A full PGD/HBM-based algorithm for a low-dimensional NNM continuation with modal enrichment

Louis Meyrand, Emmanuelle Sarrouy, Bruno Cochelin, Guillaume Ricciardi

A full PGD/HBM-based algorithm for a low-dimensional NNM continuation with modal enrichment

L. Meyrand[1], **E. Sarrouy**[1], **B. Cochelin**[1], **G. Ricciardi**[2]
[1] Aix Marseille Univ, CNRS, Centrale Marseille, LMA, Marseille, France
e-mail: **meyrand@lma.cnrs-mrs.fr**

[2] DEN/DTN/STRI/LHC, CEA Cadarache, France

Abstract

This work proposes a full algorithm to compute Nonlinear Normal Modes (NNMs) using a Proper Generalized Decomposition (PGD) approach combined to a continuation method. Because periodic solutions are sought, a harmonic balance method (HBM) is implemented to process the time-dependent functions. The interest of this study is first to couple this approach to a continuation method with a progressive enrichment of the PGD modes family and second to improve some of the PGD steps to get a robust algorithm. In particular, the initializations of solvers are chosen consistently with the available physical data through the use of LNMs shapes. The reduced basis is kept as small as possible since it is computed with on the fly enrichment, possibly from a unique mode. No particular assumptions on the non linearity have been made and hence a wide class of mechanical problems can be processed. The resulting method is applied on several beam models with localized nonlinearities which allows for an accurate analysis of its performance.

1 Introduction

The computing power currently achieved allows to take into account nonlinear effects in physical models and allows to deal with systems with a very large number of degrees of freedom. As systems became more and more realistic and sophisticated, model reduction techniques had to be developed in order to save digital resources while obtaining a physically satisfactory result.

Nonlinear Normal Modes (NMMs) are objects that can be exploited as part of modal reduction methods, similar as with Linear Normal Modes (LNMs). Their properties are studied to estimate and interpret more precisely the dynamics of mechanical systems [1, 2]. In this paper, only the Rosenberg's definition of NNMs [3] will be considered, *i.e.* a family of periodic solutions of the underlying free and conservative mechanical system. The reader can refer to the work of Shaw and Pierre [4, 5] or Kerschen and Renson [6, 7, 8] for more information about NNMs.

From a numerical point of view, the algorithms used to compute the NNMs of high-dimensional structures do not have yet reached maturity [9, 8] and the nonlinear effects causes convergence issues. NNMs require large sets of variables to be properly described which make them hard to compute in short times and hard to reuse even if some work still take advantage of their computations [10]. This remark serves as a starting point for the work presented in this article.

The reducted model developed in this paper falls within the framework of Proper Generalized Decomposition (PGD) methods. This class of reduction techniques is based on the separation of the unknowns and a fixed point algorithm which allows to deal with smaller systems in order to reduce the computation times. PGD was already used by Grolet and Thouverez [11] to compute free and forced responses of nonlinear systems. The reader can refers to Chinesta and Nouy [12, 13, 14, 15] to learn more about the possibilities offered by this approach.

In this work, a full algorithm is proposed in order to compute NNMs using a combination of a PGD approach and classical continuation schemes. A quick reminder of the NNM framework investigated here is given in Sec. 2. Then the PGD method is detailed in Sec. 3, including the use of a harmonic balance method (HBM) to deal with temporal aspects of the problem. The contributions of this work are highlighted in this section: first the PGD approach is coupled with a continuation method and a progressive enrichment of the PGD modes family is made and second some of the PGD steps are improved to get a robust algorithm. The PGD/HBM-based continuation scheme is described in Sec. 4. The last section, Sec. 5, is dedicated to illustrative cases to show some possibilities introduced by this algorithm.

2 Mechanical framework and Nonlinear Normal Modes (NNMs)

The equations of dynamics describing the behavior of a set of general coordinates can be obtained after a spatial discretization process using for example the Finite Elements method. It usually takes the following form:

$$\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{C}\dot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} + \boldsymbol{f}_{\text{nl}}(\boldsymbol{u}, \dot{\boldsymbol{u}}) = \boldsymbol{f}(t) \tag{1}$$

Eq. (1) is a set of N second order nonlinear differential equations, the vector of the unknowns $\boldsymbol{u}$ – or vector of the degrees of freedom (dof) – is a vector of $\mathbb{R}^N$. $\boldsymbol{M}$, $\boldsymbol{C}$ and $\boldsymbol{K}$ are respectively the linear mass, damping and stiffness matrices, with $\{\boldsymbol{M}, \boldsymbol{C}, \boldsymbol{K}\} \in \mathcal{M}_N(\mathbb{R})^3$. $\boldsymbol{f}(t)$ is the vector of excitation forces and lies in $\mathbb{R}^N$ too. Finally $\boldsymbol{f}_{\text{nl}}(\boldsymbol{u}, \dot{\boldsymbol{u}})$ is the vector of non linear forces and belongs to $\mathbb{R}^N$. The general system described by Eq. (1) represents a very wide class of mechanical problems. Indeed, the $\boldsymbol{f}_{\text{nl}}$ term can be any nonlinear expression (*e.g.* cosine function, polynomial terms like $u_i u_j^3$, non smooth function, etc.).

This paper focuses on the particular study of NNMs, which will be considered as the periodic solutions of Eq. (1) in the particular case when no forcing neither damping is considered, or as an extension of the definition of LNMs which takes into account the nonlinear vector $\boldsymbol{f}_{\text{nl}}$. In this framework, introduced by Rosenberg, a NNM is a set of limit cycles – or periodic solutions – of the following system:

$$\boldsymbol{R}(\boldsymbol{u}(t)) = \boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} + \boldsymbol{f}_{\text{nl}}(\boldsymbol{u}, \dot{\boldsymbol{u}}) = \boldsymbol{0} \tag{2}$$

Unlike LNMs, NNMs do not decouple the equations, and there exists a frequency-amplitude dependency. However, they take into account non linear effects and can be useful to investigate modal interaction between widely spaced modes or modal bifurcations [7, 8]. Numerically, building a NNM branch requires a large number of descriptors and consequent computation times [8]. This observation motivates the developments described in this work.

3 PGD and HBM combination to describe periodic solutions

3.1 Harmonic Balance Method

As NNMs are periodic solutions, a HBM framework is used. It does not require a time integration scheme as in shooting methods [16, 9]. Detailed explanations about this method and its advantages are given in works [17, 18, 19, 20]. Dealing with algebraic equations in the frequency domain has the disadvantages first to introduce a truncation order dependence and second to become costly as the number of dof increases and more harmonics are involved in the underlying physics of the system. A combination with a PGD reduction technique will permit to compute smaller systems without losing HBM advantages.

Some notations used later are here introduced. We are looking for a T-periodic signal $\boldsymbol{u}(t)$, with $T = 2\pi/\omega$:

$$\boldsymbol{u}(t) = \frac{\boldsymbol{a}_0}{\sqrt{2}} + \sum_{k=1}^{H} \left(\boldsymbol{a}_k \cos(k\omega t) + \boldsymbol{b}_k \sin(k\omega t)\right) \tag{3}$$

where $\boldsymbol{a}_k$ and $\boldsymbol{b}_k$ are respectively the cosine and sine coefficients of the Fourier series. A compact way to write $\boldsymbol{u}(t)$ is then:

$$\boldsymbol{u}(t) = \boldsymbol{U}_H \boldsymbol{h}_H(t,\omega), \text{ where } \begin{cases} \boldsymbol{U}_H = [\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{b}_1, \dots] \\ \boldsymbol{h}_H = [1/\sqrt{2}, \cos(\omega t), \sin(\omega t), \dots]^{\mathrm{T}} \end{cases} \tag{4}$$

This formalism first shows computing $\boldsymbol{u}(t)$ is equivalent to compute both $\boldsymbol{U}_H$ and ω, second allows to link the Fourier coefficient of $\boldsymbol{u}(t)$ to those of its temporal derivative $\dot{\boldsymbol{u}}(t)$ via a matrix $\boldsymbol{D}$:

$$\boldsymbol{D} = \mathrm{diag}\left(0, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \dots, H \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\right) \tag{5}$$

As HBM is a Galerkine method, this decomposition is injected in Eq. (2) and the equations are projected on each element of the Fourier basis $\boldsymbol{h}_H$ using the inner product:

$$< f, g >_T = \frac{2}{T}\int_0^T f(t)g(t)dt \tag{6}$$

This leads to a set of $N(2H+1)$ equations with $N(2H+1)+1$ unknowns, where N is the number of dofs. As there is no external forcing in Eq. (2), ω is also an unknown, and the final problem can be written:

$$\boldsymbol{H}(\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{b}_1, \dots, \omega) = \boldsymbol{0} \tag{7}$$

Whether a HBM is applied on the forced equation Eq. (1), ω is taken equal to the forcing frequency, and the system described by Eq. (7) is square. In the case of NNMs computation, an extra equation usually called *phase condition* [21] is needed. Indeed, no "initial time" exists for a given periodic solution $\boldsymbol{u}(t)$ so the phase condition allows to define the Fourier coefficients uniquely. Here are given some possibilities of equations: to lock the amplitude or the velocity of a given dof at $t = 0$ in Eq. (3) [21], or to nullify one of the Fourier coefficient components [22]. Considering a conservative system, please note that nullifying one sine coefficient implies that all sine coefficients are null, hence only (H+1) $\boldsymbol{a}_k$ coefficients must be computed.

