Non-incremental response evaluation in geometrically nonlinear structural dynamics using a space-time stiffness operator

Tahar Arjoune · Bernd Markert · Franz Bamer

Received: 15 December 2021 / Accepted: 21 March 2022 / Published online: 30 April 2022
© The Author(s) 2022

Abstract
This contribution presents a proper generalized decomposition-based nonlinear solver for an efficient solution of geometrically nonlinear dynamic problems. The solution is built as a sum of dyadic products of space and time modes, and this sum of so-called enrichments is truncated when the required accuracy is achieved. In the proposed algorithm, we apply a consistent linearization of the residual vectors around the currently known solution over the whole space-time domain. At first, the set of vectorized tangent stiffness matrices is separated in space and time using the singular value decomposition. Then, the left and right singular vectors are reshaped into matrices to separate the space-time stiffness operator. The latter can be incorporated into the alternating fixed-point algorithm to compute couples of space and time modes. Numerical examples of a two-dimensional geometrically exact beam model demonstrate the accuracy, efficiency, and limits of the method.

Keywords Proper generalized decomposition · Nonlinear solver · Model order reduction · Geometrical nonlinearities · Large displacements

1 Introduction

An extensive range of numerical analysis tools can predict the dynamical behavior of complex structures and mechanical systems. The associated governing equations of motion are commonly discretized in space with the finite element method [1] and in time using time integration schemes [2,3]. In order to solve these algebraic equations, the classical, incremental, step-by-step approach is widely used in linear and nonlinear numerical analysis. Geometrical nonlinearity is required to model structures undergoing large displacements and rotations, which is a relevant feature in many engineering applications. However, solving the corresponding nonlinear equations of motion is computationally demanding, especially when the transient response of the system over a large time interval is needed since an iterative Newton–Raphson procedure [4] with several residual and Jacobian evaluations is required within every calculation step. In order to find sufficiently accurate solutions with significantly lower computational costs, several approximation methods and model order reduction strategies have been proposed in the literature. They can be divided into methods that require offline computations to achieve online speedups and methods that do not need any offline step.

Projection-based model order reduction requires the execution of offline computations to define a reduction basis before performing more efficient online computations. It aims to decrease the computational effort by considering a reduced number of unknowns, which can describe almost all relevant configurations of the structure for the problem at hand. This smaller subspace, where the solution is assumed to exist, is spanned by a set of basis vectors, which define possible displacement shapes and form a reduction basis. In linear structural dynamics, an appropriate reduction basis can be identified with the modal truncation method, which goes back to Lord Rayleigh [5]. It approximates the dynamical system by a superposition of a selected set of vibration modes. The Krylov Subspace method [6] exploits the information about the location of external excitation points to build the reduction basis from static displacement fields and...
higher-order approximations. Lately, Sonneville et al. [7] presented an extensive review of component mode synthesis methods a comprehensive review of modal reduction procedures for flexible multibody dynamics introducing a general approach for the modal reduction of geometrically nonlinear problems. The component mode synthesis and substructuring techniques [8–13] enable the definition of individual reduction bases in different subdomains of a partitioned structure. These methods enable the assembly of the reduced substructures by keeping the degrees of freedom of the boundaries unaltered.

Although the methods mentioned above are very well applicable to linear systems, they perform very poorly for geometrically nonlinear structures. Idelson and Cardona [14] propose the augmentation of the linear reduction basis with modal derivatives in order to capture the nonlinear behavior of the system. The computation of these additional basis vectors is based on the perturbation of the eigenvalue problem around the equilibrium position. This method is extensively used for the reduction of geometrically nonlinear structures [15–20] since it exploits intrinsic physical properties of the system and is less sensitive to changes in model parameters, boundary conditions, or excitations in comparison with data-driven approaches [21]. The most prominent data-driven approach, the proper orthogonal decomposition, also known as Karhunen–Loève decomposition, identifies the optimal subspace from a set of displacement snapshots, which are computed using training simulations of the full, high dimensional system [22,23]. The basis vectors can be extracted from the set of snapshots by applying the singular value decomposition. The proper orthogonal decomposition is widespread in nonlinear structural dynamics [24–28] and fluid mechanics [29,30] as well as in many other fields, as it is independent of the considered system and can be applied to any model as long as representative training simulations are available.

The projection of the nonlinear equations of motion onto a smaller subspace spanned by a meticulously chosen reduction basis reduces the required computational effort by reducing the linear systems of equations that need to be solved in every Newton–Raphson iteration. However, the evaluation of the nonlinear internal force vectors and tangent stiffness matrices of the reduced system is still related to the size of the high dimensional system because the internal forces in geometrically nonlinear finite elements are formulated on the element level. In order to overcome this computational bottleneck and fully make use of the dimensionality reduction, hyperreduction techniques were proposed in the literature. The polynomial tensors method [31,32] exploits the polynomial structure of the internal forces of geometrically nonlinear structures. Computational savings are related to the cheaper evaluation of a multidimensional polynomial with fewer variables and precomputed coefficients. The energy-conserving mesh sampling and weighting method [33] reduces the computational costs by evaluating only a subset of finite elements and weighting their contributions by suitably chosen weights during the assembly process. The discrete empirical interpolation method [34] approximates the high dimensional nonlinear force vector with an empirical force basis using the proper orthogonal decomposition in a similar manner to the approximation of displacements.

The a priori identification of a suitable subspace for the application of projection-based reduction is, in general, a tedious and problem-specific task, in particular in the nonlinear regime. It requires the execution of offline computations and a certain familiarity with the applied reduction methods. Note that the smaller set of nonlinear equations obtained with this method is solved with the same incremental Newton–Raphson algorithm used for the full model. On the contrary, the proper generalized decomposition (PGD) method is a non-incremental solution procedure, which does not require the execution of any problem-specific offline computations before being able to achieve speedups in online simulations [35–38]. Using an iterative process, it approximates the entire solution over the space-time domain as a sum of dyadic products of space and time mode couples. Therefore, it leads to a space-time separated representation of the solution. A so-called enrichment couple composed of a space mode and its corresponding time mode is computed in each iteration and added to the sum of the previous ones until the required accuracy is achieved. Different strategies for the computation of these space and time modes are presented by Nouy [39]. The main advantage of PGD compared to projection-based methods is that the relevant subspace for the problem is identified during the online computation. This non-incremental solution procedure was first introduced by Ladevèze [40,41] for efficiently solving nonlinear problems in structural mechanics. The PGD was then implemented for the reduction of multidimensional problems [42,43] and multiparametric models [74]. In addition, It was used for solving crack propagation problems in brittle materials [75] and lattice structures problems [76]. It was also applied in many other fields such as fluid dynamics [44–48], heat transfer [49,50] and chemistry [51].

The equations of motion derived from structural dynamics problems contain damping and inertia, dependent on velocities and accelerations. A discretization scheme is hence necessary for the approximation of these time derivatives. Implicit time integration schemes, such as the Newmark [52] scheme, are very popular in the field of structural dynamics. They are usually employed for incremental solution procedures so that the displacements, velocities, and accelerations at the current time step can be computed using the solution of the previous time step. Boucinha et al. proposed a tensorial formulation for the integration of the Newmark scheme into the non-incremental multi-field PGD solution procedure.
The proper generalized decomposition was efficiently implemented for solving several nonlinear dynamic and static problems, where the nonlinear terms were handled with various strategies. The most straightforward and primarily used alternative is the fixed point Picard iteration [38]. It was implemented within the PGD framework for solving nonlinear problems in computational rheology [55], multidimensional parametric modeling [42], magneto-thermal coupling [56] and heat transfer [49,57]. The fixed point Picard iteration approximates the nonlinear terms by partially substituting the unknown variables with the solution computed in the previous PGD iteration. This approach can only be applied to nonlinear equations with a specific structure [58]. The nonlinear algebraic equations describing geometrically nonlinear models do not generally satisfy this condition and hence can not be solved with the fixed point Picard technique. The asymptotic numerical method proposed in [59] was implemented within the PGD framework in order to approximate the nonlinear expressions in terms of an expansion parameter around the equilibrium configuration [60,61]. Although this efficient method leads to a solution of a sequence of linear problems with the same constant tangent stiffness matrix, it only delivers a good approximation of the solution in the neighborhood of the equilibrium configuration [62]. In the context of the quasistatic analysis of problems with material nonlinearities, the linearization of the internal forces using a unique constant tangent stiffness matrix at the equilibrium configuration in an analogous manner to the modified Newton method was successfully implemented in a PGD framework [63,64]. However, while quasi-static materially nonlinear problems have successfully and efficiently been solved using the PGD approach it has yet not been possible to provide appropriate PGD algorithms that ensure the convergence in the geometrically nonlinear case.

Thus, this paper presents a new space-time formulation for geometrically nonlinear static and dynamic problems in structural dynamics using the PGD method. In doing so, this paper proposes a consistent linearization of the nonlinear algebraic equations over the whole space-time domain within a progressive PGD framework. For this purpose, the residual vector, evaluated at every time step, is linearized around its corresponding currently known solution by evaluating the first two terms of a Taylor expansion analogously to the Newton–Raphson procedure. The evaluated set of tangent stiffness matrices is then separated in space and time using the singular value decomposition to be incorporated into the proposed PGD based nonlinear solver. This valuable Jacobi information ensures the convergence of the space-time solution of a broader range of nonlinear problems, amongst others geometrical nonlinearity.

We organize the paper as follows: In Sect. 2, the semi-discretized equation of motion, describing the dynamical behavior of a geometrically nonlinear hyperelastic structure, is introduced. Subsequently, the discretization in time and the solution procedure of the obtained algebraic system of equations by the incremental Newton–Raphson method are presented. In Sect. 3, the fundamentals of the proper generalized decomposition method and the embedding of the Newmark time integration scheme are described for the linear case. Building on this, the proposed solution procedure for the nonlinear case is introduced. In Sect. 4, the suggested algorithm is demonstrated on a static and a dynamic numerical example of a two-dimensional geometrically exact beam model, showing the accuracy, efficiency, and limits of the method. Finally, conclusions are drawn, and further discussions are presented in Sect. 5.

2 Preliminaries

The spatial discretization of the equation of motion of a geometrically nonlinear hyperelastic structure with the finite element method is written as:

\[ M \ddot{u}(t) + f_{\text{int}}(u(t)) = f_{\text{ext}}(t), \]  

(1)

with the vector of nodal displacements \( u(t) \in \mathbb{R}^{n_u} \), the mass matrix \( M \in \mathbb{R}^{n_u \times n_u} \), the internal restoring force vector \( f_{\text{int}}(u(t)) \in \mathbb{R}^{n_u} \) and the external force vector \( f_{\text{ext}}(t) \in \mathbb{R}^{n_u} \), where \( n_u \) is the number of degrees of freedom. The explicit dependence of \( u \) and \( f_{\text{ext}} \) on time will be henceforth omitted for the reason of abbreviation. The internal restoring force is a nonlinear function of the nodal displacements, since a nonlinear strain measure is required for the description of large displacements and finite rotations. A more detailed description of the two-dimensional geometrically exact beam element used in this paper is presented in appendix A.

