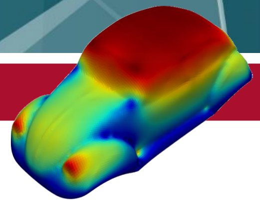
AI-Based Parameterization of Full Vehicle Models Considering Manufacturing Effects for Crash Simulations

Dr. Ingolf Lepenies[1*], Dr. Peter Krause[2],
Dr. Sebastian Kriechenbauer[1], Thorsten Pohl[3], Rolf Schwarzer[4]

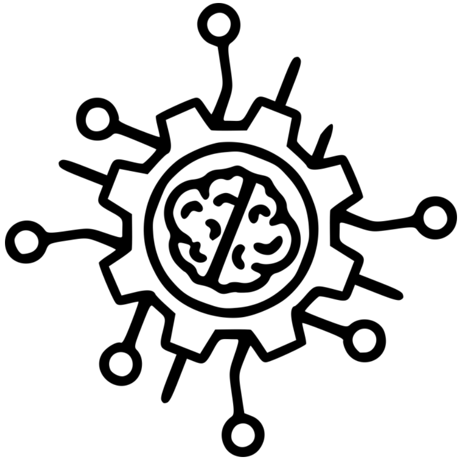
1 SCALE GmbH,
2 divis intelligent solutions GmbH
3 Stellantis OPEL AUTOMOBILE GmbH, FAT – Project UmMatCrashML
4 KIRCHHOFF Automotive Deutschland GmbH, FAT

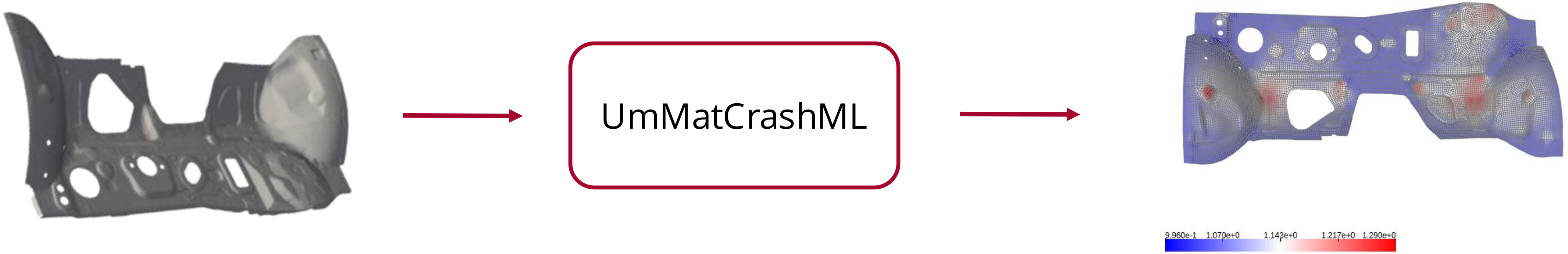


Ansys EMEA Transportation Summit and LS-DYNA User Conference,
Munich, 28.-29.10.2025

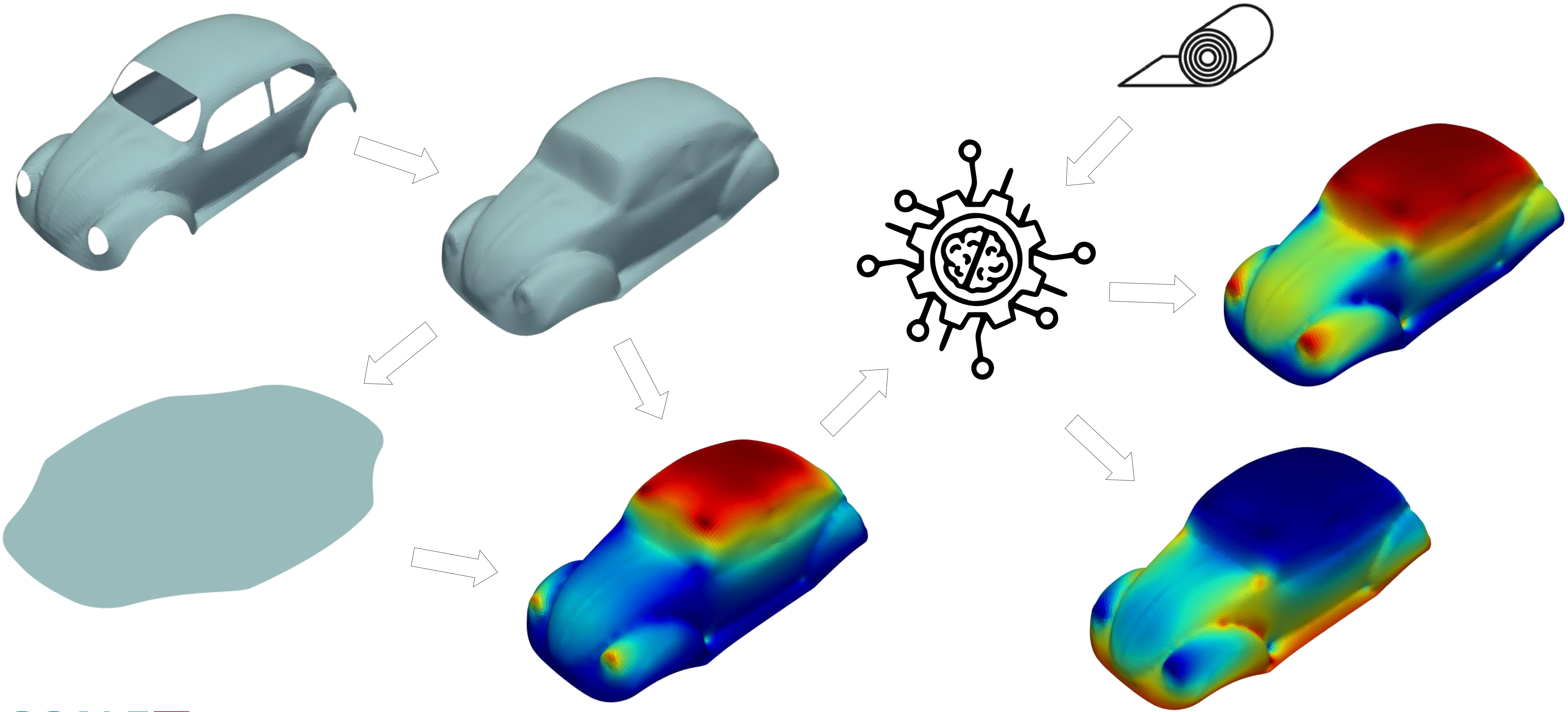
divis SCALE
IT-Solutions for CAE

Contents





Idea: remesh → unfold → area strain → AI(mat) → thickness & plastic strain

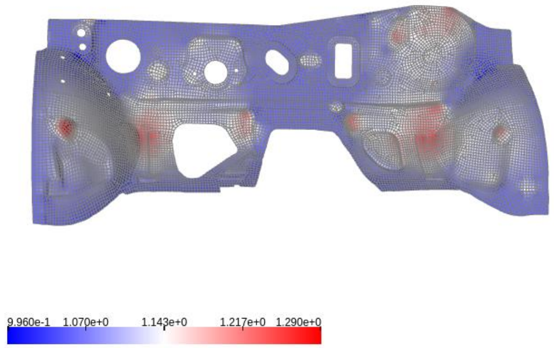


UmMatCrashML workflow

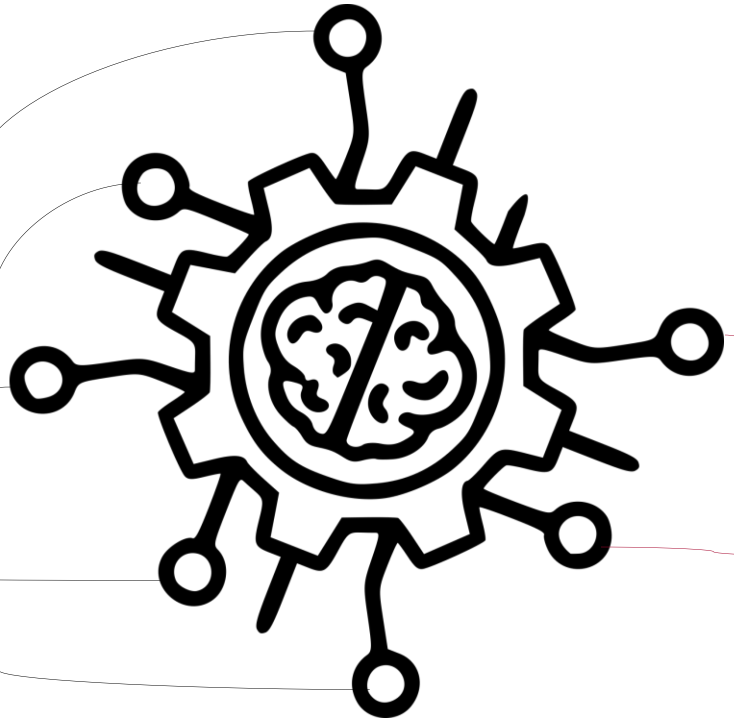
FEM model of a part

UmMatCrashML

Initial conditions of a part



Input vector
Geometry: Ar_uvmap
Material: sig_0
Material: sig_c
Material: eps_c
Material: sig_a

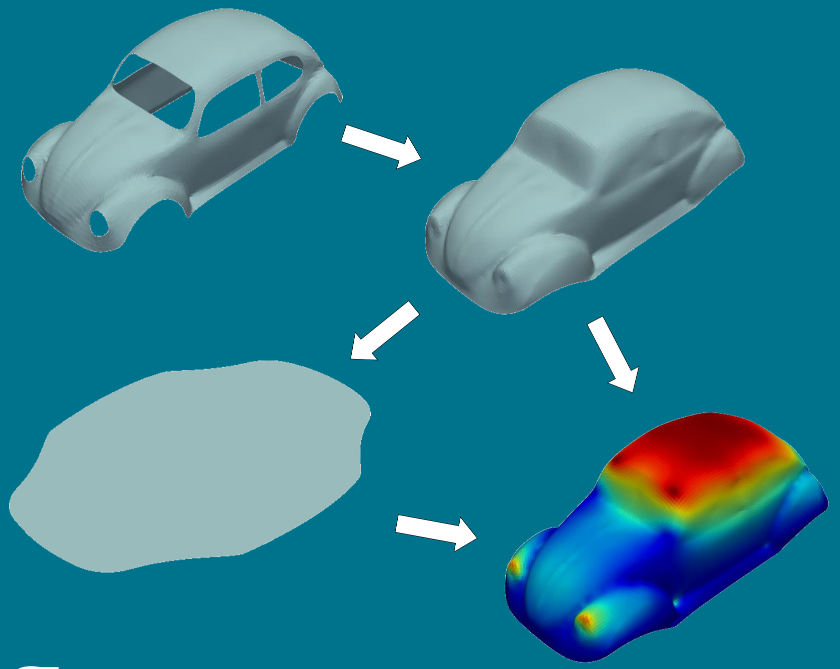


ClearVu AI model

Output vector
Ar-lsdyna (→ sheet thickness)
effective plastic strain

SCALE

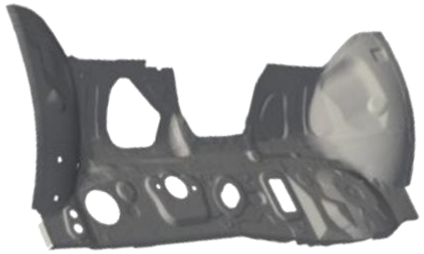
Pre-Processing



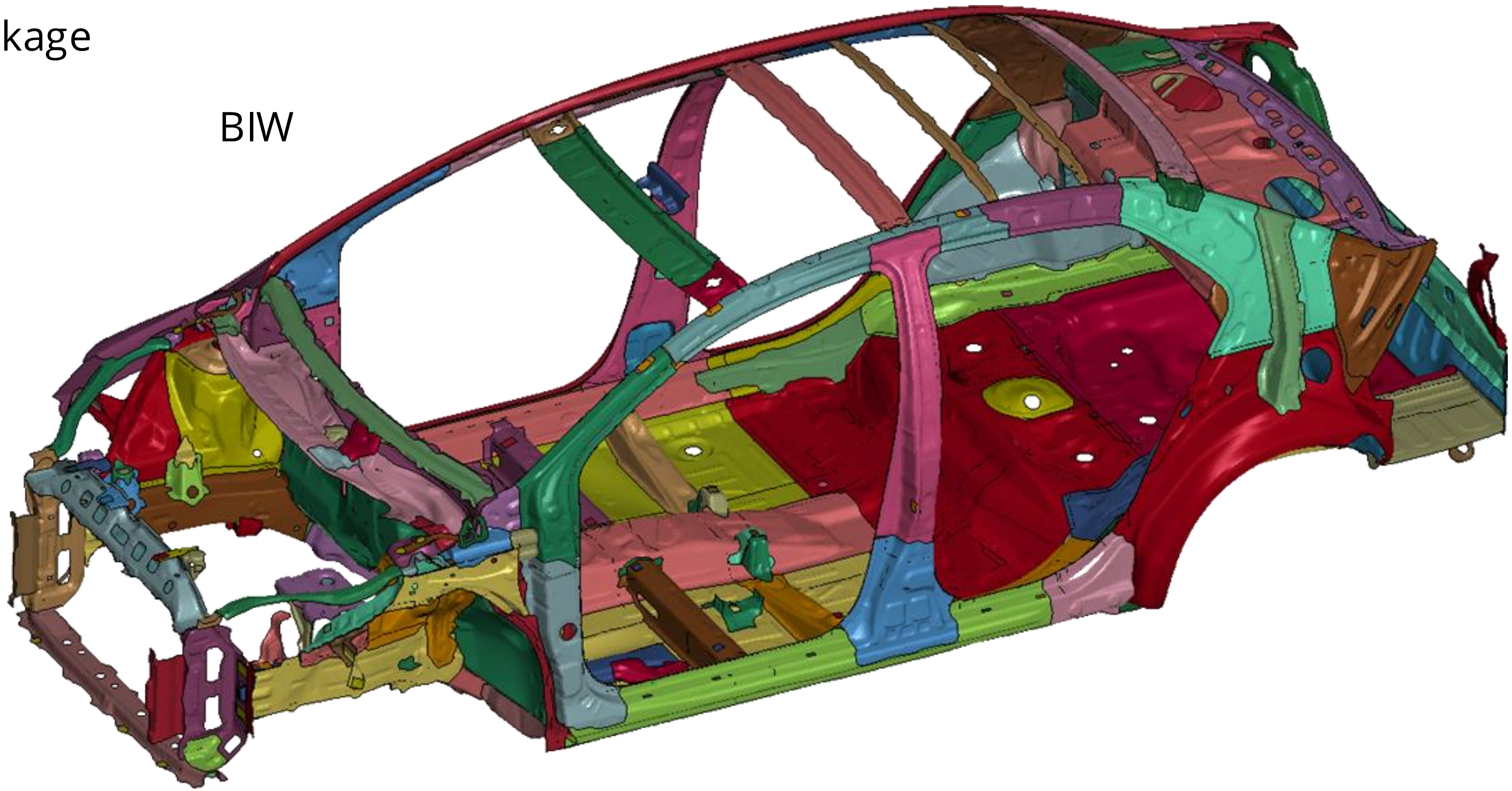
Geometrical one-step unfolding

Extraction of structural components from LS-DYNA FE models

- Script-based component extraction with python package scale.fem



BIW



- automatic data upload SCALE.sdm

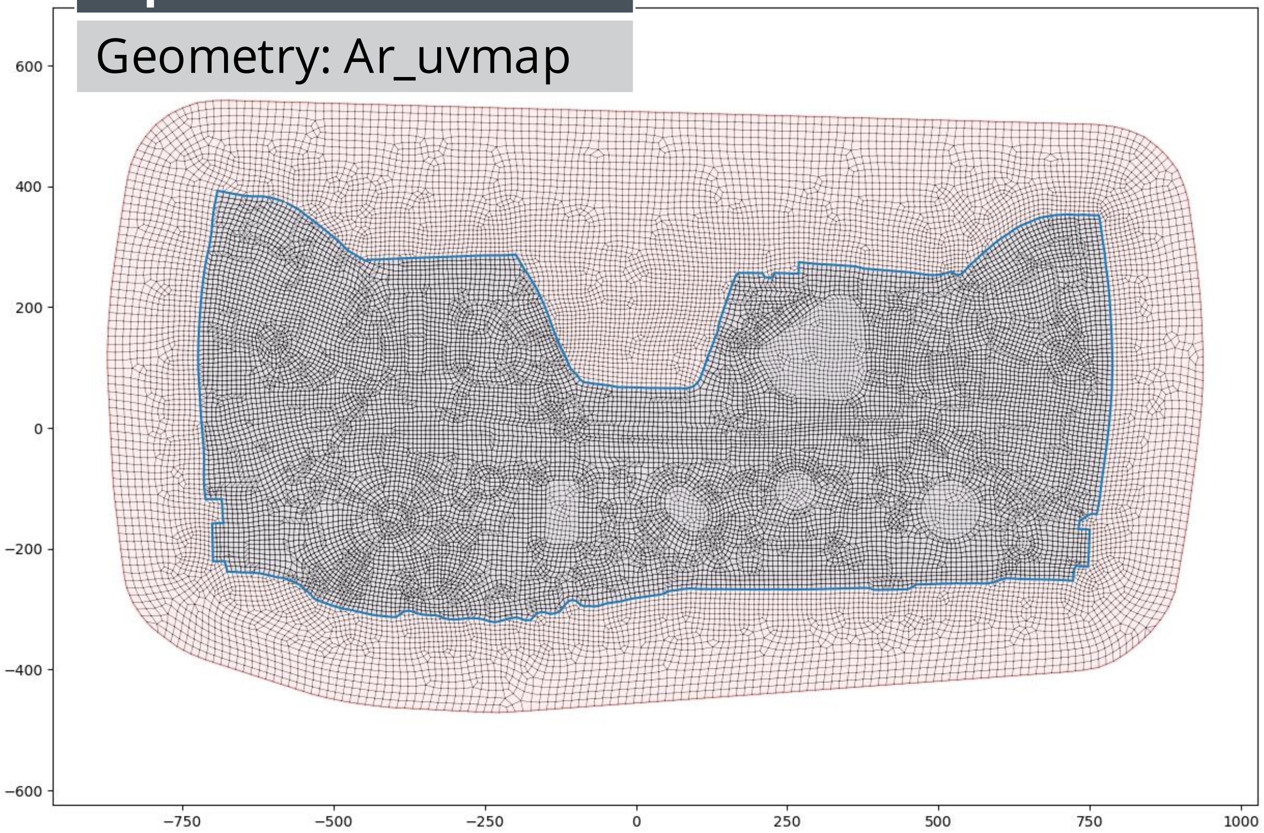
[https://www.ccsa.gmu.edu/models/]