Considering Eq. (7) again, it is possible to analytically establish the expression of the projection of the linear part $\boldsymbol{H}_1(\omega)\,\boldsymbol{u}_H$ of Eq. (2), where $\boldsymbol{u}_H = \{\boldsymbol{a}_0^{\mathrm{T}}, \dots, \boldsymbol{a}_k^{\mathrm{T}}, \boldsymbol{b}_k^{\mathrm{T}}, \dots\}^{\mathrm{T}}$. Hence, Eq. (7) is rewritten $\boldsymbol{H}_1(\omega)\,\boldsymbol{u}_H + \boldsymbol{H}_{\mathrm{nl}}(\boldsymbol{u}_H, \omega) = \boldsymbol{0}$, with:

$$\begin{cases} \boldsymbol{H}_1(\omega) = \mathrm{diag}(\boldsymbol{\Lambda}_0, \boldsymbol{\Lambda}_1, \dots, \boldsymbol{\Lambda}_H) \\ \boldsymbol{\Lambda}_0 = \boldsymbol{K}, \ \boldsymbol{\Lambda}_k = \mathrm{diag}(\boldsymbol{K} - (k\omega)^2\boldsymbol{M}, \boldsymbol{K} - (k\omega)^2\boldsymbol{M}) \end{cases} \tag{8}$$

The term $\boldsymbol{H}_{\mathrm{nl}}(\boldsymbol{u}_H, \omega)$ corresponds to the projections of nonlinear forces $\boldsymbol{f}_{\mathrm{nl}}(\boldsymbol{u}, \dot{\boldsymbol{u}})$ onto the Fourier basis. Evaluating this nonlinear contribution is time consuming as analytical expressions can scarcely be derived. The most commun approach to deal with this term is to use the Alternating Frequency/Time (AFT) method introduced by Cameron and Griffin [23]. Basically, $\boldsymbol{u}(t_k)$ and $\dot{\boldsymbol{u}}(t_k)$ are evaluated for specific t_k values in $[0, T]$ using an Inverse Fast Fourier Transform (IFFT) based on $\boldsymbol{a}_k$ and $\boldsymbol{b}_k$ coefficients. Then, $\boldsymbol{f}_{\mathrm{nl}}(\boldsymbol{u}(t_k), \dot{\boldsymbol{u}}(t_k))$ is computed for each time t_k; finally, a Fast Fourier Transform (FFT) is used to evaluate the projections, that is the coefficients of the Fourier series of $\boldsymbol{f}_{\mathrm{nl}}(\boldsymbol{u}, \dot{\boldsymbol{u}})$.

3.2 PGD combined to HBM

For the purpose of this work, the PGD process is divided in three main steps, which are separating variables (space and time here), obtaining as many subproblems as there are variables (time and space problems here), and run an alternated directions fixed point loop which solves each subproblem with the other variables fixed, for a given loop iteration. Here both subproblems are coupled with HBM features introduced in Sec. 3.1 to

produce a global PGD/HBM algorithm. Many variants of the PGD method exist and we chose to describe the one usually called *optimized PGD* (oPGD) [11, 15] in the following theoretical developments. A lighter adaptation of the oPGD from a computational point of view and called *progressive PGD* (pPGD) is detailed in Sec. 4.1.3.

Given a positive integer $m << N$, we look for a space-time separated solution $\boldsymbol{u}(t)$ of Eq. (2) such as:

$$\boldsymbol{u}(t) \approx \sum_{j=1}^{m} \boldsymbol{p}_j q_j(t) \Leftrightarrow \boldsymbol{u}(t) \approx \boldsymbol{P}\boldsymbol{q}(t) \text{ with } \boldsymbol{P} = [\boldsymbol{p}_1, \dots, \boldsymbol{p}_m] \in \mathcal{M}_{N,m} \tag{9}$$

Assuming this notation, the m *PGD modes* are defined by their PGD mode shapes $\boldsymbol{p}_j$ and their time dependences $q_j(t)$, assembled in a vector $\boldsymbol{q}(t) \in \mathbb{R}^m$. In this two-term product, the amplitude information is chosen to be given by $q_j(t)$ only so the $\boldsymbol{p}_j$ PGD mode shapes are normalized to 1.

This decomposition will to split the problem into two smaller problems (*cf.* Sec. 3.2.1 and Sec. 3.2.2) whose sizes depend on m. As NNMs are periodic solutions of Eq. (2), the temporal part $\boldsymbol{q}(t)$ will be decomposed using HBM:

$$\boldsymbol{q}(t) = \boldsymbol{Q}_H \boldsymbol{h}_H(t,\omega), \text{ where } \begin{cases} \boldsymbol{Q}_H = [\boldsymbol{a}_0, \boldsymbol{a}_1, \boldsymbol{b}_1, \dots] \\ \boldsymbol{h}_H = [1/\sqrt{2}, \cos(\omega t), \sin(\omega t), \dots]^{\mathrm{T}} \end{cases} \tag{10}$$

Computing a NNM point by a combined PGD/HBM approach is then equivalent to compute a $\{\boldsymbol{P}, \boldsymbol{Q}_H, \omega\}$ set.

3.2.1 Temporal problem $\mathcal{T}_m$

The objective of this section is to compute the temporal part – *i.e* $\boldsymbol{Q}_H$ and ω – knowing the PGD mode shapes – *i.e.* the matrix $\boldsymbol{P}$. A weak formulation is written on an oscillation period $I_T = [0, 2\pi/\omega]$, with the test function $\boldsymbol{u}^\star(t) = \boldsymbol{P}\,\boldsymbol{q}^\star(t)$:

$$\forall t \in I_T \;\; \forall \boldsymbol{q}^\star(t) \int_{I_T} \boldsymbol{q}^{\star T}(t) \boldsymbol{P}^T \boldsymbol{R}(\boldsymbol{P}\boldsymbol{q}(t)) dt = 0 \tag{11}$$

The final temporal problem is established from the weak formulation Eq. (11):

$$\forall t \in I_T \;\; \mathbf{M_r}\ddot{\mathbf{q}}(t) + \mathbf{K_r}\boldsymbol{q}(t) + \boldsymbol{f}_{\mathrm{nl}r}(\boldsymbol{P}\boldsymbol{q}(t), \boldsymbol{P}\dot{\boldsymbol{q}}(t)) = \mathbf{0} \tag{12}$$

where $\{\mathbf{M_r} = \boldsymbol{P}^T\boldsymbol{M}\boldsymbol{P}, \mathbf{K_r} = \boldsymbol{P}^T\boldsymbol{K}\boldsymbol{P}\} \in \mathcal{M}_m(\mathbb{R})^2$ and $\boldsymbol{f}_{\mathrm{nl}r} = \boldsymbol{P}^T\boldsymbol{f}_{\mathrm{nl}} \in \mathbb{R}^m$. This system of m second order nonlinear ODEs is solved by HBM. Using the same notations as in Sec. 3.1, the temporal problem $\mathcal{T}_m$ is defined as a square algebraic system with $m(2H+1)+1$ unknowns in the following manner:

$$\mathcal{T}_m(\boldsymbol{Q}_H, \omega | \boldsymbol{P}) = \mathbf{0} \Leftrightarrow \begin{cases} \boldsymbol{H}_{\mathrm{l}}(\omega)\boldsymbol{q}_H + \boldsymbol{H}_{\mathrm{nl}}(\boldsymbol{q}_H, \omega) = \mathbf{0} \\ c(\boldsymbol{q}_H, \omega) = 0 \end{cases} \tag{13}$$

where $\boldsymbol{q}_H = \{\boldsymbol{a}_0^{\mathrm{T}}, \dots, \boldsymbol{a}_k^{\mathrm{T}}, \boldsymbol{b}_k^{\mathrm{T}}, \dots\}^{\mathrm{T}}$ are the Fourier coefficients put into a vectorial form, $\boldsymbol{H}_{\mathrm{l}}$ is the linear contribution, $\boldsymbol{H}_{\mathrm{nl}}$ is the vector of Fourier coefficients of the nonlinear contribution of Eq. (12), and $c(\boldsymbol{q}_H, \omega) \in \mathbb{R}$ is an arbitrary constraint equation which allows to get a square system. $c(\boldsymbol{q}_H, \omega)$ will be specified in Sec. 4.1 as a continuation criterion.