We introduce viscous damping using the Rayleigh assumption:

\[ C = \alpha_d M + \beta_d K(u = 0), \]  

(2)

with \( 0 \leq \alpha_d, \beta_d \in \mathbb{R} \). This assumption is a weighted sum of the mass matrix and the initial tangent stiffness matrix evaluated at zero displacement, i.e., \( u = 0 \) [65]. Although this purely phenomenological damping approximation is based on the linearized model, it is often used in the nonlinear regime leading to the following damped nonlinear equations of motion:
\[ M\ddot{u} + C\dot{u} + f_{int}(u) = f_{ext}. \quad (3) \]

### 2.1 Time discretization

The semi-discretized equations of motion (1) and their damped counterpart (3) are ordinary differential equations stemming from the spatial discretization by the finite element method. These equations are solved based on specified initial displacements \( u(t=0) \) and velocities \( \dot{u}(t=0) \) in the time interval \( t \in [t_0, t_{end}] \). A temporal discretization with a time integration scheme is required in order to transform the ordinary differential equations to a set of algebraic equations, which can be solved with established algorithms. This means, that the solution \( u \) is defined on a set of discrete time instants \( \mathcal{P} = \{t_0, t_1, \ldots, t_n = t_{end}\} \) and in between these time instants one considers polynomial approximations.

The implicit one-step time integration scheme proposed by Newmark [52] is one the most popular schemes in the realm of structural dynamics since it makes use of the second-order structure of the equations of motion (3). The approximation of the velocities \( \dot{u}_{j+1} \) and displacements \( u_{j+1} \) at the time step \( t_{j+1} \) is expressed in terms of the unknown accelerations \( \ddot{u}_{j+1} \) and all quantities of the previous time step as:

\[
\dot{u}_{j+1} = \dot{u}_{j} + (1 - \gamma) \Delta t \ddot{u}_{j} + \gamma \Delta t \dot{u}_{j+1}, \quad (4)
\]

\[
u_{j+1} = u_{j} + \Delta t \dot{u}_{j} + \left( \frac{1}{2} - \beta \right) \Delta t^{2} \ddot{u}_{j} + \beta \Delta t^{2} \ddot{u}_{j+1}, \quad (5)
\]

where \( \Delta t \) is the time step size and the parameters \( \gamma \in [0, 1] \) and \( \beta \in [0, \frac{1}{2}] \) represent the weighting factors of the previous and the current time steps, which determine the stability and accuracy of the time integration scheme. The rearrangement of the Eqs. (4) and (5) yields the approximation of the velocities \( \dot{u}_{j+1} \) and accelerations \( \ddot{u}_{j+1} \) of the time step \( t_{j+1} \) in terms of the unknown displacements \( u_{j+1} \) and all quantities of the previous time step:

\[
\dot{u}_{j+1} = \frac{\gamma}{\beta \Delta t} (u_{j+1} - u_{j}) - \frac{\gamma - \beta}{\beta} \dot{u}_{j} - \frac{\gamma - 2 \beta}{2 \beta} \Delta t \ddot{u}_{j}, \quad (6)
\]

\[
\ddot{u}_{j+1} = \frac{1}{\beta \Delta t^{2}} (u_{j+1} - u_{j}) - \frac{1}{\beta \Delta t} \dot{u}_{j} - \frac{1 - 2 \beta}{2 \beta} \ddot{u}_{j}, \quad (7)
\]

Inserting Eqs. (6) and (7) into the semi-discretized equations (3) leads to the nonlinear algebraic balance equations:

\[
M \dddot{u}_{j+1} + C \dot{u}_{j+1} + f_{int}(u_{j+1}) = f_{ext,j+1}, \quad (8)
\]

where \( j = 0, \ldots, n_t - 1 \). They are solved for the unknown displacements \( u_{j+1} \) for a number \( n_t \) of time steps.

### 2.2 Incremental solution procedure

The nonlinear system of equations (8) are usually solved incrementally, where the solution \( u_{j+1} \) is computed iteratively using a Newton–Raphson solver based on the converged solution \( u_j \) of the previous time step [65]. This iterative process within each time step calculation tries to find the root of the residual function of the system of equations (8):

\[
r(u_{j+1}) = M \dddot{u}_{j+1} + C \dot{u}_{j+1} + f_{int}(u_{j+1}) - f_{ext,j+1} = 0. \quad (9)
\]

In each iteration \( i \) of the Newton–Raphson loop, a linearization of the residual around the current solution \( i u_{j+1} \) is evaluated and the linear system of equations:

\[
\frac{\partial r(u_{j+1})}{\partial u_{j+1}} \bigg|_{u_{j+1}} i \Delta u_{j+1} = -r(i u_{j+1}), \quad (10)
\]

is solved in order to compute the correction \( i \Delta u_{j+1} \) yielding the updated solution for the next iteration step:

\[
i+1 u_{j+1} = i u_{j+1} + i \Delta u_{j+1}. \quad (11)
\]

The steps (10) and (11) are repeated until convergence is achieved. A commonly used convergence criterion is the decrease of the \( L^2 \)-norm of the residual vector below a certain tolerance, which must be chosen carefully. The linearization of the residual (9) around a certain displacement \( i u_{j+1} \) is given as:

\[
iK_{dyne,j+1} = \frac{\partial r(u_{j+1})}{\partial u_{j+1}} \bigg|_{u_{j+1}} = \left( \frac{1}{\beta \Delta t^2} M + \frac{\gamma}{\beta \Delta t} C + K(i u_{j+1}) \right), \quad (12)
\]

where \( K(i u_{j+1}) \) is the tangent stiffness matrix (A23), evaluated for the corresponding configuration of the structure.

In the case of static analysis, the same aforementioned procedure is adopted to find a solution satisfying the equilibrium:

\[
r(u_{j+1}) = f_{int}(u_{j+1}) - f_{ext,j+1} = 0, \quad (13)
\]

where only internal and external forces play a role. Although there is no time discretization involved in this static problem, the external forces are incrementally increased, and the solution is computed in a step-wise manner to ensure the convergence of the Newton–Raphson nonlinear solver.
Algorithm 1 PGD solution procedure

1: initialize $U^{(0)}$
2: for $m = 1$ to $m_{\text{max}}$ do
3: \hspace{1em} initialize $u_i^{(m)}, u_j^{(m)}$
4: \hspace{1em} for $k = 1$ to $k_{\text{max}}$ do
5: \hspace{2em} $k_{u_i^{(m)}} \leftarrow \text{solve } R\left(U^{(m-1)} + k_{u_i^{(m)}} \otimes k_{1-u_i^{(m)}} \right) k_{1-u_i^{(m)}} = 0$ \hfill $\triangleright$ with a fixed time mode
6: \hspace{2em} normalize $k_{u_i^{(m)}}$
7: \hspace{2em} $k_{u_j^{(m)}} \leftarrow \text{solve } R\left(U^{(m-1)} + k_{u_j^{(m)}} \otimes k_{1-u_j^{(m)}} \right)^T k_{1-u_j^{(m)}} = 0$ \hfill $\triangleright$ with a fixed space mode
8: \hspace{2em} if $\| k_{u_k^{(m)}} \otimes k_{u_k^{(m)}} - k_{1-u_k^{(m)}} \otimes k_{1-u_k^{(m)}} \| < \epsilon_{fp}$ then $\triangleright$ convergence check
9: \hspace{2em} $u_i^{(m)} := k_{u_i^{(m)}}$
10: \hspace{2em} $u_j^{(m)} := k_{u_j^{(m)}}$
11: \hspace{1em} break
12: \hspace{2em} end if
13: \hspace{1em} end for
14: end if $\| R(U^{(m)}) \| < \epsilon_{res}$ then $\triangleright$ convergence check
15: \hspace{1em} break
16: end if
17: end for

3 Non-incremental solution procedure

3.1 The proper generalized decomposition

In this section, the non-incremental solution procedure, using the proper generalized decomposition (PGD), is introduced. For the reason of convenience, we apply a tensorial formulation by means of numerical tensors of order 1, 2, 3 and 4. Using the PGD, the solution matrix $U = [u_1 \cdots u_{n_2}] \in \mathbb{R}^{n_2 \times n_1}$ containing the $n_2$ solution vectors of the $n_1$ time steps of the discretized equations of motion (9) and (13) are approximated by the following rank-$m$ separated representation:

$$U^{(m)} = \sum_{i=0}^{m} u_i^{(m)} \otimes u_i^{(m)} = [u_1^{(m)} \cdots u_{n_2}^{(m)}]. \quad (14)$$

The approximation $U^{(m)}$ of the second-order tensor $U$ is constructed by the sum of a number $m$ of rank-one tensors, also called enrichments [53,66]. Each enrichment constitutes a tensor product of a space mode $u_i^{(1)} \in \mathbb{R}^{n_2}$ and a time mode $u_i^{(1)} \in \mathbb{R}^{n_1}$. The size $n_2$ of a space mode corresponds to the number of degrees of freedom of the considered mechanical system, and the size $n_1$ of a time mode represents the number of time steps associated with the time discretization. Using the non-incremental PGD procedure, we are searching for a solution $U^{(m)}$, for which the Frobenius norm of the matrix composed of the residual vectors of the equations of motion of all time steps is below a certain tolerance $\epsilon_{res}$:

$$\| R(U^{(m)}) \| < \epsilon_{res}. \quad (15)$$

where

$$R(U^{(m)}) = \left[ r(u_1^{(m)}) \cdots r(u_{n_2}^{(m)}) \right] \in \mathbb{R}^{n_2 \times n_1}. \quad (16)$$

If we assume that the decomposition $U^{(m-1)}$ of rank $m-1$ has been already computed in a previous step, then a new optimal enrichment couple $(u_i^{(m)}, u_j^{(m)})$ can be computed with the fixed point strategy [38,39,53,67–70], which represents the inner loop of algorithm 1. This strategy breaks the space-time problem into two coupled nonlinear problems, one in space (solving for $u_i^{(m)}$) and the other in time (solving for $u_j^{(m)}$), which are solved in an alternating manner until convergence. The starting point for solving this fixed point problem is an arbitrary initial guess $0_u^{(m)}$. In each iteration $k$, a space mode $k_u^{(m)}$ is computed for the fixed time mode $k_{1-u_i^{(m)}}$ (line 5 in algorithm 1), and then a new time mode $k_{u_j^{(m)}}$ is computed based on the fixed space mode $k_{u_i^{(m)}}$ (line 7 in algorithm 1), until the following convergence criterion in terms of the Frobenius norm is met:

$$\| k_{u_i^{(m)}} \otimes k_{u_j^{(m)}} - k_{1-u_i^{(m)}} \otimes k_{1-u_j^{(m)}} \| < \epsilon_{fp}, \quad (17)$$

where $\epsilon_{fp}$ is a carefully chosen tolerance. Note, that the fixed point problem can also start with an arbitrary space mode $0_u^{(m)}$ as an initial guess and that a maximum number of iterations can be specified with the parameters $k_{\text{max}}$. The normalization performed in line 6 of algorithm 1 in not absolutely necessary, however, it avoids very large or very small entries in the computed space and time modes [39,69].