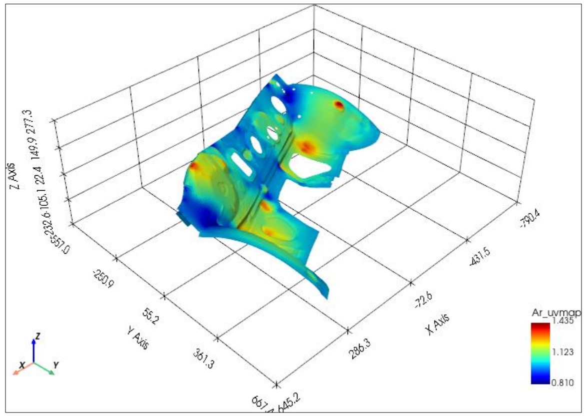
Geometric Onestepsolver

- Automatic preprocessing with scale.dynapy
- Optional estimation of the coil blank size and shape

Input vector

Geometry: Ar_uvmap



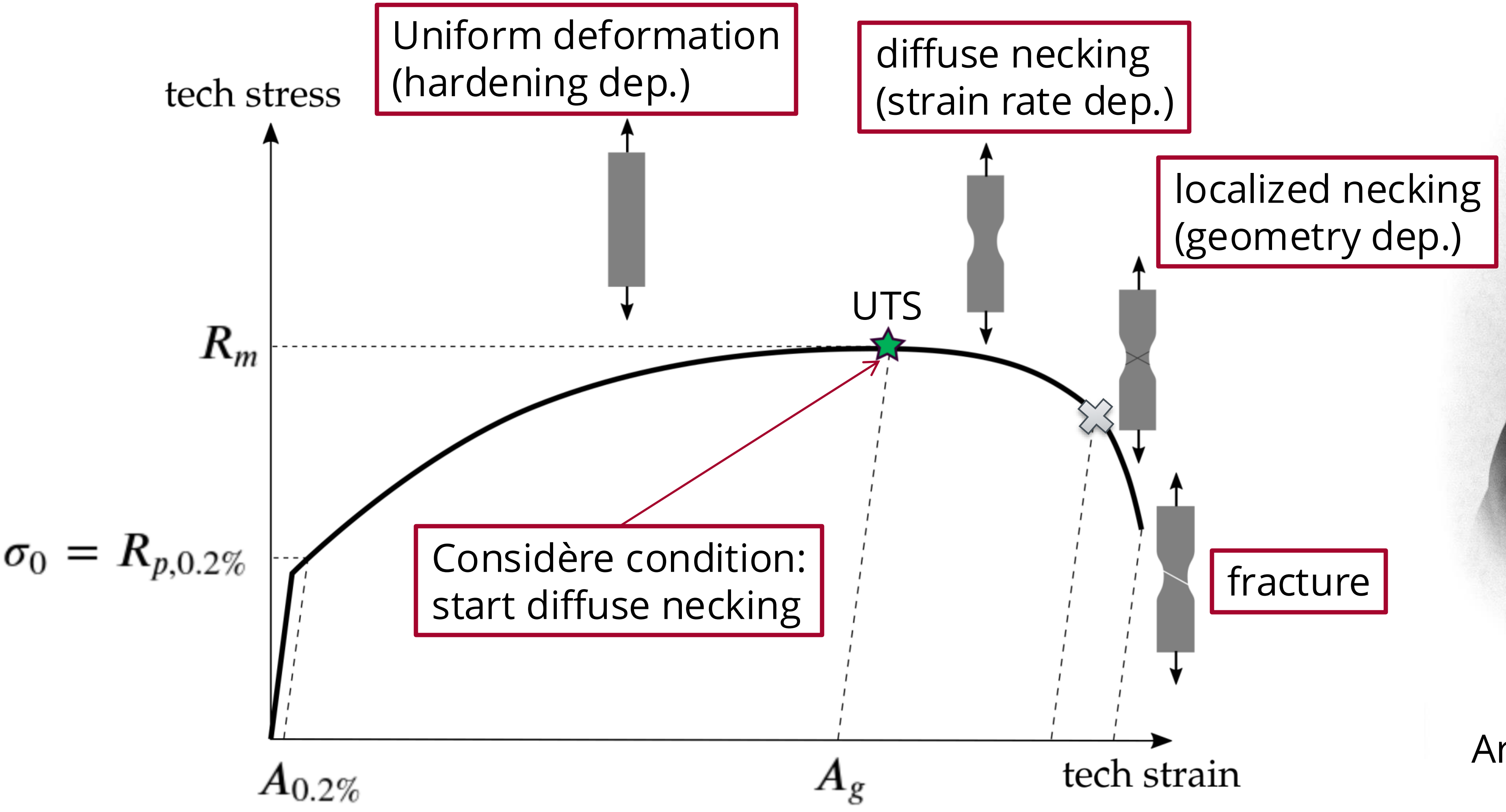


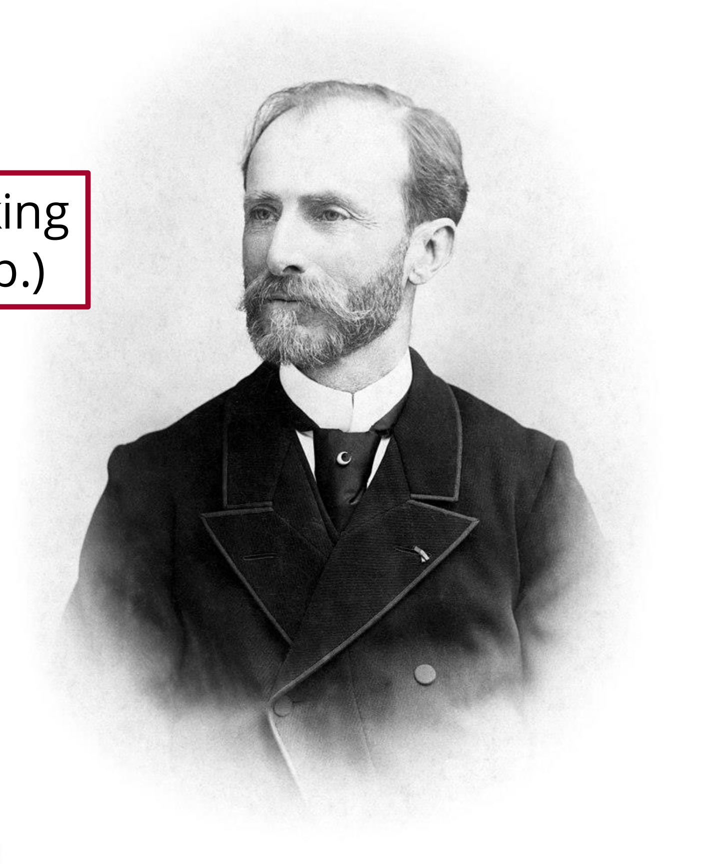
Material description

Generic, continuous material database

Considère condition for plastic instability

[Armand Gabriel Considère, The Use of Iron and Steel in Construction, 1885]



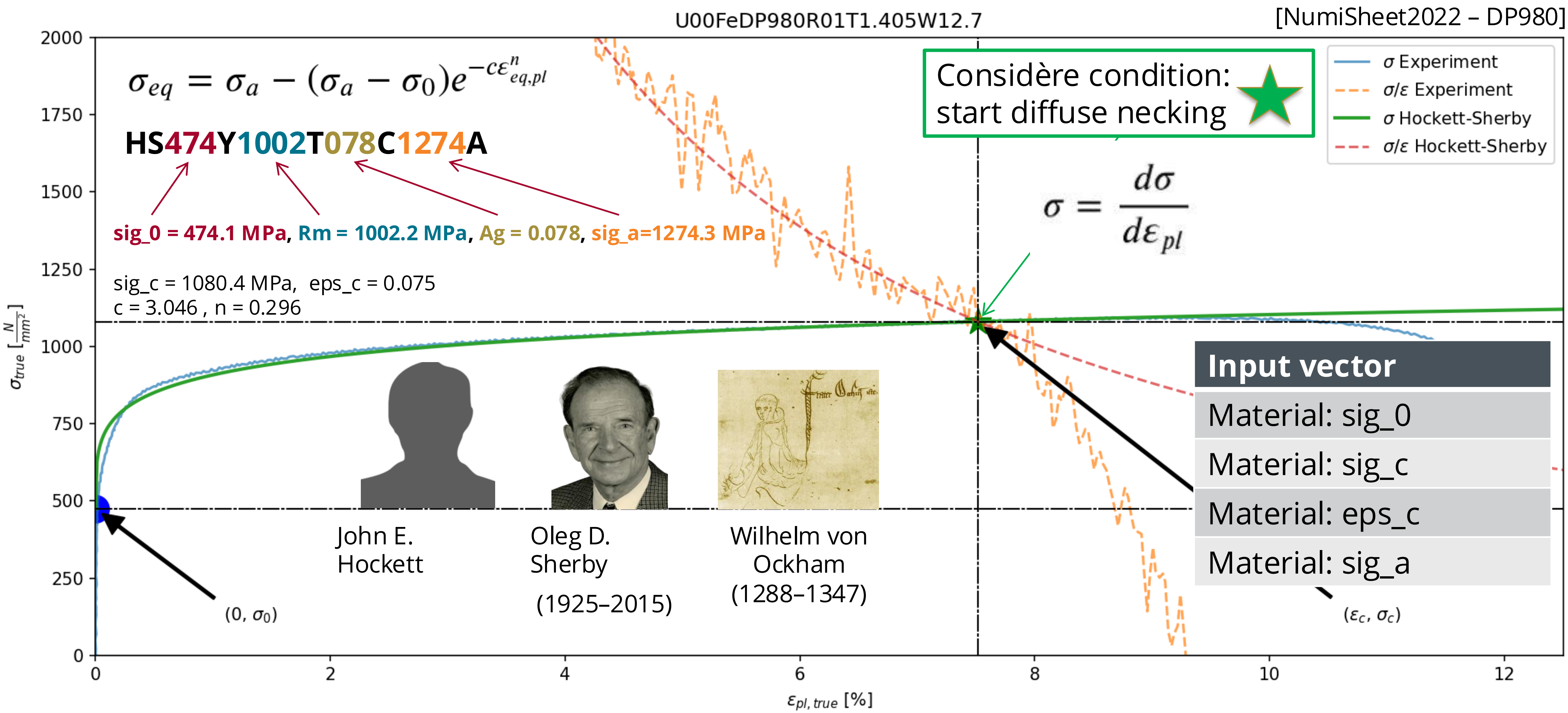


Armand Gabriel Considère
(1841-1914)

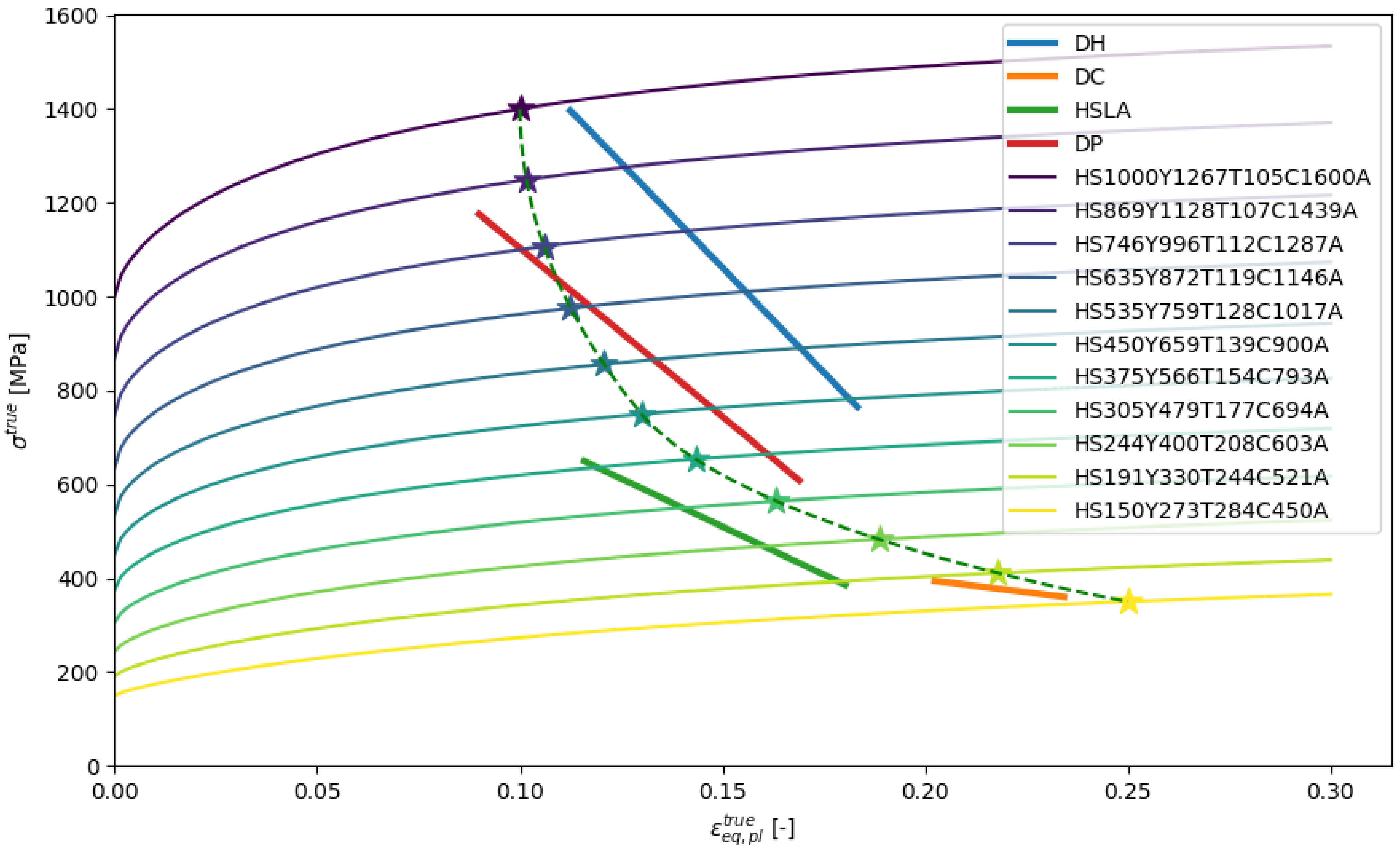
[fr.wikipedia.org/wiki/Armand_Consid%C3%A8re]

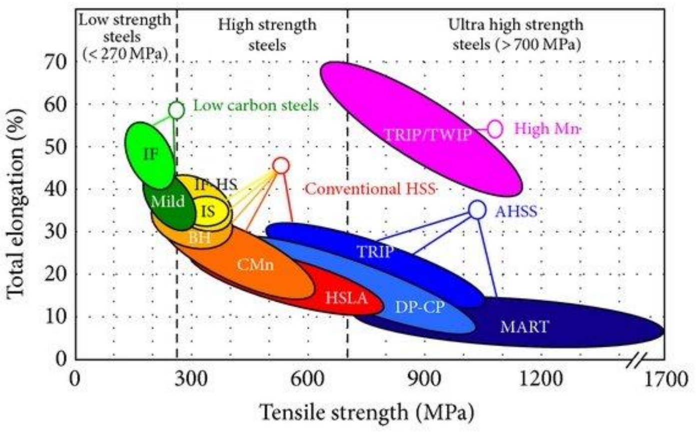
SCALE divis

Parametrization of the materials $(\sigma_0, \sigma_c, \varepsilon_c, \sigma_a)$



“Steel banana” with uniaxial material instability (diffuse necking ★)





Suwanpinij, Piyada. (2016). The Synchrotron Radiation for Steel Research. Advances in Materials Science and Engineering. 2016. 10.1155/2016/2479345.

