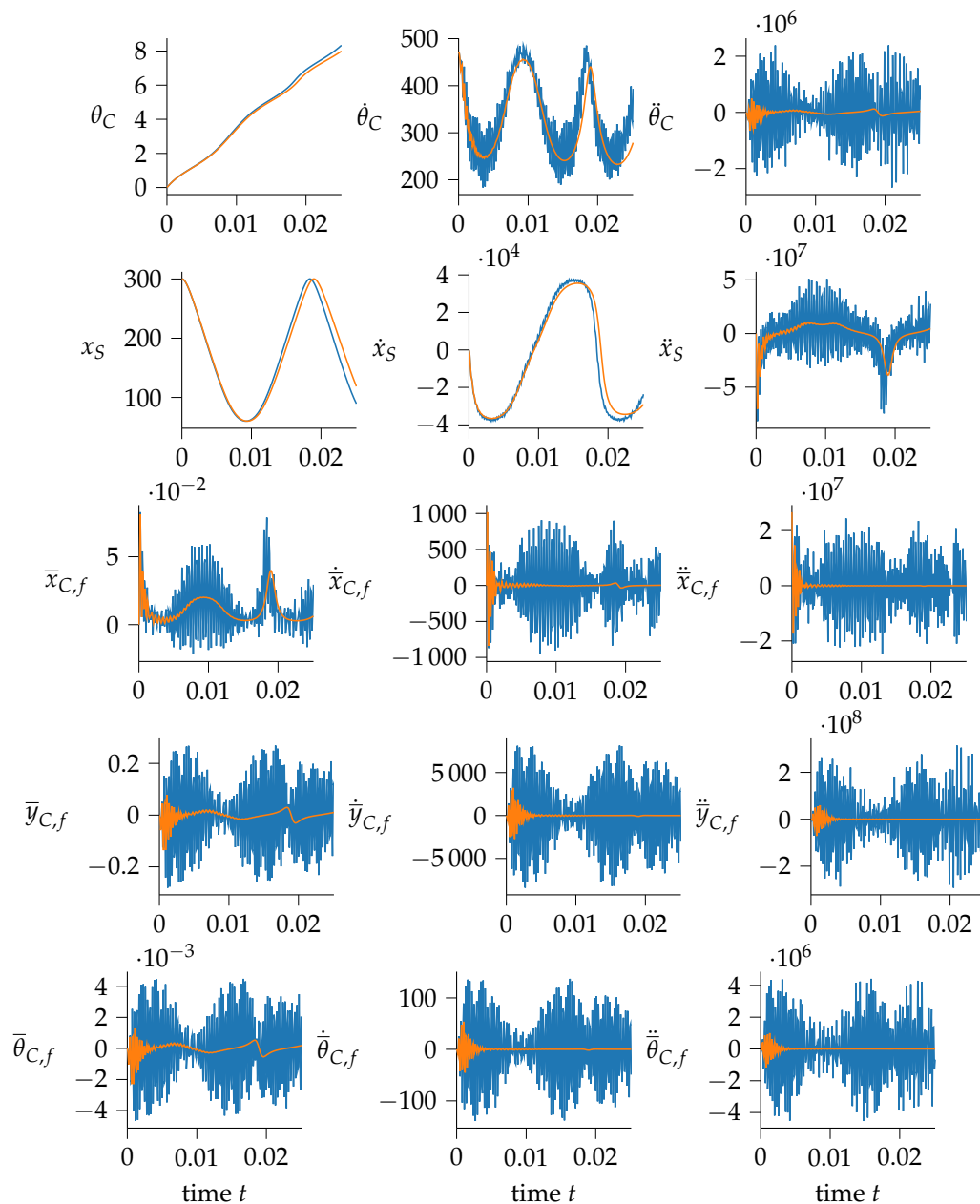


Figure 10. Responses of the slider–crank mechanism (values in MPa units, i.e., s, mm, $\frac{\text{mm}}{\text{s}}$, $\frac{\text{mm}}{\text{s}^2}$ for translations and rad, $\frac{\text{rad}}{\text{s}}$, $\frac{\text{rad}}{\text{s}^2}$ for rotations) calculated with — Newmark- β and — generalized- α .

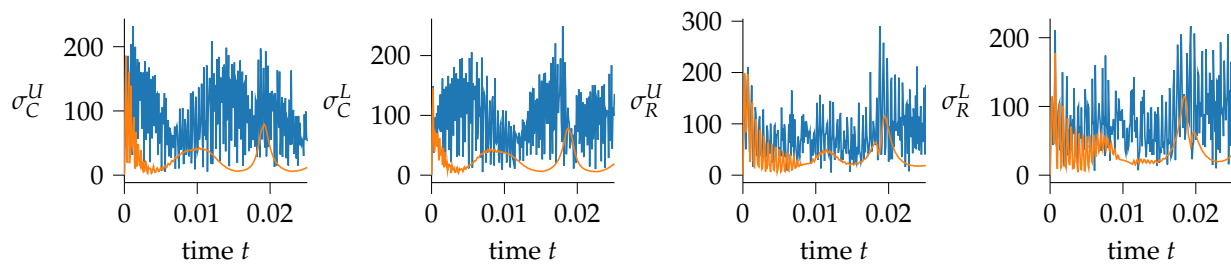


Figure 11. Maximum stress in MPa on the upper and lower fiber of the crank and the rod over time in s calculated with —Newmark- β and —generalized- α .

The results of the sensitivity analysis of the coordinates that are highlighted in Figure 8 are shown in Figures 12–14. The analyzed coordinates include the rotation of the floating frame of the crank θ_C , the translation in the x direction of the floating frame of the slider x_S and the three flexible coordinates of the node in A on the crank $\bar{x}_{C,f}$, $\bar{y}_{C,f}$ and $\bar{\theta}_{C,f}$. The design variables used for the sensitivity analysis are the cross-sectional dimensions of the crank h_C , w_C and the rod h_R , w_R .

Similarly to the above, with Newmark- β , the results show a vibrating behavior that would lead to significant problems when using the sensitivities in gradient-based design optimization. With generalized- α , the numerical damping leads to usable sensitivity values after a few time steps. Figure 12 shows that the position sensitivities of the position and orientation coordinates of the floating frames are less noisy compared to the flexible coordinates.

Position sensitivities that suffer little from noise are limited to the position and orientation coordinates of the floating frames with respect to several design variables and include the following:

- sensitivities of θ_C with respect to w_C and w_R ;
- sensitivities of x_S with respect to h_C , w_C , and w_R .

Position sensitivities with significant noise include the position and orientation coordinates of the floating frames with respect to a few specific design variables as well as the flexible coordinates with respect to all design variables. The list of noisy responses of position sensitivities is given in the following:

- sensitivities of θ_C with respect to h_C and h_R ;
- sensitivities of x_S with respect to h_R ;
- sensitivities of $\bar{x}_{C,f}$ with respect to h_C , w_C , h_R and w_R ;
- sensitivities of $\bar{y}_{C,f}$ with respect to h_C ; w_C , h_R and w_R ;
- sensitivities of $\bar{\theta}_{C,f}$ with respect to h_C , w_C , h_R and w_R .

The sensitivities of the velocities as shown in Figure 13 and the sensitivities of the accelerations as shown in Figure 14 have noisy responses with Newmark that are damped with generalized- α for all investigated coordinates and all used design variables. The same behavior also continues with the sensitivity values for the maximum stresses in the crank and the rod shown in Figure 15.

To check the correctness of the results, the sensitivity values with the introduced method are compared to numerical sensitivity analysis with forward differencing. The results are in good agreement, with only small numerical differences. In fact, there are no visible differences in Figures 12–15 with the finite differencing values lying under the respective analytical values.

Table 4 shows a comparison of the computational effort with different methods. The computational effort with generalized- α is about 30 % lower compared to the computational effort with Newmark- β . The main reason for that is the continuing vibrating behavior with Newmark- β requires more iteration steps of the nonlinear solver. The comparison of the sensitivity methods with numerical finite differencing and analytical direct differentiation shows the high efficiency of the analytical method. With the four shown design variables,

as shown in (63), an acceleration of about $S \approx 4$ of the analytical method compared to the numerical method is obtained. The additional computational effort of analytical sensitivity analysis is just a fraction of the computational effort of the primal analysis. This is because of the simplification shown in (59), which also allows one to reuse the Jacobian matrix of the residual built in the primal analysis for the sensitivity analysis.

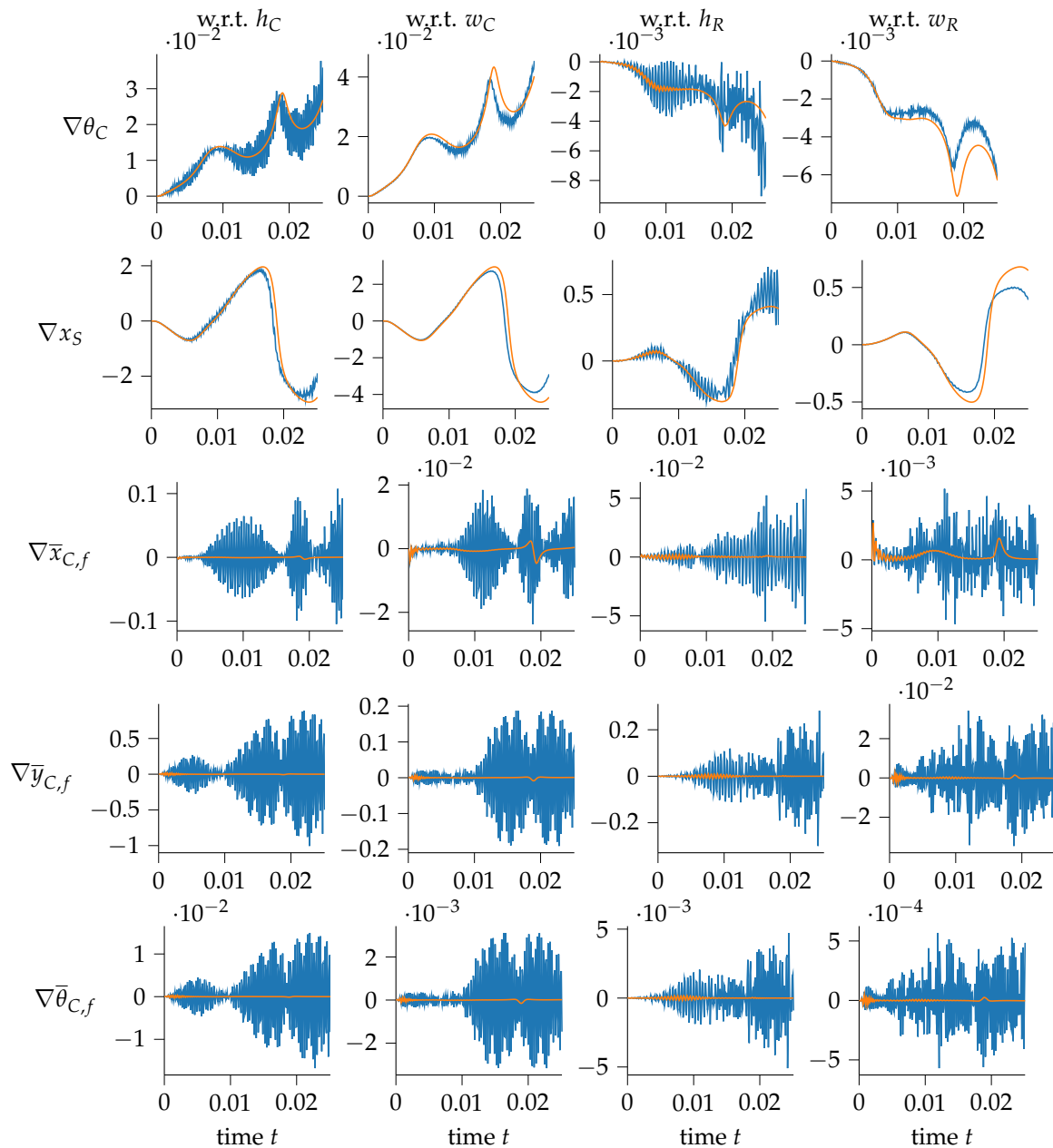


Figure 12. Position sensitivities ∇q of the slider-crank mechanism with respect to h_C, w_C, h_R and w_R (values in MPa system of units, i.e., s, mm, $\frac{\text{mm}}{\text{s}}, \frac{\text{mm}}{\text{s}^2}$ for translations and rad, $\frac{\text{rad}}{\text{s}}, \frac{\text{rad}}{\text{s}^2}$ for rotations) calculated with —Newmark- β and —generalized- α compared with those numerically calculated via finite differencing with —Newmark- β and —generalized- α (undermost lines). Note: there is no visible difference between finite differencing and analytical sensitivities.

