Advanced Model Order Reduction and Artificial Intelligence Techniques Empowering Advanced Structural Mechanics Simulations. Application to Crash Test Analyses

Victor Limousin[1], Xavier Delgerie[1], Emmanuel Leroy[1],
Rubén Ibáñez[2], Clara Argerich[2], Fatima Daim[3],
Jean Louis Duval[3] & Francisco Chinesta[2,3]

[1] Gestamp Autotech Engineering France
17-19, Rue Jeanne Braconnier, 92360 Meudon, France
vlimousin@fr.gestamp.com
xdelgerie@fr.gestamp.com
eleroy@fr.gestamp.com

[2] ESI Group Chair @ PIMM Laboratory, Arts et Métiers ParisTech
151 Boulevard de l'Hôpital, 75013 Paris, France
rubibpi1@gmail.com
clara.argerich_martin@ensam.eu
Francisco.Chinesta@ensam.eu

[3] ESI Group
Batiment Seville, 3 bis Saarinen, F-50468, Rungis, France
fatima.daim@esi-group.com
jean-louis.duval@esi-group.com
Francisco.Chinesta@esi-group.com

May 25, 2019

This paper proposes a general framework for expressing parametrically quantities of interest related to the solution of complex structural mechanics models, in particular the ones involved in crash analyses where strongly coupled nonlinear and dynamic behaviors coexist with space-time localized mechanisms. Advanced nonlinear regressions able to proceed in the low-data limit, enabling to accommodate heterogeneous parameters, will be proposed and their performances evaluated in the case of crash simulations. As soon as these parametric expressions will be determined, they can be used for generating large amounts of realizations of the quantity of interest for different choices of the parameters, for supporting data-analytics. On the other hand, such parametric representations allow the use advanced optimization techniques, evaluate sensitivities and propagate uncertainty all them under the stringent real-time constraint.

1 Introduction

Plenty of effort has been dedicated throughout history to design an optimization process. It is certain that the primary design of a new component could become a tough task, especially when constraints coming from different fields have to be satisfied. Furthermore, evaluating the improvement of a given attribute under a change in the input parameter may become a tedious task when each evaluation of the direct problem involves either, numerous experimental tests or several high-fidelity simulations.

Thus, design is usually stated as an optimization problem, where a cost function is intended to be minimized using any appropriate technique, as for instance a steepest descent method. The main drawback of such a procedure relies in the necessity of solving many times the problem for each tentative choice of the design parameters.

To enhance design performances Model Order Reduction – MOR – techniques enable faster simulations [7]. On the other hand the construction and use of metamodels (surrogate models) also facilitates the design process because the solution for a given choice of the parameters can be evaluated online in almost real-time. Thus, as soon as the parametric solution of a given problem is available, simulation, optimization, inverse analysis, uncertainty propagation and even control, can be efficiently performed under the stringent real-time constraint [4].

Among the different MOR techniques, Proper Generalized Decomposition – PGD – allows the off-line construction of a parametric solution, as reported in [5]. However, standard PGD constructors induced too invasive computational procedures [6]. To mitigate that issue, we proposed in our recent works some strategies enabling non-intrusive solution procedures, able to compute the parametric solution from some runs of the associated high-fidelity solver. The first proposal was applying a hierarchical approximation and considering as sampling points its associated Gauss-Lobatto-Chebyshev points, leading to the so-called SSL-PGD (for Sparse Subspace Learning Proper Generalized Decomposition) [2].

The main drawback of SSL-PGD-based procedures is the increase of the sampling points with the design space dimensionality. For mitigating that issue, we recently proposed an alternative procedure using sparser sampling, and proved that reasonable results can be obtained with a number of sampling points (runs of the high-fidelity model) scaling with the number of parameters involved in the considered model. This technique was called sPGD (for sparse Proper

Generalized Decomposition) [8] and is being nowadays checked in a panoply of application at ESI Group.

However, many times, more than being interested in parametric fields, the last associated with the solution of parametric models (expressed from partial differential equation), the design requires the evaluation of one (or some) quantity (quantities) of interest –QoI– . In what follows and for the sake of simplicity we will assume a single QoI, explicitly or implicitly dependent on the parametric field itself. In that case a procedure is needed for extracting the parametric form of the QoI. The main specificities of such a procedure are: (i) the fact of dealing with high-dimensional spaces, with as many dimensions as the number of parameters involved in the model; (ii) the fact of considering many parameters of different nature (some parameters could be discrete and even qualitative).

In the present work we are dealing with crash simulations, and more particularly on the parametric deformation of a B-Pillar provided by Gestamp. Some high-fidelity simulations were performed by varying different structural parameters, from which a QoI (the maximum intrusion able to quantify the car occupants safety) was extracted for the different parameter choices. In order to proceed with design optimization, a parametric expression of the QoI seems very valuable, however, for reducing the design cost and complexity, it seems compulsory reducing as much as possible the sampling points, that is, the required runs of the high-fidelity model.

In that sense, in this paper we propose two different methodologies, the first is based on the use of a PGD-based sparse nonlinear regression making use of the separated representations at the heart of the PGD methodologies, and the second that efficiently circumvents the issue related to potential parameters heterogeneity. The last technique was called *Code2Vect* because it proceeds by mapping points in a representative domain into a vector space equipped with a convenable metric able to safely construct parametric approximations.

When dealing with time-dependent QoI, the PGD-based regression was generalized in a multi-time-domain framework, allowing for compact local regressions evolving (with continuity) in time, the so-called multi-local sparse nonlinear PGD-based regression.

As soon as the parametric QoI is available, sensibility analyses become straigth-forward, as well as uncertainty quantification and propagation when parameters are assumed being statistically distributed, by using standard Monte Carlo approaches or by calculating the different QoI statistical moments. In all cases and again thanks to the compact parametric expression of the QoI such analyses can be performed extremely accurately and under the stringent real-time constraint, facilitating the design process. Finally, because extra-data can be generated from the parametric QoI, one could use any available data-analytics procedure for visualizing, classifying or modeling. In the present work MINESET™ by ESI will be used for those purposes.

After this introduction the paper outline is as follows: Section 2 defines the problem to be addressed. Section 3 summarizes the main numerical technologies. Finally Section 4 presents and discusses numerical results, before finishing with some general conclusions and prospects in Section 5.

2 Problem statement

2.1 Context and Glab simulation model presentation

With constant evolution of regulations regarding environment and pollutants emission, carmakers target an optimal vehicle light weighting while fulfilling cost and safety requirements. Given that physical validation tests are long and expensive, car makers consider numerical models as a parallel option. Based on Finite Element theories, these models (FEM) enable a better understanding of physical phenomena and case-and-effect relationships between design parameters, manufacturing processes and crash outputs. The constant improvement in the accuracy and efficiency make them reliable to achieve car manufacturers objectives.

In this context, Gestamp, an international group dedicated to the design, development and manufacturing of automotive components, has developed the G-Lab family (see Fig. 1). It is an R&D program focusing on the development of numerical vehicle prototypes. The objective is to represent different automotive segments (B, C/D, SUV) with several types of powertrain (ICE, PHEV, EV) to validate new concepts and technologies.

Figure 1: G-Lab family

Each G-lab model is an advanced numerical model, dealing with all types of non-linearities: geometries, material, buckling instabilities and multiple contacts. For example, material cards consider high deformation, plasticity and failure with speed dependent properties. To handle these non-linearities and ensure stability, explicit schemes with a very small timestep (close to 3e-4s) are used to solve numerical equations. Crash duration is usually inferior to 100ms. Therefore, for the G3 model (with more than 6,7 million elements and a 4mm mesh size) calculation time is between 10 to 20 hours for a single crash, depending on boundary conditions and regulations.

To achieve mass and performance targets, single or multi-objective optimizations are used. This is an appropriate way to find and define the best solutions to attain the desired targets. The primary objective is to find the optimal concepts, for which one of the essential points is to understand the influence of parameters variation. As the vehicle structure is more and more complex (new materials, technologies, assemblies, ...), the number of parameters that can be optimized increases. Moreover, the level of detail of FEM models, and so the number of elements, is continuously increasing. Therefore, optimization loops become time consuming and their analysis quickly gets tedious.

Thereby, testing all or part of the parameters combinations can be time consuming in a full car simulation. Classic optimizations using Response Surface Method are efficient but do not allow to reduce the number of calculations.

PGD technique has been used to solve this issue, by reducing simulation number and increasing predictability. This technique has been applied on a Body-in-white component (B Pillar) optimization and a sensitivity analysis. The crash selected is a EuroNCAP AEMDB side crash [10], with a 1400kg barrier impacting a static vehicle, at 60kph (see Fig. 2). Results accuracy and prediction will be compared as well as the time spent and ease of analysis.

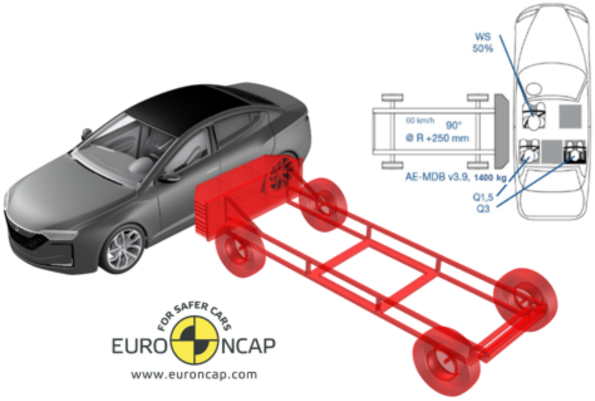

Figure 2: EURONCAP AEMDB Side crash regulation applied on G3 model

2.2 BIW, B-Pillar scope and parametrization

The Body in White (BIW) is the set of sheet metal components that forms the structure of a vehicle. It is the main passive security element that ensures passengers safety during a crash. In case of a side crash, lateral components are the most solicited, particularly the B-Pillar (see Fig. 3). This set of parts, localized between the frontal and rear doors, represents a strategic component for side and rollover crash performances.