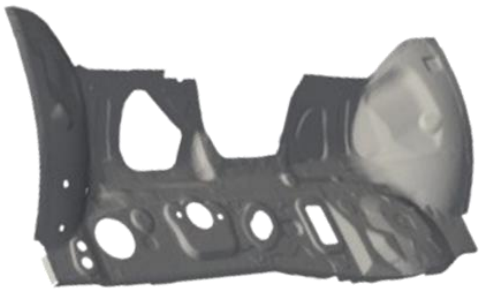
SCALE divis

ClearVu AI model

training data, accuracy, performance

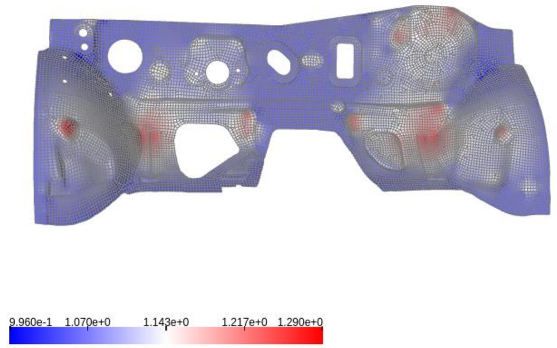
UmMatCrashML workflow

FEM model of a part

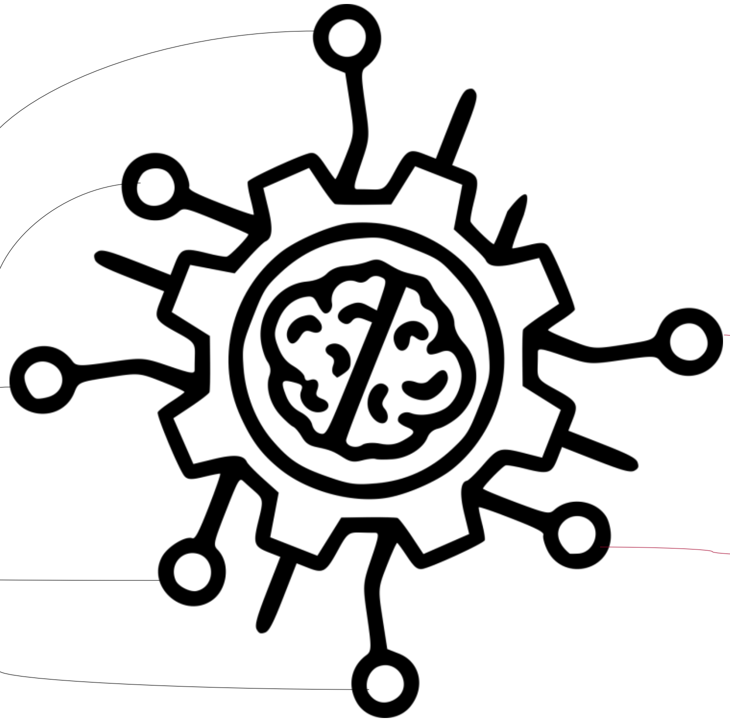


UmMatCrashML

Initial conditions of a part



Input vector
Geometry: Ar_uvmap
Material: sig_0
Material: sig_c
Material: eps_c
Material: sig_a



Output vector
Ar-lsdyna (→ sheet thickness)
effective plastic strain

ClearVu AI model

Integration

ClearVu Python Package

- CVA-Python (Python Paket)
- trained AI model package (Python Paket)
- Integrated models for fast mapping
- Updateable via standard Python mechanisms

ClearVu Python Package

```
import pandas
from cva import Base
from umatcra import Base
manager = Base.Manager()
```

```
manager.get_input_variable_names()
```

```
{'Ar_lsdyna': ['Ar_uvmap', 'mat_eps_c', 'mat_sig_0', 'mat_sig_a', 'mat_sig_c'],
 'epseqpl': ['Ar_uvmap', 'mat_eps_c', 'mat_sig_0', 'mat_sig_a', 'mat_sig_c']}
```

```
df = pandas.DataFrame([[.2, .1, 200., 300, 250]],
                      columns=['Ar_uvmap', 'mat_eps_c', 'mat_sig_0', 'mat_sig_a', 'mat_sig_c'])
df
```

	Ar_uvmap	mat_eps_c	mat_sig_0	mat_sig_a	mat_sig_c
0	0.2	0.1	200.0	300	250

```
manager.predict(df)
```

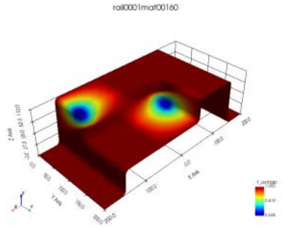
	Ar_lsdyna	epseqpl
0	1.084302	0.138254

umatcra Python Package (0.1.1)

Search docs

CONTENTS:

- The umatcra Python Package
  - Changelog
  - Getting started
- Python API



The umatcra Python Package

The umatcra Python Package

This package contains a specialized multi-objective optimizer with focus of minimizing the number of evaluation calls. In addition, restrictions can be added to the optimization task.

Changelog

Version	Changes
0.0.1	Initial version.

Getting started

First, you need to instantiate a manager object.

```
from umatcra import Base

manager = Base.Manager()
```

The manager object loads the models for Ar_lsdyna and epseqpl. You can check the names for the input and output variables with

```
manager.get_input_variable_names()
manager.get_output_variable_names()
```

In order to use the models, you need a pandas data frame with columns named as the input variables of the models.

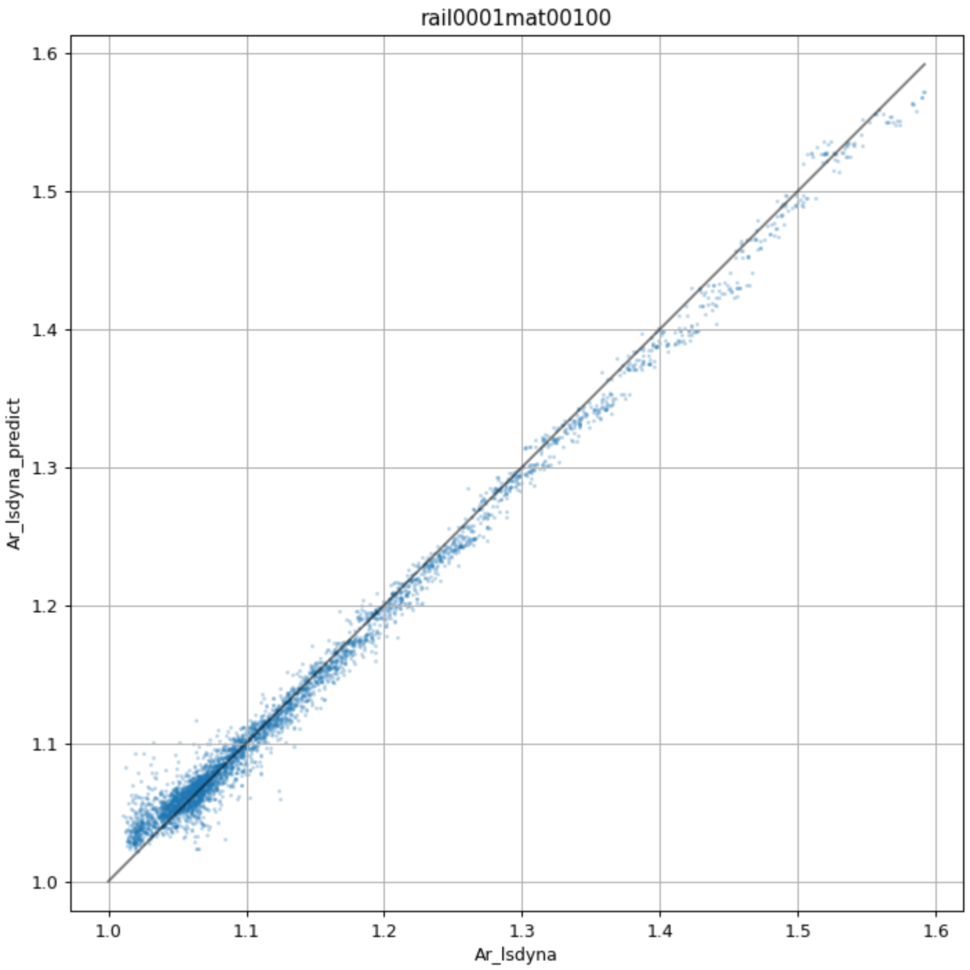
	Ar_uvmap	mat_eps_c	mat_sig_0	mat_sig_a	mat_sig_c
0	1.046333	0.1921093	168.35937	325.78125	228.51562
1	1.417264	0.1601818	648.98071	1324.4262	909.67407
...	...	...	...	...	...

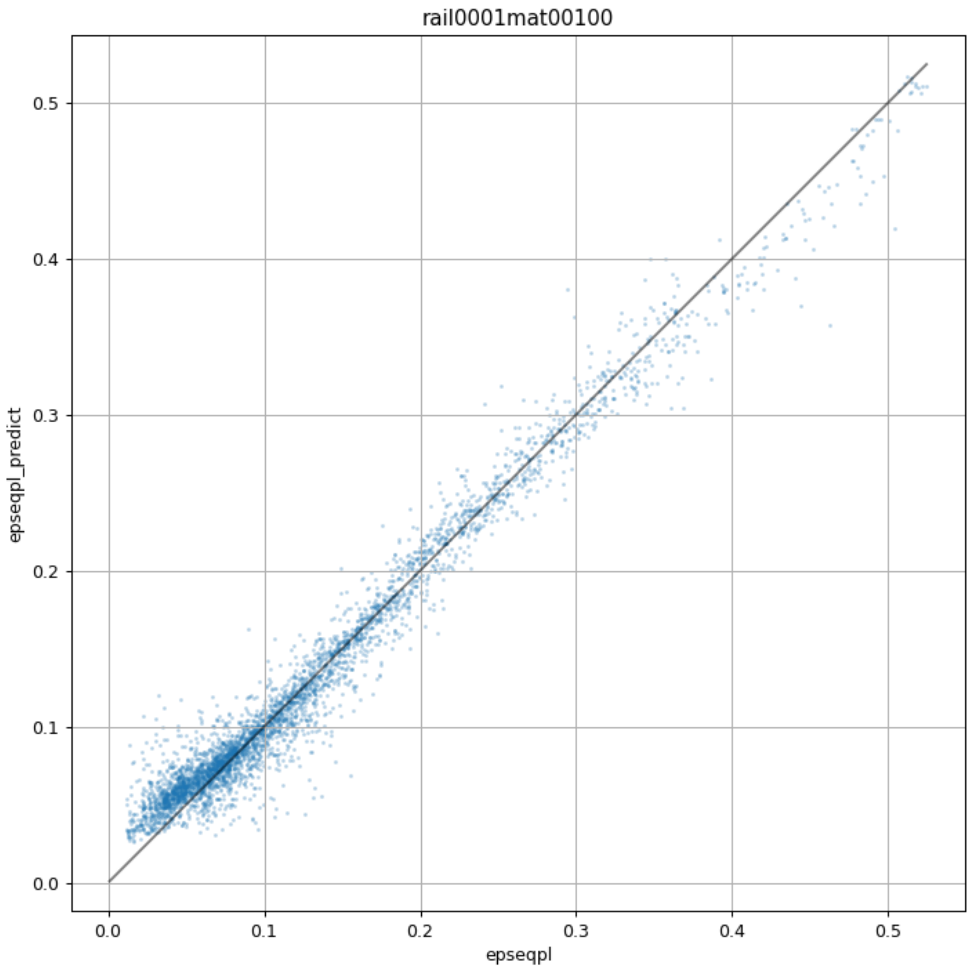
```
df = pandas.DataFrame(....)

manager.predict(df)
```

SCALE dvis

Accuracy LFS vs. AI & Performance

- prediction time for 10000 Elements 0.767 s ± 43.3 ms o Intel i7-8550U CPU @ 1.80GHz

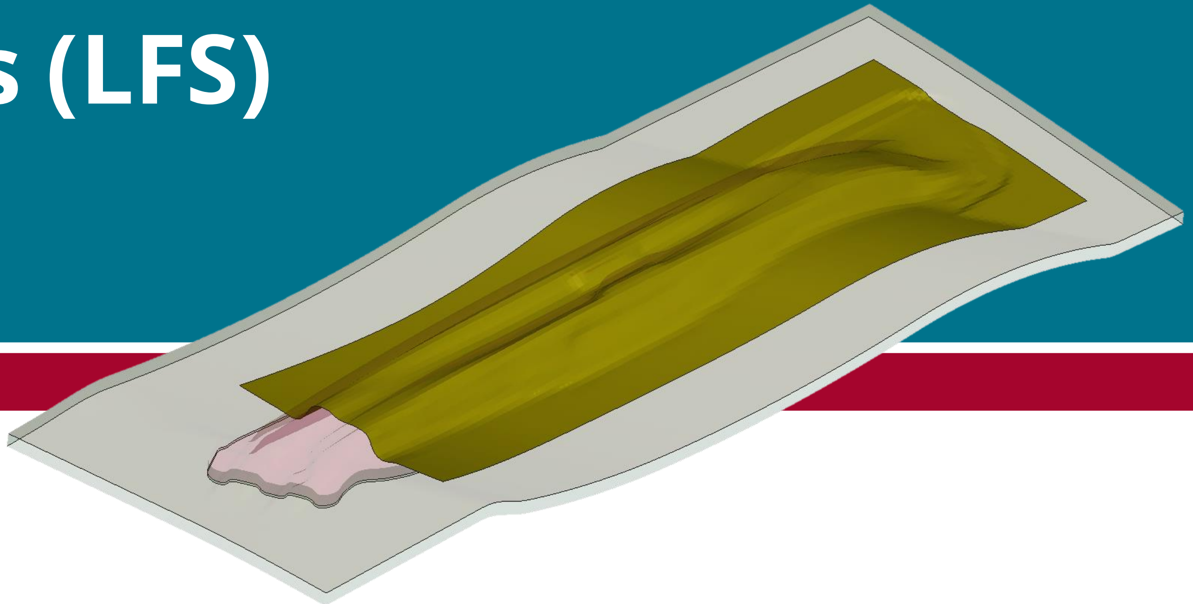




SCALE dvis

Verification & Validation

High fidelity simulations (HFS)
Low fidelity simualtions (LFS)



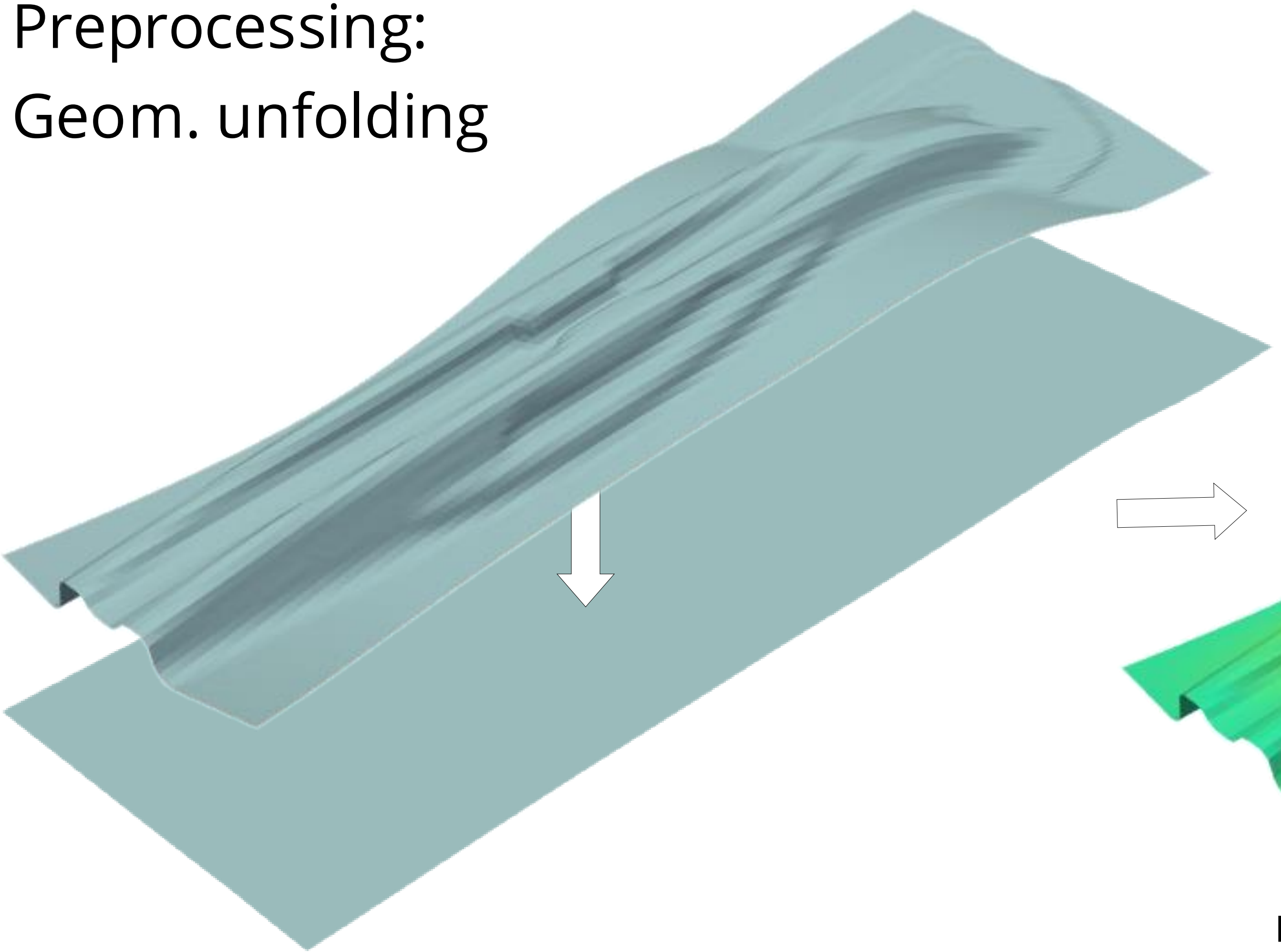
FE model „KA Demonstrator“: KIRCHHOFF Automotive Deutschland GmbH

SCALE dvis

Validation FE model „KA Demonstrator": Mesh unfolding → Ar_uvmap

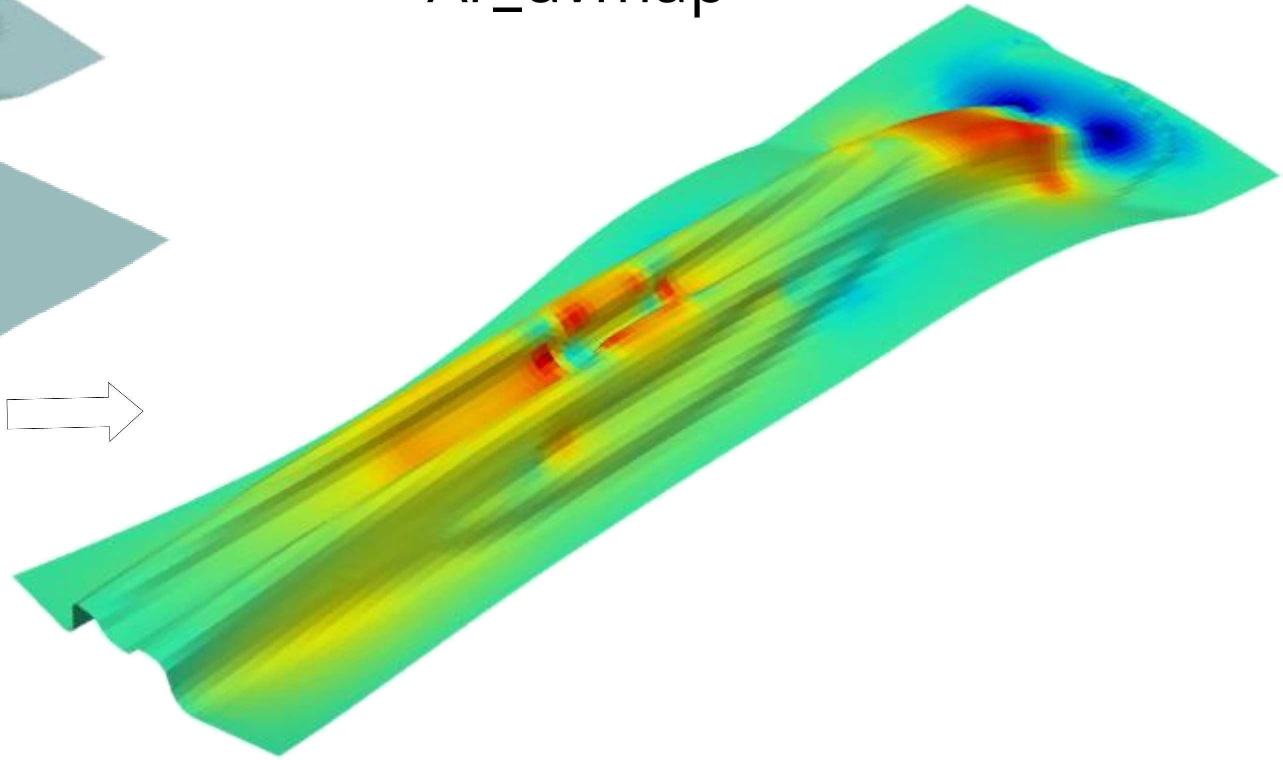
Preprocessing:

