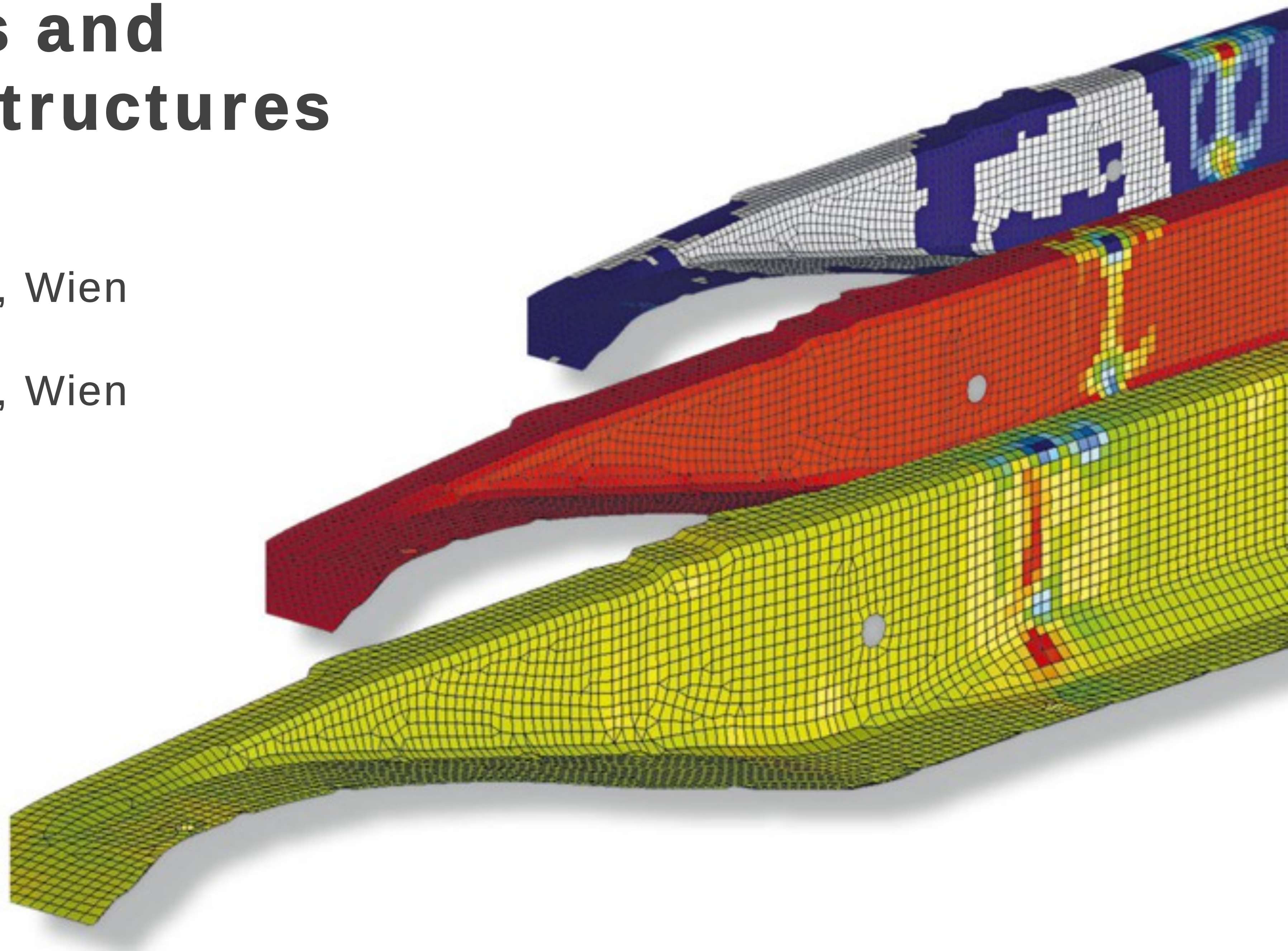
Recent Developments for Random Fields and Statistics on Structures



Christian Bucher
DYNARDO Austria GmbH, Wien
Sebastian Wolff*
DYNARDO Austria GmbH, Wien
Florian Quetting
Daimler AG, Sindelfingen

Introduction

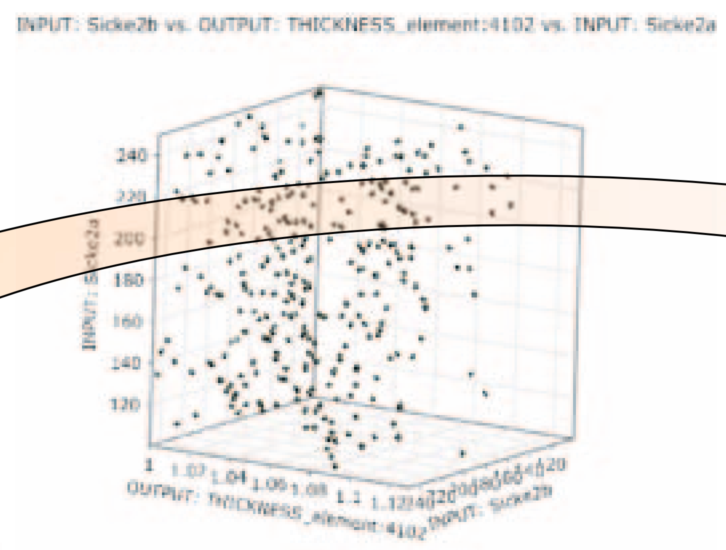
- Spatial variation of structural properties lead to uncertainties in the structural performance
- Slightly different variation of results in each node/element of the structural mesh
- Large number of random variables, can be described as random field
- For statistical analyses it is important to reduce the number of random variables
- For engineering interpretation it is helpful to reduce noise and keep essential features

Motivation: SoS 2

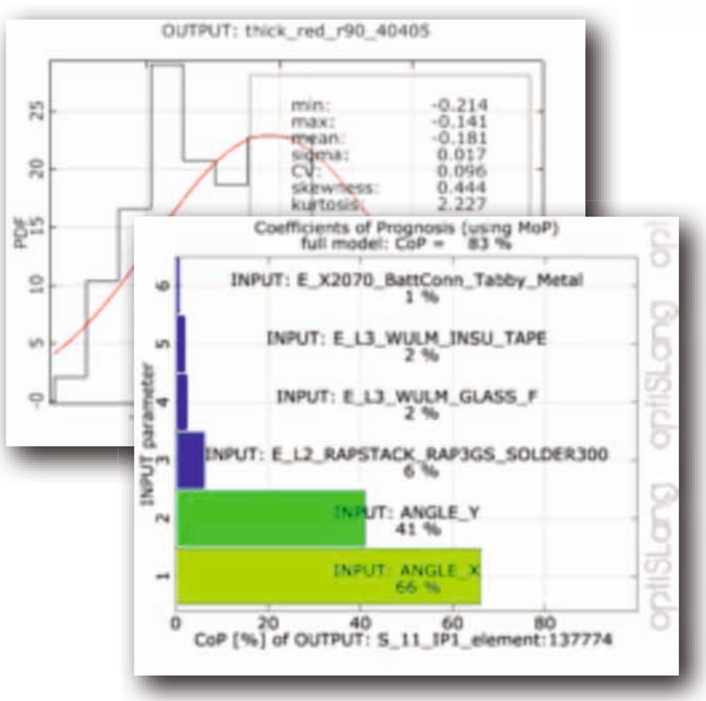
- Random influences cause spatially distributed random results on structures. Engineers need to evaluate the statistics on the structure to locate „hot spots“ of variation and investigate correlations
- SoS is a post processor for statistics on finite element structures
  - Visualization of descriptive statistics on the structure
  - Visualization of correlations and CoD between random input and structural results
  - Visualization of quality performance (QCS)
  - Identification of spatial dependencies using Random Fields
- Limitations:
  - memory usage
  - cpu time
  - no large FEM meshes

Integration of SoS 2 and optiSLang

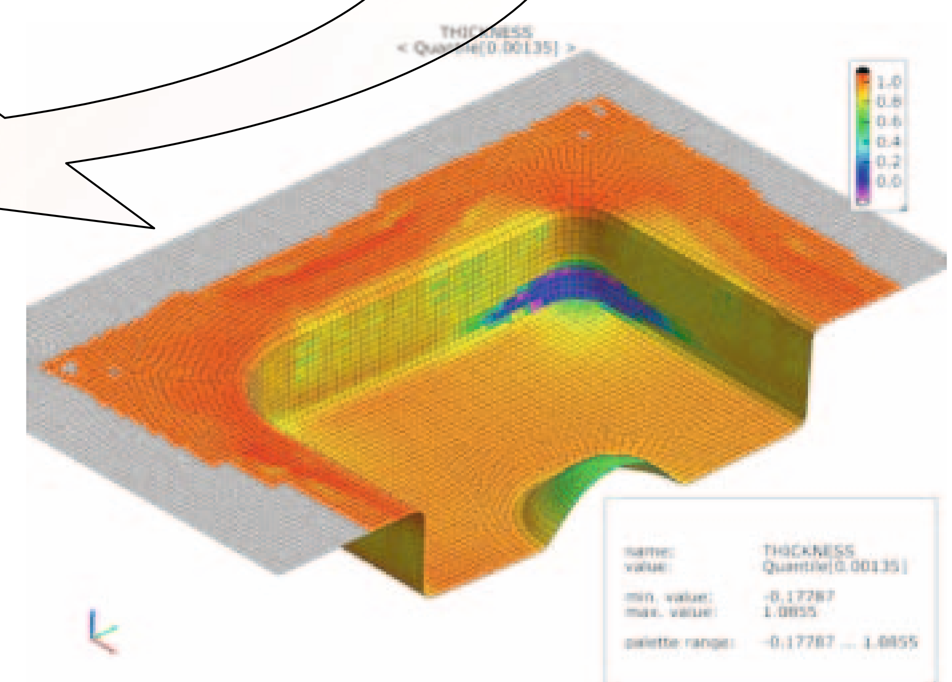
Process:
Import data (multiple simulation runs or measurements)



