Bridging Proper Orthogonal Decomposition methods and augmented Newton-Krylov algorithms: an adaptive model order reduction for highly nonlinear mechanical problems

Pierre Kerfriden, Pierre Gosselet, Sondipon Adhikari, Stephane Pierre-Alain Bordas

Bridging Proper Orthogonal Decomposition methods and augmented Newton-Krylov algorithms: an adaptive model order reduction for highly nonlinear mechanical problems

P. Kerfriden[1], P. Gosselet[2] *, S. Adhikari[3] †, S. Bordas[1] ‡

[1] Cardiff University
Queen's Buildings, The Parade, Cardiff CF24 3AA, Wales, UK
[2] Ecole Normale Superieure de Cachan
61 Avenue du Président Wilson, F-94230 Cachan, France
[3] Swansea University
Singleton Park, Swansea, SA2 8PP, Wales UK

September 22, 2011

Abstract

This article describes a bridge between POD-based model order reduction techniques and the classical Newton/Krylov solvers. This bridge is used to derive an efficient algorithm to correct, "on-the-fly", the reduced order modelling of highly nonlinear problems undergoing strong topological changes. Damage initiation problems are addressed and tackle via a corrected hyperreduction method. It is shown that the relevancy of reduced order model can be significantly improved with reasonable additional costs when using this algorithm, even when strong topological changes are involved.

Keywords: Model order reduction (MOR); Time-dependant nonlinear mechanical problems; Proper orthogonal decomposition (POD); Newton/Krylov Solver; Projected conjugate gradient; System approximation; Hyperreduction; CH-POD; Damage propagation;

1 Introduction

The simulation of failure in complex materials has been one important issue throughout engineering. Micro and mesoscale models have demonstrated their ability to predict complex phenomena such as crack initiation and propagation in various fields: composite materials in aeronautics [1, 2], cement-based structures [3] in civil engineering or biostructures [4] in medical engineering. However, these models usually describe the material at a very fine scale compared to the size of the structure, and the finite element discretization of the underlying partial differential equations leads to large, potentially highly nonlinear (therefore requiring a fine discretization in time), numerical problems. Various multiscale computational strategies have been developed to tackle this important issue, such as enrichment

*CNRS Research Associate
†Chair of Aerospace Engineering
‡Royal Academy of Engineering/Leverhulme Senior Research Fellow

techniques [5, 6], homogenization strategies [7, 8], domain decomposition methods [9, 10], or model order reduction[11, 12]. The most efficient of these techniques couple several of these advanced tools to efficiently perform multiscale simulations [13, 14, 15, 16, 10, 17, 18].

Among them, Proper Orthogonal Decomposition-based (POD) [11, 12, 19] model order reduction strategies provide extremely valuable tools for an automatic derivation of a multiscale computational method (no *a priori* physical understanding of the different scales involved is required). They consist in using a set of potential solutions to the initial problem (snapshot) and extracting, by a spectral analysis a few basis vectors spanning a space of small dimension in which the solution to the, initially large, numerical problem is well approximated [20]. These techniques have been proved extremely efficient when speed is more important that accuracy, for instance if the goal is to provide an approximation of the response of a complex structure in real-time (an example of biomedical application is given in [21]), or quasi real-time (in the case of early stage design). This type of applications is our main focus.

In the context of structural problems involving plasticity or damage, two severe drawbacks limit the direct application of POD-based model order reduction:

- The initial snapshot might be too poor to represent accurately the solution of the damaged structure. This can happen if:
  - The "a priori" mechanical understanding of the structure is too poor, which might result in a bad choice of the snapshot simulations.
  - The number of parameters involved is too important. The snapshot needed to compute a relevant reduced basis might be very large, and its reduction inefficient.
  - Strong topological changes in the structure occur. Improvements have been proposed by Ryckelynk's team [22] where the reduced basis is enriched during the computation. When required, a Krylov subspace associated to a linearised operator of the initial problem is generated, and a few additional basis vectors are obtained by a spectral analysis of this space. However, it seems that, when dealing with complex constitutive laws, this adaptive strategy can be computationally expensive [18].
- The integration of the constitutive law needs to be done at each integration point, regardless of the dimension of the reduced space. These problems have been handled in [23] by a technique called hyperreduction, which consists in considering only a few local residuals to compute the internal forces.

On the other hand, various model order reduction strategies have also been used to derive good initializations and preconditioners for iterative algorithms for complex nonlinear problems at the fine scale [24, 25, 9, 15, 16].

We propose here a novel strategy that couples these two approaches. Our vision is that if complex changes in the topology of the structure appear (local crack initiation for instance), they can only be accurately predicted by solving the fine scale model, at least locally. This can be done, for instance, by adding the finite element shape functions to the snapshot, or by using relocalization techniques. However, the long-range effects of these topological changes do not require the fine description, and can be obtained by solving the full system very crudely. Hence, when the global residual exceeds a given threshold value, we propose to perform conjugate gradient iterations on the full system, orthogonally to the snapshot space by means of a classical projection. The new vector obtained by the

conjugate gradient does not belong to the initial approximation space and is added to the reduced basis. It is interpreted as an"on-the-fly" enrichment of the reduced model. The proposed algorithm largely differs from what had been done in [22] in the sense that only a single vector belonging to the Krylov algorithm is used to enrich the reduced basis, for each of the corrections performed within a time step. It thus keeps the size of the reduced basis to a minimum. In addition, the solution of the current time step is not restarted after enriching the reduced basis. At last, using the properties of the projected Krylov during the correction steps automatically ensures the required orthogonality between the additional reduced basis vectors and the initial reduced basis.

This strategy can be interpreted as a bridge between "exact" Newton-Krylov strategies, and projection-based model order reduction methods (like the POD) for nonlinear problems. Indeed, setting the residual threshold to a low value leads to the former strategy, while setting it to a high value leads to the latter. An intermediate value yields an adaptive model order reduction method with control of the global residual. We will show that this method permits to obtain a correction of the reduced model at cheap price, when the current reduced basis is no more sufficient to represent the solution accurately.

The paper is organized as follows. In section 1, the nonlinear system of equations resulting from the discretization of a continuum mechanics problem is introduced, and the model order reduction and associated time solution procedure are described. In section 2, we perform the reduction of a damage model, and show that the basic application of the proper orthogonal decomposition is inadequate. The corrective reduced order modelling is introduced in section 3, and an exhaustive numerical study of its efficiency is proposed in section 4. In the last section, this technique is applied to the "on-the-fly" correction of hyperreduced damage models.

2 Problem statement and classical model order reduction for nonlinear problems

2.1 Problem statement

2.1.1 Nonlinear structural Problem

Let us consider a structure occupying a continuous domain Ω with boundary $\partial\Omega$. This structure is subjected to prescribed displacements $\underline{U}_{\mathrm{D}}$ on its boundary $\partial\Omega_u$, over time interval $[0,T]$. Let $\underline{u}$ be the unknown displacement field, it belongs to the space $\mathcal{U}$ of kinematically admissible fields:

$$\mathcal{U} = \left\{ \underline{u} \in H^1(\Omega) \mid \underline{u}_{|\partial\Omega_u} = \underline{U}_{\mathrm{D}} \right\} \tag{1}$$

Let $\mathcal{U}_0$ be the associated vector space. Prescribed tractions $\underline{F}_{\mathrm{D}}$ are applied on the complementary boundary $\partial\Omega_f = \partial\Omega \backslash \partial\Omega_u$. Under the assumptions of quasi-static evolution of the structure and small perturbations, the weak form of the equilibrium and the constitutive relation read, at any time $t \in [0,T]$:

$$\begin{aligned} &\forall \underline{u}^\star \in \mathcal{U}_0, \text{ find } \underline{u} \in \mathcal{U} \text{ such that:} \\ &\int_\Omega \underline{\underline{\sigma}} : \underline{\underline{\epsilon}}(\underline{u}^\star)\ d\Omega = \int_\Omega \underline{f}_{\mathrm{D}} . \underline{u}^\star\ d\Omega + \int_{\partial\Omega_f} \underline{F}_{\mathrm{D}} . \underline{u}^\star\ d\Gamma \\ &\underline{\underline{\sigma}} = \underline{\underline{\sigma}} \left(\left(\underline{\underline{\epsilon}}(\underline{u})_{|\tau} \right)_{\tau \leq t} \right) \end{aligned} \tag{2}$$

where $\underline{\underline{\sigma}}$ is the Cauchy stress tensor and $\underline{\underline{\epsilon}}(\underline{u})$ is the symmetric part of the gradient of displacements. The constitutive relation between $\underline{\underline{\sigma}}$ and $\underline{\underline{\epsilon}}(\underline{u})$ is nonlinear and described

using internal variables (for instance damage or plasticity). It is assumed to be local and rate-independent.

2.1.2 Space discretization

We perform a standard finite element approximation of the space of unknown displacement field $\mathcal{U}$ (and a similar approximation of the space of test functions $\mathcal{U}_0$):

$$\mathcal{U}^h(\Omega) = \left\{ \underline{u}(\underline{x}) \mid \underline{u}(\underline{x}) = \sum_{i=1}^{n_n} N_i(\underline{x})\, \underline{u}_i \right\} \tag{3}$$

where n_n is the number of nodes, $\underline{x}$ is the position vector, $(N_i)_{i\in[\![1\ n_n]\!]}$ are the finite element shape functions, and $(\underline{u}_i)_{i\in[\![1\ n_n]\!]}$ the nodal values of the displacement field.

The introduction of the finite element approximation into equations (1) and (2) leads to the following nonlinear vectorial equation at any time $t \in [0, T]$:

$$\underline{\mathbf{F}}_{\text{Int}}\left(\left(\underline{\mathbf{U}}_{|\tau}\right)_{\tau\leq t}\right) + \underline{\mathbf{F}}_{\text{Ext}} = \underline{\mathbf{0}} \tag{4}$$

where $\underline{\mathbf{U}} \in \mathbb{R}^{n_u}$ (n_u is the number of nodal unknowns) is the vector of nodal displacement unknowns, $\underline{\mathbf{F}}_{\text{Int}} \in \mathbb{R}^{n_u}$ and $\underline{\mathbf{F}}_{\text{Ext}} \in \mathbb{R}^{n_u}$ are the internal forces resulting from the discretization of the internal virtual work (left-hand side of (2)) and the external forces resulting from the discretization of the external virtual work (right-hand side of (2)), respectively.

2.1.3 Time discretization

The nonlinear solution strategy used here is a classical time discretization scheme for quasi-static and rate-independent problems. This procedure consists in finding a set of consecutive solutions at times $(t_n)_{n\in[\![0\ n_t]\!]}$ (see [26]). Hence, the constitutive law (2) is discrete in time. This time discretization scheme yields the following vectorial system of n_u nonlinear equations at time t_{n+1}:

$$\underline{\mathbf{F}}_{\text{Int}}\left(\left(\underline{\mathbf{U}}_{|t_m}\right)_{m\in[\![0\ n+1]\!]}\right) + \underline{\mathbf{F}}_{\text{Ext}} = \underline{\mathbf{0}} \tag{5}$$

For the sake of clarity, the dependency of the internal forces with the history of the unknown fields will not be written explicitly.

2.1.4 Nonlinear solution strategy

At each time t_{n+1} the nonlinear system of equations is solved by a Newton-Raphson algorithm. At any iteration $(i+1)$ of the algorithm, the linearisation of equation (5) around $\underline{\mathbf{U}}^i$ leads to the following prediction stage:

$$\underline{\underline{\mathbf{K}}}^i_{\text{T}}\, \underline{\boldsymbol{\delta}\mathbf{U}}^{i+1} = -\underline{\mathbf{R}}^i \tag{6}$$

where we have introduced the residual vector $\underline{\mathbf{R}} = \underline{\mathbf{F}}_{\text{Ext}} + \underline{\mathbf{F}}_{\text{Int}}(\underline{\mathbf{U}})$, and the correction of the increment of displacement $\underline{\boldsymbol{\delta}\mathbf{U}}^{i+1} = \underline{\mathbf{U}}^{i+1} - \underline{\mathbf{U}}^i$. This linear prediction is followed by a correction stage:

$$\underline{\mathbf{R}}^{i+1} = \underline{\mathbf{F}}_{\text{Ext}} + \underline{\mathbf{F}}_{\text{Int}}(\underline{\mathbf{U}}^{i+1}) \qquad \underline{\underline{\mathbf{K}}}^{i+1}_{\text{T}} = \left.\frac{\partial \underline{\mathbf{F}}_{\text{Int}}(\underline{\mathbf{U}})}{\partial \underline{\mathbf{U}}}\right|_{\underline{\mathbf{U}}=\underline{\mathbf{U}}^{i+1}} \tag{7}$$

2.2 Model order reduction

The set of equations (5) can be very large if multiscale problems are simulated (see for instance [6, 16]). Various strategies can be used to reduce the size of this system without loosing accuracy. We apply here the classical model order reduction by projection to the solution of problem (5).

2.2.1 Reduction by projection

The solution vector is searched for in a space of small dimension (several orders smaller than the number of finite element degrees of freedom). Let us call $\underline{\underline{\mathbf{C}}}$ the matrix whose columns form a basis of this space:

$$\underline{\underline{\mathbf{C}}} = \left(\begin{array}{cccc} \underline{\mathbf{C}}^1 & \underline{\mathbf{C}}^2 & \ldots & \underline{\mathbf{C}}^{n_\mathrm{C}} \end{array} \right) \tag{8}$$

where n_C is the dimension of the reduced space, and $(\underline{\mathbf{C}}^k)_{k\in[\![1\ n_\mathrm{C}]\!]} \in (\mathbb{R}^{n_u})^{n_\mathrm{C}}$ are the chosen reduced basis vectors. Applied to the reduction of problem (5), the solution field is searched for under the form:

$$\underline{\mathbf{U}} = \underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}} \tag{9}$$

The residual of equation (5) is constrained to be orthogonal to a space of small dimension, which can, in theory, be different from the one spanned by $(\underline{\mathbf{C}}^k)_{k\in[\![1\ n_\mathrm{C}]\!]}$. Let us limit our study to the Galerkin procedure, which writes the following constraint:

$$\underline{\underline{\mathbf{C}}}^T\,\underline{\mathbf{R}} = \underline{\mathbf{0}} \tag{10}$$

Hence, the reduced form of problem (5) reads:

$$\underline{\underline{\mathbf{C}}}^T \left(\underline{\mathbf{F}}_\mathrm{Ext} + \underline{\mathbf{F}}_\mathrm{Int}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}}) \right) = \underline{\mathbf{0}} \tag{11}$$

