Enabling the evidence theory through non-intrusive parametric model order reduction for crash simulations

J. S. Jehle[1),2)], V. A. Lange[1),3)], M. Gerdts[2)]

[1)] *Research and Innovation Center, BMW Group, D-80788 Munich, Germany, jonas.jehle@bmw.de, volker.vl.lange@bmw.de*

[2)] *Institute of Applied Mathematics and Scientific Computing, Department of Aerospace Engineering, Universität der Bundeswehr München, D-85577 Neubiberg, Germany, jonas.jehle@unibw.de, matthias.gerdts@unibw.de*

[3)] *Associate Professorship of Computational Mechanics, Department of Civil, Geo and Environmental Engineering, Technical University of Munich, D-80333 Munich, Germany, volker.vl.lange@tum.de*

Abstract. To improve the crashworthiness of cars, it is crucial for vehicle safety departments to consider uncertainties. In addition to hardware crash tests, finite element simulations (FES) serve to assess the behavior in a crash scenario. As these black-box simulations are computationally expensive, e.g. due to the huge number of finite elements, uncertainty quantification has prohibitively high cost. We therefore propose to approximate the FES using non-intrusive parametric model order reduction (MOR). The use of MOR requires model outputs of training simulations—called snapshots—resulting from different input parameter sets. Here, snapshots consist of the node displacements of all finite elements at specific time states. MOR strongly reduces the high dimension of the problem to a lower one by performing a singular value decomposition. Afterwards, the lower dimensional subspace is approximated by metamodels—one for each dimension. This study applies Gaussian process regressions. Combining MOR with metamodels allows the reconstruction of the full FES. New parameter sets can be quickly evaluated, i.e. uncertainties can be rapidly propagated through the reduced model. This enables the use of the evidence theory for each key result of the FES. The resulting plausibility and belief curve help vehicle safety engineers to improve the crashworthiness under uncertainty.

Keywords: non-intrusive parametric model order reduction, metamodel, uncertainty quantification, evidence theory, crash simulation

1. Introduction

Hardware crash tests of vehicles are subject to various uncertainties that influence the test result. It is common to distinguish between the epistemic and aleatory category of uncertainties in literature (Der Kiureghian and Ditlevsen, 2009; Beyer and Sendhoff, 2007). Epistemic uncertainties are caused by lack of knowledge when the proper value for a certain quantity can not be fixed. They can be reduced by clearer insight through additional or improved data (Swiler and Giunta, 2007). Aleatory uncertainties are irreducible and imply that the system response under investigation is intrinsically random (Oberkampf et al., 2004). Both forms of uncertainty are encountered during vehicle development.

Component properties such as sheet metal thickness, elasticity modules, or bumper height are not specified in early development phases. Later on, they are fixed and these epistemic uncertainties vanish. Test scatter like impact angle, velocity, or the position of the dummy is always existent and thus a representative of the aleatory class.

Apart from the hardware tests, vehicle safety engineers work with computer simulations to study the behavior of the car in a collision. These simulations are often finite element simulations (FES). Compared to hardware tests, they are cheap with respect to monetary cost but nonetheless they incur immense computational effort, i.e. time and mobilization of high-performance computing hardware. The expensive calculation costs are due to nonlinear phenomena and the necessary detail of the model, which is reflected by the high number of finite elements. To gain an overview of possible test results, the uncertainties occurring in hardware tests must be incorporated into simulations.

We suggest an approach to propagate these uncertainties, epistemic and aleatory, through automotive crash FES and quantify the outcomes. The main part of this approach is an extended version of the Evidence Theory (ET) based on the work of (Oberkampf et al., 2001) and (Oberkampf and Helton, 2002). This method propagates the predetermined uncertain inputs through the model. The uncertainty of each input is modeled through the use of (sub-)intervals. Experts are responsible for subdividing the intervals and assigning a probability of occurrence to each subinterval. Next, ET considers all possible combinations of subintervals between the different inputs and calculates the composite probabilities for each combination.

Then, optimization strategies are used to find the infimum and supremum values of the model for each subinterval combination. Afterwards, all infimum and supremum values are aggregated, sorted in ascending order, and stored together with their corresponding composite probabilities. Cumulating the probabilities yields two curves belonging to the infimum and supremum values. They have the form of a cumulative distribution function. On the basis of these curves, it is possible to identify with two different measures which outputs of the model are likely to occur.

Executing these optimizations can form a bottleneck in the application of ET. As FES of vehicle crash models are regarded as black-boxes, optimization requires many evaluations. This makes larger analyses with ET prohibitively expensive in terms of time. This bottleneck can be overcome by using faster models.

We thus propose to use non-intrusive parametric Model Order Reduction (MOR) (Guo and Hesthaven, 2018; Le Guennec et al., 2018; Swischuk et al., 2019) for approximating the computer simulation. MOR requires model outputs of training simulations that are generated from various input parameter sets. Here, the model outputs comprise the node displacements of all finite elements at particular time states. A singular value decomposition is performed on the training data to generate a lower dimensional subspace. Metamodels are trained to approximate each of the remaining dimensions. The complete FES can then be reconstructed from the output of a small number of metamodels. As these can be evaluated efficiently, this reduced model is an attractive choice to replace the FES in ET.

The proposed combined approach of non-intrusive parametric MOR and ET is described in more detail in the following sections. The first part of Section 2 guides through the different steps of ET. The second part presents the elements of the non-intrusive MOR approach. The crashbox simulation characterized in the first part of Section 3 serves as proof of concept. The deformation of the finite element crashbox is simulated with the software (LS-DYNA, 2018). The second part of Section 3 demonstrates the application steps of the new approach needed for the crashbox. We implement our combined method in (MATLAB, 2018). Section 4 analyzes the quality of the approach and discusses results by interpreting the ET curves. Section 5 concludes the paper.

