Real-time direct integration of reduced solid dynamics equations

David González[1], Elías Cueto[1,*], Francisco Chinesta[2,3]

[1] *Aragón Institute of Engineering Research (I3A). Universidad de Zaragoza. Spain.*
[2] *École Centrale de Nantes. France.*
[3] *Institut Universitaire de France.*

SUMMARY

A new method is developed here for the real-time integration of the equations of solid dynamics based on the use of POD-PGD approaches and direct time integration. The method is based upon the formulation of solid dynamics equations as a parametric problem, depending on their initial conditions. A sort of black-box integrator is obtained that takes the resulting displacement field of the current time step as input and (via POD) provides the result for the subsequent time step at feedback rates on the order of 1 kHz. To avoid the so-called *curse of dimensionality* produced by the large amount of parameters in the formulation (one per degree of freedom of the full model) a combined POD-PGD strategy is implemented. Examples are included that show the promising results of this technique. Copyright © 2012 John Wiley & Sons, Ltd.

KEY WORDS: Solid dynamics, PGD, POD, model order reduction, real-time simulation.

1. Introduction

Despite the well-known (and very often mentioned) growth in computer capabilities, some engineering problems still constitute today an intensive task. Structural dynamics is one of these problems, where the set of differential equations describing the system is usually discretized in space by finite elements (leading to an ODE in time) and then explicit or implicit methods in time are used to further discretize this ODE. This results in a system of very large number of degrees of freedom if fine enough meshes are employed in space, and (specially for explicit approaches) small-enough time steps are employed to avoid instability issues.

For this reason, structural dynamics has been one of the first fields of engineering practice where model order reduction has been employed. Classical modal analysis can be seen as a sort of model order reduction while, for instance, [34] seems to have been the first in employing eigenmodes of the structure with the purpose of reducing the number of degrees of freedom in non-linear applications. In essence, (projection-based) model order reduction consists in changing the original degrees of freedom of the model (i.e., finite element mesh nodal displacements) by an as much as possible reduced number

---

*Correspondence to: Elías Cueto. Universidad de Zaragoza. Edificio Betancourt. María de Luna, s..n. E-50018 Zaragoza, Spain.
e-mail: `ecueto@unizar.es`

of generalized variables that describe adequately the large-scale structures dominating the solution of the problem [31]. Recently, a number of works have pursued this same idea in the field of solid dynamics, see, to name a few, [40] [21] [23] [27] [47]. In these last works it is shown that model order reduction allows not only for a significant reduction of the number of spatial degrees of freedom, but also for larger time steps verifying stability constraints. However, in these references, an implicit simulation of 1.5 seconds of real time consumes 8540 seconds of computing time. Even if savings on the order of ten times have been reported for a judicious choice of the number of degrees of freedom, this is still far from real-time constraints.

Some applications are specially demanding for feedback responses of structural dynamics simulations. For instance, it is well known that haptic (force) peripherals require feedback responses on the order of 500 Hz to 1 kHz. This is equivalent to solve non-linear solid dynamics each millisecond, or one thousand times per second. To the best of our knowledge, only linear elastic models undergoing large deformations (Saint Venant-Kirchhoff models) have been solved under such an astringent requirement by employing either GPUs [45] or explicit algorithms combined with model reduction [46] [9] [8]. The extension of these techniques to account for the stability provided by implicit time integrators and general non-linear models seems to be difficult, even for reduced order models.

In this work a completely different (yet still based on some form of model order reduction) philosophy is employed. A *computational vademecum* [13] is developed that acts as a time integrator. This sort of black box is fed with the results of the previous time step, t, (in the form of the vector of nodal displacements of the model) and provides, under the severe feedback restrictions commented before, the solution at time $t + \triangle t$, still providing the stability features of your preferred time integrator, implicit or not.

To achieve this kind of "universal" time integrator, the approach here presented first poses the problem in a parametric framework. This means that the vector of displacements at time t is considered as parameters of the problem (in addition to being the initial conditions of the subsequent time step), the displacements at time $t + \triangle t$ being the unknowns. Of course, the number of degrees of freedom in a nowadays problem of industrial interest is huge, so the number of parameters could be simply unaffordable. To alleviate this problem, a reduced order basis, obtained via Proper Orthogonal Decomposition (POD) [22] [29] [42] [44][43] is employed, thus guaranteeing the parametrization of the space of initial conditions with a minimum of degrees of freedom. Even with this efficient approach for the initial conditions, the number of parameters (initial conditions) usually reaches some tens.

The method is then structured in an off-line/on-line strategy. The parametric problem, depending on the initial conditions in the interval $(t, t + \triangle t)$ is then solved off-line, and once for life, for *any* value of the parameters within a prescribed interval. This parametric problem can be solved by using your favorite time integrator, implicit or explicit, and the time step for these integrators need not to be the same $\triangle t$ (can be smaller if, for some applications, the feedback response requirement is bigger than the stability constraints of the particular integrator used). Once this off-line phase of the method is computed (as already mentioned, only once for life), the on-line part can run under severe real-time constraints by simply feeding the "vademecum" with the results at the previous converged time step.

The only conceptual difficulty in this approach is to solve the parametric problem. Note that this problem considers every degree of freedom of the (reduced) model as a parameter. Thus, it is frequent to consider some tens of parameters for large (full) problems comprising thousands of finite element degrees of freedom. It is well-known that it is not possible to solve this kind of parametric problems by simply employing any mesh-based technique (finite elements, finite differences), since the number of degrees of freedom increases exponentially with the number of state space dimensions (equal to the number of spatial dimensions plus the number of parameters) [28].

To alleviate this *curse of dimensionality*, the authors, among others, have developed a technique coined as Proper Generalized Decomposition (PGD) [7] [6] [12] [14] [5] [17][18] that approximates the solution as a finite sum of separable functions. These functions are determined in the off-line part of the method by invoking a greedy algorithm (whose solution involves a non-linear problem, usually solved by simple fixed-point, alternate directions algorithm). The method, whose roots can be traced back to the space-time separation in the so-called *radial loading* technique within the LATIN method [24] [25] [26] [41][19], allows for an efficient solution of problems defined in high dimensional parameter spaces. In the PGD framework, the complexity of the problem scales linearly with the number of state space dimensions (parameters, actually), instead of exponentially.

With all these ingredients, that will be detailed in the sequel, the method here proposed provides with a sort of meta-model or surface response (computational vademecum, in words of [13]) of the system subjected to any type of force and initial conditions (taking values in a given interval).

The outline of the paper is as follows. In Section 2 we describe the basics of the formulation of the problem of solid dynamics here addressed. In Section 3 the parametric form of the problem is introduced as a pre-requisite for the establishment of the off-line phase of the method. Section 4 develops the PGD-based formulation of the resulting multi-dimensional problem while Section 5 explains how to obtain an efficient parametrization of the space of initial conditions. Finally, Section 6 includes some numerical examples showing the potentiality of the proposed method. The paper is completed with some conclusions and comments on future works.

2. Solid dynamics

The strong form of the problem of solid dynamics is well-known, but it is reviewed here for completeness. We consider first the linear elastic case. Non linear strain measures will be analyzed in Section 6.2. For the former case the problem can be stated as: given $\boldsymbol{f}$, $\boldsymbol{g}$, $\boldsymbol{h}$, $\boldsymbol{u}_0$ and $\dot{\boldsymbol{u}}_0$, find $\boldsymbol{u} : \bar{\Omega} \times]0, T] \to \mathbb{R}^3$ such that

$$\rho\ddot{\boldsymbol{u}} = \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \boldsymbol{f} \quad \text{in } \Omega \times]0, T] \tag{1a}$$

$$\boldsymbol{u} = \boldsymbol{g} \quad \text{on } \Gamma_u \times]0, T] \tag{1b}$$

$$\boldsymbol{\sigma}\boldsymbol{n} = \boldsymbol{h} \quad \text{on } \Gamma_t \times]0, T] \tag{1c}$$

$$\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{u}_0(\boldsymbol{x}) \quad \boldsymbol{x} \in \Omega \tag{1d}$$

$$\dot{\boldsymbol{u}}(\boldsymbol{x}, 0) = \dot{\boldsymbol{u}}_0(\boldsymbol{x}) \quad \boldsymbol{x} \in \Omega. \tag{1e}$$

The corresponding weak form is: given $\boldsymbol{f}$, $\boldsymbol{g}$, $\boldsymbol{h}$, $\boldsymbol{u}_0$ and $\dot{\boldsymbol{u}}_0$ find $\boldsymbol{u}(t) \in \mathcal{S}_t = \{\boldsymbol{u} | \boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{g}(\boldsymbol{x}, t), \ \boldsymbol{x} \in \Gamma_u, \ \boldsymbol{u} \in \mathcal{H}^1(\Omega)\}$, $t \in [0, T]$, such that for all $\boldsymbol{w} \in \mathcal{V}\{\boldsymbol{u} | \boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{0}, \ \boldsymbol{x} \in \Gamma_u, \ \boldsymbol{u} \in \mathcal{H}^1(\Omega)\}$,

$$(\boldsymbol{w}, \rho\ddot{\boldsymbol{u}}) + a(\boldsymbol{w}, \boldsymbol{u}) = (\boldsymbol{w}, \boldsymbol{f}) + (\boldsymbol{w}, \boldsymbol{h})_\Gamma \tag{2a}$$

$$(\boldsymbol{w}, \rho\boldsymbol{u}(0)) = (\boldsymbol{w}, \rho\boldsymbol{u}_0) \tag{2b}$$

$$(\boldsymbol{w}, \rho\dot{\boldsymbol{u}}(0)) = (\boldsymbol{w}, \rho\dot{\boldsymbol{u}}_0), \tag{2c}$$

Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2012; **00**:1–28

where we have used the so-called *direct notation*:

$$a(\boldsymbol{w}, \boldsymbol{u}) = \int_\Omega \boldsymbol{\nabla}^s \boldsymbol{w} : \mathsf{C} : \boldsymbol{\nabla}^s \boldsymbol{u} \, d\Omega,$$

$$(\boldsymbol{w}, \boldsymbol{f}) = \int_\Omega \boldsymbol{w} \boldsymbol{f} d\Omega,$$

$$(\boldsymbol{w}, \boldsymbol{h})_\Gamma = \int_{\Gamma_t} \boldsymbol{w} \boldsymbol{h} d\Gamma.$$

This constitutes the main ingredient for a finite element semi-discretization of the problem, leading to an ODE to be discretized in time by your favorite time integrator, implicit or explicit. The method here proposed proceeds by stating the problem (2) in a parametric form. In this case, the parameters will be the initial conditions.