B-pillar is generally made of several parts (see Fig. 4):

1. Outer: main structural element of the B-pillar. The use of full press hardening allows to achieve a very high ultimate tensile grade (up to 1500MPa). To control bending and improve energy absorption, it is possible to create a ductile area (UTS between 400 and 800 MPa);

2. Inner: part used to close the B-pillar and to receive surroundings parts. Its contribution to crash performances is low;

3. Reinforcement: additional part to locally reinforce the pillar. To avoid using two different tools for the outer and reinforcement, the component can be reinforced by applying a local patch on the blank before press hardening. Thus, outer and patch are stamped together in only one tool.

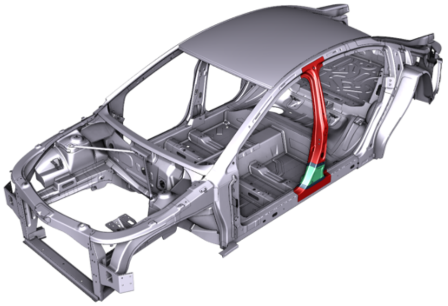

Figure 3: Glab G3 BIW with highlighted BPillar

To improve side crash performances, modifications at different level can be made, such as changing materials, thicknesses, and geometry. The key of light weighting is to apply the right material with the right design at the right place. For this B-pillar optimization study, five parameters have been selected:

1. Outer thickness t_1: Once the outer material is fixed, its thickness is supposed to be the most influential parameter. Increasing this thickness improves performances but adds weight. The parameter range varies between 1,3mm to 2,0mm, with a discrete distribution every 0,05mm;

2. Ductile zone material grade y: By locally controlling the temperature of the tools, partial hardening of the part can be achieved allowing tailored material properties in hot stamped monolithic component. A more ductile zone will absorb more energy but will allow more deformation. For this study, ultimate tensile stress of this zone continuously varies between 400MPa to 600MPa.

3. Ductile zone size z: like the material grade of the ductile zone, its size can also be changed. Two different sizes of soft zone have been included in the design space: 60mm and 90mm width.

4. and 5. Inner part and patch thicknesses t_2 and t_3: Decreasing parts thicknesses is the main way to save mass. Inner thickness varies between 0,9mm to 1,3mm while patch thickness evolves between 1,0mm and 1,6mm every 0,05mm.

The objective of this optimization is to find the best combination of parameters that minimize the mass of the B-pillar while achieving safety targets.

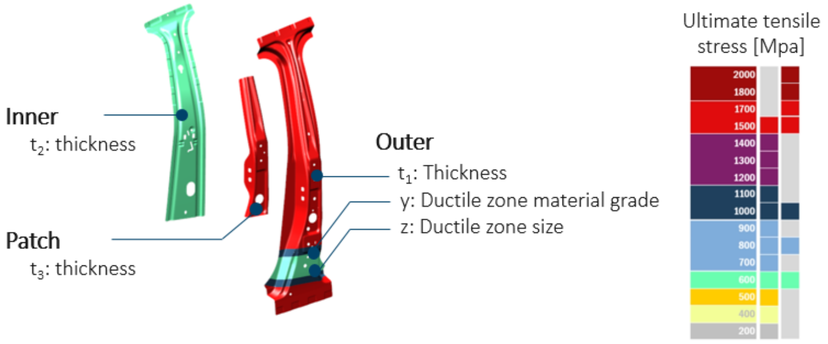

Figure 4: B-pillar parts and parameters with color legend

2.3 Quantity of Interest and expected results (crash analysis)

A considerable amount of kinetic energy is generated by the barrier, which the BIW must absorb to its utmost to protect the passengers. If insufficient, dummies may suffer from internal and external damages. Internal injuries are linked to a too high speed or deceleration change. External injuries appear when intrusions into the cabin cell are too high. A delicate compromise must be found between energy absorption (to reduce deceleration) and cabin cell resistance (to reduce intrusion).

To protect the passengers efficiently, it is important to avoid deformation in the upper area, close to the head. To do so, the B-pillar concept with SoftZone has an upper part in an ultra-high strength steel to limit intrusion. Its lower area displays a soft zone with a lower material grade to localize deformation and absorb a maximum of energy.

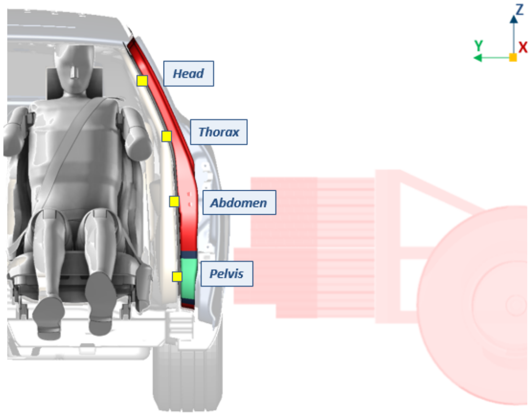

Figure 5: B-pillar measurement points

In this study, the post processing is done on the B-pillar with two main Quantities of Interest (QoI): intrusions and velocities. These QoI have been measured at four specific points, representing the main zones of the dummy to protect: Head, Thorax, Abdomen and Pelvis. Measurement points position are displayed on Fig. 5.

The first QoI is the intrusion, measured as Y-displacement (see Fig. 5) in the car referential. As the vehicle moves during the crash, this local coordinate system allows to analyze B-pillar deformations, as seen by the passengers. Maximum values of the four intrusions are assessed with some specific constraints to respect (see Fig. 6). For example, one constraint is that the maximum value of intrusion on Abdomen (N3) must be lower than 205mm.

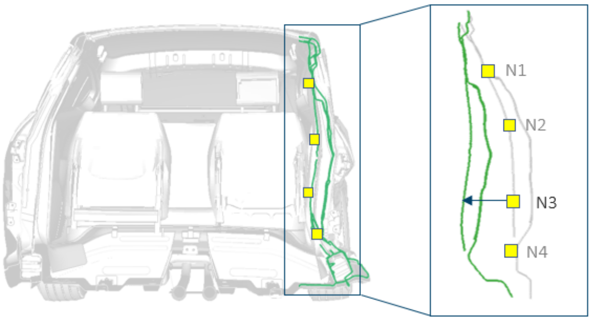

Figure 6: Car deformed view with highlighted B-pillar section and location of points: Head (N1), Thorax (N2), Abdomen (N3) and Pelvis (N4)

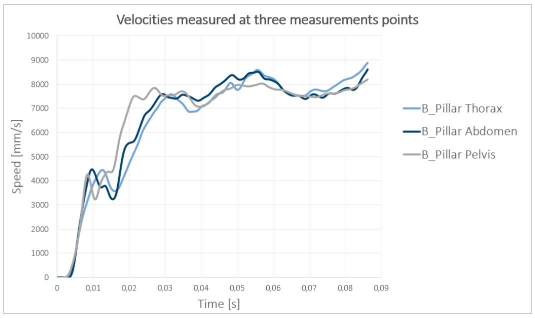

Figure 7: Measured velocities at three measurements points

The second QoI are the velocities, measured as Y-velocities in the global coordinate system. The three lower measurement points are considered: thorax, abdomen and pelvis. A filter has been applied to smooth curves and reduce numerical noise. Contrary to the intrusion, on which only the maximum value is considered, the temporal evolution of the velocities is studied (see Fig. 7). Alongside maximum value, the average velocity is assessed on several time zones.

A high accuracy level of the QoI is needed for a parametric model to be used. For the intrusion, the maximum tolerated error for the PGD model is 5mm. Concerning velocities, maximum tolerated accuracy is an average error rate inferior to 2,5%.

3 Methods

When considering a QoI depending on a set of $\mathtt{M}$ parameters $\mu_1, \cdots, \mu_{\mathtt{M}}$, defining vector $\boldsymbol{\mu}$, $\boldsymbol{\mu} = (\mu_1, \cdots, \mu_{\mathtt{M}})$, i.e. $\mathcal{O}(\boldsymbol{\mu})$, its explicit form is a valuable tool in design optimization. However, in general QoI depends on the structural model solution.

The parametrized model solution is expressed into a separated form by making use of the SSL-PGD or the sPGD solvers, it reads:

$$u(\boldsymbol{x}, t, \mu_1, \cdots, \mu_{\mathtt{M}}) = \sum_{i=1}^{\mathtt{N}} X_i(\boldsymbol{x}) T_i(t) M_i^1(\mu_1) \cdots M_i^{\mathtt{M}}(\mu_{\mathtt{M}}). \tag{1}$$

Sometimes the quantity of interest can be directly and explicitly extracted from Eq. (1), inheriting its separated form. Imagine for a while, that one is interested by the final value of the field at a certain point $\boldsymbol{P}$ and time Θ. In that case the parametric QoI reads:

$$\mathcal{O}(\boldsymbol{\mu}) = u(\boldsymbol{P}, \Theta, \mu_1, \cdots, \mu_{\mathtt{M}}) = \sum_{i=1}^{\mathtt{N}} X_i(\boldsymbol{P}) T_i(\Theta) M_i^1(\mu_1) \cdots M_i^{\mathtt{M}}(\mu_{\mathtt{M}})$$

$$= \sum_{i=1}^{\mathtt{N}} \alpha_i M_i^1(\mu_1) \cdots M_i^{\mathtt{M}}(\mu_{\mathtt{M}}). \tag{2}$$

However, in most of cases when dealing with complex quantities of interest such an extraction becomes difficult to perform. The simplest alternative consists of evaluating the QoI for different choices of the parameters, i.e. $\mathcal{O}(\boldsymbol{\mu}^j) = \mathcal{O}(\mu_1^j, \mu_2^j, \cdots, \mu_{\mathtt{M}}^j) \equiv \mathcal{O}^j$, $j = 1, \cdots, \mathtt{S}$, and then from those values infer the parametric form of the QoI, $\mathcal{O}(\boldsymbol{\mu})$, by enforcing the best fitting of the data $\mathcal{O}^j$, $j = 1, \cdots, \mathtt{S}$. In what follows we assume being operating in the last scenario.

The main difficulties related to the construction of the parametric expression $\mathcal{O}(\boldsymbol{\mu})$, by assimilating the available data $\mathcal{O}^j$, are multiple:

- The choice of the parameters. How to be sure that the hidden parameters affecting the output were considered?