Geom. unfolding



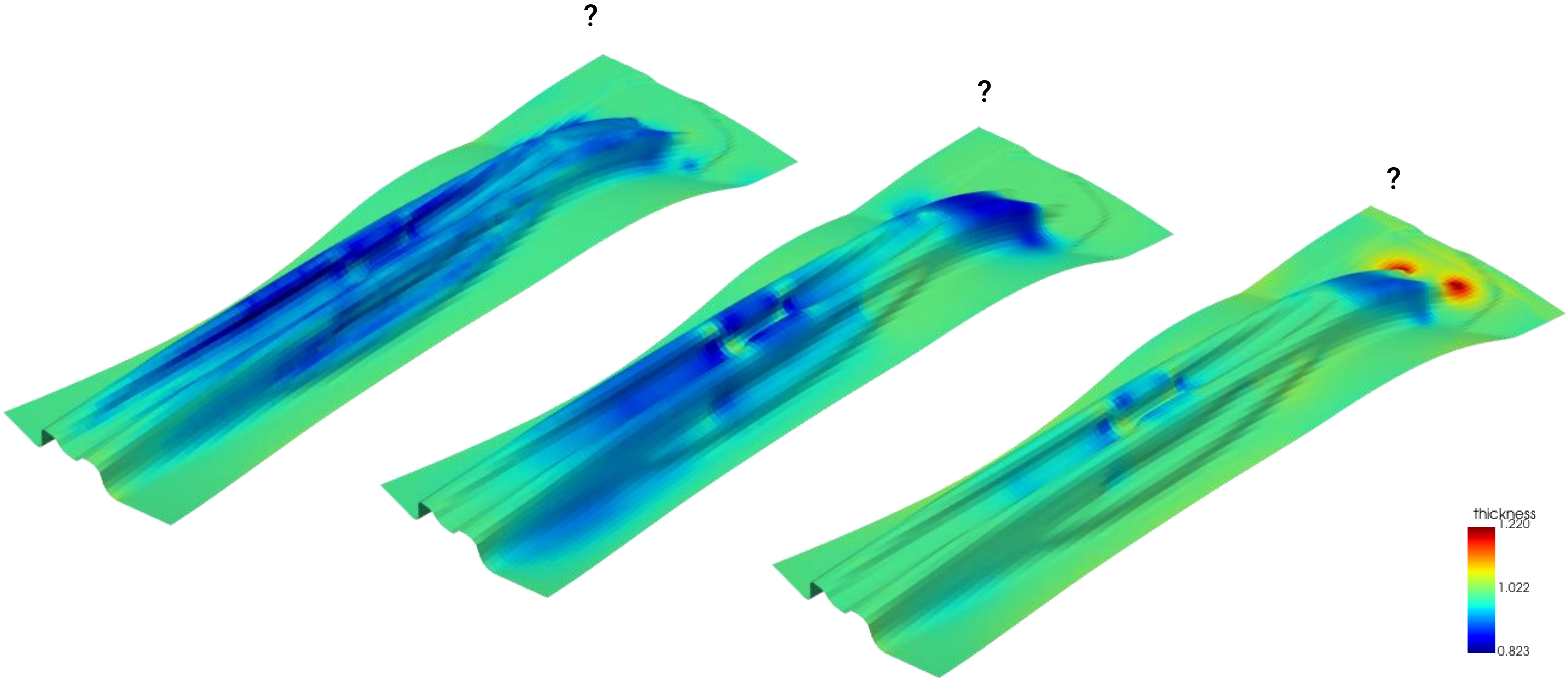
„Geometric" Area strain

Ar_uvmap

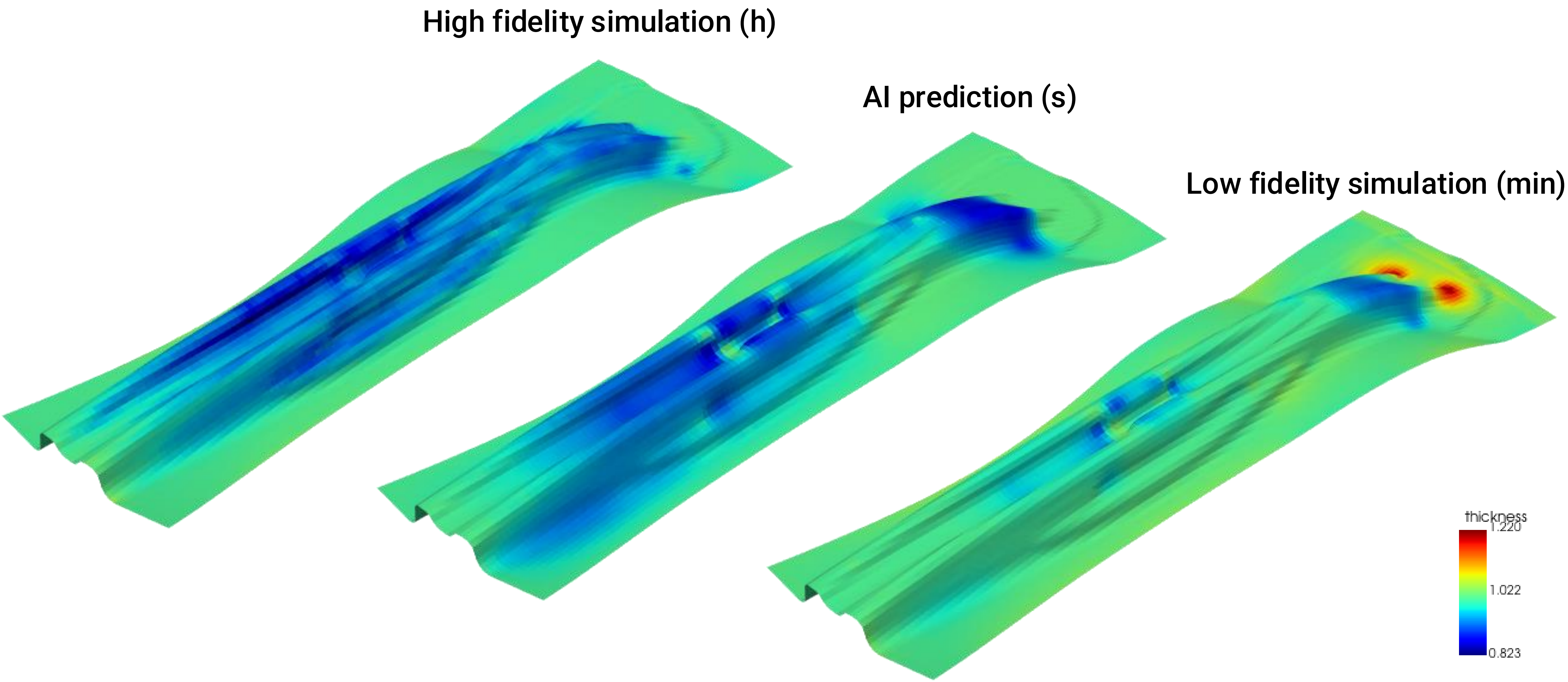


FE model „ssab": KIRCHHOFF Automotive Deutschland GmbH

Validation ssab HS150Y273T284C450A - Which one is HFS, LFS, SKI?



Validation ssab HS150Y273T284C450A - Which one is HFS, LFS, SKI?

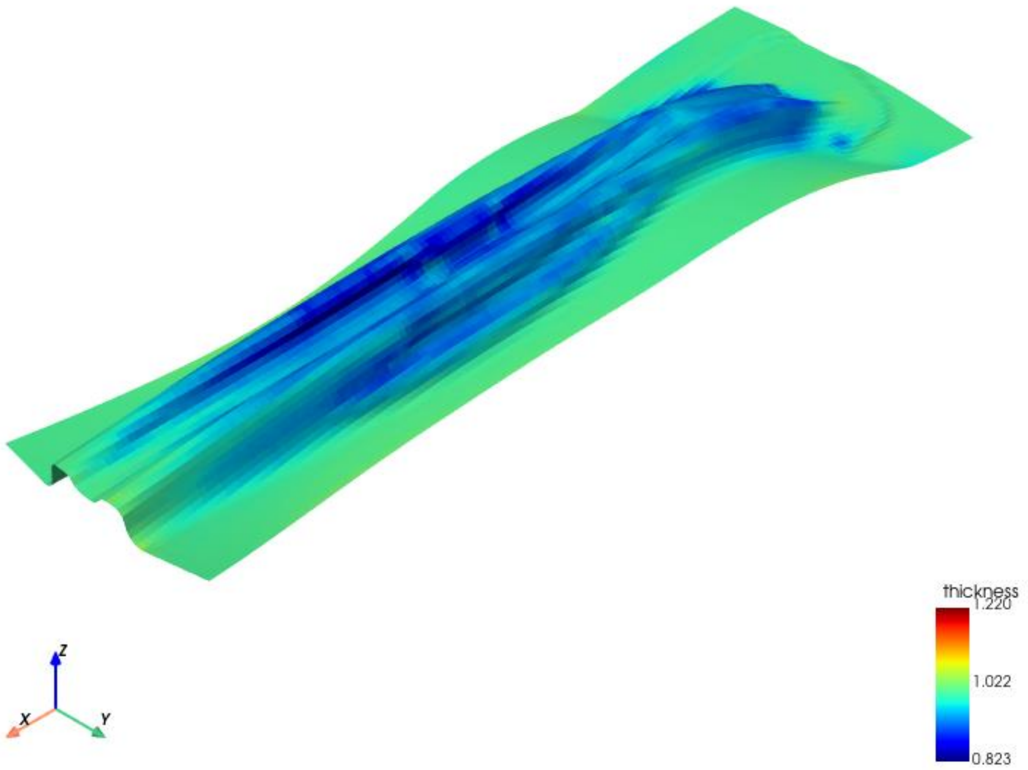


Validation - HS150Y273T284C450A – HFS vs. LFS

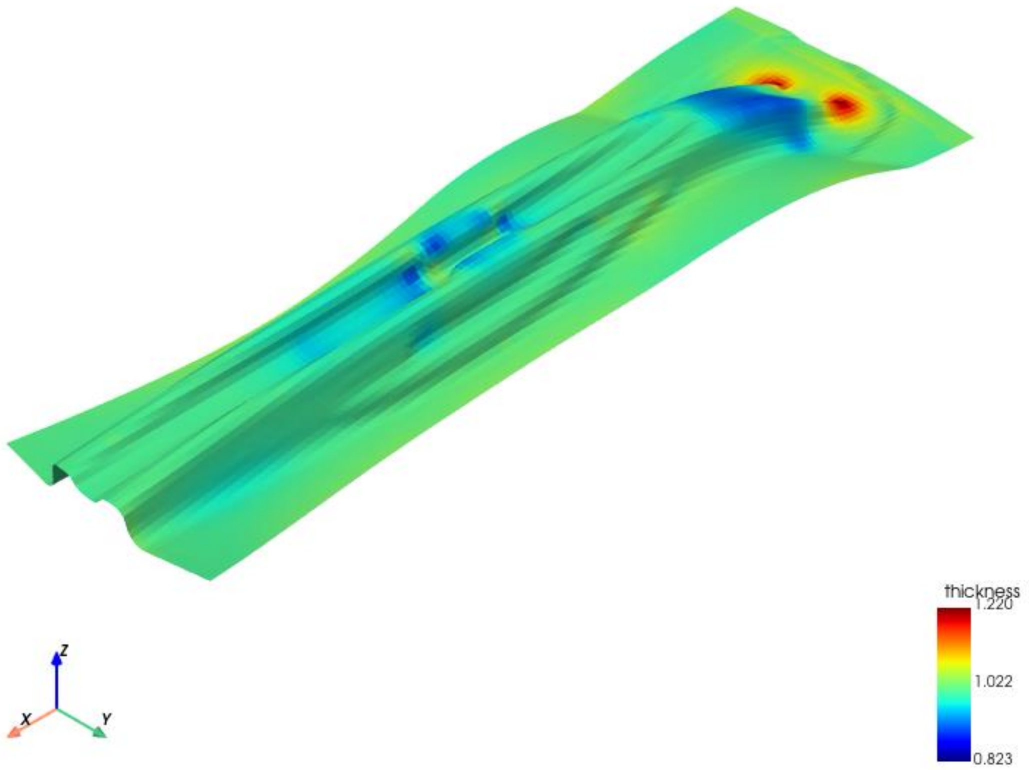
HFS - HS150Y273T284C450A - thickness (0.82 - 1.03) mm

LFS - HS150Y273T284C450A - thickness (0.88 - 1.22) mm

High fidelity simulation (h)



Low fidelity simulation (min)

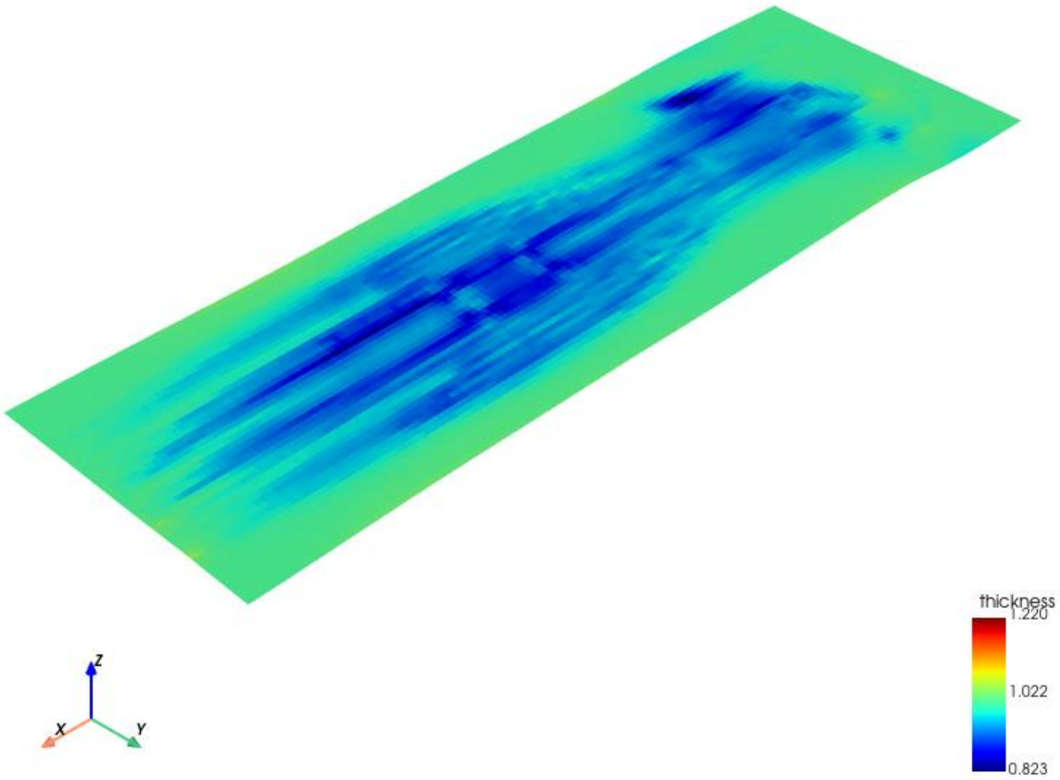


Validation - HS150Y273T284C450A – HFS vs. LFS

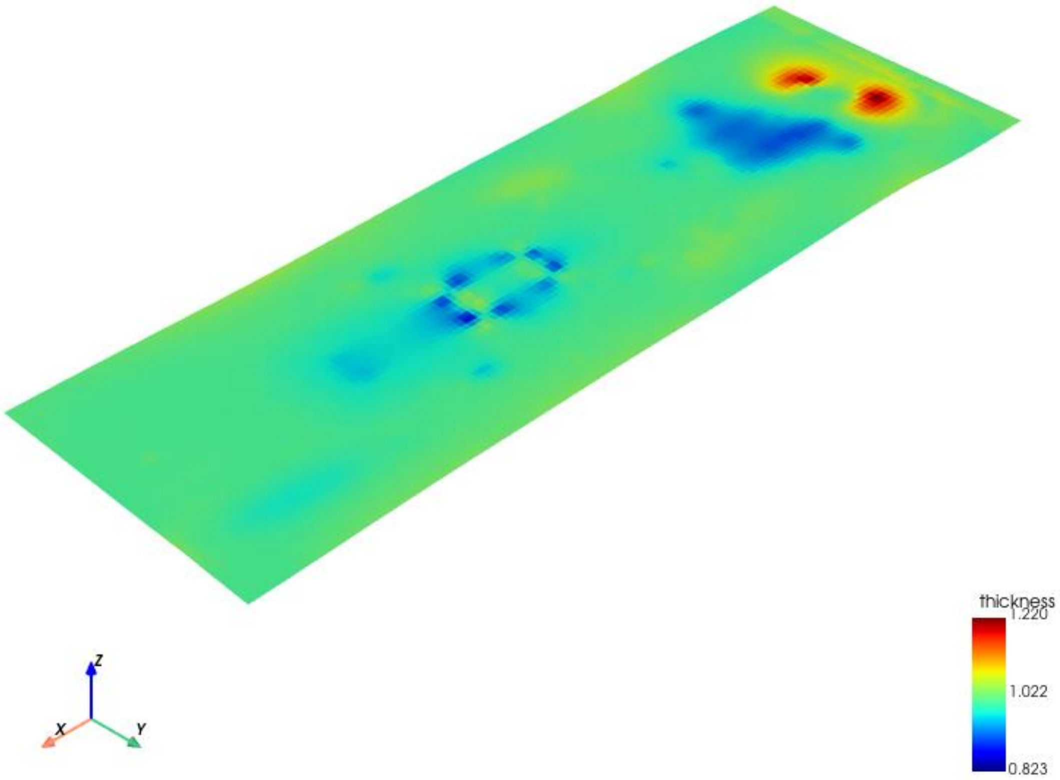
HFS - HS150Y273T284C450A - thickness (0.82 - 1.03) mm

