DOI: 10.13009/EUCASS2022-7310
9TH EUROPEAN CONFERENCE FOR AERONAUTICS AND SPACE SCIENCES (EUCASS)

Reduced-order models for aeronautics based on the LATIN-PGD method within non-linear industrial Simcenter Samcef software

Ronan Scanff [⋆,†], *Bruno Cauville*[⋆], *David Néron*[◇], *Pierre Ladevèze*[◇]
[⋆]*Siemens Industry Software SAS*
107 avenue de la République, 92320 Châtillon, France
ronan.scanff@siemens.com · bruno.cauville@siemens.com
[◇]*Université Paris-Saclay, CentraleSupélec, ENS Paris-Saclay, CNRS, LMPS - Laboratoire de Mécanique Paris-Saclay*
91190 Gif-sur-Yvette, France
david.neron@ens-paris-saclay.fr · pierre.ladeveze@ens-paris-saclay.fr
[†]Corresponding author

Abstract

A weakly-invasive version of the LATIN-PGD method is described to build reduced-order models within general-purpose industrial finite element software, such as Simcenter Samcef™ developed by Siemens Digital Industries Software. The originality of the approach lies in its generic character, which allows to manage of all types of non-linearities classically encountered in solid mechanics: material, geometric and contact non-linearities. Results are illustrated through two industrial test-cases and show impressive performance gains. Such reduced models or virtual charts can then be broadly used to deploy Digital Twins.

1. Introduction

As the complexity of systems explodes due to the desire to represent more finely the physics, increase the performance of products and foster innovation, simulations are becoming increasingly expensive in terms of computational time. Carrying out parametric studies on multi-scale or multi-physics problems with complex materials on millions of unknowns and over long periods of time remains challenging or even sometimes unreachable, despite recent tremendous growth in computing resources. To fulfil these growing expectations, new numerical methods have been emerging: Reduced-Order Modeling (ROM) methods[2] are one example, recognised in particular for drastically reducing computational times by taking advantage of the redundancy of information while breaking the so-called curse of dimensionality. On the one hand, the Proper Orthogonal Decomposition (POD)[6,17] and Reduced Basis (RB)[15,16] methods are characterised by a first costly learning step (*offline*) from a multitude of high fidelity resolutions (snapshots). On the other hand, the Proper Generalized Decomposition (PGD) method[8,10] follows a different path in the sense that the reduced model is built simultaneously with the resolution of the problem thanks to a minimisation process.

Because of their obvious asset emphasized many times through academic research, ROM methods are being looked at more and more by the industrial sector. However, the fact remains that their industrial deployment is still relatively limited. In order to be more widely conveyed to the entire scientific community – including non-expert users –, these methods must fit with the habits of design offices, which means introducing them natively into industrial software. To the authors' knowledge, ROM methods are still not implemented in general-purpose industrial finite element software and therefore are not part of engineers' everyday tools. One of the main obstacles is the intrusive nature[11] of these methods, which require operations and a data storage format atypical of commercial codes.

Recent efforts have focused on the development of non-intrusive ROM methods. Mainly oriented on *a posteriori* POD or RB approaches[1,5] at first, it is only later that non-intrusive PGD approaches[3,12] have appeared. The latter are still based on the use of snapshots due to the *a priori* formulation of the PGD, which is intrinsically more intrusive. Alternatively, and for specific applications, one can find some non-intrusive implementations dedicated to PGD[9,14,19,20] which systematically rely on the external coupling between an industrial software and an *in-house* code developed in one laboratory.

Assuming quasi-staticity and moderate displacement outside instability zones, the proposed work provides a methodology for the construction, the enhancement and the handling of reduced-order models natively into the general-

DOI: 10.13009/EUCASS2022-7310

purpose industrial finite element software Simcenter Samcef™ by proposing a weakly-invasive version of the LATIN-PGD method. By doing that, we combine the advantages of the PGD reduced-order modelling method with all facilities already included in the industrial software, implying de facto any kind of non-linearities – materials, contacts, large transformations, etc. – frequently encountered in industrial component models. Robustness being a fundamental component as well as performance, the proposed approach does not require any *in-house* external code that would entail redesigning entire non-linear parts to achieve a sufficient level of robustness.

The remaining part of this paper is organized as follows: after recalling the type of problems addressed, a brief presentation is given of the weakly-invasive LATIN-PGD methodology deployed. Then, we present the results on different industrial test-cases specific to the aeronautics sector, highlighting attractive performance gains. Furthermore, we explain how fast and efficient construction of such reduced basis or virtual charts can be interesting for the deployment of Digital Twins: coupled with *a posteriori* real in-service data, the rapid evaluation of these reduced-order models within an end-to-end industrial software chain enables predictive maintenance operations to be carried out more easily.