2. Developing the Combined Approach of Non-intrusive Parametric MOR and ET

2.1. Uncertainty Propagation through the System via Evidence Theory

The Evidence Theory (ET) devised by the mathematicians Dempster and Shafer (Shafer, 1976)—also known as Dempster-Shafer Theory or Theory of the Belief Functions—was originally invented to combine information from different sources to a comprehensive statement. Each of these sources has a credibility allocation which is taken into acount during the calculation. In (Nikolaidis et al., 2004; Swiler et al., 2009; Eldred and Swiler, 2009; Auer et al., 2010), ET is extended to propagate interval-based uncertainties of input parameters through the system function. As a result, one obtains lower and upper limits for the cumulative distribution function of the output. These are called belief and plausibility curve, respectively.

To use the ET, it is not necessary to know the form of the system function, f, i.e. we consider a general input-output model,

$$y = f(\mu) \in \mathbb{R}, \tag{1}$$

for the inputs $\mu \in \mathbb{R}^n, n \in \mathbb{N}$, without any further knowledge about f. Hence, f can also be a black-box model. The components, μ_i, $i \in \{1, \ldots, n\}$, of the input vector, μ, are viewed as independent probability variables. We assume that each component, μ_i, has a probability density whose support is contained in the connected interval $I_i \subset \mathbb{R}$, that is, I_i covers the range of possible values of μ_i. To start the ET analysis, experts assess these overall intervals in two steps. They are illustrated in Fig. 2 for the exemplary application.

1. The intervals I_i are split into $k_i \in \mathbb{N}$ subintervals,

   $$I_i^j \subset I_i, \qquad j \in \{1, \ldots, k_i\}, \tag{2}$$

   for each input dimension, $i \in \{1, \ldots, n\}$. These subsets are called focal elements. By forming their union, a new input set,

   $$\hat{I}_i := \bigcup_{j=1}^{k_i} I_i^j, \tag{3}$$

   for μ_i is obtained. Note that it is admissible for these subintervals to be overlapping, connected, or disjoint. Clearly, $\hat{I}_i$ is a subset of the overall interval I_i. It is, however, possible that $\hat{I}_i$ is not connected, i.e. it can contain gaps.

2. The experts proceed with a so-called Basic Probability Assignment (BPA) where they allocate probabilities to all focal elements, I_i^j. In mathematical terms, the experts define a BPA operator,

m_i, for every μ_i that maps the focal elements, I_i^j, to a probability. These operators can be expressed as

$$m_i(J) \begin{cases} \in [0,1], & \text{for } J \in \mathcal{I}_i := \left\{ I_i^j \middle| j \in \{1,\dots,k_i\} \right\}, \\ = 0, & \text{for } J \in \mathcal{B}(\mathbb{R}) \backslash \mathcal{I}_i, \end{cases} \tag{4}$$

where $\mathcal{B}(\mathbb{R})$ is the Borel σ-algebra of the real numbers. For each m_i, it must hold that the sum of probabilities distributed to the focal elements equals one,

$$\sum_{J \in \mathcal{I}_i} m_i(J) = 1, \qquad i \in \{1,\dots,n\}. \tag{5}$$

The assessment does not have to be unanimous between the experts. If there are differences, one can combine deviating opinions by the traditional ET approach, e.g. via Dempster's rule of combination (Sentz and Ferson, 2002). Note that the expert assessments can be based on their knowledge, existing data, physical models, related information, etc. For simplicity, only one fictive expert is considered here.

In case the expert assumes an input to be a continuous random variable, its probability density function can be represented approximately by a histogram. For this purpose, there exist different methods to choose this histogram appropriately, see (Knuth, 2019; Scott, 1979; Freedman and Diaconis, 1981). Histograms can be converted into ET structures. This allows the theory to deal roughly with continuous random variables.

After generating these initial data, the ET uncertainty propagation is accomplished in four steps:

3. We combine all possible focal elements between input dimensions to n-dimensional hypercubes called interval cells. There are in total $k := \prod_{i=1}^{n} k_i$ interval cells of the form

$$C^{j_1,j_2,\dots,j_n} := I_1^{j_1} \times I_2^{j_2} \times \cdots \times I_n^{j_n}, \tag{6}$$

for $j_i \in \{1,\dots,k_i\}$ and $i \in \{1,\dots,n\}$. We can simplify the notation by C^l, $l \in \{1,\dots,k\}$, instead of writing $C^{j_1,j_2,\dots,j_n}$, $j_i \in \{1,\dots,k_i\}$, respectively, since there is a one-to-one mapping between $\{1,\dots,k\}$ and $\prod_{i=1}^{n} \{1,\dots,k_i\}$. Furthermore, we denote the set of resulting interval cells as $\mathcal{C}$.

4. The composite BPAs, p^l, for C^l are computed by

$$p^l := p^{j_1,\dots,j_n} := \prod_{i=1}^{n} m_i \left(I_i^{j_i} \right). \tag{7}$$

Herein, we exploited the independence of the probability variables, $\mu_i, i \in \{1,\dots,n\}$. In this way, the single BPAs corresponding to composite focal elements are multiplied together. Note that the probabilities of all interval cells sum up to one, i.e.

$$\sum_{l=1}^{k} p^l = 1. \tag{8}$$

Steps 3 and 4 are depicted in Fig. 3 for the exemplary application, whereas the result of the following steps 5 and 6 are shown in Fig. 4(c).

5. Every interval cell, C^l, has to be propagated through the model. To obtain lower and upper limits for the cumulative distribution function, we solve two optimization problems

$$\overline{y}^l = \sup_{\mu \in C^l} f(\mu), \tag{9}$$

$$\underline{y}^l = \inf_{\mu \in C^l} f(\mu), \tag{10}$$

for each interval cell. In total, we have to solve $2k$ optimization problems. We remark that other works (Helton et al., 2006; Tian et al., 2018) use sampling strategies to find these values where their accuracy strongly depends on the size of the sample.