LFS - HS150Y273T284C450A - thickness (0.88 - 1.22) mm

High fidelity simulation (h)

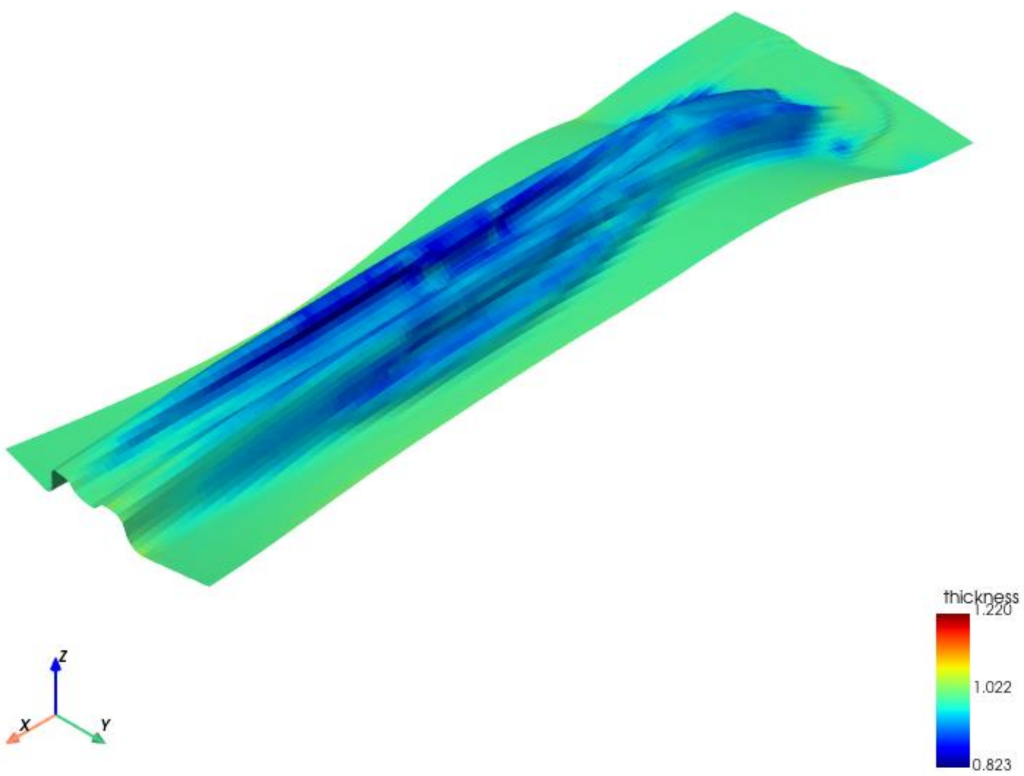


Low fidelity simulation (min)

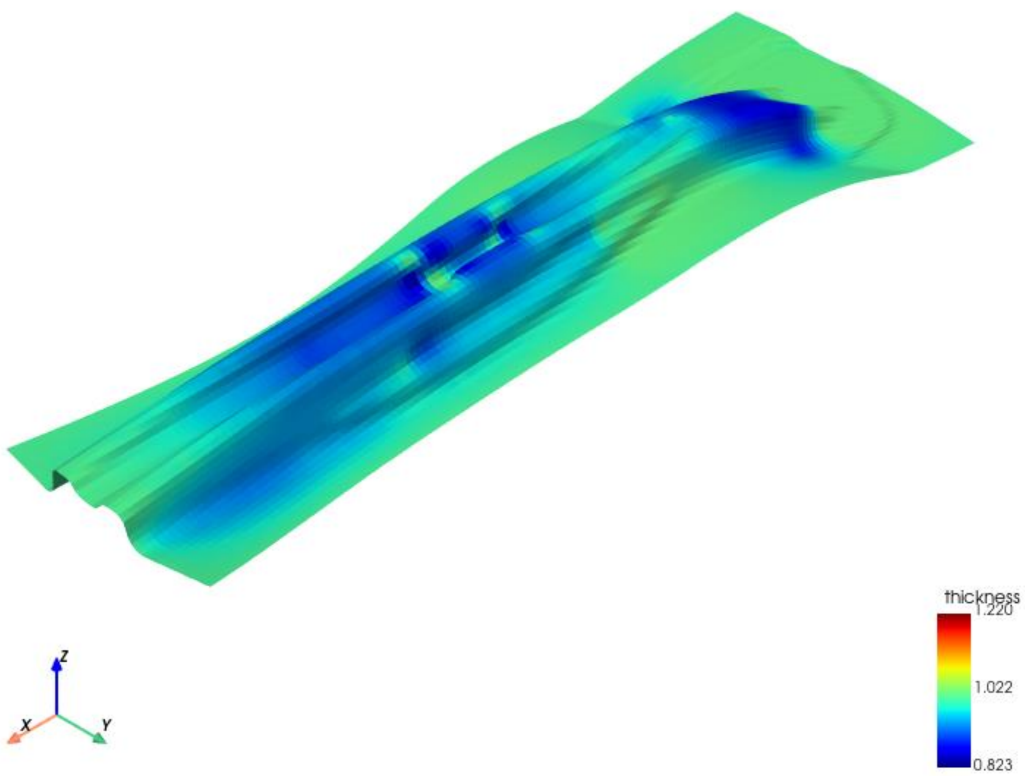


Validation - HS150Y273T284C450A – HFS vs. SKIc

High fidelity simulation (h)



AI prediction (s)

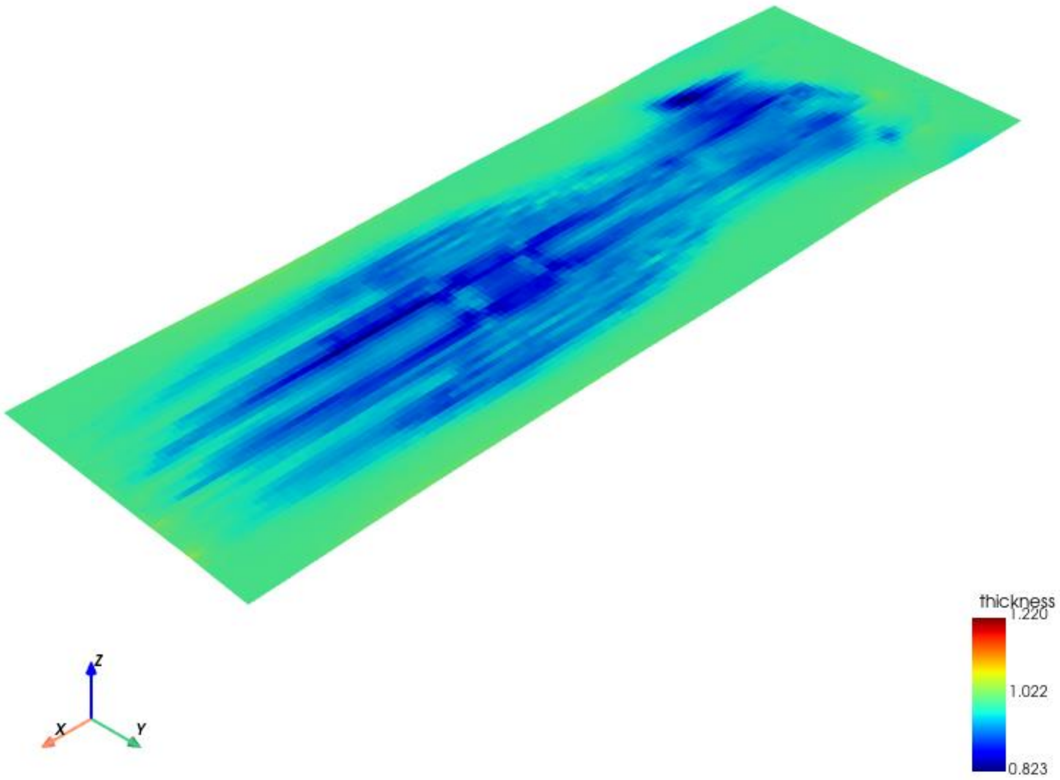


Validation - HS150Y273T284C450A – HFS vs. SKIc

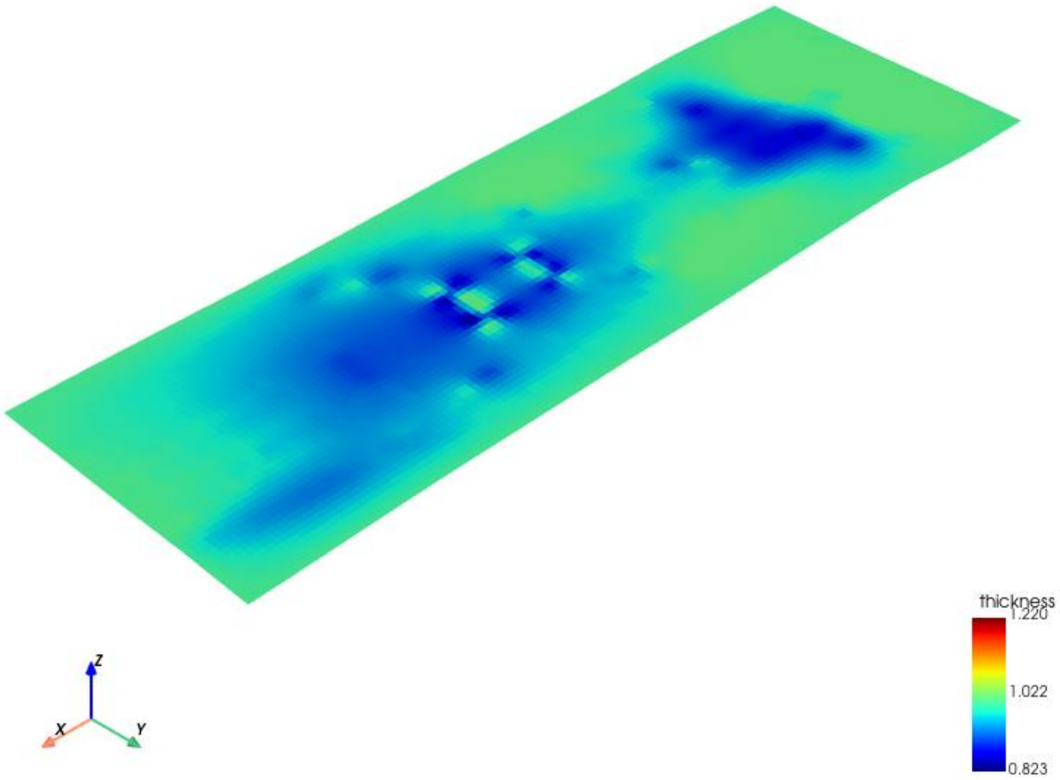
HFS - HS150Y273T284C450A - thickness (0.82 - 1.03) mm

SKIc - HS150Y273T284C450A - thickness (0.83 - 1.01) mm

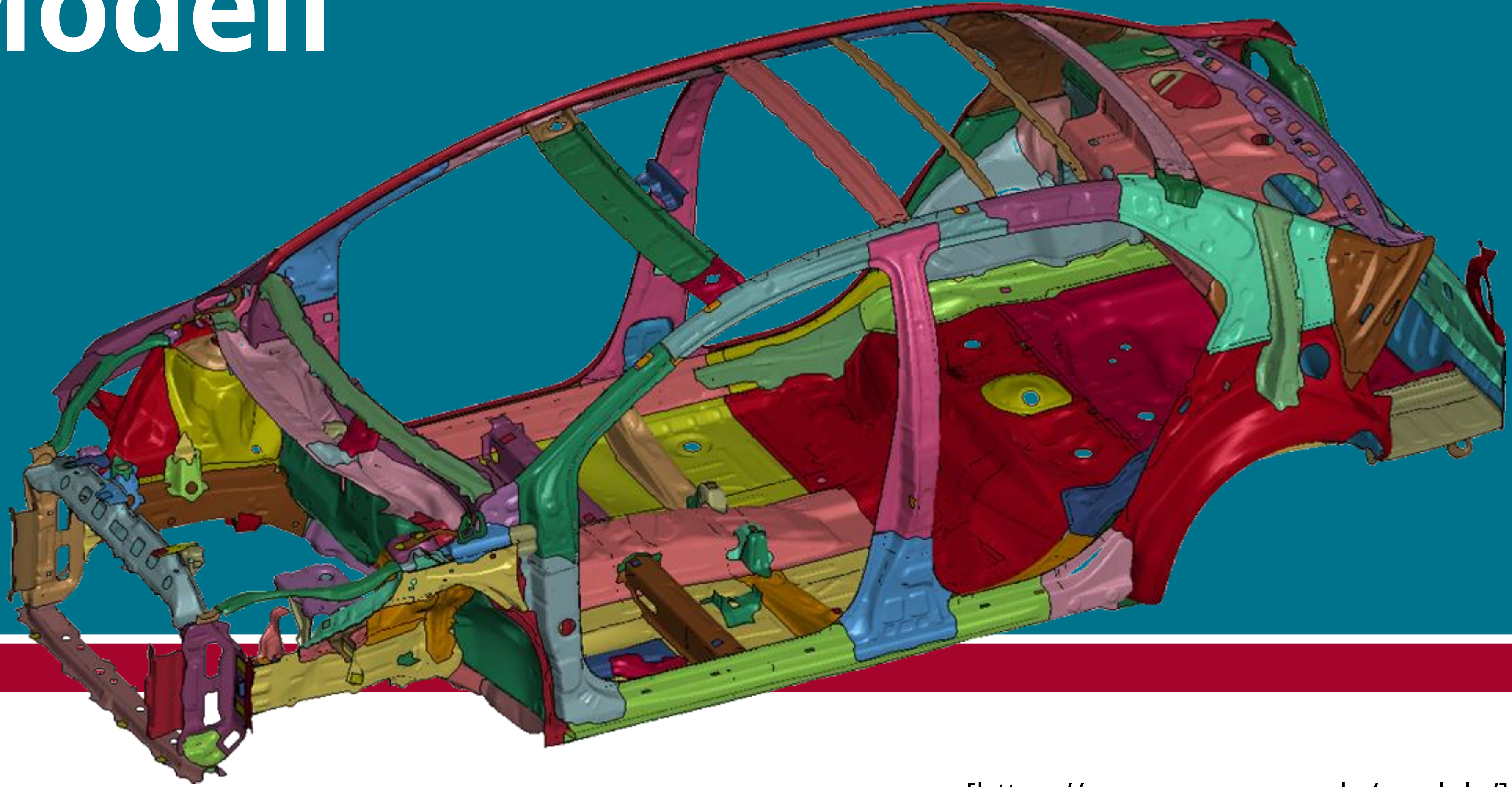
High fidelity simulation (h)



AI prediction (s)