2. Methodology

2.1 Reference problem

We consider the case of a classical mechanical problem solved by any general-purpose industrial finite element software: assuming a time interval $I \subset \mathbb{R}_+$, a domain $\Omega \subset \mathbb{R}^3$ is subject to different kinds of loading. Several sources of time-dependent non-linearities may be involved: contacts, (visco-)plasticity, large transformations, etc. Written in a formalism of generalized quantities on the finite element space $\boldsymbol{\mathcal{V}} \equiv \mathbb{R}^n$ with n the number of degrees of freedom, we are looking for the displacement field $\underline{u} \in \mathcal{I}_{\boldsymbol{\mathcal{V}}} \equiv L^2(I;\boldsymbol{\mathcal{V}})$, as well as the possible Lagrange multipliers $\underline{\xi} \in \mathcal{I}_{\boldsymbol{\mathcal{V}}_c} \equiv L^2(I;\boldsymbol{\mathcal{V}}_c)$ – characterising the additional contact conditions, if any, with $\boldsymbol{\mathcal{V}}_c \equiv \mathbb{R}^{n_c}$ and n_c the number of degrees of freedom associated with the contacts. By noting $\underline{v}$ the trace of the generalized displacements on the contact zone and $\underline{f}^{\text{ctc}}$ the corresponding forces, the problem is written:

Find $(\underline{u}, \underline{\xi}) \in \mathcal{I}_{\boldsymbol{\mathcal{V}}} \times \mathcal{I}_{\boldsymbol{\mathcal{V}}_c}$ verifying the initial conditions $\underline{u}\big|_{t=0} = \underline{0}$ such that :

- the generalized displacements $\underline{u}$ coincide with the imposed displacements $\underline{u}_d$ – kinematic constraints ;

$$\forall t \in I, \quad \begin{cases} \boldsymbol{C}_u\Big(\underline{u}(t)\Big) = \underline{u}_d(t) \\ \boldsymbol{C}_v\Big(\underline{u}(t)\Big) = \underline{v}(t) \end{cases} \tag{1}$$

- the external forces $\underline{f}^{\text{ext}}$ counterbalance the internal forces $\underline{f}^{\text{int}}$ – equilibrium equations – to which the contact forces $\underline{f}^{\text{ctc}}$ may be added ;

$$\forall t \in I, \quad \begin{cases} \underline{f}^{\text{int}}(t, \underline{u}(t)) - \underline{f}^{\text{ext}}(t) + \underline{f}^{\text{ctc}}(t, \underline{\xi}(t)) = \underline{0} \\ \underline{f}^{\text{ctc}}(t, \underline{\xi}(t)) = \boldsymbol{C}_{\xi}\Big(\underline{\xi}(t)\Big) \end{cases} \tag{2}$$

- the generalized internal forces $\underline{f}^{\text{int}}$ and the contact forces $\underline{f}^{\text{ctc}}$ result respectively from the integration of the non-linear material and contact laws – behavioural and contact relationship.

$$\forall t \in I, \quad \begin{cases} \underline{f}^{\text{int}}(t, \underline{u}(t)) = \boldsymbol{\mathcal{A}}_u\Big(t, \underline{u}(\tau \leqslant t)\Big) \\ \boldsymbol{\mathcal{A}}_c\Big(t, \underline{v}(t), \underline{\xi}(t)\Big) = \underline{0} \end{cases} \tag{3}$$

The operators $\boldsymbol{C}_u$, $\boldsymbol{C}_v$ and $\boldsymbol{C}_{\xi}$ are linear, while the non-linear operators $\boldsymbol{\mathcal{A}}_u$ and $\boldsymbol{\mathcal{A}}_c$ account for behavioural and contact relations respectively. This type of problem is commonly solved many times according to certain parameters $\underline{\mu} \in \boldsymbol{\mathcal{D}}$ during the optimization processes. These parameters can be of different natures: loads, geometry variations, material data, etc. Classically, the resolution is orchestrated by a Newton-Raphson algorithm in industrial finite element software. This work proposes to use another scheme to linearise the equations by applying the non-linear solver LATIN, which is suitable for the implementation of the PGD technique.

DOI: 10.13009/EUCASS2022-7310

2.2 A weakly-invasive version of LATIN-PGD method

The weakly-invasive proposed version of the LATIN-PGD method differs from the version most commonly used in recent years, which exploits the internal variable description of materials.[13] It can rather be seen as an extension of the early functional versions. Still based on the separation of difficulties, Equations (1 – 3) are divided into two groups: a first one (A_d) gathers all the linear equations, possibly global in space – Equations (1) and (2) –, while a second group (Γ) concentrates all the non-linear but local equations – Equation (3). Then, by means of two search directions, we alternately construct an approximation in each of these two groups (A_d) and (Γ) until we reach convergence at the intersection $\boldsymbol{s}_\chi \in (A_d) \cap (\Gamma)$ (see Figure 1).