6. We aggregate all supremum and infimum values, $\overline{y}^l$ and $\underline{y}^l$, along with their composite BPAs, p^l. The belief and plausibility curve result from sorting $\overline{y}^l$ and $\underline{y}^l$ in ascending order and cumulating their probabilities, respectively. Expressed as formulas, we define the belief curve as

$$\mathrm{Bel}(y) = \sum_{\left\{l \middle| y \geq \overline{y}^l\right\}} p^l \tag{11}$$

and the plausibility curve as

$$\mathrm{Pl}(y) = \sum_{\left\{l \middle| y \geq \underline{y}^l\right\}} p^l. \tag{12}$$

From a probabilistic point of view, since μ is considered as a random variable, $y = f(\mu)$ is a random variable as well. The cumulative distribution function of y is unknown but it can be limited using the ET curves. For an arbitrary but fixed $\tilde{y}$, the belief curve forms a lower limit of the cumulative distribution function of y, i.e. it holds

$$\mathrm{Bel}(\tilde{y}) \leq P(y \leq \tilde{y}), \tag{13}$$

where $P(A)$ denotes the probability of event A. Beyond, the plausibility curve provides an upper limit, i.e.

$$P(y \leq \tilde{y}) \leq \mathrm{Pl}(\tilde{y}). \tag{14}$$

Note that computational costs of ET increase, when the number of inputs or the number of focal elements grow, since one has to run more and more optimizations. In addition, when the optimization of the system function is expensive due to nonlinearities or its black-box character, applying ET gets prohibitively costly. Different works therefore propose to couple ET with metamodels to accelerate the optimization, see for example the Multi-Point Approximation Method (Bae et al., 2004) and Stochastic Expansion Methods (Eldred et al., 2011). The metamodel has to cover the input space $\prod_{i=1}^{n} \hat{I}_i$ and builds a fast-responding mathematical model for the output based on training sample points.

2.2. Non-intrusive Parametric Model Order Reduction

We suggest another approach to overcome the problem of solving the costly ET optimizations for FES. It is similar to traditional metamodels in that we want to quickly produce a scalar output—referred to as the key result here. Conventional metamodels only train on the key results themselves. In contrast, non-intrusive parametric Model Order Reduction (MOR) allows to reconstruct the whole simulation by approximating the node displacements of the finite elements.

Key results can then be extracted from the approximated simulation. When multiple key results are considered, they can be derived from a single model rather than creating separate metamodels. Furthermore, the outputs of finite element crash simulations are vectors that together with the FES time steps form a curve, e.g. the displacement curve for the center node of the bumper. The key results are then extracted from these curves. These extractions are based on known mathematical functions that may be nonlinear or discontinuous. Common metamodels have to model these functions as well. MOR does not have to consider these extractions. The extractions can be applied on the MOR approximations afterwards which may yield better results.

The theory for non-intrusive parametric MOR was introduced in (Guo and Hesthaven, 2018) for quasi-static problems. The same authors extended this approach for time-dependent problems in (Guo and Hesthaven, 2019). We summarize the important points and modify the theory for our problem.

We consider a model expressed by a time-dependent parametrized vector function

$$u(t, \mu), \quad (t, \mu) \in \mathcal{T} \times \mathcal{M} \tag{15}$$

where $\mathcal{T} = [0, T], T \in \mathbb{R}_{>0}$, denotes the time interval and $\mathcal{M} \subset \mathbb{R}^n$ is called the parameter space. In our crash test application, the function u describes the displacements of all nodes of the finite element in x-, y-, and z-direction at a specific time, $t \in \mathcal{T}$. The x-, y-, and z-displacements of each node are understood as three individual degrees of freedom. The total number of degrees of freedom, $N_h \in \mathbb{N}$, is thus obtained by multiplying the number of nodes, $N_{\mathrm{nd}} \in \mathbb{N}$, by three, i.e. $N_h = 3N_{\mathrm{nd}}$. Evaluated at $\left(\tilde{t}, \tilde{\mu}\right) \in \mathcal{T} \times \mathcal{M}$, $u\left(\tilde{t}, \tilde{\mu}\right)$ is a vector of length N_h. We assume that the vector $u\left(\tilde{t}, \tilde{\mu}\right)$ first contains all x-displacements, followed by the y-displacements, and finally the z-displacements; node-by-node in a predetermined order, i.e.

$$u\left(\tilde{t}, \tilde{\mu}\right) = \left(u_{\mathrm{x}_1}, \ldots, u_{\mathrm{x}_{N_{\mathrm{nd}}}}, u_{\mathrm{y}_1}, \ldots, u_{\mathrm{y}_{N_{\mathrm{nd}}}}, u_{\mathrm{z}_1}, \ldots, u_{\mathrm{z}_{N_{\mathrm{nd}}}}\right)^T \left(\tilde{t}, \tilde{\mu}\right) \in \mathbb{R}^{N_h}. \tag{16}$$

In FES, the time space is represented by the discrete set $\mathcal{T}_d := \{t_1, \ldots, t_{N_t}\} \subset \mathcal{T}$ of size $N_t \in \mathbb{N}$. We assume equidistant points in time for simplicity. For an arbitrary but fixed parameter configuration $\tilde{\mu} \in \mathcal{M}$, the whole simulation can then be described by the matrix

$$U(\tilde{\mu}) := \left[u\left(t_1, \tilde{\mu}\right), \ldots, u\left(t_{N_t}, \tilde{\mu}\right)\right] \in \mathbb{R}^{N_h \times N_t}. \tag{17}$$

The non-intrusive parametric MOR approach approximates this matrix using dimensionality reduction and regression in decoupled offline and online stages. First, in an offline fashion, the high number of dimensions in the FES is reduced in three steps:

1. We collect $N_s \in \mathbb{N}$ snapshots, i.e. evaluations of U, by running FES with different parameter constellations, $\mu^1, \ldots, \mu^{N_s} \in \mathcal{M}$, and record them in the snapshot matrix
$$S = (S_1, \ldots, S_{N_s}) := \left[U\left(\mu^1\right), \ldots, U\left(\mu^{N_s}\right)\right] \in \mathbb{R}^{N_h \times N_t N_s}. \tag{18}$$
We assume
$$N_h > N_t N_s \tag{19}$$
which is typically fulfilled for FES due to the high number of nodes.