SoS: Visualize variation, correlation, identify random field shapes



Export local statistics and imperfection amplitudes to optiSLang for further post-processing



Random field

- Real-valued function in n-dimensional space

$$H \in \mathbb{R}; \quad \mathbf{x} = [x_1, x_2, \ldots x_n]^T \in \mathcal{D} \subset \mathbb{R}^n$$

- Mean value function

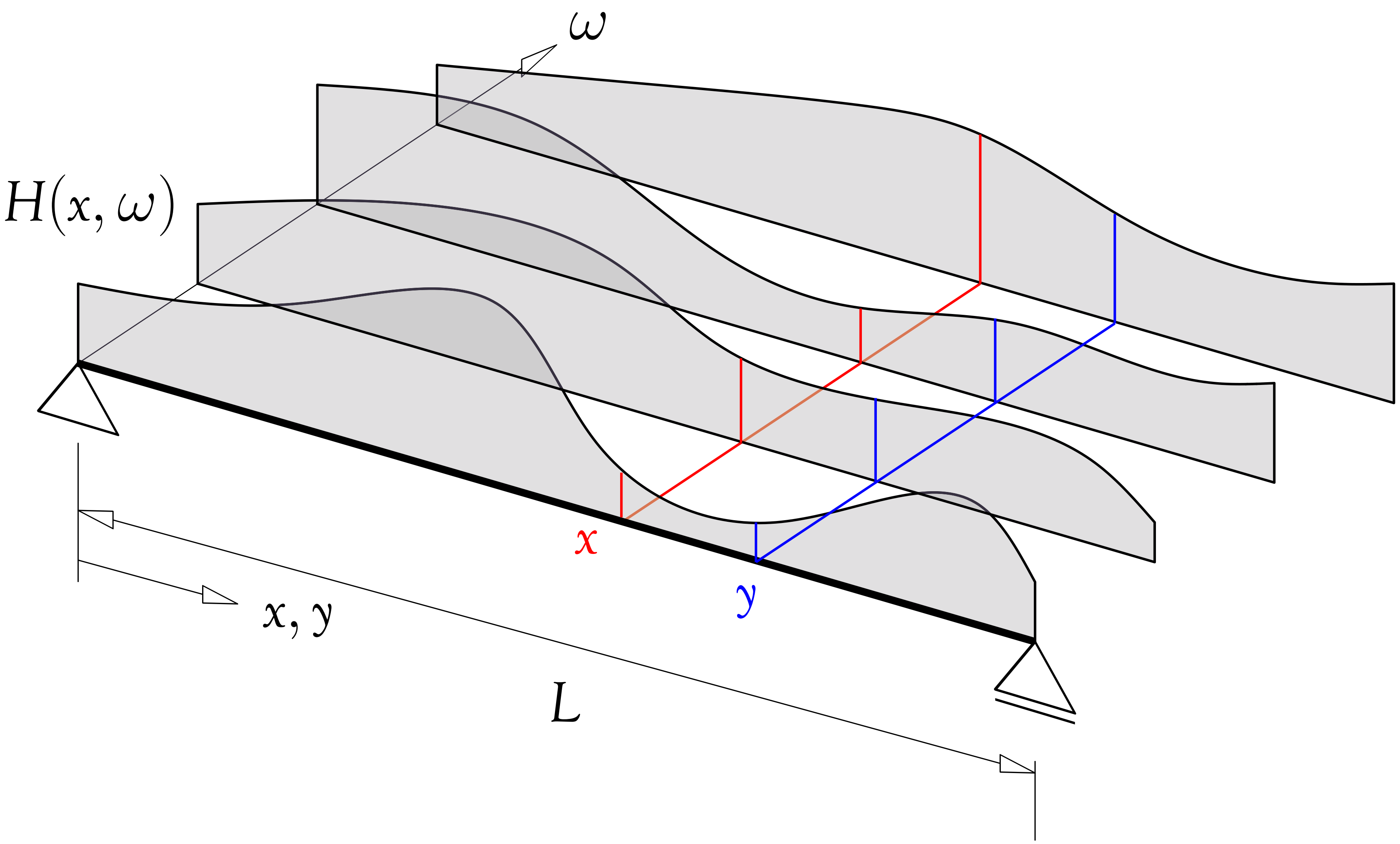
$$\bar{H}(\mathbf{x}) = \mathbf{E}[H(\mathbf{x})]$$

- Auto-covariance function

$$C_{HH}(\mathbf{x}, \mathbf{y}) = \mathbf{E}[\{H(\mathbf{x}) - \bar{H}(\mathbf{x})\}\{H(\mathbf{y}) - \bar{H}(\mathbf{y})\}]$$

Ensemble

- Different realizations of one-dimensional field



Spectral decomposition

- Fourier-type series expansion using deterministic basis functions φk and random coefficients ck

$$H(\mathbf{x}) = \sum_{k=1}^{\infty} c_k \phi_k(\mathbf{x}), \quad \mathbf{x} \in \mathcal{D} \subset \mathbb{R}^n; c_k, \phi_k \in \mathbb{R}$$

- Karhunen-Loeve expansion based on eigenvalue decomposition of the auto-covariance function

$$\mathrm{C}_{HH}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{\infty} \lambda_k \phi_k(\mathbf{x}) \phi_k(\mathbf{y}) \qquad \int_{\mathcal{D}} \mathrm{C}_{HH}(\mathbf{x}, \mathbf{y}) \phi_k(\mathbf{x}) \mathrm{d}\mathbf{x} = \lambda_k \phi_k(\mathbf{y})$$

- Leads to orthogonal basis functions and uncorrelated coefficients

Spatially discrete formulation

- Discrete values of random field

$$H_i = H(\mathbf{x_i}); \quad i = 1 \dots N$$

- Spectral representation

$$H_i = \sum_{k=1}^{N} \phi_k(\mathbf{x_i}) c_k = \sum_{k=1}^{N} \phi_{ik} c_k$$

- Written as matrix-vector multiplication

$$\mathbf{H} = \mathbf{\Phi c}$$

Choice of basis functions

- Reduce number of random variables significantly
  - Improves statistical significance for small sample size
  - Reduces numerical effort in statistical analysis
  - Simplifies representation of input/output relations based on meta-models
- Basis functions should be orthogonal
  - Reduces computational effort for projection/reduction
- Random coefficients should be uncorrelated
  - Simplifies digital simulation of random fields

Main purpose of SoS 3

- Want to reduce the number of variables involved in the description of the fields
- Need to identify input-output relations based on meta-models
- Want to simulate realizations of random fields

SoS 3 features

- Estimation of second-order statistics (mean value function, covariance function) based on sampled data, either from measurements or from computations such as Monte Carlo studies
- Reduction of random field models based on second order statistics by applying Karhunen-Loève expansion
- Digital simulation of random field samples

SoS 3 features

- Import/export of random field samples from/to third party finite element codes as element/node properties (primarily LS-Dyna)
- Mapping of random field data between incompatible meshes (typically fine-coarse, but also dislocated meshes)
- Identification of geometrical deviations
- Correlation analysis between input/output variables (→ optiSLang)
- Identification of important input variables (→ optiSLang)
- Treat exceptional situations (e.g. eroded elements)

Pre-processing

- Treatment for non-Gaussianity
  - nonlinear mapping of data (Box-Cox etc)
  - use simple indicators (skewness, kurtosis)
- Treatment for non-homogeneity
  - map to zero mean/unit standard deviation
  - guarantees homogeneity in the mean square sense (weak homogeneity)

Mesh coarsening

- Introduced to reduce storage requirements and computational efforts
- SoS2 is effectively limited to ~16000 elements, 12 hrs computing time
- Depending on problem, 30% of variability may be lost
- Local peaks of variability can be maintained only if they are in finely meshed areas (or where there is no substantial mesh coarsening)
- Iterative process, user interaction required

Fast analysis of random fields

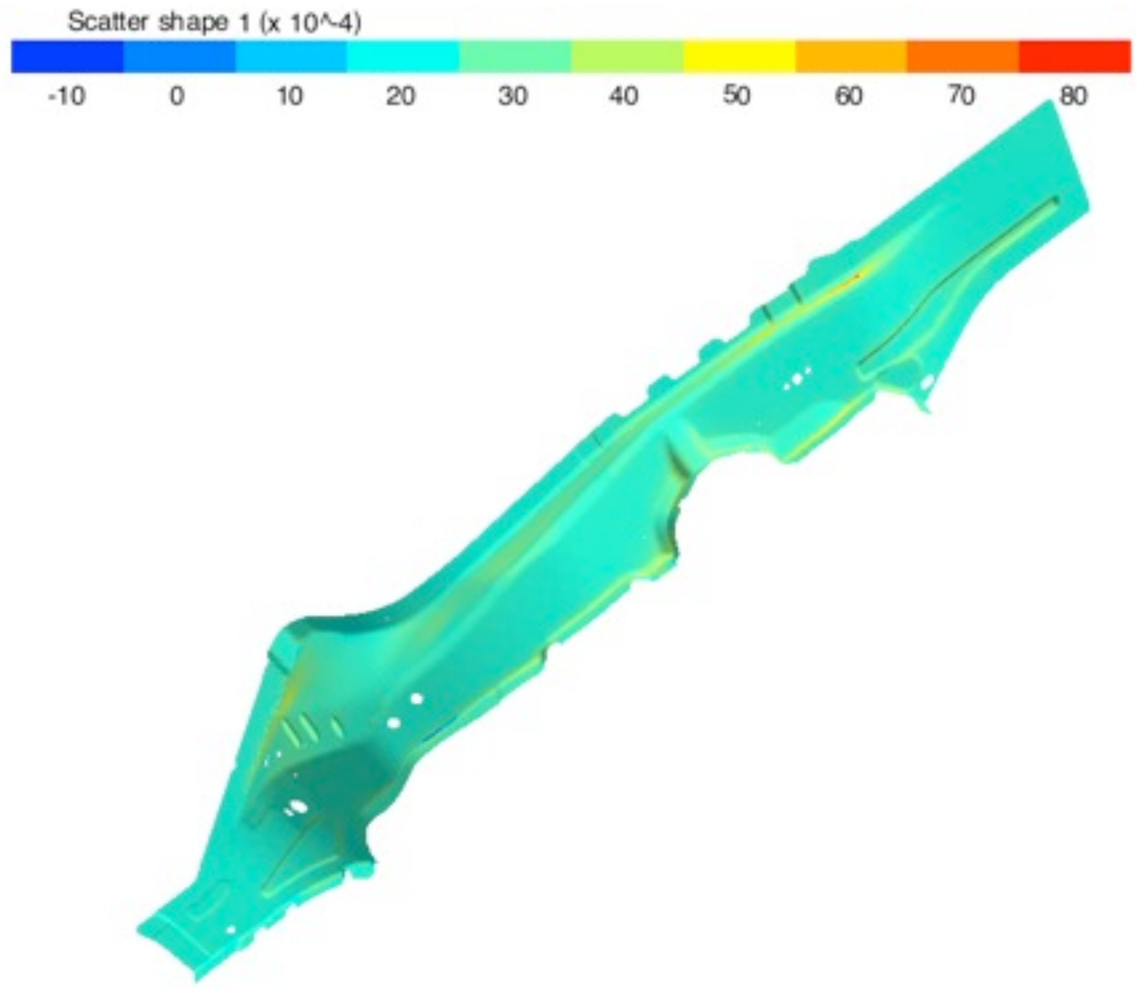
- Avoid large storage requirements (do not store full covariance matrix)
- Fast computation of eigenvectors/eigenvalues
- No smoothing required, hence no loss of accuracy
- Full PCA possible (number of eigenvectors equal to number of samples)
- Error estimation due to reduced PCA readily possible