- In some cases the list of parameters, a priori selected, is too large, and many of them have not a significant influence on the output;

- In many cases some parameters are exhibiting correlations, that means that the list of explicative uncorrelated parameters is more reduced that the initially considered. Here nonlinear dimensionality reduction techniques (manifold learning in particular) can help to extract the dimensionality of the slow manifolds;

- In some cases the parameters do not act individually, but in a combined manner. Imagine for a while the Euler theory of buckling. For a beam of length L with a rectangular cross section $b \times h$ (b being the width and h its height), the buckling critical force depends on bh^3/L^2. Thus, a polynomial regression expressing the buckling critical load from the geometrical parameters b, h and L needs the appropriate richness (third order in h) and the sufficient number of terms for representing L^{-2} from a polynomial expansion.

Most of the last points are object of intense researches, and today no definitive answer exist for most of them in the more general settings.

In any case, as soon as the different parameters composing the entries of vector $\boldsymbol{\mu}$ are assumed able to express the output, regressions based on decision trees or its random forest counterpart, neural networks at the heart of deep learning, whose main drawback is the amount of data needed in the training stage, dynamic mode decomposition [9], sparse identification [3], or usual linear and nonlinear regressions, to cite few, can be applied.

A first choice consists of using classical regression strategies. In that case one could consider a polynomial dependence of the QoI, $\mathcal{O}$, on the parameters μ_k, $k = 1, \cdots, \mathtt{M}$. The simplest choice, linear regression, reads

$$\mathcal{O}(\boldsymbol{\mu}) = \beta_0 + \beta_1 \mu_1 + \cdots + \beta_{\mathtt{M}} \mu_{\mathtt{M}}, \tag{3}$$

where the $\mathtt{M}+1$ coefficients β_k can be computed from the available data. If $1+\mathtt{M}$ data are available, $\mathcal{O}^j$, $j = 1, \cdots, 1+\mathtt{M}$, we can write the matrix system

$$\begin{pmatrix} \mathcal{O}^1 \\ \mathcal{O}^2 \\ \vdots \\ \mathcal{O}^{\mathtt{M}+1} \end{pmatrix} = \begin{pmatrix} 1 & \mu_1^1 & \mu_2^1 & \cdots & \mu_{\mathtt{M}}^1 \\ 1 & \mu_1^2 & \mu_2^2 & \cdots & \mu_{\mathtt{M}}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \mu_1^{\mathtt{M}+1} & \mu_2^{\mathtt{M}+1} & \cdots & \mu_{\mathtt{M}}^{\mathtt{M}+1} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{\mathtt{M}} \end{pmatrix}, \tag{4}$$

that allows calculating coefficients β_k and from them the linear regression (3).

When the number of available data is smaller or larger than $\mathtt{M}+1$, the previous systems results under or over-determined respectively. Different techniques exist for solving them: pseudo-inverse, L2 or L1 optimization, the last intimately related to compressed sensing, or the usual Matlab$^{\text{TM}}$ or Scilab$^{\text{TM}}$ backslash.

Linear regression is simple to use because it requires a reasonable amount of available data, of the same order than the number of the parameters involved in the approximation, however, in some case the approximation becomes too poor for representing rich nonlinear behaviors.

Higher degree approximations (nonlinear regressions) are possible without major difficulties when the number of involved parameters remain small enough. For instance the quadratic approximation reads

$$\mathcal{O}(\boldsymbol{\mu}) = \beta_0 + \sum_{i=1}^{\mathtt{M}} \beta_i \mu_i + \sum_{i=1}^{\mathtt{M}} \sum_{j \geq i}^{\mathtt{M}} \beta_{ij} \mu_i \mu_j, \tag{5}$$

where it can be noticed that the number of coefficients (and consequently the required data) roughly scales with $\mathtt{M}^D$, where D is the approximation degree.

Thus, if $\mathtt{M}$ remains reasonably small, one could expect increasing the approximation degree D, however the multi-parametric case seems privileging linear regressions. The crucial question is: can be the multi-parametric case compatible with high-degree approximations while keeping as reduced as possible the sampling?

A response to that question is provided in the next section (Section 3.1) that proposes a multi-local sparse nonlinear PGD-based regression. Even if such a proposal enables rich approximations in multi-parametric settings, one difficulty persists, the one related to the use of L2 norm that determines its own choice of the over-determined approximation. At present, a work in progress consists in extending the procedure presented in the next section, by using the L1 norm enabling sparser approximations.

When the parameters are of very different nature, the definition of metrics in the parametric space becomes a tricky issue. Separated representations circumvent that issue when using an alternated directions constructor that allows avoiding parameters mixing. In Section 3.2 we propose an alternative procedure able to circumvent that issue in a very general setting.

Finally, as soon as the parametric expression of the QoI is available, it allows generating as many parametric particularizations as desired, making possible performing efficiently data-analytics, sensitivity analyses and uncertainty propagation, as addressed in Section 3.3.

3.1 Sparse PGD: a nonlinear regression processing at the low-data limit

Sparse PGD regression consists in defining a sparse approximation in high dimensional settings [8] revisited for the sake of completness in what follows. For the ease of exposition and without loss of generality, let us begin by assuming that the QoI lives in $\mathbb{R}^2$, $\mathcal{O}(\mu_1, \mu_2)$, $\boldsymbol{\mu} = (\mu_1, \mu_2) \in \Omega \subset \mathbb{R}^2$, and that it is to be recovered from sparse data $\mathcal{O}^j$. For that purpose we consider the Galerkin projection for calculating the approximate $\tilde{\mathcal{O}}(\boldsymbol{\mu})$ of $\mathcal{O}(\boldsymbol{\mu})$:

$$\int_\Omega w(\boldsymbol{\mu}) \left(\tilde{\mathcal{O}}(\boldsymbol{\mu}) - \mathcal{O} \right) d\boldsymbol{\mu} = 0, \tag{6}$$

where $w(\boldsymbol{\mu}) \in \mathcal{C}^0(\Omega)$ is an arbitrary test function and

$$\mathcal{O} = \sum_{j=1}^{\mathtt{S}} \mathcal{O}^j. \tag{7}$$

Following the Proper Generalized Decomposition (PGD) rationale, the next step is to express the approximated function $\tilde{\mathcal{O}}$ in the separated form

$$\tilde{\mathcal{O}}(\boldsymbol{\mu}) \approx \sum_{k=1}^{\mathtt{N}} M_i^1(\mu_1) M_i^2(\mu_2), \tag{8}$$

constructed by using the standard rank-one update [6].

It is worth noting that the product of the test function $w(\boldsymbol{\mu})$ times the objective function $\mathcal{O}(\boldsymbol{\mu})$ is only evaluated at few locations (the ones corresponding to the available sampled data). Since information is just known at these $\mathtt{S}$ sampling points $\boldsymbol{\mu}^j$, $j = 1, \cdots, \mathtt{S}$, it seems reasonable to express the test function not in a finite element context, but to express it as a set of Dirac delta functions collocated at the sampling points,

$$w(\boldsymbol{\mu}) = \tilde{\mathcal{O}}^*(\boldsymbol{\mu}) \sum_{j=1}^{\mathtt{S}} \delta(\boldsymbol{\mu}^j). \tag{9}$$

In the expressions above nothing has been specified about the basis in which each one of the one-dimensional modes was expressed. An appealing choice ensuring accuracy and avoiding spurious oscillations consists of using interpolants based on Kriging techniques.

The just described procedure defines a powerful nonlinear regression. It is important to note that when calculating functions $M_i^j(\mu_j)$ (that defines a one-dimensional problem in the coordinate μ_j) $\mathtt{S}$ data-points are available. Thus, $\mathtt{S}$ points enables quite rich approximations in each parametric dimension. The only drawback, as commented at the beginning of the present section, is the use of the L2-norm in the Galerkin projection (6) leading to a particular solution of the under-determined problem, among the infinity of possible solutions. The use of a L1 minimization should lead to sparser approximations. This last route constitutes a work in progress.

3.2 Code2Vect

Defining distances between qualitative data has no sense, and usual learning approaches suffers of such an illness. For instance, one could consider that yellow and red are quite close because both are representing colors, however, from both words such a proximity becomes difficult to quantify.

In what follows we propose a technique, sketched in Fig. 8, for mapping points in a representation space into a target space equipped of an euclidean metric allowing quantifying distances, crucial for constructing approximations.

We assume that points in the origin space (space of representation) consist of $\mathtt{S}$ arrays composed on $\mathtt{M}$ entries, noted by $\boldsymbol{\mu}^j$. Their images in the vector space are noted by $\mathbf{x}^j \in \mathbb{R}^{\mathtt{D}}$. That vector space is equipped with the standard scalar product and the associated Euclidian distance. The mapping is described by the $\mathtt{D} \times \mathtt{M}$ matrix $\mathbf{W}$,

$$\mathbf{x} = \mathbf{W}\boldsymbol{\mu}, \tag{10}$$

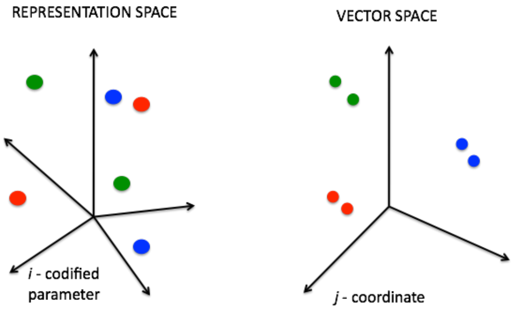

Figure 8: Input space (left) and target vector space (right).

where both, the components of $\mathbf{W}$ and the images $\mathbf{x}^j \in \mathbb{R}^{\mathtt{D}}$ of $\boldsymbol{\mu}^j$, $j = 1, \cdots, \mathtt{S}$, must be calculated.

Each point $\mathbf{x}^j$ keep the label (value of the output of interest) associated with is origin point $\boldsymbol{\mu}^j$, denoted by $\mathcal{O}^j$.

We would like placing points $\mathbf{x}^j$, such that the Euclidian distance with each other point $\mathbf{x}^i$ scales with their outputs difference, i.e.

$$(\mathbf{W}\boldsymbol{\mu}^i - \mathbf{W}\boldsymbol{\mu}^j) \cdot (\mathbf{W}\boldsymbol{\mu}^i - \mathbf{W}\boldsymbol{\mu}^j) = \|\mathbf{x}^i - \mathbf{x}^j\|^2 = |\mathcal{O}^i - \mathcal{O}^j|, \tag{11}$$

where the coordinates of one of the points can be arbitrarily chosen.