2.2.2 Solution procedure for the reduced nonlinear problem

At each time step of the time discretization scheme, the reduced problem (11) is solved by a Newton-Raphson algorithm. The $(i+1)^{th}$ prediction stage is performed using the following set of n_C linearised equation:

$$\underline{\underline{\mathbf{K}}}^i_{\mathrm{T,R}}\,\underline{\boldsymbol{\delta\alpha}}^{i+1} = -\underline{\mathbf{R}}^i_R \tag{12}$$

where the residual of the reduced nonlinear problem (11) is defined by $\underline{\mathbf{R}}_R = \underline{\underline{\mathbf{C}}}^T\,\underline{\mathbf{F}}_\mathrm{Ext} + \underline{\underline{\mathbf{C}}}^T\,\underline{\mathbf{F}}_\mathrm{Int}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}})$ and the increment $\underline{\boldsymbol{\delta\alpha}}^{i+1} = \underline{\boldsymbol{\alpha}}^{i+1} - \underline{\boldsymbol{\alpha}}^i$. The correction stage reads:

$$\underline{\mathbf{R}}^{i+1}_R = \underline{\underline{\mathbf{C}}}^T\,\underline{\mathbf{F}}_\mathrm{Ext} + \underline{\underline{\mathbf{C}}}^T\,\underline{\mathbf{F}}_\mathrm{Int}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}}^{i+1}) \qquad \underline{\underline{\mathbf{K}}}^{i+1}_{\mathrm{T,R}} = \underline{\underline{\mathbf{C}}}^T\,\left.\frac{\partial \underline{\mathbf{F}}_\mathrm{Int}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}})}{\partial \underline{\boldsymbol{\alpha}}}\right|_{\underline{\boldsymbol{\alpha}}=\underline{\boldsymbol{\alpha}}^{i+1}} \tag{13}$$

There is a simple link between the previous solution procedure for the reduced problem (11) and the systematic reduction of the prediction stages performed on the full nonlinear problem (6) since $\underline{\underline{\mathbf{K}}}^i_{\mathrm{T,R}}$ can be expanded as

$$\underline{\underline{\mathbf{K}}}^i_{\mathrm{T,R}} = \underline{\underline{\mathbf{C}}}^T\,\left.\frac{\partial \underline{\mathbf{F}}_\mathrm{Int}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}})}{\partial \underline{\boldsymbol{\alpha}}}\right|_{\underline{\boldsymbol{\alpha}}=\underline{\boldsymbol{\alpha}}^i} = \underline{\underline{\mathbf{C}}}^T\,\left.\frac{\partial \underline{\mathbf{F}}_\mathrm{Int}(\underline{\mathbf{U}})}{\partial \underline{\mathbf{U}}}\right|_{\underline{\mathbf{U}}=\underline{\mathbf{U}}_{|t_n}+\underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}}^i} \left.\frac{\partial \underline{\mathbf{U}}}{\partial \underline{\boldsymbol{\alpha}}}\right|_{\underline{\boldsymbol{\alpha}}=\underline{\boldsymbol{\alpha}}^i} = \underline{\underline{\mathbf{C}}}^T\,\underline{\underline{\mathbf{K}}}^i_\mathrm{T}\underline{\underline{\mathbf{C}}} \tag{14}$$

This result can be injected in equation (12), which yields:

$$\underline{\underline{\mathbf{C}}}^T(\underline{\mathbf{R}}^i + \underline{\underline{\mathbf{K}}}^i_{\mathrm{T}}\,\underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\delta\alpha}}^{i+1}) = \underline{\mathbf{0}} \tag{15}$$

This last equation is a simple reduction by projection of the $(i+1)^{th}$ linear prediction of the Newton-Raphson scheme (6) used to solve the initial problem (5). In other words, the prediction stage of any Newton iteration performed on the reduced problem yields exactly the same solution as a reduction of the linear prediction stage used to solve the initial problem.

2.2.3 POD reduced approximation space

The snapshot-POD is a projection-based model order reduction technique which requires the knowledge of a representative family of solutions to the global problem. This set of vectors is called $(\underline{\mathbf{S}}^k)_{k\in[\![1\ n_{\mathrm{S}}]\!]}$ and forms an operator $\underline{\underline{\mathbf{S}}} = \left(\begin{array}{cccc}\underline{\mathbf{S}}^1 & \underline{\mathbf{S}}^2 & \ldots & \underline{\mathbf{S}}^{n_{\mathrm{S}}}\end{array}\right)$. The aim is to find an orthonormal basis $(\underline{\mathbf{C}}^k)_{k\in[\![1\ n_{\mathrm{C}}]\!]}$, of dimension n_{C} smaller than n_{S} such that the distance between spaces $\mathrm{Im}(\mathbf{C})$ and $\mathrm{Im}(\underline{\underline{\mathbf{S}}})$ is minimum (in the sense of the Frobenius norm).

This problem is classically solved by computing the singular value decomposition of $\underline{\underline{\mathbf{S}}}$:

$$\underline{\underline{\mathbf{S}}} = \underline{\underline{\mathbf{U}}}\,\underline{\underline{\boldsymbol{\Sigma}}}\,\underline{\underline{\mathbf{V}}}^T = \sum_{k=1}^{n_{\mathrm{S}}} \Sigma^k\,\underline{\mathbf{U}}^k\,\underline{\mathbf{V}}^{k\,T} \tag{16}$$

where $\underline{\underline{\mathbf{U}}} = \left(\begin{array}{cccc}\underline{\mathbf{U}}^1 & \underline{\mathbf{U}}^2 & \ldots & \underline{\mathbf{U}}^{n_u}\end{array}\right)$ and $\underline{\underline{\mathbf{V}}} = \left(\begin{array}{cccc}\underline{\mathbf{V}}^1 & \underline{\mathbf{V}}^2 & \ldots & \underline{\mathbf{V}}^{n_{\mathrm{S}}}\end{array}\right)$ are two orthonormal matrices of respective sizes n_u and n_{S} and the upper block of $\underline{\underline{\boldsymbol{\Sigma}}}$ is a diagonal matrix of positive entries $(\Sigma^k)_{k\in[\![1\ n_{\mathrm{S}}]\!]}$ (singular values), ordered decreasingly (the lower block is a null matrix). One can show that the optimum reduced snapshot basis $(\underline{\mathbf{C}}^k)_{k\in[\![1\ n_{\mathrm{C}}]\!]}$ is such that:

$$\underline{\underline{\mathbf{C}}} = \left(\begin{array}{cccc}\underline{\mathbf{U}}^1 & \underline{\mathbf{U}}^2 & \ldots & \underline{\mathbf{U}}^{n_{\mathrm{C}}}\end{array}\right) \tag{17}$$

2.3 POD for the reduction of damage evolution problems

2.3.1 Damageable lattice structure

We focus on a very simple damageable lattice structure, made of bars under traction or compression (figure (1)). Each of the elementary bars occupies the domain Ω_b such that $\Omega = \bigcup_{b\in[\![1\ n_b]\!]}\Omega_b$, where n_b is the number of bars. The direction of the bar is $\underline{n}_{kl} = \underline{P_kP_l}/\|\underline{P_kP_l}\|$ where P_k and P_l are the two nodes connected to Bar b. The displacement field in Ω_b is searched for under the form:

$$\underline{u} = u(\tilde{y})\,\underline{n}_{kl} \tag{18}$$

where $\tilde{y} \in [0 \quad \|\underline{P_kP_l}\|]$. We suppose that no volume force is applied on the structure, that the material of each elementary bar is isotropic and homogeneous (constant material stiffness along the direction of the bar), and that the section of the bars is constant and equal to $(S_b)_{b\in[\![1\ n_b]\!]}$. Hence, the unknown displacement field is linear along the direction of the bar.

A very basic constitutive law, based on classical damage mechanics [1], is used to described the behaviour of the lattice structure. The lineic strain energy of each bar reads:

$$e_d = \frac{1}{2}E(1-d)S_b\epsilon^2 \tag{19}$$

where $\epsilon(\underline{u}) = u_{,\tilde{y}}(\tilde{y})$ is the strain measured in the direction of the bar, and d is a damage variable which ranges from 0 (undamaged material) to 1 (completely damaged material) and is constant within each elementary bar. Two state equations can be derived from the strain energy. The first one links the normal force $N = S_b\, \underline{n}_{kl}^T . \underline{\underline{\sigma}} . \underline{n}_{kl}$ to the strain, while the second one links a thermodynamic force Y to the strain:

$$N = \frac{\partial e_d}{\partial \epsilon} = E(1-d)S_b\epsilon \qquad\qquad Y = -\frac{\partial e_d}{\partial d} = \frac{1}{2}ES_b\epsilon^2 \tag{20}$$

An evolution law is defined to link the damage variable to the thermodynamic force:

$$d = \min\left(1, \sup_{\tau \leq t}\left(\alpha (Y_{|\tau})^{\beta}\right)\right) \tag{21}$$

The results presented in this paper are obtained with the following set of data. The sections and Young's modulus are unitary. The material parameters are $\beta = 0.5$ and $\alpha = \sqrt{2}$. Each vertical or horizontal bar has a unitary length.

The finite element discretization of the lattice problem is done by considering that each bar is a linear finite element. Because of the previously made assumptions on the loading and geometry, the finite element solution yields the "exact" solution to the problem.

2.3.2 Newton/arc-length solution scheme

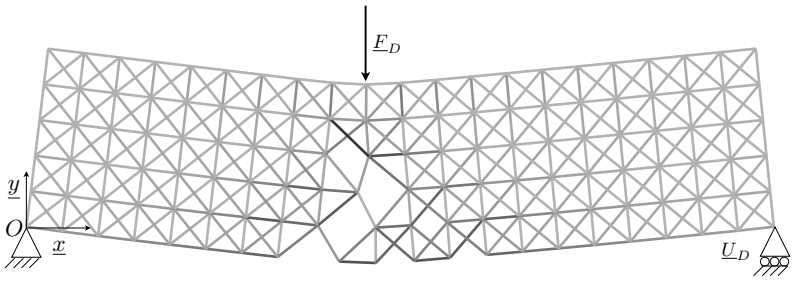

Figure 1: Reference problem. The structure is a lattice made of damageable bars in traction or compression. The problem becomes unstable as damage localizes and propagates.

The family of problems that we want to solve is described in figure (1). The load is applied on the top surface of the structure, in the $\underline{y}$ direction. The damage state represented here (light gray corresponds to $d = 0$, while darker bars are damaged, and ruined bars are removed) is obtained in 60 time increments, each of which is converged to a very low level of error.

This problem being unstable, a local arc-length procedure is combined to the Newton algorithm. This classical [27, 28] procedure is usually derived for a particular case of the modified Newton algorithm, namely the updated secant Newton. In our work, we need a certain versatility to choose the linearised operator. Therefore, the local arc-length procedure is detailed here for tangent and modified Newton algorithms, in a general manner.

At time t_{n+1}, this procedure simply consists in relaxing the norm λ of the prescribed loading, while adding a local constraint on the increments of displacement:

$$\left\{\begin{array}{l} \underline{\mathbf{F}}_{\text{Int}}\left(\left(\underline{\mathbf{U}}_{|t_m}\right)_{m\in[\![0\ n+1]\!]}\right) + \lambda\, \underline{\mathbf{F}}_{\text{Ext}} = \underline{\mathbf{0}} \\ \underline{\underline{\mathbf{c}}}\left(\left(\underline{\mathbf{U}}_{|t_m}\right)_{m\in[\![0\ n+1]\!]}\right) \Delta\underline{\mathbf{U}} = \Delta l \end{array}\right. \tag{22}$$

where $\underline{\mathbf{F}}_{\text{Ext}}$ is normalized, and $\Delta\underline{\mathbf{U}} = \underline{\mathbf{U}}_{|t=t_{n+1}} - \underline{\mathbf{U}}_{|t=t_n}$. $\underline{\underline{\mathbf{c}}}$ is a formal boolean line operator which extracts the maximum difference of displacement increment undertaken by an element (whose stiffness is strictly positive) over the current time step:

$$\underline{\underline{\mathbf{c}}}\left(\left(\underline{\mathbf{U}}_{|t_m}\right)_{m\in[\![0\ n+1]\!]}\right)\Delta\underline{\mathbf{U}} = \max_{\substack{b\in[\![1\,n_b]\!]\\ \forall M\in\Omega_b,\, d|_{M,t=t_n}<1}} \left(\Delta\underline{\mathbf{U}}|_{P_l} - \Delta\underline{\mathbf{U}}|_{P_k}\right).\underline{n}_{kl} \tag{23}$$

where $\Delta\underline{\mathbf{U}}|_{P_k}$ and $\Delta\underline{\mathbf{U}}|_{P_l}$ are the local increments of displacements over the time interval $[t_n, t_{n+1}]$, respectively at corners P_k and P_l of the bar.

These two equations are linearised around point $(\lambda^i, \underline{\mathbf{U}}^i)$, which yields, at step $(i+1)$ of the Newton algorithm, the following linear prediction:

$$\begin{cases} \delta\lambda^{i+1} = \dfrac{-R^i_{\text{cont}} - \underline{\underline{\mathbf{c}}}^i\,(\underline{\underline{\mathbf{K}}}^i_{\text{T}})^{-1}\underline{\mathbf{R}}^i_{\text{eq}}}{\underline{\underline{\mathbf{c}}}^i\,(\underline{\underline{\mathbf{K}}}^i_{\text{T}})^{-1}\underline{\mathbf{F}}_{\text{Ext}}} \\ \boldsymbol{\delta}\underline{\mathbf{U}}^{i+1} = -(\underline{\underline{\mathbf{K}}}^i_{\text{T}})^{-1}\left(\underline{\mathbf{R}}^i_{\text{eq}} + \delta\lambda^{i+1}\,\underline{\mathbf{F}}_{\text{Ext}}\right) \end{cases} \tag{24}$$

with the definition of the residuals of system (22), $\underline{\mathbf{R}}_{\text{eq}} = \underline{\mathbf{F}}_{\text{Int}} + \lambda\,\underline{\mathbf{F}}_{\text{Ext}}$, $R_{\text{cont}} = \Delta l - \underline{\underline{\mathbf{c}}}\,\Delta\underline{\mathbf{U}}$ and the increment $\delta\lambda^{i+1} = \lambda^{i+1} - \lambda^i$. During the correction stage, the tangent stiffness operator and residuals are updated similarly as described for the classical Newton algorithm, equation (7), while the new extraction operator is corrected as follows:

$$\underline{\underline{\mathbf{c}}}^{i+1} = \underline{\underline{\mathbf{c}}}\left(\underline{\mathbf{U}}^{i+1}, \left(\underline{\mathbf{U}}_{|t_m}\right)_{m\in[\![0\ n]\!]}\right) \tag{25}$$

As suggested in [28, 16], we use a quasi-Newton approximation of the previously described tangent Newton-Raphson algorithm. The tangent operator at iteration $\underline{\underline{\mathbf{K}}}^i_{\text{T}}$ is replaced by the stiffness matrix $\underline{\underline{\mathbf{K}}}^i$ obtained by setting the internal variables to previously obtained values (i.e.: last converged solution, or last Newton iteration). This procedure does not yield the quadratic convergence rate of the genuine Newton-Raphson algorithm. Yet, it is widely applied in damage simulations for its simplicity (no need to compute a tangent operator) and robustness.

