Parametric Projection-based Model Order Reduction For Crash

Mathias Lesjak[1], Fabian Duddeck[2]

[1]BMW Group, Research and Innovation Center, Knorrstraße 147, 80788 Munich, Germany
[2]Technical University of Munich, TUM School of Engineering and Design, Arcisstraße 21, 80333 Munich, Germany

1 Summary of the Presentation

Modern passive safety development is associated with numerous simulations and hardware tests. In virtual development, multi-query analysis such as optimization, sensitivity analysis and robustness studies are performed. These methods require many simulation evaluations, which can make their application impractical for large simulation models. An approach to make the development process more efficient is Model Order Reduction (MOR) which uses already generated simulation data to build a Reduced-Order Model (ROM) and accelerate future simulations.

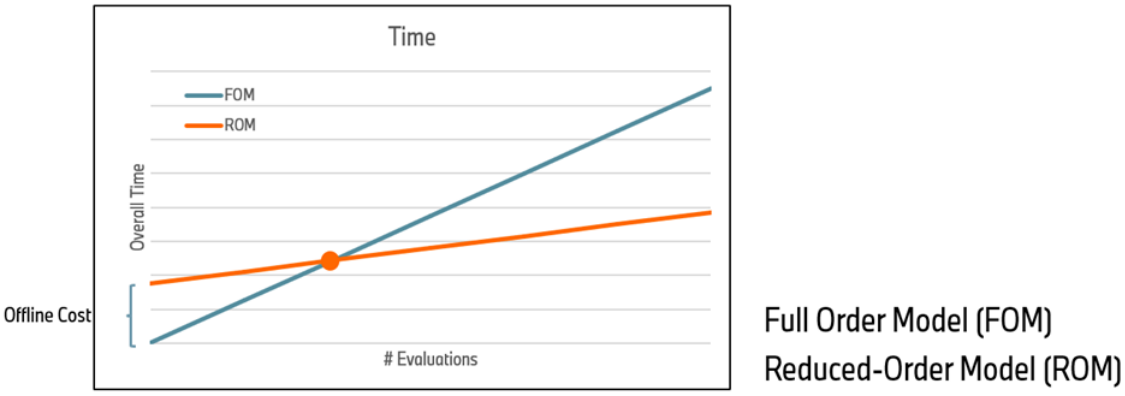


Fig.1: Economic perspective on MOR.

As can be seen in Fig.1, MOR is especially useful if the number of simulation evaluations is high. In addition, the lower execution time of the ROM must compensate for the offline cost, which are all computations related to the creation of ROM. This work is concerned with projection-based MOR (pMOR), which is a MOR method where the solution is restricted to lie in a low-dimensional subspace first, afterwards the Finite Element (FE) equations are modified and simplified to be cheaper to solve.
Bach et al. [1] showed the successful application of pMOR to large nonlinear dynamical systems as appearing in crash and impact simulations. However, the study is limited to reproductive examples. That is, no parameter variations are allowed, and the ROM is tested on the training data.

Parameter variations introduce additional complexity, as trajectories in state space choose different paths and a single linear dimensionality reduction, such as Proper Orthogonal Decomposition (POD), is not able to approximate the data using few basis vectors [2]. Amsallem et al. [2] introduce local Reduced-Order Bases (lROB), which approximates the data linearly, however, divides it first into subregions. Current research also utilizes Neural Networks (NN) in an Autoencoder (AE) architecture to perform nonlinear MOR. All three methods, global POD, lROB and AE are applied to a crashbox example, where the tube thickness and the mass of the plate is varied as can be seen in Fig.2. The data is divided into 27 training simulations, which are used to perform the dimensionality reduction and 3 test simulations, where the accuracy of the ROMs is evaluated. The test points are chosen, such that a test case with a high total deformation and one test case with a low total deformation is present. The evaluation is carried out on the highly deforming test case, as experience has shown it is the more severe case.