3.2.2 Spatial problem $\mathcal{S}_m$

The objective of this section is to compute the spatial part – *i.e* the matrix $\boldsymbol{P}$ – knowing the temporal part – *i.e.* $\boldsymbol{Q}_H$ and ω. This time, a weak formulation is obtained with the test function $\boldsymbol{u}^\star = \sum_{k=1}^{m} \mathbf{p}_k^\star q_k$, leading to a set of m systems with N equations:

$$\forall k \in [\![1;m]\!],\ \sum_{j=1}^{m} \left(\int_{I_T} q_k \ddot{q}_j \, dt \, \boldsymbol{M} + \int_{I_T} q_k q_j \, dt \, \boldsymbol{K} \right) \boldsymbol{p}_j + \int_{I_T} q_k \boldsymbol{f}_{\text{nl}}(\boldsymbol{P}\boldsymbol{q}(t)) \, dt = \mathbf{0} \tag{14}$$

Finally, these equations can be condensed in a $N \times m$ algebraic system denoted $\mathcal{S}_m$:

$$\mathcal{S}_m(\boldsymbol{P}|\boldsymbol{Q}_H, \omega) = \mathbf{0} \Leftrightarrow \boldsymbol{S}_{\text{l}} \, \tilde{\mathbf{p}} + \boldsymbol{S}_{\text{nl}}(\tilde{\mathbf{p}}) = \mathbf{0} \tag{15}$$

where $\tilde{\mathbf{p}} = \{\boldsymbol{p}_1^{\text{T}}, \ldots, \boldsymbol{p}_m^{\text{T}}\}^{\text{T}} \in \mathbb{R}^{N \times m}$ contains the columns of $\boldsymbol{P}$, $\boldsymbol{S}_{\text{l}}$ is the linear contribution with respect to $\tilde{\mathbf{p}}$, and $\boldsymbol{S}_{\text{nl}}(\tilde{\boldsymbol{p}}) = [\int_{I_T} q_k \boldsymbol{f}_{\text{nl}}(\boldsymbol{P}\boldsymbol{q}(t)) \, dt]_{1 \leq k \leq m}$ the nonlinear one. $\boldsymbol{S}_{\text{nl}}$ is computed using the AFT approach, and $\boldsymbol{S}_{\text{l}}$ can be established analitically:

$$\boldsymbol{S}_{\text{l}} = \boldsymbol{I}_2 \otimes \boldsymbol{M} + \boldsymbol{I}_0 \otimes \boldsymbol{K} \text{ , with } \boldsymbol{I}_k = \pi \omega^{k-1} \boldsymbol{Q}_H \boldsymbol{D}^{k^{\text{T}}} \boldsymbol{Q}_H^{\text{T}} \tag{16}$$

The spatial subproblem $\mathcal{S}_m$ is more expensive to solve than $\mathcal{T}_m$ as its size depends on $N >> m$.

3.2.3 PGD/HBM fixed point algorithm

The last step of the PGD process consists in integrating both temporal and spatial subproblems into a fixed point algorithm. The full fixed point loop is detailed in Algorithm 1. The subproblems resolution lies at lines 6 and 7. Two main issues have to be adressed: the choice of an error measure ε and the way in which potential additional PGD modes are initialized.

The error measure ε, defined at lines 1 and 11, compares the norm of the residue vector $\boldsymbol{R}(\boldsymbol{u}(t))$ – where $\boldsymbol{u}(t) = \boldsymbol{P}\boldsymbol{Q}_H \boldsymbol{h}_H(t, \omega)$ – to $||\boldsymbol{K}\boldsymbol{u}(t)||$ over a period I_T. Unlike in Grolet and Thouverez work [11] in which $k < k_{max} = 3$ was the only stopping criterion of the loop, here the convergence is checked by the use of this physical criterion, introduced at line 4. Moreover the fixed point loop is broken when $k > k_{max}$ or if the current iteration does not reduce ε enough, that is, when ε is greater than $r_{\text{fp}} \varepsilon_{\text{prev}}$ where $r_{\text{fp}} < 1$ and $\varepsilon_{\text{prev}}$ denotes the error measured at the end of the previous iteration.

If the convergence criterion $\varepsilon \leq \varepsilon_{\text{max}}$ is not fulfilled at the end of the fixed point loop, the choice is made to add a new PGD mode (from line 13): a new column $\boldsymbol{p}_j$ is added to $\boldsymbol{P}$ and the associated row is added to the $\boldsymbol{Q}_H$ matrix. In previous works [11], random values were used to fill these new vectors, and the order in which subproblems are dealt with in the algorithm did not seem to be important. Here, the initial observation is that it is difficult to initialize $\boldsymbol{Q}_H$ with new temporal information. Hence, the temporal problem is computed first into the fixed point loop, with a new row filled with zeros. The new column $\boldsymbol{p}_j$ is initialized with the shape of the $(n_0 + m)$-th LNM, with n_0 being the index of the n_0-th NNM being computed. Indeed, this spatial data is easily computable and it seems relevant to add new shapes from the LNMs as they form a basis. Moreover, avoiding random initializations makes the algorithm more robust and results reproducible.

4 NNM continuation using PGD/HBM modes enrichment

In this section, a full NNM computation algorithm combining PGD/HBM and continuation schemes is detailed. Near the n_0-th LNM, first NNM points are searched using a single PGD mode ($m = 1$) which initially has the shape of the n_0-th LNM. When the mechanical energy is growing, new PGD modes are included to

Algorithm 1: PGD/HBM global algorithm

Parameters: Values for m_{max}, k_{max}, ε_{max} and r_{fp};
Definition of $c(\boldsymbol{q}_H, \omega)$ used in $\mathcal{T}_m$;
LNMs: $\boldsymbol{\Phi} = [\boldsymbol{\phi}_{n_0}, \ldots, \boldsymbol{\phi}_n]$.

Data: Initial values for m, $\boldsymbol{P}$, $\boldsymbol{Q}_H$ and ω.

$\varepsilon \leftarrow \max_{t\in I_T}(||\boldsymbol{R}(\boldsymbol{u}(t))||/||\boldsymbol{K}\boldsymbol{u}(t)||)$;
while $m \leq m_{\text{max}}$ **and** $\varepsilon > \varepsilon_{\text{max}}$ **do**
  $k \leftarrow 0$, $\varepsilon_{\text{prev}} \leftarrow 2\,\varepsilon$;
  while $k \leq k_{\text{max}}$ **and** $\varepsilon \geq \varepsilon_{\text{max}}$ **and** $\varepsilon \leq r_{\text{fp}}\varepsilon_{\text{prev}}$ **do** /* FP loop */
    $k \leftarrow k + 1$, $\varepsilon_{\text{prev}} \leftarrow \varepsilon$;
    $\boldsymbol{Q}_H, \omega \leftarrow$ solutions of $\mathcal{T}_m(\boldsymbol{Q}_H, \omega | \boldsymbol{P}) = \boldsymbol{0}$;
    $\boldsymbol{P} \leftarrow$ solution of $\mathcal{S}_m(\boldsymbol{P} | \boldsymbol{Q}_H, \omega) = \boldsymbol{0}$;
    for $j \leftarrow 1$ **to** m **do** /* Mode normalization */
      $\boldsymbol{p}_j \leftarrow \boldsymbol{p}_j / ||\boldsymbol{p}_j||$;
    end
    $\varepsilon \leftarrow \max_{t\in I_T}(||\boldsymbol{R}(\boldsymbol{u}(t))||/||\boldsymbol{K}\boldsymbol{u}(t)||)$;
  end
  if $\varepsilon \geq \varepsilon_{\text{max}}$ **then** /* Next mode initialization */
    $\boldsymbol{P} \leftarrow [\boldsymbol{P}, \boldsymbol{\phi}_{n_0+m}]$, $\boldsymbol{Q}_H \leftarrow \begin{bmatrix} \boldsymbol{Q}_H \\ \boldsymbol{0}_{1,2H+1} \end{bmatrix}$;
    $m \leftarrow m + 1$;
  end
end
Return m, $\boldsymbol{P}$, $\boldsymbol{Q}_H$, ω and ε;

the decomposition only when it is necessary along the path. The hightlights of the method are a dimension varying continuation scheme which keeps the number of NNM descriptors as small as possible and with an on the fly enrichment of the PGD modes set as the system behavior becomes more complex.