2. The singular values of S are found via singular value decomposition, i.e. we decompose S into
$$S = Q \Sigma Z^T \tag{20}$$
with the left- and right-singular factors $Q \in \mathbb{R}^{N_h \times N_h}$ and $Z \in \mathbb{R}^{N_t N_s \times N_t N_s}$ being orthogonal matrices. The diagonal matrix, $\Sigma \in \mathbb{R}^{N_h \times N_t N_s}$, moreover contains the singular values, $\sigma_1, \ldots, \sigma_{N_t N_s}$, of S in decreasing order. Here, the MATLAB routine `svd` is chosen for the singular value decomposition.

3. The projection matrix $V \in \mathbb{R}^{N_h \times L}, L \in \mathbb{N}$, reduces the dimensionality of the full space model from N_h to $L \ll N_h$. It is used to map from the full space to the reduced space and back. After finding the lowest natural number L, fulfilling the ratio
$$\sum_{i=1}^{L} \sigma_i^2 \Bigg/ \sum_{j=1}^{N_t N_s} \sigma_j^2 > \epsilon, \tag{21}$$
for a fixed $\epsilon \in [0.9, 1)$, we obtain V by setting it equal to the first L columns of the left-singular factor Q. This procedure is meant to only use those columns for the reduced space that correspond to dominant modes.

After this reduction procedure, we are able to project a full simulation, $U(\tilde{\mu})$, for $\tilde{\mu} \in \mathcal{M}$, into the reduced space by multiplying with V^T, i.e. we compute
$$R := V^T U(\tilde{\mu}) \in \mathbb{R}^{L \times N_t}. \tag{22}$$
Projecting back into the full space, in turn, is performed by multiplying by V, i.e.
$$VR = VV^T U(\tilde{\mu}) \in \mathbb{R}^{N_h \times N_t} \tag{23}$$
lies again in the full space. The projection error between $U(\tilde{\mu})$ and $VV^T U(\tilde{\mu})$ is measured through the Schmidt-Eckart-Young theorem (Guo and Hesthaven, 2019).

Next, we perform the regression in the reduced space to make predictions in online stages. For that, we require another $N_r \in \mathbb{N}$ snapshots, $U\left(\hat{\mu}^1\right), \ldots, U\left(\hat{\mu}^{N_r}\right) \in \mathbb{R}^{N_h \times N_t}$, for $\hat{\mu}^1, \ldots, \hat{\mu}^{N_r} \in \mathcal{M}$, and project them into the reduced space to train our regression model. The remaining procedure proposes a tensor-decomposition-based regression including the following four steps:

4. We write parts of the projected training snapshots $V^T U\left(\hat{\mu}^1\right), \ldots, V^T U\left(\hat{\mu}^{N_r}\right) \in \mathbb{R}^{L \times N_t}$ in the tensor-decomposition matrices

$$D^l := \left[\left(V^T U\left(\hat{\mu}^1\right)\right)_l^T, \ldots, \left(V^T U\left(\hat{\mu}^{N_r}\right)\right)_l^T\right] = \left[\left(V_l^T U\left(\hat{\mu}^1\right)\right)^T, \ldots, \left(V_l^T U\left(\hat{\mu}^{N_r}\right)\right)^T\right] \in \mathbb{R}^{N_t \times N_r}, \tag{24}$$

for $l \in \{1, \ldots, L\}$, where, as notation, A_l declares the l-th column of a matrix A.

5. By singular value decompositions

$$D^l \approx \hat{D}^l = \sum_{k=1}^{d_l} \lambda_k^l \psi_k^l \left(\phi_k^l\right)^T, \tag{25}$$

of the tensor-decomposition matrices, for $l \in \{1, \ldots, L\}$, time and parameter spaces are decomposed. Here, the numbers $d_l \in \mathbb{N}$ denote the corresponding ranks of truncation calculated by the same ratio used in (21) with the singular values λ_k^l of $\hat{D}^l$. The columns $\psi_k^l \in \mathbb{R}^{N_t}$ and $\phi_k^l \in \mathbb{R}^{N_r}$, $k \in \{1, \ldots, d_l\}$, of the left- and right-singular factors correspond to the time and the parameter spaces, respectively.

6. For the predictions, we construct $2 \sum_{l=1}^{L} d_l$ regression models to approximate all ψ_k^l, ϕ_k^l for a new set of times and parameters. Practically, we apply Gaussian process regressions to calculate the regression functions

$$\hat{\psi}_k^l \;:\; \mathcal{T} \to \mathbb{R}, t \mapsto \hat{\psi}_k^l(t), \quad \text{from the training set } \left\{\left(t_i, \left(\psi_k^l\right)_i\right) \middle| i \in \{1, \ldots, N_t\}\right\}, \tag{26}$$

$$\hat{\phi}_k^l \;:\; \mathcal{M} \to \mathbb{R}, \mu \mapsto \hat{\phi}_k^l(\mu), \quad \text{from the training set } \left\{\left(\hat{\mu}^j, \left(\phi_k^l\right)_j\right) \middle| j \in \{1, \ldots, N_r\}\right\}, \tag{27}$$

for $k \in \{1, \ldots, d_l\}$ and $l \in \{1, \ldots, L\}$. Gaussian process regression—also known as Kriging—interpolates the training data and builds a fast-responding metamodel for a scalar output, see (Santner, 2003). The MATLAB routines `fitrgp` and `predict` serve to create the Gaussian process regressions and to make the predictions.