Local stage $(A_d) \rightarrow (\Gamma)$
Assuming the quantities $(\underline{u}_\ell, \underline{\xi}_\ell)$ originate from the previous linear stage are known, one seeks a new approximation of $\widehat{\boldsymbol{s}}_\ell$.To this end, a vertical search direction is employed ($\widehat{\underline{u}}_\ell = \underline{u}_\ell$). This choice allows us to naturally decouple primal and dual quantities, which is as close as possible to what is usually achieved in general-purpose industrial finite element software following a Newton-Raphson scheme. In the presence of contacts, we complete the search direction by another linear equation linking local quantities :

$$\forall t \in I, \quad \widehat{\underline{d}}_\ell(t) - \underline{d}_\ell(t) = k\,\underline{\xi}_\ell(t) - k\,\widehat{\underline{\xi}}_\ell(t) \tag{4}$$

where rather than working on the trace of the generalized displacements $\widehat{\underline{v}}_\ell$ on the contact zone, one prefers instead to manipulate the contact distance $\widehat{\underline{d}}_\ell(\widehat{\underline{v}}_\ell(t))$ and k reprensents a scalar scaling factor given by the industrial software. Finally, the non-linear material or contact laws (Equation (3)) at the level of each element completes the system equations to solve.

Linear stage $(\Gamma) \rightarrow (A_d)$
Seeking now a new approximation of the quantities $(\underline{u}_{\ell+1}, \underline{\xi}_{\ell+1})$, we choose for the second search direction (5) an operator $\mathbf{H}$ such as:

$$\mathbf{H}\,(\underline{u}_{\ell+1}(t) - \widehat{\underline{u}}_\ell(t)) = \underline{f}^{\,\mathrm{int}}_{\ell+1}(t) - \widehat{\underline{f}}^{\,\mathrm{int}}_\ell(t) \tag{5}$$

This operator $\mathbf{H}$ constitutes one of the major parameters of the method. Its choice directly influences the speed of convergence of the algorithm. This point is specified in the illustrations. In addition, in the presence of contacts, we complete the formulation by another search direction similar to the one used in the local step:

$$\forall t \in I, \quad k\,\underline{\xi}_{\ell+1}(t) - k\,\widehat{\underline{\xi}}_\ell(t) = \underline{d}_{\ell+1}(t) - \widehat{\underline{d}}_\ell(t) \tag{6}$$

As all the equations involved at this stage are linear, the resolution is carried out by PGD which boils down to determining a basis of m modes (spatial modes $\{\boldsymbol{\Lambda}_i\}_{1\leqslant i\leqslant m}$ and temporal modes $\{\lambda_i\}_{1\leqslant i\leqslant m}$) by solving a minimisation problem.

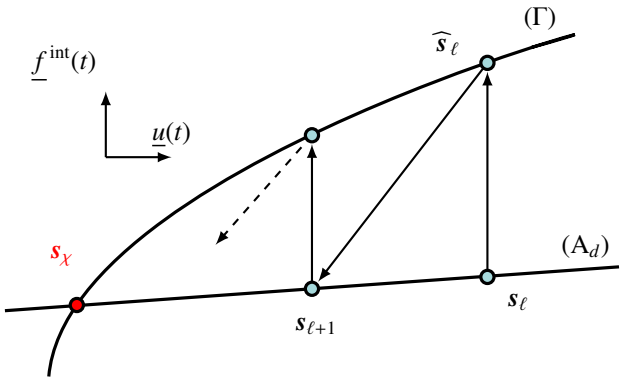

Figure 1: Convergent iterative scheme LATIN-PGD method with $\boldsymbol{s}_\ell = (\underline{u}_\ell, \underline{f}^{\,\mathrm{int}}_\ell, \underline{\xi}_\ell, \underline{v}_\ell)$.

The non-linear operators $\mathcal{A}_u$ and $\mathcal{A}_c$ do not appear naturally in the industrial codes because they require a knowledge of the whole time interval at each iteration ℓ of the algorithm. These operators are thus built in a weakly-invasive way natively in the industrial software during the local stage. This constitutes the only intrusive point of our approach. By weakly-invasive, we mean minimising the number of modifications to be made to the industrial software,

DOI: 10.13009/EUCASS2022-7310

one of the particularities of our approach being precisely to have access to the source code which is not common. We exploit this advantage to satisfy the expectations in terms of performance and robustness, while keeping the operations that can remain independent as much as possible.