The decomposition (14) can have different definitions and can be computed with several methods. An overview of these different possibilities is presented in [39]. In this paper, the progressive Galerkin-based PGD presented in algorithm 1 is used since it offers the best trade-off between robustness and computational costs and is, therefore, of high practical interest [39,69]. It projects the problem onto a subspace orthogonal to the residual in each iteration $k$, which leads to a time-independent space problem, when the new time mode
$k-1u_{(m)}^r$ is fixed, and to a scalar time problem, when the new space mode $k u_{(m)}^r$ is fixed. Note that there is a more robust but computationally less efficient alternative to the progressive Galerkin-based PGD, the progressive, minimal residual PGD method. In this formulation, the new optimal enrichment couple $(u_{s}^{(m)}, u_{t}^{(m)})$ is computed in each enrichment iteration $m$ by minimizing the residual norm:

$$(u_{s}^{(m)}, u_{t}^{(m)}) \in \arg \min_{u_{s}^{(m)} \in \mathbb{R}^{n_s}, u_{t}^{(m)} \in \mathbb{R}^{n_t}} \| R(U^{(m-1)} + u_{s}^{k} \otimes u_{t}^{k}) \|^2.$$  

(18)

The main advantage of the minimal residual formulation in comparison with the Galerkin-based strategy is that its convergence in the residual norm is proven to be monotonic [39,69].

The so-called optimal PGD is also proposed in the literature as an alternative to the progressive construction of the space and time modes [39,53,69]. It leads to an optimal decomposition (14) of rank $m$ in the sense of the Galerkin projection. This means that the residual (16) is simultaneously orthogonal to the subspace spanned by the space modes, $V_{m} = \text{span}([u_{s}^{(1)}, \ldots, u_{s}^{(m)}]) \subset \mathbb{R}^{n_s}$ and to the subspace spanned by the time modes, $T_{m} = \text{span}([u_{t}^{(1)}, \ldots, u_{t}^{(m)}]) \subset \mathbb{R}^{n_t}$. The optimal PGD is not used in this paper, as it turns out to be computationally inefficient for the problems discussed in this paper.

### 3.2 Solving linear dynamic problems with the PGD

The PGD solution procedure was extensively discussed in the literature for solving partial differential equations associated with different linear dynamic problems, such as the diffusion problem in [69] and the advection-diffusion-reaction problem in [39]. In the field of structural dynamics, PGD was implemented in [53] and [54] in order to solve the linear elastodynamic problem. When the governing partial differential equations are linear with respect to the variables, e.g., the displacements $u$, then the resulting systems of equations shown in lines 5 and 7 of algorithm 1 have to be solved in each iteration $k$ of the fixed point loop. They are linear and can, therefore, be solved with standard linear solvers [53].

The residual vector of the discretized equations of motion associated with the linear elastodynamic problem is given as:

$$r(u_{j+1}) = M \ddot{u}_{j+1} + C \dot{u}_{j+1} + K_0 u_{j+1} - f_{ext,j+1} = 0,$$

(19)

where $j = 0, \ldots, n_t - 1$ and the tangent stiffness matrix around the initial configuration is given as follows:

$$K_0 = K(u = 0) = \frac{\partial f_{int}(u)}{\partial u} \bigg|_{u=0}.$$  

(20)

Since the matrices $M, C$ and $K_0$ are constant and independent from the configuration of the structure, we can express the matrix composed of the residual vectors at all time steps as:

$$R = M \ddot{U} + C \dot{U} + K_0 U - F_{ext} = 0,$$

(21)

where

$$U = [u_1 \cdots u_{n_t}],$$

(22)

and

$$F_{ext} = [f_{ext,1} \cdots f_{ext,n_t}].$$

(23)

In the following, we introduce a tensorial formulation proposed in [53] and implemented in [54], which enables to express the matrices of velocities and accelerations, $\dot{U}$ and $\ddot{U}$, as functions of the displacements $U$ and the initial conditions, $u_0$ and $\dot{u}_0$, applying the Newmark scheme, presented in equations (6) and (7). First, equations (4) and (5) of the Newmark time integration scheme are rearranged as follows:

$$\begin{align*}
(\ddot{u}_{j+1} - \ddot{u}_j) + (\beta - 1) \Delta t \dddot{u}_{j} &= 0, \\
(\dot{u}_{j+1} - \dot{u}_j) + (-\gamma \Delta t \dddot{u}_{j+1} + (\gamma - 1) \Delta t \dddot{u}_j) &= 0,
\end{align*}$$

(24)

(25)

The key idea of this tensorial formulation is to build a matrix of the following linear combination of all couples of two successive displacement vectors:

$$\Delta U_{j+1}(c_1, c_2) = c_1 u_{j+1} + c_2 u_j,$$

(26)

with $j = 0, \ldots, n_t - 1$, as:

$$\Delta U(c_1, c_2) = [\Delta u_1(c_1, c_2) \cdots \Delta u_{n_t}(c_1, c_2)] = U(A(c_1, c_2))^T + u_0 \left( a(c_2) \right)^T.$$

(27)

where the coefficients $c_1, c_2 \in \mathbb{R}$ depend on the parameters of the time integration scheme and $U = [u_1 \cdots u_{n_t}] \in \mathbb{R}^{n_t \times n_t}$ is the matrix of the solutions at all time steps. The coefficient matrix is defined as:

$$A(c_1, c_2) = \begin{bmatrix}
c_1 & 0 & 0 & \cdots & 0 \\
c_2 & c_1 & 0 & \cdots & 0 \\
0 & c_2 & c_1 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & c_2 & c_1 \\
0 & \cdots & 0 & c_2 & c_1 \\
\end{bmatrix} \in \mathbb{R}^{n_t \times n_t}.$$  

(28)
and the coefficient vector as:

\[
a(c_2) = \begin{bmatrix}
  c_2 \\
  0 \\
  0 \\
  \vdots \\
  0
\end{bmatrix} \in \mathbb{R}^{n_t}.
\]  

(29)

The tensorial formulation (27) can be also applied to the velocity and acceleration vectors in order to express the equations (24) and (25) at all time steps as:

\[
\begin{align*}
\Delta \dot{U}(1, -1) + \Delta \ddot{U}(-\gamma \Delta t, (\gamma - 1) \Delta t) &= 0, \\
\Delta \dot{U}(1, -1) + \Delta \ddot{U}(0, -\Delta t) + \Delta \dddot{U} \left(-\beta \Delta t^2, (\beta - 0.5) \Delta t^2 \right) &= 0,
\end{align*}
\]

(30)

(31)

which can be also written as:

\[
\begin{align*}
\dot{U} A_1^T + \dot{\ddot{u}}_0 A_1^T + \ddot{U} A_4^T + \dddot{u}_0 A_4^T &= 0, \\
U A_1^T + \dot{u}_0 A_1^T + \dot{\ddot{u}}_0 A_3^T + \ddot{U} A_3^T + \dddot{u}_0 A_3^T &= 0,
\end{align*}
\]

(32)

(33)

where the definitions (28) and (29) are used as follows:

\[
\begin{align*}
A_1 &= A(1, -1), \\
A_2 &= A(0, -\Delta t), \\
A_3 &= A(-\beta \Delta t^2, (\beta - 0.5) \Delta t^2), \\
A_4 &= A(-\gamma \Delta t, (\gamma - 1) \Delta t), \\
a_1 &= a(-1), \\
a_2 &= a(-\Delta t), \\
a_3 &= a((\beta - 0.5) \Delta t^2), \\
a_4 &= a((\gamma - 1) \Delta t).
\end{align*}
\]

(34)

The initial acceleration is computed by solving the system of equations of motion for a given initial displacement and velocity:

\[
M \dddot{u}_0 = f_{ext, 0} - C \dot{u}_0 - K_0 u_0.
\]

(35)

Inserting the time integration equations (32) and (33) in the residual matrix of the equations of motion (21), leads to a set of algebraic equations with only the displacements \( U \) as variables:

\[
R(U) = M U Y - C U W + K_0 U - L = 0,
\]

(36)

In these equations, the matrices \( Y, W \) and \( L \) are defined as:

\[
Y = A_1^T D^{-1} H, \quad W = A_3^T D^{-1}, \quad L = F_{ext} + M \left[ -u_0 a_1^T + \dot{u}_0 (a_1^T (A_4^T)^{-1} A_4^T - a_2^T) + \dot{\ddot{u}}_0 (a_4^T (A_4^T)^{-1} A_4^T - a_5^T) \right] D^{-1} H
\]

(37)

and

\[
D = A_3^T - A_4^T (A_3^T)^{-1} A_3^T, \quad H = A_3^T (A_3^T)^{-1}.
\]

(38)

The solution of the space problem, as shown in line 5 of algorithm 1 and associated with the discretized equations of motion (36), is computed by solving the linear system of equations \( E_s k u_s^{(m)} = b_s \) with the corresponding coefficient matrix:

\[
E_s = \begin{pmatrix} (k-1) u_s^{(m)^T} \end{pmatrix} Y^{-1} \begin{pmatrix} (k-1) u_s^{(m)^T} \end{pmatrix} M
\]

\[
- \begin{pmatrix} (k-1) u_s^{(m)^T} \end{pmatrix} W \begin{pmatrix} (k-1) u_s^{(m)^T} \end{pmatrix} C
\]

\[
+ \begin{pmatrix} (k-1) u_s^{(m)^T} \end{pmatrix} \begin{pmatrix} k-1 u_s^{(m)^T} \end{pmatrix} K_0,
\]

(39)

(40)

and the right hand side is written as:

\[
b_s = \begin{pmatrix} \left( L - M U^{(m-1) T} \right) Y + C U^{(m-1) T} \end{pmatrix} W
\]

\[
- K_0 U^{(m-1) T} k-1 u_s^{(m)^T}.
\]

(41)

The solution of the corresponding time problem, as shown line 7 of algorithm 1, is computed by solving the linear system of equations \( E_t k u_t^{(m)} = b_t \) with the corresponding coefficient matrix:

\[
E_t = \begin{pmatrix} (k) u_s^{(m)^T} \end{pmatrix} Y \begin{pmatrix} M k u_s^{(m)^T} \end{pmatrix} \begin{pmatrix} M k u_s^{(m)^T} \end{pmatrix} Y^T
\]

\[
- \begin{pmatrix} (k) u_s^{(m)^T} \end{pmatrix} C \begin{pmatrix} k u_s^{(m)^T} \end{pmatrix} \begin{pmatrix} k u_s^{(m)^T} \end{pmatrix} W^T
\]

\[
+ \begin{pmatrix} (k) u_s^{(m)^T} \end{pmatrix} \begin{pmatrix} k u_s^{(m)^T} \end{pmatrix} \begin{pmatrix} K_0 k u_s^{(m)^T} \end{pmatrix} I_1
\]

(42)

where \( I_1 \in \mathbb{R}^{n_t \times n_t} \) is the identity matrix, and the right hand side is written as:

\[
b_t = \begin{pmatrix} \left( L - M U^{(m-1) T} \right) Y + C U^{(m-1) T} \end{pmatrix} W
\]

\[
- K_0 U^{(m-1) T} k u_s^{(m)^T}.
\]  

(43)
3.3 Solving nonlinear static and dynamic problems with the PGD

In this section, we introduce an extension of the PGD solution procedure, presented in algorithm 1, to the nonlinear case by linearizing the equations of motion of each time step \( j \) at the corresponding solution \( \mathbf{u}_j^{(m-1)} \), which represents a column vector of the space-time solution \( \mathbf{U}^{(m-1)} \), computed in the previous enrichment step. In this way, we can apply the fixed point iteration in each enrichment step \( m \) in a similar manner to the linear case, discussed in Sect. 3.2. The only difference is that in the nonlinear case, the linearization is not constant anymore and only evaluated once at the initial undeformed configuration, but it is dependent on the state of the structure.