Example 1

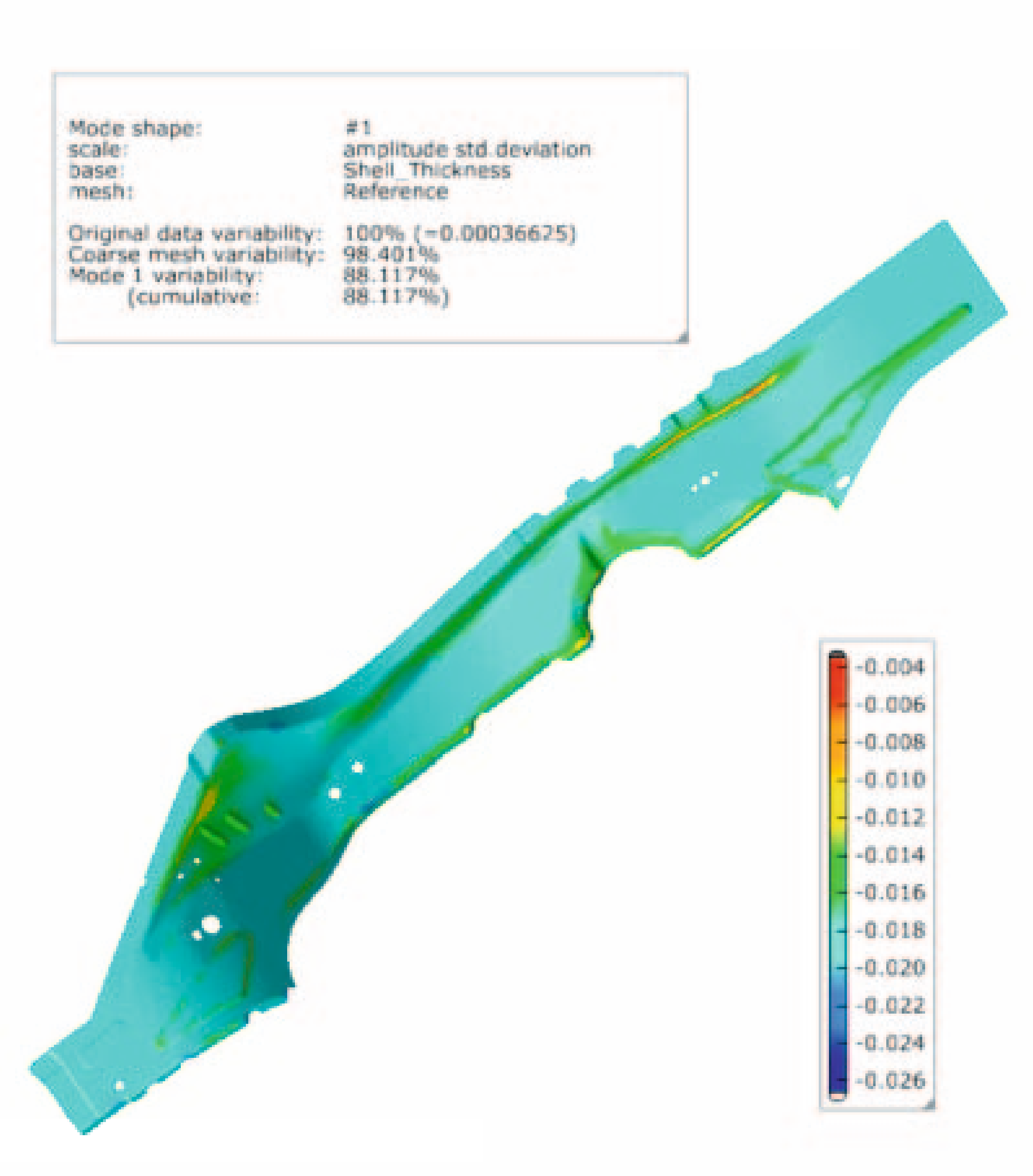
- Structure: 60.800 elements
- 150 samples
- Time for loading and creating structure: 1 s
- Time for loading one set of results: 50 s
- Time for creating projection matrix for one set of results: 12 s (i7 QuadCore, 2.93 GHz)
- Peak Memory: 0.80 GB