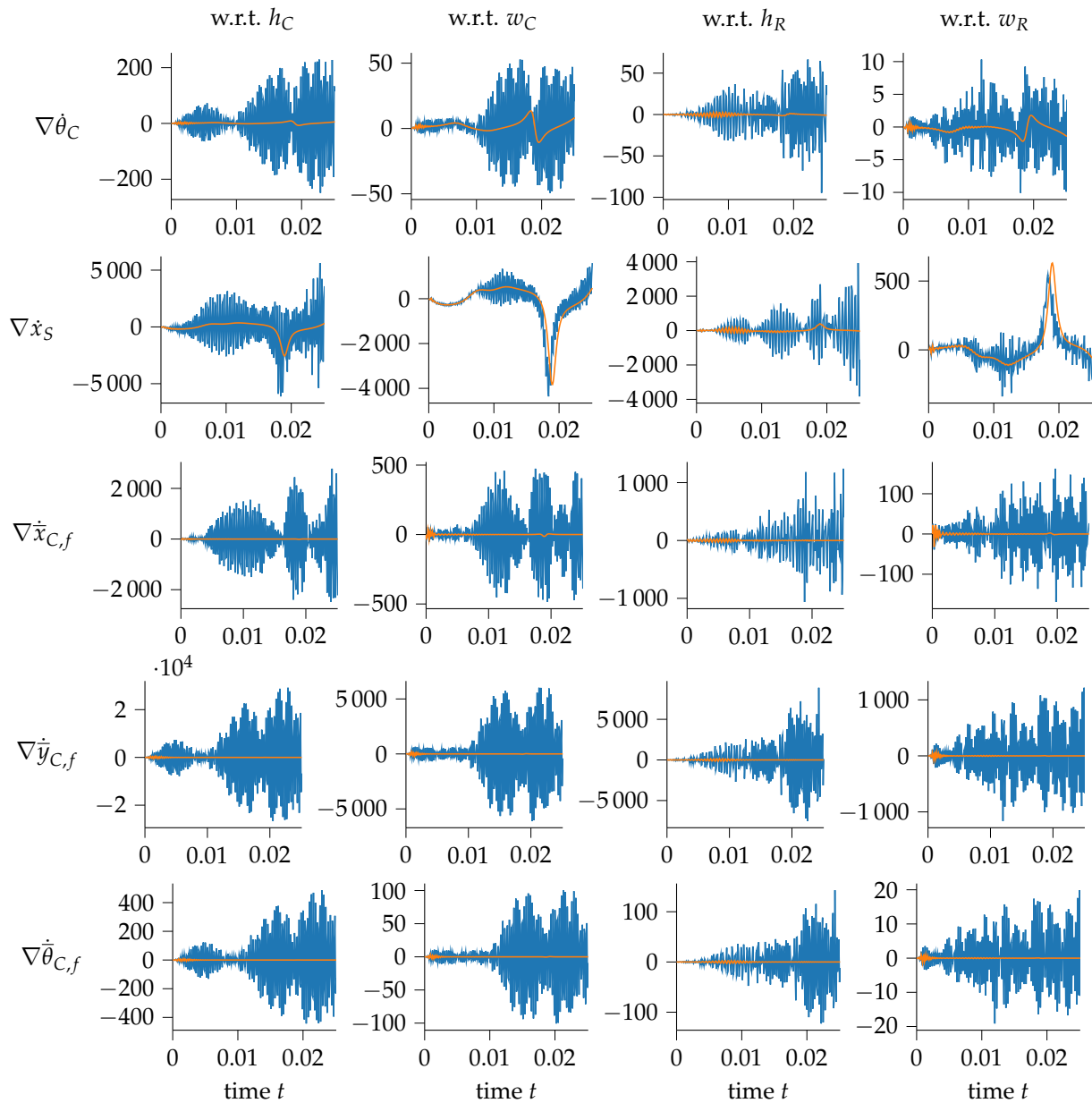


Figure 13. Velocity sensitivities $\nabla\dot{q}$ of the slider–crank mechanism with respect to h_C, w_C, h_R and w_R (values in MPa system of units, i.e., s, mm, $\frac{\text{mm}}{\text{s}}, \frac{\text{mm}}{\text{s}^2}$ for translations and rad, $\frac{\text{rad}}{\text{s}}, \frac{\text{rad}}{\text{s}^2}$ for rotations) calculated with —Newmark- β and —generalized- α compared with those calculated numerically via finite differencing with —Newmark- β and —generalized- α (undermost lines). Note: no visible difference between finite differencing and analytical sensitivities.

Table 4. Computational effort for the slider–crank mechanism.

Time Integration	Sensitivity Analysis	Calculation Time [mm:ss] ¹	Speedup ²
Newmark- β	None	03:44	–
Newmark- β	Finite differencing	18:12	–
Newmark- β	Analytical	04:29	4.0595
Generalized- α	None	02:42	–
Generalized- α	Finite differencing	13:22	–
Generalized- α	Analytical	03:30	3.8190

¹ Intel Core i7-8700 CPU 3.20 GHz, ² speedup of analytical sensitivity analysis with respect to finite differencing.

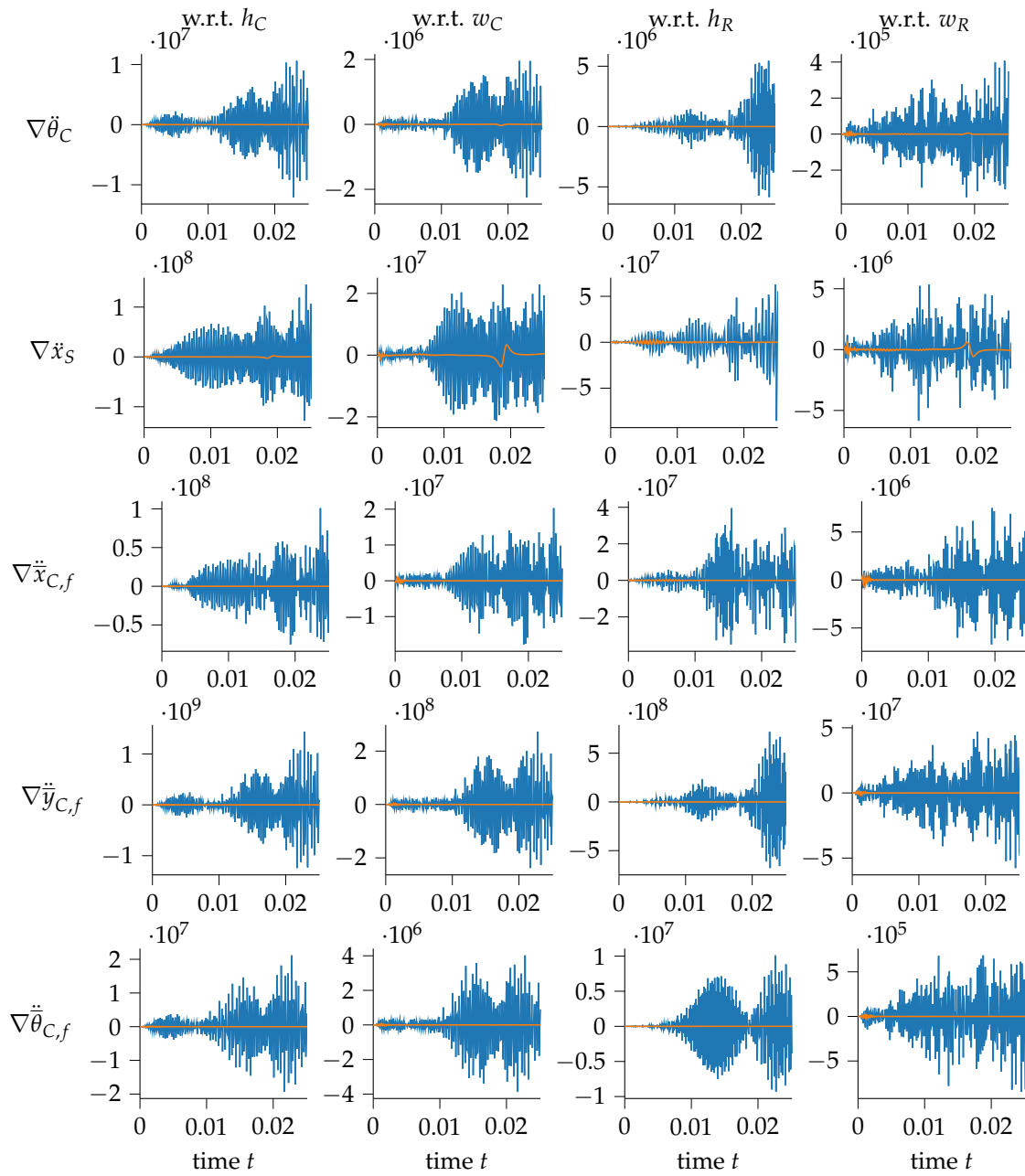


Figure 14. Acceleration sensitivities $\nabla\ddot{q}$ of the slider–crank mechanism with respect to h_C , w_C , h_R and w_R (values in MPa system of units, i.e., s, mm, $\frac{\text{mm}}{\text{s}}$, $\frac{\text{mm}}{\text{s}^2}$ for translations and rad, $\frac{\text{rad}}{\text{s}}$, $\frac{\text{rad}}{\text{s}^2}$ for rotations) calculated with —Newmark- β and —generalized- α compared with those calculated numerically via finite differencing with —Newmark- β and —generalized- α (undermost lines). Note: there is no visible difference between finite differencing and analytical sensitivities.