In this paper, the PGD solution procedure for nonlinear structures is presented with a hyperelastic geometrically nonlinear finite element model, where the nonlinear term in the system of equations of motion (8) is the vector of internal forces \( \mathbf{f}_{\text{int}}(\mathbf{u}) \). In this case, the matrix, composed of the residual vectors at all time steps, is written for the static problem as:

\[
\mathbf{R}(\mathbf{U}) = \mathbf{F}_{\text{int}}(\mathbf{U}) - \mathbf{F}_{\text{ext}} = \mathbf{0},
\]

and for the dynamic problem as:

\[
\mathbf{R} = \mathbf{M} \ddot{\mathbf{U}} + \mathbf{C} \dot{\mathbf{U}} + \mathbf{F}_{\text{int}}(\mathbf{U}) - \mathbf{F}_{\text{ext}} = \mathbf{0},
\]

where

\[
\mathbf{F}_{\text{int}}(\mathbf{U}) = [\mathbf{f}_{\text{int}}(\mathbf{u}_1) \cdots \mathbf{f}_{\text{int}}(\mathbf{u}_{n_t})].
\]

The time discretization of the nonlinear equations of motion (45) is performed in the same manner as for the linear case (36) and is written using the tensorial formulation as:

\[
\mathbf{R}(\mathbf{U}) = \mathbf{M} \mathbf{U} \dot{\mathbf{Y}} - \mathbf{C} \mathbf{U} \dot{\mathbf{W}} + \mathbf{F}_{\text{int}}(\mathbf{U}) - \mathbf{L} = \mathbf{0}.
\]

In order to compute a new enrichment couple \( (\mathbf{u}_s^{(m)} , \mathbf{u}_t^{(m)}) \), the nonlinear terms of the residual vectors, arranged in the matrices (44) and (47), have to be linearized at the corresponding configuration of the structure, which is a column vector of the space-time solution \( \mathbf{U}^{(m-1)} = [\mathbf{u}_1^{(m-1)} \cdots \mathbf{u}_{n_t}^{(m-1)}] \) computed in the previous enrichment step. The matrix of linearized internal force vectors is given as:

\[
\text{Lin} \mathbf{F}_{\text{int}}(\mathbf{u}_j^{(m)}) = \text{Lin} \mathbf{F}_{\text{int}}(\mathbf{u}_j^{(m-1)} + \mathbf{u}_s^{(m)} \otimes \mathbf{u}_t^{(m)})
\]

\[
= [\mathbf{f}_{\text{int}}(\mathbf{u}_1^{(m-1)}) + \mathbf{K}(\mathbf{u}_1^{(m-1)}) \mathbf{u}_s^{(m)} \mathbf{u}_t^{(m)}]_1
\]

\[
\cdots , \mathbf{f}_{\text{int}}(\mathbf{u}_{n_t}^{(m-1)}) + \mathbf{K}(\mathbf{u}_{n_t}^{(m-1)}) \mathbf{u}_s^{(m)} \mathbf{u}_t^{(m)}
\]

where the column vectors of the new enrichment matrix are written as:

\[
\mathbf{u}_s^{(m)} \otimes \mathbf{u}_t^{(m)} = [\mathbf{u}_s^{(m)} \mathbf{u}_t^{(m)} \cdots \mathbf{u}_s^{(m)} \mathbf{u}_t^{(m)}],
\]

with

\[
\mathbf{u}_t^{(m)} = [\mathbf{u}_{t,1}^{(m)} \cdots \mathbf{u}_{t,n_t}^{(m)}]^T.
\]

Inserting (48) in (47) yields the matrix of linearized residual vectors:

\[
\text{Lin} \mathbf{R}(\mathbf{u}_j^{(m)}) = \mathbf{M} \mathbf{U}^{(m)} \mathbf{Y} - \mathbf{C} \mathbf{U}^{(m)} \mathbf{W}
\]

\[
+ \text{Lin} \mathbf{F}_{\text{int}}(\mathbf{U}^{(m)}) - \mathbf{L} = \mathbf{0},
\]

which has to be projected alternately on a fixed space mode and on a fixed time mode in order to find a solution (49) for the fixed point problem. For this purpose, a space-time separated representation of the second term of the matrix of linearized internal force vectors (48) is formulated as follows:

\[
\mathbf{K}(\mathbf{u}_1^{(m-1)}) \mathbf{u}_s^{(m)} \mathbf{u}_t^{(m)} , \cdots , \mathbf{K}(\mathbf{u}_{n_t}^{(m-1)}) \mathbf{u}_s^{(m)} \mathbf{u}_t^{(m)}
\]

\[
= \left( \sum_{l=1}^{n_k} \mathbf{K}_s^{(l)} \otimes \mathbf{K}_t^{(l)} \right) : (\mathbf{u}_s^{(m)} \otimes \mathbf{u}_t^{(m)})
\]

\[
= \sum_{l=1}^{n_k} \mathbf{K}_s^{(l)} \mathbf{u}_s^{(m)} \otimes \mathbf{K}_t^{(l)} \mathbf{u}_t^{(m)}.
\]

In this equation, the term \( \sum_{l=1}^{n_k} \mathbf{K}_s^{(l)} \otimes \mathbf{K}_t^{(l)} \in \mathbb{R}^{n_s \times n_s \times n_t \times n_t} \) is a separated representation of rank \( n_k \) of the bilinear stiffness operator, which constitutes a tensor of fourth-order and defines a linear mapping of a second-order tensor into another one [53,66]. The main idea of the proposed PGD solution procedure in algorithm 2 for solving nonlinear problems, defined in (47) and (44) is the introduction of a consistent linearization of the residual vectors, as shown in equations (48) to (52). Hence, the space problem solved in line 6 of algorithm 2 and associated with the nonlinear dynamic case (47) yields the linear system of equations with the coefficient matrix:

\[
\mathbf{E}_s = \left( (k-1 \mathbf{u}_t^{(m)})^T Y k-1 \mathbf{u}_t^{(m)} \right) \mathbf{M}
\]

\[
- \left( (k-1 \mathbf{u}_t^{(m)})^T W k-1 \mathbf{u}_t^{(m)} \right) \mathbf{C}
\]

\[
+ \sum_{l=1}^{n_k} \left( (\mathbf{K}_t^{(l)} k-1 \mathbf{u}_t^{(m)})^T k-1 \mathbf{u}_t^{(m)} \right) \mathbf{K}_s^{(l)}. \]

The right hand side is written as:
Algorithm 2 PGD solution procedure for nonlinear problems

1: initialize $U^{(0)}$
2: for $m \leftarrow 1$ to $m_{\text{max}}$ do
3: initialize $U^{(m)}$, $U^{(m)}$
4: Lin $R \left( U^{(m-1)} + u^{(m)} \otimes u^{(m)} \right) \leftarrow$ linearize $R \left( U \right)$ around $U^{(m-1)}$
5: for $k \leftarrow 1$ to $k_{\text{max}}$ do
6: $k u^{(m)} \leftarrow \text{solve Lin} \left( U^{(m-1)} + k u^{(m)} \otimes k^{-1} u^{(m)} \right) k^{-1} u^{(m)} = 0$ \hspace{1cm} \triangleright \text{with a fixed time mode}
7: normalize $k u^{(m)}$
8: $k u^{(m)} \leftarrow \text{solve Lin} \left( U^{(m-1)} + k u^{(m)} \otimes k^{-1} u^{(m)} \right)^T k u^{(m)} = 0$ \hspace{1cm} \triangleright \text{with a fixed space mode}
9: if $\| k u^{(m)} \otimes k u^{(m)} - k^{-1} u^{(m)} \otimes k^{-1} u^{(m)} \| < \epsilon_{fp}$ then
10: break
11: end if
12: break
13: end for
14: if $\| R(U^{(m)}) \| < \epsilon_{res}$ then
15: break
16: end if
17: end if
18: end for

\[ b \left( t \right) = \left( I - M U^{(m-1)} Y + C U^{(m-1)} W \right) - F_{\text{int}} \left( U^{(m-1)} \right) k^{-1} u^{(m)}. \] \hspace{1cm} (54)

The corresponding time problem, as shown in line 8 of algorithm 2, is equivalent to the linear system of equations with the coefficient matrix:

\[ E = \left( k u^{(m)} \right)^T M k u^{(m)} Y^T - \left( k u^{(m)} \right)^T C k u^{(m)} W^T \times \sum_{l=1}^{n_k} \left( \left( K_l \right)^T k^{-1} u^{(m)} \right) k^{-1} u^{(m)} K_l, \] \hspace{1cm} (55)

The right hand side is written as:

\[ b = \left( I - M U^{(m-1)} Y + C U^{(m-1)} W \right) - F_{\text{int}} \left( U^{(m-1)} \right) k u^{(m)}. \] \hspace{1cm} (56)