3. Parametric, multidimensional, form of the problem

The main modification in the standard approach to the problem given by Eqs. (1)-(2) is to reinterpret the parametric dependence of the problem on its initial conditions $\boldsymbol{u}_0$ and $\dot{\boldsymbol{u}}_0$ (assumed for the time being constant; the general case will be addressed later on) as being actually multi-dimensional:

$$\boldsymbol{u} : \bar{\Omega} \times]0, T] \times \mathcal{I} \times \mathcal{J} \to \mathbb{R}^3,$$

where $\mathcal{I} = [\boldsymbol{u}_0^-, \boldsymbol{u}_0^+]$ and $\mathcal{J} = [\dot{\boldsymbol{u}}_0^-, \dot{\boldsymbol{u}}_0^+]$ represent the considered intervals of variation of initial boundary conditions, $\boldsymbol{u}_0$ and $\dot{\boldsymbol{u}}_0$. This makes it necessary to define a new (triply-) weak form where:

$$a(\boldsymbol{w}, \boldsymbol{u}) = \int_\mathcal{I} \int_\mathcal{J} \int_\Omega \boldsymbol{\nabla}^s \boldsymbol{w} : \mathsf{C} : \boldsymbol{\nabla}^s \boldsymbol{u} \, d\Omega d\dot{\boldsymbol{u}}_0 d\boldsymbol{u}_0,$$

$$(\boldsymbol{w}, \boldsymbol{f}) = \int_\mathcal{I} \int_\mathcal{J} \int_\Omega \boldsymbol{w} \boldsymbol{f} \, d\Omega d\dot{\boldsymbol{u}}_0 d\boldsymbol{u}_0,$$

$$(\boldsymbol{w}, \boldsymbol{h})_\Gamma = \int_\mathcal{I} \int_\mathcal{J} \int_{\Gamma_t} \boldsymbol{w} \boldsymbol{h} \, d\Gamma d\dot{\boldsymbol{u}}_0 d\boldsymbol{u}_0.$$

With this modification in mind, one can proceed as in standard textbooks of finite elements and semi-discretize, with the help of appropriate finite-dimensional approximations to $\mathcal{S}_t$ and $\mathcal{V}$, $\mathcal{S}_t^h$ and $\mathcal{V}^h$, respectively, the weak form so as to obtain the following problem: given $\boldsymbol{f}$, $\boldsymbol{g}$, $\boldsymbol{h}$, $\boldsymbol{u}_0$ and $\dot{\boldsymbol{u}}_0$ find $\boldsymbol{u}^h(t) = \boldsymbol{v}^h + \boldsymbol{g}^h \in \mathcal{S}_t^h$ (note that $\boldsymbol{g}(\boldsymbol{x}, t) = \boldsymbol{u}(\boldsymbol{x}, t)$ on Γ_u) such that for every $\boldsymbol{w}^h \in \mathcal{V}^h$,

$$(\boldsymbol{w}^h, \rho \ddot{\boldsymbol{v}}^h) + a(\boldsymbol{w}^h, \boldsymbol{v}) = (\boldsymbol{w}^h, \boldsymbol{f}) + (\boldsymbol{w}^h, \boldsymbol{h})_\Gamma - (\boldsymbol{w}^h, \rho \ddot{\boldsymbol{g}}^h) - a(\boldsymbol{w}^h, \boldsymbol{g}^h), \tag{3a}$$

$$(\boldsymbol{w}^h, \rho \boldsymbol{v}^h(0)) = (\boldsymbol{w}^h, \rho \boldsymbol{u}_0) - (\boldsymbol{w}^h, \rho \boldsymbol{g}^h(0)), \tag{3b}$$

$$(\boldsymbol{w}^h, \rho \dot{\boldsymbol{v}}^h(0)) = (\boldsymbol{w}^h, \rho \dot{\boldsymbol{u}}_0) - (\boldsymbol{w}^h, \rho \dot{\boldsymbol{g}}^h(0)). \tag{3c}$$

This results in an absolutely general solution (thus the name *computational vademecum* coined in [13]) to the problem (2) for *any* initial conditions. A similar philosophy, but for simpler problems was implemented in [18]. It can also be viewed as a sort of response surface or meta-model obtained in an entirely computational way and in one execution of the algorithm. The method here developed computes off-line and once for life this general solution so as to be able to simply post-process during the on-line phase of the algorithm.

However, there remain some difficulties to be addressed, that will soon become apparent. The most important one is the high-dimensional nature of the problem (3). Once semi-discretized, Eqs. (3) involve $\boldsymbol{u}_0$ and $\dot{\boldsymbol{u}}_0$ as parameters. When these initial conditions evolve in the domain they could be expressed by means of their respective nodal values at the finite element mesh. This means that the number of actual parameters is six times the number of nodes in the mesh (three for $\boldsymbol{u}_0$ and three more for $\dot{\boldsymbol{u}}_0$ at each nodal location). This is simply unaffordable for any known method to date (several tens or hundreds of thousands parameters for industrial applications).

Remark 1. *In previous works, see [38] [37], a PGD formulation was presented that considered the position of the load as a parameter. This would render the formulation in Eq. (3) completely general and apt for a true interactive simulation at real-time feedback rates. In order to avoid to further obscure the developments made in this work, the authors have chosen not to include here the position of the load as a parameter of the formulation. Although it is a common belief that the load position would tremendously increase the number of necessary modes, it can be demonstrated that this is not so dramatic. We refer the reader to [2] in the case of POD reduction or the previous references for a PGD reduction.*

In order to divide and conquer, we address first discrete problems in Section 4 and then move to continuous problems discretized with the help of finite elements in Section 6.

4. PGD approximation to parametric solid dynamics

4.1. High-dimensional approximation

To efficiently solve solid dynamics Eqs. (2), one possibility lies in considering space-time approximations to the problem, as in [24] [11]. Here, we follow a more standard, although multidimensional, PGD approximation [12] [14] to the problem given by Eqs. (3), by considering direct time integration[†], as:

$$\boldsymbol{v}^h(\boldsymbol{x}, t, \boldsymbol{u}_0, \dot{\boldsymbol{u}}_0) = \left[\sum_{i=1}^{N} \boldsymbol{F}_i(\boldsymbol{x}) \circ \boldsymbol{G}_i(\boldsymbol{u}_0) \circ \boldsymbol{H}_i(\dot{\boldsymbol{u}}_0) \right] \circ \boldsymbol{d}(t), \qquad (4)$$

where the nodal coefficients $\boldsymbol{d}(t)$ carry out all the time-dependency of the solution and the symbol "∘" stands for the entry-wise Hadamard or Schur multiplication of vectors.

Remark 2. *It is important to highlight that, in Eq. (4), $\boldsymbol{d}(t)$ does not represent a time function, but the usual time coefficients of finite elements. That is why it does not appear within the brackets. If a space-initial conditions-time decomposition would have been preferred, $\boldsymbol{d}(t)$ would have appeared within the brackets and therefore it would represent a set of time functions. This is the approach followed in [11]. But since it would provide the complete solution for an a priori fixed time interval, it could not be used in interactive simulations, which is the ultimate goal of this work.*

Functions $\boldsymbol{F}$, $\boldsymbol{G}$ and $\boldsymbol{H}$ will be expressed in terms of low (here, three-) dimensional finite element basis functions. The computation of these basis is one of the more salient features of PGD, since they

---

[†]The use of direct time integration is mandatory if the objective of real-time simulation is to provide the analyst with an interactive tool.

Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2012; **00**:1–28

are obtained on the fly (although in the off-line phase of the method), a priori, without the need to prior computer experiments, typical of POD-based techniques. They are computed by means of a greedy algorithm in which one sum is computed at a time, while one product is computed in a fixed point, alternated directions algorithm (here the most classic PGD algorithm is employed very satisfactorily, see [41] for alternative formulations). Thus, having an approximation to $\boldsymbol{v}^h$ converged at iteration n, the $(n + 1)$-th term is obtained as

$$\boldsymbol{v}^{n+1}(\boldsymbol{x}, t, \boldsymbol{u}_0, \dot{\boldsymbol{u}}_0) = \left[\sum_{i=1}^{n} \boldsymbol{F}_i(\boldsymbol{x}) \circ \boldsymbol{G}_i(\boldsymbol{u}_0) \circ \boldsymbol{H}_i(\dot{\boldsymbol{u}}_0) + \boldsymbol{R}(\boldsymbol{x}) \circ \boldsymbol{S}(\boldsymbol{u}_0) \circ \boldsymbol{T}(\dot{\boldsymbol{u}}_0) \right] \circ \boldsymbol{d}(t).$$

Remark 3. *Note that here n and $n + 1$ refer to the number of terms in the PGD separated representation, not to the time step.*

In this framework, an admissible variation of $\boldsymbol{v}^h$ is computed as

$$\boldsymbol{w}^h = \boldsymbol{R}^* \circ \boldsymbol{S} \circ \boldsymbol{T} + \boldsymbol{R} \circ \boldsymbol{S}^* \circ \boldsymbol{T} + \boldsymbol{R} \circ \boldsymbol{S} \circ \boldsymbol{T}^*,$$

where functional dependencies have been omitted for clarity.

If we inject the approximations to $\boldsymbol{v}^h$ and $\boldsymbol{w}^h$ into the weak form of the problem, Eqs. (3), we arrive at a semi-discrete problem that can be approximated in time using your favourite time integrator.