2.3 Parameter handling

After discretization of the parameter space $\mathcal{D}$ according to a regular grid, a solution in each of the grid's points is achieved by applying the weakly-invasive LATIN-PGD methodology. In particular, we exploit the non-intrusive techniques developed in.[18] This proves to be inexpensive by using one of the remarkable properties of the LATIN-PGD algorithm, i.e. being able to be initialised from any previously computed solution. More precisely, let us suppose that we want to calculate a new point $(\underline{\mu} + \Delta\underline{\mu}) \in \mathcal{D}$ of the parametric space. Knowing the spatio-temporal solution $\boldsymbol{s}^{(\underline{\mu})}$ for the previous $\underline{\mu} \in \mathcal{D}$ parameter set, as well as the corresponding $\{\boldsymbol{\Lambda}_i\}_{1\leqslant i\leqslant m^{(\underline{\mu})}}$ spatial modes, the first iteration of the LATIN-PGD method proceeds as follows:

- **Local stage** : given $\boldsymbol{s}_0^{(\underline{\mu})}(t)$, one looks for $\widehat{\boldsymbol{s}}_0^{(\underline{\mu}+\Delta\underline{\mu})}(t)$ such that :

$$\underline{\boldsymbol{u}}_0^{(\underline{\mu}+\Delta\underline{\mu})}(t) = \underline{\boldsymbol{u}}_0^{(\underline{\mu})}(t) \qquad \forall t \in I \tag{7}$$

- **Linear stage** : given $\widehat{\boldsymbol{s}}_0^{(\underline{\mu}+\Delta\underline{\mu})}(t)$, one seeks $\boldsymbol{s}_1^{(\underline{\mu}+\Delta\underline{\mu})}(t)$ where from the preliminary step of updating the time functions $\{\lambda_i^{(\underline{\mu})}\}_{1\leqslant i\leqslant m^{(\underline{\mu})}}$, we determine the new approximation :

$$\underline{\boldsymbol{u}}_1^{(\underline{\mu}+\Delta\underline{\mu})}(t) = \sum_{i=1}^{m^{(\underline{\mu})}} \lambda_i^{(\underline{\mu}+\Delta\underline{\mu})}(t)\boldsymbol{\Lambda}_i \tag{8}$$

then the algorithm continues classically with $\ell \geqslant 1$ as detailed previously. For many parameter values, adding new modes to the already constituted reduced basis is not necessary. After a few iterations, convergence is reached, and the algorithm stops to start the management of the next point in the parameter space $\mathcal{D}$, which represents a guarantee of performance, as we will see in the illustrations.

2.4 Implementation constraints

In an industrial context, robustness is just as fundamental as performance. Implementing the LATIN-PGD method in the industrial software Simcenter Samcef™must remain compatible with the existing architecture. One particular wish is to use the same certification steps when building each new version. These steps ensure that adding the LATIN-PGD methodology introduces no adverse effects. Specific attention has been paid to storing data in separate variable formats, which is not standard in incremental commercial software. In the form of a Fortran module, an interface dedicated to the method has been developed to keep all the ingredients related to the PGD technique independent. All information is transferred via this interface, which provides the link between the pre-existing industrial code and the new capabilities offered by the developed LATIN-PGD methodology. The main strength of this weakly-invasive approach lies in its possibility to retain all the richness of the industrial software in the diverse cases it is likely to handle. All the intelligence stored over many years can be used without modification when carrying out the local stage: all types of elements, all kinds of non-linearities, all sorts of loading, etc.

3. Results on industrial test-cases

3.1 First illustration: blade of an aircraft engine

The first test-case concerns a high-pressure blade of an aircraft engine. The mesh, consisting of 5 million degrees of freedom, was created using Simcenter 3D™software. Loadings are due to lateral forces at the top of the blade, centrifugal forces and prescribed displacements at the bottom of the fir stand. The material used follows a Chaboche elasto-visco-plastic law with 9 parameters that depend on temperature.

As a parameter of the LATIN-PGD method, several search directions characterised by various choices of $\mathbf{H}$ operator at the global stage have been studied: either the use of a time-dependent tangent operator $\mathbf{H}_\ell(t)$ or the conservation of the elastic rigidity $\mathbf{K}$ – constant and independent of time. The first one presents an optimal convergence in terms of iteration numbers ℓ while the second one leads to better performance. In the context of moderate displacements, for elasto-visco-plastic problems, with potential contacts, all our tests have given a large superiority to using an elastic