The space-time separated representation of the bilinear stiffness operator, as shown in equation (52) for the solution $U^{(m-1)}$, can be computed for a certain space-time solution $U = \left[ u_1 \cdots u_{n_t} \right]$ with the singular value decomposition based on the stiffness matrices of the structure at all time steps $K \left( u_1 \right), \ldots, K \left( u_{n_t} \right)$. For this purpose, we first need to vectorize each stiffness matrix $K \left( u_j \right) = [ k_{j,1} \cdots k_{j,n_t} ] \in \mathbb{R}^{n_t \times n_t}$ as follows:

\[ \text{vec} \left( K \left( u_j \right) \right) = \begin{bmatrix} k_{j,1} \\ \vdots \\ k_{j,n_t} \end{bmatrix}, \] \hspace{1cm} (57)

with $\text{vec} : \mathbb{R}^{n_t \times n_t} \rightarrow \mathbb{R}^{n_t^2}$ and $\text{vec}^{-1} : \mathbb{R}^{n_t^2} \rightarrow \mathbb{R}^{n_t \times n_t}$. The operator $\text{vec}(-)$ creates one column vector from a matrix $K$ by stacking its column vectors below one another. Then, the singular value decomposition of the matrix $Z \left( U \right) \in \mathbb{R}^{n_t^2 \times n_t}$, composed of the vectorized stiffness matrices of all time steps, yields:

\[ Z \left( U \right) = \left[ \text{vec} \left( K \left( u_1 \right) \right) \cdots \text{vec} \left( K \left( u_{n_t} \right) \right) \right] = \hat{U} \Sigma \hat{V}^T, \] \hspace{1cm} (58)

where the matrix $\hat{U} \in \mathbb{R}^{n_t^2 \times n_t}$ contains the left singular vectors, the diagonal matrix $\Sigma = \text{diag} \left( \sigma_1, \ldots, \sigma_{n_t} \right)$ contains the decreasingly ordered singular values, and the orthogonal matrix $\hat{V} \in \mathbb{R}^{n_t \times n_t}$ contains the right singular vectors. Using the singular value decomposition (58), an optimal low rank approximation $Z_{\text{app}}$ with rank $n_k$ of the matrix of vectorized stiffness matrices $Z \left( U \right)$ can be built as a sum of $n_k$ rank one matrices:

\[ Z_{\text{app}} = \sum_{l=1}^{n_k} \sigma_l \hat{u}_l \otimes \hat{v}_l, \quad \text{with } n_k < n_t, \] \hspace{1cm} (59)

where the vectors $\hat{u}_l$ and $\hat{v}_l$ are the first $n_k$ column vectors of the matrices $\hat{U}$ and $\hat{V}$, respectively, and are associated with the largest singular values $\sigma_l$. The left and right singular vectors and the corresponding singular values used in the approximation (59) enable to build a low-rank approximation
Fig. 1 Computation of a low rank space-time separated representation of the stiffness operator

\[ \sum_{l=1}^{n_k} K^{(l)}_{s} \otimes K^{(l)}_{t} = \sum_{l=1}^{n_k} \text{vec}^{-1}(\hat{u}_l) \otimes \text{diag}(\sigma_l \hat{v}_l), \]  

(60)

which is required for the implementation of algorithm 2. Note, that the inverse operator \( \text{vec}^{-1}(\cdot) \) of the vectorization operator \( \text{vec}(\cdot) \) transforms back a column vector \( \hat{u}_l \in \mathbb{R}^{n_s^2} \) into a matrix \( K^{(l)}_{s} \in \mathbb{R}^{n_s \times n_s} \) and the operator \( \text{diag}(\cdot) \) builds a diagonal matrix \( K^{(l)}_{t} \in \mathbb{R}^{n_t \times n_t} \) using the entries of a vector \( \sigma_l \hat{v}_l \in \mathbb{R}^{n_t} \). The steps given in equations (57) to (60) for computing a space-time separated representation of the stiffness operator for a certain space-time solution \( U \) are graphically illustrated in Fig. 1. The computation time and memory requirements of these operations, especially the singular value decomposition, become prohibitive for models with a large number of degrees of freedom \( n_s \), because the number of rows of the matrix \( Z(U) \) increases quadratically with \( n_s \). Since the stiffness matrices of finite element models are not fully populated sparse matrices with the same sparsity pattern, we propose to only consider the nonzero entries of the stiffness matrices in the vectorization step (57). The number \( n_{nz} \) of nonzero entries of a stiffness matrix \( K(u) \in \mathbb{R}^{n_s \times n_s} \) is, in general, significantly smaller than \( n_s^2 \), especially for large
finite element models, which yields a considerably smaller matrix $Z(U) \in \mathbb{R}^{n_m \times n_t}$ and, therefore, a more efficient but equally valuable singular value decomposition step (58).

### 4 Numerical examples

In this section, we investigate the ability of the proposed PGD solver presented in algorithm 2 to approximate the solution of a nonlinear problem with a space-time separated representation. For this purpose, a two-dimensional finite element model of a cantilever beam is built using a three-node geometrically exact beam element, presented in Appendix A. The cantilever beam has a length $l = 1$ m and a square cross-section with a side length of 0.02 m. Its finite element model is composed of 10 elements and 21 nodes, as illustrated in Fig. 2. Each node is associated with two translational and one rotational degree of freedom. The fixation of the translational and rotational degrees of freedom of the left-most node is performed by eliminating these variables and the equilibrium equations associated with them from the global system of equations so that the mechanical system contains 60 degrees of freedom in total. The single moment at the right-most node is applied by prescribing the value of the moment at the corresponding entry of the global vector of external forces $f_{ext}$. The considered hyperelastic Saint Venant–Kirchhoff material model is characterized by a Young’s modulus $E = 210$ GPa, a Poisson’s ratio $\nu = 0.3$ and a density $\rho = 7800$ kg/m$^3$.

In order to compare between different approximations $U_{app}$ of the solution, the following relative error measure with respect to a reference solution $U_{ref}$ is introduced:

$$RE = \frac{1}{n_t} \cdot \frac{\| U_{app} - U_{ref} \|}{\| U_{ref} \|} \cdot 100\%,$$  \tag{61}$$

where the operator $\| \cdot \|$ defines the Frobenius norm of a matrix. The averaging with the number of time steps $n_t$ in the formula (61) of $RE$ is necessary when simulations with a different number of time steps are compared since errors accumulate. In addition, the observation of the displacements of the right-most node of the model over time is provided to investigate the accuracy of the PGD solutions with respect to a reference solution.

Solving the equations of motion of the static problem in the nonlinear regime requires the same procedure as for the dynamic problem since the external forces should preferably change in sufficiently small increments to achieve convergence, which makes it also a time-dependent problem. The only difference is that inertia and damping forces drop out from the equilibrium equation in the static case, and the residual is composed only of internal and external forces. Therefore, we will first consider a geometrically nonlinear static problem in Sect. 4.1 and then we will investigate a geometrically nonlinear dynamic problem in Sect. 4.2. In both numerical demonstrations, we present the cantilever beam model, as illustrated in Fig. 2.

#### 4.1 Geometrically nonlinear static problem

For the static problem, represented by the cantilever beam model shown in Fig. 2, a system of nonlinear equations of the form (13) has to be solved for a number of time steps with a stepwise increasing external moment. Within the PGD solver in algorithm 2, the norm of the matrix composed of the residual vectors of all time steps (44) has to be minimized by adding new enrichments to the space-time solution $U$. The magnitude of the external moment applied at the right-most node is increased linearly with respect to time according to the following increment:

$$\Delta M = M_s \Delta t \text{ N} \cdot \text{m},$$  \tag{62}$$

where $M_s$ defines the rate of increase of the external moment. The solution of the static problem, illustrated in Fig. 3 at different time points, is computed with an incremental Newton–Raphson solver for a time interval $\Delta T = [0, 15]$ s with a time step $\Delta t = 0.01$ s and $M_s = 10^3$ N · m · s$^{-1}$. It will be used as a reference solution in the assessment of the accuracy of the solutions computed with the PGD solver.

#### 4.1.1 Convergence behavior and performance assessment

The numerical simulations discussed in this subsection are performed by the numerical setup presented in Table 2. The PGD solver presented in algorithm 2 is implemented to solve a static problem with $n_s = 60$ degrees of freedom and $n_t = 150$ time steps. The initial solution $U^{(0)}$ is assumed to be the undeformed configuration $u = 0$ for all time steps. The termination of the enrichment loop is only controlled by setting a maximum number of enrichments $m_{max}$ and not with a tolerance $\epsilon_{res}$ since we are interested in analyzing the convergence behavior when new enrichments are added to the current solution. The fixed point loop is terminated either when the maximum number of iterations $k_{max}$ is reached, or when an absolute tolerance $\epsilon_{fp, abs}$ and a relative tolerance
Table 1  Numerical setup of the PGD algorithm for analyzing its convergence behavior and performance for solving a static problem

| Parameter | Value |
|-----------|-------|
| $n_s$     | 60    |
| $n_t$     | 150   |
| $\Delta t$ [s] | $10^{-2}$ |
| $k_{\text{max}}$ | 20 |
| $U^{(0)}$ | $0 \ u_s^{(m)}$ |
| $u_l^{(m-1)}$ | $u_l^{(m-1)}$ |
| $\epsilon_{fp,\text{abs}}$ | $10^{-3}$ |
| $\epsilon_{fp,\text{rel}}$ | $10^{-3}$ |
| $\epsilon_{\text{res}}$ | 0 |

Varied parameters

$m_{\text{max}}$

{1, \ldots, 25}

Table 2  Numerical setup of the PGD algorithm for analyzing its performance for problems with a larger number of degrees of freedom

| Parameter | Value |
|-----------|-------|
| $m_{\text{max}}$ | 5 |
| $n_t$ | 150 |
| $\Delta t$ [s] | $10^{-2}$ |
| $k_{\text{max}}$ | 20 |
| $U^{(0)}$ | $0 \ u_s^{(m)}$ |
| $u_l^{(m-1)}$ | $u_l^{(m-1)}$ |
| $\epsilon_{fp,\text{abs}}$ | $10^{-3}$ |
| $\epsilon_{fp,\text{rel}}$ | $10^{-3}$ |
| $\epsilon_{\text{res}}$ | 0 |

Varied parameters

$n_s$

{60, \ldots, 1800}

![Fig. 3](image)  Reference solution at different time instants of the static problem computed with an incremental Newton–Raphson solver

![Fig. 4](image)  a Representation of the PGD solutions with different numbers of enrichments and the Newton–Raphson reference solution at the last time step ($n_t = 150$); b Displacement in x-direction of the right-most node of the cantilever beam of the PGD solutions with different numbers of enrichments and the Newton–Raphson reference solution
The configurations of the cantilever beam corresponding to the PGD solution with different numbers of enrichments and the Newton–Raphson reference solution are shown in Fig. 4a after the last external moment increment ($n_t = 150$). The configuration PGD-1 in red of the cantilever beam represents the last column of the rank one PGD solution $U^{(1)}$, which means that the solution includes only the first enrichment $u_s^{(1)} \otimes u_t^{(1)}$. This first solution is nothing else than the solution of the linearized system around the initial undeformed configuration since the vectors of internal forces of all time steps are linearized around the initial solution $U^{(0)} = 0$ in the first iteration $m = 1$. Note that, in this case, the spacetime separated representation of the stiffness operator is only composed of one summand $K(u = 0) \otimes I_{n_t}$, where $I_{n_t}$ is the identity matrix of size $n_t$. The first space and time modes are illustrated in the top of Fig. 5a, b, respectively. The first time mode shows that the response, represented by the first enrichment, is proportional to the external moment and is, therefore, a linear response. The configuration PGD-2 in orange represents the last column of the rank two PGD solution $U^{(2)}$ and includes already the geometrically nonlinear behavior of the structure by considering the linearization of the internal force vector of each time step around the corresponding configuration within the previous solution matrix $U^{(1)}$. The solution $U^{(2)}$ is the sum of the outer products of the two first space modes with their respective time modes, illustrated in Fig. 5. The same procedure is then repeated until a convergence criterion is achieved. Figure 4a shows that the PDG solution converges towards the Newton–Raphson reference solution presented by the black line when a larger number of enrichments is considered. The plot in Fig. 4b illustrates the time evolution of the displacement in $x$-direction of the rightmost node, which undergoes the largest displacements in the whole cantilever beam. It demonstrates that the nonlinear response of the structure in $x$-direction is already accurately represented after the second enrichment. Figure 5 represents the normalized space and time modes. It is shown that the contribution of new enrichments to the solution becomes less significant with increasing enrichment, while the temporal evolution becomes more complex.