Example 1: Thickness, shape 1

- Shape #1, 86%



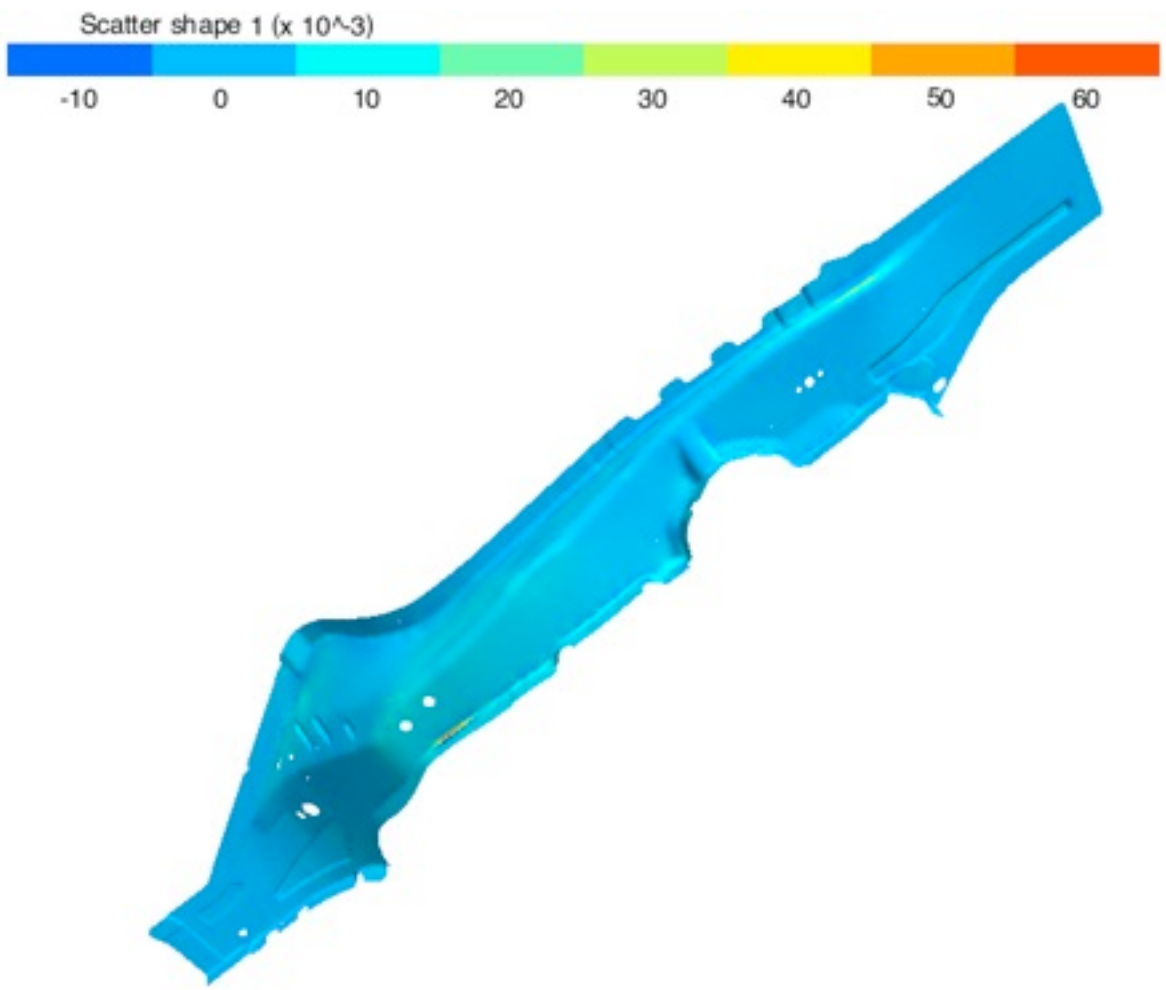
SoS 3



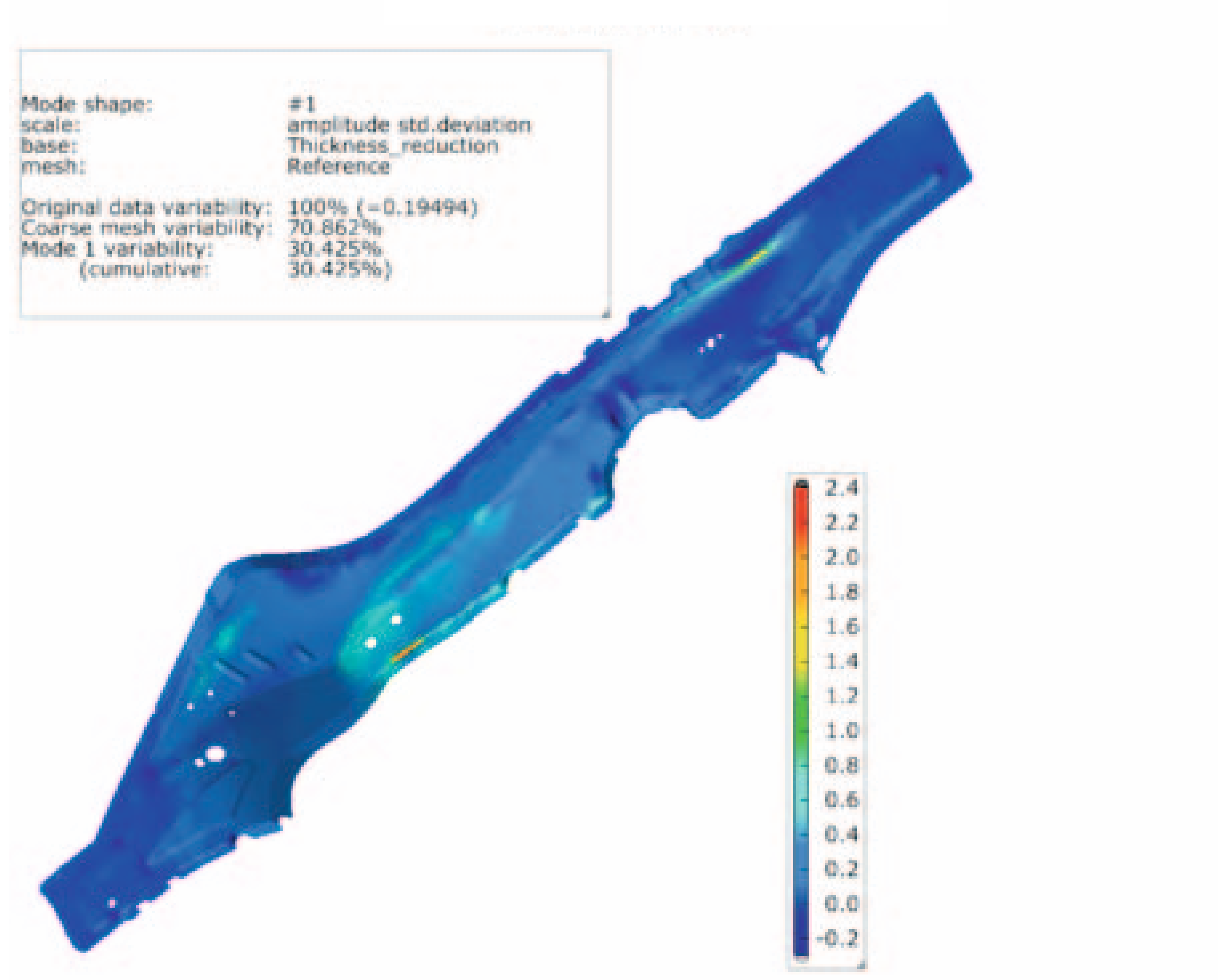
SoS 2

Example 1: Thickness reduction, shape 1

- Shape #1, 39%



SoS 3

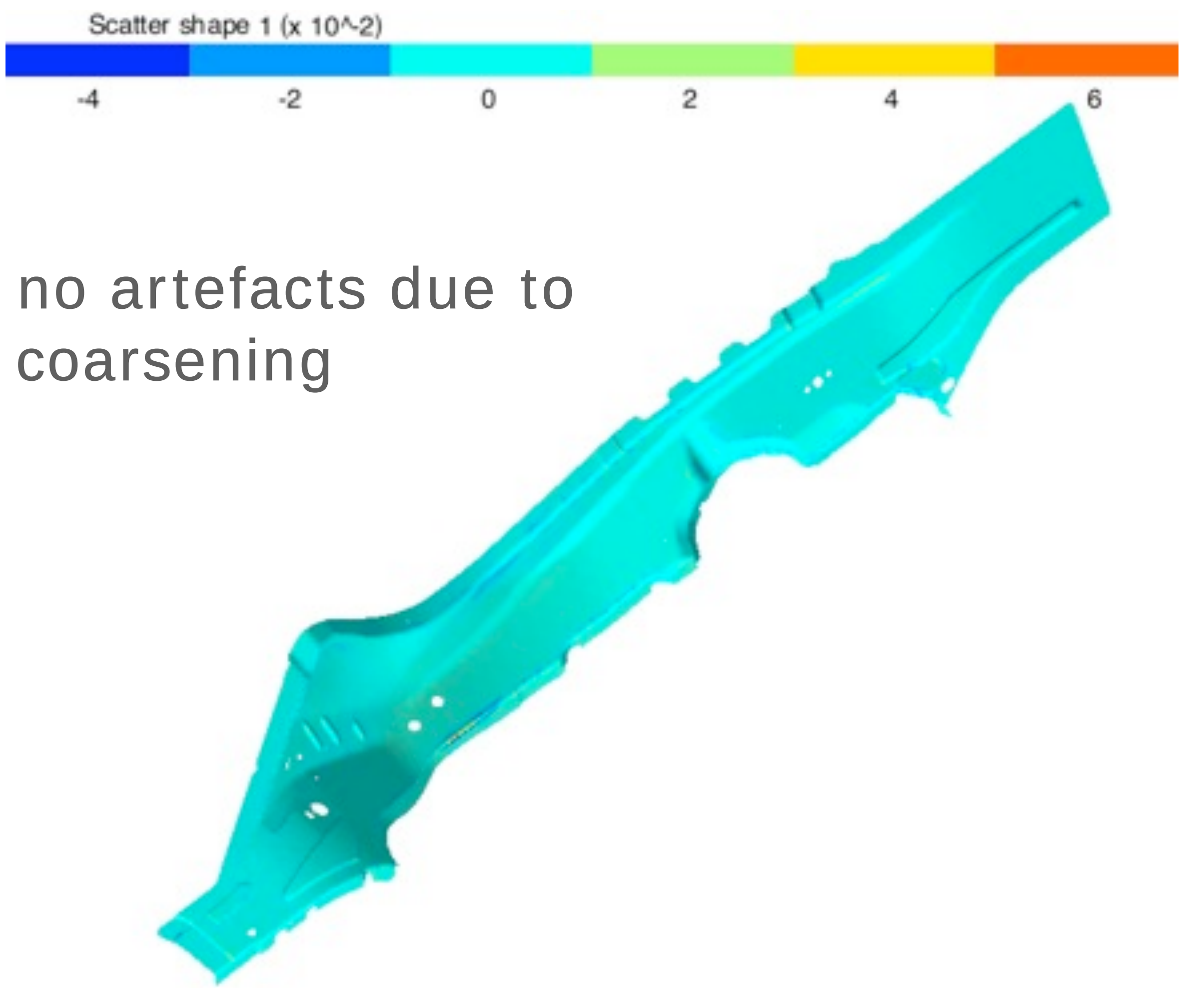


SoS 2

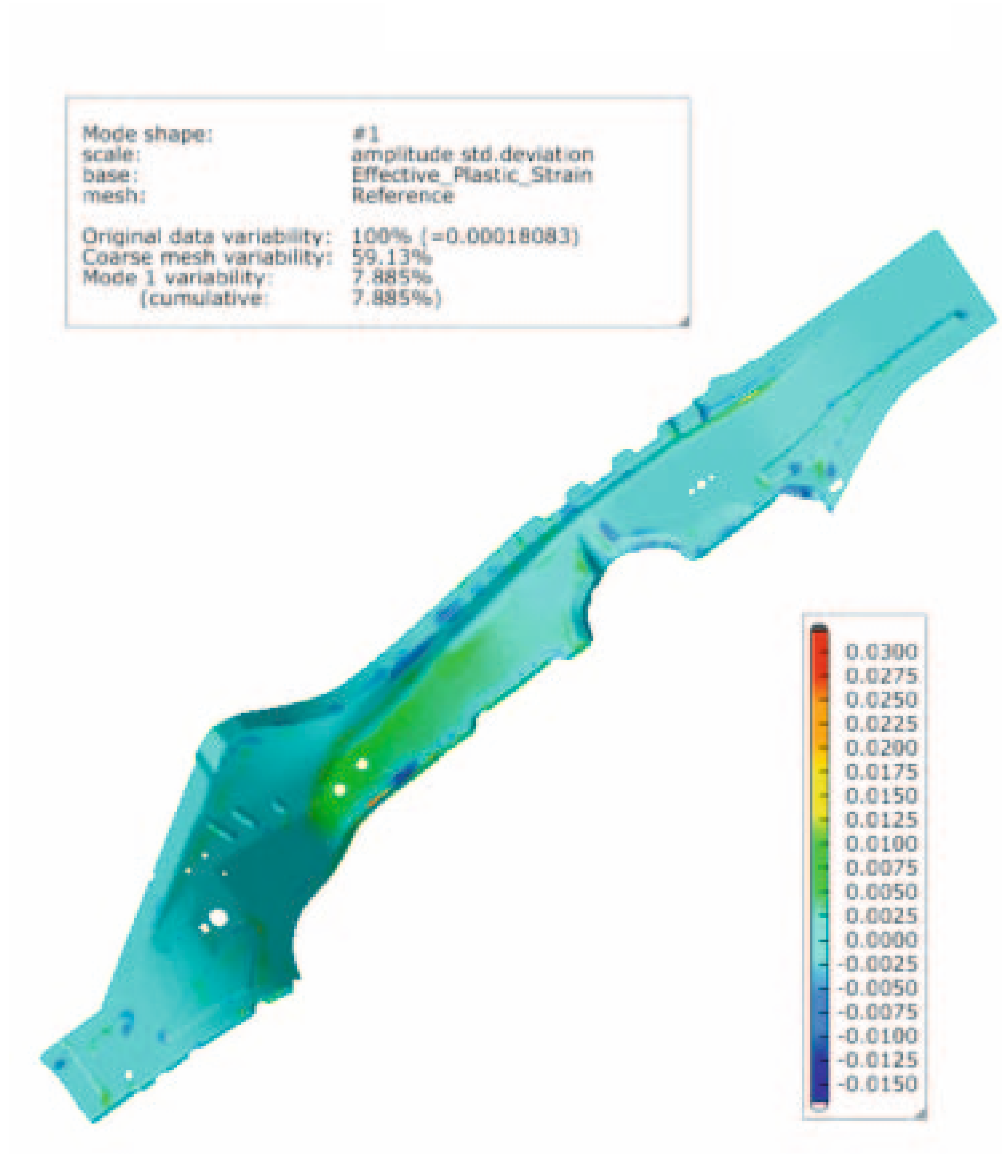
Example 1: Plastic strain, shape 1

- Shape #1, 13%

no artefacts due to coarsening

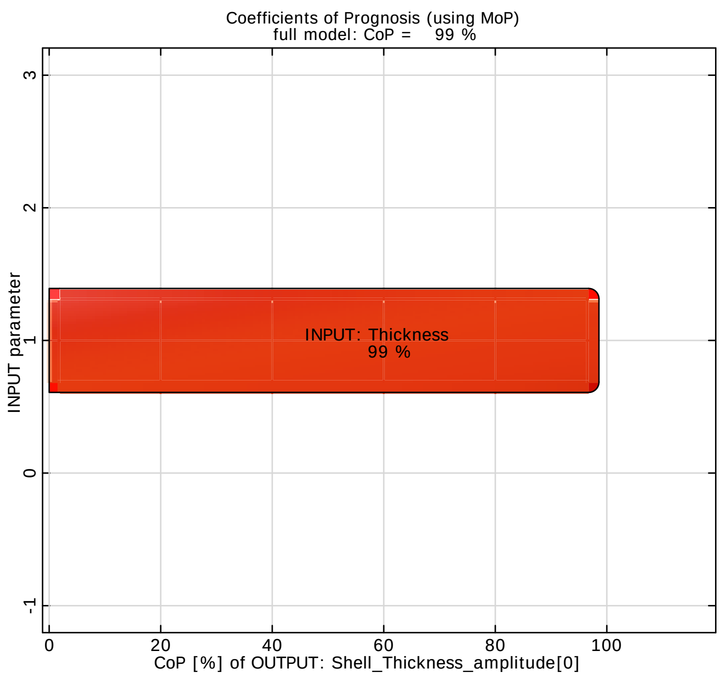