The same arc-length procedure is of course applied when reducing the nonlinear problem by projection on a reduced basis basis. The expressions of the tangent stiffness and residual are similar to the one given in section (2.2.2) (the residual is slightly modified by the introduction of the loading parameter λ). The linear prediction performed on the reduced problem reads:

$$\begin{cases} \delta\lambda^{i+1} = \dfrac{-R^i_{\text{cont}} - \underline{\underline{\mathbf{c}}}^i\,\underline{\underline{\mathbf{C}}}(\underline{\underline{\mathbf{K}}}^i_{\text{T,R}})^{-1}\underline{\mathbf{R}}^i_R}{\underline{\underline{\mathbf{c}}}^i\,\underline{\underline{\mathbf{C}}}(\underline{\underline{\mathbf{K}}}^i_{\text{T,R}})^{-1}(\underline{\underline{\mathbf{C}}}^T\,\underline{\mathbf{F}}_{\text{Ext}})} \\ \underline{\boldsymbol{\delta\alpha}}^{i+1} = -(\underline{\underline{\mathbf{K}}}^i_{\text{T,R}})^{-1}\left(\underline{\mathbf{R}}^i_R + \delta\lambda^{i+1}\,\underline{\underline{\mathbf{C}}}^T\,\underline{\mathbf{F}}_{\text{Ext}}\right) \end{cases} \tag{26}$$

2.3.3 Model order reduction

In order to reduce the family of problems described in the previous sections by a POD-based projection method, we compute a snapshot of 7 solutions (loads applied on the top surface of the lattice, at positions $x \in \{2\ 5\ 8\ 10\ 12\ 15\ 18\}$), as illustrated in figure (2). The successive loads have a very low value, so that the structure is virtually undamaged. This snapshot is reduced to three basis vectors by the computation of a truncated SVD.

Within the family considered to construct the snapshot, the particular problem which will be simulated in the following is represented in figure (3). The vertical load is applied

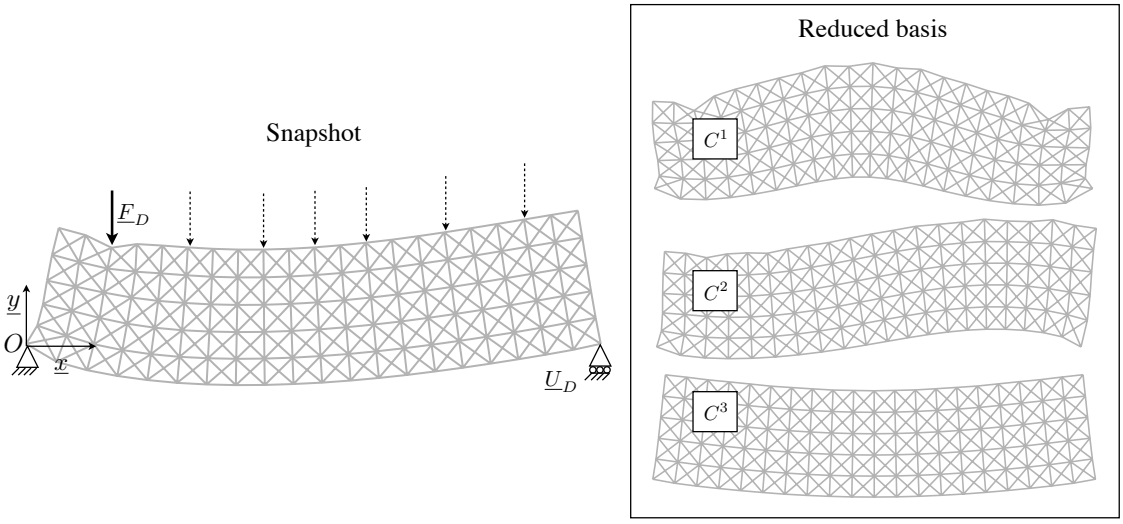

Figure 2: Computation of the snapshot made of seven solutions obtained at the very beginning of the damage process (left) and reduced basis obtained by extracting the singular vectors associated with the three highest singular values of the resulting snapshot operator (right)

at points $x \in \{7\ 8\ 9\}$, and the time of the analysis is discretized in 50 time steps. The reference solution is converged to a very low value of the norm of the residual of the Newton/arc-length algorithm (norm of the residual scaled by the norm of the right-hand side term equal to 10^{-6}).

Given the normalized current solution $\widetilde{\underline{\mathbf{U}}}_{|t} = \frac{\underline{\mathbf{U}}_{|t}}{\|\underline{\mathbf{U}}_{|t}\|_2}$ and normalized reference solution $\widetilde{\underline{\mathbf{U}}}_{\mathrm{ref}_{|t}} = \frac{\underline{\mathbf{U}}_{\mathrm{ref}|t}}{\|\underline{\mathbf{U}}_{\mathrm{ref}|t}\|_2}$, the solution error used to assess the accuracy of the current solution is:

$$\text{solution error}_{|t} = \left\| \widetilde{\underline{\mathbf{U}}}_{|t} - \widetilde{\underline{\mathbf{U}}}_{\mathrm{ref}_{|t}} \right\|_2 \tag{27}$$

In the following developments, a single value can be provided for this error. It is the maximum over time of each of these errors.

In the very early stages of the simulation, the error is very low, which can be seen in figure (5, right) (the dashed line is the error defined by equation (27) as a function of time, while the dotted one is the value of the maximum damage in the structure). However, as damage increases, which eventually leads to the cracked area represented in figure (3), a simple linear combination of the modes of the reduced snapshot is obviously not sufficient to reproduce accurately the solution, which results in a high level of error.

3 Error-controlled adaptive model order reduction

3.1 Principle

In order to handle complex changes in the topology of the structure, we propose to enrich the reduced basis $\mathbf{C}$, "on-the-fly", during the Newton iterations performed on the reduced problem (11). Prior to the actual description of the algorithm, let us remark that, at each iteration of the Newton algorithm, one can compute:

- the relative norm of the residual of the reduced problem (11), i.e.: $\dfrac{\|\underline{\mathbf{R}}_R\|_2}{\|\underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}}_{\mathrm{Ext}}\|_2}$

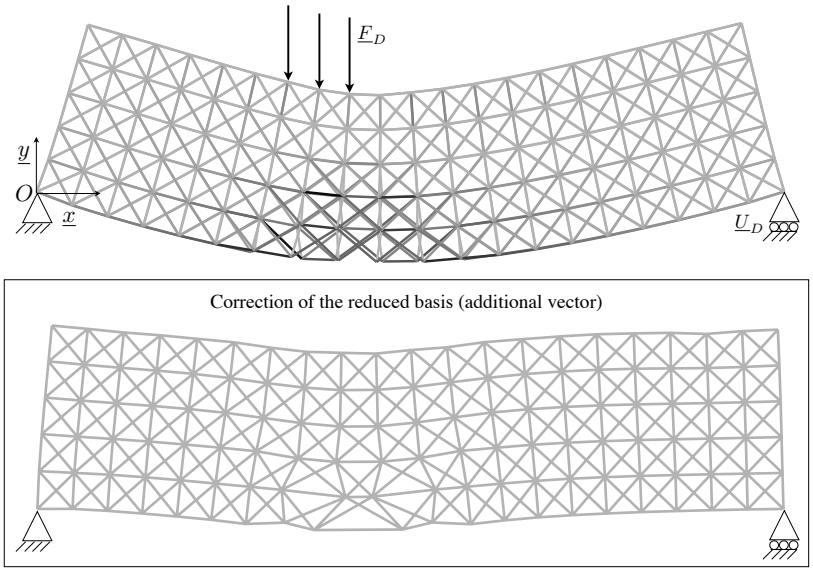

Figure 3: Superposed reference solution and solution obtained with the C-POD, for $\nu_{\text{New}} = 4.10^{-1}$ (top), additional reduced basis vectors obtained as a correction of the reduced basis (bottom).

- the relative norm of the residual of the initial problem (5), i.e.: $\dfrac{\|\underline{\mathbf{R}}\|_2}{\|\underline{\mathbf{F}}_{\text{Ext}}\|_2}$

Our aim is to control both these values during the solution process. More precisely, we will proceed to the next time increment only if the former is lower than $\nu_{\text{New,R}}$ and the latter lower than ν_{New}, with $\nu_{\text{New,R}} \ll \nu_{\text{New}}$. Typically, $\nu_{\text{New,R}}$ is set to 10^{-6} in our tests, while the value of ν_{New} (which, in our tests, ranges from 10^{-1} to 8.10^{-1}) will determine the accuracy of the successive Newton solutions, and is an important parameter of the strategy.

The following algorithm is applied. A sufficient number of Newton iterations are performed on the reduced problem, so that the norm of its residual is small enough (k_{Res} times smaller than the norm of the residual of the full problem, with $k_{\text{Res}} > 10$ a parameter of the method which can be tuned depending on the application and implementation). If the norm of the residual of the full problem is still higher than a desired value ν_{New}, the following linear prediction is enhanced by performing a few iterations of Conjugate Gradient on the linear prediction of the full problem (5). The resulting solution is eventually added to the reduced snapshot basis, as detailed in the next sections.

This process is repeated, until convergence of both the reduced problem ($\|\underline{\mathbf{R}}_R\|_2/\|\underline{\underline{\mathbf{C}}}^T\underline{\mathbf{F}}_{\text{Ext}}\|_2 \leq \nu_{\text{New,R}}$), and the full problem ($\|\underline{\mathbf{R}}\|_2/\|\underline{\mathbf{F}}_{\text{Ext}}\|_2 \leq \nu_{\text{New}}$).

3.2 Corrections on the linear predictions by an iterative solver

When required by the previously described algorithm, the enhanced linear prediction is performed by initializing a projected conjugate gradient and performing a few iterations on the linearised problem (6). The stopping criterion on the normalized residual is denoted by ν_{CG} and set to a high value, typically of the same order than ν_{New}. The reasons for choosing this iterative algorithm are the following:

- The construction of a Krylov basis requires only matrix/product operations. Therefore, no matrix assembly is required, which makes it particularly suitable for parallel

computing, and ensures the versatility of our method. In the case of serial computing, no matrix factorization is needed, which permits to save (i) memory, as we keep the sparsity of the operator (ii) time if only a coarse solution is needed, which is the case in the following developments.

- The projection framework is an ideal tool for the particular issue of enriching the reduced basis generated by the POD. Indeed, the initialization of the projected algorithm is the linear prediction of the reduced problem, equation (12). The complementary part of the solution is searched for is a space orthogonal to the reduced basis, which extends immediately the solution space and reduces the number of iterations required to reach ν_{CG} (left preconditioning). To summarize these ideas, each iteration of the CG is a correction of the solution to the reduced linear prediction (12) such that this correction is orthogonal to the initialization.

3.2.1 Projected Krylov solver

The classical projected (or augmented) conjugate gradient [29] is applied to the approximate solution of (6) (the linearised operator is assumed symmetric, positive and definite), which is recalled here:

$$\underline{\underline{\mathbf{K}}}_{\text{T}}\,\underline{\boldsymbol{\delta}\mathbf{U}} = \underline{\mathbf{F}} \tag{28}$$

Where the i superscripts indicating the iteration number have been dropped for the sake of clarity, and $\underline{\mathbf{F}} = -\underline{\mathbf{R}}$. The chosen augmentation space is the previously defined reduced snapshot space $\text{Im}(\underline{\underline{\mathbf{C}}})$.

The fundamental idea of the augmented Krylov algorithms is to split the search space into two supplementary spaces:

$$\mathbb{R}^{n_u} = \text{Im}(\underline{\underline{\mathbf{C}}}) \oplus \text{Im}(\underline{\underline{\mathbf{C}}})^{\perp_{\underline{\underline{\mathbf{K}}}_{\text{T}}}} \tag{29}$$

$\perp_{\underline{\underline{\mathbf{K}}}_{\text{T}}}$ designing the $\underline{\underline{\mathbf{K}}}_{\text{T}}-$orthogonality. The unknown solution vector is decomposed in this space under the form:

$$\underline{\boldsymbol{\delta}\mathbf{U}} = \underline{\boldsymbol{\delta}\mathbf{U}}_{\text{C}} + \underline{\boldsymbol{\delta}\mathbf{U}}_{\text{K}} \quad \text{where} \quad \left\{ \begin{array}{c} \underline{\boldsymbol{\delta}\mathbf{U}}_{\text{C}} = \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\delta\alpha}} \in \text{Im}(\underline{\underline{\mathbf{C}}}) \\ \underline{\boldsymbol{\delta}\mathbf{U}}_{\text{K}} \in \text{Im}(\underline{\underline{\mathbf{C}}})^{\perp_{\underline{\underline{\mathbf{K}}}_{\text{T}}}} = \text{Ker}(\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{K}}}_{\text{T}}) \end{array} \right. \tag{30}$$

The $\underline{\underline{\mathbf{K}}}_{\text{T}}-$orthogonality is ensured by introducing a projector $\underline{\underline{\mathbf{P}}}$ such that:

$$\left\{ \begin{array}{l} \underline{\boldsymbol{\delta}\mathbf{U}}_{\text{K}} = \underline{\underline{\mathbf{P}}}\,\underline{\boldsymbol{\delta}\mathbf{U}}_{\text{K}} \\ \underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{K}}}_{\text{T}} \underline{\underline{\mathbf{P}}} = \underline{\underline{\mathbf{0}}} \end{array} \right. \tag{31}$$

Hence, the decomposition of the correction of the displacement increment reads:

$$\underline{\boldsymbol{\delta}\mathbf{U}} = \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\delta\alpha}} + \underline{\underline{\mathbf{P}}}\,\underline{\boldsymbol{\delta}\mathbf{U}}_{\text{K}} \tag{32}$$

with the classical projector:

$$\underline{\underline{\mathbf{P}}} = \underline{\underline{\mathbf{I}}}_{\text{d}} - \underline{\underline{\mathbf{C}}}(\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{K}}}_{\text{T}} \underline{\underline{\mathbf{C}}})^{-1} \underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{K}}}_{\text{T}} \tag{33}$$

This separation of the search space into two subspaces $\text{Im}(\underline{\underline{\mathbf{C}}})$ and $\text{Im}(\underline{\underline{\mathbf{P}}})$ in direct sum leads to the following uncoupled equations:

$$(\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{K}}}_{\text{T}} \underline{\underline{\mathbf{C}}})\,\underline{\boldsymbol{\delta\alpha}} = \underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}} \quad \Rightarrow \quad \underline{\boldsymbol{\delta}\mathbf{U}}_{\text{C}} = \underline{\underline{\mathbf{C}}}(\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{K}}}_{\text{T}} \underline{\underline{\mathbf{C}}})^{-1} \underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}} \tag{34}$$

$$\left(\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\,\underline{\underline{\mathbf{P}}}\right)\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{K}} = \underline{\mathbf{F}} - \underline{\underline{\mathbf{K}}}_{\mathrm{T}}\,\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{C}} \tag{35}$$

Note that $\underline{\mathbf{R}}_{\mathrm{C}} = \underline{\mathbf{F}} - \underline{\underline{\mathbf{K}}}_{\mathrm{T}}\,\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{C}} = \underline{\underline{\mathbf{P}}}^{T}\,\underline{\mathbf{F}}$ and $\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\,\underline{\underline{\mathbf{P}}} = \underline{\underline{\mathbf{P}}}^{T}\,\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\,\underline{\underline{\mathbf{P}}}$. Equation (34) is a coarse initialization of the projected conjugate gradient. As stated previously, it is equivalent to the linear prediction of the reduced problem (equation (12)). Equation (35) is the linear prediction of the full problem projected on $\mathrm{Im}(\underline{\underline{\mathbf{C}}})^{\perp_{\underline{\underline{\mathbf{K}}}_{\mathrm{T}}}}$. This system is symmetric and can be solved by a preconditioned conjugate gradient. Hence we solve:

$$\widetilde{\underline{\underline{\mathbf{M}}}}^{-1}\left(\underline{\underline{\mathbf{P}}}^{T}\,\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\,\underline{\underline{\mathbf{P}}}\right)\underline{\boldsymbol{\delta}\mathbf{U}}_{K} = \widetilde{\underline{\underline{\mathbf{M}}}}^{-1}\,\underline{\underline{\mathbf{P}}}^{T}\,\underline{\mathbf{R}}_{\mathrm{C}} \tag{36}$$

where $\widetilde{\underline{\underline{\mathbf{M}}}}^{-1}$ is a left preconditioner (symmetric, definite and positive). In our test cases, $\widetilde{\underline{\underline{\mathbf{M}}}}$ is a diagonal matrix whose entries are the elements of the diagonal of $\underline{\underline{\mathbf{K}}}_{\mathrm{T}}$. Algorithm 1 describes the augmented preconditioned conjugate gradient.