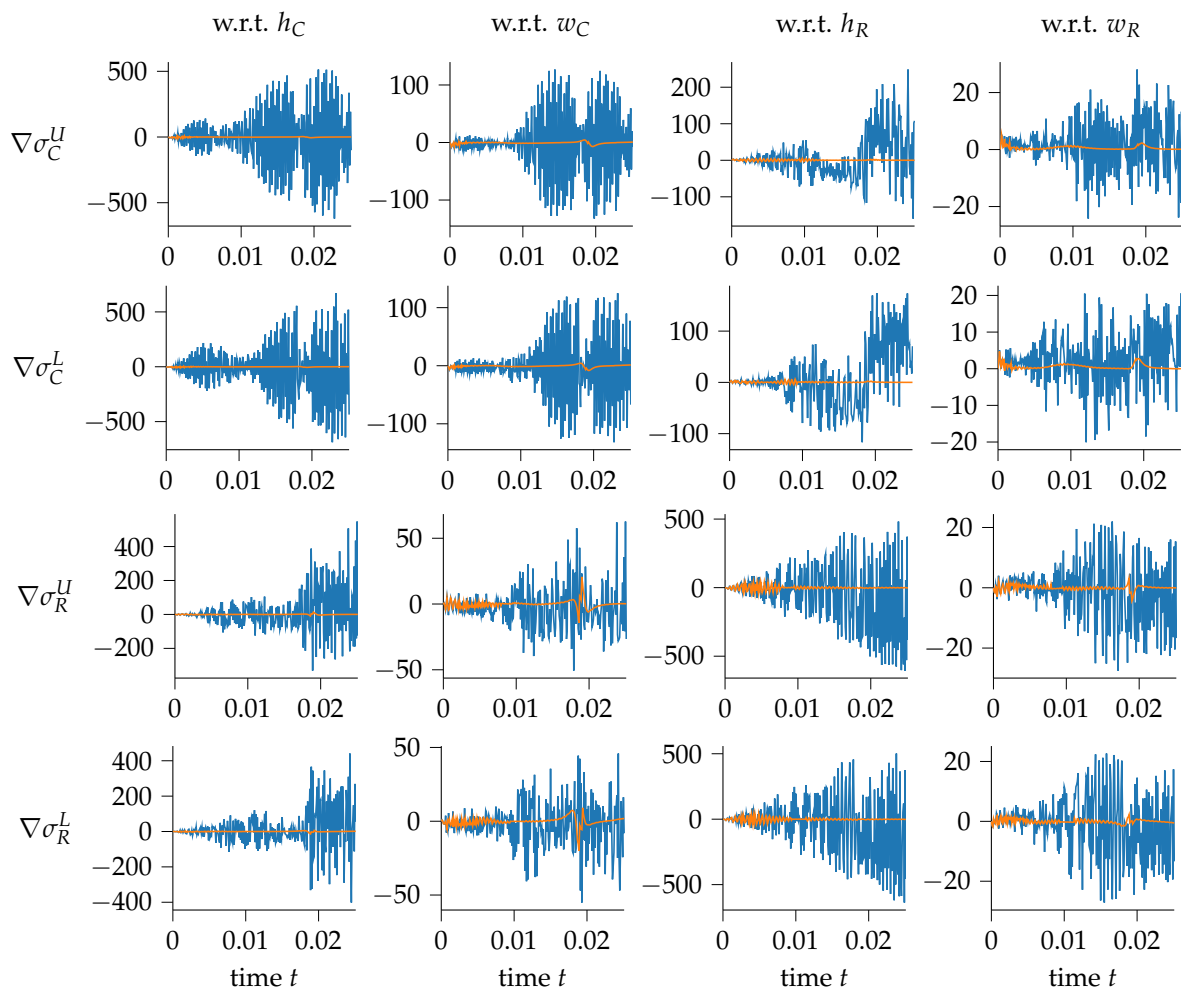


Figure 15. Maximum stress sensitivities on the upper and lower fibers of the crank and the rod (values in MPa system of units, i.e., s, mm, $\frac{\text{mm}}{\text{s}}$, $\frac{\text{mm}}{\text{s}^2}$ for translations and rad, $\frac{\text{rad}}{\text{s}}$, $\frac{\text{rad}}{\text{s}^2}$ for rotations) calculated with —Newmark- β and —generalized- α compared with those calculated numerically via finite differencing with —Newmark- β and —generalized- α (undermost lines). Note: there is no visible difference between finite differencing and analytical sensitivities.

4. Conclusions

This work develops the analytical sensitivity analysis for the α -family of time integration, which has a single implementation for several common time integration schemes. This therefore has consequences for gradient-based optimization frameworks for dynamic systems. Analytical sensitivity analysis is crucial for efficient design optimization, parameter fitting and uncertainty analysis. In this work, the direct differentiation of the generalized- α time integration is derived and implemented for linear structural dynamics and flexible multibody dynamics, the latter of which requires a nonlinear solution routine. All equations needed for implementation are provided and extendable to further governing equations. The beneficial effects of the step-size-dependent numerical damping provided by the generalized- α method can clearly be seen in the comparison of the results with Newmark- β . The numerical noise of the responses and sensitivities can lead to convergence problems when using gradient-based optimization algorithms. This can be avoided when using the generalized- α time integration and its sensitivities as developed herein.

If the functions of interest are nonsmooth, discontinuous and bifurcated, non-gradient-based methods can be successfully used, e.g., design cases with crash analysis. Such cases can be solved with gradient-free optimization algorithms (e.g., [75–77]) and approximation methods (e.g., [78–80]).

The system responses with respect to the design variables are indeed often smooth, continuous and unbifurcated and thus enabling gradient-based methods where efficient and accurate sensitivity analysis is critical for both effectiveness and efficiency. This is shown above with linear structural dynamics and flexible multibody dynamics. The method and equations developed herein are of general nature and are extendable to further formulations and applicable to larger problems, both of larger number of degrees of freedom and number of design variables. Future investigations are needed on these large-scale problems to assess the acceleration and include a comparison with the adjoint variable method.

With the above study and introduction of the direct differentiation of the α -family of time integrators, the authors hope to provide the readers as design engineers a tool in the pursuit of optimal structures and mechanisms under dynamic loading.

Author Contributions: Conceptualization, E.W. and V.G.; methodology, E.W. and V.G.; software, E.W. and V.G.; validation, E.W. and V.G.; formal analysis, E.W. and V.G.; investigation, E.W. and V.G.; resources, E.W. and V.G.; data curation, E.W. and V.G.; writing—original draft preparation, E.W. and V.G.; writing—review and editing, E.W. and V.G.; visualization, E.W. and V.G.; supervision, E.W.; project administration, E.W.; funding acquisition, E.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the project CRC 2017 TN2091 *doloMULTI Design of Lightweight Optimized structures and systems under MULTIdisciplinary considerations through integration of MULTImultibody dynamics in a MULTIpysics framework*, funded by the Free University of Bozen-Bolzano.

Data Availability Statement: Data available upon request.

Conflicts of Interest: The authors declare no conflicts of interest. The numerical studies by the first author were carried out during his previous position at the Free University of Bozen-Bolzano.

Appendix A. Derivation of Effective Equations of Motion for Structural Dynamics

Appendix A.1. Primal Analysis

The base for the α -family of integration methods is the equation of motion which for linear structural dynamics which is given by

$$\underline{m}\ddot{\underline{q}}_{n+1-\alpha_m} + \underline{d}\dot{\underline{q}}_{n+1-\alpha_f} + \underline{k}\underline{q}_{n+1-\alpha_f} = \underline{E}_{n+1-\alpha_f}, \quad (\text{A1})$$

and the intermediate values shown in Equations (3)–(5). Inserting the intermediate values into the equation of motion leads to

$$\underline{m}\left((1-\alpha_m)\ddot{\underline{q}}_{n+1} + \alpha_m\ddot{\underline{q}}_n\right) + \underline{d}\left((1-\alpha_f)\dot{\underline{q}}_{n+1} + \alpha_f\dot{\underline{q}}_n\right) + \underline{k}\left((1-\alpha_f)\underline{q}_{n+1} + \alpha_f\underline{q}_n\right) = (1-\alpha_f)\underline{E}_{n+1} + \alpha_f\underline{E}_n. \quad (\text{A2})$$

With Newmark's equation, (1) and (2), this becomes

$$\begin{aligned} &\underline{m}\left((1-\alpha_m)\ddot{\underline{q}}_{n+1} + \alpha_m\ddot{\underline{q}}_n\right) + \underline{d}\left((1-\alpha_f)\left(\dot{\underline{q}}_{\text{pred}} + \gamma\Delta t\dot{\underline{q}}_{n+1}\right) + \alpha_f\dot{\underline{q}}_n\right) + \\ &\quad + \underline{k}\left((1-\alpha_f)\left(\underline{q}_{\text{pred}} + \beta\Delta t^2\ddot{\underline{q}}_{n+1}\right) + \alpha_f\underline{q}_n\right) = (1-\alpha_f)\underline{E}_{n+1} + \alpha_f\underline{E}_n. \end{aligned} \quad (\text{A3})$$

This can be rearranged in the following form

$$\begin{aligned} &\left((1-\alpha_m)\underline{m} + (1-\alpha_f)\gamma\Delta t\underline{d} + (1-\alpha_f)\beta\Delta t^2\underline{k}\right)\ddot{\underline{q}}_{n+1} = \\ &\quad (1-\alpha_f)\underline{E}_{n+1} + \alpha_f\underline{E}_n - \alpha_m\underline{m}\ddot{\underline{q}}_n - \underline{d}\left((1-\alpha_f)\dot{\underline{q}}_{\text{pred}} + \alpha_f\dot{\underline{q}}_n\right) - \underline{k}\left((1-\alpha_f)\underline{q}_{\text{pred}} + \alpha_f\underline{q}_n\right). \end{aligned} \quad (\text{A4})$$

Therefore, the effective equation of motion is defined by

$$\underline{m}_{\text{eff}}\ddot{\underline{q}}_{n+1} = \underline{E}_{\text{eff}}, \quad (\text{A5})$$

where

$$\underline{\underline{m}}_{\text{eff}} = (1 - \alpha_m)\underline{\underline{m}} + (1 - \alpha_f)\gamma\Delta t\underline{\underline{d}} + (1 - \alpha_f)\beta\Delta t^2\underline{\underline{k}}, \tag{A6}$$

$$\underline{\underline{F}}_{\text{eff}} = (1 - \alpha_f)\underline{\underline{F}}_{n+1} + \alpha_f\underline{\underline{F}}_n - \alpha_m\underline{\underline{m}}\underline{\underline{\ddot{q}}}_n - \underline{\underline{d}}\left((1 - \alpha_f)\underline{\underline{\dot{q}}}_{\text{pred}} + \alpha_f\underline{\underline{\dot{q}}}_n\right) - \underline{\underline{k}}\left((1 - \alpha_f)\underline{\underline{q}}_{\text{pred}} + \alpha_f\underline{\underline{q}}_n\right). \tag{A7}$$

Appendix A.2. Sensitivity Analysis

The governing equation for sensitivities is the first derivative of the equation of motion,

$$\underline{\underline{\nabla m}}\underline{\underline{\ddot{q}}}_{n+1-\alpha_m} + \underline{\underline{m}}\underline{\underline{\nabla \ddot{q}}}_{n+1-\alpha_m} + \underline{\underline{\nabla d}}\underline{\underline{\dot{q}}}_{n+1-\alpha_f} + \underline{\underline{d}}\underline{\underline{\nabla \dot{q}}}_{n+1-\alpha_f} + \underline{\underline{\nabla k}}\underline{\underline{q}}_{n+1-\alpha_f} + \underline{\underline{k}}\underline{\underline{\nabla q}}_{n+1-\alpha_f} = \underline{\underline{\nabla F}}_{n+1-\alpha_f}, \tag{A8}$$

where we introduce the intermediate sensitivity values (10)–(12),

$$\begin{aligned} &\underline{\underline{\nabla m}}\left((1 - \alpha_m)\underline{\underline{\ddot{q}}}_{n+1} + \alpha_m\underline{\underline{\ddot{q}}}_n\right) + \underline{\underline{m}}\left((1 - \alpha_m)\underline{\underline{\nabla \ddot{q}}}_{n+1} + \alpha_m\underline{\underline{\nabla \ddot{q}}}_n\right) + \\ &+ \underline{\underline{\nabla d}}\left((1 - \alpha_f)\underline{\underline{\dot{q}}}_{n+1} + \alpha_f\underline{\underline{\dot{q}}}_n\right) + \underline{\underline{d}}\left((1 - \alpha_f)\underline{\underline{\nabla \dot{q}}}_{n+1} + \alpha_f\underline{\underline{\nabla \dot{q}}}_n\right) + \\ &+ \underline{\underline{\nabla k}}\left((1 - \alpha_f)\underline{\underline{q}}_{n+1} + \alpha_f\underline{\underline{q}}_n\right) + \underline{\underline{k}}\left((1 - \alpha_f)\underline{\underline{\nabla q}}_{n+1} + \alpha_f\underline{\underline{\nabla q}}_n\right) = \\ &\hspace{15em} (1 - \alpha_f)\underline{\underline{\nabla F}}_{n+1} + \alpha_f\underline{\underline{\nabla F}}_n, \end{aligned} \tag{A9}$$