Still a more general expression than that of Eq. (4) can be envisaged so as to include the dependence of the response of the system with respect to any load history. Thus, we consider an approximation of the type:

$$\boldsymbol{v}^h(\boldsymbol{x}, t, \boldsymbol{u}_0, \dot{\boldsymbol{u}}_0, \boldsymbol{h}_0) = \left[\sum_{i=1}^{N} \boldsymbol{F}_i(\boldsymbol{x}) \circ \boldsymbol{G}_i(\boldsymbol{u}_0) \circ \boldsymbol{H}_i(\dot{\boldsymbol{u}}_0) \circ \boldsymbol{J}_i(\boldsymbol{h}_0) \right] \circ \boldsymbol{d}(t),$$

where at each time instant t it is assumed that the modulus of the load $\boldsymbol{h}(t)$ (assumed constant through that time step and equal to its value at the beginning, $\boldsymbol{h}(0)$), can take any value within a prescribed interval $\boldsymbol{h}_0 \in [h^-, h^+]$. This renders a completely general, multidimensional, expression for $\boldsymbol{v}^h$ that includes its dependence on initial conditions as well as on the value of the applied load at each time interval. Knowing in advance the analytical expression of the load and particularizing it at each time instant allows to obtain the response of the structure for any load taking values within $[h^-, h^+]$.

The very last detail in the implementation is to search for an approximation not for the whole time interval of the problem, $]0, T]$, but for $]0, \triangle t]$:

$$\boldsymbol{v} : \bar{\Omega} \times]0, \triangle t] \times \mathcal{I} \times \mathcal{J} \times [h^-, h^+] \to \mathbb{R}^3,$$

where $\triangle t$ represents the necessary time to response prescribed by the particular envisaged application. For instance, for haptic feedback it has been already mentioned that a physical sensation of touch needs for some 500 Hz to 1 kHz feedback rate. This means that $\triangle t = 0.001$ seconds. This value $\triangle t$ is not the necessary time step to achieve stability in the time integration chosen (that can be smaller if needed), although it can be coincident (and will be for all the examples shown hereafter).

In this way we obtain a completely general expression of $\boldsymbol{v}$ that can be particularized for any initial displacement and velocity fields. So to speak, a sort of black-box that, conveniently fed with the last time step results, provides with the displacement field at time $t + \triangle t$.

In what follows we show how the proposed technique behaves when applied to discrete problems. These problems posses analytical solutions that are employed to measure the influence of model reduction in the accuracy of the method.

4.2. Example 1: Frame structure

We consider here a simple example of validation with two degrees of freedom. It represents the frame structure depicted in Fig. 1. Slabs are assumed perfectly rigid, of masses $m_2 = 2m_1 = 4000$ kg, while the stiffness of the columns is such that $\frac{12EI}{L^3} = 10^5$ N/m.
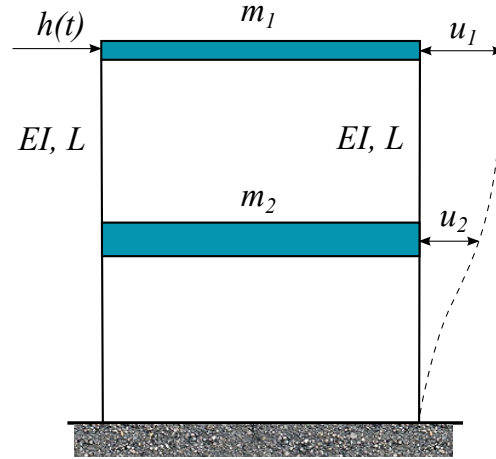

Figure 1. Frame structure considered in Section 4.2.

Consider now that the structure is subjected to a step load $h(t) = 10000$ N, applied for $t > 0$. Note that the frame has been modeled with the aid of one single beam representing both columns, discretized into two finite elements. Only the degrees of freedom relative to the displacements u_1 and u_2 are free. In Fig. 2 the response of the structure is shown by comparing the PGD results with those of the analytical solution. A forward-Euler explicit time integration algorithm has been employed. As can be noticed, results are almost indistinguishable. Eight modes (i.e., $n = 8$ in Eq. (4.1)) have been considered in this case for almost a perfect agreement with the analytical solution, as noticed in Fig. 2. In general, some error estimation must be available to determine the number of modes necessary for a prescribed error tolerance. In the framework of PGD some previous works have addressed this issue [5] [26] [33].

Functions $\boldsymbol{F}_i$, $\boldsymbol{G}_i$, $\boldsymbol{H}_i$ and $\boldsymbol{J}_i$, $i = 1, \ldots, 8$ for this problem are depicted in Fig. 3. The choice of the assumed limits of variation for each magnitude (intervals $\mathcal{I}$, $\mathcal{J}$) is heuristic. If, during the simulation, any parameter escapes from its assumed interval of validity, a new simulation is started with bigger intervals. It has been noticed during this work, however, that uniformly-meshed intervals should be preferred in favor of a better convergence of the PGD algorithm.

4.3. Example 2: Vibration of a clamped beam

We consider now the problem of a doubly-clamped beam subjected to a force in its middle point as sketched in Fig. 4. In IS units, the beam has an area of 2850×10^{-6}, inertia moment 1.94×10^{-5}, Young's modulus of 2.2×10^7, length $L = 1$, and mass density $\rho = 325$.

As mentioned before, one only execution of the program provides with the response of the system

Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

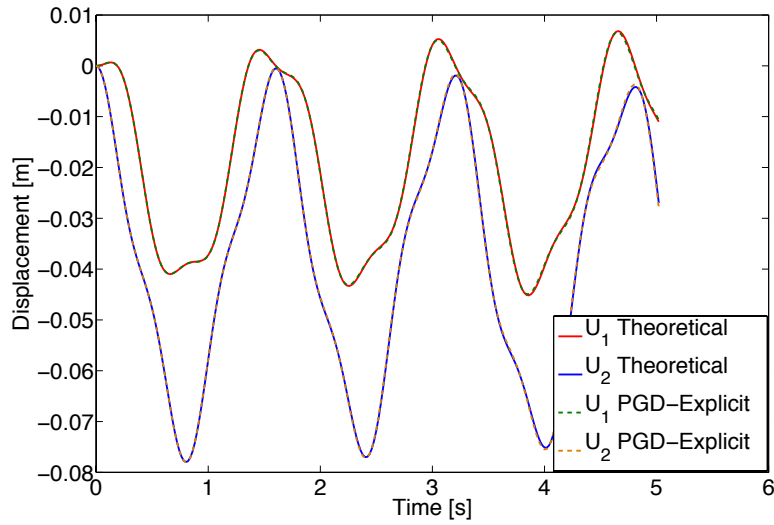*Int. J. Numer. Meth. Engng* 2012; **00**:1–28

Figure 2. Response of the frame structure. PGD results vs. analytical solution.

under virtually any load (at least those taking values within a prescribed interval). Thus, it could be evaluated in real-time for a load $h(t) = 20\sin(2\pi t)$, rendering the response in Fig. 5. In it, the sensibility of the explicit (forward-Euler) algorithm to different time steps, from 0.05 to 0.005 seconds, is shown. It must be emphasized that in this approach load is considered constant within every time increment, but variable in the whole simulation.

The proposed algorithm is by no means restricted to explicit time integration. Any time integrator can be used. For instance, we compare in Fig. 6 the response of the beam computed by an explicit, forward-Euler algorithm and an Energy-Momentum conserving integrator proposed in [10]. The increase in accuracy can be noticed. Thus, energy conservation shows to be a question not only of the number of modes retained in the model (for which an appropriate error estimation should be available) but, notably, the time integrator used. At the limit, the just developed PGD approach converges to the full finite element solution is sufficient number of modes are considered, as proved in many previous references [14] [12].

5. Parametrization of the space of initial conditions

So far discrete problems have been considered (they can be modeled with spring-mass systems of two and one degrees of freedom, respectively). Thus, initial conditions were composed by a vector of four components (two displacements plus two velocities) at most. When moving to large-scale continuous problems, initial conditions constitute a vector field in the domain Ω. After finite element discretization in space, both the unknown field $\boldsymbol{v}$ and initial conditions, $\boldsymbol{u}_0$ and $\dot{\boldsymbol{u}}_0$, will be represented by the usual vector of nodal values for these variables. Thus, complex meshes could imply maybe several hundreds of thousands of degrees of freedom (if not millions). The number of parameters in the formulation presented before for discrete problems becomes thus too high. PGD has shown success for problems
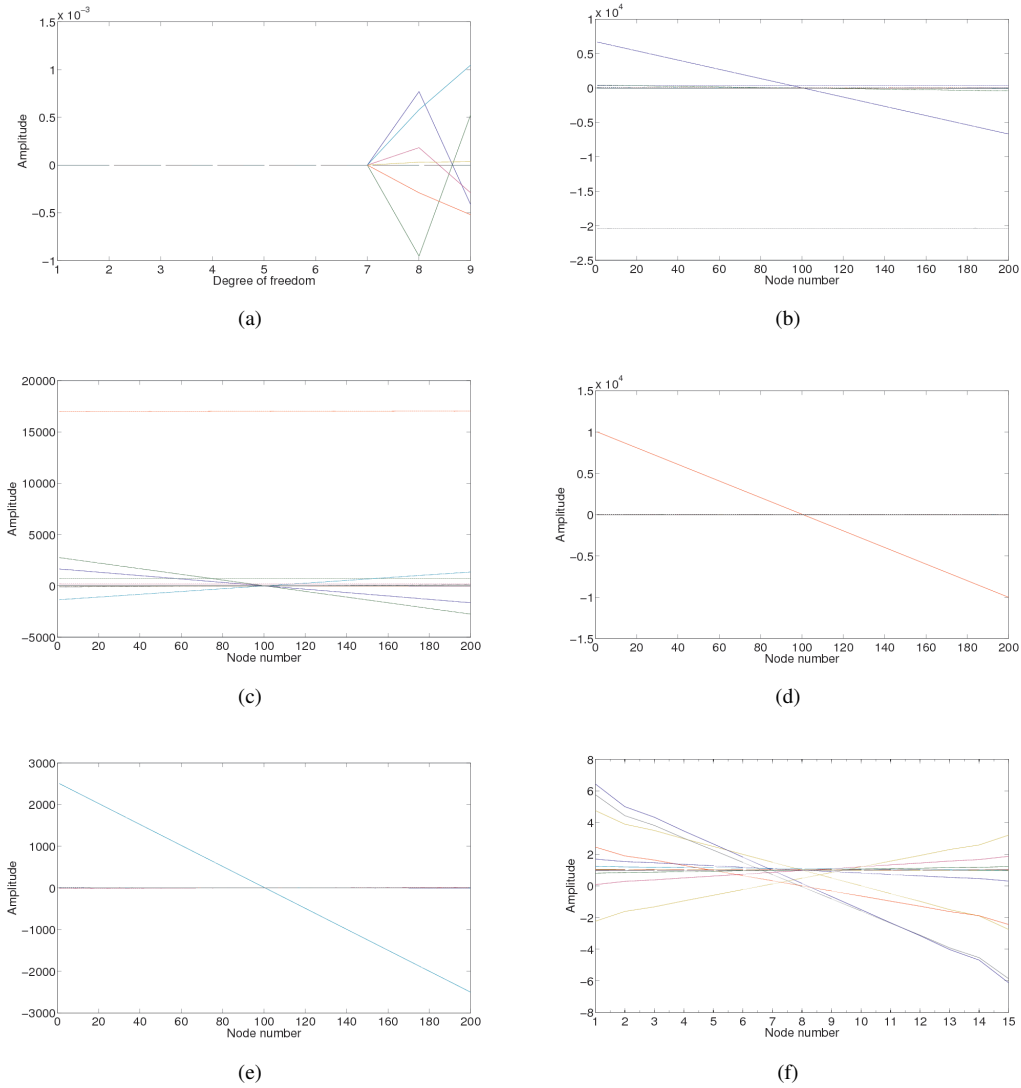
Figure 3. Monodimensional functions used to approximate the response of the frame in Fig. 1 subjected to a step load. (a) $\boldsymbol{F}_i(\boldsymbol{x})$. Note that the frame was modeled with the help of one single bar, discretized into two finite elements, and hence the nine degrees of freedom in the abscisa. Only the two non-clamped horizontal displacements are free (degrees of freedom 8 and 9 in the figure). (b) and (c) represent, respectively, the dependence of the solution on u_1^n and u_1^{n-1}, where they have been assumed to lie in the interval $[-3,3]m/s$. The same applies, but for degree of freedom u_2, with (d) and (e). Finally, (f) represents functions $\boldsymbol{J}_i(\boldsymbol{h}_0)$. Even if the load is constant, it has been assumed to vary between 1000 and $15000N$. All amplitudes are expressed in IS units.