Before giving some details about the continuation scheme, we will define $\boldsymbol{y}_n$ as the the n-th point on the current path. As explained in Sec. 3.2, a point of the NNM is fully determined by its $\{\boldsymbol{P}, \boldsymbol{Q}_H, \omega\}$ set. Hence we will note:

$$\boldsymbol{y}_n \equiv \{\boldsymbol{P}_n, \boldsymbol{Q}_{Hn}, \omega_n\} \quad \text{(size: } N\,m + (2H+1)\,m + 1\text{)} \tag{17}$$

A distance d between two consecutive points $\boldsymbol{y}_{n-1}$ and $\boldsymbol{y}_n$ is also defined. It is weighted by a set of positive constants $\{\alpha_P, \alpha_Q, \alpha_\omega\}$ which are arbitrary continuation control parameters, such as:

$$d(\boldsymbol{y}_{n-1}, \boldsymbol{y}_n) = \sqrt{\alpha_P^2 ||\boldsymbol{P}_n - \boldsymbol{P}_{n-1}||^2 + \alpha_Q^2 ||\boldsymbol{Q}_{Hn} - \boldsymbol{Q}_{Hn-1}||^2 + \alpha_\omega^2 |\omega_n - \omega_{n-1}|^2} \tag{18}$$

The prediction and correction equations used in Sec. 4.1 both rely on this choice of distance.

4.1 Prediction/Correction continuation process

Although more complex continuation procedures exist [24], the algorithm presented in this work can be used with a choice of simple predictor and correctors. The next point sought, denoted $\boldsymbol{y}_\star$, is computed from a predicted point $\boldsymbol{y}_\star^{pred}$ obtained with a secant method whose the arclength is set by Δs. This predictor is chosen because of its simplicity. The correction step relies on the PGD/HBM solver described previously in Sec. 3.2 with a specific choice of constraint equation $c(\boldsymbol{q}_H, \omega) = 0$, linked to a classical correction method (*e.g.* arclength, pseudo-arclength, etc.). The global algorithm is depicted in Algorithm 2.

Algorithm 2: Continuation with on the fly PGD enrichment algorithm

Parameters: Values for Δm, $m_{\text{max tot}}$, r_ε, r_s^{mult}, r_s^{div} and Δs;
Values for Algorithm 1:
- k_{max}, ε_{max} and r_{fp};
- LNMs: $\boldsymbol{\Phi} = [\boldsymbol{\phi}_{n_0}, \ldots, \boldsymbol{\phi}_n]$;
- Constraint equations: $c_2(\boldsymbol{q}_H, \omega)$ and $c_p(\boldsymbol{q}_H, \omega)$.

Data: Circular frequency of n_0-th mode ω_{n_0}.

1 $\boldsymbol{y}_1 \leftarrow \{\boldsymbol{\phi}_{n_0}, \mathbf{0}_{1,2H+1}, \omega_{n_0}\}$; // First point
2 $c(\boldsymbol{q}_H, \omega) \leftarrow c_2(\boldsymbol{q}_H, \omega)$, $m_{\text{max}} \leftarrow m_{\text{max tot}}$; // Second point
3 Compute $\boldsymbol{y}_2$ using Algorithm 1 with $\boldsymbol{y}_1$ as initial value;
4 $c(\boldsymbol{q}_H, \omega) \leftarrow c_p(\boldsymbol{q}_H, \omega)$; // Continuation loop
5 $p \leftarrow 2$;
6 **while** $m \leq m_{\text{max tot}}$ **do**
7 $\quad m_{\text{max}} \leftarrow m + \Delta m$;
8 $\quad$ Compute $\boldsymbol{y}_\star$ using secant method with $\boldsymbol{y}_p$, $\boldsymbol{y}_{p-1}$ and Δs; // Prediction
9 $\quad$ Compute $\boldsymbol{y}_\star$ using Algorithm 1 with $\boldsymbol{y}_\star$ as initial value; // Correction
10 $\quad$ **if** $\varepsilon < \varepsilon_{\text{max}}$ **then**
11 $\quad\quad \boldsymbol{y}_{p+1} \leftarrow \boldsymbol{y}_\star$;
12 $\quad\quad p \leftarrow p + 1$;
13 $\quad\quad$ **if** $\varepsilon < r_\varepsilon \varepsilon_{\text{max}}$ **then**
14 $\quad\quad\quad \Delta s \leftarrow r_s^{mult} \Delta s$;
15 $\quad\quad$ **end**
16 $\quad$ **else**
17 $\quad\quad \Delta s \leftarrow r_s^{div} \Delta s$;
18 $\quad$ **end**
19 **end**

4.1.1 Prediction step features

The prediction arclength Δs is managed in a classical way, as follows: if the error ε measured after a correction step is too large (see Algorithm 1 in Sec. 3.2.3), Δs is decreased by multiplying it by $r_s^{div} < 1$ (see line 17) leading to a closer prediction point $\boldsymbol{y}_\star^{pred}$. On the contrary, if $\varepsilon < r_\varepsilon \varepsilon_{max}$ with $r_\varepsilon < 1$, Δs is multiplied by $r_s^{mult} > 1$ (see lines 13 and 14). Else Δs is kept constant for the next prediction step.

A secant prediction needs two solution points computed beforehand. The first point of the n_0-th NNM is already known: $\boldsymbol{y}_1 \equiv \{\boldsymbol{\phi}_{n_0}, \mathbf{0}_{1,2H+1}, \omega_{n_0}\}$. An amplitude condition imposed on the j-th dof at $t = 0$ is chosen to compute $\boldsymbol{y}_2$: $u_j(t = 0) = u_{j0}$ with $u_{j0} \neq 0$. This particular constraint, denoted $c_2(\boldsymbol{q}_H, \omega)$ in Algorithm 2, is injected into Algorithm 1 to find $\boldsymbol{y}_2$ starting from $\boldsymbol{y}_1$.

The reader could remark that during the prediction step, $\boldsymbol{y}_n$ can have a greater size than $\boldsymbol{y}_{n-1}$ because of the possibility to get different problem dimensions m along the path. The distance given by Eq. (18) need summations of matrices that do not have the same sizes, and the choice of padding $\boldsymbol{y}_{n-1}$ missing components with zeros instead of truncating $\boldsymbol{y}_n$ is made. Hence, all computed data are taken into account to reach a better prediction.

4.1.2 Correction step features

The startpoint of the correction step is $\boldsymbol{y}_\star^{pred}$. It is injected in the PGD/HBM solver described in Algorithm 1, in which the constraint equation involved in the temporal problem $\mathcal{T}_m$ is set to $c_p(\boldsymbol{q}_H, \omega)$, as denoted in Algorithm 2. $c_p(\boldsymbol{q}_H, \omega)$ defines the correction method and can be freely chosen among the classical possibilities

[19]. Here an arclength correction is used: $c_p(\boldsymbol{q}_{H\star}, \omega_\star) = d(\boldsymbol{y}_n, \boldsymbol{y}_\star)^2 - \Delta s^2$. This condition involves $\boldsymbol{P}_\star$ which is the last value of the spatial modes returned after the spatial problem $\mathcal{S}_m$ when processing the current temporal problem $\mathcal{T}_m$.

A remarkable point of this continuation scheme is the ability to increase the size of the problem when a new point $\boldsymbol{y}_{n+1}$ is computed from $\boldsymbol{y}_n$. At line 7 in Algorithm 2, the highest difference of size between $\boldsymbol{y}_n$ and $\boldsymbol{y}_{n+1}$ is defined as Δm. Consequently, two consecutive points can have different sizes, which has to be taken into account when $c_p(\boldsymbol{q}_H, \omega)$ is evaluated. Indeed, an arclength condition requires $d(\boldsymbol{y}_n, \boldsymbol{y}_\star)$. This time, a truncation is applied for the current point $\boldsymbol{y}_\star$ so the arclength condition is evaluated in a space which has the size of $\boldsymbol{y}_n$. Thus, no geometrical condition is imposed on the new component introduced in $\boldsymbol{y}_\star$.