and which we update with Newmark’s sensitivity Equations (16) and (17)

$$\begin{aligned} &\underline{\underline{\nabla m}}\left((1 - \alpha_m)\underline{\underline{\ddot{q}}}_{n+1} + \alpha_m\underline{\underline{\ddot{q}}}_n\right) + \underline{\underline{m}}\left((1 - \alpha_m)\underline{\underline{\nabla \ddot{q}}}_{n+1} + \alpha_m\underline{\underline{\nabla \ddot{q}}}_n\right) + \\ &+ \underline{\underline{\nabla d}}\left((1 - \alpha_f)\underline{\underline{\dot{q}}}_{n+1} + \alpha_f\underline{\underline{\dot{q}}}_n\right) + \underline{\underline{d}}\left((1 - \alpha_f)\left(\underline{\underline{\nabla \dot{q}}}_{\text{pred}} + \gamma\Delta t\underline{\underline{\nabla \ddot{q}}}_{n+1}\right) + \alpha_f\underline{\underline{\nabla \dot{q}}}_n\right) + \\ &+ \underline{\underline{\nabla k}}\left((1 - \alpha_f)\underline{\underline{q}}_{n+1} + \alpha_f\underline{\underline{q}}_n\right) + \underline{\underline{k}}\left((1 - \alpha_f)\left(\underline{\underline{\nabla q}}_{\text{pred}} + \beta\Delta t^2\underline{\underline{\nabla \ddot{q}}}_{n+1}\right) + \alpha_f\underline{\underline{\nabla q}}_n\right) = \\ &\hspace{15em} (1 - \alpha_f)\underline{\underline{\nabla F}}_{n+1} + \alpha_f\underline{\underline{\nabla F}}_n. \end{aligned} \tag{A10}$$

This equation can be rearranged to the following form,

$$\begin{aligned} &\left((1 - \alpha_m)\underline{\underline{m}} + (1 - \alpha_f)\gamma\Delta t\underline{\underline{d}} + (1 - \alpha_f)\beta\Delta t^2\underline{\underline{k}}\right)\underline{\underline{\nabla \ddot{q}}}_{n+1} = \\ &\hspace{4em} (1 - \alpha_f)\underline{\underline{\nabla F}}_{n+1} + \alpha_f\underline{\underline{\nabla F}}_n - \underline{\underline{\nabla m}}\left((1 - \alpha_m)\underline{\underline{\ddot{q}}}_{n+1} + \alpha_m\underline{\underline{\ddot{q}}}_n\right) - \alpha_m\underline{\underline{m}}\underline{\underline{\nabla \ddot{q}}}_n + \\ &\hspace{4em} - \underline{\underline{\nabla d}}\left((1 - \alpha_f)\underline{\underline{\dot{q}}}_{n+1} + \alpha_f\underline{\underline{\dot{q}}}_n\right) - \underline{\underline{d}}\left((1 - \alpha_f)\underline{\underline{\nabla \dot{q}}}_{\text{pred}} + \alpha_f\underline{\underline{\nabla \dot{q}}}_n\right) + \\ &\hspace{4em} - \underline{\underline{\nabla k}}\left((1 - \alpha_f)\underline{\underline{q}}_{n+1} + \alpha_f\underline{\underline{q}}_n\right) - \underline{\underline{k}}\left((1 - \alpha_f)\underline{\underline{\nabla q}}_{\text{pred}} + \alpha_f\underline{\underline{\nabla q}}_n\right). \end{aligned} \tag{A11}$$

Therefore, the effective equation of motion for sensitivities is defined by

$$\underline{\underline{m}}_{\text{eff}}\underline{\underline{\nabla \ddot{q}}}_{n+1} = \underline{\underline{F}}_{\text{pseudo}}, \tag{A12}$$

where $\underline{\underline{m}}_{\text{eff}}$ is given by Equation (A6) and the pseudo load is defined by

$$\begin{aligned} \underline{\underline{F}}_{\text{pseudo}} = & (1 - \alpha_f)\underline{\underline{\nabla F}}_{n+1} + \alpha_f\underline{\underline{\nabla F}}_n - \underline{\underline{\nabla m}}\left((1 - \alpha_m)\underline{\underline{\ddot{q}}}_{n+1} + \alpha_m\underline{\underline{\ddot{q}}}_n\right) - \alpha_m\underline{\underline{m}}\underline{\underline{\nabla \ddot{q}}}_n + \\ & - \underline{\underline{\nabla d}}\left((1 - \alpha_f)\underline{\underline{\dot{q}}}_{n+1} + \alpha_f\underline{\underline{\dot{q}}}_n\right) - \underline{\underline{d}}\left((1 - \alpha_f)\underline{\underline{\nabla \dot{q}}}_{\text{pred}} + \alpha_f\underline{\underline{\nabla \dot{q}}}_n\right) + \\ & - \underline{\underline{\nabla k}}\left((1 - \alpha_f)\underline{\underline{q}}_{n+1} + \alpha_f\underline{\underline{q}}_n\right) - \underline{\underline{k}}\left((1 - \alpha_f)\underline{\underline{\nabla q}}_{\text{pred}} + \alpha_f\underline{\underline{\nabla q}}_n\right). \end{aligned} \tag{A13}$$

Appendix B. Derivation of Effective Equations of Motion for Flexible Multibody Dynamics

Appendix B.1. Primal Analysis

The base of the generalized- α method for index-1 differential algebraic equations of a flexible multibody system is given by

$$\left(\underline{\underline{m}}\ddot{\underline{q}}\right)_{n+1-\alpha_m} + \left(\underline{\underline{d}}\dot{\underline{q}}\right)_{n+1-\alpha_f} + \left(\underline{\underline{k}}\underline{q}\right)_{n+1-\alpha_f} + \left(\underline{\underline{J}}_{\Phi}^T \underline{\underline{\lambda}}\right)_{n+1-\alpha_m} = \underline{E}_{\text{ext},n+1-\alpha_f} + \underline{E}_{v,n+1-\alpha_f}, \quad (\text{A14})$$

$$\left(\underline{\underline{J}}_{\Phi} \dot{\underline{q}}\right)_{n+1-\alpha_m} = \underline{E}_{c,n+1-\alpha_f}. \quad (\text{A15})$$

The intermediate point approximations of the force terms acting on the flexible multibody system are given by

$$\left(\underline{\underline{m}}\ddot{\underline{q}}\right)_{n+1-\alpha_m} = (1 - \alpha_m)\underline{\underline{m}}_{n+1}\ddot{\underline{q}}_{n+1} + \alpha_m\underline{\underline{m}}_n\ddot{\underline{q}}_n, \quad (\text{A16})$$

$$\left(\underline{\underline{d}}\dot{\underline{q}}\right)_{n+1-\alpha_f} = (1 - \alpha_f)\underline{\underline{d}}_{n+1}\dot{\underline{q}}_{n+1} + \alpha_f\underline{\underline{d}}_n\dot{\underline{q}}_n, \quad (\text{A17})$$

$$\left(\underline{\underline{k}}\underline{q}\right)_{n+1-\alpha_f} = (1 - \alpha_f)\underline{\underline{k}}_{n+1}\underline{q}_{n+1} + \alpha_f\underline{\underline{k}}_n\underline{q}_n, \quad (\text{A18})$$

$$\left(\underline{\underline{J}}_{\Phi}^T \underline{\underline{\lambda}}\right)_{n+1-\alpha_m} = (1 - \alpha_m)\underline{\underline{J}}_{\Phi,n+1}^T \underline{\underline{\lambda}}_{n+1} + \alpha_m\underline{\underline{J}}_{\Phi,n}^T \underline{\underline{\lambda}}_n, \quad (\text{A19})$$

$$\left(\underline{\underline{J}}_{\Phi} \dot{\underline{q}}\right)_{n+1-\alpha_m} = (1 - \alpha_m)\underline{\underline{J}}_{\Phi,n+1} \dot{\underline{q}}_{n+1} + \alpha_m\underline{\underline{J}}_{\Phi,n} \dot{\underline{q}}_n, \quad (\text{A20})$$

$$\underline{E}_{\text{ext},n+1-\alpha_f} = (1 - \alpha_f)\underline{E}_{\text{ext},n+1} + \alpha_f\underline{E}_{\text{ext},n}, \quad (\text{A21})$$

$$\underline{E}_{v,n+1-\alpha_f} = (1 - \alpha_f)\underline{E}_{v,n+1} + \alpha_f\underline{E}_{v,n}, \quad (\text{A22})$$

$$\underline{E}_{c,n+1-\alpha_f} = (1 - \alpha_f)\underline{E}_{c,n+1} + \alpha_f\underline{E}_{c,n}. \quad (\text{A23})$$

Applying these linear approximations of the generalized- α method to the equations of motion leads to

$$\begin{aligned} & (1 - \alpha_m)\underline{\underline{m}}_{n+1}\ddot{\underline{q}}_{n+1} + \alpha_m\underline{\underline{m}}_n\ddot{\underline{q}}_n + \\ & + (1 - \alpha_f)\underline{\underline{d}}_{n+1}\dot{\underline{q}}_{n+1} + \alpha_f\underline{\underline{d}}_n\dot{\underline{q}}_n + \\ & + (1 - \alpha_f)\underline{\underline{k}}_{n+1}\underline{q}_{n+1} + \alpha_f\underline{\underline{k}}_n\underline{q}_n + \\ & + (1 - \alpha_m)\underline{\underline{J}}_{\Phi,n+1}^T \underline{\underline{\lambda}}_{n+1} + \alpha_m\underline{\underline{J}}_{\Phi,n}^T \underline{\underline{\lambda}}_n = (1 - \alpha_f)\underline{E}_{\text{ext},n+1} + \alpha_f\underline{E}_{\text{ext},n} + \\ & + (1 - \alpha_f)\underline{E}_{v,n+1} + \alpha_f\underline{E}_{v,n}, \end{aligned} \quad (\text{A24})$$

$$(1 - \alpha_m)\underline{\underline{J}}_{\Phi,n+1} \dot{\underline{q}}_{n+1} + \alpha_m\underline{\underline{J}}_{\Phi,n} \dot{\underline{q}}_n = (1 - \alpha_f)\underline{E}_{c,n+1} + \alpha_f\underline{E}_{c,n}. \quad (\text{A25})$$