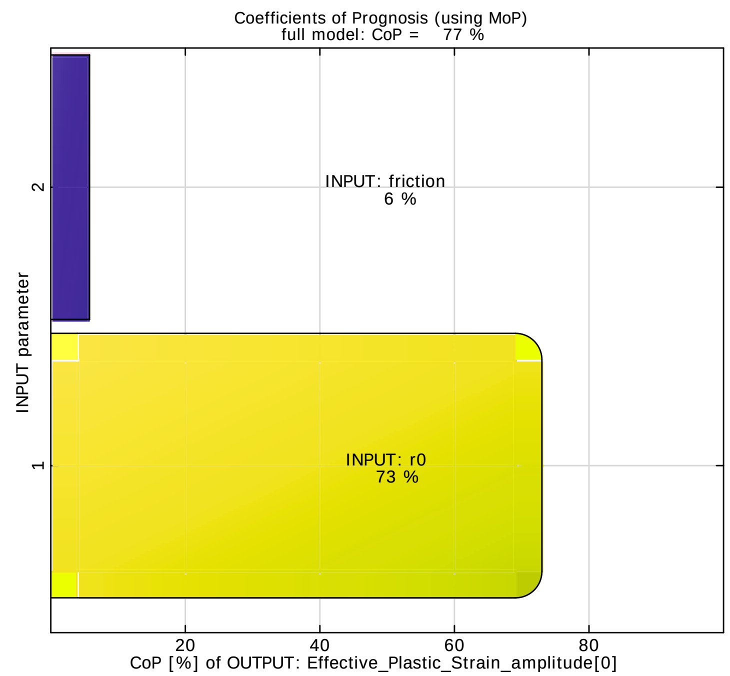
SoS 3

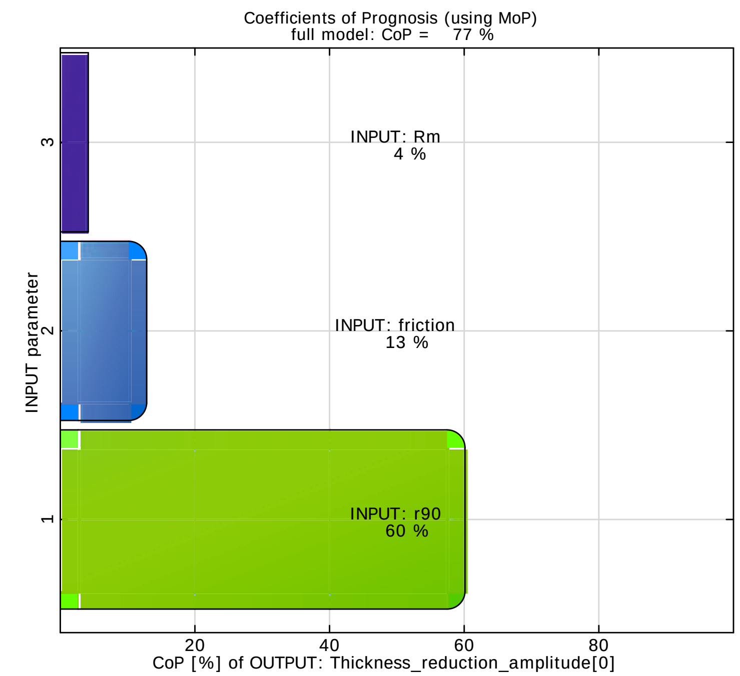


SoS 2

Example 1: MOP



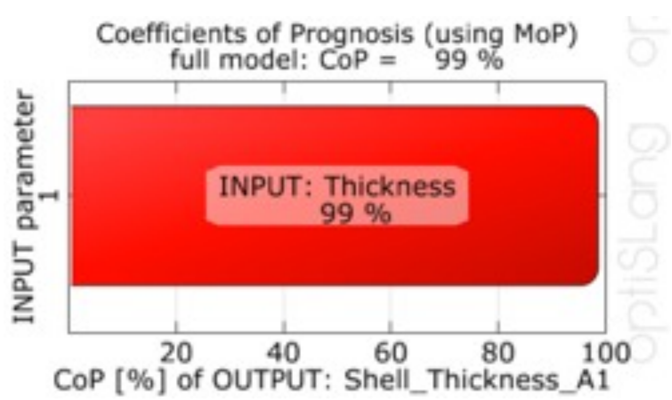


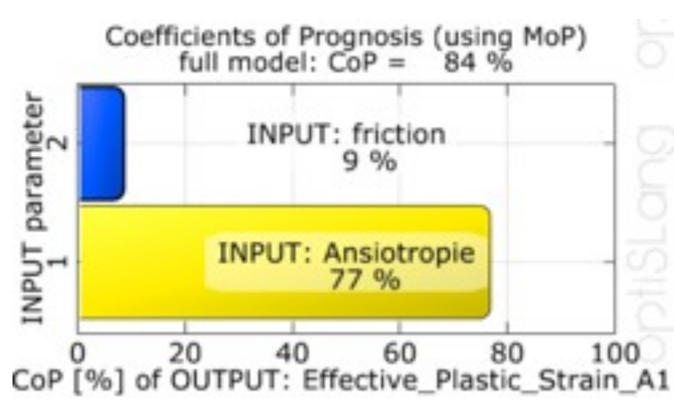


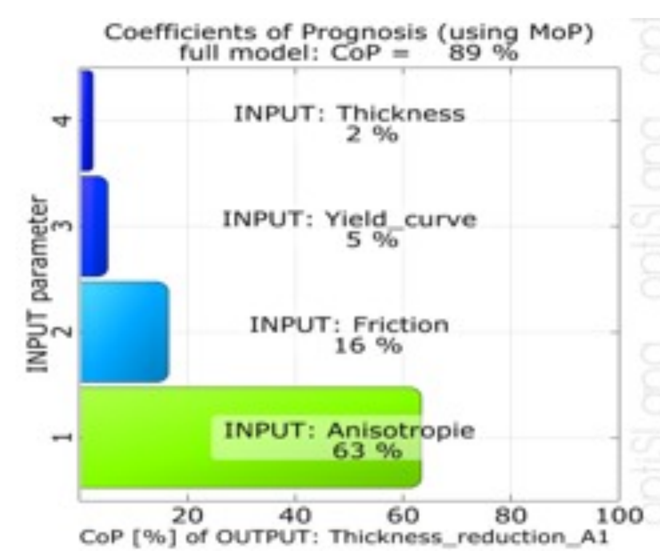
Thickness, shape 1

Plastic strain, shape 1

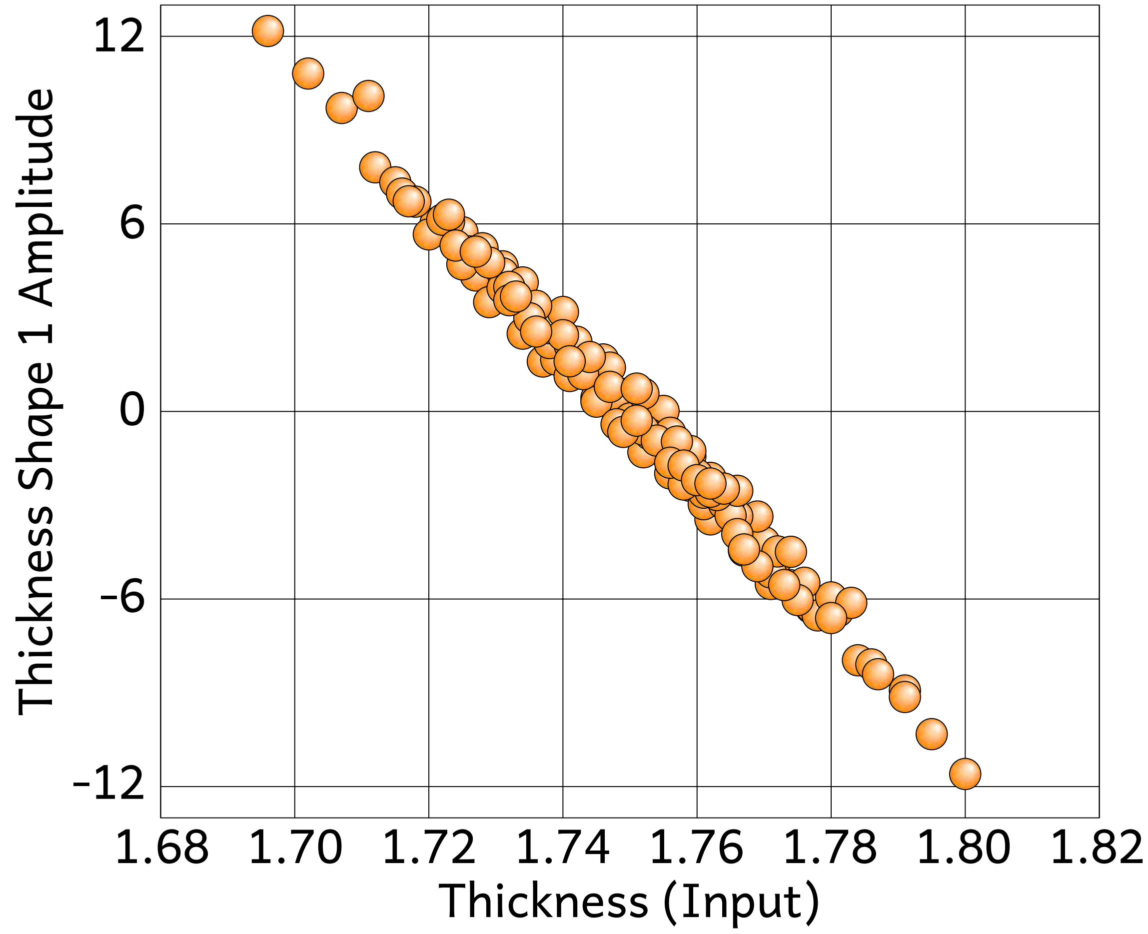
Thickness reduction, shape 1



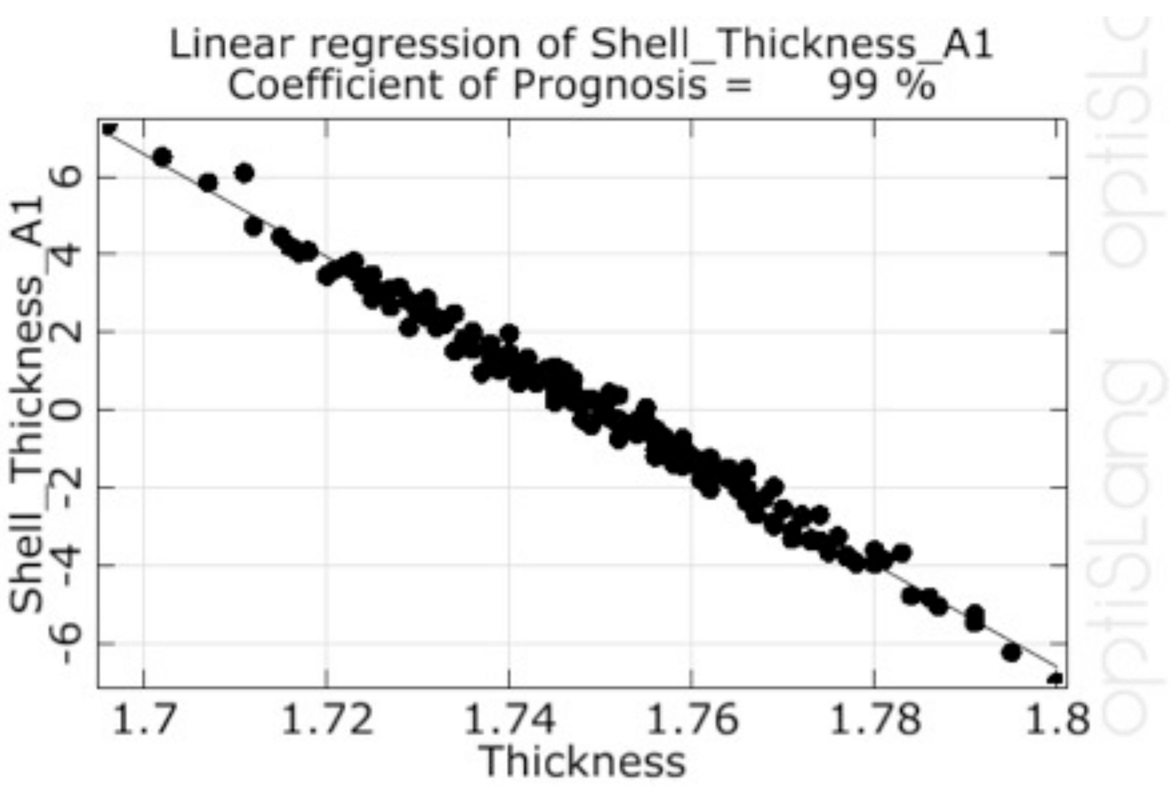