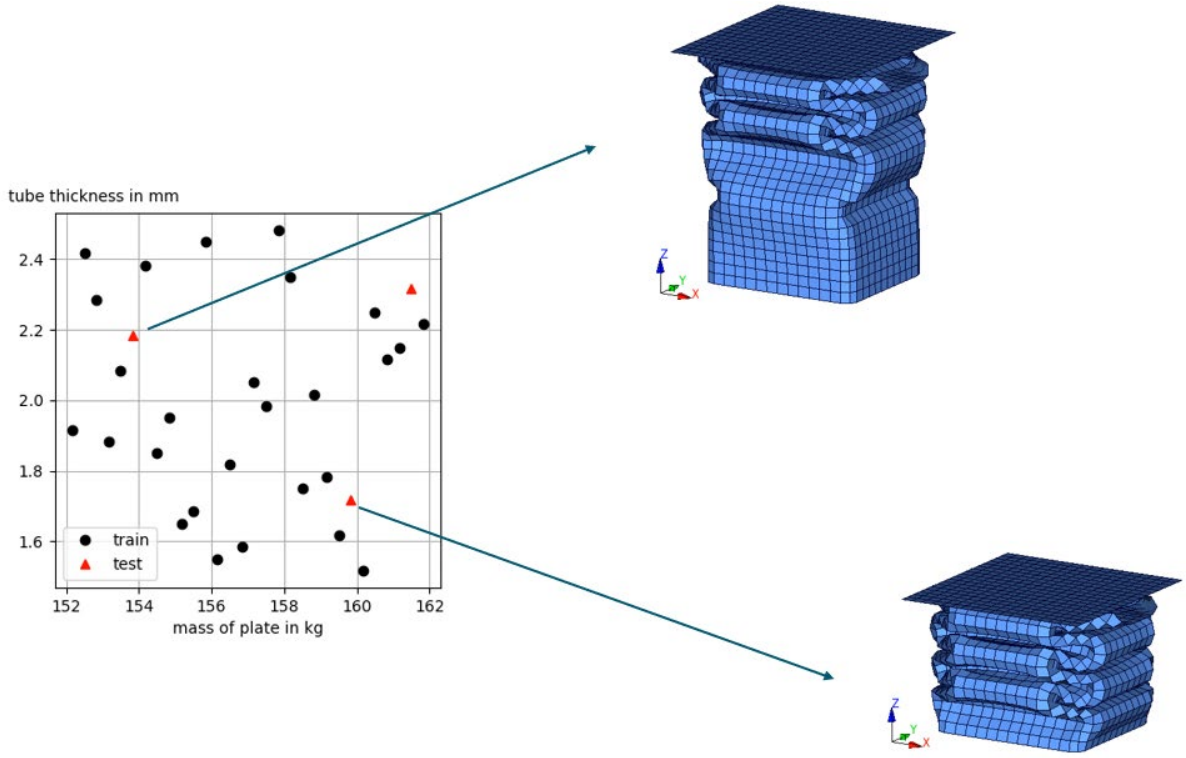


Fig.2: Parameter variations of crash box example.

After the training data is generated, the accuracy of the dimensionality reduction is assessed for an increasing number of dimensions. Fig.3 shows the approximation error of the training data for an increasing number of dimensions k. It can be seen, that the local bases approximate the respective data more accurately than the global basis except local basis6, which is the basis associated to the initial time steps of the simulation. The AE is able to represent the data in a global sense using less dimensions than the linear global basis.