Inserting Newmark's Equations (1) and (2) gives

$$\begin{aligned} & (1 - \alpha_m)\underline{\underline{m}}_{n+1}\ddot{\underline{q}}_{n+1} + \alpha_m\underline{\underline{m}}_n\ddot{\underline{q}}_n + \\ & + (1 - \alpha_f)\underline{\underline{d}}_{n+1}\left(\dot{\underline{q}}_{\text{pred}} + \gamma\Delta t\ddot{\underline{q}}_{n+1}\right) + \alpha_f\underline{\underline{d}}_n\dot{\underline{q}}_n + \\ & + (1 - \alpha_f)\underline{\underline{k}}_{n+1}\left(\underline{q}_{\text{pred}} + \beta\Delta t^2\ddot{\underline{q}}_{n+1}\right) + \alpha_f\underline{\underline{k}}_n\underline{q}_n + \\ & + (1 - \alpha_m)\underline{\underline{J}}_{\Phi,n+1}^T \underline{\underline{\lambda}}_{n+1} + \alpha_m\underline{\underline{J}}_{\Phi,n}^T \underline{\underline{\lambda}}_n = (1 - \alpha_f)\underline{E}_{\text{ext},n+1} + \alpha_f\underline{E}_{\text{ext},n} + \\ & + (1 - \alpha_f)\underline{E}_{v,n+1} + \alpha_f\underline{E}_{v,n}, \end{aligned} \quad (\text{A26})$$

$$(1 - \alpha_m)\underline{\underline{J}}_{\Phi,n+1} \dot{\underline{q}}_{n+1} + \alpha_m\underline{\underline{J}}_{\Phi,n} \dot{\underline{q}}_n = (1 - \alpha_f)\underline{E}_{c,n+1} + \alpha_f\underline{E}_{c,n}. \quad (\text{A27})$$

which can be rewritten to

$$\begin{aligned} & \left((1 - \alpha_m) \underline{\underline{m}} + (1 - \alpha_f) \gamma \Delta t \underline{\underline{d}} + (1 - \alpha_f) \beta \Delta t^2 \underline{\underline{k}} \right) \underline{\underline{\dot{q}}}_{n+1} + \\ & + (1 - \alpha_m) \underline{\underline{J}}_{\Phi, n+1}^T \underline{\underline{\lambda}}_{n+1} = (1 - \alpha_f) E_{\text{ext}, n+1} + \alpha_f E_{\text{ext}, n} + \\ & + (1 - \alpha_f) E_{v, n+1} + \alpha_f E_{v, n} - \alpha_m \underline{\underline{m}} \underline{\underline{\ddot{q}}}_n + \\ & - (1 - \alpha_f) \underline{\underline{d}}_{n+1} \underline{\underline{\dot{q}}}_{\text{pred}} - \alpha_f \underline{\underline{d}}_n \underline{\underline{\dot{q}}}_n + \\ & - (1 - \alpha_f) \underline{\underline{k}}_{n+1} \underline{\underline{q}}_{\text{pred}} - \alpha_f \underline{\underline{k}}_n \underline{\underline{q}}_n + \\ & - \alpha_m \underline{\underline{J}}_{\Phi, n}^T \underline{\underline{\lambda}}_n, \end{aligned} \quad (\text{A28})$$

$$(1 - \alpha_m) \underline{\underline{J}}_{\Phi, n+1} \underline{\underline{\dot{q}}}_{n+1} = (1 - \alpha_f) E_{c, n+1} + \alpha_f E_{c, n} - \alpha_m \underline{\underline{J}}_{\Phi, n} \underline{\underline{\dot{q}}}_n. \quad (\text{A29})$$

The equations of motion of the effective system are given by

$$\begin{bmatrix} \underline{\underline{m}}_{\text{eff}} & \underline{\underline{J}}_{\Phi, \text{eff}}^T \\ \underline{\underline{J}}_{\Phi, \text{eff}} & \underline{\underline{0}} \end{bmatrix} \begin{bmatrix} \underline{\underline{\dot{q}}}_{n+1} \\ \underline{\underline{\lambda}}_{n+1} \end{bmatrix} = \begin{bmatrix} \underline{\underline{F}}_{a, \text{eff}} \\ \underline{\underline{F}}_{c, \text{eff}} \end{bmatrix}, \quad (\text{A30})$$

where

$$\underline{\underline{m}}_{\text{eff}} = (1 - \alpha_m) \underline{\underline{m}}_{n+1} + (1 - \alpha_f) \gamma \Delta t \underline{\underline{d}}_{n+1} + (1 - \alpha_f) \beta \Delta t^2 \underline{\underline{k}}_{n+1}, \quad (\text{A31})$$

$$\underline{\underline{J}}_{\Phi, \text{eff}} = (1 - \alpha_m) \underline{\underline{J}}_{\Phi, n+1}, \quad (\text{A32})$$

$$\begin{aligned} \underline{\underline{F}}_{a, \text{eff}} &= (1 - \alpha_f) E_{\text{ext}, n+1} + \alpha_f E_{\text{ext}, n} + (1 - \alpha_f) E_{v, n+1} + \alpha_f E_{v, n} - \alpha_m \underline{\underline{m}} \underline{\underline{\ddot{q}}}_n + \\ & - (1 - \alpha_f) \underline{\underline{d}}_{n+1} \underline{\underline{\dot{q}}}_{\text{pred}} - \alpha_f \underline{\underline{d}}_n \underline{\underline{\dot{q}}}_n - (1 - \alpha_f) \underline{\underline{k}}_{n+1} \underline{\underline{q}}_{\text{pred}} - \alpha_f \underline{\underline{k}}_n \underline{\underline{q}}_n - \alpha_m \underline{\underline{J}}_{\Phi, n}^T \underline{\underline{\lambda}}_n, \end{aligned} \quad (\text{A33})$$

$$\underline{\underline{F}}_{c, \text{eff}} = (1 - \alpha_f) E_{c, n+1} + \alpha_f E_{c, n} - \alpha_m \underline{\underline{J}}_{\Phi, n} \underline{\underline{\dot{q}}}_n. \quad (\text{A34})$$

Appendix B.2. Nonlinear Solver

The base for the nonlinear solver are the residual equations of the flexible multibody system, given by

$$\begin{aligned} \underline{\underline{R}}_{1, n+1-\alpha_f} &= (1 - \alpha_m) \underline{\underline{m}}_{n+1} \underline{\underline{\dot{q}}}_{n+1} + \alpha_m \underline{\underline{m}}_n \underline{\underline{\dot{q}}}_n + (1 - \alpha_f) \underline{\underline{d}}_{n+1} \underline{\underline{\dot{q}}}_{n+1} + \alpha_f \underline{\underline{d}}_n \underline{\underline{\dot{q}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{k}}_{n+1} \underline{\underline{q}}_{n+1} + \alpha_f \underline{\underline{k}}_n \underline{\underline{q}}_n + (1 - \alpha_m) \underline{\underline{J}}_{\Phi, n+1}^T \underline{\underline{\lambda}}_{n+1} + \alpha_m \underline{\underline{J}}_{\Phi, n}^T \underline{\underline{\lambda}}_n + \\ & - (1 - \alpha_f) E_{\text{ext}, n+1} - \alpha_f E_{\text{ext}, n} - (1 - \alpha_f) E_{v, n+1} - \alpha_f E_{v, n}, \end{aligned} \quad (\text{A35})$$

$$\underline{\underline{R}}_{2, n+1-\alpha_f} = (1 - \alpha_m) \underline{\underline{J}}_{\Phi, n+1} \underline{\underline{\dot{q}}}_{n+1} + \alpha_m \underline{\underline{J}}_{\Phi, n} \underline{\underline{\dot{q}}}_n - (1 - \alpha_f) E_{c, n+1} - \alpha_f E_{c, n}. \quad (\text{A36})$$

The Newton–Raphson method is used here

$$\begin{bmatrix} \frac{\partial \underline{\underline{R}}_{1, n+1-\alpha_f}}{\partial \underline{\underline{\dot{q}}}_{n+1}} & \frac{\partial \underline{\underline{R}}_{1, n+1-\alpha_f}}{\partial \underline{\underline{\lambda}}_{n+1}} \\ \frac{\partial \underline{\underline{R}}_{2, n+1-\alpha_f}}{\partial \underline{\underline{\dot{q}}}_{n+1}} & \frac{\partial \underline{\underline{R}}_{2, n+1-\alpha_f}}{\partial \underline{\underline{\lambda}}_{n+1}} \end{bmatrix} \begin{bmatrix} \Delta \underline{\underline{\dot{q}}}_{n+1} \\ \Delta \underline{\underline{\lambda}}_{n+1} \end{bmatrix} + \begin{bmatrix} \underline{\underline{R}}_{1, n+1-\alpha_f} \\ \underline{\underline{R}}_{2, n+1-\alpha_f} \end{bmatrix} = \underline{\underline{0}}, \quad (\text{A37})$$

where the terms of the Jacobian matrix are given by

$$\begin{aligned}
\frac{\partial R_{1,n+1-\alpha_f}}{\partial \ddot{q}_{n+1}} &= (1 - \alpha_m) \frac{\partial \underline{m}_{n+1}}{\partial \ddot{q}_{n+1}} \ddot{q}_{n+1} + \alpha_m \frac{\partial \underline{m}_n}{\partial \ddot{q}_{n+1}} \ddot{q}_n + (1 - \alpha_m) \underline{m}_{n+1} \frac{\partial \ddot{q}_{n+1}}{\partial \ddot{q}_{n+1}} + \alpha_m \underline{m}_n \frac{\partial \ddot{q}_n}{\partial \ddot{q}_{n+1}} + \\
&+ (1 - \alpha_f) \frac{\partial \underline{d}_{n+1}}{\partial \ddot{q}_{n+1}} \dot{q}_{n+1} + \alpha_f \frac{\partial \underline{d}_n}{\partial \ddot{q}_{n+1}} \dot{q}_n + (1 - \alpha_f) \underline{d}_{n+1} \frac{\partial \dot{q}_{n+1}}{\partial \ddot{q}_{n+1}} + \alpha_f \underline{d}_n \frac{\partial \dot{q}_n}{\partial \ddot{q}_{n+1}} + \\
&+ (1 - \alpha_f) \frac{\partial \underline{k}_{n+1}}{\partial \ddot{q}_{n+1}} q_{n+1} + \alpha_f \frac{\partial \underline{k}_n}{\partial \ddot{q}_{n+1}} q_n + (1 - \alpha_f) \underline{k}_{n+1} \frac{\partial q_{n+1}}{\partial \ddot{q}_{n+1}} + \alpha_f \underline{k}_n \frac{\partial q_n}{\partial \ddot{q}_{n+1}} + \\
&+ (1 - \alpha_m) \frac{\partial J_{\Phi,n+1}^T}{\partial \ddot{q}_{n+1}} \lambda_{n+1} + \alpha_m \frac{\partial J_{\Phi,n}^T}{\partial \ddot{q}_{n+1}} \lambda_n + (1 - \alpha_m) J_{\Phi,n+1}^T \frac{\partial \lambda_{n+1}}{\partial \ddot{q}_{n+1}} + \alpha_m J_{\Phi,n}^T \frac{\partial \lambda_n}{\partial \ddot{q}_{n+1}} + \\
&- (1 - \alpha_f) \frac{\partial F_{ext,n+1}}{\partial \ddot{q}_{n+1}} - \alpha_f \frac{\partial F_{ext,n}}{\partial \ddot{q}_{n+1}} - (1 - \alpha_f) \frac{\partial F_{v,n+1}}{\partial \ddot{q}_{n+1}} - \alpha_f \frac{\partial F_{v,n}}{\partial \ddot{q}_{n+1}}, \tag{A38}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial R_{2,n+1-\alpha_f}}{\partial \ddot{q}_{n+1}} &= (1 - \alpha_m) \frac{\partial J_{\Phi,n+1}}{\partial \ddot{q}_{n+1}} \ddot{q}_{n+1} + \alpha_m \frac{\partial J_{\Phi,n}}{\partial \ddot{q}_{n+1}} \ddot{q}_n + (1 - \alpha_m) J_{\Phi,n+1} \frac{\partial \ddot{q}_{n+1}}{\partial \ddot{q}_{n+1}} + \alpha_m J_{\Phi,n} \frac{\partial \ddot{q}_n}{\partial \ddot{q}_{n+1}} + \\
&- (1 - \alpha_f) \frac{\partial F_{c,n+1}}{\partial \ddot{q}_{n+1}} - \alpha_f \frac{\partial F_{c,n}}{\partial \ddot{q}_{n+1}}, \tag{A39}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial R_{1,n+1-\alpha_f}}{\partial \lambda_{n+1}} &= (1 - \alpha_m) \frac{\partial \underline{m}_{n+1}}{\partial \lambda_{n+1}} \ddot{q}_{n+1} + \alpha_m \frac{\partial \underline{m}_n}{\partial \lambda_{n+1}} \ddot{q}_n + (1 - \alpha_m) \underline{m}_{n+1} \frac{\partial \ddot{q}_{n+1}}{\partial \lambda_{n+1}} + \alpha_m \underline{m}_n \frac{\partial \ddot{q}_n}{\partial \lambda_{n+1}} + \\
&+ (1 - \alpha_f) \frac{\partial \underline{d}_{n+1}}{\partial \lambda_{n+1}} \dot{q}_{n+1} + \alpha_f \frac{\partial \underline{d}_n}{\partial \lambda_{n+1}} \dot{q}_n + (1 - \alpha_f) \underline{d}_{n+1} \frac{\partial \dot{q}_{n+1}}{\partial \lambda_{n+1}} + \alpha_f \underline{d}_n \frac{\partial \dot{q}_n}{\partial \lambda_{n+1}} + \\
&+ (1 - \alpha_f) \frac{\partial \underline{k}_{n+1}}{\partial \lambda_{n+1}} q_{n+1} + \alpha_f \frac{\partial \underline{k}_n}{\partial \lambda_{n+1}} q_n + (1 - \alpha_f) \underline{k}_{n+1} \frac{\partial q_{n+1}}{\partial \lambda_{n+1}} + \alpha_f \underline{k}_n \frac{\partial q_n}{\partial \lambda_{n+1}} + \\
&+ (1 - \alpha_m) \frac{\partial J_{\Phi,n+1}^T}{\partial \lambda_{n+1}} \lambda_{n+1} + \alpha_m \frac{\partial J_{\Phi,n}^T}{\partial \lambda_{n+1}} \lambda_n + (1 - \alpha_m) J_{\Phi,n+1}^T \frac{\partial \lambda_{n+1}}{\partial \lambda_{n+1}} + \alpha_m J_{\Phi,n}^T \frac{\partial \lambda_n}{\partial \lambda_{n+1}} + \\
&- (1 - \alpha_f) \frac{\partial F_{ext,n+1}}{\partial \lambda_{n+1}} - \alpha_f \frac{\partial F_{ext,n}}{\partial \lambda_{n+1}} - (1 - \alpha_f) \frac{\partial F_{v,n+1}}{\partial \lambda_{n+1}} - \alpha_f \frac{\partial F_{v,n}}{\partial \lambda_{n+1}}, \tag{A40}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial R_{2,n+1-\alpha_f}}{\partial \lambda_{n+1}} &= (1 - \alpha_m) \frac{\partial J_{\Phi,n+1}}{\partial \lambda_{n+1}} \ddot{q}_{n+1} + \alpha_m \frac{\partial J_{\Phi,n}}{\partial \lambda_{n+1}} \ddot{q}_n + (1 - \alpha_m) J_{\Phi,n+1} \frac{\partial \ddot{q}_{n+1}}{\partial \lambda_{n+1}} + \alpha_m J_{\Phi,n} \frac{\partial \ddot{q}_n}{\partial \lambda_{n+1}} + \\
&- (1 - \alpha_f) \frac{\partial F_{c,n+1}}{\partial \lambda_{n+1}} - \alpha_f \frac{\partial F_{c,n}}{\partial \lambda_{n+1}}. \tag{A41}
\end{aligned}$$