4.1.3 Progressive PGD – a lighter variant

As said in Sec. 3.2.2, considering the oPGD method, $\mathcal{S}_m$ is more expensive to solve than $\mathcal{T}_m$ as its size is $N \times m$. This observation motivates the use of a variant of the oPGD approach developed before, in which $\mathcal{S}_m$ is solved as few times as possible, and in which only a few number of $\mathcal{S}_m$ equations is solved. In order to reach these goals, severals choices are made:

- During the correction step, the spatial problem is only solved when a new PGD mode is added, *i.e.* when the condition at line 13 in Algorithm 1 is fulfilled. Hence, most of the fixed point loops only solve the temporal problem. The matrix $\boldsymbol{P}_\star$ is put as the initialization of $\mathcal{T}_m$. This is consistent with the fact that in practice, the nature of many points close together in the NNM branch is not sufficiently different to require the introduction of a new PGD mode shape.
- In addition to that, in order to deal with the case when $\mathcal{S}_m$ has to be solved, a new lighter spatial problem is defined. The PGD mode shapes contained in $\boldsymbol{P}$ are definitively locked once computed, and only the new $\boldsymbol{p}_j$ added at line 13 in Algorithm 1 is updated by the spatial solver. This constitutes a gain in algorithmic complexity since the new size of the spatial problem is only N. The procedure leading to the analytical expression of the spatial problem is analogous to the Galerkine process described in Sec. 3.2.2, except that this time the projection is only made on the new PGD mode shape $\boldsymbol{p}_j$. The expression of this N-sized system can be found in [11].

This variant is usually called *progressive PGD* (pPGD). It allows to improve the numerical performance of the algorithm described before. However, its main disavantage is the lack of flexibility of the PGD mode shapes, which can finally generate NNM branches of higher dimension m_{max} because more spatial information is needed when the system behavior becomes more complex.

5 Numerical application on two beam models

The main branch of the first NNM of a beam model with a cubic nonlinearity described in Sec. 5.1 is built using the full PGD/HBM continuation algorithm. The main results proper to the presented approach are given in Sec. 5.2. This numerical study has been performed for the presented oPGD/HBM method and its pPGD/HBM variant, but for the sake of clarity only the pPGD results are presented on the figures of this section. Finally in Sec. 5.3 some numerical considerations are made for the presented example and another similar beam model with a localized contact. Hence, two different classes of conservative nonlinearities are adressed in this paper.

5.1 Description of the beam models

The considered system is an Euler-Bernouilli bending beam with one end clamped and a spring at its free end; it is spatially discretized by FEM, as illustrated in Fig. 1. The beam properties are: Young modulus

$E = 210$ GPa, density $\rho = 7800$ kg.m^{-3}, area of the square section $S = 9\ 10^{-4}$ m^2, second area moment $I = 6.71\ 10^{-8}$ m^4 for an in-plane bending and length $L = 1$ m. The linear stiffness of the spring is $k = 4\ 10^4$ N.m^{-1}. The first eigenvalues of the underlying linear system consisting of the beam plus the linear spring are $\{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5\} = \{217, 999, 2769, 5420, 8960\}$ rad.s^{-1}. There are $N_e = 20$ elements with two dof per node: a transverse displacement u and a rotation θ, so there are $N = 40$ dof.

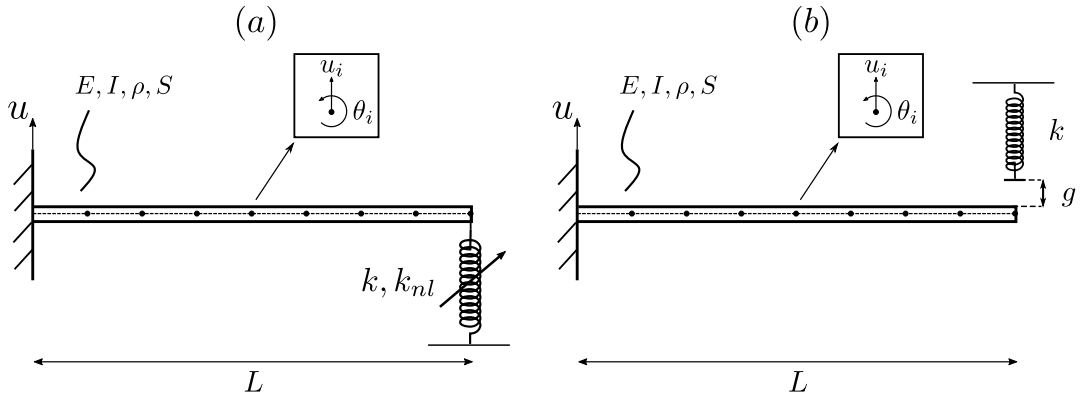

Figure 1: Euler-Bernouilli cantilever beam (a) with a transverse cubic spring at dof 1 (b) with a transverse gap between dof 1 and the linear spring.

- On the left in Fig. 1, a polynomial non linearity is chosen by introducing a cubic stiffness k_{nl} to the spring. The nonlinear contribution is $\boldsymbol{f}_{\text{nl}}(t) = [k_{nl} u_1^3(t)\ \mathbf{0}_{N-1}]^T$ with $k_{nl} = 9.2\ 10^5$ N.m^{-3}. Otherwise, we consider an underlying linear system made of the mass and stiffness matrices of the beam plus the linear stiffness of the spring – *i.e.* $K_{11} = K_{11beam} + k$. $H = 50$ harmonics are considered on an exploratory basis, in particular because the algorithmic complexity stays reasonable. Indeed, about twenty harmonics would be enough to deal with the involved physics.

- On the right in Fig. 1, a one-sided contact problem is also described: a gap $g = 2$ cm is introduced between the free end of the beam and the linear spring. The underlying linear system is the Euler-Bernouilli cantilever beam. The unilateral contact is modelled as follows:

$$\boldsymbol{f}_{\text{nl}}(t) = \begin{cases} [k(u_1(t) - g)\ \ \mathbf{0}_{N-1}]^T, & \text{if } u_1(t) > g \\ \mathbf{0}_N, & \text{otherwise} \end{cases} \tag{19}$$

  This unsmoothed contact is quite hard as $k/\dfrac{3EI}{L^3} = 94.56\%$ with $\dfrac{3EI}{L^3}$ the equivalent stiffness of the cantilever beam taken alone. A contact problem between two identical beams would be similar from the stiffness point of view. Consistently with this class of problem, the arbitrary value $H = 70$ is chosen.

  The results obtained for this second model will not be detailed in this paper. However, some numerical values will be given in Sec. 5.3 to extend some conclusions to this class of nonlinearities.

Parameters required by Algorithm 2 are given in Table 1. As both problems are conservative, a cosine basis is sufficient to describe the harmonic behavior of the system, so only $(H + 1)$ Fourier coefficients have to be computed instead of $(2H + 1)$.

5.2 Bending beam with a cubic spring at its free end

A convenient way to present the NNM branches is a Frequency-Energy plot (FEP) [1, 8], E being the mechanical energy of the NNM. The cubic spring case NNM1 (tangent to $\boldsymbol{\phi}_1$ at almost null amplitudes) is

Δm	$m_{\text{max tot}}$	r_ε	r_s^{mult}	r_s^{div}	Δs	$k_{\max}$	$\varepsilon_{\max}$	r_{fp}	$\boldsymbol{y}_2 : u_1(t=0)$	$\{\alpha_P, \alpha_q, \alpha_\omega\}$
1	10	0.1	1	0.5	1.5	10	10^{-3}	0.8	10^{-3} m	$\{10^{-6}, 10^{-1}, 10^{-4}\}$
(5)		(0.5)	(1.2)		(1)				(5 10^{-4} m)	($\{0, 10^{-6}, 1\}$)

Table 1: Algorithm 2 required parameters for the cubic spring case. The contact case parameters are given in parentheses on the second line, only when they differ from the previous ones.

displayed in a FEP given in Fig. 2. N_{pt} is the index of converged solution points through the 131 points computed during this pPGD/HBM continuation. Only 7 PGD modes are required by the pPGD algorithm (*cf.* Fig. 3) to build this branch which reaches about $E_{max} = 2.25\ 10^5$ J. The same branch can be computed with only 6 PGD modes using oPGD. Indeed the shapes of the PGD modes are always the same through a pPGD/HBM continuation process because the method involves blocking the shapes once they are integrated into the $\boldsymbol{P}$ matrix. This lack of flexibility can lead to the introduction of more PGD modes than with oPGD. Both variants obviously give identical FEP.