DOI: 10.13009/EUCASS2022-7310

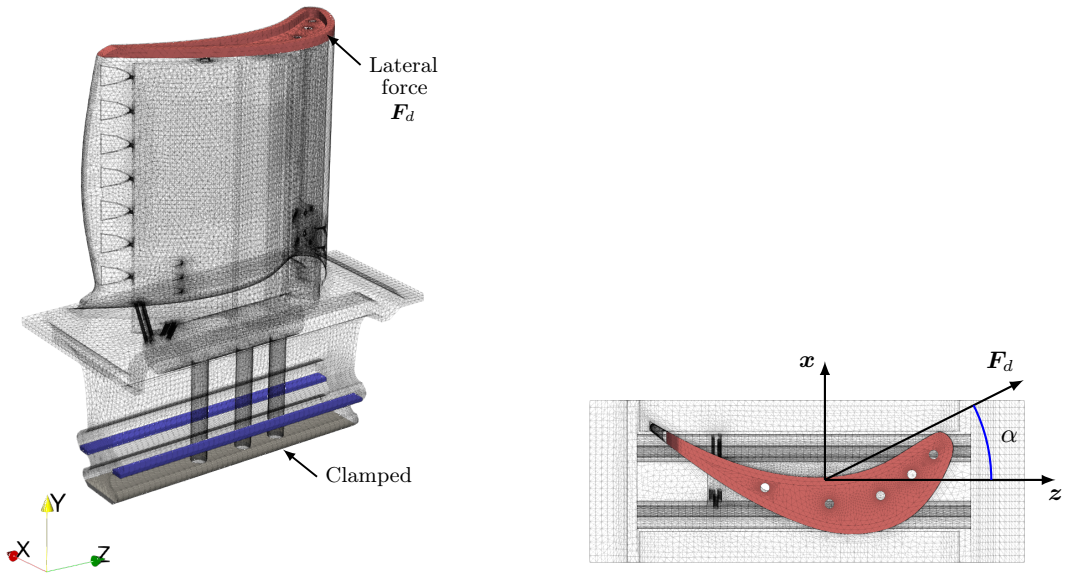

Figure 2: Geometry and loadings applied to the reactor blade

stiffness **K** as soon as the considered problem becomes large. One of the reasons is that we can then take advantage of the factorisation of the operator across the LATIN iterations when a direct solver is used. Table 1 compares the computation times for the realisation of a single random point in the parameter space. Figure 3 details the distribution of the computation times using the constant **K** operator, where we note the important weight of the local step. Hyper-reduction technics[4,7] are being looked at to reduce this effect and therefore increase performance even more.

Table 1: Comparison of the computation times and the number of iterations according to two search directions: tangent direction $\mathbf{H}_\ell(t)$ and elastic direction **K** (for the computation of a point in the parameter space $\mathcal{D}$)

	Computation time [s]								**Number of iterations**
	Local stage		*Global stage*		*Additional*		*Total*		-
$\mathbf{H}_\ell$(t)	1324	(32.9%)	2373	(59.0%)	326	(8.1%)	4023	(100%)	13
K	2206	(73.9%)	354	(11.9%)	425	(14.2%)	2985	(100%)	26

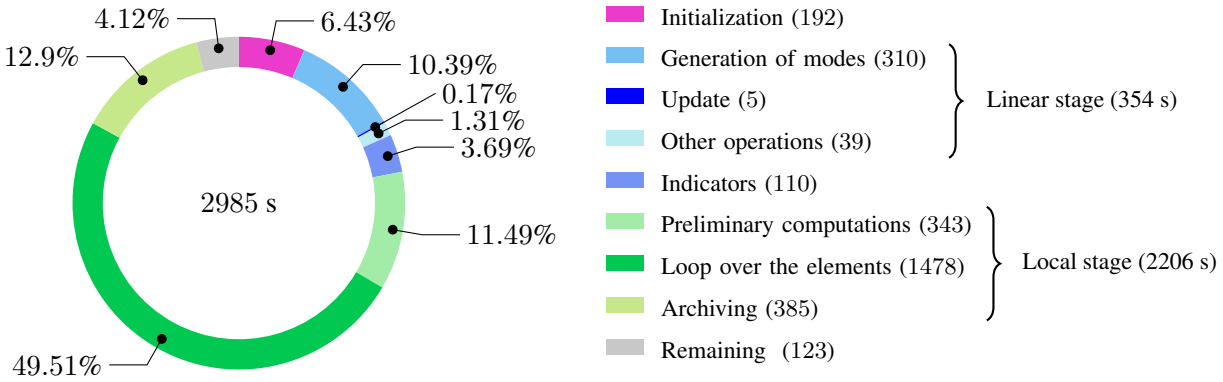

Figure 3: Distribution of calculation times for the constant **K** operator (in seconds)

A parametric study is also conducted on a two-dimensional space $\mathcal{D} = \mathcal{D}_\alpha \times \mathcal{D}_T$ where the loading angle $\alpha \in \mathcal{D}_\alpha = [45, 170]$ in degrees (cf. Figure 2) and the temperature $T \in \mathcal{D}_T = [850, 1010]$ in degrees Celsius represent the two model parameters. These parameters have an influence on the two groups (A_d) and (Γ): each new value of the temperature T affects the 9 material parameters and thus mainly affects (Γ) while each new value of the loading angle α impacts the external forces involved in (A_d). The construction of the reduced model involves the calculation of