Algorithm 1: Augmented preconditioned conjugate gradient

Compute $\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\underline{\mathbf{C}}}$, $(\underline{\underline{\mathbf{C}}}^{T}\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\underline{\mathbf{C}}})^{-1}$; $(\underline{\underline{\mathbf{P}}} = \underline{\underline{\mathbf{I}}}_{\mathrm{d}} - \underline{\underline{\mathbf{C}}}\left(\underline{\underline{\mathbf{C}}}^{T}\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\underline{\mathbf{C}}}\right)^{-1}\underline{\underline{\mathbf{C}}}^{T}\underline{\underline{\mathbf{K}}}_{\mathrm{T}})$;
$\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{C}} = \underline{\underline{\mathbf{C}}}(\underline{\underline{\mathbf{C}}}^{T}\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\underline{\mathbf{C}}})^{-1}\underline{\underline{\mathbf{C}}}^{T}\underline{\mathbf{F}}$ and $\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{K}0} = 0$;
$\underline{\mathbf{R}}_0 = \underline{\mathbf{R}} - \underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{C}}$;
$\underline{\mathbf{Z}}_0 = \underline{\underline{\mathbf{P}}}\widetilde{\underline{\underline{\mathbf{M}}}}^{-1}\underline{\mathbf{R}}_0$, $\underline{\mathbf{W}}_0 = \underline{\mathbf{Z}}_0$;
for $j = 1, \ldots, n$ **do**
  $\alpha_{j-1} = (\underline{\mathbf{R}}_{j-1}, \underline{\mathbf{W}}_{j-1})/(\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\mathbf{W}}_{j-1}, \underline{\mathbf{W}}_{j-1})$
  $\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{K}j} = \underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{K}j-1} + \alpha_{j-1}\underline{\mathbf{W}}_{j-1}$
  $\underline{\mathbf{R}}_j = \underline{\mathbf{R}}_{j-1} - \alpha_{j-1}\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\mathbf{W}}_{j-1}$
  $\underline{\mathbf{Z}}_j = \underline{\underline{\mathbf{P}}}\widetilde{\underline{\underline{\mathbf{M}}}}^{-1}\underline{\mathbf{R}}_{j+1}$
  $\beta_j = (\underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\mathbf{W}}_{j-1}, \underline{\mathbf{Z}}_j)/(\underline{\mathbf{W}}_{j-1}, \underline{\underline{\mathbf{K}}}_{\mathrm{T}}\underline{\mathbf{W}}_{j-1})$
  $\underline{\mathbf{W}}_j = \underline{\mathbf{Z}}_j - \beta_j\underline{\mathbf{W}}_{j-1}$
end

3.2.2 Adaptation of the reduced basis

The augmentation space, also used as reduced space for the projection of the linear prediction performed on the reduced problem, is chosen as:

$$\underline{\underline{\mathbf{C}}} = \left(\begin{array}{ccc} \underline{\underline{\mathbf{C}}}_{\mathrm{POD}} & \underline{\underline{\mathbf{C}}}_{\mathrm{NewT}} & \underline{\underline{\mathbf{C}}}_{\mathrm{CG}} \end{array}\right) \tag{37}$$

where:

- $\underline{\underline{\mathbf{C}}}_{\mathrm{POD}}$ is the initial reduced snapshot (classical POD). It is not changed, unless the size of the adaptive reduced basis becomes too large. This issue is discussed in the following.
- $\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}}$ is a set of converged solution vectors obtained during previous time steps, when at least one enhanced linear prediction has been performed (otherwise the solution obtained is a simple linear combination of the reduced basis vectors). At the end of such a time step t_n, the solution $\underline{\mathbf{U}}_{t=t_n}$ is orthonormalized with respect to $\underline{\underline{\mathbf{C}}}_{\mathrm{POD}}$ and $\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}}$ by a Gram-Schmidt procedure (using the classical Euclidian dot product). This orthonormalized vector is denoted by $\widetilde{\underline{\mathbf{U}}}_{t=t_n}$ and used to adapt $\underline{\underline{\mathbf{C}}}$:

$$\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}|t=t_{n+1}} = \left(\begin{array}{cc} \underline{\underline{\mathbf{C}}}_{\mathrm{NewT}|t=t_n} & \widetilde{\underline{\mathbf{U}}}_{t=t_n} \end{array}\right) \tag{38}$$

- $\underline{\underline{\mathbf{C}}}_{\mathrm{CG}}$ is a set of solution vectors obtained during the Newton process at the current time step. let $i+1$ be the current Newton iteration, the linear prediction requiring an enrichment by the augmented Conjugate Gradient. If $\underline{\boldsymbol{\delta}\bar{\mathbf{U}}}^{i+1}$ is the solution obtained at the end of this projected iterative linear solver, it is added to the reduced basis:

$$\underline{\underline{\mathbf{C}}}_{\mathrm{CG}}{}^{i+1} = \left(\begin{array}{cc} \underline{\underline{\mathbf{C}}}_{\mathrm{CG}}{}^{i} & \underline{\boldsymbol{\delta}\bar{\mathbf{U}}}^{i+1} - \underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{C}} \end{array} \right) \tag{39}$$

  where $\underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{C}}$ is the initialization of the projected iterative solver, given by solving the first row of system (34). The linearised reduced operator in equation (12) is recomputed, and the linear prediction (12) is performed, which yields the new solution vector $\underline{\mathbf{U}}^{i+1}$. Note that ,by construction, $(\underline{\boldsymbol{\delta}\bar{\mathbf{U}}}^{i+1} - \underline{\boldsymbol{\delta}\mathbf{U}}_{\mathrm{C}})$ is $\underline{\underline{\mathbf{K}}}_{\mathrm{T}}{}^{i}-$orthogonal to the previous reduced basis $\underline{\underline{\mathbf{C}}}^{i}$.

  When the convergence of the current time step is achieved, $\underline{\underline{\mathbf{C}}}_{\mathrm{CG}}$ is emptied.

The size of the reduced basis needs to be controlled. Indeed, the size of $\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}}$ increases during the simulation. Several techniques can be used when the number of additional vectors exceeds $n_{\mathrm{C,NewT,Max}}$:

- Keep only the last vectors of $\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}}$.
- Keep a few linear combination of the vectors of $\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}}$, these linear combination being extracted by a SVD computation.
- Consider the following operator as a Snapshot:

$$\underline{\underline{\widetilde{\mathbf{S}}}} = \left(\begin{array}{cc} \underline{\underline{\mathbf{C}}}_{\mathrm{POD}|t=t_n} & \underline{\underline{\widetilde{\mathbf{C}}}}_{\mathrm{NewT}_{|t=t_n}} \end{array} \right) \tag{40}$$

  where $\underline{\underline{\widetilde{\mathbf{C}}}}_{\mathrm{NewT}}$ is the set of all the previsous converged solutions, and choose as the new reduced basis the vectors associated to the highest singular values of $\underline{\underline{\widetilde{\mathbf{S}}}}$, in decreasing order. This new basis is split as follows:

$$\underline{\underline{\mathbf{C}}}_{|t=t_{n+1}} = \left(\begin{array}{cc} \underline{\underline{\mathbf{C}}}_{\mathrm{POD}|t=t_{n+1}} & \underline{\underline{\mathbf{C}}}_{\mathrm{NewT}|t=t_{n+1}} \end{array} \right) \tag{41}$$

  where, $\underline{\underline{\mathbf{C}}}_{\mathrm{POD}|t=t_{n+1}}$ and $\underline{\underline{\mathbf{C}}}_{\mathrm{POD}|t=t_n}$ have the same number of columns, and $\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}|t=t_{n+1}}$ have a number of columns lower than the one of $\underline{\underline{\mathbf{C}}}_{\mathrm{NewT}|t=t_n}$ (divided by two in the following examples). $\underline{\underline{\mathbf{C}}}_{\mathrm{POD}|t=t_{n+1}}$ usually contains the information of the initial reduced snapshot (strongly uncoupled and associated with singular values greater than 1).

The latter technique is the one used in our examples. It has the advantage to yield a basis whose vectors have a very weak energetic coupling, hence granting the method stability (we have observed that the two former ideas, despite the orthogonality, between the vectors of the reduced basis, easily lead to the stagnation of the Newton process when instabilities and multiple solutions have to be handled).

4 Behaviour of the adaptative strategy

We have introduced several parameters in the corrective model order reduction strategy (denoted by C-POD). The aim of this section is to provide an understanding of the sensitivity of the solver to a variation of these parameters, and when possible, to provide optimal or practical values. This study is not required to understand the method, and the reader might want to refer directly to the results given in section 4.3.

Algorithm 2: Corrective model order reduction strategy (the control of the size of the reduced basis is not taken described)

Load the reduced basis $\underline{\underline{\mathbf{C}}}$;

Initialise the solution of the reduced problem ;

$\underline{\boldsymbol{\alpha}}^0 = \underline{\mathbf{0}}$;

for $n = 1..n_t$ *(time loop)* **do**

Construct the linearized reduced operator

$\underline{\underline{\mathbf{K}}}_{\mathrm{T,R}}^{i+1} = \underline{\underline{\mathbf{C}}}^T \left. \frac{\partial \underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}}_{n-1} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}})}{\partial \underline{\boldsymbol{\alpha}}} \right|_{\underline{\boldsymbol{\alpha}} = \underline{\boldsymbol{\alpha}}^0}$

Compute the initial residuals

$\underline{\mathbf{R}}^0 = \underline{\mathbf{F}}_{\mathrm{Ext}}^n + \underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}}_{n-1} + \underline{\underline{\mathbf{C}}}\underline{\boldsymbol{\alpha}}^0)$

$\underline{\mathbf{R}}_R^0 = \underline{\underline{\mathbf{C}}}^T \underline{\mathbf{R}}^0$

Save the reduced basis

$\underline{\underline{\mathbf{C}}}^0 \leftarrow \underline{\underline{\mathbf{C}}}$ **while** $\frac{\|\underline{\mathbf{R}}_R\|_2}{\|\underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}}_{Ext}^n\|_2} \geq \nu_{New,R}$ *and* $\frac{\|\underline{\mathbf{R}}\|_2}{\|\underline{\mathbf{F}}_{Ext}^n\|_2} \geq \nu_{New}$ *(Newton loop)* **do**

if $\frac{\|\underline{\mathbf{R}}_R\|_2}{\|\underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}}_{Ext}^n\|_2} \leq k_{Res} \frac{\|\underline{\mathbf{R}}\|_2}{\|\underline{\mathbf{F}}_{Ext}^n\|_2}$ **then**

Assemble the global stiffness

$\underline{\underline{\mathbf{K}}}_{\mathrm{T}} = \left. \frac{\partial \underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}})}{\partial \underline{\mathbf{U}}} \right|_{\underline{\mathbf{U}} = \underline{\mathbf{U}}_{n-1} + \underline{\underline{\mathbf{C}}}^0\,\underline{\boldsymbol{\alpha}}^0}$

Save the reduced basis

$\underline{\underline{\mathbf{C}}}_{\mathrm{Tmp}} \leftarrow \underline{\underline{\mathbf{C}}}$

Compute the reduced basis correction by the projected conjugate gradient

$\widetilde{\underline{\boldsymbol{\delta}\mathbf{U}}} \approx -\underline{\underline{\mathbf{K}}}_{\mathrm{T}}^{-1} \underline{\mathbf{R}}^i \quad \text{where} \quad \widetilde{\underline{\boldsymbol{\delta}\mathbf{U}}} = \underline{\underline{\mathbf{C}}}\,\widetilde{\underline{\boldsymbol{\alpha}}} + \widetilde{\underline{\boldsymbol{\delta}\mathbf{U}}}_{\mathrm{K}}$

Update the reduced basis

$\underline{\underline{\mathbf{C}}} \leftarrow \left(\underline{\underline{\mathbf{C}}} \quad \frac{\widetilde{\underline{\boldsymbol{\delta}\mathbf{U}}}_{\mathrm{K}}}{\|\widetilde{\underline{\boldsymbol{\delta}\mathbf{U}}}_{\mathrm{K}}\|_2} \right)$

Project the previous solution in the new reduced space

$\underline{\boldsymbol{\alpha}}^i \leftarrow (\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{C}}})^{-1} \left(\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{C}}}_{\mathrm{Tmp}}\, \underline{\boldsymbol{\alpha}}^i \right)$

end

Solve the reduced linearized system

$\underline{\boldsymbol{\delta\alpha}}^{i+1} = -(\underline{\underline{\mathbf{K}}}_{\mathrm{R}}^i)^{-1} \underline{\mathbf{R}}_R^i$

Update the solution

$\underline{\boldsymbol{\alpha}}^{i+1} = \underline{\boldsymbol{\delta\alpha}}^{i+1} + \underline{\boldsymbol{\alpha}}^i$

Compute the residuals

$\underline{\mathbf{R}}^{i+1} = \underline{\mathbf{F}}_{\mathrm{Ext}}^n + \underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}}_{n-1} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}}^{i+1})$

$\underline{\mathbf{R}}_R^{i+1} = \underline{\underline{\mathbf{C}}}^T \underline{\mathbf{R}}^{i+1}$

$i \leftarrow i + 1$

end

Update the reduced basis if a correction has been performed ;

$\underline{\underline{\mathbf{C}}} \leftarrow \left(\underline{\underline{\mathbf{C}}}^0 \quad \widetilde{\underline{\mathbf{U}}} \right)$;

Initialise the solution of the reduced problem for the next time step ;

$\underline{\boldsymbol{\alpha}}^0 = \underline{\boldsymbol{\alpha}}^i$

end

4.1 Error criteria

In a first set of numerical experiments, we investigate the influence of the precision of the conjugate gradient iterations on the efficiency and accuracy of the method. We use the following set of fixed criteria: $k_{\text{Res}} = 10^3$, $n_{\text{C,NewT,Max}} = 3$. The relative norm of the residual of the full problem is successively set to (a) $\nu_{\text{New}} = 4.10^{-1}$ and (b) $\nu_{\text{New}} = 5.10^{-2}$. The results obtained when decreasing threshold values ν_{CG} are reported in table (1), Experiment A. The following tendencies can be observed:

- If $\nu_{\text{CG}} > \nu_{\text{New}}$ (i.e.: the conjugate gradients are solved finely compared to the actual maximum level of error required for the convergence of the adaptive strategy), the accuracy of the algorithm (i.e: the maximum value over time of the error in solution) increases with decreasing values of ν_{CG}.
- Unless $\nu_{\text{CG}} > \nu_{\text{New}}$ (i.e.: if the conjugate gradients are solved very loosely), the value of ν_{New} has no influence on the accuracy of the algorithm.
- The number of enhanced predictions performed to reach a given accuracy is relatively independent on the accuracy of the conjugate gradients resolutions, hence independent on ν_{CG}.
- The number of cumulated iterations of the conjugate gradients increases when the accuracy of the conjugate gradients resolutions increases (i.e.: ν_{CG} decreases), which results in the global computation costs to increase.