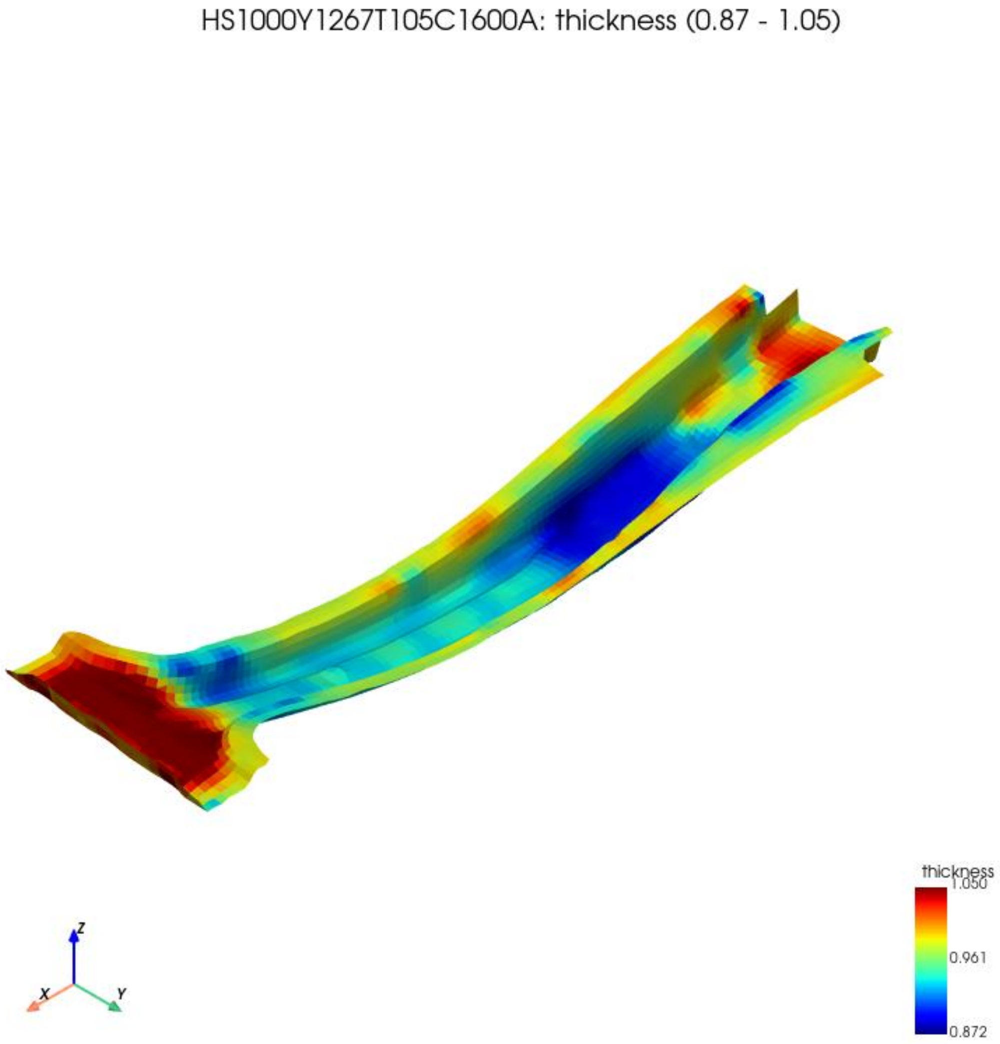
Yaris Modell

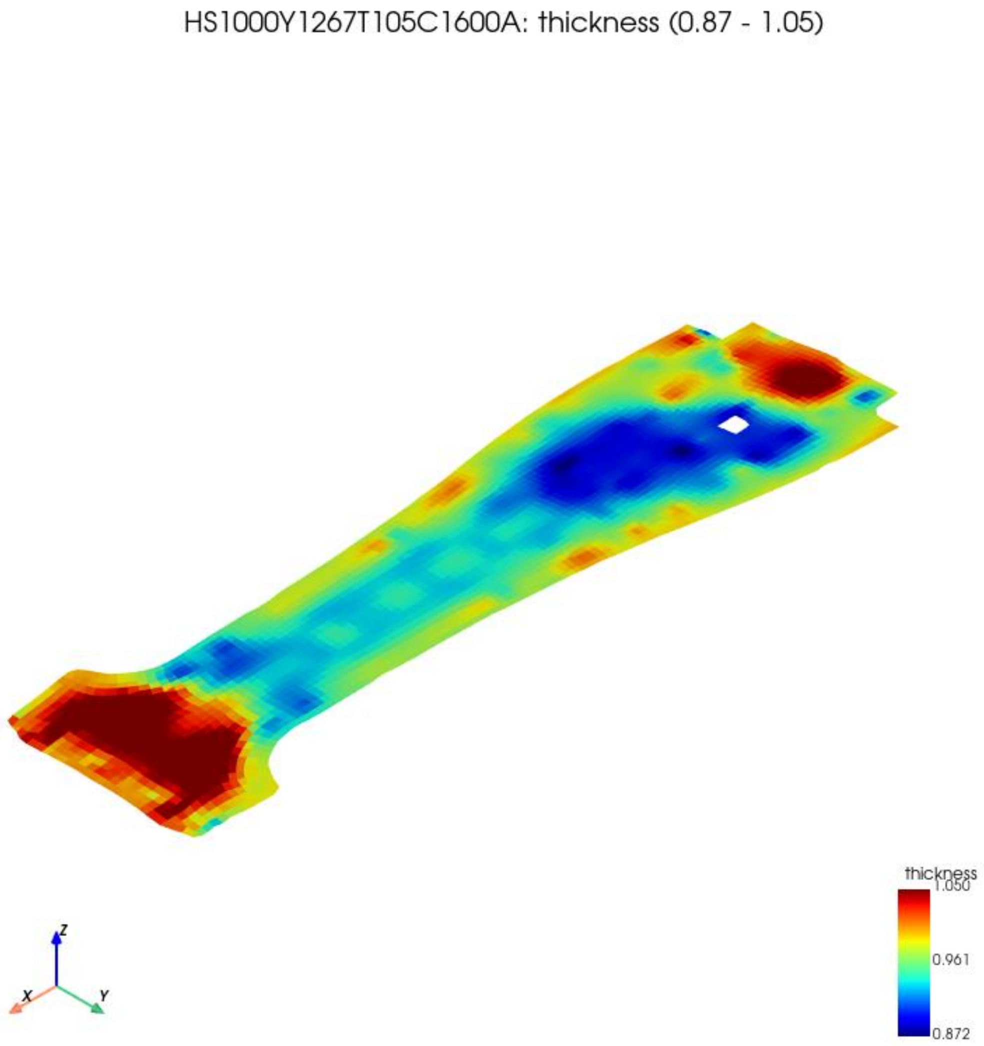


[https://www.ccsa.gmu.edu/models/]

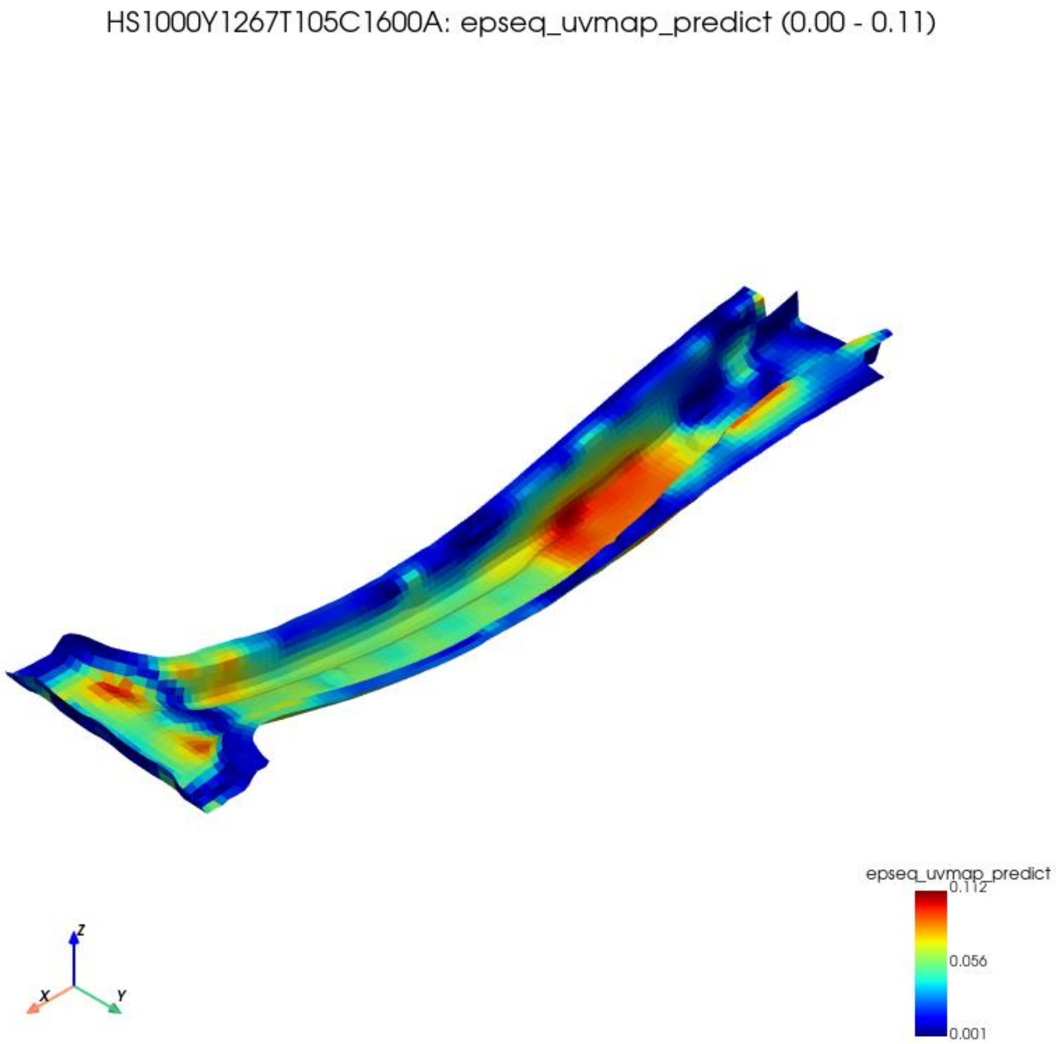
SCALE dvis

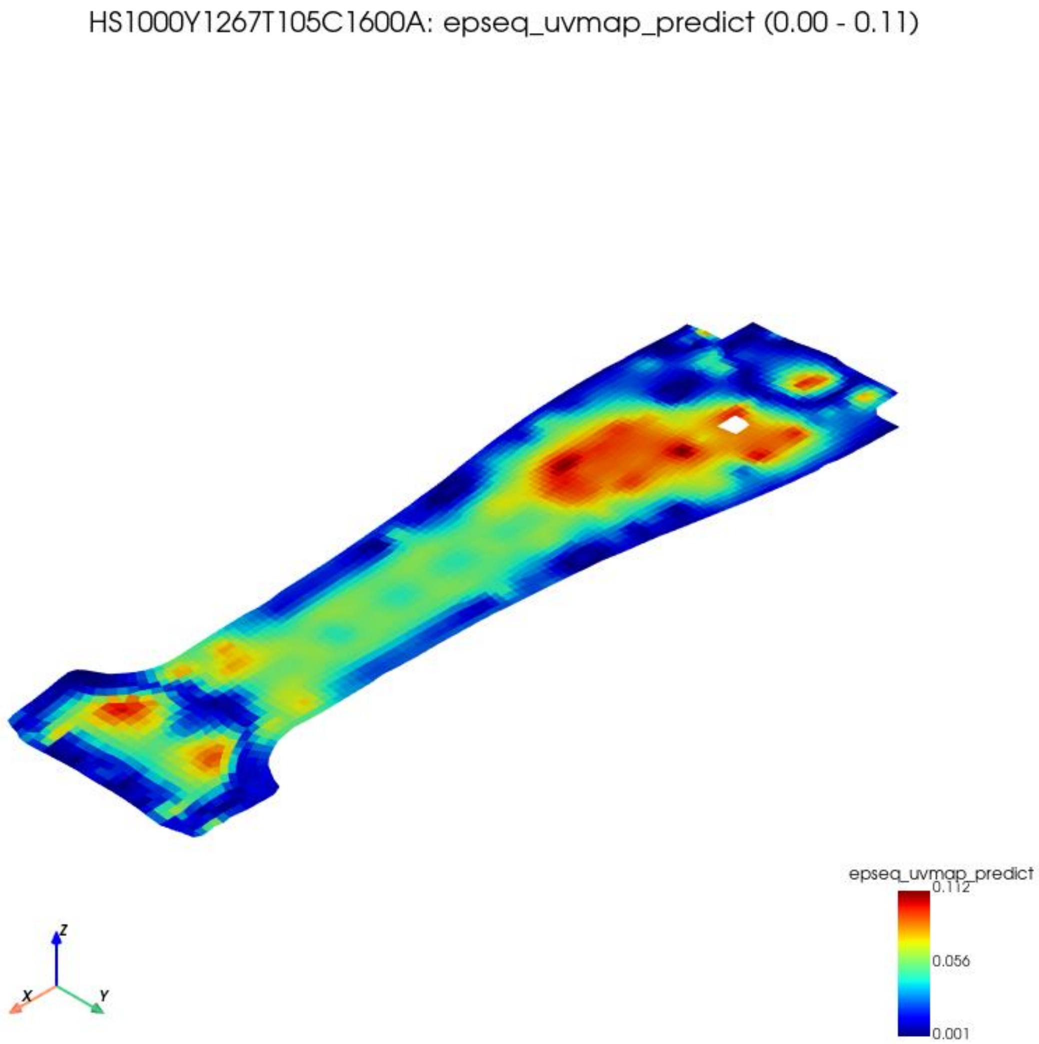
Validation – Yaris – Part 235 – b pillar upper internal r





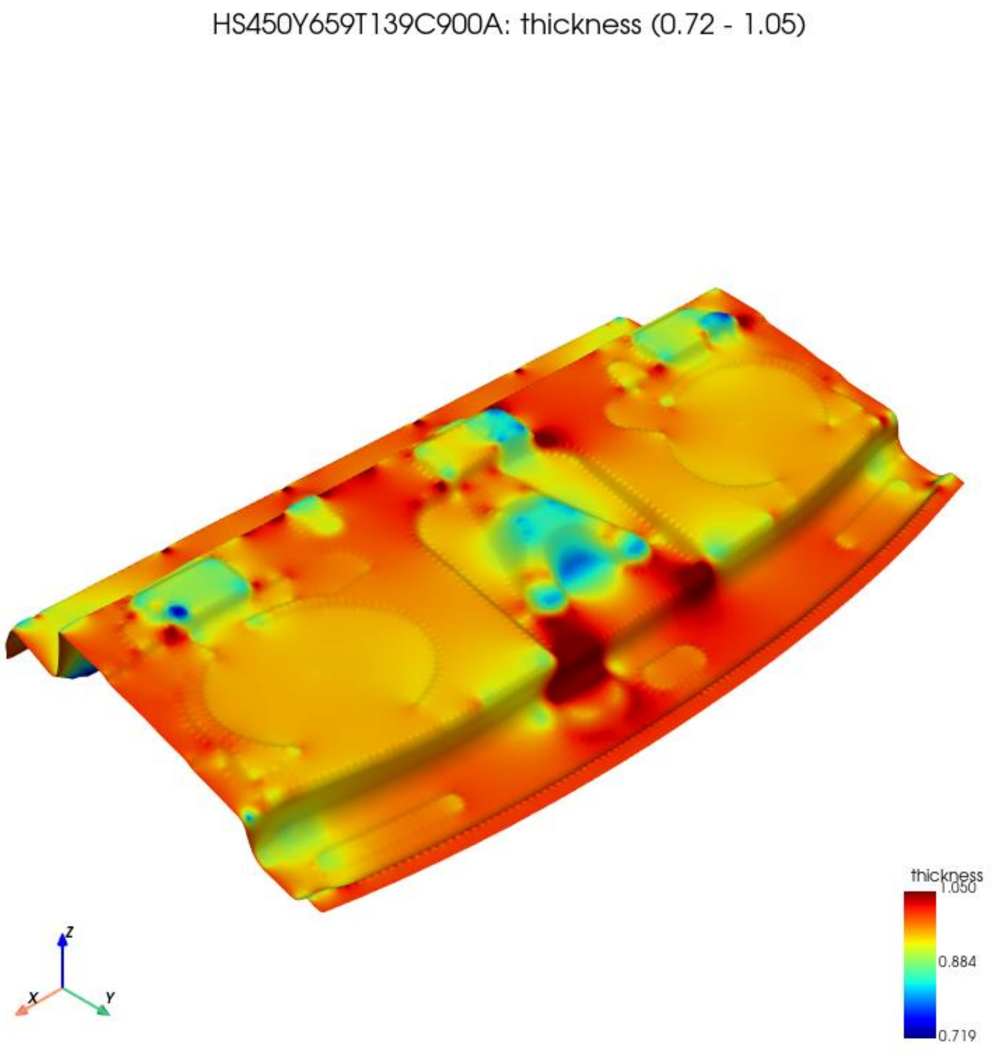
Validation – Yaris – Part 235 – b pillar upper internal r