DOI: 10.13009/EUCASS2022-7310

255 points uniformly distributed in the $\mathcal{D}$ parametric space. The application of the multi-parametric strategy explained above allows reaching an acceleration factor of more than 3 times compared to the classical use of the commercial software (18 days with the commercial version following a Newton-Raphson algorithm against only 6 days with the weakly-invasive LATIN-PGD strategy we have deployed natively in the industrial software Simcenter Samcef™). This gain would undoubtedly be more important with more points or more parameters.

3.2 Second illustration: blade-disk assembly

The second illustrative test-case focuses on the study of a gas turbine blade-disk assembly involving material non-linearities (plasticity and creep) and contact non-linearities at the junction between the disk and the blade. Hence, one additional particularity of this case compared to the previous one is that it takes into account the contact non-linearities. The structure is subject to centrifugal forces as well as pressure forces on the blade surface. In addition, a temperature gradient is applied along the radial coordinate. Cyclic symmetry conditions are prescribed on the side parts. The mesh, composed of volume elements of different natures – tetrahedral or hexahedral – and specific elements dedicated to the interfaces between the different parts of the mesh – to write the corresponding kinematic conditions –, gathers more than one million degrees of freedom. In what follows, a two-parameter study is conducted where the parameter space $\mathcal{D} = \mathcal{D}_E \times \mathcal{D}_{\alpha_{th}}$ consists of the values of the Young's modulus $E \in \mathcal{D}_E \equiv [200, 260]$ MPa and the coefficient of thermal expansion $\alpha_{th} \in \mathcal{D}_{\alpha_{th}} \equiv [1.38 \times 10^{-5}, 1.42 \times 10^{-5}]$ K^{-1} of the blade. Due to the different manufacturing processes, the values of these parameters are assumed to be insufficiently known. A small variation in the nominal value of these coefficients directly affects the value of the maximum Von Mises stress in the contact zone, i.e. in the zone of occurrence of creep phenomena over long periods of time, which may damage the structure faster than expected. Although other parameters may have an influence, we focus only on the parameter space $\mathcal{D} = \mathcal{D}_E \times \mathcal{D}_{\alpha_{th}}$ in this study.

Figure 4 provides the Von Mises stress map obtained for a given point ($E = 220$ MPa and $\alpha_{th} = 14 \times 10^{-6}$ K^{-1}) in the $\mathcal{D}$ parameter space. One can notice the relative error with respect to the reference – issue of the software Simcenter Samcef™in its commercial version. We observe a good agreement of the solutions with a relative error of less than one percent in the zone of interest for 10 PGD modes composing the reduced basis.