Thus, there are $\frac{\mathtt{S}^2}{2} - \mathtt{S}$ relations to determine the $\mathtt{M} \times \mathtt{D}$ unknowns (the components of $\mathbf{W}$).

Linear mappings are limited and do not allow proceeding in nonlinear settings. Thus, a better choice consists of the nonlinear mapping $\mathbf{W}(\boldsymbol{\mu})$ suitably approximated [1].

3.3 Sensitivities and uncertainty propagation

With the QoI expressed parametrically,

$$\mathcal{O}(\boldsymbol{\mu}) \approx \sum_{i=1}^{\mathtt{N}} M_i^1(\mu_1) \cdots M_i^{\mathtt{M}}(\mu_{\mathtt{M}}), \tag{12}$$

sensitivity of the output to a given parameter, e.g. to μ_1 reads

$$\frac{\partial \mathcal{O}(\boldsymbol{\mu})}{\partial \mu_1} \approx \sum_{i=1}^{\mathtt{N}} \frac{\partial M_i^1(\mu_1)}{\partial \mu_1} M_i^2(\mu_2) \cdots M_i^{\mathtt{M}}(\mu_{\mathtt{M}}). \tag{13}$$

Now, if parameters are totally uncorrelated the probability distribution of all them becomes independent and then the probability density function can be expressed as

$$\Xi(\mu_1, \cdots, \mu_{\mathtt{M}}) = \xi_1(\mu_1) \cdots \xi_{\mathtt{P}}(\mu_{\mathtt{M}}). \tag{14}$$

When correlations cannot be totally avoided, we can express the joint probability density $\Xi(\mu_1, \cdots, \mu_{\mathtt{M}})$ in a separated form (by invoking the SSL, the sPGD or even the HOSVD):

$$\Xi(\mu_1, \cdots, \mu_{\mathtt{M}}) \approx \sum_{i=1}^{\mathtt{R}} F_i^1(\mu_1) \cdots F_i^{\mathtt{M}}(\mu_{\mathtt{M}}), \tag{15}$$

Now, with both the output and joint probability density expressed in a separated form, the calculation of the different statistical moments becomes straightforward. Thus, the first moment, the average field results

$$\overline{\mathcal{O}} = \int_{\Omega_1 \times \cdots \times \Omega_{\mathtt{M}}} \mathcal{O}(\mu_1, \cdots, \mu_{\mathtt{M}}) \; \Xi(\mu_1, \cdots, \mu_{\mathtt{M}}) \; d\mu_1 \cdots d\mu_{\mathtt{M}}, \tag{16}$$

where Ω_k defined the domain of parameter μ_k. The separated representation is a key point for the efficient evaluation of this multidimensional integral, that becomes a series of one dimensional integrals.

The calculation of higher order statistical moments (variance, ...) requires the pre-calculation of the output powers $\mathcal{O}^s$, $s > 1$ within the SSL or sPGD frameworks to define a separated representation to be introduced in the calculation of the s-statistical moment:

$$\int_{\Omega_1 \times \cdots \times \Omega_{\mathtt{M}}} \mathcal{O}^s(\mu_1, \cdots, \mu_{\mathtt{M}}) \; \Xi(\mu_1, \cdots, \mu_{\mathtt{M}}) \; d\mu_1 \cdots d\mu_{\mathtt{M}}. \tag{17}$$

An alternative procedure for propagating uncertainty consists in using Monte Carlo techniques. The parametric QoI allows particularizing it in almost real-time to any choice of the parameters, according to their probability distribution, in order to evaluate the QoI probability distribution. These simple and cheap particularizations can be also used for performing a variety of data-analytics.

4 Numerical results

The numerical results are structured as follows: first, the performance to create a surrogate model of the maximum intrusion displacement is shown. Afterwards, the sPGD will be employed to capture not only the maximum intrusion displacement but also the temporal evolution of the intrusion. In the last application of the sPGD, a surrogate model regarding the temporal evolution of the intrusion velocity will be created as well. Finally, the *Code2Vect* will be used for two purposes, first to find a representation space where all the intrusion data is displayed; second, to study as well its prediction capability for calculating the maximum intrusion value.

4.1 Maximum intrusion from sPGD

Our parameter space consists of five independent parameters, namely z, t_2, t_1, t_3 and y. The Design of Experiments (DoE) consists of 22 high-fidelity simulations

Simulation	z	t_2	t_1	t_3	y
1	108	1.3	1.5	1.5	1.10
2	108	1.15	1.7	1.5	1.38
3	107	1.25	1.7	1.1	1.28
4	107	0.9	1.65	1.05	1.23
5	107	1.25	1.55	1.15	0.90
6	107	1.3	1.15	1.55	0.83
7	108	1.2	1.15	1.4	1.33
8	108	1.25	1.7	1	0.85
9	108	1.1	1.75	1.1	1.38
10	108	1	1.1	1.05	0.98
11	108	0.95	1.15	1.55	1.23
12	108	0.9	1.65	1.55	1.14
13	108	1.2	1.2	1.6	0.84
14	107	0.95	1.6	1.1	0.82
15	107	1.1	1.35	1.5	1.30
16	108	1.1	1.4	1.25	0.81
17	108	1.3	1.1	1.1	1.15
18	108	1.1	1.65	1.55	0.91
19	107	1.3	1.1	1.25	1.10
20	107	0.9	1.15	1.2	1.25
21	107	1	1.2	1.35	0.92
22	107	1.3	1.6	1.6	1.24

Table 1: Design of Experiments –DoE–

where different values of the independent parameters have been used as shown in Table 1.

As previously mentioned, the B-Pillar structure inside the car plays an important role to guarantee the safety of all passengers. Therefore, four different spatial points placed at different heights on this B-Pillar structure will be our object of study. This four points, illustrated in Fig. 5 will be referred in the sequel: “Head”, “Thorax”, “Abdomen” and ‘Pelvis”, also sometimes will be referred from their first letter: H, T, A and P respectively. Indeed, the intrusion of these four points when a crash occurs will determine the degree of safety of a given configuration.

Fig. 9 shows the temporal evolution at the four positions of the B-Pillar structure for each one of the DoE appearing in Table 1. As it can be seen all of them follow the main trend, from 0 to 0.05 seconds there is a increment on the intrusion value due to the crash, afterwards there is a relaxation of the intrusion once the main impact has finished due to springback effects.

Since the maximum intrusion of the B-Pillar is going to provide a reliable safety indicator, the first surrogate model is based on selecting the maximum intrusion at the H, T, A and P points as a function of the five parameters

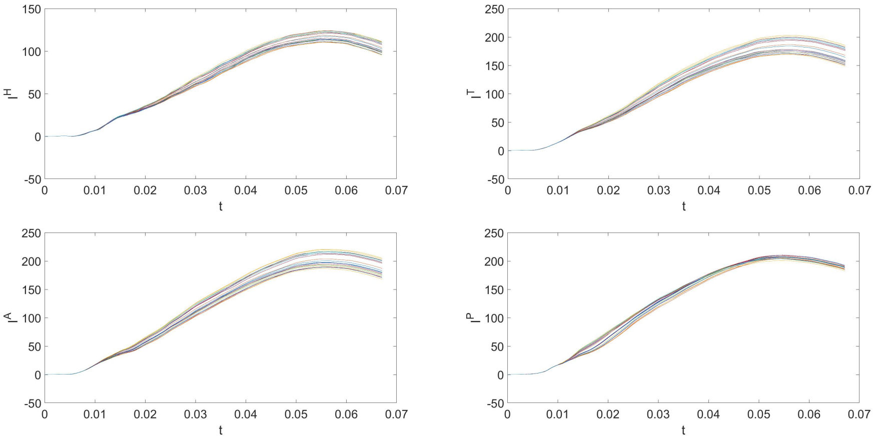

Figure 9: Temporal evolution of the intrusion I at four different locations of the B-Pillar: $I^H,\ I^T, I^A,\ \&\ I^P$.

previously introduced. The parameters are set into vector $\boldsymbol{\mu}$

$$\boldsymbol{\mu} = (\mu_1, \cdots, \mu_5) = [\mathrm{z},\ \mathrm{t}_2, \mathrm{t}_1, \mathrm{t}_3, \mathrm{y}]. \tag{18}$$

The maximum intrusion I_M at point Q (Q refers to locations H, T, A and P) is defined from

$$I_M^Q(\boldsymbol{\mu}) = \max_t I(t; \boldsymbol{\mu}, Q). \tag{19}$$

To test the performance of the sPGD algorithm, only the maximum intrusions related to points $[1, 4, 13, 15, 17, 21]$ inside the DoE are taken into account to construct the regression. The other 16 points are used as an error indicator to show how accurate the regression is. It is important to remark that the low amount of data to build the sPGD regression forces the algorithm to work with low order interpolation basis. Indeed, when generating such a surrogate model up to linear interpolations in each one of the directions will be used. More in detail, the greedy nature of the sPGD algorithm is used to adapt the basis for each one of the modes i.e. the first mode will be constant in each direction, the second mode linear in the first direction and constant in the other ones, the third mode linear in the second direction and constant in the other ones, etc.