The interpretation of the modal interactions drawn in Fig. 2 (interaction tongue and final internal resonance) are deliberately limited in order to focus on the aspects specific to the numerical method developed.

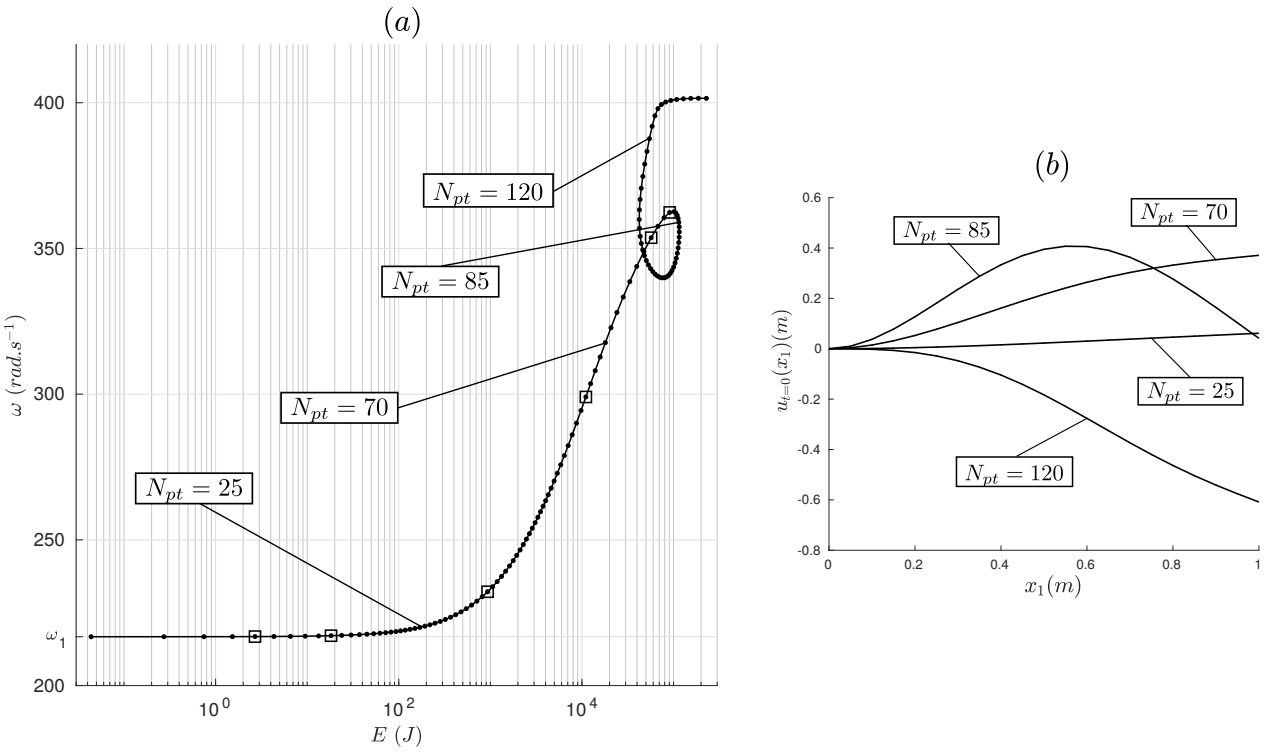

Figure 2: (a) Main branch of NNM1 in a FEP and (b) its mode shape at null velocity points for four continuation points. Squares: solution points where a PGD mode is added.

It is recalled that the NNM framework used in this paper is the Rosenberg's one [3]. In particular, all dofs reach their maximum at the same time, hence the NNM shapes at null velocity plotted in Fig. 2 gives interesting physical data. The more energy grows the more the nonlinear spring takes importance against the bending inertia of the beam, generating localized curvature inversions near the free end. Qualitatively similar results are obtained by Kerschen for another cantilever beam with cubic spring in [25].

On Fig. 3, the error measure ε along the path is depicted along with the number of PGD modes. The process for adding PGD modes is clearly identified: ε reaches the limit value ε_{max} then a new PGD mode is embedded in the calculations leading to a decrease of the error. Fig. 2 shows that the PGD modes are gradually added along the branch as expected with a smooth cubic nonlinearity. Please note that the error

for the first point $N_{pt} = 1$ is not displayed as it is null and cannot appear in a logarithmic scale. All these interpretations are retrieved in oPGD results.

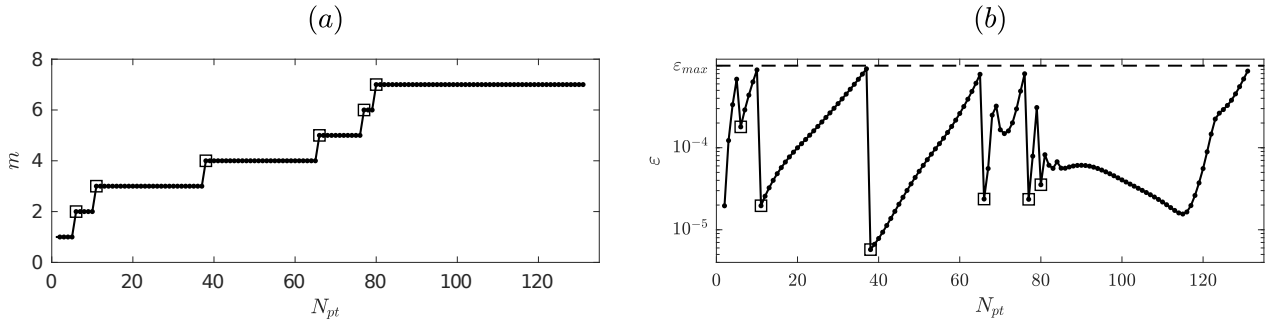

Figure 3: (a) Number of PGD modes m and (b) error measure ε against point index N_{pt}. Squares: solution points where a PGD mode is added.

An analysis of the seven PGD mode shapes $\boldsymbol{p}_j$ is provided on Fig. 4. An interesting feature proper to both pPGD and oPGD approaches is the presence of PGD modes which involve the participation of several LNMs. For instance such "combined mode" is here the PGD mode 3, which gathers contributions of the first 6 LNMs. With such an approach, high order LNMs can be taken into account in a reduced range of PGD mode shapes: here 5 PGD modes are sufficient to describe the system behavior until the interaction tongue and they involve the contributions of at least the first ten LNMs. Analoguously only 4 PGD modes are required to build the same area using oPGD, with the contributions of the first 6 LNMs.