up to dimension 100 [14] and maybe more, but it is not realistic to dream with efficient solutions for any number of dimensions.

Therefore it seems reasonable to somehow optimize the parametrization of the space of initial
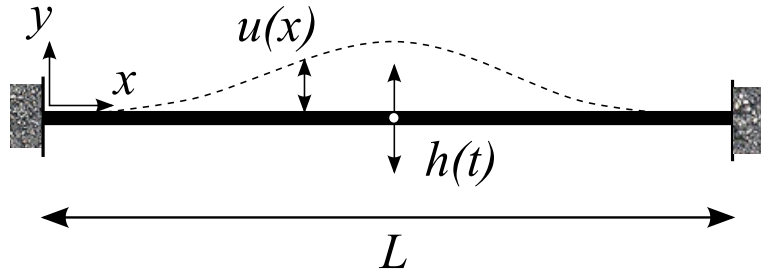
Figure 4. Geometry of the doubly-clamped beam problem.
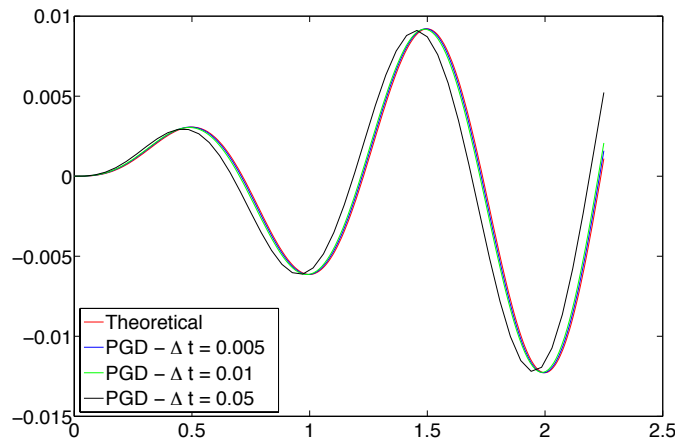
Figure 5. Analysis of the influence of the time step in the explicit integration of the response of the beam in Section. 4.3.

conditions. To this end an optimal (in the sense of POD) basis is here constructed, based upon the results for complete problems similar to the one at hand.

5.1. A POD-PGD approach

One key aspect in the method developed so far arises when, as mentioned before, we move to continuous rather than discrete problems. Finite element discretizations of these problems lead to an impressive number of degrees of freedom if we employ a finite element parametrization of the space of initial conditions, which no more than a vectorial field defined in Ω. Consider, for instance, a mesh composed of, say, one thousand nodes in 3D. It will have 3000 degrees of freedom to parametrize the initial displacement condition and other 3000 for the initial velocity one. Each degree of freedom will be considered in the method here developed as a new parameter —or state-space dimension— of the problem. But a problem defined in 6000 dimensions is not feasible, even for PGD methods, due to its computational complexity. Therefore it is mandatory to develop suitable parametrizations of these spaces that eventually lead to fewer number of degrees of freedom. This is accomplished in this section.

It is well-known that assuming the displacement or velocity fields to be interpolated by piece-wise

Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls
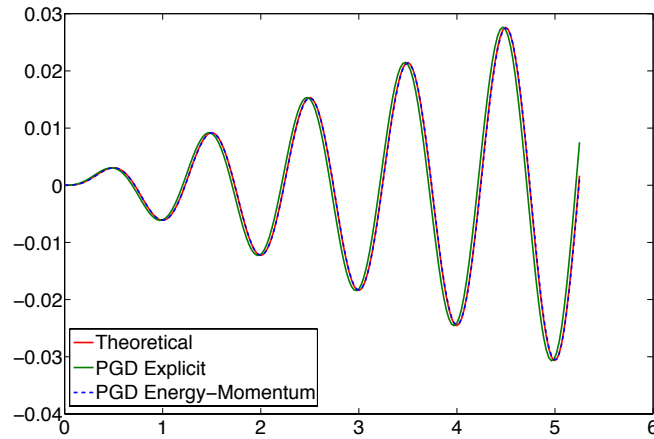
Int. J. Numer. Meth. Engng 2012; **00**:1–28

Figure 6. Comparison of the results of the beam in Section. 4.3. Explicit vs. Energy-Momentum conserving integration proposed in [10].

polynomials constitutes a reasonable assumption, that is on the basis of the success of finite element methods. However, it is not the best choice if we known in advance some information about the behavior of the response of the system. This is on the root of the development of model order reduction techniques [22] [29] [30]. Instead, it is preferred to run some tests on similar geometries and/or loading conditions and to obtain the best (in some norm) global, Ritz-like shape functions that best approximate the solution for these particular problems, in the hope to be also a good choice for the original problem. This rationale is on the basis of Proper Orthogonal Decomposition (POD) methods.

In this work a mixed POD-PGD approach is envisaged: based upon the previous solution of similar problems and the use of POD decompositions, a suitable parametrization of the space of initial conditions is obtained by employing the lowest possible number of generalized degrees of freedom. Results for a particular problem will be then projected onto this reduced-order basis and re-injected into the (off-line, multi-dimensional) obtained PGD solution to advance in time. We thus obtain a sort of response surface, meta-model or computational vademecum [13], that takes as input parameters the projection of the previous time step displacement and velocity fields onto the reduced basis parametrization of initial conditions. A sketch of the proposed method is shown in Fig. 7.

Therefore, the proposed method needs for the solution of some previous problems, similar to the one at hand, to obtain a set of reduced-order basis. These basis functions will be employed to approximate any subsequent initial condition to on-line executions of the PGD integrator. As a result, the integrator will provide with the displacement at every node of the mesh. This (complete) vector of nodal displacements must be then projected into the space spanned by the reduced basis set so as to be injected again into the PGD integrator.
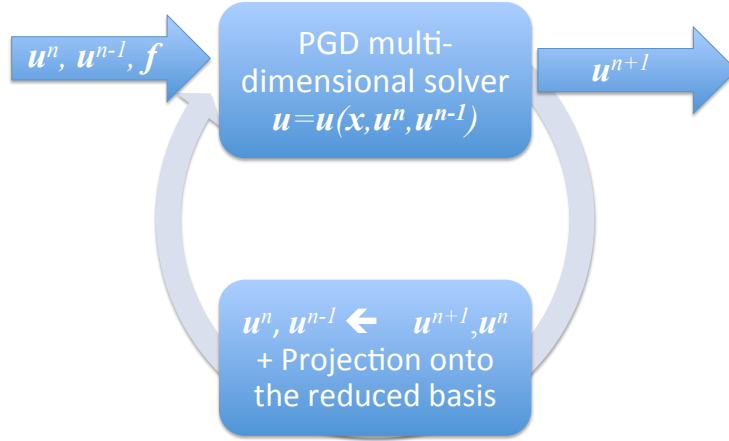
Figure 7. Sketch of the proposed algorithm. A POD basis is employed for the parametrization of the space of initial conditions of the problem (for forward Euler schemes, for instance, this means in practice that the solution deepens on the displacement at time steps n and $n-1$, respectively).

6. Numerical Examples

To show how the proposed technique behaves for some engineering problems we consider two problems. The first represents the propagation of a wave on a solid block, while the second represents the palpation of a human liver during an endoscopic surgery procedure [35] [39] [36].

6.1. Wave propagation on a block

To show the performance of the proposed technique, we have analyzed an example of the propagation of a wave within a solid block of a continuous elastic material. The block is $5 \times 5 \times 10$ m, with Young's modulus 2.66×10^5 kN/m^2, Poisson's ratio 0.33, density $\rho = 2.0$ t/m^3 (P-wave velocity $c_1 = 443.9$ m/s, data is obtained from [48]). The block is discretized with a very poor tetrahedral mesh involving 45 nodes, as shown in Fig. 8. In this example we do not care about the accuracy of the model itself, but rather to compare with state of the art numerical techniques.

The block is subjected to an impulsive loading $h(t) = 4.0 \cdot H(t)$ kN/m^2 on the top face, where $H(t)$ stands for the Heaviside function. At all other surfaces, the normal displacements are restraint to zero.