7. After these preparations, we can approximate new evaluations of u—and consequently U—through computing

$$u(t, \mu) \approx \hat{u}(t, \mu) := V \begin{pmatrix} \sum_{k=1}^{d_1} \lambda_k^1 \hat{\psi}_k^1(t) \hat{\phi}_k^1(\mu) \\ \vdots \\ \sum_{k=1}^{d_L} \lambda_k^L \hat{\psi}_k^L(t) \hat{\phi}_k^L(\mu) \end{pmatrix} \in \mathbb{R}^{N_h} \tag{28}$$

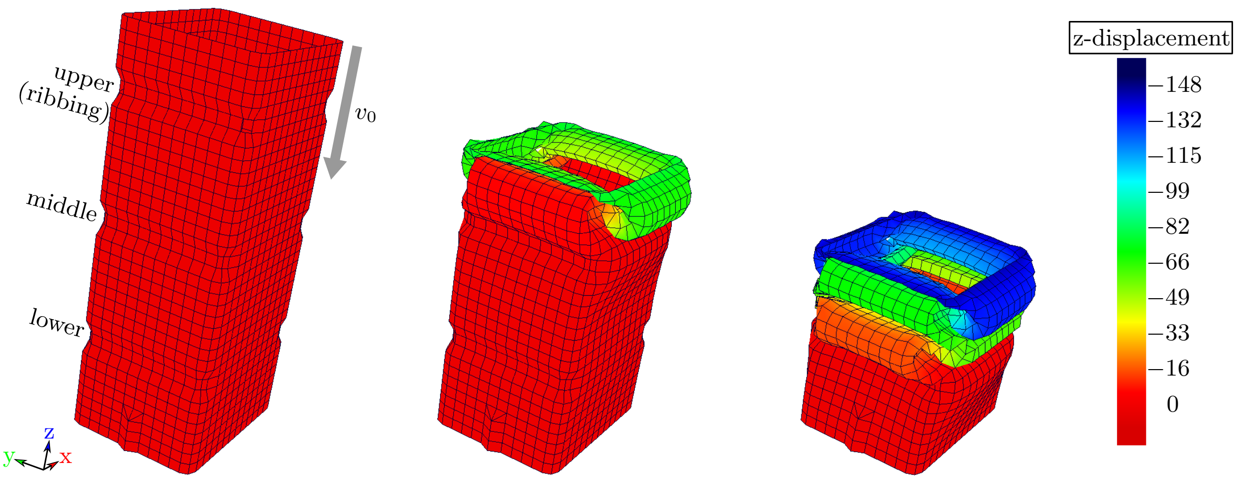
for $(t, \mu) \in \mathcal{T} \times \mathcal{M}$. Multiplying by V in (28) projects the approximated solution into the full space.

Following these seven steps, we are able to reconstruct a FES. The desired key results—further processed scalar outputs—can then be derived from the approximations in a fast manner. For the purpose of ET, this enables to quickly solve the optimizations (9) and (10) which lets the bottleneck disappear. In total $N_r + N_s$ snapshots were needed to run the non-intrusive parametric MOR approach.

3. Application of the MOR-ET Approach

3.1. Finite Element Simulation of a Crashbox Deformation Model

Using the resulting model for the system function (1) of ET leads to an accelerated and thus feasible execution of ET. To investigate this approach, we employ the benchmark problem of a crashbox deformation. This FES is publicly accessible on the LS-DYNA homepage, see (LS-DYNA, 2020), and run in the LS-DYNA crash solver.



(a) Time state 1 (b) Time state 11 (c) Time state 22

Figure 1. Different time states of the LS-DYNA crashbox simulation with input $\mu = (2.7, 8.8, 0.7, -0.6, -1.2)$. The colors indicate the amount of the nodes z-displacement.

In the LS-DYNA crashbox model, a symmetric tube-shaped crashbox with six ribbings is impacted by a plate moving in negative z-direction, see Fig. 1(a)-(c). The crashbox is about 272 mm tall, 116 mm wide, and 96 mm long. Each two ribbings are located opposite one another at the same height. The model consists of $N_{\mathrm{nd}} = 1925$ nodes, i.e. there are $N_h = 3 \cdot N_{\mathrm{nd}} = 5775$ degrees of freedom. The LS-DYNA solver writes $N_t = 22$ time states of the interval $\mathcal{T} = [0, 20]$ in the .d3plot output file.

With the post-processor software (Animator4, 2019), one can view the simulation video as well as convert the .d3plot output file into a .a4db output format that is readable in MATLAB. To load the required N_h x-,y-,z-displacements of the nodes for every time state from the .a4db file into

Table I. Inputs of the LS-DYNA crashbox model.

Input	Interval	Unit	Uncertainty type	Description
μ_1	$[2,3]$	mm	epistemic	wall thickness
μ_2	$[8.5,9]$	$\frac{\text{m}}{\text{s}}$	aleatory	initial velocity
μ_3	$[-3,1]^*$	mm	epistemic	relative upper ribbings depth
μ_4	$[-3,1]^*$	mm	epistemic	relative middle ribbings depth
μ_5	$[-3,1]^*$	mm	epistemic	relative lower ribbings depth

* The interval will be trimmed later by the expert, see Fig. 2.

MATLAB, we use the routines `h5info` and `h5read`. Time is measured in milliseconds, displacements in millimeters, the velocity in meters per second, and the mass in kilograms.

We consider $n = 5$ inputs, $\mu = (\mu_1, \mu_2, \mu_3, \mu_4, \mu_5)$, see Table I. The first input dimension characterizes the wall thickness of the crashbox which varies between two and three millimeters, i.e. $\mu_1 \in I_1 = [2,3]$. The initial velocity of the moving plate lies in the interval $I_2 = [8.5, 9]$ described by the second input μ_2. Entries μ_3, μ_4, and μ_5 represent the depths of the respective opposite upper, middle, and lower ribbings, see Fig. 1, relative to the original configuration. They can be varied within $I_i = [-3,1], i \in \{3,4,5\}$. Negative values of $\mu_i, i \in \{3,4,5\}$, indicate that the respective ribbings do not sink as deep as the ribbings of the default crashbox. Positive values have the reverse effect. Based on the uncertain variables $\mu_1, \mu_3, \mu_4, \mu_5$, statements can be made about the epistemic component robustness. The velocity, μ_2, is considered to be an irreducible aleatory uncertainty since it scatters in each experiment.

3.2. Application of the MOR-ET Approach to a Crashbox Simulation

We apply our proposed MOR-ET approach to the maximal z-deformation of the crashbox simulation. This scalar key result is measured at the last time step by one specific node corresponding to the plate—in our FES, this node defines the plate. More precisely, the maximal z-deformation of the crashbox is equal to the final z-displacement of the plate node plus the initial z-distance between the plate and the crashbox of 1.517 mm. Adding the initial distance is correct since the plate is moving in negative z-direction and therefore, all displacements are negative in the example. In this case, the extraction function for the key result is straightforward. Since we consider only one key result for this purpose, conventional metamodels could also be applied. However, we will take this example as a proof of concept.