Fig. 10 shows the real versus the estimated prediction on the maximum intrusion for Head (H), Thorax (T), Abdomen (A) and Pelvis (P) points i.e. I_M^H, I_M^T, I_M^A, I_M^P, respectively. The yellow points correspond to the ones used to build the sPGD regression, whereas the blue points are just used to measure the sPGD regression predictive accuracy. If all points were on the red line, the surrogate model would be perfect. Nevertheless, the dispersion of these points with respect to the red line gives us a visual indicator of how good the surrogate model is. Indeed, the relative error based on the blue points is shown in Table

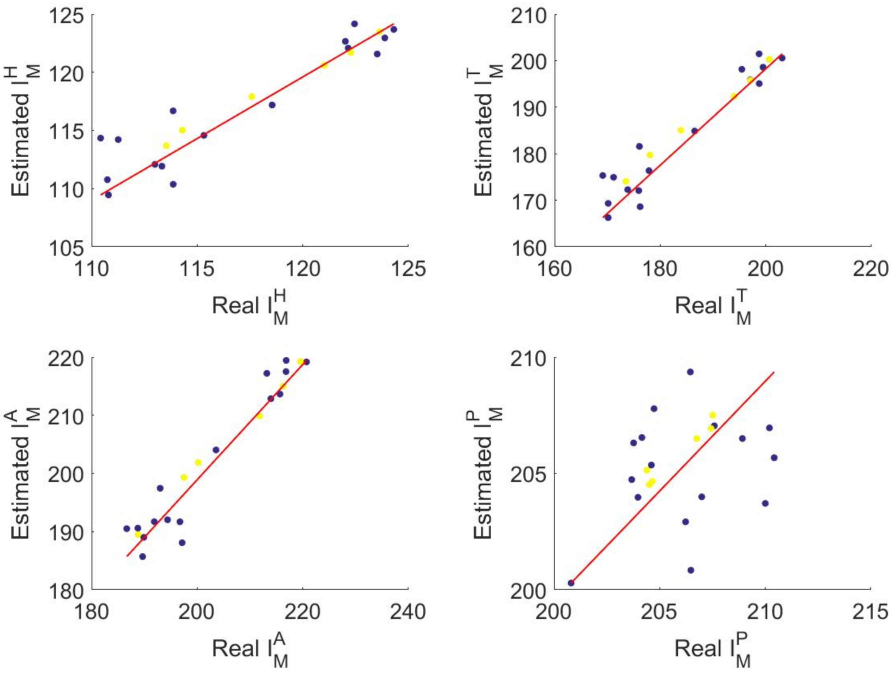

Figure 10: Estimated versus real maximum intrusion for Head (H) I_M^H, Thorax (T) I_M^T, Abdomen (A) I_M^A and Pelvis (P) I_M^P B-pillar points. Yellow points, used in the sPGD. Blue points, used as error indicator of the regression model.

2. As it can be seen, the highest error is the one present in Pelvis (P) point, reaching a value of 4.8%, the closest one to the soft zone.

4.2 Modeling the intrusion time evolution using the sPGD

The previous section analyzed the surrogate models based on the maximum intrusion. However, it is also important to understand how the intrusion evolves in time. Indeed, this temporal evolution gathers the information of the maximum intrusion just like the cumulated energy stored in the B-Pillar structure. Nevertheless, the generation of the surrogate model involving the time coordinate becomes more complex.

To be consistent with the former subsection, in order to build the sPGD

	Head (H)	Thorax (T)	Abdomen (A)	Pelvis (P)
Rel. Error [%]	1.35	1.73	1.38	4.87

Table 2: Relative error in maximum intrusion for sPGD.

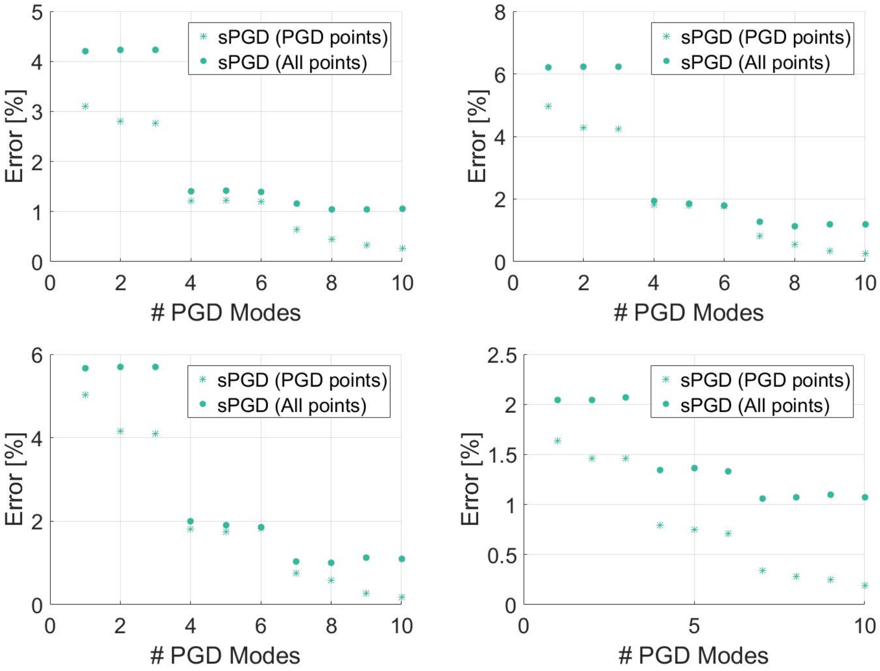

Figure 11: Convergence with respect number of sPGD modes for Head (Top-Left), Thorax (Top-Right), Abdomen (Bottom-Left) and Pelvis (Bottom-Right)

model only the time evolution of intrusion associated to parameters in Table 1 ($[1, 4, 13, 15, 17, 21]$) is taken into consideration. It is important to note that the time coordinate exhibits a rather richer behavior compared to the evolution along the other coordinates. However, it is possible to increase the interpolation order along the time coordinate while keeping low order in the other coordinates. Thus, Chebyshev polynomials of degree 40 are used in the time approximation, while keeping either constant or linear interpolation in the parametric $\boldsymbol{\mu}$ space. The intrusion reads now $I^Q(t; \boldsymbol{\mu}, Q)$.

Fig. 11 depicts both the error considering only the points inside the sPGD regression and the one considering both sPGD regression points plus the ones outside the training dataset. As it can be seen, the error decreases as a function of the number of modes introduced in our sPGD approximation. The most important decrease in the error is seen in the fourth sPGD which involves a linear interpolation along μ_3 (that corresponds to t_1), hence it can be assumed that this parameter plays an important role in order to explain the variation of the QoI inside the parameter space. Moreover, the error decrease almost monotonic while exhibiting different plateaux.

Fig. 12 shows the predicted temporal evolution (red) against the real temporal evolution (blue) for a Head point inside the training set (i.e. DOE 1 left) and

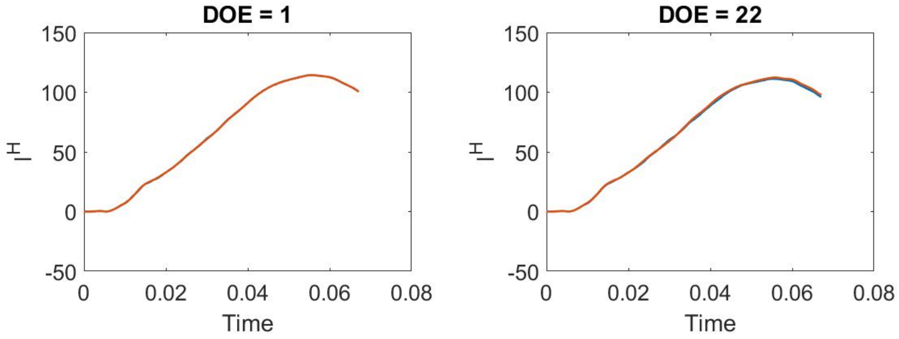

Figure 12: Predicted temporal evolution (red) versus real temporal evolution (blue) for a Head point inside the training set (left) and outside the training set (right).

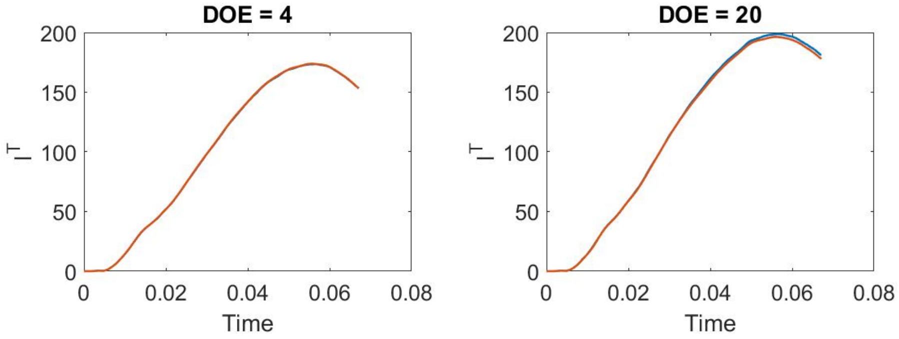

Figure 13: Predicted temporal evolution (red) versus real temporal evolution (blue) for a Thorax point inside the training set (left) and outside the training set (right).

outside the training set (i.e. DOE 22 right). As it can be seen, there is almost no difference between predicted and real values for the point in the data set, i.e. blue and red curves almost overlap, whereas for the point outside the data set there is a very slight difference between red and blue curves. Nevertheless, points outside the trainning set present acceptable errors in this particular case.

Analogously, Figs. 13-15 contains the same information than Fig. 12 but in the case of Thorax, Abdomen and Pelvis points, respectively. The predicted behaviour is in good agreement with respect to the real behaviour.

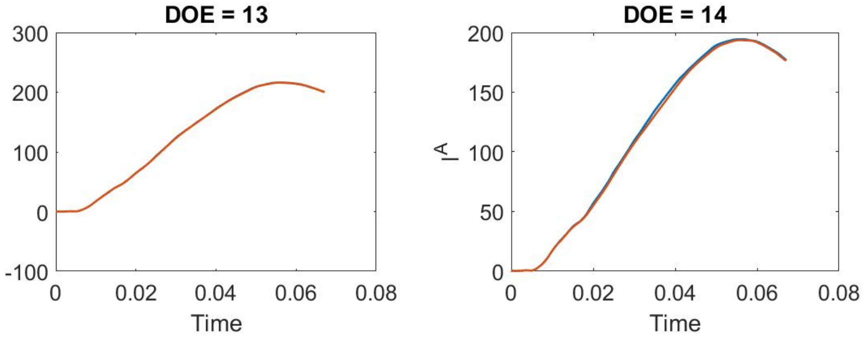

Figure 14: Predicted temporal evolution (red) versus real temporal evolution (blue) for a Abdomen point inside the training set (left) and outside the training set (right).