Example 1: Thickness, shape 1 vs thickness



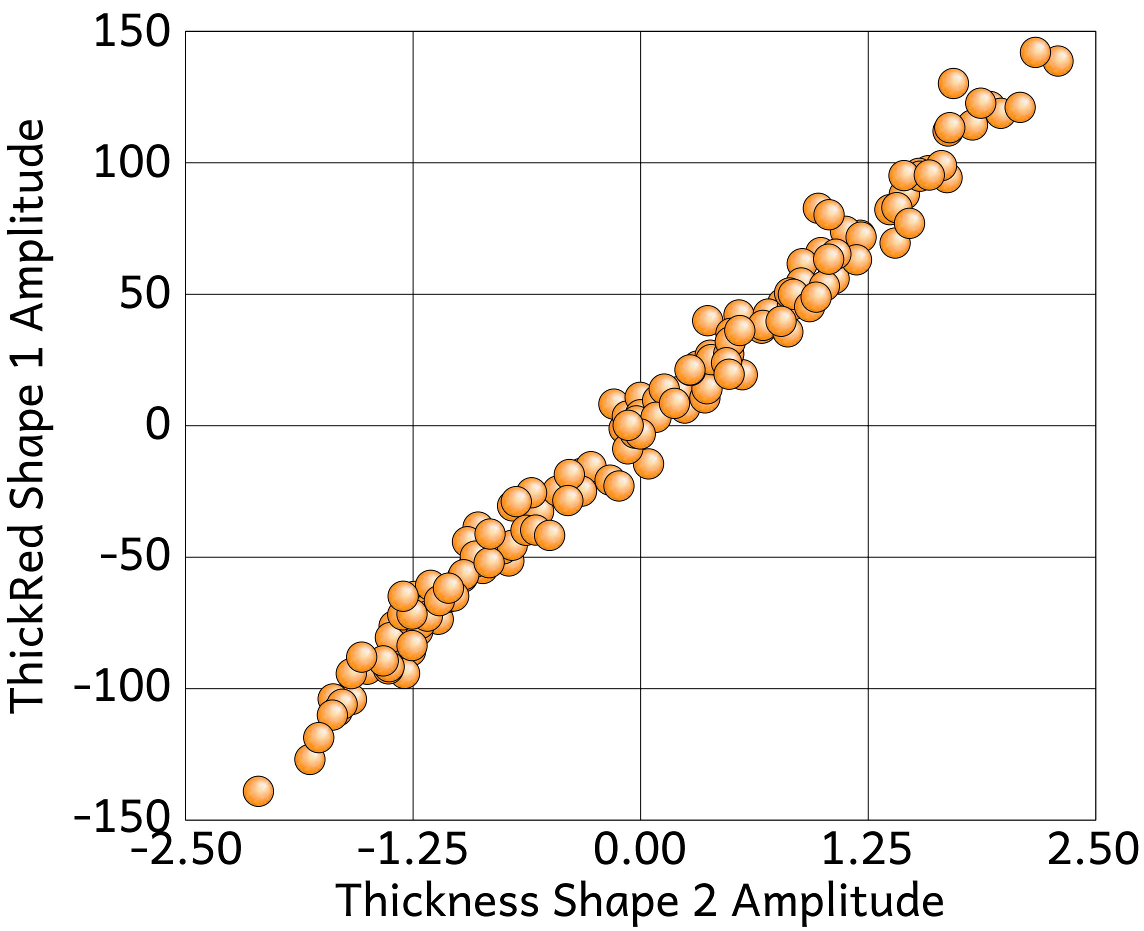
SoS 3



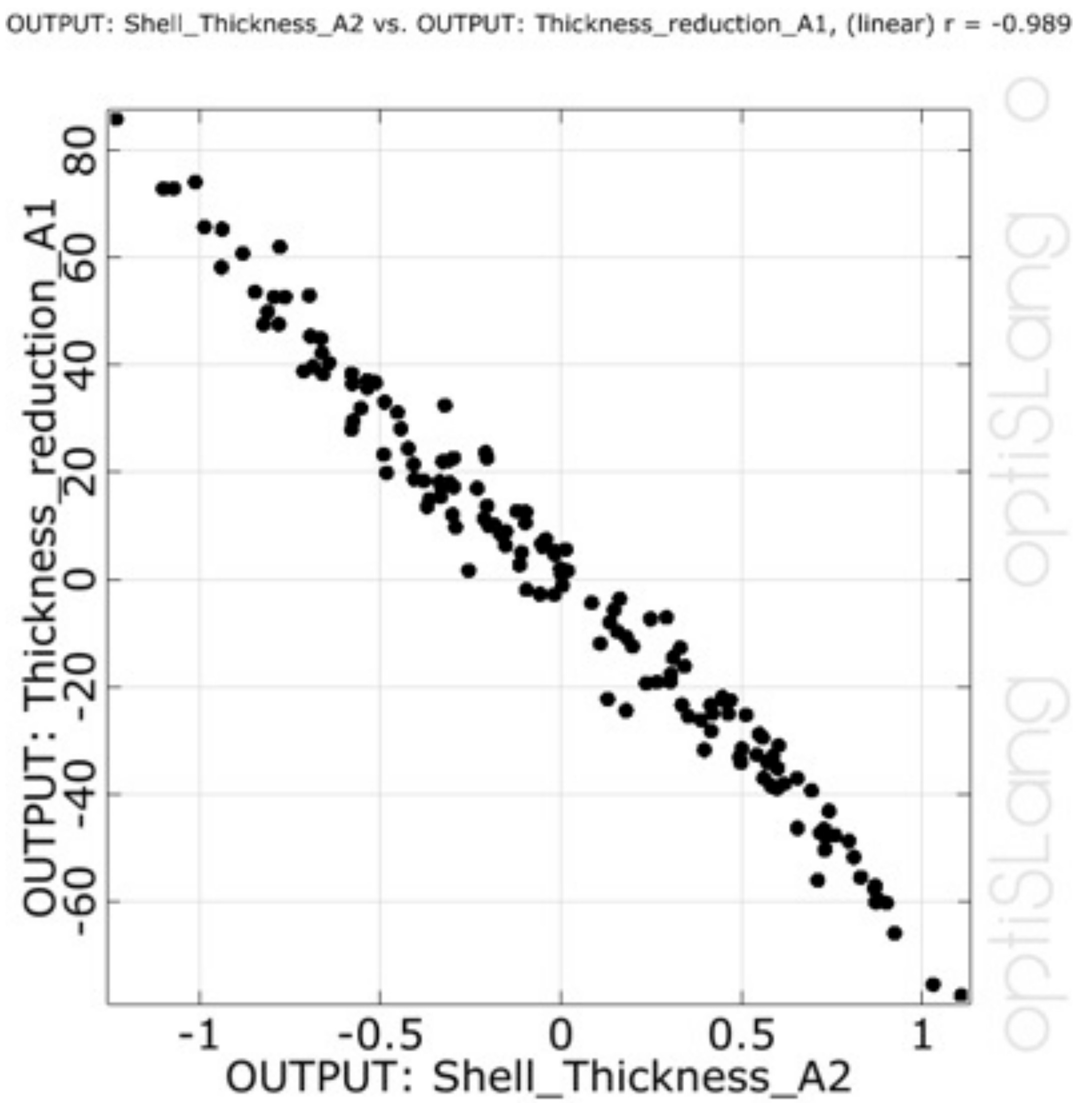
SoS 2

Example 1: Thickness, shape 2 vs. Thickness reduction, shape 1

Different sign of scatter shape (not relevant)



SoS 3



SoS 2

Example 2

- Structure: 440.000 elements
- 150 samples
- Time for loading and creating structure: 18 s
- Time for loading one set of results: 383 s