The maximal z-deformation of the crashbox simulation is taken as the ET system function f. The function f has a black-box character as it is derived by the FES. We follow the six points explained in Section 2.1 to obtain the plausibility and belief curve:

1. & 2. First, a fictive expert assesses the intervals $I_1, \ldots, I_5$ by splitting them into focal elements and providing BPAs, see Fig. 2. Regarding the ribbings, the expert wants the crashbox to deform first above the upper ribbing, second between upper and the middle ribbing, and third at the remaining parts of the crashbox. For that reason, the intervals I_3, I_4, and I_5 are cut.

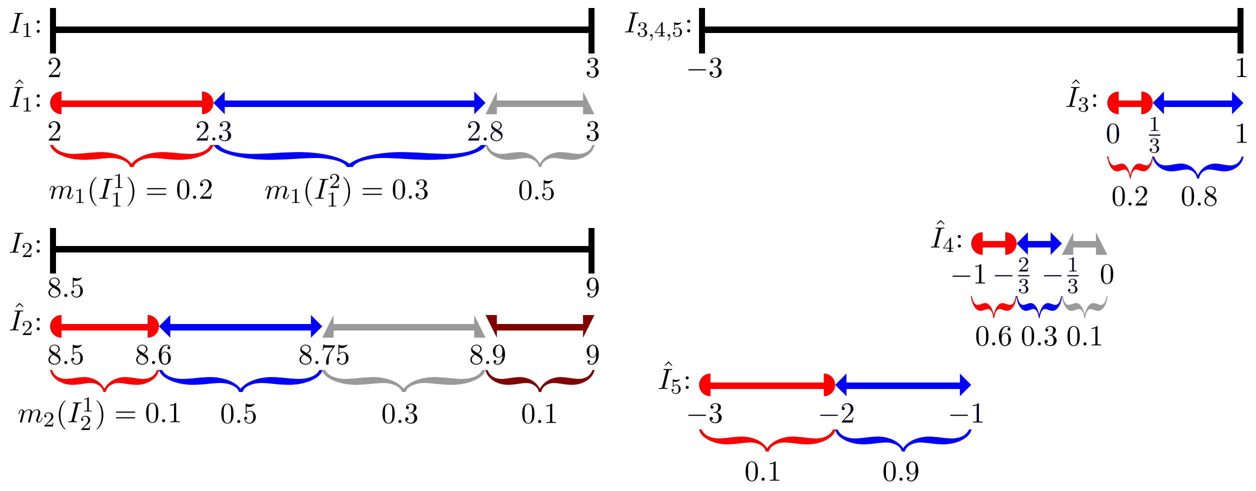


Figure 2. An expert divides the intervals into focal elements and allocates BPAs.

3. & 4. Interval cells are formed and the corresponding composite BPAs are computed, see Fig. 3 for some examples. In total, there are $k = 3 \cdot 4 \cdot 2 \cdot 3 \cdot 2 = 144$ interval cells.

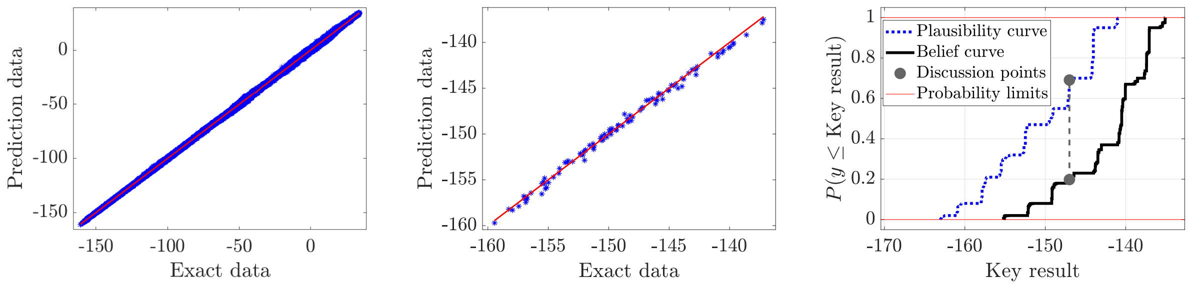
$$C^1 = I_1^1 \times I_2^1 \times I_3^1 \times I_4^1 \times I_5^1 = \text{●} \times \text{●} \times \text{●} \times \text{●} \times \text{●} \quad \text{with } p^1 = 0.2 \cdot 0.1 \cdot 0.2 \cdot 0.6 \cdot 0.1 = 2.4 \cdot 10^{-4},$$
$$C^2 = I_1^1 \times I_2^1 \times I_3^1 \times I_4^1 \times I_5^2 = \text{●} \times \text{●} \times \text{●} \times \text{●} \times \text{◆} \quad \text{with } p^2 = 0.2 \cdot 0.1 \cdot 0.2 \cdot 0.6 \cdot 0.9 = 0.0022,$$
$$\vdots$$
$$C^{144} = I_1^3 \times I_2^4 \times I_3^2 \times I_4^3 \times I_5^2 = \text{▲} \times \text{▼} \times \text{◆} \times \text{▲} \times \text{◆} \quad \text{with } p^{144} = 0.5 \cdot 0.1 \cdot 0.8 \cdot 0.1 \cdot 0.9 = 0.0036.$$

Figure 3. Some interval cells and their composite BPAs.

5. To accelerate the optimizations (9) and (10), we approximate the system function f by non-intrusive MOR as introduced in Section 2.2. In total, 200 simulations of the full crash simulation with different input parameters μ are run. We use $N_s = 20$ simulations to set up the snapshot matrix S in (18). After the singular value composition and choosing $\epsilon = 0.9999$, $L = 8$ is calculated by (21). A lower value of ϵ leads to a lower value of L, e.g. $\epsilon = 0.999$ implies $L = 4$, which can yield a lower approximation quality. Another $N_r = 80$ simulations train the tensor-decomposition-based regressions, compare to (26) and (27). The remaining 100 full-scale simulations are taken for validation. Each of the three samples is created in Latin Hypercube style (McKay et al., 1979).