The problem is compared to a FE solution obtained by employing HHT integration [20] with the

following parameters: $\alpha = -0.05$, $\beta = \frac{1}{4}(1-\alpha)^2$ and $\gamma = \frac{1}{2} - \alpha$. This same solution is employed to extract the POD modes that will parameterize the space of initial conditions, see Fig. 9. The simulation employed a fixed time step of $\triangle t = 0.00125$ s, which corresponds more or less to the desired feedback rates for haptic devices and also provides good results for this particular problem. The displacement field at a typical time increment (in this case, the 201-th one) is shown in Fig. 10.
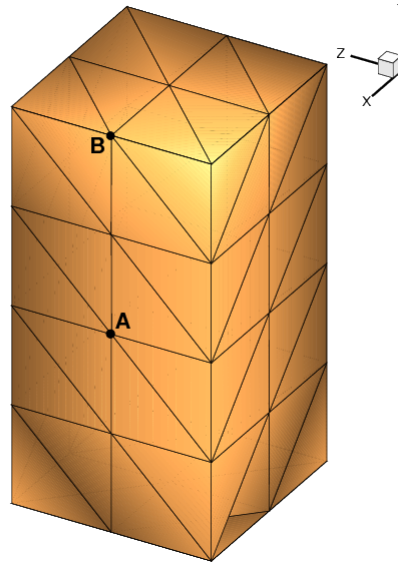
Figure 8. FE model for the problem of D-wave propagation. Points A and B are used to represent the solution in Figs. 15 and 16.
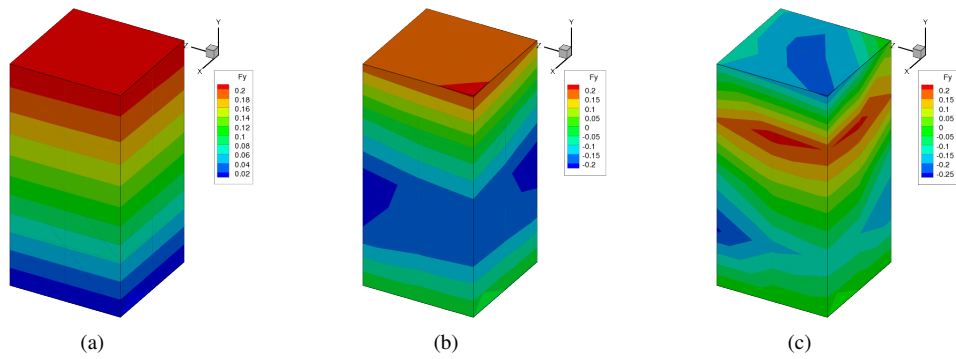
Figure 9. Three most important POD modes employed to parameterize the space of initial conditions for the wave propagation problem.

The eight first spatial modes of this PGD solution are represented in Fig. 11. Modes function of initial conditions (in fact, modes $\boldsymbol{G}_i(\boldsymbol{u}^n)$ and $\boldsymbol{H}_i(\boldsymbol{u}^{n-1})$, since we are employing forward-Euler integration

Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

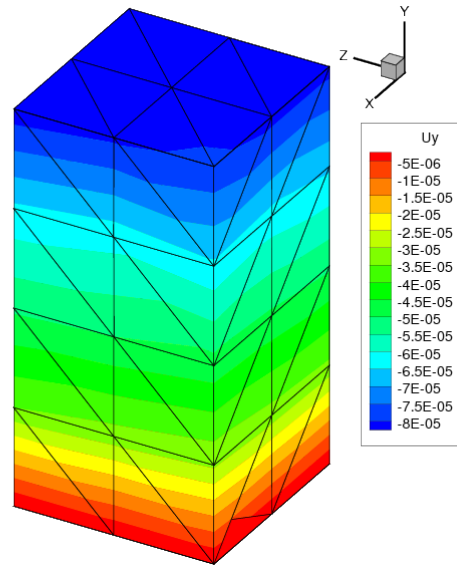*Int. J. Numer. Meth. Engng* 2012; **00**:1–28

Figure 10. Displacement field for the 201-th time step in the wave problem.

in the PGD loop) are represented in Figs. 12 and 13 respectively. Modes depending on the value of the external load are depicted in Fig. 14.

Remark 4. *Note that the discretization of the space of initial conditions must be done with an appropriate mesh. Too-coarse discretizations have led in our experience to meaningless results due to interpolation errors. For instance, in the wave propagation problem, results showing important, non-physical tractions where only compressive forces were applied have been found. It is important to estimate well the maximum amplitude that initial conditions could reach during the simulation (so as to avoid to perform extrapolations), but also a judicious choice of the meshes. Here, 100 nodes have been considered for the mesh (see Figs. 12 and 13).*

In Figs. 15 and 16 a comparison is made between the reference solution and the solution obtained with a 3-degree-of-freedom solution and one employing 8 degrees of freedom, respectively, for the parametrization of the space of initial conditions. Both employed an explicit, forward Euler, time integration.

It is noteworthy to mention that these results, for a chosen time step of $\triangle t = 0.00125$ s. are obtained between 0.000992 and 0.001325 s. This enables the use of this technique for real-time haptic response purposes, where feedback rates between $0.5 - 1.0$ kHz are needed for a realistic touch sensation. These results are obtained by using Matlab running on a Windows-operated HP laptop. While the complete simulation took approximately 8 seconds, the same simulation in the Abaqus software takes 16 minutes and 35 seconds (thus making it 125 times faster than the mentioned commercial software).

If, on the contrary, we employ an energy-momentum conserving scheme such as that by Bathe [10] results improve very much, and a great stability is conserved even for large integration times (10 seconds in the example depicted in Fig. 17).
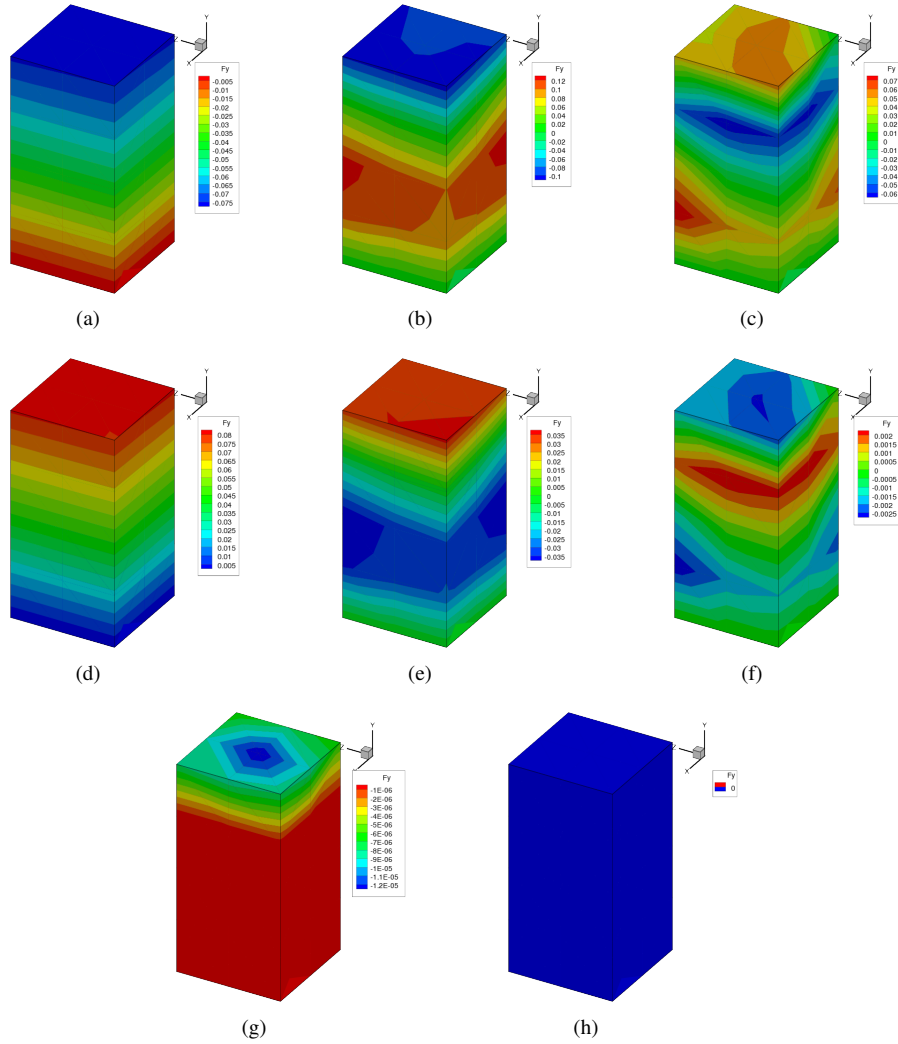
Figure 11. Eight most important spatial modes $\boldsymbol{F}_i(\boldsymbol{x})$ for the PGD solution of the wave propagation problem. Contour levels represent the value of the y component of the modes.

6.2. Including non-linear strain measures

Extending the before presented technique to non-linear (hyper-)elastodynamics does not incorporate any conceptual difficulty, albeit some details deserve a comment. In [37] and [38] the authors have presented and analyzed two different approaches for the efficient linealisation of hyperelastic constitutive models in the context of quasi-static PGD models. Notably, these works present an explicit and an implicit (respectively) approach to this problem that do not imply the need to re-construct the full tangent stiffness matrix of the complete model. This constitutes an active field of research in the framework of model order reduction methods, since non-linearities usually destroy all the gains
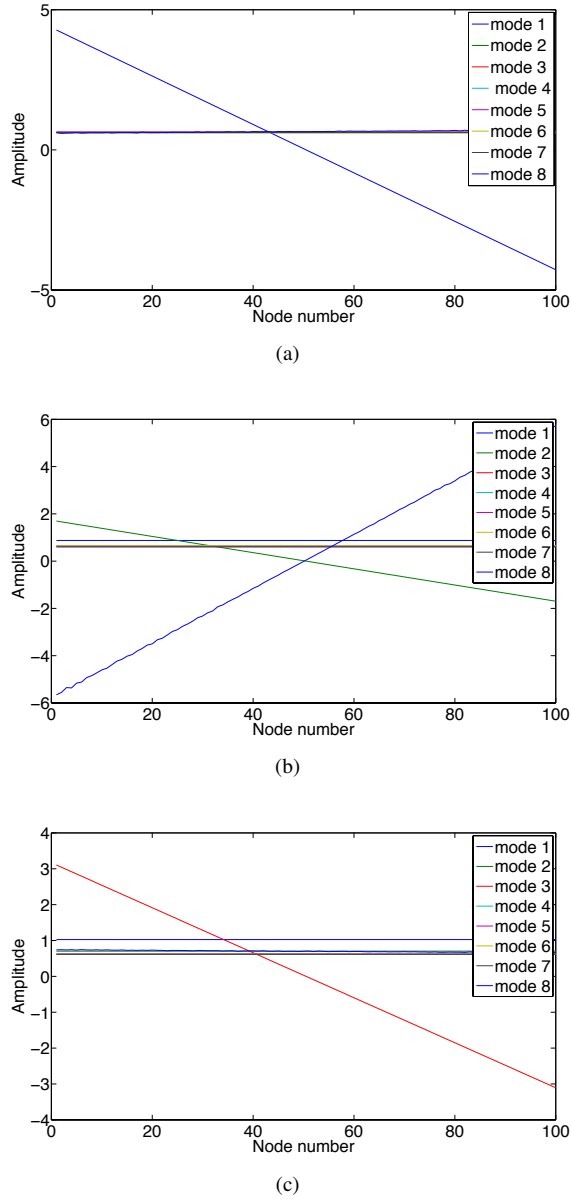
Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2012; **00**:1–28

(a)

(b)

(c)

Figure 12. Eight most important PGD modes $G_i(u^n)$ for the wave propagation problem. Note that the function is three-dimensional, so that its components are depicted in (a), (b) and (c), respectively.

obtained in reducing the model, if tangent matrices need to be assembled.