As we want to ensure a certain consistency in the accuracy of the algorithm with respect to the reduced model error criterion ν_{New}, we set $\nu_{\text{CG}} = \nu_{\text{New}}$. Note here that the corrective algorithm converges even if the convergence threshold on the conjugate gradients, ν_{CG}, is set to a loose value.

In a second numerical experiment, we try to explain the role of k_{Res} (parameter which sets the maximum value of the reduced residual norm to reach before allowing a correction to be performed) on the behaviour of the method. The following set of parameters is used, $\nu_{\text{New}} = 1.10^{-1}$, $\nu_{\text{CG}} = \nu_{\text{New}}$, $n_{\text{C,NewT,Max}} = 3$ and the results in terms of efficiency and accuracy with different values of k_{Res} are reported in table (1), Experiment B. The following remarks can be made:

- The accuracy of the solution is not influenced by k_{Res}.
- The global number of Newton iterations increases when k_{Res} increases, thus increasing the global computation time.

This behaviour can be explained by a qualitative analysis of the influence of k_{Res} on the solution strategy. During a time step where a correction of the reduced basis is required, using low values of k_{Res} results in the corrective linear predictions to be performed at the beginning of the Newton process. Conversely, when k_{Res} is set to a high value, a first nonlinear prediction is performed with the previous reduced model, and the correction is done at convergence of this process. As suggested by the results presented here, the former strategy is more efficient, as a relevant (i.e.: corrected) reduced model is constructed during the very first iterations of the Newton process.

However, we will see, in section 5, that this conclusion can be jeopardized by the actual implementation of the strategy.

EXPERIMENT A (a)					
ν_{CG}	**CG corrections**	**CG iterations**	**Newton iterations**	**CPU time**	**Solution error**
8.10^{-1}	6	82	257	20	1.75 %
3.10^{-1}	8	180	291	23	1.93 %
1.10^{-1}	8	224	290	23	1.93 %
3.10^{-2}	8	256	291	23	1.93 %
1.10^{-2}	8	294	291	23	1.93 %
1.10^{-3}	8	318	291	23	1.93 %
EXPERIMENT A (b)					
ν_{CG}	**CG corrections**	**CG iterations**	**Newton iterations**	**CPU time**	**Solution error**
8.10^{-1}	42	545	485	38	0.449 %
3.10^{-1}	41	858	496	41	0.382 %
1.10^{-1}	41	1204	498	44	0.380 %
3.10^{-2}	41	1372	498	44	0.380 %
1.10^{-2}	41	1315	498	45	0.380 %
1.10^{-3}	41	1531	498	45	0.380 %
EXPERIMENT B					
k_{Res}	**CG corrections**	**CG iterations**	**Newton iterations**	**CPU time**	**Solution error**
10^0	31	827	384	36	0.894 %
10^2	32	895	410	36	0.887 %
10^4	32	896	453	39	0.887 %
10^6	32	896	500	42	0.887 %

Table 1: Influence of the parameters of the C-POD on the accuracy and efficiency of the corrective model order reduction

4.2 Number of additional vectors in the reduced basis

A new set of numerical experiments are performed in order to study the relevancy of the adaptive reduced model created during the successive time steps. The following set of parameters fixed parameters is chosen: $\nu_{\mathrm{New}} = 10^{-1}$, $\nu_{\mathrm{CG}} = \nu_{\mathrm{New}}$, $k_{\mathrm{Res}} = 10^3$ and the value of the maximum number of solutions added to the reduced model, $n_{\mathrm{C,NewT,Max}}$ is incremented. The results obtained are given in table (2). The following general tendencies appear:

- The number of correction steps required to obtain a given accuracy drastically decreases as the size of the reduced model (number of additional reduced basis vectors) increases.
- The global CPU time rapidly decreases when using an increasing, but small, number of additional reduced basis vectors. Though, when too high a value is given for $n_{\mathrm{C,NewT,Max}}$, the costs of the assembly steps becomes dominant, and the numerical efficiency decreases.

$n_{\text{C,NewT,Max}}$	**CG corrections**	**CG iterations**	**Newton iterations**	**CPU time**	**Solution error**
0	74	2696	560	60	0.780 %
1	45	1367	435	40	0.773 %
3	32	896	429	38	0.887 %
10	14	370	409	36	1.02 %
20	14	360	418	41	1.02 %

Table 2: Influence of $n_{\text{C,NewT,Max}}$ on the efficiency of the corrective algorithm model order reduction

4.3 Results

We can finally evaluate the suitability of the adaptative algoritm to the purpose for which it has been designed, namely the automatic correction of a reduced model.

The previously studied parameters are fixed to the following values: $\nu_{\text{CG}} = \nu_{\text{New}}$, $k_{\text{Res}} = 10^3$ and $n_{\text{C,NewT,Max}} = 3$. Several simulations are performed for different values of the control parameter ν_{New} (interpreted as the error criterion on the reduced model). The results are reported in table (3). For comparison, the results obtained with the basic POD strategy and for a classical Newton algorithms on the complete system are also given.

	CG corrections	**CG iterations**	**Newton iterations**	**CPU time**	**Solution error**
POD			266	17	8.27 %
$\nu_{\text{New}} = 8.10^{-1}$	4	46	274	21	4.11 %
$\nu_{\text{New}} = 3.10^{-1}$	12	262	334	26	1.61 %
$\nu_{\text{New}} = 1.10^{-1}$	32	896	429	38	0.887 %
$\nu_{\text{New}} = 3.10^{-2}$	54	1601	556	52	0.263 %
$\nu_{\text{New}} = 1.10^{-2}$	73	2107	705	67	0.157%
$\nu_{\text{New}} = 1.10^{-3}$	132	4351	1539	144	0.00424 %
Full Newton			824	434	

Table 3: Efficiency and accuracy of the corrective model order reduction

The first global observation is that the numbers of corrections and cumulated Newton iterations increases as ν_{New} decreases. As a consequence, the CPU time increases with decreasing values of the control parameter. More importantly, the accuracy of the solution is controlled by ν_{New}. Figure (4, left) shows that the relation between the solution error and the threshold value is almost linear (log-log scaling). Similarly, figure (4, right) demonstrates that we obtain a linear relation between the accuracy of the solution and the CPU time required for the computation. Hence, the adaptive algorithm acts as a continuous corrective link between the basic POD strategy on the one hand, and a fine solution scheme for the full nonlinear problem on the other hand.

The results detailed below help to understand how the successive corrections are performed during the simulation. Figure (5, left) presents the evolution of the solution error with time, for the different values of ν_{New} used in this study. A "zoom" in these results is plotted in figure (5, right), where we focus on the curve obtained by the application of the basic POD, and the one obtained by the adaptive algorithm, with $\nu_{\text{New}} = 3.10^{-1}$.

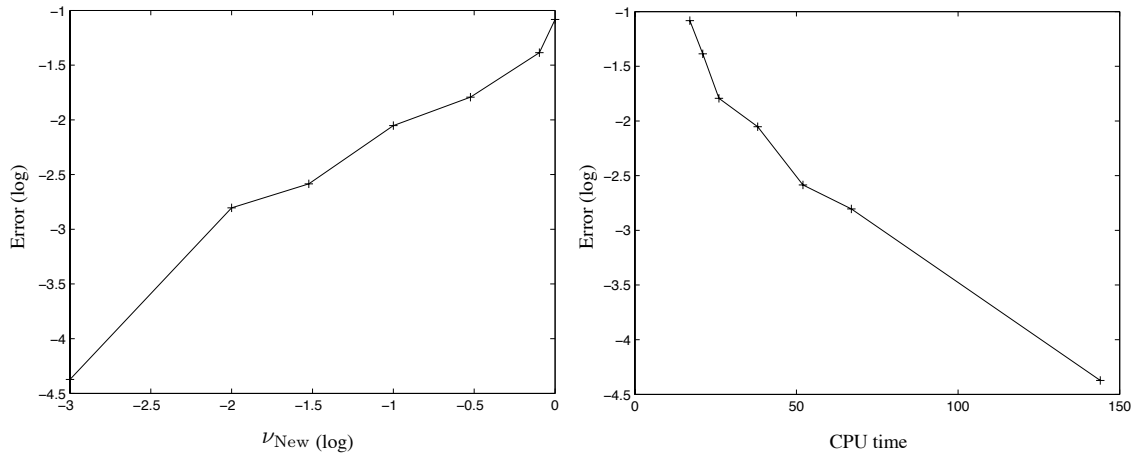

Figure 4: Solution errors obtained with decreasing values of ν_{CG}, for $\nu_{\text{New}} = 4.10^{-1}$ (left) and $\nu_{\text{CG}} = 2.10^{-1}$ (right).

The evolution of the maximum damage in the structure (linear because of the local arc-length control) is also reported. Each vertical bar point out the time steps where at least one correction of the reduced basis has been performed. One can see that a correction is necessary at the very beginning of the simulation, to take into account the specificity of the loading. The corrected reduced model constructed after the time step is sufficiently relevant to allow the simulation of the initiation of the damage in the structure, at the precision required by the predefined value of ν_{New} (a single additional correction is performed at Time step 10). However, when the maximum damage exceeds 1 (Time step 20), the rate of corrective steps increases as the fast topological changes must be taken into account in the reduced model.

For illustration, and in the particular case detailed in the previous paragraph, the solution obtained at the end of the simulation is superimposed to the reference solution in figure (3). The correction of the reduced basis is also plotted and shows an important deformation in the region where the damage localises. Elsewhere, the value of this vector is almost null, which means that, far away from the"crack", the displacement field is correctly represented by the initial reduced basis. In other terms, the correction is local, which obviously leads to the conclusion that multigrid or domain decomposition solvers could help to reduce the cost of the corrections. This issue will be discussed later on.

4.4 Discussion

We have shown in the previous section that the error-controlled POD-Krylov algorithm is a bridge between a full Newton/Krylov solver and the basic POD method. This algorithm can obviously be used to correct the reduced model of various engineering problems. Yet, in the case of mechanical models involving internal variables, the computation time required to assemble the reduced stiffness operator (evaluation of internal forces) is far too high to yield a computationally efficient strategy. Indeed, although the number of unknown is considerably reduced, the evaluation of the internal variables and internal forces are done in each element of the structure. In [23], it is proposes to use the hyperreduction to reduce the cost of these operations. In the next section, we propose to apply the adaptive algorithm to the hyperreduction of damage models.

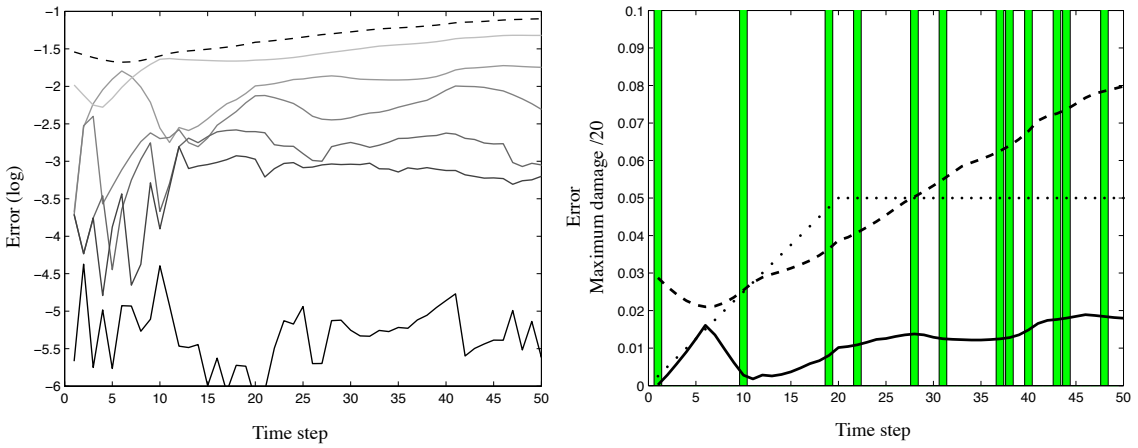

Figure 5: Error in solution for different values of the residual error of the initial problem ν_{New} (left). Darker curves correspond to lower values of ν_{New}. The dashed curve is obtained by applying the basic POD and increases significantly with time. When using the corrective POD (C-POD), the error decreases in a monotonic manner with decreasing values of ν_{New}. On the right, the dotted line is the maximum damage over the structure. The dashed line is still obtained by applying the basic POD, is reasonably low in the initiation phase (no significant topological changes), and increases when damage propagates. The bar corresponds to the time steps being corrected when applying the C-POD ($\nu_{\text{New}} = 3.10^{-1}$). The frequency of correction increases when strong topological changes occur. The dark line is the resulting error in solution, and is kept to a low level.

5 Krylov corrections for the hyperreduction of damage models

5.1 Principle of the hyperreduction strategy

The principle of the hyperreduction, which we will denote by H-POD, was first described in [22], and applied to mechanical problems involving internal variables in [23]. It drastically reduces the cost of evaluating the internal forces required to compute the successive linearised operators and residuals in the Newton algorithm for a problem reduced by the POD. The main idea is to consider only a few of the equations of the initial problem to find the coefficients associated to each of the reduced basis vectors. If the controlled equations and the reduced basis vectors are correctly chosen, the norm of the residual due to the equations which are not controlled will be small. Note that we do not aim at questioning the validity of this strategy, but to use it as an efficient solver for damage models, and test the efficiency of the adaptive reduced basis in such a framework

5.2 Petrov-Galerkin formulation of the nonlinear system of equations

The starting point of the strategy is the full system (5), in which the approximation of the displacement as a linear combination of reduced basis vectors has been introduced:

$$\underline{\mathbf{F}}_{\text{Int}}\left(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}}\right) + \underline{\mathbf{F}}_{\text{Ext}} = \underline{\mathbf{0}} \tag{42}$$

This system of equations is obviously overconstrained. A solution to this system is obtained in the classical model order reduction by using the Galerkin orthogonality condition with respect to the reduced basis. But one can choose any orthogonality conditions leading

to a well-posed problem. A "natural" choice can be to satisfy only a very few number of equations n_h of (42) (called controlled equations), with $n_h \geq n_{\mathrm{C}}$

$$\underline{\underline{\mathbf{E}}}^T \left(\underline{\mathbf{F}}_{\mathrm{Int}} \left(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}} \right) + \underline{\mathbf{F}}_{\mathrm{Ext}} \right) = \underline{\mathbf{0}} \tag{43}$$

$\underline{\underline{\mathbf{E}}}$ is a Boolean operator of n_u (number of nodal unknown) rows and n_h columns. For each column of $\underline{\underline{\mathbf{E}}}$, the only non-zero value correspond to the i^{th} controlled equation, for $i \in [\![1\ n_h]\!]$, and is set to 1.