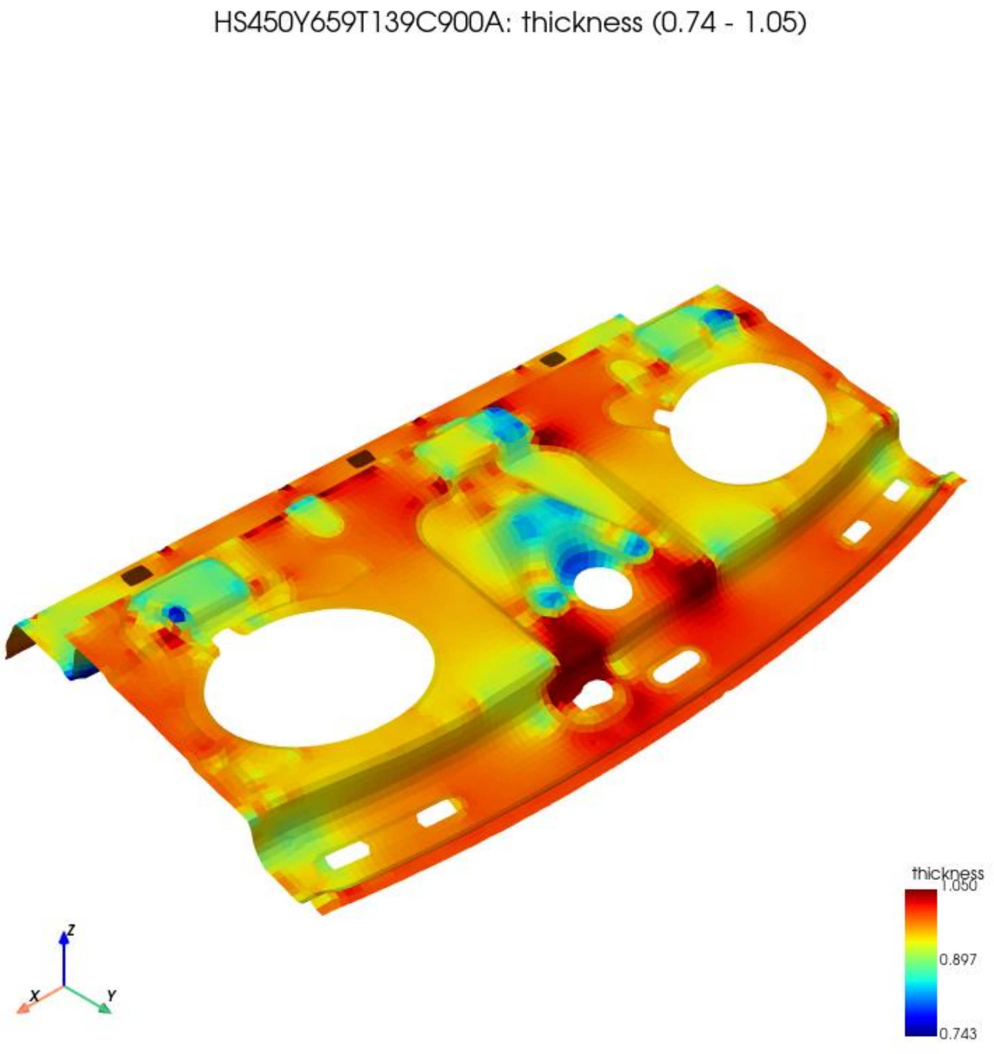




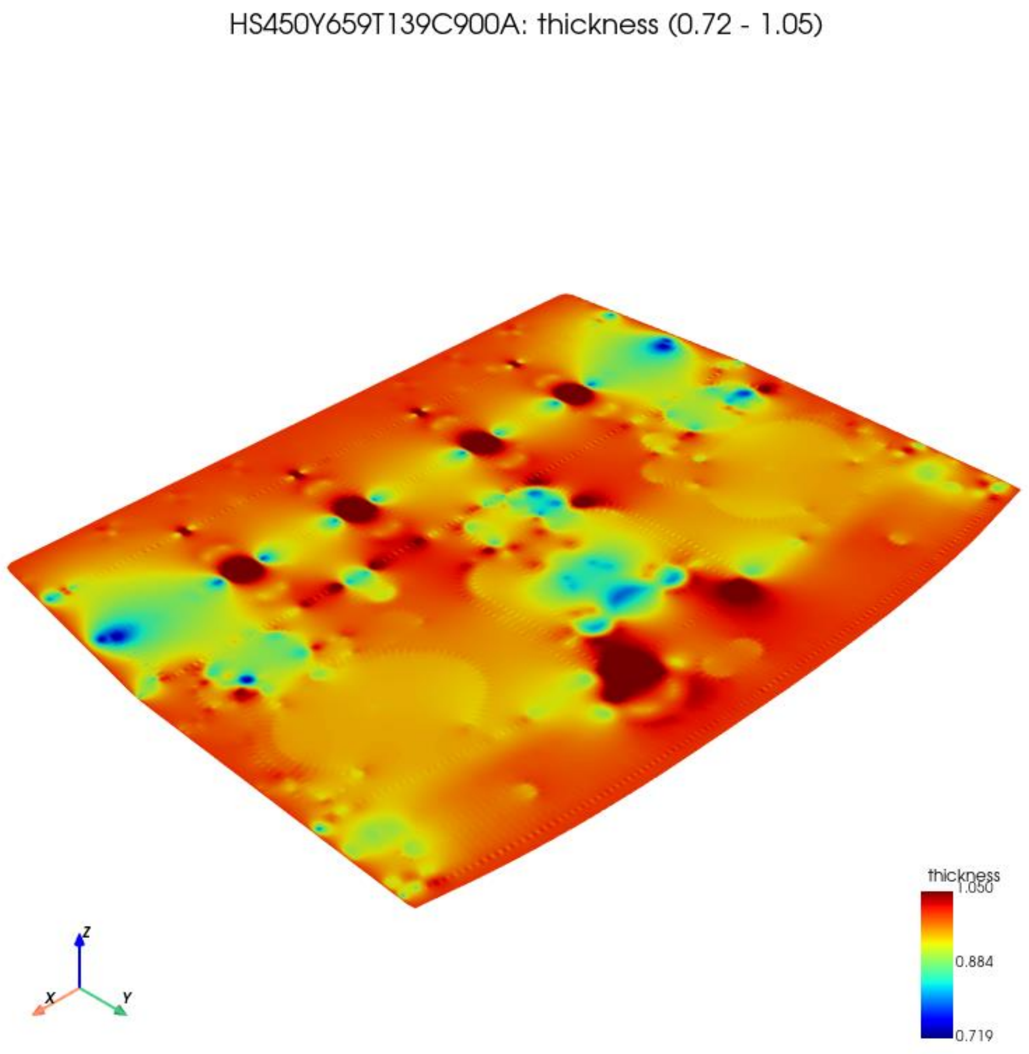
SCALE divis

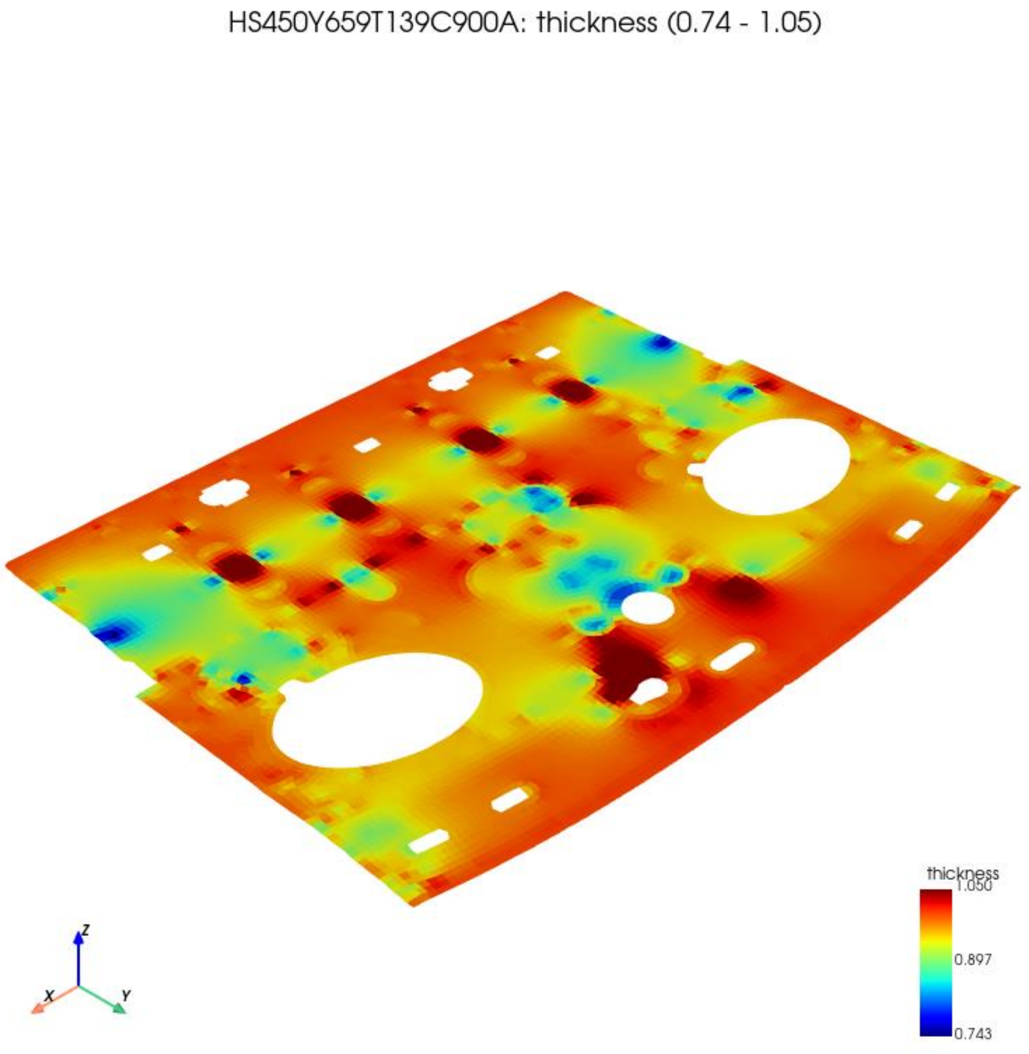
Validation – Yaris – Part 76 - trunkframe upper middle



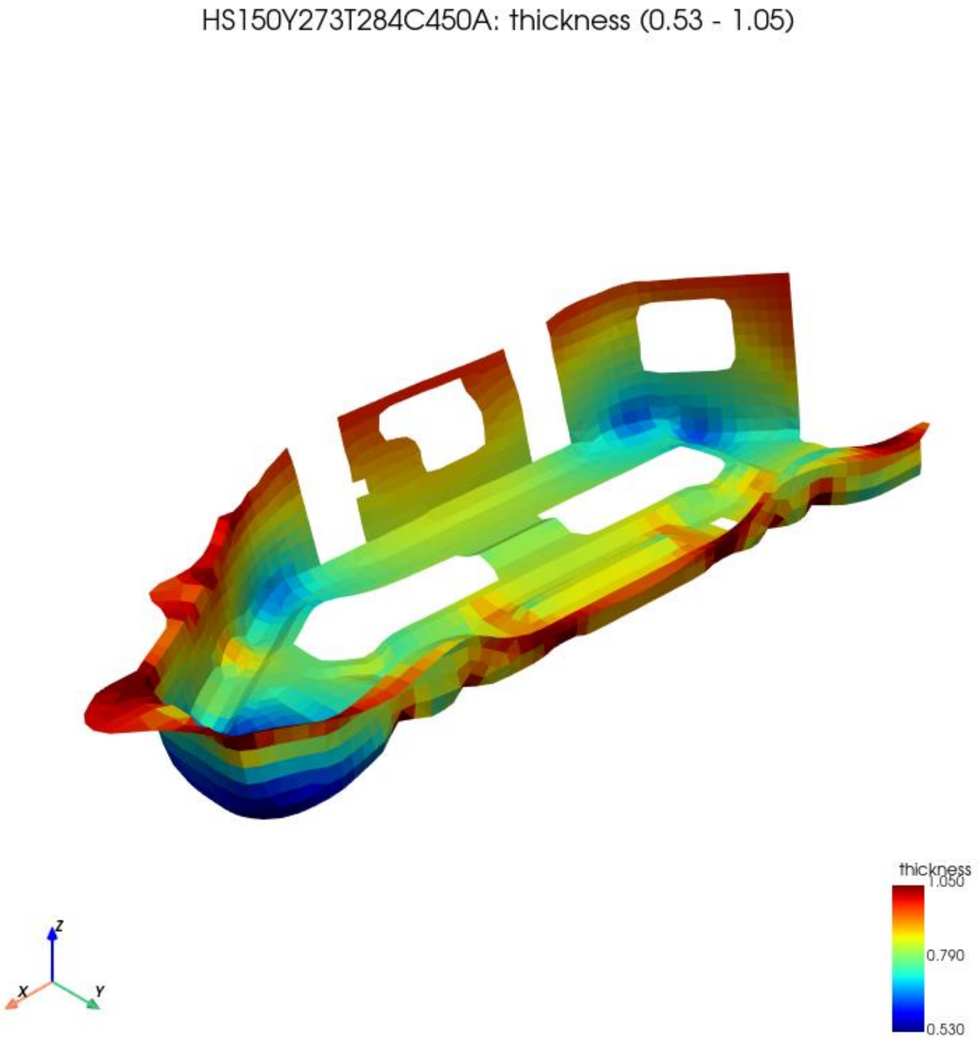


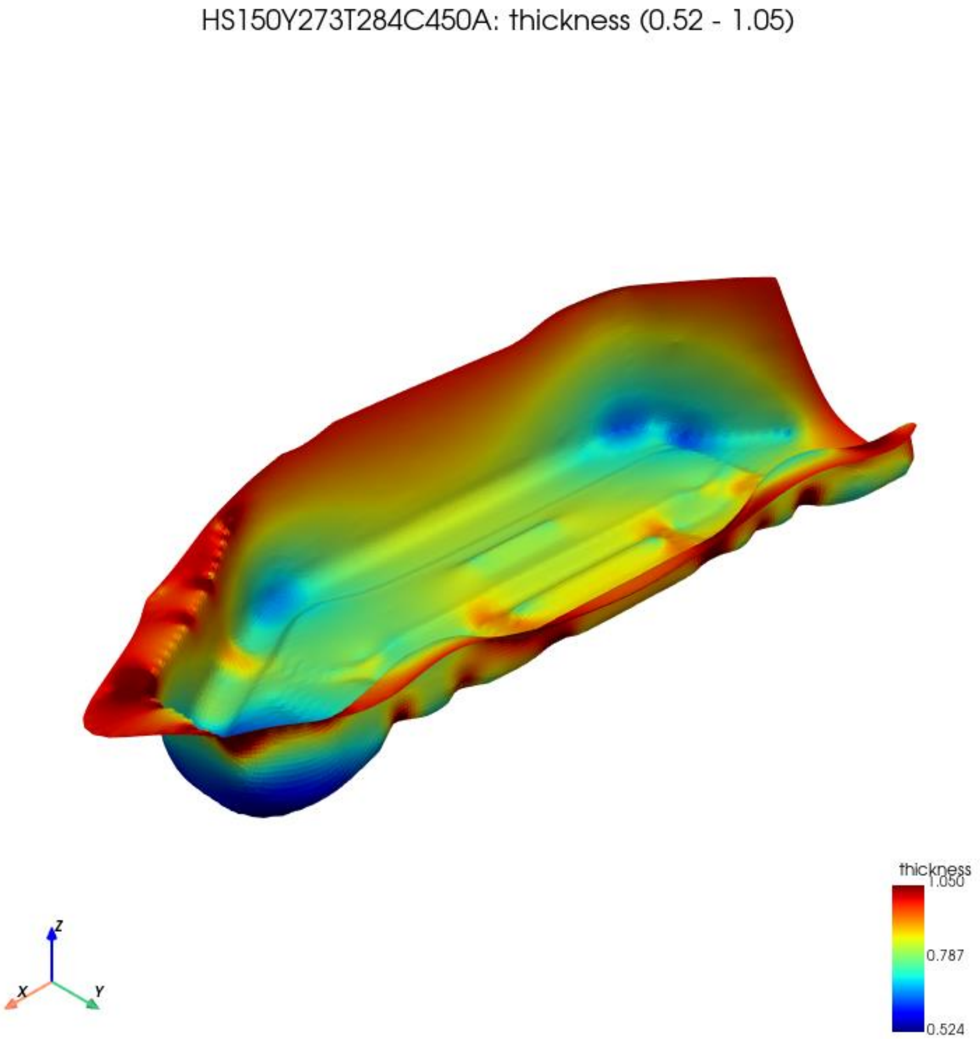
Validation – Yaris – Part 76 - trunkframe upper middle



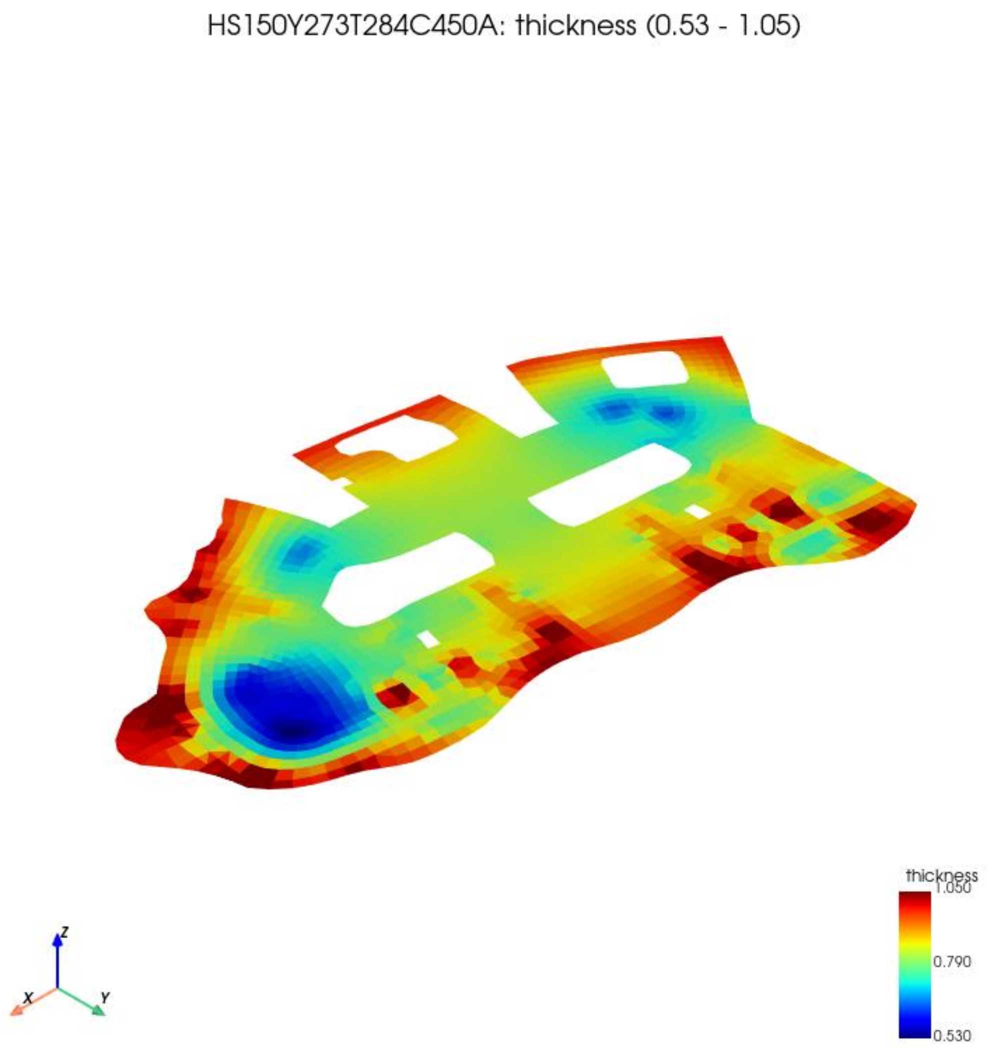


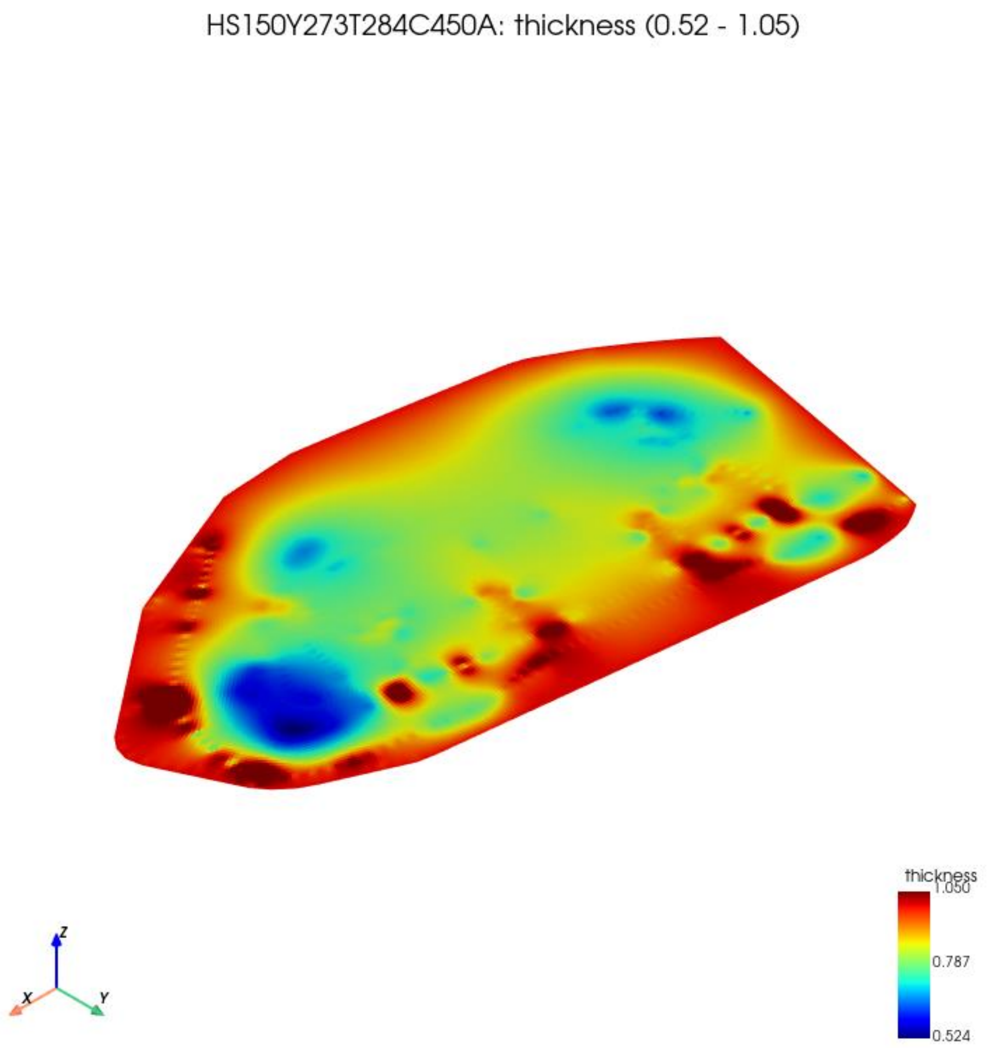
Validation – Yaris – Part 73 - radiator framer