Here, an explicit approach, similar to that in [37] has been implemented. We consider again the problem of the wave propagation in Section 6.1 and, without modifying the POD basis employed to
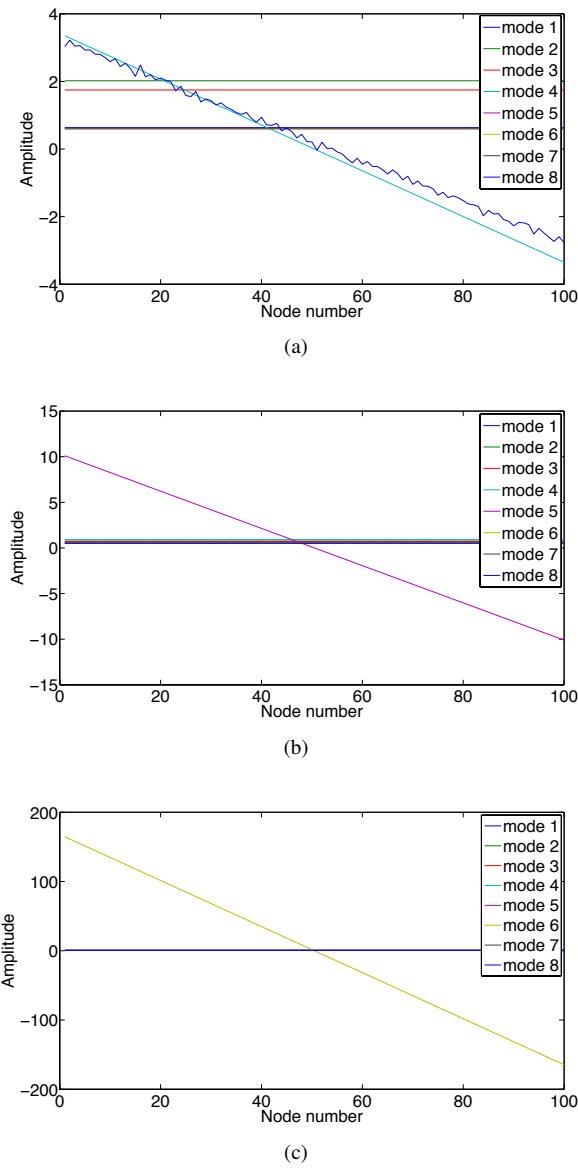
(a)

(b)

(c)

Figure 13. Eight most important PGD modes $\boldsymbol{H}_i(\boldsymbol{u}^{n-1})$ for the wave propagation problem. Note that the function is three-dimensional, so that its components are depicted in (a), (b) and (c), respectively.

parameterize the space of initial conditions, nor any of the data of the problem, consider a Kirchhoff-Saint Venant constitutive equation for the block. We are thus considering a linear and a non-linear term
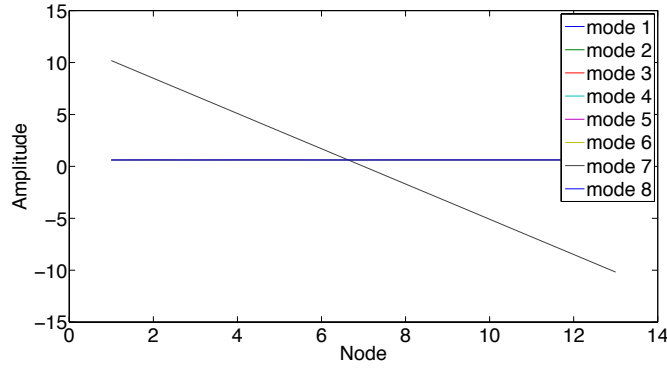
Figure 14. Eight most important PGD modes $\boldsymbol{J}_i(\boldsymbol{h}_0)$ for the wave propagation problem.
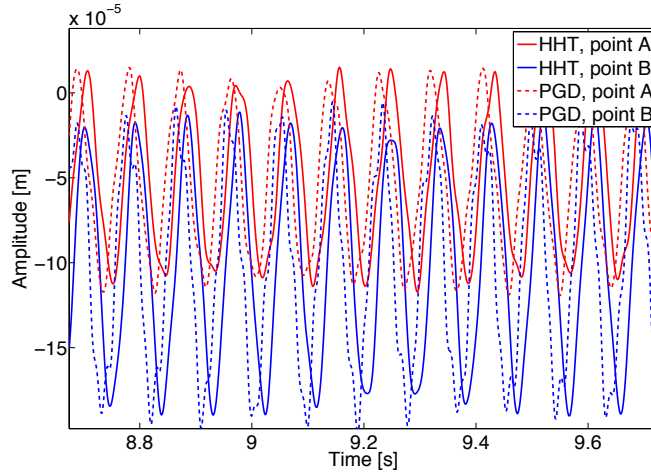

Figure 15. Wave propagation problem. Solution obtained with a three-degree-of-freedom parametrization of the space of initial conditions.

for the Green-Lagrange strain tensor, $\boldsymbol{E}$, in the form

$$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^T \boldsymbol{F} - 1) = \gamma_l(\boldsymbol{u}) + \gamma_{nl}(\boldsymbol{u}, \boldsymbol{u}),$$

where $\boldsymbol{F} = \boldsymbol{\nabla}\boldsymbol{u} + \boldsymbol{I}$ is the gradient of deformation tensor and

$$\gamma_l(\boldsymbol{u}) = \frac{1}{2}(\boldsymbol{\nabla}(\boldsymbol{u}^T) + \boldsymbol{\nabla}(\boldsymbol{u})), \tag{5a}$$

$$\gamma_{nl}(\boldsymbol{u}, \boldsymbol{u}) = \frac{1}{2}\boldsymbol{\nabla}(\boldsymbol{u}^T)\boldsymbol{\nabla}(\boldsymbol{u}). \tag{5b}$$

Hyperelastic materials are base on the assumption of a particular strain-energy function, Ψ. Then
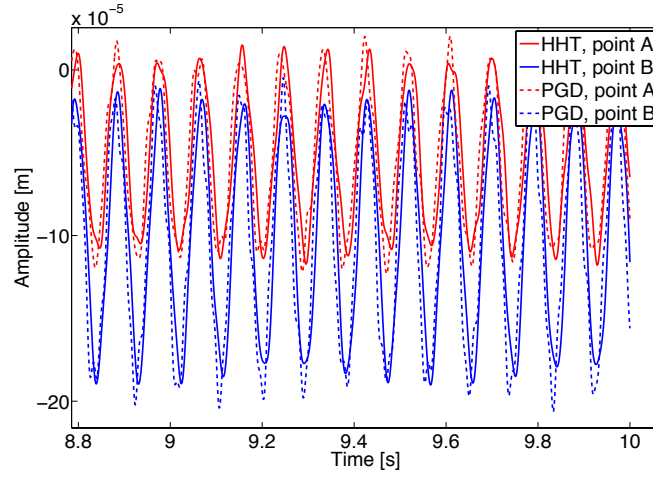
Figure 16. Wave propagation problem. Solution obtained with an eight-degree-of-freedom parametrization of the space of initial conditions and an explicit integration in time.
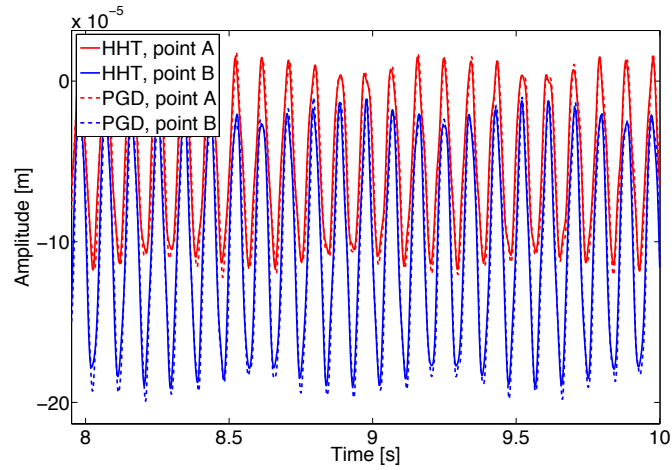

Figure 17. Wave propagation problem. Solution obtained with an eight-degree-of-freedom parametrization of the space of initial conditions and employing the energy momentum conserving scheme by Bathe [10].

the second Piola-Kirchhoff stress tensor $\boldsymbol{S}$ can thus be obtained by

$$\boldsymbol{S} = \frac{\partial \Psi}{\partial \boldsymbol{E}},$$

that is a symmetric tensor and is related to the first Piola-Kirchhoff stress tensor, $\boldsymbol{P}$, by $\boldsymbol{P} = \boldsymbol{F}\boldsymbol{S}$. The Kirchhoff-Saint Venant model is characterized by the energy function given by

$$\Psi = \frac{\lambda}{2}(\text{tr}(\boldsymbol{E}))^2 + \mu \boldsymbol{E} : \boldsymbol{E},$$

where λ and μ are Lame's constants. The second Piola-Kirchhoff stress tensor can be obtained by

$$S = \frac{\partial \Psi(E)}{\partial E} = \mathbf{C} : E,$$

in which $\mathbf{C}$ is the fourth-order constitutive (elastic) tensor.