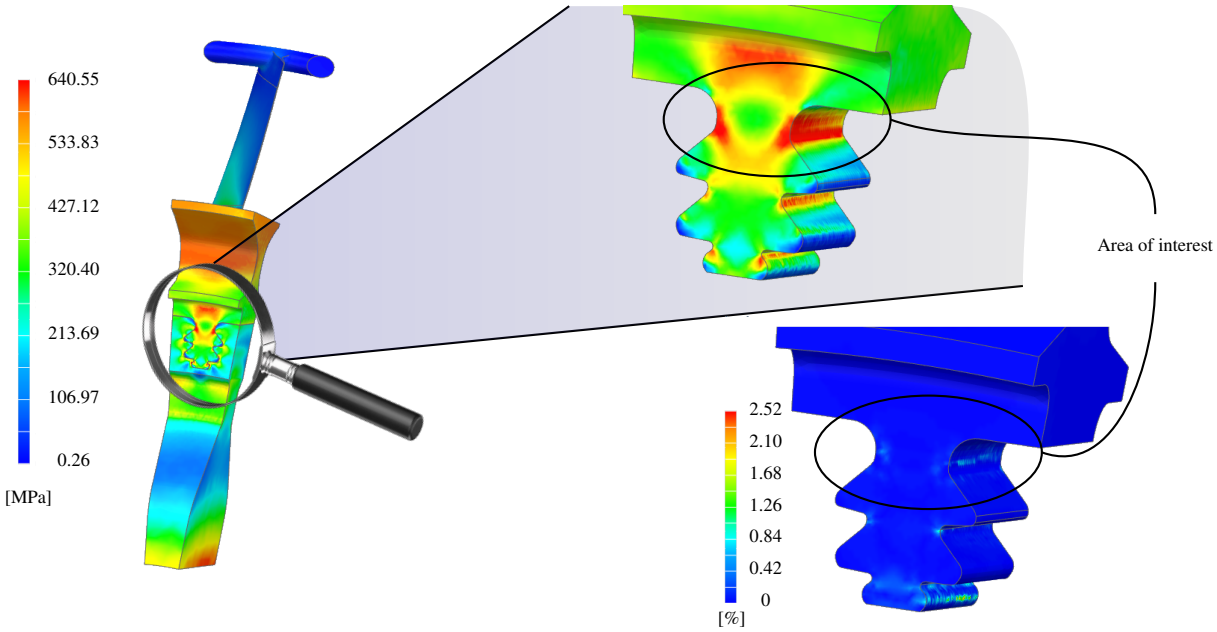

Figure 4: Map of the Von Mises stress field [MPa] and comparison with the reference [%] on the area of interest

The virtual chart of the maximum Von Mises stress according to the parameter space $\mathcal{D}$ in the contact zone is given in Figure 5. This plot highlights a substantial non-linear variation with respect to the Young's modulus, while a linear trend is observed with respect to the second parameter α_{th}. Using the same methodology as before, this chart results from the calculation of 70 points in the parameter space by a sequential and uniform scan of the data. Regarding performance, only 3 hours were necessary to sketch this virtual chart. In contrast, 21 hours would have been required with a classical use of the commercial version of the software Simcenter Samcef™ which leads to a gain of a factor of 7 compared to this reference. This speed-up is due to the restart capability of our LATIN-PGD

DOI: 10.13009/EUCASS2022-7310

solver from any previously converged point close to the parametric space. Previously generated PGD modes are then reused at initialisation to speed-up convergence by providing a better approximation to the solution from the start. In addition, this second test case highlights one of the advantages of having natively incorporated the LATIN-PGD method into one industrial software. All the capabilities offered by the host software can then be easily re-exploited: all types of elements, all types of behaviour laws, all kinds of non-linearities, etc. This test-case includes no less than 3 different non-linear behaviour laws (as a function of temperature) and 6 types of elements with notably non-linear contact elements.

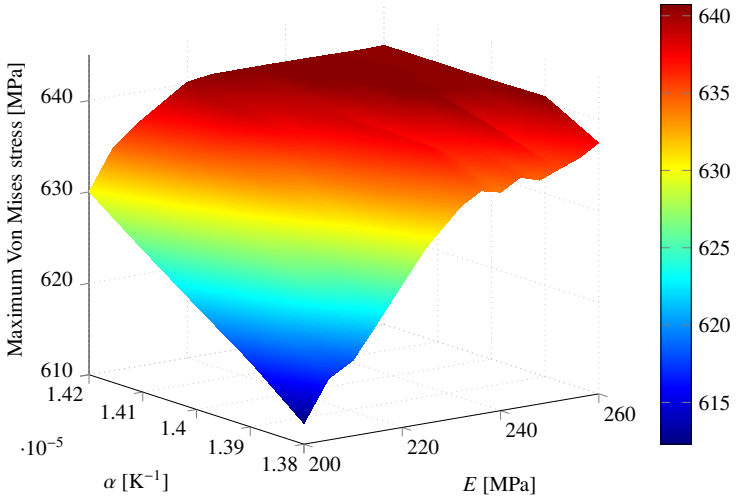

Figure 5: Virtual chart of the maximum Von Mises stress in the contact zone over the entire parameter space $\mathcal{D}$ resulting from 70 points uniformly distributed in the parameter space

4. Conclusions and perspectives

In this work, we have developed a methodology to incorporate reduced-order modelling tools into general-purpose industrial finite element software in a weakly-invasive way. This is the first time that the LATIN-PGD method has been implemented at such a high level in a general-purpose industrial software package, thus achieving attractive gains in terms of performance while allowing the native and efficient construction of reduced-order models. Hyper-reduction strategies are currently being investigated to reduce the weight of the local stage, which would allow even more impressive gains to be achieved. One of the main strengths of the developed approach lies in its robustness: all the pre-existing capabilities of the industrial software, optimised over many years, can be reused without recoding the processing of the various non-linearities already supported. Good results have already been obtained with multiple problems involving material non-linearities, contact phenomena or large rotations. The recognised advantages of reduced models can then be mobilised to solve non-linear mechanical problems in an end-to-end process within an industrial simulation chain. This work can be seen as a first step to building some Digital Twins afterwards: reduced-order models made within Simcenter Samcef™ are likely to provide a quick and reliable evaluation of complex models, describing non-linear physics phenomena, which can then be coupled with real in-service data.

5. Acknowledgments

We are grateful to Siemens Digital Industries Software and the French National Association for Research and Technology (ANRT) for funding this work.