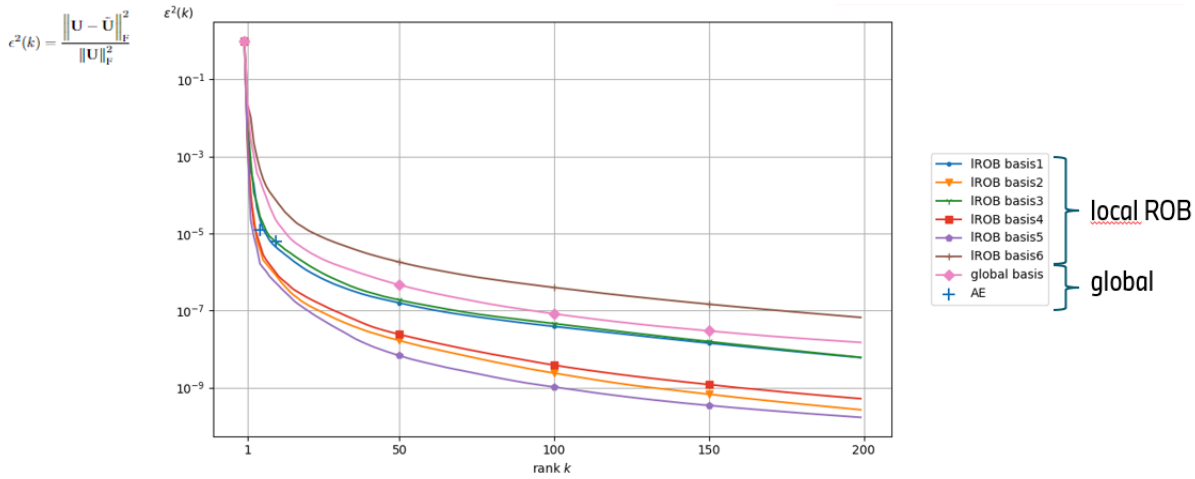


Fig.3: Approximation error of the dimensionality reduction method for an increasing number of dimensions k.

Next, the low-dimensional representations are used to construct the ROMs and their accuracy is assessed for the highly deforming test case.

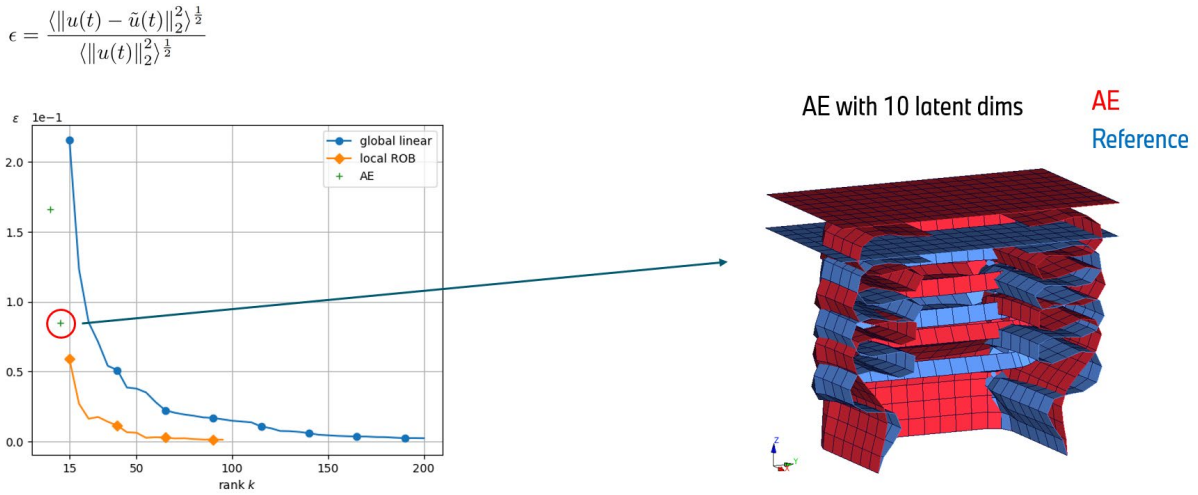
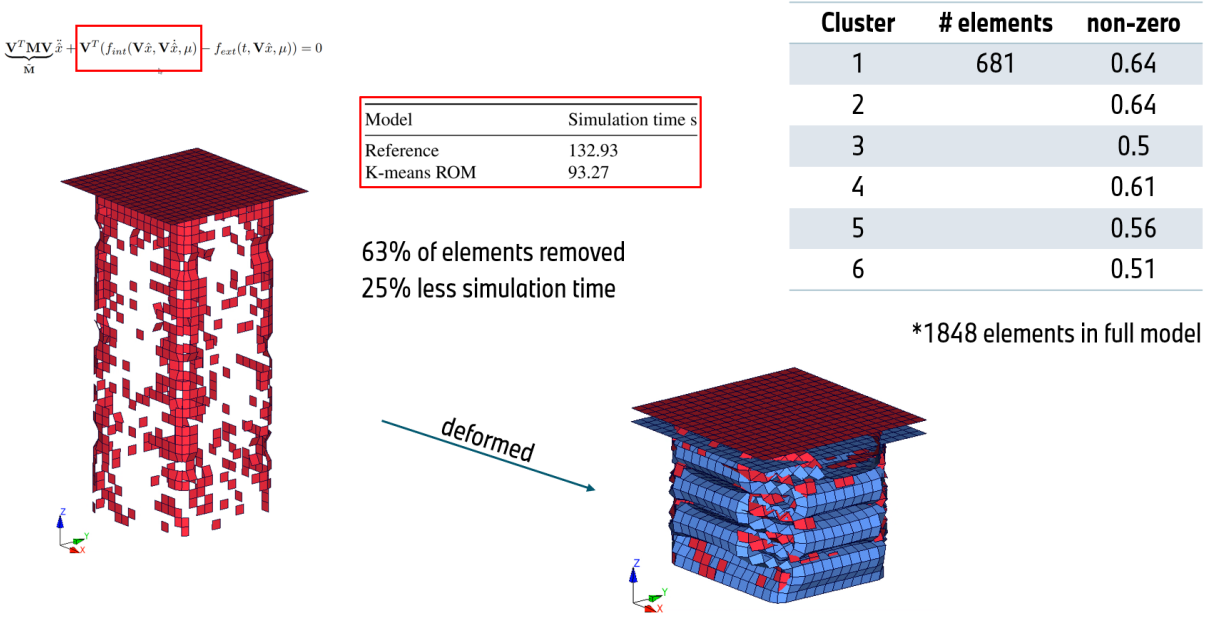