Although it is very well known that Kirchhoff-Saint Venant model shows instabilities in compression, it provides with a very neat formulation to compare with the linear, small strains case. In Fig. 18 a comparison is made between the already obtained (via HHT time integration scheme) reference, linear solution and the one considering non-linear strain measures (the so-called Kirchhoff-Saint Venant already described). As expected, non-linear strains, when considering all the remaining parameters of the problem unchanged, only provides with some bigger displacements in the solution, but an overall similar behavior of the block. Considering hyperelastic models does not imply any further delay in the response feedback given by the method here presented, since the number of on-line operations is exactly the same as for the case of linear elastic models. Of course, the off-line part of the method is somewhat more involved, but it is solved only once and stored, as repeatedly mentioned.

In the on-line part of the algorithm results compatible with haptic feedback rates (i.e., more than 1 KHz) have been obtained without any special difficulty on a MacBook Pro laptop. Essays with a Geomagic Touch Haptic device (Geomagic, Morrisville, NC 27560, USA) did not provide any spurious jump in the perceived feedback force.
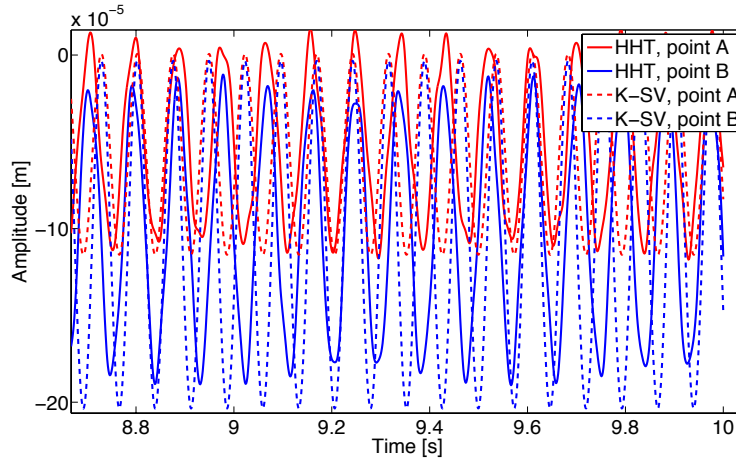

Figure 18. Wave propagation problem. Comparison of the (linear elastic) HHT reference solution with the Kirchhoff-Saint Venant one. Note that the inclusion of non-linear strain measures induces higher displacements, as expected, but an overall similar response in the problem. Solution obtained with one only mode in the POD basis of the space of initial conditions.

6.3. Palpation of a liver

As mentioned before, computational surgery is one of the fields where real-time requirements are more astringent [1] [3]. Surgery planning provides results corresponding to large periods after the operation in a reasonable amount of time so as to allow the surgeon to take decisions in the operation room.
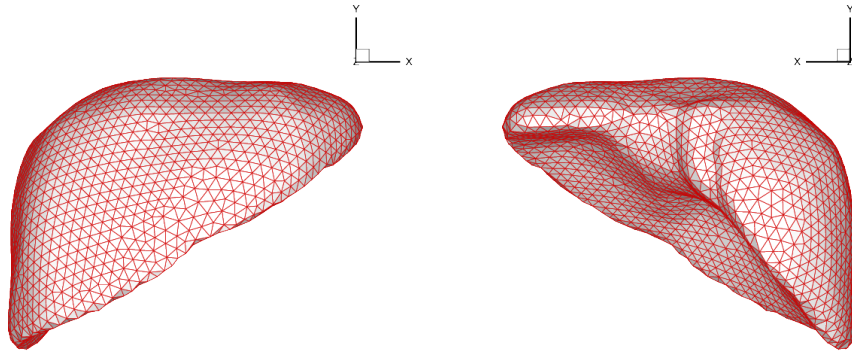
Figure 19. Geometry of the finite element model for the liver.

But, notably, minimally invasive surgery simulation and training systems, equipped with haptic (force feedback) devices require us to provide responses at kHz rates.

The liver is the biggest gland in the human body, after the skin. As in [38], liver geometry has been obtained from the SOFA project [4] and post-processed in order to obtain a mesh composed by 2853 nodes and 10519 tetrahedra, whose geometry is shown in Fig. 19. The anterior surface of the liver is considered free, while the posterior one was assumed to be supported over different organs (it is connected to the diaphragm by the coronary ligament, for instance). The inferior vena cava travels along the posterior surface, and the liver is frequently assumed clamped a that location. Although the assumed boundary conditions are not strictly correct from a physiological point of view, our main interest is to show that the model can be solved under real-time constraints with reasonable accuracy. Here, the human liver is considered, for simplicity, as a linear elastic material with $E = 0.17$ MPa and $\nu = 0.48$ [15].

Simulating a palpation, a ramp load of 5 N is applied at a particular point of the liver surface during a period of 0.25 seconds, and then released during other 0.25 seconds. The liver is then left vibrating free. Even if the liver tissue is well-known to posses some kind of viscoelastic properties, these have been neglected. The purpose of this example is not to obtain an extremely realistic simulation from a physiological point of view, but to show the performance of the technique. In particular, the influence of the number of modes chosen to parametrize the space of initial conditions on the long-term behaviour of the solution. To this end, a reference solution has been computed by employing an HHT time integrator and standard finite elements. POD modes have been extracted from this reference solution to construct the basis for the combined PGD-POD integrator. Other approaches, such as non-linear manifold learning [32] could also be employed and most likely would improve these results, but for the time being POD is considered as a suitable method to construct the basis.

It can be noticed that, in the idealized situation of absence of any type of damping, after the load release, the liver continues vibrating indefinitely. A PGD-POD solution has been computed by employing 1, 3 and 7 modes in the basis of the space of initial conditions. It can be noticed, from Fig. 20 that increasing the number of modes, as expected, provides converging results towards the reference solution. In fact, it can be noticed how the reduction in the number of degrees of freedom, very much

Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

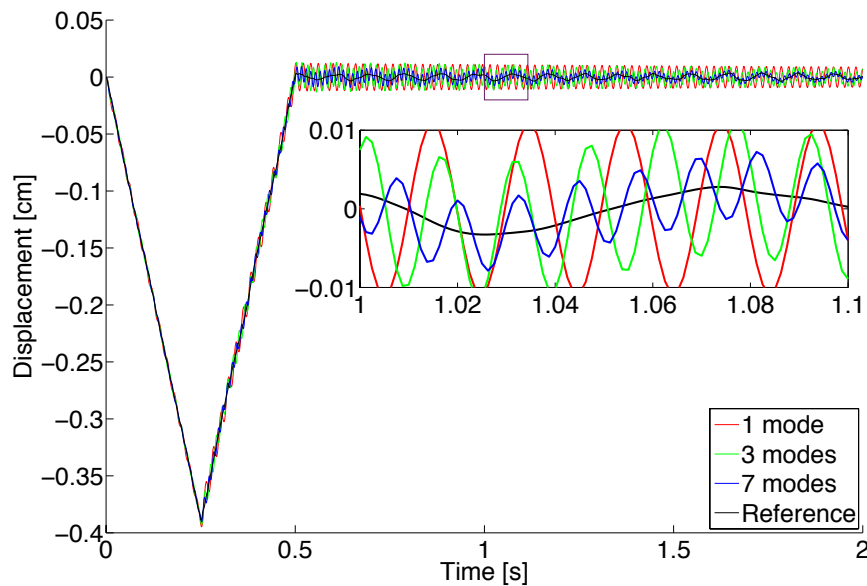*Int. J. Numer. Meth. Engng* 2012; **00**:1–28

Figure 20. Response of the liver for a peak load. Reference (FE) results, and PGD results with basis composed by 1, 3 and 7 modes.

like in classical model order reduction, eliminates high frequencies from the solution (those with the lowest content in energy) and therefore oscillations around the reference solution can be observed. The richer the basis is, the smaller the amplitude of these oscillations and the smaller their period.

It is important to note that, despite this loose of high-frequency modes in the response, common to all model reduction methods, we are considering applications (particularly virtual surgery) where important dissipation appears. In the case of soft living tissues, visco-hyperelasticity is always a common assumption (see [16] and references therein). But no dynamic phenomena beyond 1 Hz is relevant from a biological point of view. From the point of view of the haptic response, the curve in Fig. 20 indicates that the loose in high-frequency modes is accompanied by low amplitudes, impossible to perceive by the human sense of touch.

In addition to that, of course, not only the on-line part of the method is important. For a method to be useful, the off-line part should be affordable. Results for this section took between 45 minutes to one hour if one single mode is considered for the parametrization of the space of initial conditions and some 12-15 hours if eight modes were considered. This is still a very reasonable amount of CPU time if we consider that this off-line part is performed once for life.

7. Conclusions

In this paper a combined PGD-POD technique has been presented for the solution of solid dynamics problems. By formulating the problem as a parametric one, where initial conditions are the parameters, the proposed method is able to generate a sort of "black-box" integrator that takes displacement values

at previous time steps as inputs and gives the solution for the current one. Within this black box any time integrator can be used. The technique here presented is aimed at solving the parametric problem, not at proposing new integrators.

This appealing approach presents, however, some difficulties. One is the enormous amount of degrees of freedom generated if every nodal displacement representing initial conditions is taken as a parameter of the formulation. To this end, an optimal bests, from the POD point of view, is used by utilizing results of similar problems computed off-line. But even then a parametric problem should be solved that may involve tens of parameters. To overcome the curse of dimensionality provoked by this fact, a PGD approach has been employed.

Thus, by computing off-line a general solution (very much like the spirit of a response surface or meta-model, but computed without the need for prior computer experiments) a kind of computational vademecum is obtained that can be emptied on-line at extremely high feedback rates, on the order of one kHz.

Previous works of the authors have shown how to include within this general formulation parameters such as load position, value and orientation, and how to efficiently solve non-linearities arising from material constitutive equations [37] [38]. This work on the dynamics of the problem somehow closes one fundamental aspect: that related with the interactivity of the simulation with the user, thus configuring a true dynamic data driven application system (DDDAS).