References

[1] Christophe Audouze, Florian De Vuyst, and Prasanth B. Nair. Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations. *Numerical Methods for Partial Differential Equations*, 29(5):1587–1628, 2013.

DOI: 10.13009/EUCASS2022-7310

[2] Peter Benner, Stephano Grivet-Talocia, Alfio Quarteroni, Gianluigi Rozza, Wil Schilders, and Luis M. Silveira, editors. *Model Order Reduction: Snapshot-Based Methods and Algorithms*, volume 2. De Gruyter, Berlin, Boston, 2021.

[3] Domenico Borzacchiello, José V. Aguado, and Francisco Chinesta. Non-intrusive Sparse Subspace Learning for Parametrized Problems. *Archives of Computational Methods in Engineering*, 26(2):303–326, April 2019.

[4] Matteo Capaldo, Pierre-Alain Guidault, David Néron, and Pierre Ladevèze. The Reference Point Method, a "hyperreduction" technique: Application to PGD-based nonlinear model reduction. *Computer Methods in Applied Mechanics and Engineering*, 322:483–514, August 2017.

[5] Fabien Casenave, Asven Gariah, Christian Rey, and Frederic Feyel. A nonintrusive reduced order model for nonlinear transient thermal problems with nonparametrized variability. *Advanced Modeling and Simulation in Engineering Sciences*, 7(1):22, May 2020.

[6] Anindya Chatterjee. An introduction to the proper orthogonal decomposition. *Current Science*, 78(7):808–817, 2000.

[7] Saifon Chaturantabut and Danny C. Sorensen. Nonlinear Model Reduction via Discrete Empirical Interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, January 2010.

[8] Francisco Chinesta and Pierre Ladevèze, editors. *Separated Representations and PGD-based Model Reduction: Fundamentals and Applications*. Number 554 in Courses and Lectures / International Centre for Mechanical Sciences. Springer, Wien, 2014.

[9] Amaury Courard. *PGD-Abaques virtuels pour l'optimisation géométrique des structures*. PhD thesis, Ecole Normale Supérieure de Cachan, Université Paris-Saclay, January 2016.

[10] Elías Cueto, David González, and Icíar Alfaro. *PGD for Non-linear Problems*, pages 39–63. Springer International Publishing, Cham, 2016.

[11] Loïc Giraldi, Dishi Liu, Hermann G. Matthies, and Anthony Nouy. To Be or Not to be Intrusive? The Solution of Parametric and Stochastic Equations—Proper Generalized Decomposition. *SIAM Journal on Scientific Computing*, 37(1):A347–A368, January 2015.

[12] Rubén Ibáñez, Emmanuelle Abisset-Chavanne, Amine Ammar, David González, Elías Cueto, Antonio Huerta, Jean-Louis Duval, and Francisco Chinesta. A Multidimensional Data-Driven Sparse Identification Technique: The Sparse Proper Generalized Decomposition. *Complexity*, 2018.

[13] Pierre Ladevèze. *Nonlinear Computational Structural Mechanics: New Approaches and Non-Incremental Methods of Calculation.* Springer, New York, 1999.

[14] Angel Leon, Sebastien Mueller, Patrick de Luca, Rajab Said, Jean-Louis Duval, and Francisco Chinesta. Non-intrusive proper generalized decomposition involving space and parameters: Application to the mechanical modeling of 3D woven fabrics. *Advanced Modeling and Simulation in Engineering Sciences*, 6(1):1–20, December 2019.

[15] Yvon Maday and Einar M. Rønquist. A reduced-basis element method. *Comptes Rendus Mathematique*, 335(2):195–200, January 2002.

[16] Yvon Maday and Einar M. Rønquist. The Reduced Basis Element Method: Application to a Thermal Fin Problem. *SIAM Journal on Scientific Computing*, 26(1):240–258, January 2004.

[17] Annika Radermacher and Stefanie Reese. POD-based model reduction with empirical interpolation applied to nonlinear elasticity. *International Journal for Numerical Methods in Engineering*, 107(6):477–495, 2016.

[18] Nicolas Relun, David Néron, and Pierre-Alain Boucard. A model reduction technique based on the PGD for elastic-viscoplastic computational analysis. *Computational Mechanics*, 51(1):83–92, January 2013.

[19] Vasileios Tsiolakis, Matteo Giacomini, Ruben Sevilla, Carsten Othmer, and Antonio Huerta. Nonintrusive proper generalised decomposition for parametrised incompressible flow problems in OpenFOAM. *Computer Physics Communications*, 249:107013, April 2020.

[20] X. Zou, M. Conti, P. Díez, and F. Auricchio. A nonintrusive proper generalized decomposition scheme with application in biomechanics. *International Journal for Numerical Methods in Engineering*, 113(2):230–251, 2018.