The derivatives of the system parameters of the previous time step with respect to the state variables of the next time step disappear,

$$\frac{\partial \underline{\circ}_n}{\partial \bullet_{n+1}} = \underline{0}, \quad \frac{\partial \underline{\circ}_n}{\partial \bullet_{n+1}} = \underline{0}. \tag{A42}$$

The dependencies of the system parameters are shown in (33) and (34), and applying the chain rule of differentiation leads to

$$\frac{\partial \underline{m}}{\partial \underline{\ddot{q}}} = \frac{\partial \underline{m}}{\partial \underline{q}} \frac{\partial \underline{q}}{\partial \underline{\ddot{q}}'} \quad (\text{A43})$$

$$\frac{\partial \underline{d}}{\partial \underline{\ddot{q}}} = \frac{\partial \underline{d}}{\partial \underline{q}} \frac{\partial \underline{q}}{\partial \underline{\ddot{q}}'} \quad (\text{A44})$$

$$\frac{\partial \underline{k}}{\partial \underline{\ddot{q}}} = \frac{\partial \underline{k}}{\partial \underline{q}} \frac{\partial \underline{q}}{\partial \underline{\ddot{q}}'} \quad (\text{A45})$$

$$\frac{\partial \underline{J}_\Phi}{\partial \underline{\ddot{q}}} = \frac{\partial \underline{J}_\Phi}{\partial \underline{q}} \frac{\partial \underline{q}}{\partial \underline{\ddot{q}}'} \quad (\text{A46})$$

$$\frac{\partial \underline{F}_{\text{ext}}}{\partial \underline{\ddot{q}}} = \frac{\partial \underline{F}_{\text{ext}}}{\partial \underline{q}} \frac{\partial \underline{q}}{\partial \underline{\ddot{q}}'} + \frac{\partial \underline{F}_{\text{ext}}}{\partial \underline{\dot{q}}} \frac{\partial \underline{\dot{q}}}{\partial \underline{\ddot{q}}'} \quad (\text{A47})$$

$$\frac{\partial \underline{F}_v}{\partial \underline{\ddot{q}}} = \frac{\partial \underline{F}_v}{\partial \underline{q}} \frac{\partial \underline{q}}{\partial \underline{\ddot{q}}'} + \frac{\partial \underline{F}_v}{\partial \underline{\dot{q}}} \frac{\partial \underline{\dot{q}}}{\partial \underline{\ddot{q}}'} \quad (\text{A48})$$

$$\frac{\partial \underline{0}}{\partial \underline{\lambda}} = \underline{0}. \quad (\text{A49})$$

$$\frac{\partial \underline{0}}{\partial \underline{\lambda}} = \underline{0}. \quad (\text{A50})$$

The derivatives of the state variables with respect to $\underline{\ddot{q}}_{n+1}$ and $\underline{\lambda}_{n+1}$ are given by

$$\begin{aligned} \frac{\partial \underline{q}_{n+1}}{\partial \underline{\ddot{q}}_{n+1}} &= \beta \Delta t^2 \underline{e}_r & \frac{\partial \underline{q}_{n+1}}{\partial \underline{\lambda}_{n+1}} &= \underline{0}, \\ \frac{\partial \underline{\dot{q}}_{n+1}}{\partial \underline{\ddot{q}}_{n+1}} &= \gamma \Delta t \underline{e}_r & \frac{\partial \underline{\dot{q}}_{n+1}}{\partial \underline{\lambda}_{n+1}} &= \underline{0}, \\ \frac{\partial \underline{\ddot{q}}_{n+1}}{\partial \underline{\ddot{q}}_{n+1}} &= \underline{e}_r & \frac{\partial \underline{\ddot{q}}_{n+1}}{\partial \underline{\lambda}_{n+1}} &= \underline{0}, \\ \frac{\partial \underline{\lambda}_{n+1}}{\partial \underline{\ddot{q}}_{n+1}} &= \underline{0}, & \frac{\partial \underline{\lambda}_{n+1}}{\partial \underline{\lambda}_{n+1}} &= \underline{e}_r. \end{aligned} \quad (\text{A51})$$

Applying Equation (A42), Equations (A43)–(A51) lead to the terms of the Jacobian matrix given by

$$\begin{aligned} \frac{\partial \underline{R}_{1,n+1-\alpha_f}}{\partial \underline{\ddot{q}}_{n+1}} &= (1 - \alpha_m) \left(\beta \Delta t^2 \frac{\partial \underline{m}_{n+1}}{\partial \underline{q}_{n+1}} \underline{\ddot{q}}_{n+1} + \underline{m}_{n+1} \right) + \\ &+ (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{d}_{n+1}}{\partial \underline{q}_{n+1}} \underline{\dot{q}}_{n+1} + \gamma \Delta t \underline{d}_{n+1} \right) + \\ &+ (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{k}_{n+1}}{\partial \underline{q}_{n+1}} \underline{q}_{n+1} + \beta \Delta t^2 \underline{k}_{n+1} \right) + \\ &+ (1 - \alpha_m) \beta \Delta t^2 \frac{\partial \underline{J}_{\Phi,n+1}^T}{\partial \underline{q}_{n+1}} \underline{\lambda}_{n+1} + \\ &- (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{F}_{\text{ext},n+1}}{\partial \underline{q}_{n+1}} + \gamma \Delta t \frac{\partial \underline{F}_{\text{ext},n+1}}{\partial \underline{\dot{q}}_{n+1}} \right) + \\ &- (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial \underline{F}_{v,n+1}}{\partial \underline{q}_{n+1}} + \gamma \Delta t \frac{\partial \underline{F}_{v,n+1}}{\partial \underline{\dot{q}}_{n+1}} \right), \end{aligned} \quad (\text{A52})$$

$$\frac{\partial R_{2,n+1-\alpha_f}}{\partial \underline{\underline{\underline{\dot{q}}}}_{n+1}} = (1 - \alpha_m) \left(\beta \Delta t^2 \frac{\partial J_{\Phi,n+1}}{\partial \underline{\underline{\underline{\dot{q}}}}_{n+1}} \underline{\underline{\underline{\dot{q}}}}_{n+1} + J_{\Phi,n+1} \right) + (1 - \alpha_f) \left(\beta \Delta t^2 \frac{\partial F_{c,n+1}}{\partial \underline{\underline{\underline{\dot{q}}}}_{n+1}} + \gamma \Delta t \frac{\partial F_{c,n+1}}{\partial \underline{\underline{\underline{\dot{q}}}}_{n+1}} \right), \tag{A53}$$

$$\frac{\partial R_{1,n+1-\alpha_f}}{\partial \lambda_{n+1}} = (1 - \alpha_m) J_{\Phi,n+1}^T, \tag{A54}$$

$$\frac{\partial R_{2,n+1-\alpha_f}}{\partial \lambda_{n+1}} = \underline{\underline{\underline{0}}}. \tag{A55}$$