Some fundamental aspects remain open, however. For instance, within the computational surgery applications, surgery cutting is a fundamental aspect to achieve realism in the simulation. Within this framework, an algorithm should be developed able to predict the response of the organ to a surgical cut placed at any point of the model and with arbitrary length. This constitutes one of our current efforts of research.

<div align="center">REFERENCES</div>

1. I. Alfaro and D. Bel. Model order reduction of structures with moving loads. Technical report, ENSAM Paris, 2007.
2. I. Alfaro, D. Gonzalez, F. Bordeu, A. Leygue, A. Ammar, E. Cueto, and F. Chinesta. Real-time in silico experiments on gene regulatory networks and surgery simulation on handheld devices. *Journal of Computational Surgery*, in press, 2013.
3. Jérémie Allard, Stéphane Cotin, François Faure, Pierre-Jean Bensoussan, François Poyer, Christian Duriez, Hervé Delingette, and Laurent Grisoni. SOFA an Open Source Framework for Medical Simulation. In *Medicine Meets Virtual Reality (MMVR'15)*, Long Beach, USA, February 2007.
4. A. Ammar, F. Chinesta, P. Diez, and A. Huerta. An error estimator for separated representations of highly multidimensional models. *Computer Methods in Applied Mechanics and Engineering*, 199(25-28):1872 – 1880, 2010.
5. A. Ammar, B. Mokdad, F. Chinesta, , and R. Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. part ii: transient simulation using space-time separated representations. *J. Non-Newtonian Fluid Mech.*, 144:98–121, 2007.
6. A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *J. Non-Newtonian Fluid Mech.*, 139:153–176, 2006.
7. Jernej Barbič and Doug James. Time-critical distributed contact for 6-DoF haptic rendering of adaptively sampled reduced deformable models. In Metaxas, D and Popovic, J, editor, *SYMPOSIUM ON COMPUTER ANIMATION 2007: ACM SIGGRAPH/ EUROGRAPHICS SYMPOSIUM PROCEEDINGS*, pages 171–180, 1515 BROADWAY, NEW YORK, NY 10036-9998 USA, 2007. ACM SIGGRAPH; Eurog Assoc, ASSOC COMPUTING MACHINERY. Symposium on Computer Animation, San Diego, CA, AUG 03-04, 2007.
8. Jernej Barbič and Doug L. James. Real-time subspace integration for St. Venant-Kirchhoff deformable models. *ACM Transactions on Graphics (SIGGRAPH 2005)*, 24(3):982–990, August 2005.
9. K. J. Bathe. Conserving energy and momentum in nonlinear dynamics: A simple implicit time integration scheme. *Computers and Structures*, 85:437–445, 2007.
10. L. Boucinha, A. Gravouil, and A. Ammar. Space-time proper generalized decompositions for the resolution of transient elastodynamic models. *Computer Methods in Applied Mechanics and Engineering*, 255(0):67 – 88, 2013.

Copyright © 2012 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2012; **00**:1–28

11. F. Chinesta, A. Ammar, and E. Cueto. Recent advances in the use of the Proper Generalized Decomposition for solving multidimensional models. *Archives of Computational Methods in Engineering*, 17(4):327–350, 2010.

12. F. Chinesta, A. Leygue, F. Bordeu, J.V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, and A. Huerta. PGD-Based Computational Vademecum for Efficient Design, Optimization and Control. *Archives of Computational Methods in Engineering*, 20(1):31–59, 2013.

13. Francisco Chinesta, Pierre Ladeveze, and Elias Cueto. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18:395–404, 2011.

14. E. Cueto and F. Chinesta. Real time simulation for computational surgery. a review. *Advanced Modeling and Simulation in Engineering Sciences*, submitted, 2013.

15. H. Delingette and N. Ayache. Soft tissue modeling for surgery simulation. In N. Ayache, editor, *Computational Models for the Human Body*, Handbook of Numerical Analysis (Ph. Ciarlet, Ed.), pages 453–550. Elsevier, 2004.

16. A. García, M. A. Martínez, and E. Peña. Viscoelastic properties of the passive mechanical behavior of the porcine carotid artery: Influence of proximal and distal positions. *Biorheology*, 49:271–288, 2012.

17. D. Gonzalez, A. Ammar, F. Chinesta, and E. Cueto. Recent advances on the use of separated representations. *International Journal for Numerical Methods in Engineering*, 81(5), 2010.

18. D. Gonzalez, F. Masson, F. Poulhaon, E. Cueto, and F. Chinesta. Proper generalized decomposition based dynamic data driven inverse identification. *Mathematics and Computers in Simulation*, 82:1677–1695, 2012.

19. Ch. Heyberger, P.-A. Boucard, and D. Neron. A rational strategy for the resolution of parametrized problems in the {PGD} framework. *Computer Methods in Applied Mechanics and Engineering*, 259(0):40 – 49, 2013.

20. H. M. Hilber, T. J. R. Hughes, and R. L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and Structural Dynamics*, 5:283–292, 1977.

21. S. R. Idelsohn and R. Cardona. A reduction method for nonlinear structural dynamics analysis. *Computer Methods in Applied Mechanics and Engineering*, 49:253–279, 1985.

22. K. Karhunen. Uber lineare methoden in der wahrscheinlichkeitsrechnung. *Ann. Acad. Sci. Fennicae, ser. Al. Math. Phys.*, 37, 1946.

23. P. Krysl, S. Lall, and J.E. Marsden. Dimensional model reduction in non-linear finite element dynamics of solids and structures. *Int. J. Numer. Meth. in Engng.*, 51:479–504, 2001.

24. P. Ladeveze. *Nonlinear Computational Structural Mechanics*. Springer, N.Y., 1999.

25. P. Ladeveze, J.-C. Passieux, and D. Neron. The latin multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(21-22):1287 – 1296, 2010.

26. Pierre Ladeveze and Ludovic Chamoin. On the verification of model reduction methods based on the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 200(23-24):2032–2047, 2011.

27. Sanjay Lall, Petr Krysl, and Jerrold E Marsden. Structure-preserving model reduction for mechanical systems. *Physica D: Nonlinear Phenomena*, 184(1-4):304–318, 2003.

28. R. B. Laughlin and David Pines. The theory of everything. *Proceedings of the National Academy of Sciences*, 97(1):28–31, 2000.

29. M. M. Loève. *Probability theory*. The University Series in Higher Mathematics, 3rd ed. Van Nostrand, Princeton, NJ, 1963.

30. E. N. Lorenz. *Empirical Orthogonal Functions and Statistical Weather Prediction*. MIT, Departement of Meteorology, Scientific Report Number 1, Statistical Forecasting Project, 1956.

31. M. Meyer and H. G. Matthies. Efficient model reduction in non-linear dynamics using the Karhunen-Loève expansion and dual-weighted-residual methods. *Computational Mechanics*, 31(1-2):179–191, 2003.

32. Daniel Millán and Marino Arroyo. Nonlinear manifold learning for model reduction in finite elastodynamics. *Computer Methods in Applied Mechanics and Engineering*, 261-262(0):118 – 131, 2013.

33. J. P. Moitinho de Almeida. A basis for bounding the errors of proper generalised decomposition solutions in solid mechanics. *International Journal for Numerical Methods in Engineering*, 94(10):961–984, 2013.

34. R.E. Nickell. Nonlinear dynamics by mode superposition. *Computer Methods in Applied Mechanics and Engineering*, 7(1):107 – 129, 1976.

35. S. Niroomandi, I. Alfaro, E. Cueto, and F. Chinesta. Real-time deformable models of non-linear tissues by model reduction techniques. *Computer Methods and Programs in Biomedicine*, 91(3):223 – 231, 2008.

36. S. Niroomandi, I. Alfaro, E. Cueto, and F. Chinesta. Accounting for large deformations in real-time simulations of soft tissues based on reduced-order models. *Computer Methods and Programs in Biomedicine*, 105(1):1–12, 2012.

37. S. Niroomandi, D. González, I. Alfaro, F. Bordeu, A. Leygue, E. Cueto, and F. Chinesta. Real-time simulation of biological soft tissues: a pgd approach. *International Journal for Numerical Methods in Biomedical Engineering*, 29(5):586–600, 2013.

38. S. Niroomandi, D. Gonzalez, I. Alfaro, E. Cueto, and F. Chinesta. Model order reduction in hyperelasticity: a proper generalized decomposition approach. *International Journal for Numerical Methods in Engineering*, 96(3):129–149, 2013.

39. Siamak Niroomandi, Iciar Alfaro, Elias Cueto, and Francisco Chinesta. Model order reduction for hyperelastic materials. *International Journal for Numerical Methods in Engineering*, 81(9):1180–1206, 2010.

40. A. NOOR and J. PETERS. *Reduced basis technique for nonlinear analysis of structures*. American Institute of Aeronautics and Astronautics, 2013/10/16 1979.

41. Anthony Nouy. A priori model reduction through Proper Generalized Decomposition for solving time-dependent partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 199(23-24):1603–1626, 2010.
42. H. M. Park and D. H. Cho. The use of the Karhunen-Loève decomposition for the modeling of distributed parameter systems. *Chemical Engineering Science*, 51(1):81–98, 1996.
43. A. Radermacher and S. Reese. Proper orthogonal decomposition-based model reduction for nonlinear biomechanical analysis. *International Journal of Materials Engineering Innovation*, 4(4):149–165, 2013.
44. L. Sirovich. Turbulence and the dynamics of coherent structures part I: coherent structures. *Quaterly of applied mathematics*, XLV:561–571, 1987.
45. Z.A. Taylor, M. Cheng, and S. Ourselin. High-speed nonlinear finite element analysis for surgical simulation using graphics processing units. *Medical Imaging, IEEE Transactions on*, 27(5):650 –663, may 2008.
46. Z.A. Taylor, S. Crozier, and S. Ourselin. A reduced order explicit dynamic finite element algorithm for surgical simulation. *Medical Imaging, IEEE Transactions on*, 30(9):1713 –1721, sept. 2011.
47. Z.A. Taylor, S. Ourselin, and S. Crozier. A reduced order finite element algorithm for surgical simulation. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 239 –242, 31 2010-sept. 4 2010.
48. Otto von Estorff and Christian Hagen. Iterative coupling of fem and bem in 3d transient elastodynamics. *Engineering Analysis with Boundary Elements*, 30(7):611 – 622, 2006.