In the H-POD, problem (43) is required to satisfy the Galerkin orthogonality condition with respect to the reduced snapshot, which finally yields the following system:

$$\underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{E}}}\,\underline{\underline{\mathbf{E}}}^T \left(\underline{\mathbf{F}}_{\mathrm{Int}} \left(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}} \right) \right) + \underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{E}}}\underline{\underline{\mathbf{E}}}^T\, \underline{\mathbf{F}}_{\mathrm{Ext}} = \underline{\mathbf{0}} \tag{44}$$

5.2.1 Nonlinear solver

The Newton Procedure described in section 2.2.2 can be applied to the hyperreduced problem (44). At iteration i of the Newton algorithm the following linear prediction is performed,

$$\underline{\underline{\mathbf{K}}}^i_{\mathrm{T,HR}}\, \underline{\boldsymbol{\delta\alpha}}^{i+1} = -\underline{\mathbf{R}}^i_{\mathrm{HR}} \tag{45}$$

where $\underline{\mathbf{R}}^i_{\mathrm{HR}} = \underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{E}}}\,\underline{\underline{\mathbf{E}}}^T (\underline{\mathbf{F}}_{\mathrm{Ext}} + \underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}}^i))$. The correction stage reads:

$$\begin{aligned} \underline{\mathbf{R}}^{i+1}_{\mathrm{HR}} &= \underline{\underline{\mathbf{C}}}^T \underline{\underline{\mathbf{E}}}\,\underline{\underline{\mathbf{E}}}^T \underline{\mathbf{F}}_{\mathrm{Ext}} + \underline{\underline{\mathbf{C}}}^T\, \underline{\underline{\mathbf{E}}}\,\underline{\underline{\mathbf{E}}}^T \underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}}^{i+1}) \\ \underline{\underline{\mathbf{K}}}^{i+1}_{\mathrm{T,HR}} &= \underline{\underline{\mathbf{C}}}^T\, \left. \frac{\partial\, \underline{\underline{\mathbf{E}}}\,\underline{\underline{\mathbf{E}}}^T \underline{\mathbf{F}}_{\mathrm{Int}}(\underline{\mathbf{U}}_{|t_n} + \underline{\underline{\mathbf{C}}}\,\underline{\boldsymbol{\alpha}})}{\partial \underline{\boldsymbol{\alpha}}} \right|_{\underline{\boldsymbol{\alpha}} = \underline{\boldsymbol{\alpha}}^{i+1}} \end{aligned} \tag{46}$$

The resulting linearized operator $\underline{\underline{\mathbf{K}}}^i_{\mathrm{T,HR}}$ is square, fully populated, non-symmetric and of very small size $n_h \times n_h$. It is solved by a direct solver (LU factorization).

Note that $\underline{\underline{\mathbf{E}}}\,\underline{\underline{\mathbf{E}}}^T$ is not explicitly computed. It simply extracts the internal and external forces of the controlled equations used to calculate the residuals and linearised operators. The integration of the internal and external forces needs only be done on a reduced integration domain Ω_h, made of all the elements connected to a controlled node (see figure (9)).

The same developments can be done for the derivation of the arc-length procedure, which is used in the examples of this section.

5.2.2 Reduced integration domain

The choice of the controlled equations is the keypoint of the hyperreduction strategy. In [22, 23], it is shown that in order all the reduced basis vectors to be "observable", one should choose to control the equations associated to the nodes whose connected elements are subjected to the largest strain energy from the reduced basis vectors. Hence, it is recommended to perform a loop on the nodes and compute the average of the strain energy due to each individual reduced basis vector in the connected elements. Then, for each reduced basis vector, one chould choose a given, and small, number of nodes with the maximum average strain energy observed by the connected elements. Finally, the controlled equations should correspond to all the degrees of freedom of these particular nodes, and the connected elements define the reduced integration domain.

5.3 Modified Error-controlled algorithm

When solving the hyperreduced system by our adaptive algorithm (the resulting strategy will be called corrective hyperreduction, and denoted by CH-POD), certain operations become cost-inefficient:

- The calculation of the residual $\underline{\mathbf{R}}$.
- The assembly of the global stiffness, required to enrich the reduced basis.
- The conjugate gradient itself if not correctly preconditioned.

These issues are discussed next.

5.3.1 Calculation of the damage variables and residual of the initial problem

In the H-POD method, the calculation of the internal variables needs only be done on the reduced integration domain Ω_h. In [23], it is proposed to interpolate these internal variables over $\Omega \backslash \Omega_h$ thanks to pre-calculated snapshot functions. In our case, this particular feature is not of interest, for:

- The residual of the global system needs to be controlled.
- The residual of the global system needs to be calculated when required to perform an enhanced prediction.

Therefore, the internal variables are computed systematically when the relative norm of the residual of the reduced problem reaches a treshold value. The norm of the full residual can thus be computed, and if its value is too high, an enhanced linear prediction (e.e.: a correction of the reduced basis by the Krylov solver) is performed. Note that this calculation is very expensive compared to the cost of the Newton iterations performed on the hyperreduced system, and must be limited to a minimum number.

Let us recall that in the former algorithm given in section 3.1, an enhanced linear prediction could be performed if $\frac{\|\underline{\mathbf{R}}_R\|_2}{\|\underline{\underline{\mathbf{C}}}^T \underline{\mathbf{F}}_{\text{Ext}}\|_2} < k_{\text{Res}} \frac{\|\underline{\mathbf{R}}\|_2}{\|\underline{\mathbf{F}}_{\text{Ext}}\|_2}$. It requires the computation of the full residual at any iteration of the Newton solver, which is, in the hyperreduction strategy, computationally inefficient.

Hence, for this particular application, this comparison of the two norms is bypassed, and an enrichment of the reduced basis will not be made unless the relative norm of the reduced problem is lower than $\nu_{\text{New,R}}$ (convergence criterion on the reduced nonlinear problem, set to a very low value), thus limiting the number of Newton iterations at which the internal damage variables must be calculated in $\Omega \backslash \Omega_h$. An another modification permits to suppress a large quantity of the numerical costs. The norm of the full residual is controlled at the first time step, and, if no correction is needed, no control is done during the following ${n_{C,T}}^{th} timestep$ ($n_{C,T} = 2$ in the following test cases). The same procedure is repeated after each controlled time step.

5.3.2 Global stiffnesses operators

Obviously, the assembly of the global stiffness is the weak point of this error-controlled hyperreduction strategy. Yet, as the conjugate gradient algorithm is only used to enrich the reduced basis, one does not need to use the current stiffness matrix. The "naive" technique to suppress the costs associated with this operation would then be to use a

stiffness matrix assembled during the computation of the snapshot. Though, as the damage in the structure increases, the relevance of the computed additional reduced basis vectors can be questioned, leading to a slower convergence of the adaptive strategy, which will be illustrated later on.

We propose an intermediate technique called the"patch assembly" (PA). Before an enhanced prediction stage, the residual of the full system is known. We can reasonably assume that the part of the domain Ω_{PA} of highest local residuals correspond to high levels of structural changes. Therefore, we get an approximation of the current stiffness operator by removing the previous elementary stiffnesses corresponding to the elements in Ω_{PA} from the last stiffness approximation, and replacing them by their updated values.

This procedure is performed in three steps:

1. Identify the nodes corresponding to the highest values of the residual.
2. For each of this nodes, identify the connected elements, whose union defines Ω_{PA}.
3. In the existing approximation fo the stiffness operator $\widehat{\underline{\underline{\mathbf{K}}}}^i_{\text{T}}$, remove the elementary matrices corresponding to the elements in Ω_{PA}:

$$\widetilde{\underline{\underline{\mathbf{K}}}}_{\text{T}} = \widehat{\underline{\underline{\mathbf{K}}}}^i_{\text{T}} - \sum_{E/\Omega_E \in \Omega_{\text{PA}}} \underline{\underline{\mathbf{A}}}^T_E \, \underline{\underline{\mathbf{K}}}_{\text{T},E}(Z_{\text{Old}}) \, \underline{\underline{\mathbf{A}}}_E \tag{47}$$

   where Z_{Old} represent the state variables and material parameters used to assemble $\underline{\underline{\mathbf{K}}}^i_{\text{T}}$, $(\underline{\underline{\mathbf{A}}}_E)_{E\in[\![1\, n_E]\!]}$ are Boolean assembly operators, which extract the rows of the global stiffness corresponding to the degrees of freedom of Element E, and $\underline{\underline{\mathbf{K}}}_{\text{T},E}(Z_{\text{Old}})$ the elementary stiffness operator.

4. Calculate the elementary matrices corresponding to the elements in Ω_{PA} and assemble them in $\widetilde{\underline{\underline{\mathbf{K}}}}_{\text{T}}$:

$$\widehat{\underline{\underline{\mathbf{K}}}}^{i+1}_{\text{T}} = \widetilde{\underline{\underline{\mathbf{K}}}}_{\text{T}} + \sum_{E/\Omega_E \in \Omega_{\text{PA}}} \underline{\underline{\mathbf{A}}}^T_E \, \underline{\underline{\mathbf{K}}}_{\text{T},E}(Z^i) \, \underline{\underline{\mathbf{A}}}_E \tag{48}$$

The performance of this simple technique will be demonstrated later on.

5.3.3 Preconditionning

As the size of the initial problem increases, the preconditioning of the conjugate gradient used to enrich the reduced basis becomes an issue. In order to be consistent with the projection framework that we have introduced, we make use of the Selective Reuse of Krylov Subspace (SRKS), proposed in [30]. Briefly, the preconditioner itself is unchanged (diagonal matrix whose entries are the inverse of the diagonal elements of the stiffness operator), but additional reduced basis vectors are added to the augmentation space:

$$\widetilde{\underline{\underline{\mathbf{C}}}} = \left(\begin{array}{cc} \underline{\underline{\mathbf{C}}} & \underline{\underline{\mathbf{C}}}_{\text{SRKS}} \end{array} \right) \tag{49}$$

The columns of $\underline{\underline{\mathbf{C}}}_{\text{SRKS}}$ are made of eigenvectors of the successive linearised operators projected on $\text{Im}(\widetilde{\underline{\underline{\mathbf{C}}}}^{\perp_{\underline{\underline{\mathbf{K}}}_{\text{T}}}})$ during the computation of the snapshot. These eigenvectors are the converged Ritz vectors obtained when solving the linear systems by a conjugate gradient (eigenvectors of the complete linearised operator associated with the highest eigenvalues).

This feature permits to rapidly access the super-convergence of the conjugate gradient.

5.3.4 Hyperreduced arc-length strategy

The search for the largest increment of displacement in equation (23) is only done on the reduced integration domain. A sufficient number of point corresponding to the highest damage rate is set as controlled nodes at the end of each time step. Therefore, one can expect that the control equations is reasonably satisfied.

5.4 Efficiency for parametric simulations

5.4.1 Presentation of the test cases

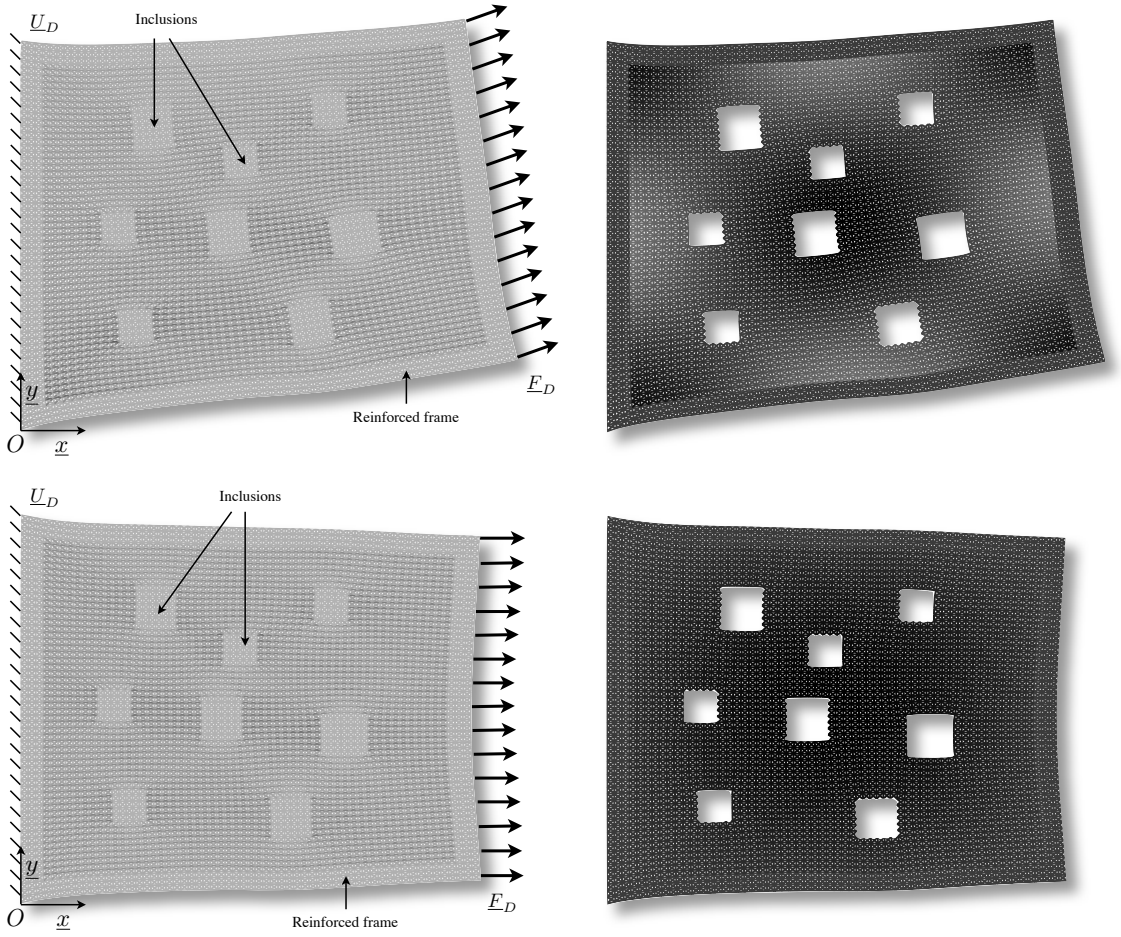

Figure 6: Reference problem and associated solutions obtained for different material parameters and direction of the prescribed load (left), where darker colours correspond to higher value of damage, and Young's modulus cosinusoidal distribution in these two cases (right), where darker colours correspond to higher values of the stiffness parameter.

The test case studied here is a square damageable lattice (the constitutive modelling is the one given in section 2.3.1) represented in figures (6). The lattice structure is made of 14,520 bars connected by 3721 nodes. Hence the number of degree of freedom involved in this problem is $3721 \times 2 = 7442$. A non-zero homogeneous traction force such that $\underline{F}_{\mathrm{D}} = F_x\,\underline{x} + F_y\,\underline{y}$ is applied on the right edge of the structure, while zero-displacement

are prescribed on its right edge. We define the angle:

$$\phi = \arctan\left(\frac{F_y}{F_x}\right) \tag{50}$$

The load is applied through a reinforced frame occupying domain Ω_{F}. The Young's Modulus of the bars belonging to the frame is equal to one, while their damage parameter α is set to 0.1. Hard inclusions, occupying domain Ω_{I} are located in domain $\Omega \backslash \Omega_{\mathrm{F}}$ (see their position and shape on figure (6)). The Young's Modulus of the bars such that $\Omega_b \in \Omega_{\mathrm{I}}$ is set to 10 while $\alpha = 0.01$. Due to these low values of α in $\Omega_{\mathrm{F}} \cup \Omega_{\mathrm{I}}$, the damage will localize in the complementary domain $\Omega_{\mathrm{C}} = \Omega \backslash (\Omega_{\mathrm{F}} \cup \Omega_{\mathrm{I}})$, where α is set to 1. The young modulus of the bars in this domain is defined by the following law:

$$E|_{\Omega_b \in \Omega_{\mathrm{C}}} = E_{\mathrm{C}}\left(1 + \epsilon_{\mathrm{C}}\left(\sin\left(\omega_{\mathrm{C}}\left((X_b - \bar{X}) + (Y_b - \bar{Y})\right)\right) + \sin\left(\omega_{\mathrm{C}}\left((X_b - \bar{X}) - (Y_b - \bar{Y})\right)\right)\right)\right) \tag{51}$$

where the position of the barycentre G_b of a bar occupying domain Ω_{b} is

$$\underline{OG_b} = X_b\,\underline{x} + Y_b\,\underline{y} \tag{52}$$

and the position of barycentre G of the structure is defined by:

$$\underline{OG} = \bar{X}\,\underline{x} + \bar{Y}\,\underline{y} \tag{53}$$

The purpose of our study is to obtain the dissipated energy in the structure, for a value of the load that is close to the instability point (hence the arc-length control is not required), as a function of the two following parameters:

- the angle ϕ, which controls the direction of the force.
- the angular frequency ω_{C}, which controls the spatial distribution of the Young's modulus of the bars belonging to Ω_{C}.