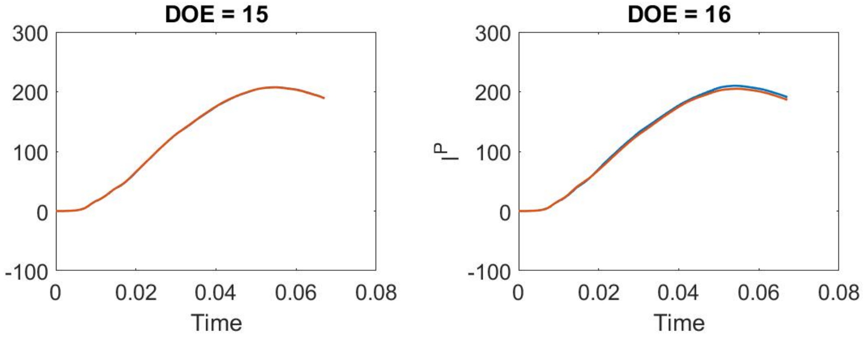

Figure 15: Predicted temporal evolution (red) versus real temporal evolution (blue) for a Pelvis point inside the training set (left) and outside the training set (right).

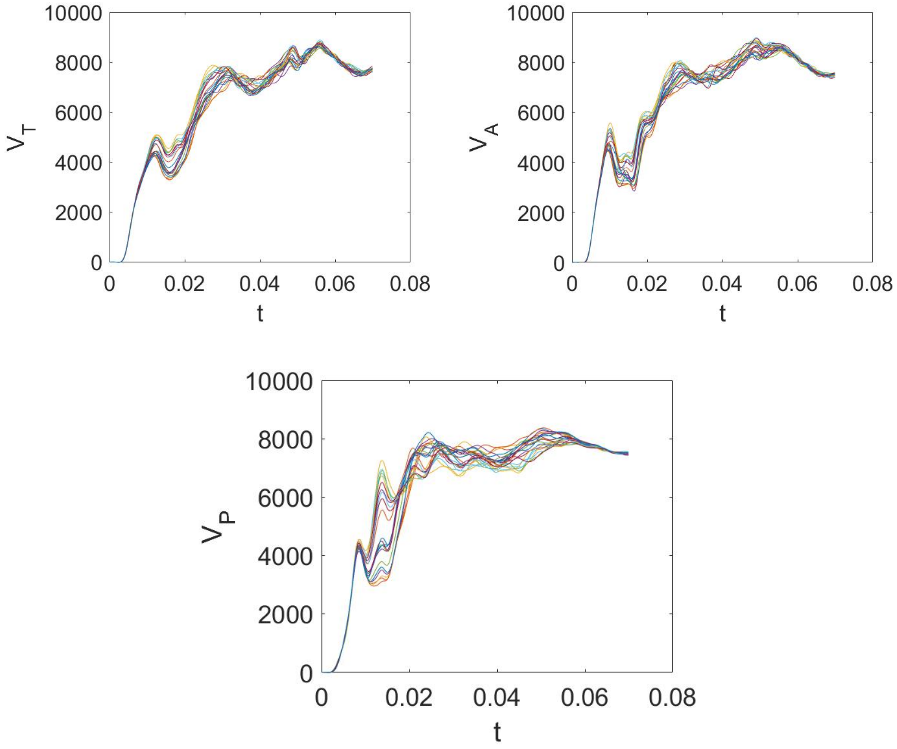

Figure 16: Temporal evolution of the thorax, abdomen and pelvis velocity magnitude throughout the crash simulation for different values of the parametric space.

4.3 Modeling the intrusion velocity time evolution by using the sPGD

Another quantity of interest that is important from a safety point of view is the velocity magnitude experienced at certain points belonging to the body of the passenger throughout the impact. Indeed, the velocity magnitude at three points placed at the Thorax, Abdomen and Pelvis is our subject of study, namely $V_T(\boldsymbol{\mu}, t)$, $V_A(\boldsymbol{\mu}, t)$ and $V_P(\boldsymbol{\mu}, t)$, respectively.

Fig. 16 depicts the temporal evolution of the velocity magnitude throughout the crash for different set of parameters $\boldsymbol{\mu}$. As it can be seen, there is a notable change of this curves within the parameter space. Initially the car is static, and consequently the velocity magnitude is zero, then it starts to grow due to the impact. Hence, the main task is to be able to predict such temporal evolution for different values of parameters. Again, richer time approximation are again expected.

Fig. 17 shows the convergence of the estimated solution for different number

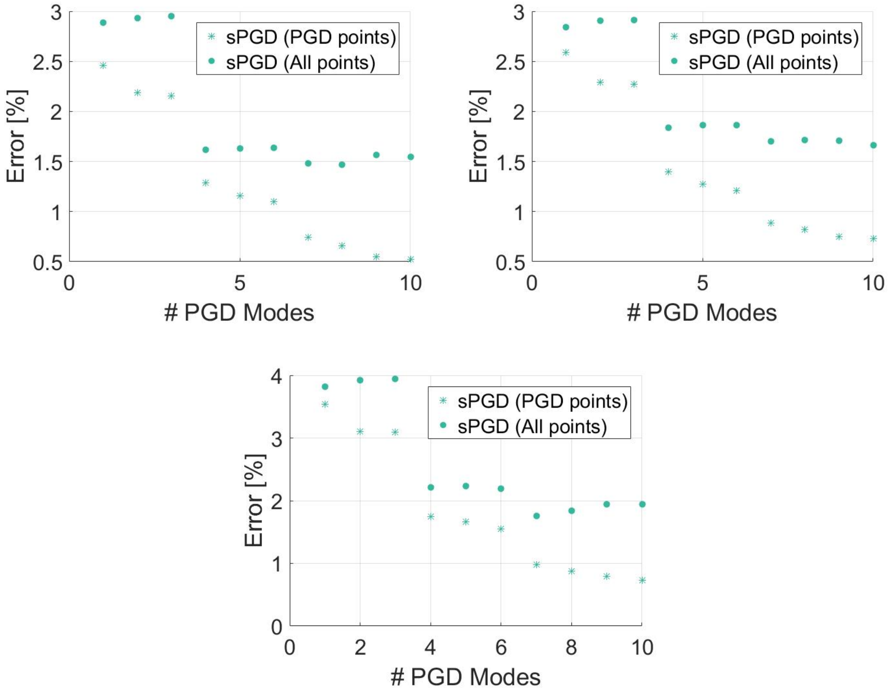

Figure 17: Convergence with respect number of sPGD modes for thorax velocity (Top-Left), abdomen velocity (Top-Right) and pelvis velocity (Bottom)

of sPGD modes. The error is measured using a L2 relative error norm. The points used in the sPGD regression are again $[1, 4, 13, 15, 17, 21]$, with respect to Table 1, the other points are only used to measure the regression accuracy. As it can be seen, the magnitude of all error indicators are acceptable (around 1%-2%) being the highest error the one associated with the pelvis velocity. Indeed, the pelvis velocity is the one presenting the highest error since the variation of the curve along the parameter space is the highest one as well.

Figs. 18-20 show the predicted temporal evolution (red) versus the real temporal evolution (blue) for velocity magnitudes inside the training set (left) and outside the training set (right). It can be notice how the predicted curve is closer to the real curve when the point is inside the training data set. Nevertheless, the prediction outside the training points also captures the main features of the real curve.

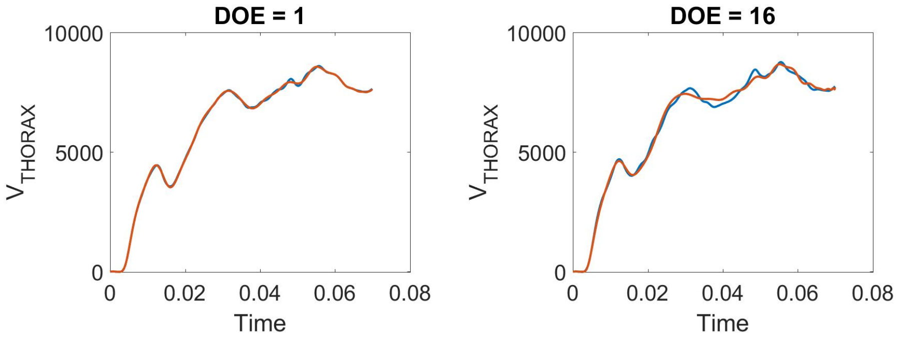

Figure 18: Predicted temporal evolution (red) versus real temporal evolution (blue) for a thorax velocity magnitude inside the training set (left) and outside the training set (right).

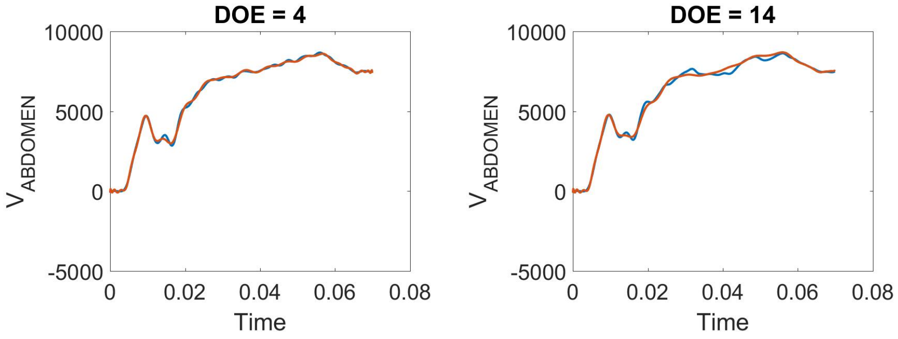

Figure 19: Predicted temporal evolution (red) versus real temporal evolution (blue) for a abdomen velocity magnitude inside the training set (left) and outside the training set (right).

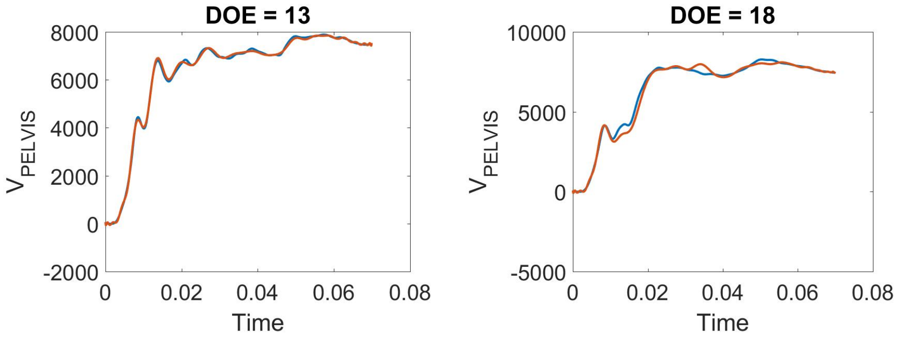

Figure 20: Predicted temporal evolution (red) versus real temporal evolution (blue) for a pelvis velocity magnitude inside the training set (left) and outside the training set (right).