The convergence behavior of the solution of the static problem computed with the proposed nonlinear PGD solver can also be shown quantitatively by observing the evolution of the relative error measure in Fig. 6a or the Frobenius norm of the residual matrix in Fig. 6b with respect to the number of enrichments. The relative error is computed in accordance with formula (61), where the PGD solution $U^{(m)}$ after the $m$-th enrichment is inserted as $U_{app}$, and the highly accurate Newton–Raphson solution is denoted as $U_{ref}$. The
The convergence of the residual norm of the solution, computed by algorithm 2, is, in general, non-monotonic because it is based on a Galerkin formulation, as discussed in Sect. 3.1 [39,69].

The total computation time needed by the PGD solver increases linearly with respect to the number of enrichments, as shown in the plot of Fig. 6a, although the number of fixed-point iterations within each enrichment differs significantly. This phenomenon is caused by the fact that the evaluation of the residual vectors and the stiffness matrices at $U^{(m-1)}$ during the linearization step, which is shown in line 4 of algorithm 2 and depends only on the size of the finite element model and the number of time steps, take the major part of the computational effort (97.2%). In comparison, the space-time decomposition of the stiffness operator and the space and time problems solved within the fixed point iterations take a negligible part of the total computational time.

For the numerical setup in Table 2, the plot 6a shows that the computation of the PGD solution with two enrichments ($m_{\text{max}} = 2$) is approximately two times faster than the computation of the reference Newton–Raphson solution, which is represented by the horizontal dashed green line, while it exhibits a lower level of accuracy. The PGD solution based on three enrichments requires as much computation time as the Newton–Raphson solution, and starting from the fourth enrichment, it becomes slower. Therefore, the enhancement of the linearization step can reduce the slope of the green linear curve and pave the way for more efficient computations with the proposed nonlinear PGD solver.

4.1.2 Influence of the number of degrees of freedom

Since the number of degrees of freedom of our benchmark model is relatively small we investigate the efficiency of the proposed algorithm with an increasing number of degrees of freedom. The parameter setting of this study is presented in Table 2. As shown in this table, we increase the number of degrees of freedom of the cantilever beam problem from 60 to 1800. The computational time versus the number of degrees of freedom is depicted in Fig. 7. As shown in this figure, one observes a linear increase in computation time when increasing the number of degrees of freedom. We note that this linear relation can be achieved since we apply a vectorization operator on the sparse tangential stiffness matrices in Eq. (60) that considers only the non-zero entries. This approach reduces the memory requirements of the space-time stiffness matrix $Z(U)$ considerably and allows for a significantly less expensive decomposition of the space-time stiffness matrix $Z(U)$ using the singular value decomposition, as presented in Eq. (58).

4.1.3 Influence of the number of time steps and the size of external force increments

In contrast to the incremental Newton–Raphson solver, the convergence behavior of a PGD solver depends on the number of time steps considered in the computation. In order to illustrate this dependency, we performed different computations of static problems with different numbers of time steps $n_t$ for a particular external moment increment size. The same computation was then repeated with external moment incre-
Table 3  Numerical setup of the PGD algorithm for the static problem in order to analyze the influence of the number and size of external moment increments

Fixed parameters

| ns | Δt [s] | kmax |
|----|--------|------|
| 60 | 10⁻²   | 20   |

| ϵfp,abs | ϵfp,rel | ϵres |
|---------|---------|------|
| 10⁻³    | 10⁻³    | 8n₁ |

Varied parameters

\[ M_s \text{[N·m·s⁻¹]} \]
\{1·10³, 2·10³, 4·10³\} \quad \{10, \ldots, 180\}

ments \( ΔM \) of different sizes. We present the numerical setup of the performed numerical experiments in Table 3.

The termination criterion of the PGD solver is only based on reaching a certain residual norm \( \| R \| < \epsilon_{res} \), where the tolerance \( \epsilon_{res} \) is proportional to the number of time steps and, therefore, to the size of the matrix \( R \). The termination criterion does, therefore, not depend on the considered number of time steps. The plots in Fig. 8 show that the computation time of PGD solutions increases exponentially with respect to the number of considered time steps, while it increases linearly for an incremental Newton–Raphson solver. This phenomenon is because the convergence of the PGD solver worsens with an increasing number of time steps and is even lost at a certain point, whereas a higher number of time steps does not influence the convergence behavior of an incremental Newton–Raphson solver. It should also be mentioned that when the size of the external moment increment is larger, the required computation time increases faster with an increasing number of time steps. The numerical results, depicted in Fig. 8a, indicate that the convergence is lost in all considered three cases when the sum of all external moment increments of the static problem reaches nearly the same value of \( M = 1.8 \cdot 10^3 \text{ N·m} \). For this reason, the PGD solver in algorithm 2 can not be implemented for solving the static problem illustrated in Fig. 3 within the whole time domain at once using only one space-time decomposition with \( n_t = 1500 \). Thus, in the next section, we propose a piecewise formulation of the PGD solver that operates in a sequence of smaller time intervals. Within each time interval, the time modes have a smaller size, and convergence can be ensured.

4.1.4 A piecewise PGD formulation

In this section, we present a piecewise PGD formulation for solving problems over large time intervals, such as the static problem presented in Fig. 3. The motivation of this formulation is to ensure the convergence of the solution and minimize computation time by solving the problem sequentially on a number of smaller subintervals. We propose to divide the
overall time interval of \( n_t = 1500 \) time steps into a number of \( n_p \) subintervals with \( n_{ts} = n_t / n_p \) time steps, so that a space-
time solution \( \mathbf{U} \in \mathbb{R}^{n_s \times n_{ts}} \) is computed in each subinterval 
\( h = 1, \ldots, n_p \). The first and last columns of the matrix \( h \mathbf{U} \) 
correspond to the solutions at time steps \( j = n_{ts} (h-1) + 1 \) 
and \( j = n_{ts} h \), respectively. The plot in Fig. 9 shows the evo-
lution of the required computation time to solve the whole static problem of \( n_t = 1500 \) time steps when the size \( n_{ts} \) of 
the subintervals is varied. The numerical setup of the computa-
tions is presented in Table 4. The termination criterion of 
the enrichment loop is only based on reaching a certain resi-
dual matrix norm, which is proportional to its size, so that all 
the considered computations exhibit approximately the same 
accuracy. The initialization of all columns of the space-time 
solution \( \mathbf{U}^{(0)} \) of the first subinterval is defined as the zero 
vector associated with the initial undeformed configuration. 
All the columns of the space-time solution \( h \mathbf{U}^{(0)} \) of the sub-
sequent subintervals are initialized with the solution \( h^{-1} \mathbf{u}_{n_{ts}} \) 
at the last time step of the corresponding previous subinterv-
ual. We conclude from the plot in Fig. 9 that the piecewise 
PGD formulation is more efficient for a larger number \( n_p \) of 
subintervals, which have a smaller number of time steps \( n_{ts} \). 
This conclusion is in accordance with the behavior observed 
in Fig. 8. Although the overall time interval has to be divided 
into subintervals, this method enables solving problems with 
an arbitrarily large time interval using the PGD solver in 
algorithm 2.

### Table 4 Numerical setup of the piecewise PGD formulation for solv-
ing the static problem with a large number of time steps \((n_t = 1500)\) illustrated in Fig. 3

| Fixed parameters |   |   |   |
|------------------|---|---|---|
| \( n_s \)       | \( n_t \) | \( \Delta t \) | \( \times 10^{-2} \) |
| 60               | 1500 | 10^{-2} |
|                  |      | \( h \mathbf{U}^{(0)} \) | \( h \mathbf{U}^{(0)} \) |
|                  |      | \( h^{-1} \mathbf{u}_{n_{ts}} \) | \( h^{-1} \mathbf{u}_{n_{ts}} \) |
|                  |      | \( k_{max} \) |   |
|                  |      | \( m_{max} \) |   |
|                  |      | \( 10^{3} \) |   |

| Varied parameters | \( n_{ts} \) |
|                  | \( 5, \ldots, 150 \) |

### Table 5 Numerical setup of the PGD algorithm for the harmonic 
excitation problem in order to analyze its convergence behavior and 
performance

| Fixed parameters | \( n_s \) | \( n_t \) | \( \Delta t \) | \( \times 10^{-3} \) | \( k_{max} \) |
|------------------|---|---|---|---|---|
| 60               | 300 | 5 \times 10^{-3} | 20 |
|                  | \( U^{(0)} \) | \( 0 \) | \( 0 \) | \( 0 \) | \( M_0 \) |
|                  | \( \epsilon_{fp, abs} \) | \( \epsilon_{fp, rel} \) | \( \epsilon_{res} \) | \( \omega \) | \( \times 10^{-3} \) |
|                  | \( f \) | \( f \) | \( 10^{-3} \) | \( 8 n_{ts} \) | |
|                  | \( M_f \) | \( M_f \) |
|                  | \( 10^{3} \) | \( \infty \) |

| Varied parameter | \( m_{max} \) |
|                 | \( 1, \ldots, 25 \) |

### 4.2 Geometrically nonlinear dynamic problem

In this section, we present the numerical demonstration of 
the PGD approach using the cantilever beam geometry, as 
shown in Fig. 2, subjected to dynamic excitations. In doing 
so, we present the applicability of the new strategy consider-
ing two different excitation types, the harmonic excitation 
problem in Sect. 4.2.1 and the white noise excitation problem 
in Sect. 4.2.2.

#### 4.2.1 Harmonic excitation problem

In order to solve the dynamic problem associated with the 
cantilever beam model in Fig. 2, the equations of motion (9) 
have to be satisfied at all time steps. Using the PGD solver in 
algorithm 2, the norm of the matrix composed of the residual 
vectors of all time steps (47) is minimized by adding new 
enrichments to the space-time solution \( \mathbf{U} \). The values of the external 
moment, applied at the right-most node, are defined with 
the following sinusoidal function \( M(t) = M_0 \sin \omega t \).

The equations of motion are discretized in time using the 
Newmark integration scheme, defined by the equations (4) 
and (5), choosing \( \beta = 0.25 \) and \( \gamma = 0.5 \) for the New-
mark parameters. The integration of the time discretization 
using the PGD formulation is discussed in Sect. 3.2. 
The damping matrix \( C \) is defined according to the Rayleigh damp-
ing approximation (2). The weighting factors \( \alpha_d \) and \( \beta_d \) are 
adjusted, so that the first two eigenfrequencies of the lin-
earized problem around the undeformed configuration have 
a damping ratio of \( \xi = 0.005 \).

The numerical simulations discussed in this section 
demonstrate the ability of the proposed PGD solver to solve 
geometrically nonlinear dynamic problems. They are per-
duced in accordance with the numerical setup presented in 
Table 5. For the considered dynamic problem, the motion of 
the cantilever beam, consisting of \( n_s = 60 \) degrees of free-
Fig. 10  Displacement in the $y$-direction of the right-most node of the cantilever beam corresponding to the PGD solutions with different numbers of enrichments and to the Newton–Raphson reference solution of the harmonic excitation dynamic problem.