Fig.4: ROM accuracy for an increasing number of dimensions.

Fig.4 shows that the error decays slower for the globally linear ROM than for the IROB ROM. Nonlinear ROMs require an additional reduction step to achieve computational speedup which scales with the dimension of the ROM. Therefore, the global linear ROM is unsuitable for highly nonlinear parametric problems. The AE ROM can further reduce the dimension of the ROM to 5 or 10 dimensions. However, additional error is introduced. Comparing the complexity of AE and the IROB method, the latter is the preferred one. In addition, desirable linear ROM properties are preserved.

Since IROB is identified to be the most suitable method, energy conserving sampling and weighting hyper-reduction is applied and the speedup is measured.



Cluster	# elements	non-zero
1	681	0.64
2		0.64
3		0.5
4		0.61
5		0.56
6		0.51

*1848 elements in full model

Fig.5: Hyper-reduced crash box model in undeformed and deformed configuration.

Fig.5 shows the hyper-reduced crash box model in the initial and in the deformed configuration. The size of the model is reduced from 1848 elements to 681 elements. In addition, Fig.5 lists the number of non-zero elements in each subregion/cluster. However, the zero elements must be carried along the simulation to save the history variables. The removal of the elements results in a reduction of simulation time by 25%.

This is because element processing accounts for about 50% of the total simulation time. An additional mechanism to achieve speedup is the adjustment of the critical time step. However, this is not addressed in this article.

Finally, Implementation details are given for the IROB ROM and the AE ROM. We begin with IROB where elements are removed for hyper-reduction. Since hyper-reduction only affects the internal force vector, contact forces and the mass calculation remains unchanged. The removed elements are assigned **`*MAT_NULL.`** This material definition fulfills the previously defined requirements and yields a correct assembly of the mass matrix, assigns a contact stiffness to the elements and automatically adjusts the iteration in the element processing routines, as exemplarily shown in Fig.6. The identification of elements to assign the correct weighting factors to the remaining ones is achieved by calculating the external id using the built-in function ide_ext = lqfinv8(ide_int,4). The transfer of data such as ROBs, weighting factors or the reduced element set is achieved using the hdf format. Simulation options are specified and passed to the simulation using the JSON format.