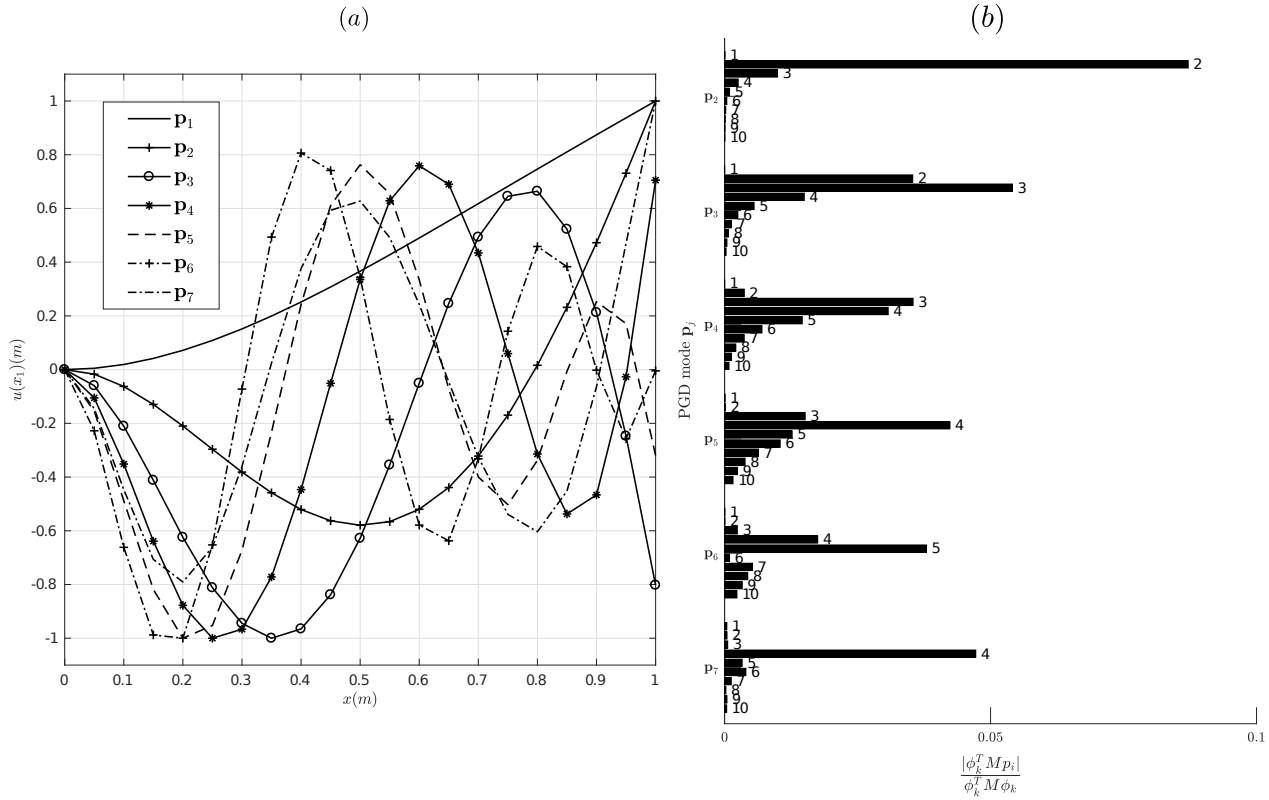

Figure 4: Analysis of PGD spatial modes. (a) PGD mode shapes. Normalization is made at the maximum deflection point. (b) Participation factors of LNMs in each PGD mode. In each bar group, LNMs participations are numbered in ascending order until the tenth linear mode. $\boldsymbol{p}_1$ is not displayed as it is LNM1 shape.

The temporal information associated with each PGD mode shape is plotted on Fig. 5. The significant variations of the temporal part through the NNM branch are expected as the previously described spatial part does not change through the pPGD/HBM continuation and the temporal part contains all the amplitude information. Only odd harmonics are plotted as all the even ones are null because of the cubic nature of the nonlinearity. As a global trend high order harmonics have less influence: at the higest energy point, all the amplitudes $|a_k|$ are sub-millimetric from $k = 23$, for all the PGD modes.

The PGD mode 1 has the simplest behavior as it stays mainly led by the fundamental cosine $|a_1|$ all along the continuation. Other PGD modes have a richer harmonic behavior with shared influences of the cosine coefficients. For example at the end of the branch, the fifteen first odd harmonics are predominant in PGD modes 2–4 and the highest harmonics are more represented by PGD modes 5–7. PGD mode shapes are combination of LNMs, as shown on Fig. 4, and this partly explains why the harmonics spectrum is complex on Fig. 5. All these qualitative interpretations also highlight the growing complexification of the beam behavior and the growing influence of high harmonics as energy grows. Here also similar results and interpretations about the temporal information are found with oPGD.

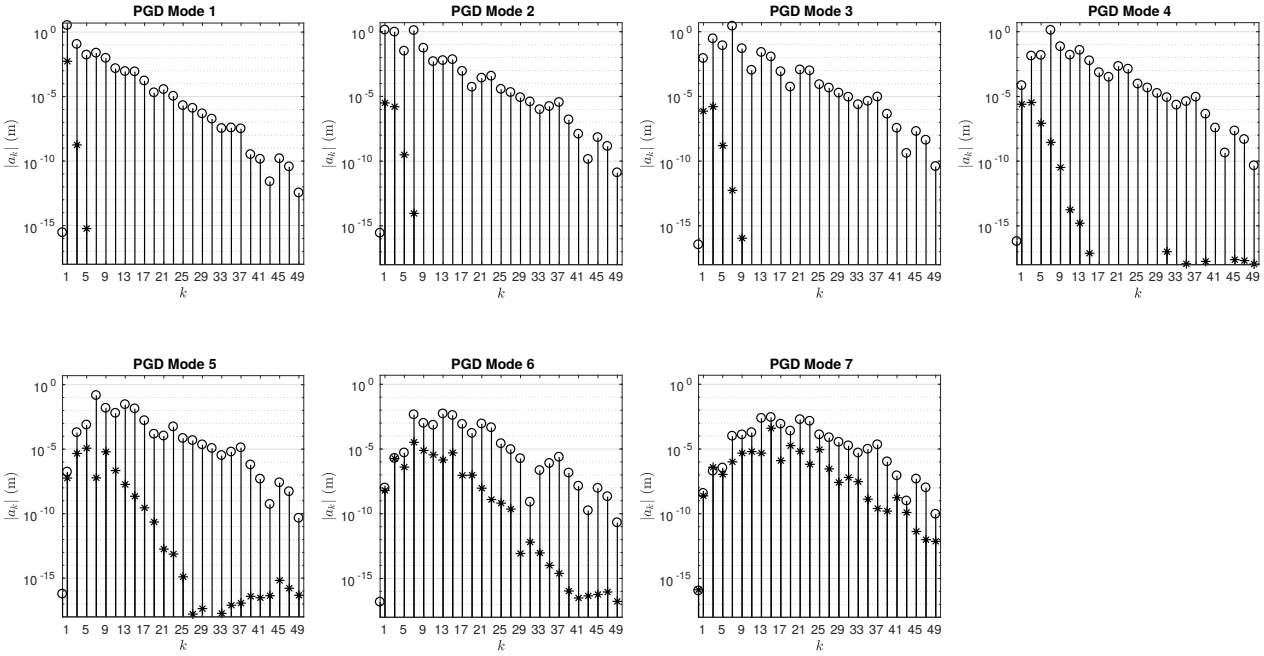

Figure 5: Amplitudes of (cosine) Fourier coefficients of PGD modes. Crosses: coefficients when the PGD mode is first introduced; Circles: coefficients at the end of the branch ($N_{pt} = 131$).

Finally the main objectives of the method are met: a NNM branch is built with a few variables, the number of PGD modes is kept as small as possible by incrementing m only when the convergence criterion $\varepsilon \leq \varepsilon_{max}$ is not fulfilled, and all physical data belonging to the case under study is retrieved. In the next section some details about the algorithmic performance are detailed. The presented oPGD algorithm is compared with its pPGD variant. Not only information about the the cubic spring case is given but also about the contact model described in Sec. 5.1.

5.3 Numerical considerations

Few descriptors are stored to describe the NNM branch so the main objective of reduced model approaches is reached. Table 2 shows this improvement for both PGD/HBM algorithms.

Please note that the first NNM of the contact problem has been obtained with the parameters given in Table 1 until a mechanical energy $E \approx 6.3\ 10^4$ J. Succintly, 8 PGD modes and 115 points were required to compute

the whole branch and setting $\Delta m = 5$ allowed to directly add 3 PGD modes in a single solution point (that is when the beam first hits the spring). This shows that describing sharp or gradual changes of the system behavior by the PGD/HBM continuation scheme with on the fly enrichment is feasible.

	pPGD/HBM $\Sigma_m\{N_{pt}^m \times (m(H+1)+1) + N\Delta m\}$	**oPGD/HBM** $N_{pt} m(N+H+1)$	**HBM** $N_{pt}\ (N(H+1)+1)$
Cubic Spring case	**33266**	**58695**	**267371**
Contact case	**39161**	**60606**	**326715**

Table 2: Comparison between the number of descriptors needed by pPGD/HBM, oPGD/HBM and classical HBM for both examples. The reference for the N_{pt}^m values are the ones provided by pPGD/HBM computation.

Only $m \times (N + H + 1)$ descriptors by solution point are required by the oPGD/HBM continuation whereas a classical HBM continuation needs $N(H + 1) + 1$ ones. In the case of the pPGD implementation, a single spatial problem of size $N \times \Delta m$ is solved for a whole NNM segment with m PGD modes: the total number of required descriptors is further reduced to reach $\Sigma_m\{N_{pt}^m \times (m(H + 1) + 1) + N\Delta m\}$, where N_{pt}^m is the number of solution points in the NNM segment with m PGD modes.

Obviously calculating a solution point via PGD/HBM is relevant if and only if $m \times (N + H + 1) < N \times (H + 1)$. An easy proof gives the necessary condition for the PGD/HBM method to be more interesting than a classical HBM approach: $m < H + 1$. In other words, the PGD/HBM method is useful for problems which require a large number of harmonics.