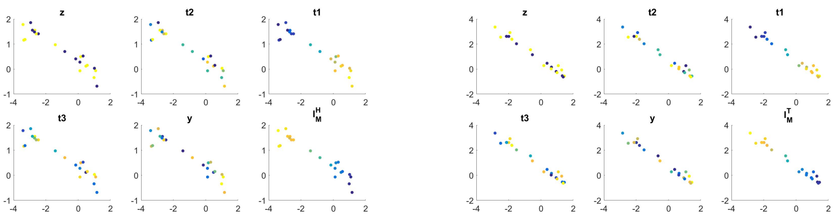

Figure 21: Mapped data concerning Head (left) and Thorax (right) intrusion.

4.4 Classifying multi-dimensional data by employing the *Code2Vect* technique

In this section the application of *Code2Vect* to classify multi-dimensional data is discussed. As before stated the GESTAMP dataset is conformed by five parameters, being the maximum intrusion the QoI. The obtained low-dimensional vectors $\mathbf{x}^j \in \mathbb{R}^2$ (see Section 3.2) are depicted for the Head and Thorax cases in Fig. 21 and for the Abdomen and Pelvis ones in Fig. 22. Each sampling point of the data-set becomes a point in the target vector space as described in Section 3.2. For the sake of easy visualization the vector space was enforced to have low dimensionality, 2D in the examples discussed below.

As soon as all the points were mapped into the 2D vector space, a color was assigned to each of them, corresponding to the QoI but also to the value of the different parameters in order to see which of them clusterize in a similar way that the QoI. Such correlations serve to conclude on which parameters directly explain the QoI.

Fig. 21 proves that the third parameter $\mu_3 \equiv \mathrm{t}_1$ has a direct influence on the output. This conclusion reinforces the observation when using the sPGD. The other parameters are not individually correlated, and consequently if they influence the output, they should proceed in a combined manner. It has been proved that by removing them (all parameters except the third parameter $\mu_3 \equiv \mathrm{t}_1$) predictions remain very accurate, mainly at H, T & A locations, proving the fact that responses are almost dictated by the third parameter. This fact is physically interpretable because, as soon as the soft zone localizes the deformation, the mechanical response is almost determined by parameter $\mu_3 \equiv \mathrm{t}_1$ related to the outer thickness t_1, being quite insensible to the soft zone location and grade as well as to the inner and patch thicknesses. Obviously, when approaching the soft zone (Pelvis) this tendency seems, as expected, less pronounced as Fig. 22(right) proves.

4.5 Maximum Intrusion using the *Code2Vect* technique

In this section *Code2Vect* is used as regression procedure. The same points considered in Section 4.1 were used here for training purposes, that is, for computing matrix $\mathbf{W}$. Then, the remaining points were mapped and the QoI interpolated

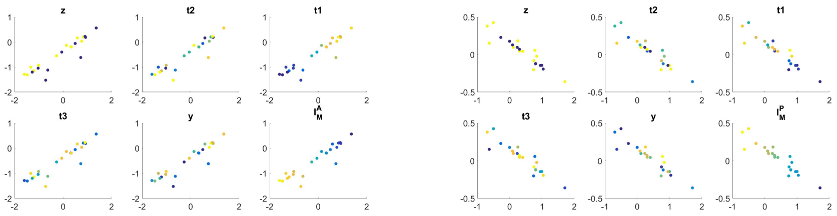

Figure 22: Mapped data concerning the Abdomen (left) and Pelvis (right) intrusion.

	Head	Thorax	Abdomen	Pelvis
Rel. Error [%]	2.50	4.21	5.12	1.73

Table 3: Relative error in the prediction of the maximal intrusion when using the *Code2Vect*-based regression.

from the neighbor data. Thus, the error between the known QoI at that points and the one predicted was calculated, as reported in Table 3 and depicted in Fig. 23. The best prediction is achieved for the intrusion at the Pelvis location, while the worst prediction is related to the one related to the Abdomen.

4.6 Sensitivity analysis and uncertainty propagation

Part thickness and material properties can slightly change from one steel blank to another, due to process variability. To perform a statistical analysis on the intrusions of the points of interest, we assumed that parameters $[\mathrm{t}_2, \mathrm{t}_1, \mathrm{t}_3, \mathrm{y}]$ follow gaussian distributions (Fig. 24), respectively $\mathcal{N}(1.1, 0.06)$, $\mathcal{N}(1.4, 0.09)$, $\mathcal{N}(1.3, 0.08)$, $\mathcal{N}(1.09, 0.07)$, where the discrete parameter $\mathrm{z} = 107$. Using the constructed PGD regression from points $[1, 4, 13, 15, 17, 21]$ (previously discussed), 10000 extra-configurations were generated by varying parameter values.

Maximum intrusions were calculated at the four points of interest: Head, Thorax, Abdomen, and Pelvis, and their resulting distributions are represented in Fig. 25. Two probability density functions are shown for each quantity: (i) the gaussian density distribution and (ii) the kernel density estimation (KDE). On the one hand, the Abdomen intrusion distribution is relatively wide because a local bending may appear if the B-pillar is not strong enough to withstand the resultant forces. On the other hand, the Pelvis intrusion displays a low standard deviation that can be explained physically from the fact that the lower grade applied in this area is used to localize the deformation. The bending will hence always occur here. Moreover, the resultant displacement is also driven by the deformation of the side sill component, which remains unchanged. The impact of the energy absorption by the lower area has a greater effect on upper intrusions than on the lower intrusion because it drives the overall distribution of forces in the B-pillar.

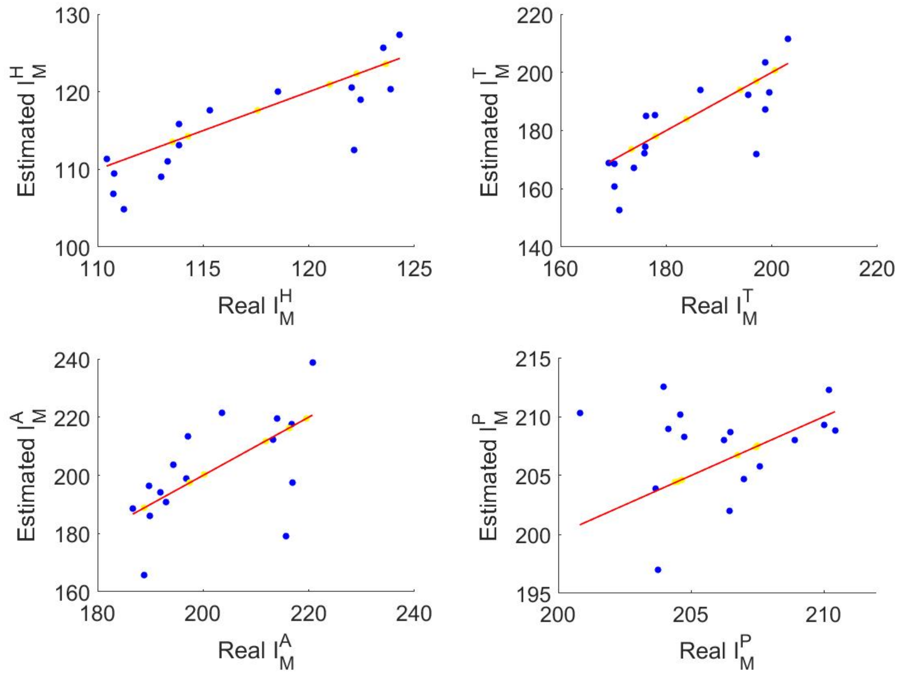

Figure 23: Estimated versus real maximum intrusion for Head (H), Thorax (T), Abdomen (A) and Pelvis (P) B-pillar points. Yellow points, are used for training purposes in *Code2Vect* whereas blue points serves to quantify the prediction accuracy.

In order to explore better the design space, we generate 20000 new configurations and then compute (by using the PGD nonlinear regression model) the related maximum intrusion for the points of interest: H, T, A and P. Using the analytics features of Mineset™, sensitivity analysis and optimization studies were performed.

Fig. 26 displays the parameters influence on the intrusion. As expected, it confirms that $\mu_3 \equiv \mathrm{t}_1$ is the critical design parameter for this QoI. Similar conclusions can be stressed for the other intrusions points; the outer thickness has an overwhelming importance compared to the other parameters.

Finally, two optimization studies were done, one considering a single constraint related to the Abdomen, Fig. 27, and the other by considering constraints on every measurement point (see Fig. 28). Optimized parameter values are consistent for this type of crash example. As several values are proposed for z and y parameters, it seems that their influence is low for the selected constraints.

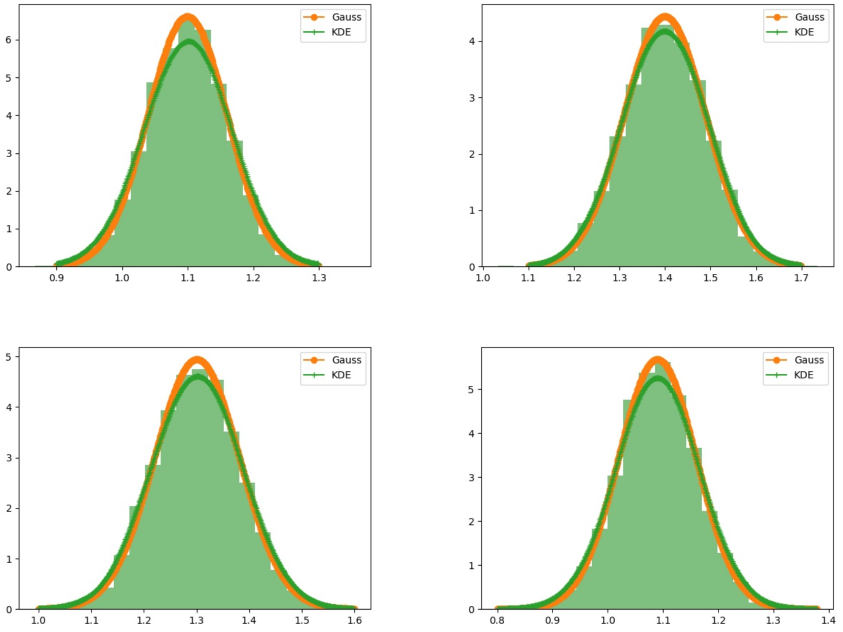

Figure 24: Distributions of (Top-left) t_2; (Top-right) t_1; (Bottom-left) t_3 and (Bottom-right) y.