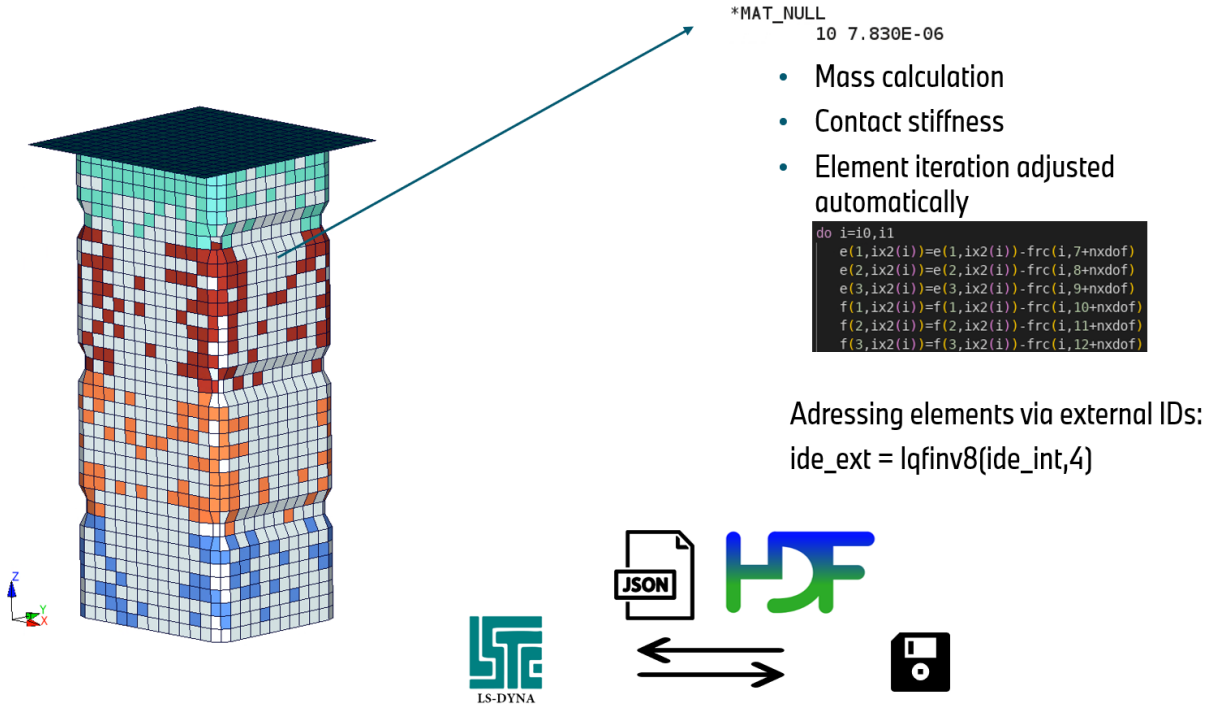


Fig.6: Implementation details of hyper-reduction.

The AE ROM requires the evaluation of the AE NN during the runtime of LS-Dyna. Most ML frameworks offer a powerful Python API, that can be used to implement different architectures easily. Querying the AE model implemented in Python during runtime of LS-Dyna is achieved by the illustrated Fortran-Python interface. As can be seen in Fig.7, multiple layers are introduced to pass data directly from Fortran to Python and vice versa. The first layer defines a Fortran – C interface, which is realizable using Fortran iso_c_binding module definitions that ensure interoperability. As the most commonly used Python implementation is in C and a C-API of Python exists, we have almost reached our goal. The last challenge is to use the Python C-API to correctly wrap the raw data arrays from Fortran and C into Python Objects and call the desired module function, which is the “test” function in this illustration.