Appendix B.3. Sensitivity Analysis

The derivative of the equations of motion leads to the governing equation of the sensitivities given by

$$\begin{aligned} & (1 - \alpha_m) \underline{\underline{\underline{\nabla m}}}_{n+1} \underline{\underline{\underline{\ddot{q}}}}_{n+1} + \alpha_m \underline{\underline{\underline{\nabla m}}}_n \underline{\underline{\underline{\ddot{q}}}}_n + \\ & + (1 - \alpha_m) \underline{\underline{\underline{m}}}_{n+1} \underline{\underline{\underline{\nabla \ddot{q}}}}_{n+1} + \alpha_m \underline{\underline{\underline{m}}}_n \underline{\underline{\underline{\nabla \ddot{q}}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{\underline{\nabla d}}}_{n+1} \underline{\underline{\underline{\dot{q}}}}_{n+1} + \alpha_f \underline{\underline{\underline{\nabla d}}}_n \underline{\underline{\underline{\dot{q}}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{\underline{d}}}_{n+1} \underline{\underline{\underline{\nabla \dot{q}}}}_{n+1} + \alpha_f \underline{\underline{\underline{d}}}_n \underline{\underline{\underline{\nabla \dot{q}}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{\underline{\nabla k}}}_{n+1} \underline{\underline{\underline{q}}}_{n+1} + \alpha_f \underline{\underline{\underline{\nabla k}}}_n \underline{\underline{\underline{q}}}_n + \\ & + (1 - \alpha_f) \underline{\underline{\underline{k}}}_{n+1} \underline{\underline{\underline{\nabla q}}}_{n+1} + \alpha_f \underline{\underline{\underline{k}}}_n \underline{\underline{\underline{\nabla q}}}_n + \\ & + (1 - \alpha_m) \underline{\underline{\underline{\nabla J}}}_{\Phi,n+1}^T \lambda_{n+1} + \alpha_m \underline{\underline{\underline{\nabla J}}}_{\Phi,n}^T \lambda_n + \\ & + (1 - \alpha_m) J_{\Phi,n+1}^T \underline{\underline{\underline{\nabla \lambda}}}_{n+1} + \alpha_m J_{\Phi,n}^T \underline{\underline{\underline{\nabla \lambda}}}_n = (1 - \alpha_f) \underline{\underline{\underline{\nabla F}}}_{\text{ext},n+1} + \alpha_f \underline{\underline{\underline{\nabla F}}}_{\text{ext},n} + \\ & + (1 - \alpha_f) \underline{\underline{\underline{\nabla F}}}_{v,n+1} + \alpha_f \underline{\underline{\underline{\nabla F}}}_{v,n}, \end{aligned} \tag{A56}$$

and

$$\begin{aligned} & (1 - \alpha_m) \underline{\underline{\underline{\nabla J}}}_{\Phi,n+1} \underline{\underline{\underline{\dot{q}}}}_{n+1} + \alpha_m \underline{\underline{\underline{\nabla J}}}_{\Phi,n} \underline{\underline{\underline{\dot{q}}}}_n + \\ & + (1 - \alpha_m) J_{\Phi,n+1} \underline{\underline{\underline{\nabla \dot{q}}}}_{n+1} + \alpha_m J_{\Phi,n} \underline{\underline{\underline{\nabla \dot{q}}}}_n = (1 - \alpha_f) \underline{\underline{\underline{\nabla F}}}_{c,n+1} + \alpha_f \underline{\underline{\underline{\nabla F}}}_{c,n}. \end{aligned} \tag{A57}$$

Including Newmark’s sensitivity, Equations (16) and (17) lead to

The governing equation of the sensitivity analysis of the effective system is given by

$$\begin{bmatrix} \underline{m}_{\text{eff}} & \underline{J}_{\Phi,\text{eff}}^T \\ \underline{J}_{\Phi,\text{eff}} & \underline{0} \end{bmatrix} \begin{bmatrix} \underline{\nabla \dot{q}}_{n+1} \\ \underline{\nabla \lambda}_{n+1} \end{bmatrix} = \begin{bmatrix} \underline{F}_{a,\text{pseudo}} \\ \underline{F}_{c,\text{pseudo}} \end{bmatrix}, \quad (\text{A62})$$

where

$$\begin{aligned} \underline{F}_{a,\text{pseudo}} = & (1 - \alpha_f) \underline{\nabla F}_{\text{ext},n+1} + \alpha_f \underline{\nabla F}_{\text{ext},n} + (1 - \alpha_f) \underline{\nabla F}_{v,n+1} + \alpha_f \underline{\nabla F}_{v,n} + \\ & - (1 - \alpha_m) \underline{\nabla m}_{n+1} \underline{\ddot{q}}_{n+1} - \alpha_m \underline{\nabla m}_n \underline{\ddot{q}}_n - \alpha_m \underline{m}_n \underline{\nabla \ddot{q}}_n + \\ & - (1 - \alpha_f) \underline{\nabla d}_{n+1} \underline{\dot{q}}_{n+1} - \alpha_f \underline{\nabla d}_n \underline{\dot{q}}_n - (1 - \alpha_f) \underline{d}_{n+1} \underline{\nabla \dot{q}}_{\text{pred}} - \alpha_f \underline{d}_n \underline{\nabla \dot{q}}_n + \\ & - (1 - \alpha_f) \underline{\nabla k}_{n+1} \underline{q}_{n+1} - \alpha_f \underline{\nabla k}_n \underline{q}_n - (1 - \alpha_f) \underline{k}_{n+1} \underline{\nabla q}_{\text{pred}} - \alpha_f \underline{k}_n \underline{\nabla q}_n + \\ & - (1 - \alpha_m) \underline{\nabla J}_{\Phi,n+1}^T \underline{\lambda}_{n+1} - \alpha_m \underline{\nabla J}_{\Phi,n}^T \underline{\lambda}_n - \alpha_m \underline{J}_{\Phi,n}^T \underline{\nabla \lambda}_n, \end{aligned} \quad (\text{A63})$$

$$\begin{aligned} \underline{F}_{c,\text{pseudo}} = & (1 - \alpha_f) \underline{\nabla F}_{c,n+1} + \alpha_f \underline{\nabla F}_{c,n} + \\ & - (1 - \alpha_m) \underline{\nabla J}_{\Phi,n+1} \underline{\ddot{q}}_{n+1} - \alpha_m \underline{\nabla J}_{\Phi,n} \underline{\ddot{q}}_n - \alpha_m \underline{J}_{\Phi,n} \underline{\nabla \ddot{q}}_n. \end{aligned} \quad (\text{A64})$$

References

- Hien, T.D.; Kleiber, M. Stochastic design sensitivity in structural dynamics. *Int. J. Numer. Methods Eng.* **1991**, *32*, 1247–1265. [\[CrossRef\]](#)
- Cho, S.; Choi, K.K. Design sensitivity analysis and optimization of non-linear transient dynamics. Part I—sizing design. *Int. J. Numer. Methods Eng.* **2000**, *48*, 351–373. [\[CrossRef\]](#)
- Cho, S.; Choi, K.K. Design sensitivity analysis and optimization of non-linear transient dynamics. Part II—configuration design. *Int. J. Numer. Methods Eng.* **2000**, *48*, 375–399. [\[CrossRef\]](#)
- van Keulen, F.; Haftka, R.; Kim, N. Review of options for structural design sensitivity analysis—Part 1: Linear systems. *Comput. Methods Appl. Mech. Eng.* **2005**, *194*, 3213–3243. [\[CrossRef\]](#)
- Martins, J.R.R.A.; Hwang, J.T. Review and unification of methods for computing derivatives of multidisciplinary computational models. *AIAA J.* **2013**, *51*, 2582–2599. [\[CrossRef\]](#)
- Tortorelli, D.A.; Michaleris, P. Design sensitivity analysis: Overview and review. *Inverse Probl. Eng.* **1994**, *1*, 71–105. [\[CrossRef\]](#)
- Wehrle, E.; Gufler, V. Lightweight engineering design of nonlinear dynamic systems with gradient-based structural design optimization. In *Proceedings of the Munich Symposium on Lightweight Design 2020*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 44–57. [\[CrossRef\]](#)
- Wehrle, E. Optimal lightweight engineering via a three-block solver scheme for mechanical analysis. In *Optimal Design and Control of Multibody Systems*; Nachbagauer, K., Held, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2024; Volume 42, pp. 16–29. [\[CrossRef\]](#)
- Courant, R.; Friedrichs, K.; Lewy, H. Über die partiellen Differenzgleichungen der mathematischen Physik. *Math. Ann.* **1928**, *100*, 32–74. [\[CrossRef\]](#)
- Newmark, N.M. A method of computation for structural dynamics. *J. Eng. Mech.* **1959**, *85*, 67–94. [\[CrossRef\]](#)
- Hilber, H.M.; Hughes, T.J.R.; Taylor, R.L. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthq. Eng. Struct. Dyn.* **1977**, *5*, 283–292. [\[CrossRef\]](#)
- Wood, W.L.; Bossak, M.; Zienkiewicz, O.C. An alpha modification of Newmark's method. *Int. J. Numer. Methods Eng.* **1980**, *15*, 1562–1566. [\[CrossRef\]](#)
- Chung, J.; Hulbert, G.M. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- α method. *J. Appl. Mech.* **1993**, *60*, 371–375. [\[CrossRef\]](#)
- Gufler, V.; Wehrle, E.; Vidoni, R. Sensitivity analysis of flexible multibody dynamics with generalized- α time integration and Baumgarte stabilization: A study on numerical stability. In *Mechanisms and Machine Science*; Springer International Publishing: Berlin/Heidelberg, Germany, 2022; pp. 147–155. [\[CrossRef\]](#)
- Griewank, A.; Walther, A. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, 2nd ed.; SIAM: Philadelphia, PA, USA, 2008.
- Naumann, U. *The Art of Differentiating Computer Programs: An Introduction to Algorithmic Differentiation*; SIAM: Philadelphia, PA, USA, 2012.
- Callejo, A.; Dopico, D. Direct sensitivity analysis of multibody systems: A vehicle dynamics benchmark. *J. Comput. Nonlinear Dyn.* **2019**, *14*, 021004. [\[CrossRef\]](#)