The construction of such functions is very classical is stochastic analysis. It is often associated with the Monte-Carlo method to determine the random distribution of a quantity of interest (in our case the dissipated energy) as a function of the stochastic input parameters (here ϕ and ω_{C}). If an analytical derivation of this function is not possible, the classical procedure consists in performing a few deterministic simulations, for different values of the input parameters, and interpolate the value of the quantity of interest between these points.

Let us recall that the dissipated energy of the chosen damage model is given by:

$$E_{\mathrm{Dissi}} = \int_\Omega \left(\int_{t=0}^{T} \frac{N}{S_b}\dot{\epsilon}\,dt - \frac{1}{2}E(1 - d|_{t=T})\epsilon^2 \right) d\Omega = \int_\Omega \int_{t=0}^{T} \frac{Y}{S_b}\dot{d}\,dt\,d\Omega \tag{54}$$

The following values are chosen for the constant parameters: $E_{\mathrm{C}} = 1$, $\epsilon_{\mathrm{C}} = 0.2$. Particular distributions of the Young's modulus are represented on figure (6, right).

We will consider that the input parameters can vary within the following range:

$$\begin{cases} \omega_{\mathrm{C}} \in [\omega_{\mathrm{C},1}, \omega_{\mathrm{C},2}] = [0.05, 0.1] \\ \phi \in [\phi_1, \phi_2] = [0, 5] \end{cases} \tag{55}$$

25 simulations are performed to estimate the response surface (see figure 7):

$$\begin{cases} \omega_{\mathrm{C}} \in \{0.05, 0.625, 0.75, 0.875, 0.1\} \\ \phi \in \{0, 1.25, 2.5, 3.75, 5\} \end{cases} \tag{56}$$

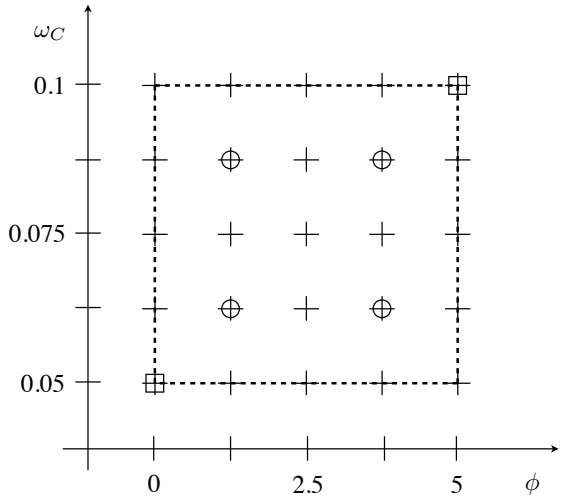

Figure 7: Estimation of the response surface. The surface relates the dissipated energy to two structural parameters ϕ and ω_C over a rectangular domain. The values of this function at 25 nodes (crosses) are computed by performing nonlinear simulations. The other values are interpolated using a piecewise linear approximation.

The response surface (linear interpolation) obtained when using a direct Newton solver over 10 time increments is given in figure (8, left). The displacement solutions, damage maps and Young's modulus distributions corresponding to the two limit points boxed in figure (7) are represented in figure (6). Our goal is to limit the costs of these nearby simulations by using POD-based reduced order model for these 25 successive solutions, and in particular to test the efficiency of the CH-POD.

5.4.2 Parameters of the hyperreduced model

In order to build a snapshot, we solve successively four problems circled in figure 7. Each of these simulations is performed in 10 time increments. The final snapshot, made of 40 vectors, is reduced to 8 reduced basis vectors by a singular value decomposition (the ratio of the first singular value of $\underline{\underline{\mathbf{S}}}$ over the eighth is $1.9\,10^{-8}$, which shows that the snapshot is correctly represented in $\underline{\underline{\mathbf{C}}}$).

We construct the reduced integration domain by choosing a list of controlled nodes (2 controlled balance equations per node).

1. The domain is divided into 100 square subdomains. For each of these subdomains, an arbitrarily chosen node is set as controlled point.

2. 5 nodes are arbitrarily (first ten in the global nodal numbering) chosen for each of the prescribed boundary conditions (left and right edge of the lattice).

3. For each vector of the reduced basis, 5 nodes whose connected element have the highest strain energy are set as controlled nodes.

4. 20 additional nodes with the highest damage rate are added to the list, hence allowing to precisely integrate the constitutive law in this particular elements.

An example of RID obtained during the CH-POD process is given in figure (9).

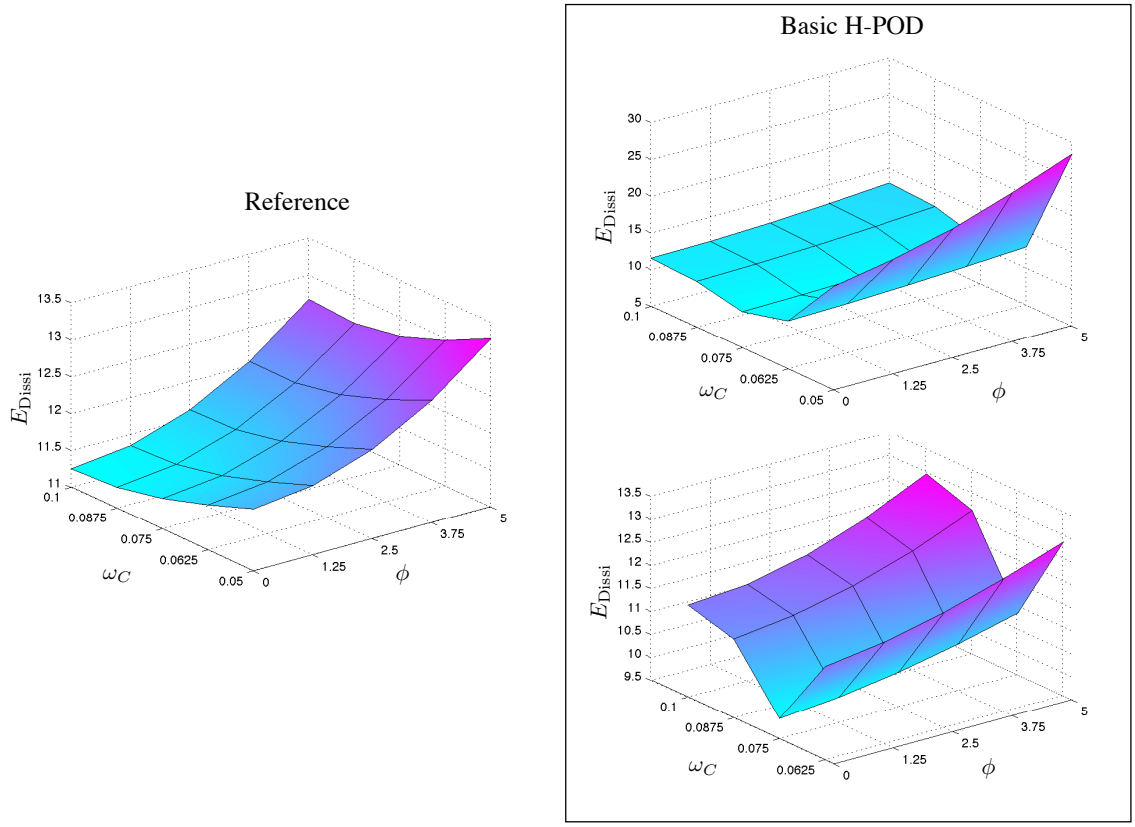

Figure 8: Response surface obtained in the reference case, and when using the H-POD. The error in the dissipated obtained when not correcting the reduced model is very high, due to the large distance between the solution which is searched for and the *a priori* computed snapshot.

5.4.3 Results

The following solvers are to estimate the response surface:

- Newton algorithm on the complete system, the linear prediction being solved by a direct solver, and $\nu_{\text{New}} = 10^{-6}$.
- "basic" hyperreduction (H-POD) with the snapshot and controlled points defined previously.
- corrective hyperreduction (CH-POD), where $\nu_{\text{New,R}} = 10^{-6}$ (threshold value on the relative norm of the residual of the reduced problem), $n_{\text{C,NewT,Max}} = 3$ (number of maximum additional vectors added in the reduced basis during the Newton iterations) and ν_{New} successively set to (a) 10^{-1} ,(b) 3.10^{-2} and (c) 10^{-2} (error criterion on the reduced model).

For each of these solvers we report the total CPU time required to construct the surface, and the maximum error in the dissipated energy over the 25 simulations (see table (4)). This latter error is defined as follows:

$$\text{error} = \frac{\text{E}_{\text{Dissi}|_{\text{Reference}}} - \text{E}_{\text{Dissi}|_{\text{Current experiment}}}}{\text{E}_{\text{Dissi}|_{\text{Reference}}}} \tag{57}$$

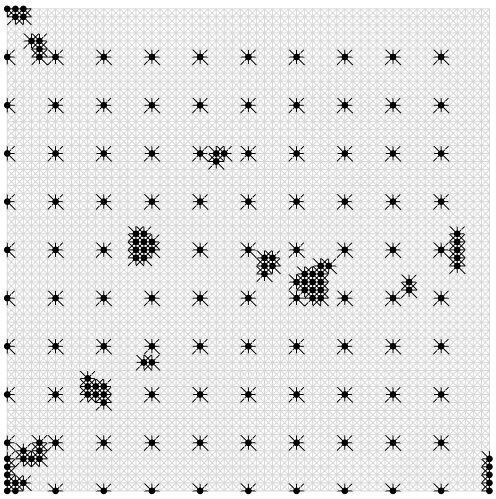

Figure 9: Reduced integration domain (dark bars) obtained at the end the first simulation performed to estimate the response surface. During the Newton process, the integration of the internal forces need be done only on this domain, which drastically reduces the computation costs.

	Maximum error in dissipation	**CPU time (s)**
Full Newton (reference)		3227
H-POD	113 %	618
CH-POD (a)	5.06 %	730
CH-POD (b)	1.98 %	890
CH-POD (c)	0.82 %	1075

Table 4: Error and CPU time obtained when estimating the response surface with the H-POD and CH-POD compared to a reference response surface obtained by a classical Newton algorithm on the complete system of equations.

The results given by the hyperreduction are not satisfying. The maximum error obtained exceeds 100%, which can be explained by observing the results presented in figure (8). One can see that the values of dissipated energy obtained for values of ω_{C} which have not not been taken into account in the snapshot are wrongly predicted, and especially for $\omega_{\mathrm{C}} = 0.05$. In this latter case, the structure is a lot weaker (greater quantity of bars with Young's modulus lower than E_{C}, as appears in figure (6)) than in the cases computed to create the snapshot. As a consequence, the dissipated energy is overestimated.

The application of the CH-POD permits to retrieve a correctly estimated response surface for small additional costs. For instance, in experiment (a) ($\nu_{\mathrm{New}} = 10^{-1}$), the maximum error drops to 5% for 18% of additional CPU time compared to the basic reduced order simulation. The response surface obtained is given in figure (10).

Decreasing values of the reduced model error criterion (ν_{New}) results in a decreasing value of the error, while slightly increasing the additional CPU costs (see figure (10) and table (4)). In experiment (c), the difference between the response surface obtained and the reference one can hardly be notice, and the evolution of the quantity of interest with respect to the input parameters is correctly taken into account. Yet, this simulation is less

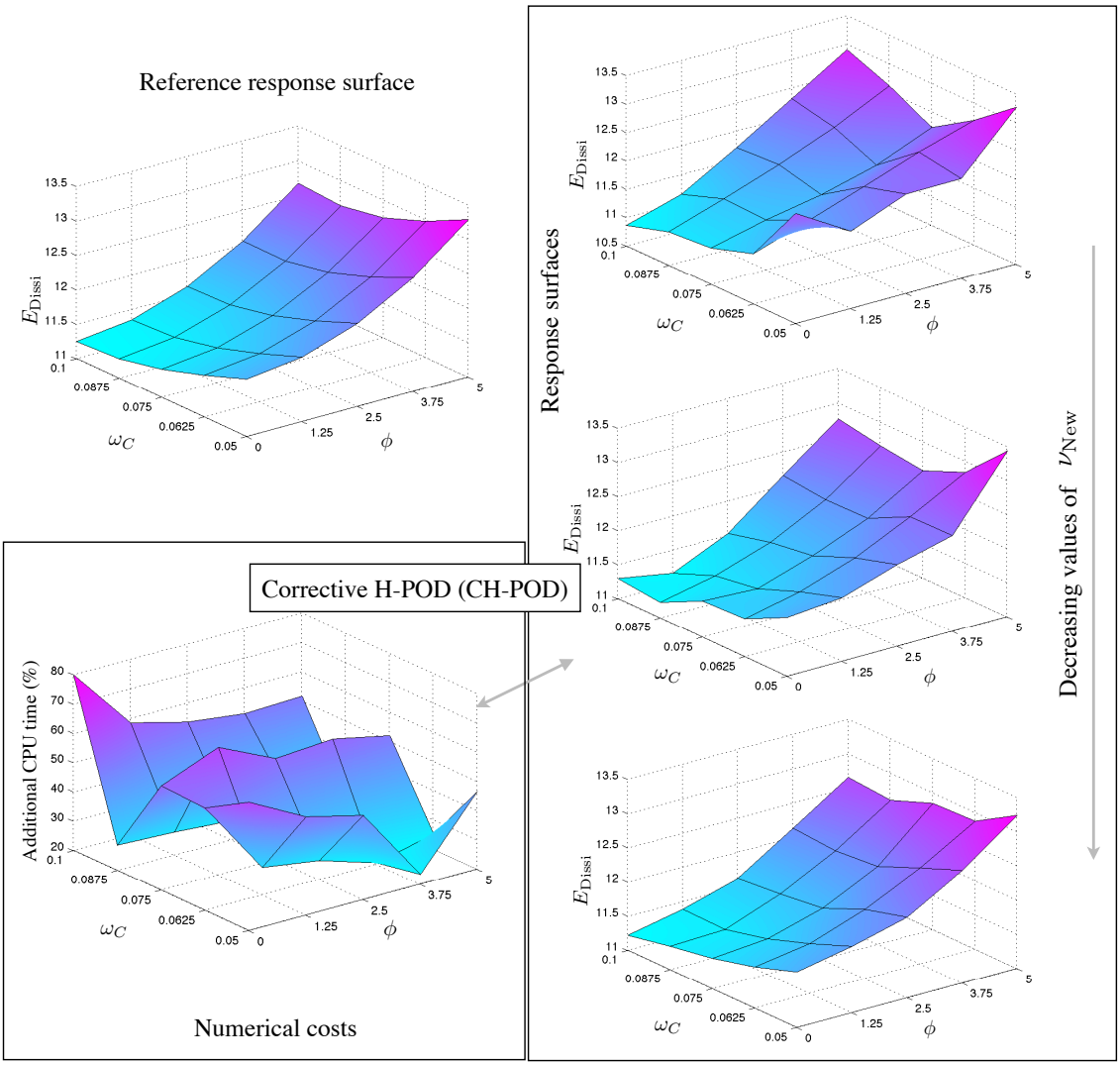

Figure 10: Response surface obtained in the reference case, and when using the CH-POD for various values of the reduced model error criterion, and CPU time for each of the 25 simulations (additional CPU times with respect to the ones obtained by applying hyper-reduction). The error in the dissipated energy monotonically drops when the criterion decreases. The additional CPU time is concentrated in the simulations whose solution is far away from the snapshot.

than two times as costly as the one made using the H-POD.