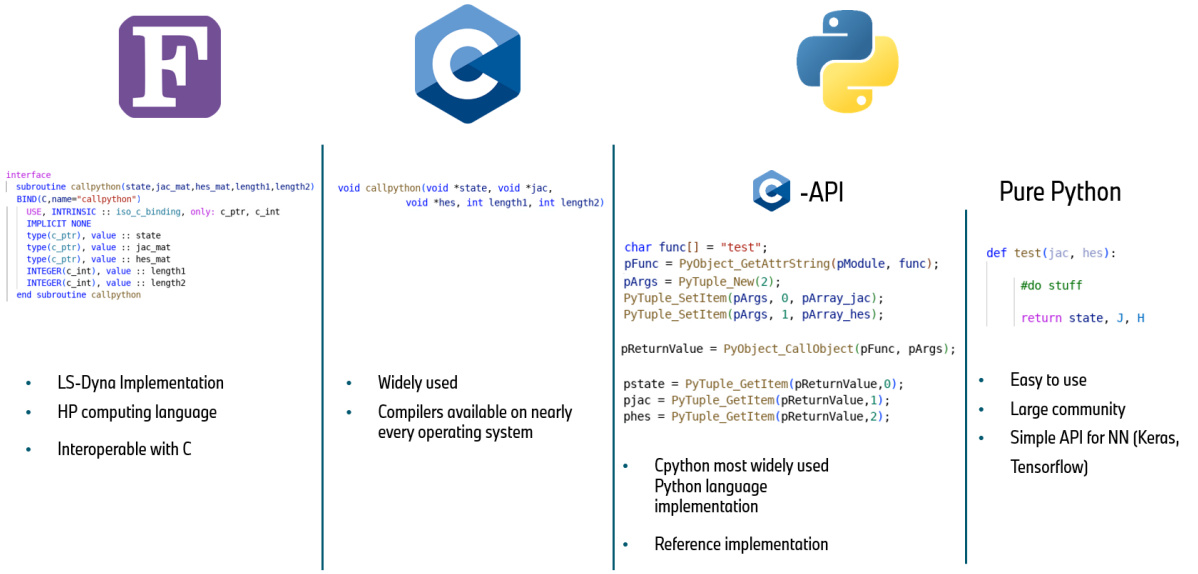


Fig.7: Fortran – Python interface.

To summarize, three different types of ROMs are compared. A linear global ROM, the IROB ROM which is a linear local ROM and the global nonlinear AE ROM. We show that error of the global linear ROM decays slowly with an increasing number of used dimensions, which leads to a high-dimensional ROM. Due to the high dimensionality of the ROM, hyper-reduction is impractical and no speedup is achievable. The AE ROM can resolve the latent dimensions better, however, this method is still a research topic. The hyperparameter tuning of the AE is associated with expert knowledge, the ROM formulation is still questionable, the implementation is sophisticated and hyper-reduction has yet to be tested.
LROB proves to effectively divide the solution space into subregions which can be approximated well by linear dimensionality reduction using less dimensions. IROB is successfully combined with ECSW hyper-reduction, demonstrating a 25% reduction in simulation time. There is further potential for speedup as zero-elements could be eliminated and the stable time step for ROMs is usually larger [3].

2 Literature

References should be given in the last paragraph of your manuscript. Please use following scheme:

[1] Bach, C., Ceglia, D., Song, and L., Duddeck, F.: "Randomized low-rank approximation methods for projection-based model order reduction of large nonlinear dynamical problems", 118, 2019, 209-241

[2] Amsallem, D., Zahr, M.J., and Farhat, C., „Nonlinear model order reduction based on local reduced-order bases", Int. J. Numer 92, 2012, 891-916

[3] Farhat, C., Avery, P., Chapman, T., and Cortial, J., Dimensional reduction of nonlinear finite elment dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency", International Journal for Numerical Methods in Engineering, 98, 2014, 625–662