In both examples the saving of variables with respect to a classical HBM can be assessed with the following ratios: $1 - \frac{m\times(N+H+1)}{N\times(H+1)}$ for oPGD, and $1 - \frac{m\times(H+1)}{N\times(H+1)}$ or $1 - \frac{m\times(H+1)+N\Delta m}{N\times(H+1)}$ for a pPGD solution point. The oPGD reduction is very efficient, with 78% and 81% less variables than HBM for the cubic spring and the contact cases respectively. The pPGD reduction goes further with about 88% less variables than HBM for both examples.

We focus now on the underlying trust-region solver and its iteration data used to solve the subproblems. This algorithm requires function and abscissa tolerances – set on 10^{-12} here – and allows to set a maximum number of iterations $N_{\text{ite max}}$. The values $N_{\text{ite max}} = 100$ and $N_{\text{ite max}} = 25$ are set respectively for the cubic spring and the contact cases but this limit is reached for only very few points (maxima reached when a PGD mode is about to be added or at the first turning point of the interaction tongue). In Table. 3 are given mean and maximum numbers of solver iterations for temporal and spatial problems through the continuation, for both examples and both PGD approaches. The oPGD iteration data is obtained with the same values than with pPGD (*cf.* Table 1) except the following ones to improve the continuation: $\alpha_P = 10^{-6}$ and $\boldsymbol{y}_2 : u_1(t = 0) = 10^{-3}\ m$ for the cubic spring case, and $r_s^{mult} = 1.2$ for the contact case.

			Mean iteration number	Standard deviation	Maximum iteration number
Cubic spring case	oPGD	$\mathcal{T}_m$	6.23	4.89	22
		$\mathcal{S}_m$	16	10.92	54
	pPGD	$\mathcal{T}_m$	6.56	17.96	206
		$\mathcal{S}_m$	1.29	6.23	46
Contact case	oPGD	$\mathcal{T}_m$	2.18	1.53	16
		$\mathcal{S}_m$	9.77	8.68	49
	pPGD	$\mathcal{T}_m$	3.56	5.52	42
		$\mathcal{S}_m$	0.35	1.92	14

Table 3: Mean, standard deviation and maximum of iteration numbers for $\mathcal{T}_m$ and $\mathcal{S}_m$ problems, through the continuation.

In a oPGD framework the spatial problem is heavier as expected. The pPGD allows to compute less spatial subproblems so many solution points include a zero iteration number for the spatial problem in the mean and

standard deviation calculation. The cost of the space problem is thus well bypassed by this variant. Otherwise $N_{\text{ite max}}$ could be an interesting control parameter for bigger problems as one can keep it very small and save computation time by "converting" convergence issues into new PGD modes or a reduced arclength Δs.

6 Conclusion

A full PGD/HBM-based continuation technique with on the fly enrichment is presented in this paper. It is applied to two examples of beam models representing different classes of nonlinearies. These examples show that a highly reduced description of the NNMs branches can numerically be obtained. It is to be noted that spatial initializations rely on the LNMs of the system, which gives to the PGD mode shapes a physical meaning that a classical PGD approach provide *a priori*. As many parameters are let free (*cf.* Table 1), this algorithm has a certain flexibility.

The presented approach can be used to compute bigger structures with different kinds of nonlinearities. In order to exploit the physical potential of NNMs, future prospects could include the computation of damped NNMs or forced responses of a structure.

References

[1] G. Kerschen, M. Peeters, J. Golinval, A. Vakakis, *Nonlinear normal modes, part i: A useful framework for the structural dynamicist*, Mechanical Systems and Signal Processing, Vol. 23, No. 1, (2009), pp. 170-194.

[2] D. Laxalde, F. Thouverez, J.-J. Sinou, *Dynamics of a linear oscillator connected to a small strongly non-linear hysteretic absorber*, International Journal of Non-Linear Mechanics, Vol. 41, No. 8, (2006), pp. 969-978.

[3] R. Rosenberg, *On nonlinear vibrations of systems with many degrees of freedom*, Advances in Applied Mechanics, Vol. 9, Elsevier, (1966), pp. 155-242.

[4] S. Shaw, C. Pierre, *Non-linear normal modes and invariant manifolds*, Journal of Sound and Vibration, Vol. 150, No. 1, (1991), pp. 170-173.

[5] S. Shaw, C. Pierre, *Normal modes for non-linear vibratory systems*, Journal of Sound and Vibration, Vol. 164, No. 1, (1993), pp. 85-124.

[6] G. Kerschen, J.-C. Golinval, A. F. Vakakis, L. A. Bergman, *The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview*, Nonlinear Dynamics, Vol. 41, No. 1, (2005), pp. 147-169.

[7] L. Renson, G. Kerschen, *Nonlinear Normal Modes of Nonconservative Systems*, in G. Kerschen, D. Adams, A. Carrella, editors, *Topics in Nonlinear Dynamics, Volume 1. Conference Proceedings of the Society for Experimental Mechanics Series*, Vol 35, Springer, New York, NY, (2013), pp. 189-202.

[8] L. Renson, G. Kerschen, B. Cochelin, *Numerical computation of nonlinear normal modes in mechanical engineering*, Journal of Sound and Vibration, Vol. 364, (2016), pp. 177-206.

[9] M. Peeters, R. Vigui, G. Srandour, G. Kerschen, J.-C. Golinval, *Nonlinear normal modes, part ii: Toward a practical computation using numerical continuation techniques*, Mechanical Systems and Signal Processing, Vol. 23, No. 1, (2009), pp. 195-216.

[10] C. Joannin, B. Chouvion, F. Thouverez, J.-P. Ousty, M. Mbaye, *A nonlinear component mode synthesis method for the computation of steady-state vibrations in non-conservative systems*, Mechanical Systems and Signal Processing, Vol. 83, (2017), pp. 75-92.

[11] A. Grolet, F. Thouverez, *On the use of the proper generalised decomposition for solving nonlinear vibration problems*, in ASME 2012 International Mechanical Engineering Congress and Exposition, Vol. 4, pp. 913-920.

[12] F. Chinesta, P. Ladeveze, E. Cueto, *A short review on model order reduction based on proper generalized decomposition*, Archives of Computational Methods in Engineering, Vol. 18, No. 4, (2011), pp. 395-404.

[13] F. Chinesta, A. Ammar, E. Cueto, *Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models*, Archives of Computational Methods in Engineering, Vol. 17, No. 4, (2010), pp. 327-350.

[14] F. Chinesta, A. Leygue, F. Bordeu, J. V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, A. Huerta, *Pgd-based computational vademecum for efficient design, optimization and control*, Archives of Computational Methods in Engineering, Vol. 20, No. 1, (2013), pp. 31-59.

[15] A. Nouy, *A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations*, Computer Methods in Applied Mechanics and Engineering, Vol. 199, No. 2324, (2010), pp. 1603-1626.

[16] P. Sundararajan, S. T. Noah, *Dynamics of forced nonlinear systems using shooting/arc-length continuation method — application to rotor systems*, Journal of Vibration and Acoustics, Vol. 119, No. 1, (1997), pp. 9-20.

[17] M. Urabe, *Galerkin's procedure for nonlinear periodic systems*, Archive for Rational Mechanics and Analysis, Vol. 20, No. 2, (1965), pp. 120-152.

[18] M. Urabe, A. Reiter, *Numerical computation of nonlinear forced oscillations by galerkin's procedure*, Journal of Mathematical Analysis and Applications, Vol. 14, No.1, (1966), pp. 107-140.

[19] A. H. Nayfeh, D. T. Mook, *Nonlinear Oscillations*, Wiley, (1995).

[20] E. Sarrouy, J.-J. Sinou, *Non-linear periodic and quasi-periodic vibrations in mechanical systems - on the use of the harmonic balance methods*, in Advances in Vibration Analysis Research, InTech, (2011).

[21] R. Seydel, *From Equilibrium to Chaos, Practical Bifurcation and Stability Analysis*, Elsevier, (1988).

[22] E. Sarrouy, O. Dessombz, J.-J. Sinou, *Stochastic study of a non-linear self-excited system with friction*, European Journal of Mechanics - A/Solids, Vol. 40, (2013), pp. 1-10.

[23] T. M. Cameron, J. H. Griffin, *An alternating frequency/time domain method for calculating the steady-state response of nonlinear dynamic systems*, Journal of Applied Mechanics, Vol. 56, No. 1, (1989), pp. 149-154.

[24] E. L. Allgower, K. Georg, *Introduction to Numerical Continuation Methods*, Springer-Verlag, 2003.

[25] G. Kerschen, *Computation of Nonlinear Normal Modes through Shooting and Pseudo-Arclength Computation*, Springer Vienna, Vienna, 2014, pp. 215-250.