The CPU time, corresponding to each of the 25 simulations, in case (b), is also reported in figure (10). The additional costs are due to the solution which is searched for to be far away, in some sense, from the snapshot. As explained previously, the Young's modulus distribution leads to the snapshot being irrelevant, which explains why the costs of the corrections in the CH-POD increases. Conversely, the variation of the angle, for $\phi > 0$ is correctly taken into account in the uncorrected model order reduction, and the CH-POD generates very few extra costs in these cases. In the particular case where $\phi = 0$ the solution is not correctly represented by a linear combination of the initial reduced basis vectors. Indeed, a strictly positive angle in the direction of the load leads to the localization of damage in one of the corner of the square structure (see figure 6), while the structure is much stiffer for $\phi = 0$. The CH-POD automatically corrects the resulting reduced model.

5.4.4 Brief validation of the Patch-assembly

In order the structural changes to be significant in the structure, Experiment A is performed with a higher value of the norm of the final loading (same value of the arc-length, but greater number of time increments). The evolution of the damage leads to the apparition of cracks in the vicinity of two of the hard inclusions.

We use three different approximation of the global stiffness operator in the ECH-POD strategy. The threshold value ν_{New} is set to 10^{-1}. In the first case, the stiffness is updated at the beginning of each time increment, which, as described previously, leads to an inefficiency due to the costs of the global assembly. In the second case, the initial stiffness is used to perform the enhanced linear prediction. The CPU time slightly decreases but the number of conjugate gradient calls required to obtain a relevant reduced basis increases significantly.

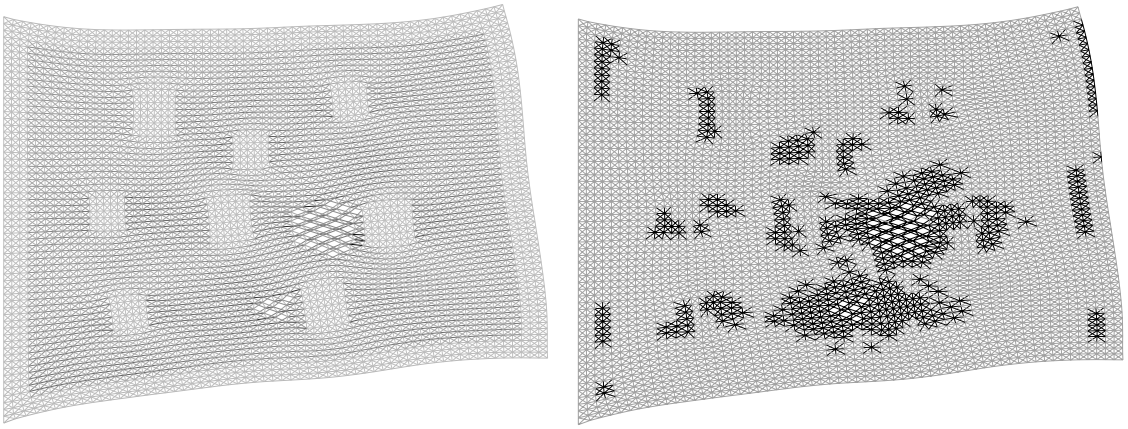

Figure 11: Solution obtained using the arc-length procedure after the limit point (left). On the right, the black bars are connected to the nodes corresponding to the highest values of the residual of the complete nonlinear system. They are mainly located in the zone of high topological changes.

Finally, the Patch-assembly technique is used to yield successive approximation of the global stiffness matrix. The number of elements in Ω_{PA} is 2440 (in comparison with the total number of bars: 14250), which gives a decrease of factor 6 of the costs of the global assembly. Figure (11) shows Domain Ω_{PA}, which is indeed mainly located in the zones

of high structural changes (the two cracked regions). The resulting CPU time is further decreased compared to the second experiment, as shown in table 5. Note that the number of conjugate gradient calls does not increase compared to the one obtain when updating systematically the stiffness operator, which shows that the approximation of this matrix is satisfying.

	CG corrections	**CG iterations**	**Newton iterations**	**CPU time**
Updated stiffness	23	424	414	421
Initial stiffness	33	500	412	407
Patch-assembly	22	440	447	335

Table 5: Efficiency of the Patch-assembly

5.5 Discussion

5.5.1 For further efficiency

A few important issues can severely limit the efficiency of the CH-POD derived in this section.

- The efficiency of the H-POD itself can be put into question for softening behaviour. Indeed, its numerical complexity is directly dependent on the size of the reduced integration domain. We have observed that choosing this domain in order to ensure the stability of the algorithm was not a simple issue, and chosen to use a large number of controlled nodes. By a better understanding of this behaviour, one should be able to reduce the size of the reduced integration domain to a minimum, and therefore diminish the costs.

- The reduced integration domain is updated at each correction of the CH-POD, for a new reduced basis vector needs to be "observed". This operation is costly as it requires to evaluate quantities on the whole domain. This operation needs to be optimized.

- The price of the correction is still high. As a result, if the solution which is searched for is far away from the initial snapshot, the method becomes computationally more expensive than a nonlinear direct solver on the complete system. It is thus limited, up-to-now, to the correction of reduced models for the resolution of nearby problems. In a forthcoming paper, we will propose a multilevel approach in order to perform the correction only in the zones of the structure undergoing the largest topological changes. This strategy should lead to a significant gain in terms of numerical efficiency, and especially in the case of localized nonlinearities, as suggested by the results provided by the patch-assembly procedure.

5.5.2 Extension to finite element problems

The proposed method has only been validated on damageable lattices so far. Slight differences in the efficiency results might appear when dealing with 2D or 3D softening problems discretised by the finite element method:

- the conditioning of lattice problems is usually better than the one obtained when using finite element discretisation. We have been able to use a very simple preconditioner for the iterative corrections of the reduced basis: deflation orthogonally to a few Ritz vectors by the SRKS method, and diagonal preconditioner on the resulting deflated problem. Obviously, an enhanced preconditioner should be preferred when dealing with finite element problems, in order to keep the costs of the corrections low.

- the connectivity in lattice structures can be very high. As a result, the reduced integration domain associated to each of the controlled nodes in the hyperreduction method is quite large, especially compared to the ones obtained when using structured finite element meshes. Therefore, the cost reduction associated to this particular step of the CH-POD should increase when decreasing the connectivity of nodes.

One of the major issues to be addressed when dealing with FE softening problems is the question of localization. It is well known that nonlocal damage models should be used to avoid spurious dependency of the solution to the mesh. The efficiency of the hyperreduction technique, which focuses on local patches in the structure, has not yet been investigated in this framework. Its extension does not seem trivial.

5.5.3 Suitability to parallel computing

The extension of our strategy to parallel computing is quite straightforward. The H-POD can be implemented in parallel as follows: the domain is divided into subdomains, assigned to separate processors. Solving the small problems resulting from the linearisation of the reduced problem is done systematically on every processor. The evaluation of the internal forces is a local operation performed on the reduced integration domain, distributed on the processors. It only requires an assembly operation, of small numerical costs as involving only degrees of freedom associated to the controlled nodes.

The corrections performed in the CH-POD are very easy to parallelize as they are based on a conjugate gradient algorithm. The projection operation itself has been widely extended to parallel computing (see [9] for example). In addition, using the condensation method, as proposed in the Schur-complement-based domain decomposition methods would provide better preconditioner than the one used in this work.

6 Conclusion

We developed an efficient corrective tool for the adaptive model order reduction of highly nonlinear mechanical problems. The novelty of the approach is that it completely integrates the corrections inside the projection framework classically used in model order reduction. More precisely, the corrections are performed using an iterative conjugate gradient, which is enabled by the strong link between, on the first hand, POD-based model order reduction and, on the other hand, projected Krylov algorithms. The resulting method establishes a bridge model order reduction and classical Newton/Krylov solvers, ensuring its versatility.

The robustness of the strategy has been demonstrated by correcting the reduced model associated with a very complex case of damage propagation. In terms of raw CPU time, our algorithm had been coupled to the hyperreduction method in a straightforward way,

making it computationally efficient in the case of mechanical problems involving internal variables.

Yet, the corrections are still expensive compared to solving the initial reduced model. As explained briefly in the last section of the paper, the corrections need not to be done everywhere in the structure, especially when localization phenomena are involved. This can lead to prohibitive costs if the solution that is searched for is distant from the snapshot. Therefore, a possible enhancement to increase the efficiency of the adaptive model order reduction is to use a multilevel algorithm to perform the corrections.

Acknowledgments

Stéphane Bordas and Pierre Kerfriden would like to thank the support of the Royal Academy of Engineering and of the Leverhulme Trust for his Senior Research Fellowship entitled "Towards the next generation surgical simulators" as well as the support of EPSRC under grants EP/G069352/1 Advanced discretisation strategies for "atomistic" nano CMOS simulation and EP/G042705/1 Increased Reliability for Industrially Relevant Automatic Crack Growth Simulation with the eXtended Finite Element Method.

References

[1] J. Lemaître and J.-L. Chaboche. *Mechanics of Solid Materials*. Cambridge University Press, 1990.

[2] G. Lubineau, P. Ladevèze, and D. Marsal. Towards a bridge between the micro- and mesomechanics of delamination for laminated composites. *Composites Science and Technology*, 66(6):698–712, 2007.

[3] AD Jefferson and T. Bennett. A model for cementitious composite materials based on micro-mechanical solutions and damage-contact theory. *Computers & Structures*, doi:10.1016/j.compstruc.2008.09.006, 2009.

[4] G. Franceschini, D. Bigoni, P. Regitnig, and GA Holzapfel. Brain tissue deforms similarly to filled elastomers and follows consolidation theory. *Journal of the Mechanics and Physics of Solids*, 54(12):2592–2620, 2006.

[5] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal of Engineering Science*, 46:131–150, 1999.

[6] S. Bordas and B. Moran. Enriched finite elements and level sets for damage tolerance assessment of complex structures. *Engineering Fracture Mechanics*, 73(9):1176–1201, 2006.

[7] J. Fish, K. Shek, M. Pandheeradi, and M. S. Shephard. Computational plasticity for composite structures based on mathematical homogenization: Theory and practice. *Computer Methods in Applied Mechanics and Engineering*, 148:53–73, 1997.

[8] F. Feyel and J.-L. Chaboche. FE^2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. *Computer Methods in Applied Mechanics and Engineering*, 183:309–330, 2000.

[9] P. Gosselet and C. Rey. Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering*, 13:515–572, 2006.

[10] P. Ladevèze, J.C. Passieux, and D. Néron. The latin multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(21):1287–1296, 2009.

[11] K. Karhunen. Über lineare methoden in der wahrscheinlichkeitsrechnung. *Annales Academiæ Scientiarum Fennicæ Series A1, Mathematical Physics 37*, 37:1–79, 1947.

[12] M. Loeve. *Probability Theory*. 3rd edition Van Nostrand, Princeton, 1963.

[13] P. Ladevèze and A. Nouy. On a multiscale computational strategy with time and space homogenization for structural mechanics. *Computer Methods in Applied Mechanics and Engineering*, 192:3061–3087, 2003.

[14] J. Yvonnet and Q. C. He. The reduced model multiscale method (r3m) for the non-linear homogenization of hyperelastic media at finite strains. *Journal of Computational Physics*, 223(1):341–368, 2007.

[15] J. Pebrel, C. Rey, and P. Gosselet. A nonlinear dual domain decomposition method: application to structural problems with damage. *International Journal for Multiscale Computational Engineering*, 6(3), 2008.

[16] P. Kerfriden, O. Allix, and P. Gosselet. A three-scale domain decomposition method for the 3D analysis of debonding in laminates. *Computational Mechanics*, 44(3):343–362, 2009.

[17] F. Chinesta, A. Ammar, and E. Cueto. Proper generalized decomposition of multiscale models. *International Journal for Numerical Methods in Engineering*, DOI:10.1002/nme.2794, 2009.

[18] D. Ryckelynck and D. M. Benziane. Multi-level a priori hyper-reduction of mechanical models involving internal variables. *Computer Methods in Applied Mechanics and Engineering*, 199(17-20):1134–1142, 2010.

[19] Y. C. Liang, H. P. Lee, S. P. Lim, W. Z. Lin, K. H. Lee, and C. G. Wu. Proper orthogonal decomposition and its applications - part I: theory. *Journal of Sound and Vibration*, 252(3):527–544, 2002.

[20] L. Sirovich. Turbulence and the dynamics of coherent structures. part I: coherent structures. *Quarterly of Applied Mathematics*, 45:561–571, 1987.

[21] 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.

[22] D. Ryckelynck. A priori hyperreduction method: an adaptive approach. *Journal of Computational Physics*, 202(1):346–366, 2005.

[23] D. Ryckelynck. Hyper-reduction of mechanical models involving internal variables. *International Journal for Numerical Methods in Engineering*, 77(1):75 – 89, 2008.

[24] R. Markovinovic and J.D. Jansen. Accelerating iterative solution methods using reduced-order models as solution predictors. *International Journal for numerical Methods in Engineering*, 68:525–54, 2006.

[25] F. Risler and C. Rey. Iterative accelerating algorithms with krylov subspaces for the solution to large-scale nonlinear problems. *Numerical algorithms*, 23:1–30, 2000.

[26] O. Allix, P. Kerfriden, and P. Gosselet. On the control of the load increments for a proper description of multiple delamination in a domain decomposition framework. *International Journal for Numerical Methods in Engineering*, DOI:10.1002/nme.2884, 2010.

[27] J.C.J. Schellenkens and R. De Borst. On the numerical integration of interface elements. *International Journal for Numerical Methods in Engineering*, 36(1):43–66, 1993.

[28] O. Allix and A. Corigliano. Modeling and simulation of crack propagation in mixed-modes interlaminar fracture specimens. *International Journal of Fracture*, 77:11–140, 1996.

[29] Z. Dostál. Conjugate gradient method with preconditioning by projector. *International Journal of Computer Mathematics*, 23(3):315–323, 1988.

[30] P. Gosselet. *Méthodes de décomposition de domaine et méthodes d'accélération pour les problèmes multichamps en mécanique non-linéaire.* PhD thesis, Université Paris 6, 2003.