18. Margossian, C.C. A review of automatic differentiation and its efficient implementation. *WIREs Data Min. Knowl. Discov.* **2019**, *9*, e1305. [[CrossRef](#)]
19. Pi, T.; Zhang, Y.; Chen, L. First order sensitivity analysis of flexible multibody systems using absolute nodal coordinate formulation. *Multibody Syst. Dyn.* **2012**, *27*, 153–171. [[CrossRef](#)]
20. Held, A.; Knüfer, S.; Seifried, R. Structural sensitivity analysis of flexible multibody systems modeled with the floating frame of reference approach using the adjoint variable method. *Multibody Syst. Dyn.* **2016**, *40*, 287–302. [[CrossRef](#)]
21. Boopathy, K.; Kennedy, G. Adjoint-based derivative evaluation methods for flexible multibody systems with rotorcraft applications. In Proceedings of the 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, USA, 9–13 January 2017; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2017. [[CrossRef](#)]
22. Martins, J.R.R.A.; Kennedy, G.J. Enabling large-scale multidisciplinary design optimization through adjoint sensitivity analysis. *Struct. Multidiscip. Optim.* **2021**, *64*, 2959–2974. [[CrossRef](#)]
23. Nachbagauer, K.; Oberpeilsteiner, S.; Sherif, K.; Steiner, W. The use of the adjoint method for solving typical optimization problems in multibody dynamics. *J. Comput. Nonlinear Dyn.* **2015**, *10*, 061011. [[CrossRef](#)]
24. Lauß, T.; Oberpeilsteiner, S.; Steiner, W.; Nachbagauer, K. The discrete adjoint method for parameter identification in multibody system dynamics. *Multibody Syst. Dyn.* **2017**, *42*, 397–410. [[CrossRef](#)]
25. Boopathy, K.; Kennedy, G.J. Parallel finite element framework for rotorcraft multibody dynamics and discrete adjoint sensitivities. *Aiaa J.* **2019**, *57*, 3159–3172. [[CrossRef](#)]
26. Boopathy, K. Adjoint Based Design Optimization of Systems with Time Dependent Physics and Probabilistically Modeled Uncertainties. Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA, USA, 2020.
27. Ebrahimi, M.; Butscher, A.; Cheong, H.; Iorio, F. Design optimization of dynamic flexible multibody systems using the discrete adjoint variable method. *Comput. Struct.* **2019**, *213*, 82–99. [[CrossRef](#)]
28. Nejat, A.A.; Moghadasi, A.; Held, A. Adjoint sensitivity analysis of flexible multibody systems in differential-algebraic form. *Comput. Struct.* **2020**, *228*, 106–148. [[CrossRef](#)]
29. Held, A. On design sensitivities in the structural analysis and optimization of flexible multibody systems. *Multibody Syst. Dyn.* **2022**, *54*, 53–74. [[CrossRef](#)]
30. Solano, D.; Sarojini, D.; Rajaram, D.; Mavris, D.N. Adjoint-based analysis and optimization of beam-like structures subjected to dynamic loads. *Struct. Multidiscip. Optim.* **2022**, *65*, 52. [[CrossRef](#)]
31. Haug, E.J.; Arora, J.S. Design sensitivity analysis of elastic mechanical systems. *Comput. Methods Appl. Mech. Eng.* **1978**, *15*, 35–62. [[CrossRef](#)]
32. Hsieh, C.C.; Arora, J.S. Structural design sensitivity analysis with general boundary conditions: Dynamic problem. *Int. J. Numer. Methods Eng.* **1985**, *21*, 267–283. [[CrossRef](#)]
33. Baier, H.; Seeßelberg, C.; Specht, B. *Optimierung in der Strukturmechanik*; Vieweg: Braunschweig/Wiesbaden, Germany 1994. [[CrossRef](#)]
34. Trier, S.; Marthinsen, A.; Sivertsen, O. Design sensitivities by the adjoint variable method in nonlinear structural dynamics. In Proceedings of the SIMS Simulation Conference, Trondheim, Norway, 11–13 June 1996.
35. Michaleris, P.; Tortorelli, D.A.; Vidal, C.A. Tangent operators and design sensitivity formulations for transient non-linear coupled problems with applications to elastoplasticity. *Int. J. Numer. Methods Eng.* **1994**, *37*, 2471–2499. [[CrossRef](#)]
36. Bhalerao, K.D.; Poursina, M.; Anderson, K.S. An efficient direct differentiation approach for sensitivity analysis of flexible multibody systems. *Multibody Syst. Dyn.* **2010**, *23*, 121–140. [[CrossRef](#)]
37. Zhu, Y. Sensitivity Analysis and Optimization of Multibody Systems. Ph.D. Thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA, 2014.
38. Dopico, D.; Zhu, Y.; Sandu, A.; Sandu, C. Direct and adjoint sensitivity analysis of ordinary differential equation multibody formulations. *J. Comput. Nonlinear Dyn.* **2014**, *10*, 011012. [[CrossRef](#)]
39. Callejo, A.; Sonneville, V.; Bauchau, O. Discrete adjoint method for the sensitivity analysis of flexible multibody systems. *J. Comput. Nonlinear Dyn.* **2018**, *14*, 021001. [[CrossRef](#)]
40. Cao, Z.; Yao, J.; Jia, Z.; Liang, D. Transient response sensitivity analysis of localized nonlinear structure using direct differentiation method. *Machines* **2022**, *10*, 1039. [[CrossRef](#)]
41. Haug, E.J.; Ehle, P.E. Second-order design sensitivity analysis of mechanical system dynamics. *Int. J. Numer. Methods Eng.* **1982**, *18*, 1699–1717. [[CrossRef](#)]
42. Ding, J.Y.; Pan, Z.K.; Chen, L.Q. Second order adjoint sensitivity analysis of multibody systems described by differential-algebraic equations. *Multibody Syst. Dyn.* **2007**, *18*, 599–617. [[CrossRef](#)]
43. Belotti, R.; Palomba, I.; Wehrle, E.; Vidoni, R. An approximation-based design optimization approach to eigenfrequency assignment for flexible multibody systems. *Appl. Sci.* **2021**, *11*, 11558. [[CrossRef](#)]
44. Palomba, I.; Vidoni, R. Flexible-link multibody system eigenvalue analysis parameterized with respect to rigid-body motion. *Appl. Sci.* **2019**, *9*, 5156. [[CrossRef](#)]
45. Brüls, O.; Lemaire, E.; Duysinx, P.; Eberhard, P., Multibody dynamics. In *Multibody Dynamics: Computational Methods and Applications*; Arczewski, K., Blajer, W., Fraczek, J., Wojtyra, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; Chapter Optimization of Multibody Systems and Their Structural Components, pp. 49–68. [[CrossRef](#)]

46. Gufler, V.; Wehrle, E.; Vidoni, R. Multiphysical design optimization of multibody systems: Application to a Tyrolean weir cleaning mechanism. In Proceedings of the 3rd International Conference of IFToMM Italy, Naples, Italy, 9–11 September 2020. [\[CrossRef\]](#)
47. Gufler, V.; Wehrle, E.; Achleitner, J.; Vidoni, R. A semi-analytical approach to sensitivity analysis with flexible multibody dynamics of a morphing forward wing section. *Multibody Syst. Dyn.* **2023**, *58*, 1–20. [\[CrossRef\]](#)
48. Gufler, V.; Zwölfer, A.; Wehrle, E. Analytical derivatives of flexible multibody dynamics with the floating frame of reference formulation. *Multibody Syst. Dyn.* **2022**, *60*, 257–288. [\[CrossRef\]](#)
49. Brüls, O.; Golinval, J. The generalized- α method in mechatronic applications. *ZAMM* **2006**, *86*, 748–758. [\[CrossRef\]](#)
50. Jansen, K.E.; Whiting, C.H.; Hulbert, G.M. A generalized- α method for integrating the filtered Navier–Stokes equations with a stabilized finite element method. *Comput. Methods Appl. Mech. Eng.* **2000**, *190*, 305–319. [\[CrossRef\]](#)
51. Köbis, M.A.; Arnold, M. Convergence of generalized- α time integration for nonlinear systems with stiff potential forces. *Multibody Syst. Dyn.* **2015**, *37*, 107–125. [\[CrossRef\]](#)
52. Bestle, D. *Analyse und Optimierung von Mehrkörpersystemen*; Springer: Berlin/Heidelberg, Germany, 1994. [\[CrossRef\]](#)
53. Haug, E.J.; Arora, J.S. *Applied Optimal Design: Mechanical and Structural Systems*; John Wiley & Sons: Hoboken, NJ, USA, 1979.
54. Gufler, V.; Wehrle, E.; Zwölfer, A. A review of flexible multibody dynamics for gradient-based design optimization. *Multibody Syst. Dyn.* **2021**, *53*, 379–409. [\[CrossRef\]](#)
55. Shabana, A.A. Flexible multibody dynamics: Review of past and recent developments. *Multibody Syst. Dyn.* **1997**, *1*, 189–222. [\[CrossRef\]](#)
56. Shabana, A.A. *Dynamics of Multibody Systems*, 3rd ed.; Cambridge University Press: Cambridge, UK, 2013. [\[CrossRef\]](#)
57. Petzold, L.; Lötstedt, P. Numerical solution of nonlinear differential equations with algebraic constraints II: Practical implications. *SIAM J. Sci. Stat. Comput.* **1986**, *7*, 720–733. [\[CrossRef\]](#)
58. Bottasso, C.L.; Bauchau, O.A.; Cardona, A. Time-step-size-independent conditioning and sensitivity to perturbations in the numerical solution of index three differential algebraic equations. *SIAM J. Sci. Comput.* **2007**, *29*, 397–414. [\[CrossRef\]](#)
59. Bauchau, O.A. *Flexible Multibody Dynamics*; Springer: Berlin/Heidelberg, Germany, 2011. [\[CrossRef\]](#)
60. Dopico, D.; González, F.; Luaces, A.; Saura, M.; García-Vallejo, D. Direct sensitivity analysis of multibody systems with holonomic and nonholonomic constraints via an index-3 augmented Lagrangian formulation with projections. *Nonlinear Dyn.* **2018**, *93*, 2039–2056. [\[CrossRef\]](#)
61. Baumgarte, J. Stabilization of constraints and integrals of motion in dynamical systems. *Comput. Methods Appl. Mech. Eng.* **1972**, *1*, 1–16. [\[CrossRef\]](#)
62. Gufler, V.; Wehrle, E.; Vidoni, R. Analytical sensitivity analysis of flexible multibody dynamics with index-1 differential-algebraic equations and Baumgarte stabilization. *Int. J. Mech. Control* **2023**, *24*, 3–14.
63. Arnold, M.; Brüls, O. Convergence of the generalized- α scheme for constrained mechanical systems. *Multibody Syst. Dyn.* **2007**, *18*, 185–202. [\[CrossRef\]](#)
64. Bathe, K.J.; Noh, G. Insight into an implicit time integration scheme for structural dynamics. *Comput. Struct.* **2012**, *98*–99, 1–6. [\[CrossRef\]](#)
65. Huang, C.; Fu, M. A composite collocation method with low-period elongation for structural dynamics problems. *Comput. Struct.* **2018**, *195*, 74–84. [\[CrossRef\]](#)
66. Kadapa, C.; Dettmer, W.; Perić, D. On the advantages of using the first-order generalised-alpha scheme for structural dynamic problems. *Comput. Struct.* **2017**, *193*, 226–238. [\[CrossRef\]](#)
67. Malakiyeh, M.M.; Shojaee, S.; Javaran, S.H. Development of a direct time integration method based on Bezier curve and 5th-order Bernstein basis function. *Comput. Struct.* **2018**, *194*, 15–31. [\[CrossRef\]](#)
68. Namadchi, A.H.; Jandaghi, E.; Alamatian, J. A new model-dependent time integration scheme with effective numerical damping for dynamic analysis. *Eng. Comput.* **2020**, *37*, 2543–2558. [\[CrossRef\]](#)
69. Noh, G.; Bathe, K.J. The Bathe time integration method with controllable spectral radius: The ρ_∞ -Bathe method. *Comput. Struct.* **2019**, *212*, 299–310. [\[CrossRef\]](#)
70. Shojaee, S.; Rostami, S.; Abbasi, A. An unconditionally stable implicit time integration algorithm: Modified quartic B-spline method. *Comput. Struct.* **2015**, *153*, 98–111. [\[CrossRef\]](#)
71. Wen, W.; Wei, K.; Lei, H.; Duan, S.; Fang, D. A novel sub-step composite implicit time integration scheme for structural dynamics. *Comput. Struct.* **2017**, *182*, 176–186. [\[CrossRef\]](#)
72. Bathe, K.J. *Finite Element Procedures*; Prentice Hall: Englewood Cliffs, NJ, USA, 1996; p. 1037.
73. Kreisselmeier, G.; Steinhauser, R. Systematic control design by optimizing a vector performance index. In Proceedings of the International Federation of Active Controls Symposium on Computer-Aided Design of Control Systems, Zürich, Switzerland, 29–31 August 1979. [\[CrossRef\]](#)
74. Martins, J.R.R.A.; Poon, N.M.K. On structural optimization using constraint aggregation. In Proceedings of the 6th World Congress on Structural and Multidisciplinary Optimization, Rio de Janeiro, Brazil, 30 May–3 June 2005.
75. Duddeck, F. Multidisciplinary optimization of car bodies. *Struct. Multidiscip. Optim.* **2008**, *35*, 375–389. [\[CrossRef\]](#)
76. Kurtaran, H.; Eskandarian, A.; Marzougui, D.; Bedewi, N.E. Crashworthiness design optimization using successive response surface approximations. *Comput. Mech.* **2002**, *29*, 409–421. [\[CrossRef\]](#)
77. Langer, H. Extended Evolutionary Algorithms for Multiobjective and Discrete Design Optimization of Structure. Ph.D. Dissertation, Lehrstuhl für Leichtbau, Technische Universität München, Munich, Germany, 2005.

78. Boursier Niutta, C.; Wehrle, E.J.; Duddeck, F.; Belingardi, G. Surrogate modeling in design optimization of structures with discontinuous responses: A new approach for ill-posed problems in crashworthiness design. *Struct. Multidiscip. Optim.* **2018**, *57*, 1857–1869. [[CrossRef](#)]
79. Forrester, A.I.J.; Sábester, A.; Keane, A.J. *Engineering Design via Surrogate Modelling: A Practical Guide*; Wiley: Hoboken, NJ, USA, 2008; p. 227.
80. Xu, Q.; Wehrle, E.; Baier, H. Knowledge-based surrogate modeling in engineering design optimization. In *Surrogate-Based Modeling and Optimization*; Koziel, S., Leifsson, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 313–336. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.