   The MOR approach predicts the N_h degrees of freedoms for all N_t time states with a high approximation quality for the validation sample. Fig. 4(a) shows the regression plot with $N_h \cdot N_t \cdot 100 = 12\,705\,000$ points. A second regression plot, see Fig. 4(b), focuses on the key result of the validation sample, i.e. the maximal z-deformation. The $2k$ optimization problems for the different interval cells are now quickly solved by applying MATLAB's routine `particleswarm` on the MOR surrogate model and its negative counterpart. These took about seven hours wall time on a common office laptop.

6. Now, it is a straightforward task to compile the plausibility and belief curve as well as plot them, see Fig. 4(c).



(a) Regression plot for all degrees of freedom. (b) Regression plot for the considered key result. (c) Plausibility and belief curve.

Figure 4. Regression plots for the 100 validation simulations and the resulting plausibility and belief curve.

4. Application Results and Discussion

The first four steps of the ET remain unchanged: Experts specify and assess the uncertainties and—according to their opinions—interval cells are obtained. In general, the more subintervals and the more finely defined these intervals are, the more the belief curve and the plausibility curve converge against each other. In other words, if the experts choose to subdivide the initial intervals in Fig. 2 more detailed, the ET curves in Fig. 4(c) slide closer together. However, the number of optimization problems to be solved would increase strongly.

We focus on the regression process in step five. The reconstruction of the crashbox simulation by non-intrusive MOR is considered sufficiently good. The whole simulation is approximated by a coefficient of determination equal to 0.9998—calculated from the 100 validation simulations. The coefficient of determination for the key result has the value 0.9952. The slightly lower number can be explained by the different ranges of the considered outputs. Note that the regression plot, Fig. 4(b), shows a much smaller range than the plot in Fig. 4(a). This causes deviations to appear larger and the coefficient of determination hence is lower.

To compute the infimum and supremum values of the k interval cells, the MATLAB routine `particleswarm` calls the MOR key result function 345 450 times. If we assume that we require the same number of full-scale simulations to find these values, the LS-DYNA simulation would have needed to be run over 345 thousand times. This would have taken longer than 2100 hours since one simulation has an approximate CPU time of 22 seconds. Instead, we only ran 100 full FES to train the MOR key result function and took another 100 to validate our approximation. Together with the seven hours of solving the optimizations, this adds up to less than nine hours for running the proposed MOR-ET approach.

We now proceed to interpret the results of our uncertainty analysis. Since a deformation of less than one millimeter is not relevant for us, we leave out the decimal places and round to integers.

The smallest point of the plausibility curve is located on the far left at −163 mm in Fig. 4(c). The largest point of the belief curve is located on the far right at −135 mm. These two values define the target set $[-163, -135]$ of the key result due to the uncertain inputs. This means that the key result can take on values within this interval. The courses of the plausibility and belief curve state how likely certain outputs are—with two different measures.

Let us exemplarily consider the key result −147 mm on the x-axis in Fig. 4(c), see the marked discussion points. On the one hand, the plausibility curve has the y-value 0.69 at −147 mm. Equation (14) thus yields the upper bound 0.69 for the cumulative distribution function of the key result at −147 mm, i.e. it holds

$$P(y \leq -147) \leq \mathrm{Pl}(-147) = 0.69. \tag{29}$$

On the other hand, the y-value of the belief curve is equal to 0.20 at −147 mm. Equation (13) therefore defines a lower bound for the cumulative distribution function, i.e. one obtains

$$0.20 = \mathrm{Bel}(-147) \leq P(y \leq -147). \tag{30}$$

Together, this limits the value of the cumulative distribution function at −147 mm to be between 0.20 and 0.69.

For a clearer understanding, we introduce the terms plausible and believable. Concerning the exemplary data points just mentioned, we define these terms as follows. The plausibility curve says it is 69% plausible that the key result is smaller than or equal to −147 mm. In contrast, the belief curve states it is 20% believable that the key result is smaller than or equal to −147 mm. We regard a smaller key result as worse than a larger. Depending on the measure, it is 69% plausible or 20% believable that the highest—and therefore best—crashbox z-deformation will not be larger—and hence not be better—than −147 mm. In summary, the plausibility curve here is the worst case approximation of the key result cumulative distribution function whereas the belief curve is the best case approximation.

As a practical interpretation, note that the −272 mm high crashbox suffers a deformation between −163 mm and −135 mm. Although it is 0% believable that the deformation is less than −157 mm, it is still 20% plausible that it is less than or equal to −157 mm, see Fig. 4(c). Therefore, we can not limit the target set by excluding its boundary values. For vehicle safety engineers, this means that the deformation varies by 28 mm due to the uncertainties which is more than 10% of the total height of the crashbox. Because of this high percentage, we advise engineers to reduce the epistemic uncertainties.

5. Conclusion

Uncertainties are inherent in automotive hardware crash tests. Especially in early phase development, finite element models are used to simulate different crash scenarios. The uncertainties of the hardware tests must be taken into account in the simulations.

The Evidence Theory (ET) was utilized to propagate uncertain inputs through an LS-DYNA finite element crash simulation. As inputs, we chose the wall thickness of the crashbox, the initial velocity of the moving plate and the depths of three ribbings. The wall thickness and the ribbing depths are epistemic uncertainties that are eliminated in later stages of development. The initial velocity is an aleatory input due to deviations in the test procedure.

To enable the computationally expensive optimization step of the ET, the time-dependent crash simulation was approximated by a non-intrusive parametric Model Order Reduction (MOR) approach. This regression method reconstructs the simulation for new input parameter sets in a fast and efficient way. Using this reduced model, the plausibility and belief curve of ET were computed. Output uncertainties as well as bounds of their distribution were derived by these curves. This information can then be used by vehicle safety engineers to assess the performance of the crashbox.