Fig. 11  Representation of the first five space modes (a) and their respective time functions (b), or also called time modes, corresponding to the solution of the harmonic excitation problem with the numerical setup in Table 5. They constitute the enrichment couples computed in the first five iterations $m = \{1, \ldots, 5\}$.

The plot in Fig. 10 shows the evolution over time of the displacement of the right-most node of the cantilever beam in the $y$-direction computed using the proposed PGD solver.
and the incremental Newton–Raphson solver. A PGD-$m_{\text{max}}$ solution includes a number of $m_{\text{max}}$ enrichments. The illustrated displacement is one of the 60 degrees of freedom of the considered finite element model. Its evolution over time is one part of the total solution and is extracted from the corresponding row in the PGD solution matrix $U$. The red curve PGD-1 is extracted from the rank one solution matrix $U^{(1)}$, which is the outer product of the first space mode $u_{s}^{(1)}$ and its corresponding time mode $u_{t}^{(1)}$, shown in the top of Fig. 11. The orange curve PGD-2 and the green curve PGD-3 in Fig. 10 are extracted from the PGD solution matrices $U^{(2)}$ of rank two and $U^{(3)}$ of rank three, which are defined as a sum of the two first enrichments and the three first enrichments, respectively. Convergence of the PGD solution towards the Newton–Raphson reference solution in black is observed when a larger number of enrichments is considered. It should be mentioned that a faster convergence of the solution is achieved in the periodic steady-state at the end of the time interval than during the transient behavior at the beginning. The space and time modes corresponding to the first five enrichments are illustrated in Fig. 11. Note that the space modes are normalized, so that the extent of the contribution of a certain enrichment to the total solution is represented by the time modes. With some exceptions, the contribution of a new computed enrichment has, in general, a smaller magnitude than the previous enrichments.

The convergence behavior of the solution of the dynamic problem computed with the proposed nonlinear PGD solver can also be shown quantitatively by observing the evolution of the relative error measure in Fig. 12a or the Frobenius norm of the residual matrix in Fig. 12b with respect to the number of enrichments. The convergence in residual norm is non-monotonic since the proposed nonlinear PGD solver is based on the Galerkin formulation. However, the convergence of the relative error measure shown in blue in figure 12a exhibits monotonic behavior. The relative error measure is computed in accordance with the formulation equation (61), where the PGD solution $U^{(m)}$ after the $m$-th enrichment is inserted as $U_{\text{app}}$ and a highly accurate Newton–Raphson solution is used as $U_{\text{ref}}$.

The evolution of the computation time, illustrated by the green line in Fig. 12a, is linear with respect to the number of enrichments and exhibits similar behavior as in the static problem since the largest part of the computational effort is required for the evaluation of the stiffness matrices and internal force vectors. The computational time of these evaluations is approximately the same in all enrichment steps because they are, in general, independent from the considered current solution $U^{(m-1)}$. For the numerical setup in Table 5, Fig. 12a shows that the computation of the PGD solution with two enrichments ($m_{\text{max}} = 2$) is more than twice as fast as the computation of the reference Newton–Raphson solution, which is represented by the horizontal dashed green line, while it exhibits a lower level of accuracy. The PGD solution based on three enrichments is slightly faster than the Newton–Raphson solution, and starting from the fourth enrichment, it
becomes slower. The achieved speedups in solving nonlinear dynamic problems with the proposed nonlinear PGD solver can be increased by making the evaluations of the stiffness matrices and the internal force vectors more efficient.

4.2.2 White noise excitation problem

The cantilever beam is excited by a concentrated load, as depicted in Fig. 13a. The time history is realized as a stationary Gaussian random white noise [71], as depicted in Fig. 13b. The required parameters for the calculation are presented in Table 6.

The solution of the transient excitation problem using the proposed PGD approach is depicted in Fig. 14 and compared to the classical step-by-step Newton–Raphson procedure. After one enrichment the PGD solution (red line) converges to a time history that is very similar to the geometrically linear response (gray dashed line). At this stage of the algorithm, the solution is spanned by the dyadic product of the first space and time enrichment modes, presented in the first row of Fig. 15. One observes a high correlation of the space mode with the first mode of vibration while the time mode reveals the time history of the linear response function. The linear solution, presented by the gray dashed line in Fig. 14, is, in this example, essentially governed by the response of the lowest vibration mode, which is, therefore, responsible for the highest amount of the vibration energy of the geometrically linear system. Since the PGD solution almost coincides with the geometrically linear solution after the first enrichment one observes a high similarity of the response with the modal truncated representation considering only the lowest linear mode of vibration. We further present the convergence behavior of the PGD algorithm in Fig. 14, plotting the solution after two, three, and 20 enrichments, in orange, green and blue color, respectively. One observes very good agreement with the Newton–Raphson solution after 20 enrichments.

![Fig. 14](image-url) Displacement in the y-direction of the right-most node of the cantilever beam corresponding to the PGD solutions with different numbers of enrichments and to the Newton–Raphson reference solution of the white noise vibration problem.
Fig. 15 Representation of the first five space modes (a) and their respective time functions (b), or also called time modes, corresponding to the solution of the white noise vibration problem with the numerical setup in Table 5. They constitute the enrichment couples computed in five different iterations \( m = [1, 5, 10, 15, 20] \).

Additionally, we present the space and time modes after the fifth, tenth, 15th, and 20th enrichment in the second, third, fourth, and fifth row of Fig. 15, respectively. Looking at the time modes in this figure, one clearly observes that the participation of the enrichments to the full solution decreases significantly with increasing enrichment, although we again emphasize here that the convergence behavior of the PGD algorithm is not strictly monotonic decreasing with respect to the Frobenius norm of the residual matrix, as presented in Fig. 12b.

5 Conclusion

In this paper, a proper generalized decomposition based solver for nonlinear dynamic and static structural problems is proposed.

The proper generalized decomposition based solver builds the solution of an initial boundary value problem in the whole time domain as sum of rank one space-time enrichments in an iterative process. Within every iteration a space mode and its corresponding time mode are computed by solving two coupled nonlinear problems, one in space and the other in time, in an alternating manner using the fixed point algorithm until convergence is achieved.

The performed numerical examples show that the proposed PGD solver achieves a limited speedup in comparison with the incremental Newton–Raphson solver when a small number of enrichments is considered. One key feature of a successful numerical speedup is the consideration of the sparsity of the tangential stiffness matrix during the vectorization operation of the stiffness matrix in Equation (57), which is required to obtain a space-time stiffness formulation. The computational bottleneck causing the limitation of the proposed approach to date is mainly the evaluation of the stiffness matrices and the internal force vectors during the linearization step, which is also a key feature in classical model order reduction approaches that follow the step-by-step scheme. However, the proposed method follows an algorithmic approach that differs significantly when compared to classical step-by-step methods and the possibilities regarding numerical speed-up and applicability to various types of mechanical problems are yet to discover.

Thus, the next step should be the integration of strategies into the proposed algorithm, which decreases the computational effort by reducing the number of required costly stiffness matrix evaluations while preserving a good convergence behavior. The additional application of hyperreduction strategies for a more efficient evaluation of internal force vectors and stiffness matrices is also an interesting topic for future research.
Furthermore, future research should deal with the extension of the space-time solutions to three dimensional beam structures and buckling of beams as well as problems including external loads that depend on the displacements. It is of high interest if the space-time algorithm can also be applied if such challenging types of nonlinearities are present or if additional adaptions may be required to ensure stable convergence. This ambitious enterprise should not only include extensions or adaptions of the proposed space-time algorithm but also the implementation of a numerically efficient framework for practical applications.

Finally, we note that the new method should be compared to existing model reduction strategies for geometrically nonlinear problems, as presented by Sonneville et al. [7]. This objective demands rigorous numerical investigations that are beyond the scope of this paper, however, they are on the agenda for future research projects.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

**Appendix A: Two-dimensional geometrically exact beam element**

In the following, we present the theory of the two-dimensional geometrically exact beam proposed in [72] and its finite element discretization, discussed in [73], for initially straight beam axis. This theory is based on the assumption that plane cross sections remain plane and takes into account the transverse shear deformation, which means that the cross sections do not necessarily remain normal to the axis of the beam. It results in nonlinear strain displacement relations and is valid for finite deflections, rotations, and strains. In many applications, it is sufficient to consider small strains, even when deflections and rotations are large, so that the Saint Venant–Kirchhoff constitutive equations can be used. This hyperelastic constitutive law defines linear stress-strain relations and can be considered as an extension of Hooke’s law of linear theory of elasticity to the geometrically nonlinear regime. The nonlinear kinematics of the beam are shown in Fig. 16. In the undeformed reference configuration $B_0$ the local axis $X$ coincides with the initially straight beam axis. The beam can exhibit an arbitrary initial inclination $\Theta$ with respect to the global axis $X_g$.

In the current configuration $B$, the position of each material point is expressed with respect to the local coordinate system by the following mapping function:

$$\varphi = \begin{bmatrix} X + \bar{u}(X) \\ \bar{w}(X) \end{bmatrix} + Y \begin{bmatrix} -\sin \bar{\psi}(X) \\ \cos \bar{\psi}(X) \end{bmatrix},$$

where the three independent fields $\bar{u}(X)$, $\bar{w}(X)$ and $\bar{\psi}(X)$ are the displacement in axial direction, the deflection and the rotation of the cross section, respectively. The fields are arranged in a vector $\bar{d} = [\bar{u}, \bar{w}, \bar{\psi}]^T$, which can be transformed to the global coordinate system using a rotation matrix as:

$$\bar{u} = (T(\Theta))^T \bar{d},$$

where

$$\bar{u} = \begin{bmatrix} \bar{u}_g \\ \bar{w}_g \\ \bar{\psi}_g \end{bmatrix},$$

and

$$T(\Theta) = \begin{bmatrix} \cos \Theta & \sin \Theta & 0 \\ -\sin \Theta & \cos \Theta & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$
\[ \epsilon = T(\psi) \delta \vec{d} - \phi, \]  

where \((\cdot)’\) denotes the partial derivative with respect to the coordinate \(X\) and the following matrices and vectors are introduced:

\[
\epsilon = \begin{bmatrix} e \\ \gamma \\ \kappa \end{bmatrix}, \quad T(\psi) = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \delta \vec{d} = \begin{bmatrix} \delta \bar{u} \\ \delta \bar{u}’ \\ \delta \psi \end{bmatrix}, \quad \phi = \begin{bmatrix} 1 - \cos \psi \\ \sin \psi \\ 0 \end{bmatrix}. \tag{A7}
\]

The linear hyperelastic constitutive law, Saint Venant–Kirchhoff model, yields the following cross-sectional force vector:

\[ S = D\epsilon, \tag{A8} \]

with

\[
S = \begin{bmatrix} N \\ Q \\ M \end{bmatrix}, \quad D = \begin{bmatrix} EA & 0 & 0 \\ 0 & GA_s & 0 \\ 0 & 0 & EI \end{bmatrix}. \tag{A9}
\]

The entries of the force vector \(S\) are the axial force \(N\), the shear force \(Q\) and the bending moment \(M\). The first diagonal entry of the matrix \(D\) is the product of the Young’s modulus \(E\) and the cross section area \(A\), which describes the axial stiffness of the beam. The second diagonal entry is the product of the shear modulus \(G\) and the shear-corrected cross section area \(A_s\), which defines the shear stiffness of the beam. A shear-corrected cross section area is introduced in order to correct the violation of the boundary condition at \(Y = \pm h/2\), where the shear stresses should vanish. The violation of this boundary condition is caused by the assumption that plane cross-sections remain plane. The third diagonal entry, the bending stiffness, is computed as the product of the Young’s modulus \(E\) and the second moment of area \(I\).