- Time for creating projection matrix for one set of results: 45 s (i7 QuadCore, 2.93 GHz)
- Peak Memory: 5.75 GB

Example 2: plastic strain

- First scatter shape: 64%, second shape 10%, third shape 2%
- For 90% representation: 21 scatter shapes

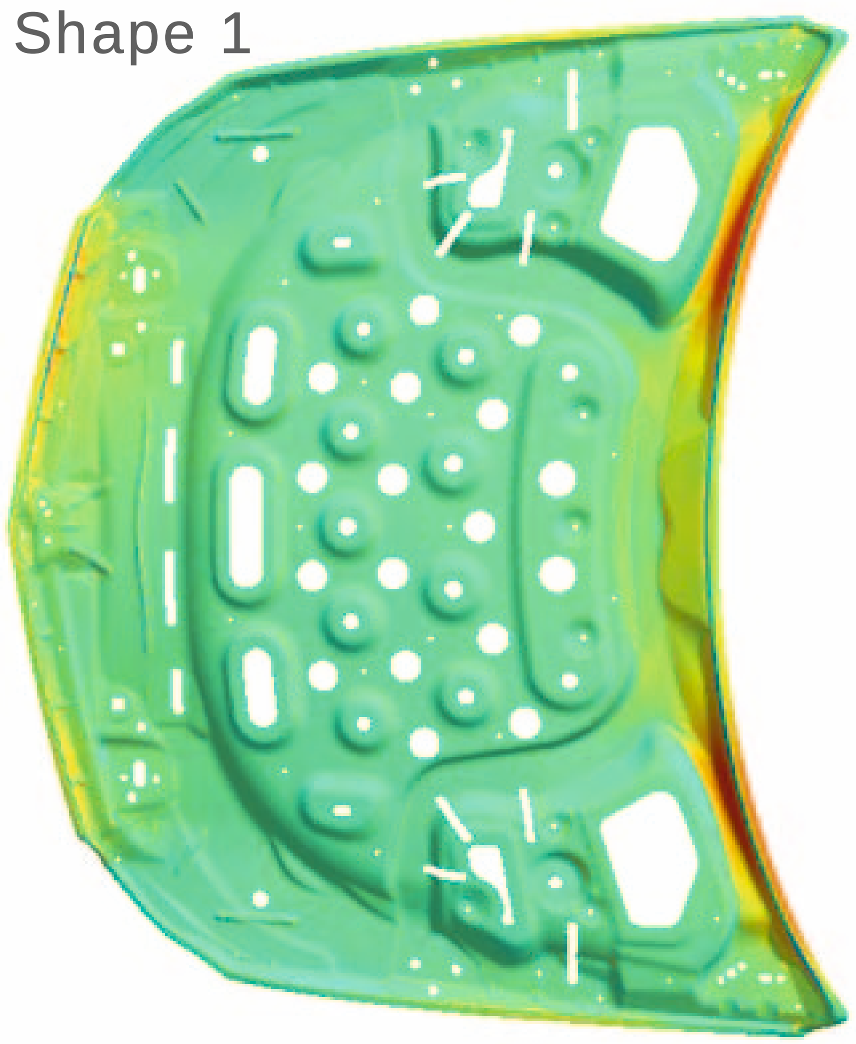
Shape 1

Shape 2

Example 2: thinning

- First scatter shape: 71%, second shape 11%, third shape 2%
- For 90% representation: 8 scatter shapes