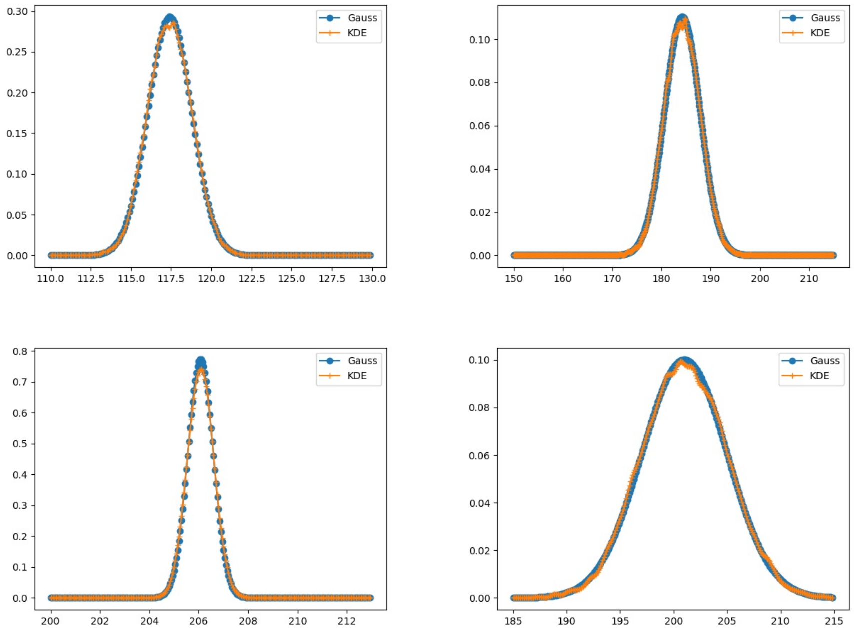

Figure 25: (Top left) the Maximum Head intrusion distribution; (Top right) Maximum Thorax Intrusion distribution; (Bottom left) Maximum Pelvis intrusion distribution, (Bottom Right) the Maximum Abdomen intrusion distribution.

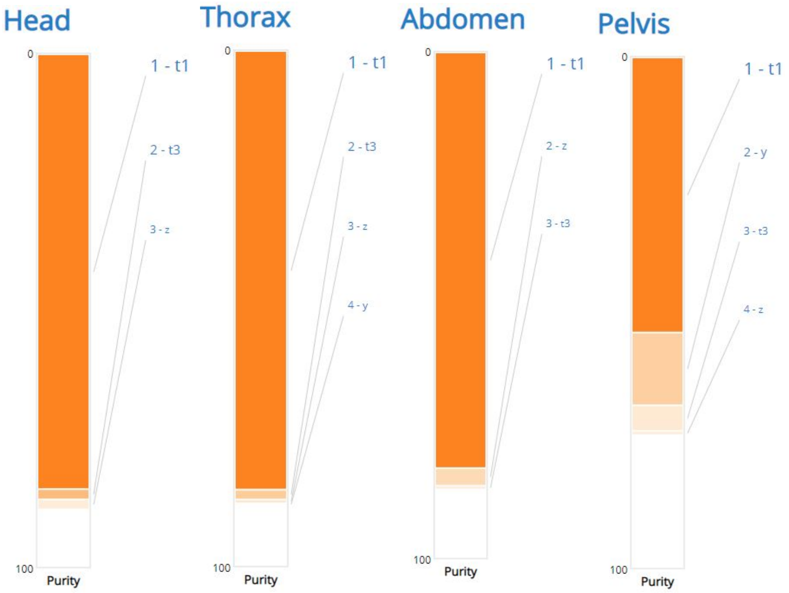

Figure 26: Column of importance on the maximum intrusion on the four points.

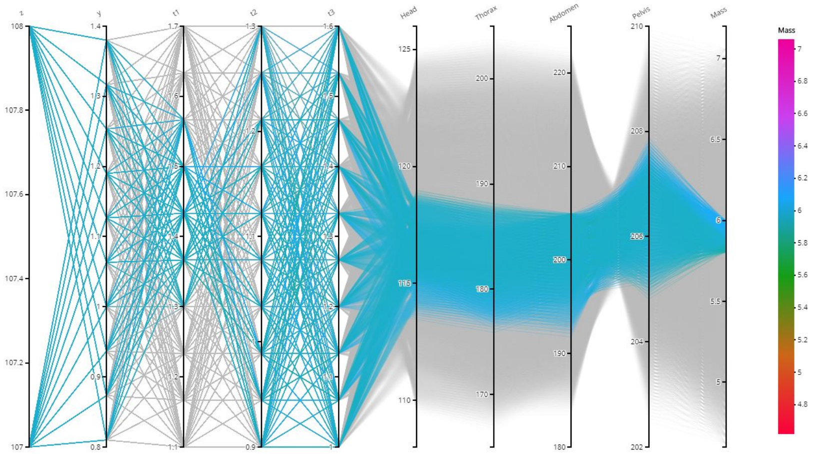

Figure 27: Optimization using parallel coordinates considering Abdomen intrusions.

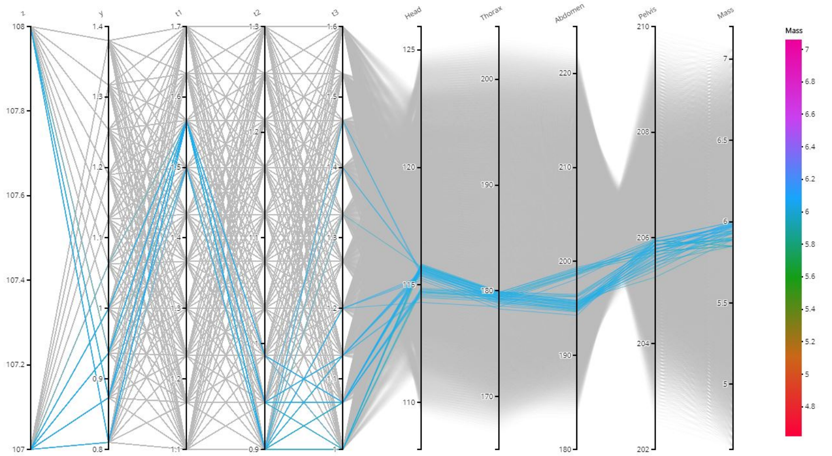

Figure 28: Optimization using parallel coordinates considering all constraints.

5 Conclusions

In this paper we proposed two different techniques able to extract quantities of interest – QoI – and express them parametrically, in the low-data limit. These procedures were successfully applied to analyze parametrically a B-Pillar within a full vehicle crash test. Indeed, it allowed to reduce significantly the number of simulations needed to perform prediction, optimization and sensitivity analyses.

It was proved that in this particular case, the computational complexity scales linearly with the number of parameters, i.e., 6 data suffice for expressing a parametric solution involving 5 parameters, a quite impressive performance due to the features of sparse PGD formulations. Moreover, the data generated by using these parametric solutions was considered for performing sensitivity and uncertainty propagation analyses.

The prediction of maximal intrusions and time evolution of intrusion and intrusion velocities were successfully accomplished by invoking the sPGD. By using the *Code2Vect* the influence of the different parameters on the quantity of interest – QoI – was determined, and it was concluded that it remains quite insensible to the variation of 4 among the 5 parameters. The QoI obviously depends on those parameters, but it does not vary significantly when these 4 parameters vary in their considered intervals. These tendencies were confirmed by all the analysis techniques considered: sPGD, *Code2Vect* and MINESET™ software on data-analytics, the last considered in the present work for performing uncertainty propagation.

To go further, performances could be evaluated based on more complex cases, involving highly sensitive parameters and bifurcations.

References

[1] C. Argerich, R. Ibanez, F. Chinesta. *Code2vect*: An efficient heterogenous data classifier and nonlinear regression technique. CRAS Mécanique. Submitted.

[2] D. Borzacchiello, J.V. Aguado, F. Chinesta. Non-intrusive sparse subspace learning for parametrized problems. Arch. Comput. Methods Eng., In press.

[3] S.L. Brunton, J.L. Proctor, N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. In: Proceedings of the national academy of sciences, 2016.

[4] F. Chinesta, P. Ladeveze, E. Cueto. A short review in model order reduction based on proper generalized decomposition. Arch. Comput. Methods Eng., 18, 395-404, 2011.

[5] 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. Arch. Comput. Methods Eng., 20, 31-59, 2013.

[6] F. Chinesta, R. Keunings, A. Leygue. The Proper Generalized Decomposition for Advanced Numerical Simulations. A primer. Springerbriefs, Springer, 2014.

[7] F. Chinesta, A. Huerta, G. Rozza, K. Willcox. Model Order Reduction. Chapter in the Encyclopedia of Computational Mechanics, Second Edition, Erwin Stein, René de Borst & Tom Hughes Edt., John Wiley & Sons Ltd., 2015.

[8] R. Ibanez, E. Abisset-Chavanne, A. Ammar, D. Gonzalez, E. Cueto, A. Huerta, J.L. Duval, F. Chinesta. A multi-dimensional data-driven sparse identification technique: the sparse proper generalized decomposition. Complexity 2018:5608286, 2018, https://doi.org/10.1155/2018/5608286

[9] P.J. Schmid. Dynamic mode decomposition of numerical and experimental data. J. Fluid Mech., 656, 5-28, 2010.

[10] Safety Wissen, February 2019, European News Car Assessment Programme (EuroNCAP), available at https://www.safetywissen.com/#/requirement/