The weak form of equilibrium, which is equivalent to the principle of virtual work, is written for the considered geometrically nonlinear beam model as:

\[ \delta \Pi_{int} - \delta \Pi_{ext} = \int_0^l \delta \epsilon^T S \, dX - \int_0^l \delta \vec{d}^T \cdot \mathbf{g} \, dX = 0, \tag{A10} \]

where the virtual displacements:

\[ \delta \vec{d} = \begin{bmatrix} \delta \bar{u} \\ \delta \bar{u}’ \\ \delta \psi \end{bmatrix}, \tag{A11} \]

and the virtual strains:

\[ \delta \epsilon = T(\psi) \delta \vec{d}’ + \frac{\partial T(\psi)}{\partial \psi} \delta \vec{d} - \frac{\partial \phi}{\partial \psi} \delta \psi. \tag{A12} \]

are introduced. The vector \(\mathbf{g}\) is the vector of external forces and moments.

The spatial discretization is performed by the finite element method using quadratic Lagrangian shape functions [73]. The three quadratic shape functions associated with the three nodes of the beam element are expressed by the local coordinate \(\xi \in [-1, 1]\) in the parameter space as:

\[ N_1(\xi) = \frac{1}{2} \xi (\xi - 1), \quad N_2(\xi) = \frac{1}{2} \xi (1 + \xi), \quad N_3(\xi) = \left(1 - \xi^2\right). \tag{A13} \]

The mapping between the physical space and the parameter space is illustrated in figure 17. The geometry \(X\) and the displacement fields \(\bar{d}\) are both discretized using the shape functions (A13) as:

\[ X(\xi) = \sum_{i=1}^3 N_i(\xi) X_i^e, \tag{A14} \]

and

\[ \bar{d}(\xi) = \sum_{i=1}^3 N_i(\xi) d_i^e = \sum_{i=1}^3 N_i(\xi) \begin{bmatrix} u_i^e \\ w_i^e \\ \psi_i^e \end{bmatrix}, \tag{A15} \]

where \(X_i^e\) is the initial position of node \(i\) along the \(X\) axis of the local coordinate system of an element \(e\) and \(d_i^e \in \mathbb{R}^3\) is the corresponding vector of local nodal quantities of the three displacement fields, which are the primary variables in this formulation. The discretized displacement fields can also be written as:

\[ \bar{d}(\xi) = N^e d^e \]

\[ Y \quad \text{Physical space} \]

\[ \begin{array}{ccc}
\text{node 1} & \text{node 3} & \text{node 2} \\
0 & 1 & 2 \\
X_1^e = 0 & X_3^e = l_e/2 & X_2^e = l_e
\end{array} \]

Fig. 17 The mapping between the parameter space and the physical space of a beam element \(e\) with three nodes.
where the vectors of nodal displacements $d_1^e, d_2^e$ and $d_3^e$ of the three nodes are concatenated in a column vector $d^e \in \mathbb{R}^9$ and the matrix $I \in \mathbb{R}^{3 \times 3}$ denotes the identity matrix. Inserting the discretizations (A15) into the internal virtual work of element $e$ yields:

$$\delta \Pi_{int} = \int_0^{\epsilon^e} (\delta \epsilon)^T S \, dX = (\delta d^e)^T f_{int}^e (d^e),$$  

(A17)

where the internal force vector $f_{int}^e$ is a nonlinear function of the nodal displacements $d^e$. The evaluation of the integrals in equation (A17), which are required for the computations of $f_{int}^e$, are performed by Gaussian quadrature in parameter space. The vector of external forces $f_{ext}^e$ is computed in an analogous manner using the external virtual work.

In the dynamic case, the virtual work of the inertia forces $\delta \Pi_{kin}$ is included in the weak form of equilibrium as:

$$\delta \Pi_{kin} + \delta \Pi_{int} - \delta \Pi_{ext} = 0.$$  

(A18)

The virtual work of the inertia forces associated with an element $e$ is expressed as:

$$\delta \Pi_{kin} = \int_0^{\epsilon^e} \rho_0 \left( \frac{\partial d}{\partial t} \right)^T A \ddot{d} \, dX$$

$$= \rho_0 \int_0^{\epsilon^e} \left( \frac{\partial d^e}{\partial t} \right)^T \left( N^e \right)^T A \, N^e \ddot{d} \, dX$$

$$= (\delta \dot{d}^e)^T M^e \ddot{d}^e.$$  

(A19)

where $\rho_0$ is the constant density in initial configuration, the diagonal matrix $A \in \mathbb{R}^{3 \times 3}$ is given by $A = \text{diag}(A, A, I)$ and the vector of nodal accelerations $\ddot{d}^e$ are the second time derivatives of the nodal displacements $d^e$. The elemental mass matrix $M^e$ is also computed by Gaussian quadrature.

The virtual work of a whole beam structure (A18) is composed of the virtual work contributions of all elements. The assembly of the discretized version of these contributions leads to:

$$\delta u^T (M \ddot{u} + f_{int}(u) - f_{ext}) = 0,$$  

(A20)

with

$$M = \bigcup_{e=1}^{n_e} M^e_{g} \ddot{u}$$

$$f_{int}(u) = \bigcup_{e=1}^{n_e} f_{int,g}^e (u^e)$$

$$f_{ext} = \bigcup_{e=1}^{n_e} f_{ext,g}^e (u^e).$$  

(A21)

where $u^e, M^e, f_{int,g}^e$ and $f_{ext,g}^e$ are the transformations of $d^e, M^e, f_{int}^e$ and $f_{ext}^e$, respectively, from the local coordinate system of element $e$ to the global coordinate system using the rotation matrix from equation (A4). The vectors $u, \ddot{u}, f_{int}$ and $f_{ext}$ are the global nodal displacements, nodal accelerations, internal forces and external forces, respectively. The matrix $M$ represents the global mass matrix, which is independent of the state of the beam structure and therefore constant.

Finally, the virtual work expressed in Equation (A20) must be satisfied for arbitrary variations $\delta u$ and, therefore, leads to the semi-discretized set of equations of motion:

$$M \ddot{u} + f_{int}(u) = f_{ext}.$$  

(A22)

The tangent stiffness matrix $K^e = \frac{\partial f_{int}^e}{\partial d^e}$ of element $e$ is also computed in the corresponding local coordinate system and then transformed into the global coordinate system before the assembly of the global tangent stiffness matrix, which is written as:

$$K(u) = \frac{\partial f_{int}}{\partial u} = \bigcup_{e=1}^{n_e} K^e_{g}(u^e).$$  

(A23)

The matrix $K(u)$ defines the Jacobian of the global internal forces with respect to the global nodal displacements. Note that the tangent stiffness matrix of the considered geometrically exact beam element is configuration dependent since the internal forces are nonlinear with respect to the displacements.

References

1. Bathe K-J (2007) Finite element method. Wiley encyclopedia of computer science and engineering, pp 1–12
2. Dokainish M, Subbaraj K (1989) A survey of direct time-integration methods in computational structural dynamics—i. explicit methods. Comput Struct 32(6):1371–1386
3. Subbaraj K, Dokainish M (1989) A survey of direct time-integration methods in computational structural dynamics—II. Implicit methods. Comput Struct 32(6):1387–1401
4. Ypma TJ (1995) Historical development of the Newton–Raphson method. SIAM Rev 37(4):531–551
5. Rayleigh JWSB (1896) The theory of sound, vol 2. Macmillan, London
6. Bai Z (2002) Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. Appl Numer Math 43(1–2):9–44
7. Sonneveld V, Scapelan M, Shan M, Bauchau O (2021) Modal reduction procedures for flexible multibody dynamics. Multibody Syst Dyn 51:377–418
8. Craig RR Jr, Bampton MC (1968) Coupling of substructures for dynamic analyses. AIAA J 6(7):1313–1319

© Springer
52. Newmark NM (1959) A method of computation for structural dynamics. J Eng Mech Div 85(3):67–94
53. Boucinha L, Gravouil A, Ammar A (2013) Space-time proper generalized decompositions for the resolution of transient elastodynamics. Comput Methods Appl Mech Eng 255:67–88
54. Bamer F, Shirafkan N, Cao X, Oueslati D, de Saxcé G, Markert B (2021) A newmark space-time formulation in structural dynamics. Comput Mech 67(5):1331–1348
55. Ammar A, Normandin M, Daim F, Gonzalez D, Cueto E, Chinesta F et al (2010) Non incremental strategies based on separated representations: applications in computational rheology. Commun Math Sci 8(3):671–695
56. Qin Z, Talleb H, Ren Z (2015) A proper generalized decomposition-based solver for nonlinear magnetothermal problems. IEEE Trans Magn 52(2):1–9
57. Aghighi MS, Ammar A, Metivier C, Normandin M, Chinesta F (2013) Non-incremental transient solution of the Rayleigh–Bénard convection model by using the PGD. J Nonnewton Fluid Mech 200:65–78
58. Langtangen HP, Linge S (2017) Finite difference computing with PDEs: a modern software approach. Springer Nature, Berlin
59. Cochelin B, Damil N, Potier-Ferry M (1994) The asymptotic-numerical method: an efficient perturbation technique for nonlinear structural mechanics. Revue européenne des éléments finis 3(2):281–297
60. Chinesta F, Leygue A, Beringhier M, Nguyen LT, Granddidier J-C, Schrefler B, Pesavento F (2013) Towards a framework for nonlinear thermal models in shell domains. Int J Numer Methods Heat Fluid Flow
61. Niroomandi S, Alfaro I, González D, Cueto E, Chinesta F (2013) Model order reduction in hyperelasticity: a proper generalized decomposition approach. Int J Numer Methods Eng 96(3):129–149
62. Niroomandi S, Alfaro I, Cueto E, Chinesta F (2012) Accounting for large deformations in real-time simulations of soft tissues based on reduced-order models. Comput Methods Programs Biomed 105(1):1–12
63. Shirafkan N, Bamer F, Stoffel M, Markert B (2020) Quasistatic analysis of elastoplastic structures by the proper generalized decomposition in a space-time approach. Mech Res Commun 104:103500
64. Nasri MA, Robert C, Ammar A, El Arem S, Morel F (2018) Proper generalized decomposition (pgd) for the numerical simulation of polycrystalline aggregates under cyclic loading. C R Méc 346(2):132–151
65. Gérardin M, Rixen DJ (2014) Mechanical vibrations: theory and application to structural dynamics. Wiley, Hoboken

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