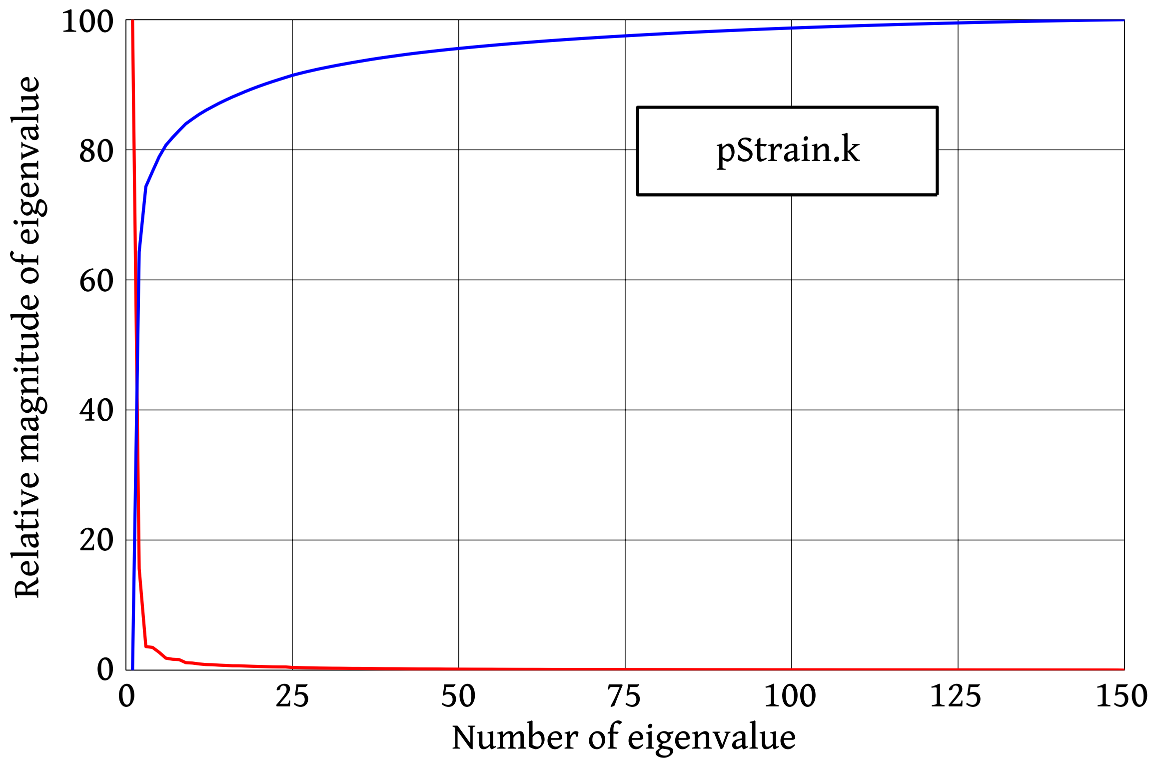
Shape 1

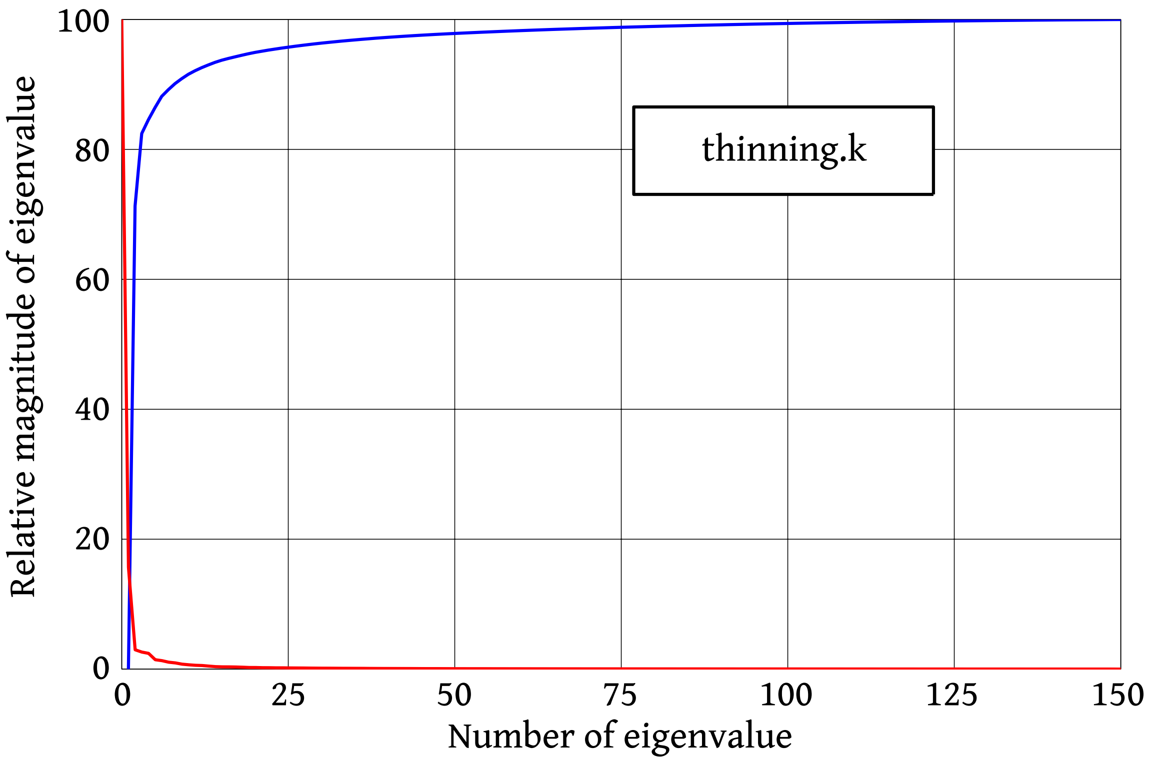


Shape 2

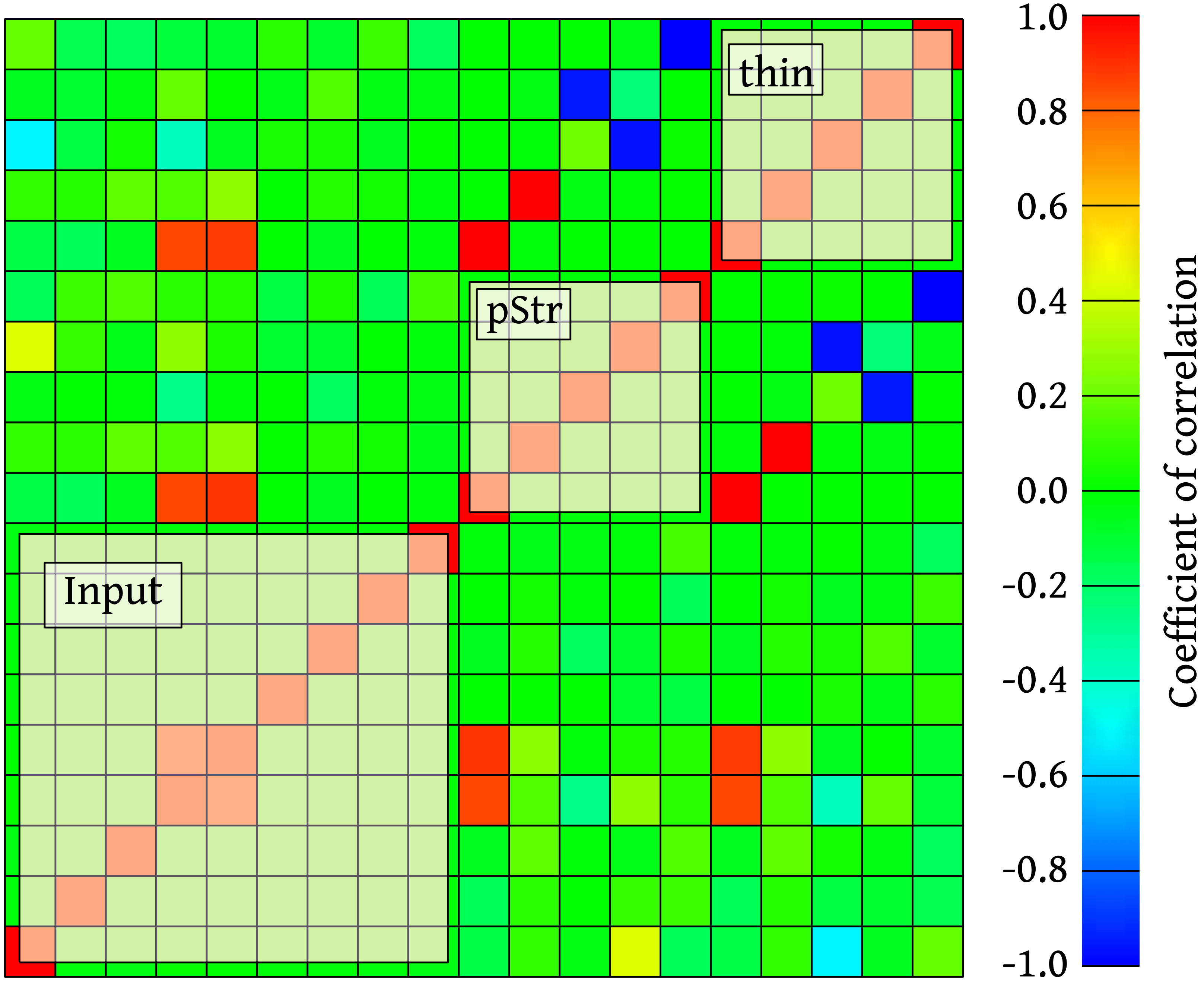
Example 2: KL decomposition

- Relative and cumulative size of eigenvalues



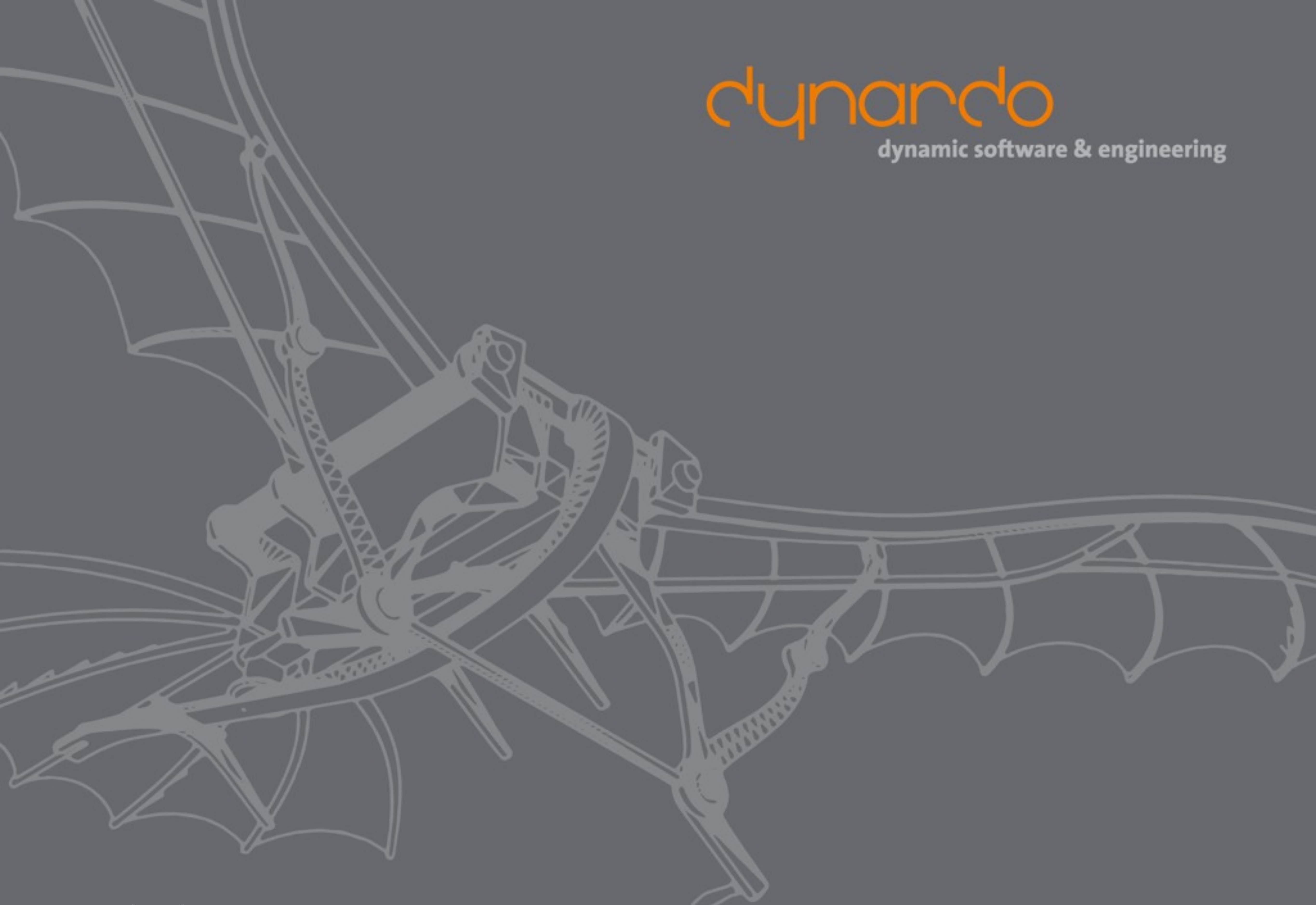


Example 2: Correlations



Concluding remarks

- SoS 3 now capable of handling large FE models
- Identification of important scatter shapes readily possible
- Scatter shapes belonging to different response quantities may have significant correlations
- Using CoP/MOP the relevant input variables can be identified
- Very good agreement with results obtained previously by SoS 2
- Substantial improvement of numerical performance



dynardo

dynamic software & engineering

contact us:
kontakt@dynardo.at