While the approximation results were satisfactory for the purpose of this study, it remains an open question how the regression quality of the MOR approach compares to conventional metamodels. A comparative study is hence proposed for future research. The MOR approach learns the behavior of the whole simulation and then extracts the key result. The basic physics of the system is therefore integrated in the regression model. In contrast, usual metamodels only process the key result itself, without any knowledge about the underlying system. This may has a positive effect for the approximation quality of the MOR approach. Moreover, if several key results are of interest, they can be extracted from one central model instead of creating different individual metamodels—one for each key result.

We are aware that the example simulation of this study is smaller in scope than present industrial problems. Engineers face total vehicle simulations that include millions of finite elements, complex material behavior, and element failures. We think that our work is an important step in tackling this challenge.

References

Animator4. Version 2.4.1. Braunschweig, Niedersachsen: Gesellschaft für numerische Simulation mbH, 2019.

Auer, E., W. Luther, G. Rebner, and P. Limbourg. A Verified MATLAB Toolbox for the Dempster-Shafer Theory. *Proceedings of the Workshop on the Theory of Belief Functions*, 2010.

Bae, H.-R., R. V. Grandhi, and R. A. Canfield. An Approximation Approach for Uncertainty Quantification Using Evidence Theory. *Reliability Engineering & System Safety*, 86(3):215–225, 2004.

Beyer, H.-G. and B. Sendhoff. Robust Optimization – A Comprehensive Survey. *Computer Methods in Applied Mechanics and Engineering*, 196(33—34):3190–3218, 2007.

Der Kiureghian, A. and O. Ditlevsen. Aleatory or Epistemic? Does It Matter?. *Structural Safety*, 32(2):105–112, 2009.

Eldred, M. S. and L. P. Swiler. Efficient Algorithms for Mixed Aleatory-epistemic Uncertainty Quantification with Application to Radiation-hardened Electronics, Part I : Algorithms and Benchmark Results. Technical Report SAND2009-5805, Sandia National Laboratories, 2009.

Eldred, M. S., L. P. Swiler, and G. Tang. Mixed Aleatory-Epistemic Uncertainty Quantification with Stochastic Expansions and Optimization-based Interval Estimation. *Reliability Engineering & System Safety*, 96(9):1092–1113, 2011.

Freedman, D. and P. Diaconis. On the Histogram as a Density Estimator: L_2 Theory. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 57:453–476, 1981.

Guo, M. and J. S. Hesthaven. Data-driven Reduced Order Modeling for Time-dependent Problems. *Computer Methods in Applied Mechanics and Engineering*, 345:75–99, 2019.

Guo, M. and J. S. Hesthaven. Reduced Order Modeling for Nonlinear Structural Analysis Using Gaussian Process Regression. *Computer Methods in Applied Mechanics and Engineering*, 341:807–826, 2018.

Helton, J. C., J. D. Johnson, W. L. Oberkampf, and C. B. Storlie. A Sampling-Based Computational Strategy for the Representation of Epistemic Uncertainty in Model Predictions with Evidence Theory. Technical Report SAND2006-5557, Sandia National Laboratories, 2006.

Knuth, K. H. Optimal Data-based Binning for Histograms and Histogram-based Probability Density Models. *Digital Signal Processing*, 95:102581, 2019.

Le Guennec, Y., J.-P. Brunet, F.-Z. Daim, M. Chau, and Y. Tourbier. A Parametric and Non-intrusive Reduced Order Model of Car Crash Simulation. *Computer Methods in Applied Mechanics and Engineering*, 338:186–207, 2018.

LS-DYNA. Examples Homepage, https://www.dynaexamples.com. Livermore, California: Livermore Software Technology Corporation, 2020.

LS-DYNA. Version R9.3. Livermore, California: Livermore Software Technology Corporation, 2018.

MATLAB. Version 9.5.0.944444 (R2018b). Natick, Massachusetts: The MathWorks Inc., 2018.

McKay, M. D., R. J. Beckman, and W. J. Conover. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*, 21(2):239—245, 1979.

Nikolaidis, E., D. M. Ghiocel, and S. Singhal. Engineering Design Reliability Handbook. CRC Press, Boca Raton, Florida, 2004.

Oberkampf, W. L. and J. C. Helton. Investigation of Evidence Theory for Engineering Applications. *43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2002.

Oberkampf, W. L., J. C. Helton, and K. Sentz. Mathematical Representation of Uncertainty. *19th AIAA Applied Aerodynamics Conference*, 2001.

Oberkampf, W. L., J. C. Helton, C. A. Joslyn, S. F. Wojtkiewicz, and S. Ferson. Challenge Problems: Uncertainty in System Response Given Uncertain Parameters. *Reliability Engineering & System Safety*, 85(1–3):11–19, 2004.

Santner, T. J., B. J. Williams, and W. I. Notz. The Design and Analysis of Computer Experiments. Vol. 1, Springer, New York City, 2003.

Scott, D. W. On Optimal and Data-based Histograms. *Biometrika*, 66:605-–610, 1979.

Sentz, K. and S. Ferson. Combination of Evidence in Dempster-Shafer Theory. Technical Report SAND2002-0835, Sandia National Laboratories, 2002.

Shafer, G. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.

Swiler, L. P. and A. A. Giunta. Aleatory and Epistemic Uncertainty Quantification for Engineering Applications. *Proceedings of the Joint Statistical Meetings, American Statistical Association*, 2007.

Swiler, L. P., T. L. Paez, and R. L. Mayes. Epistemic Uncertainty Quantification Tutorial. *Proceedings of the IMAC-XXVII*, 2009.

Swischuk, R., L. Mainini, B. Peherstorfer, and K. Willcox. Projection-based Model Reduction: Formulations for Physics-based Machine Learning. *Computers & Fluids*, 179:704–717, 2019.

Tian, W., P. de Wilde, Z. Li, J. Song, and B. Yin. Uncertainty and Sensitivity Analysis of Energy Assessment for Office Buildings Based on Dempster-Shafer Theory. *Energy Conversion and Management*, 174:705–718, 2018.