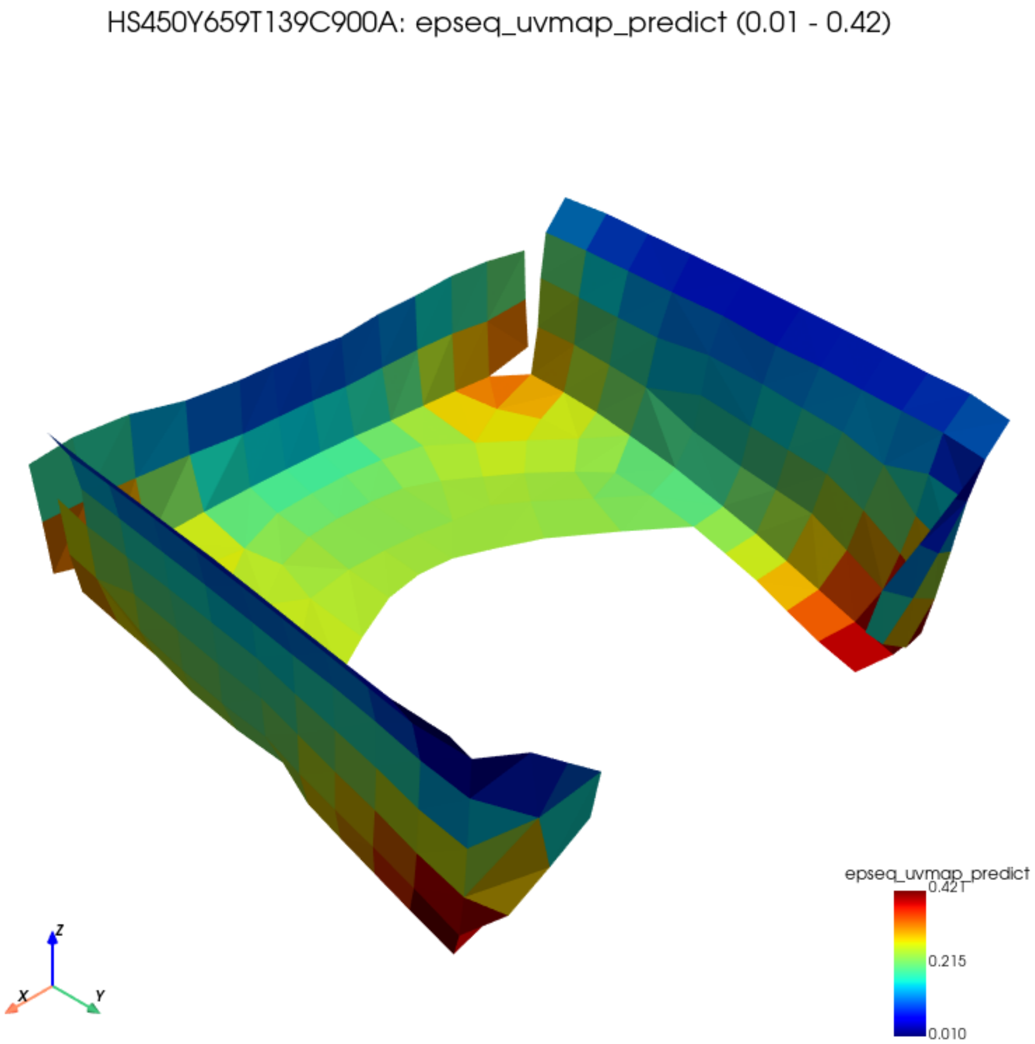


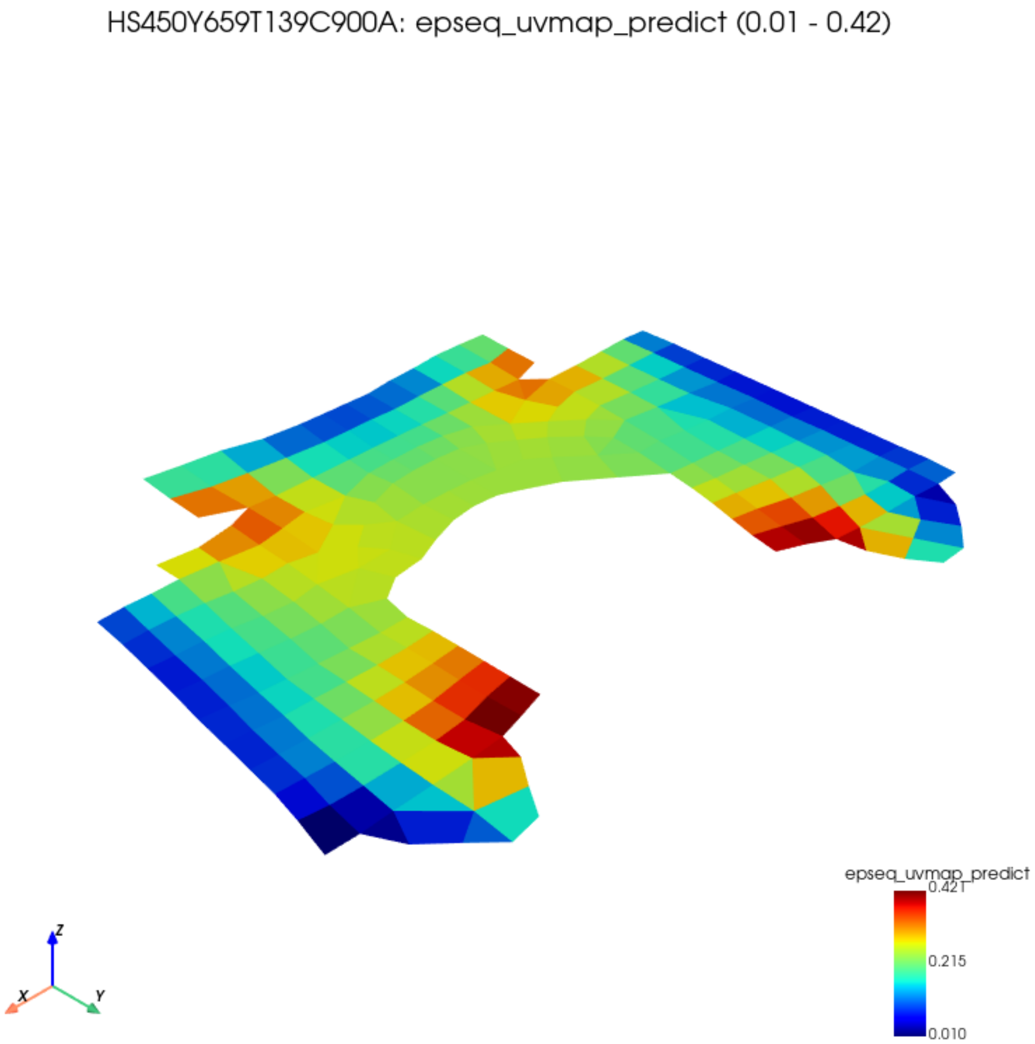
Validation – Yaris – Part 73 - radiator framer



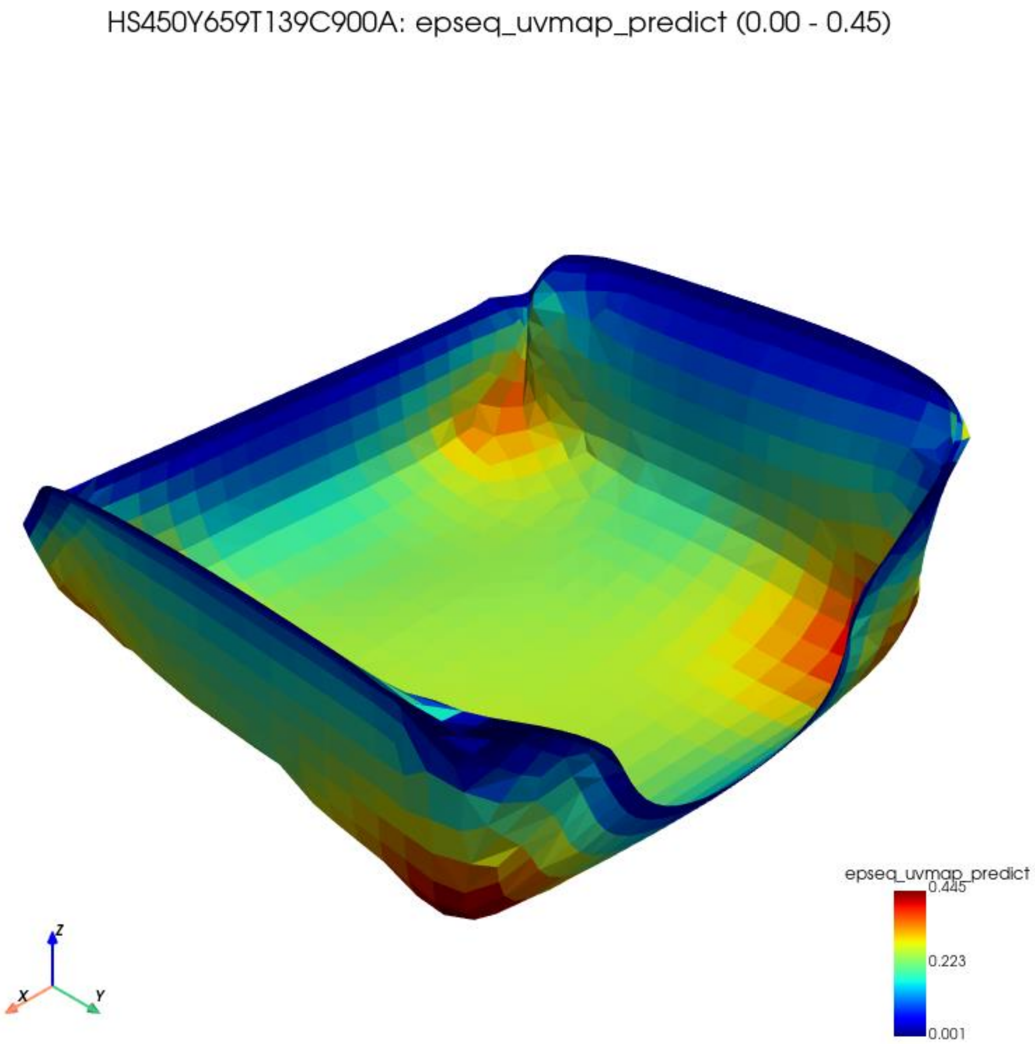


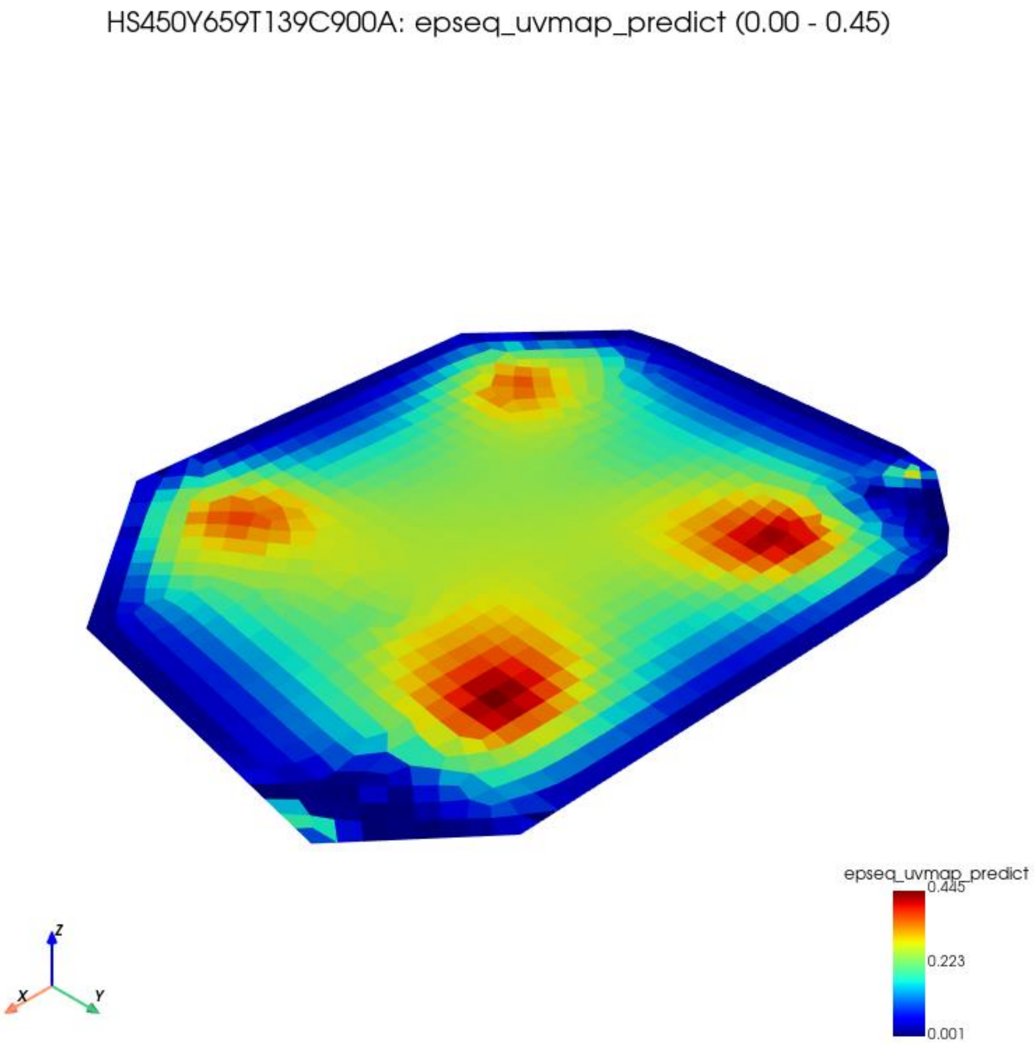
Validation – Yaris – Part 106 – rail inner support 2



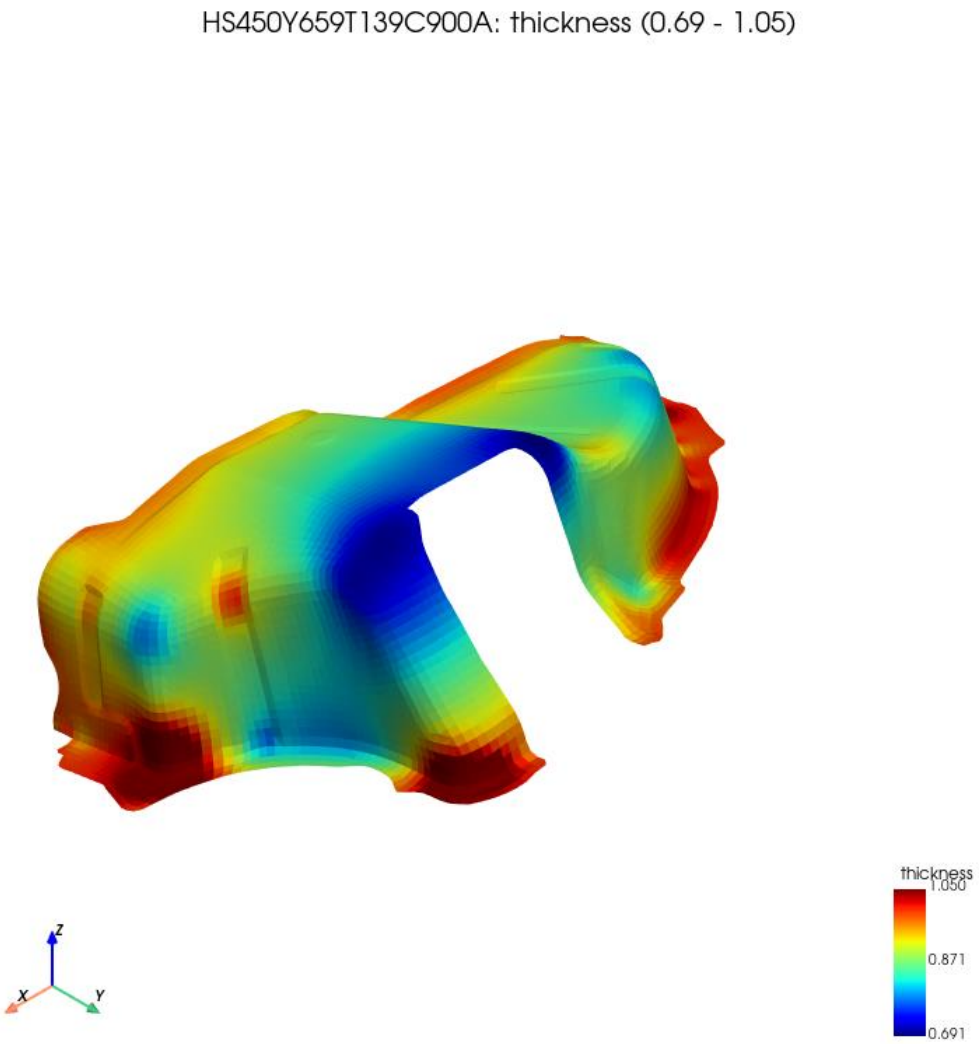


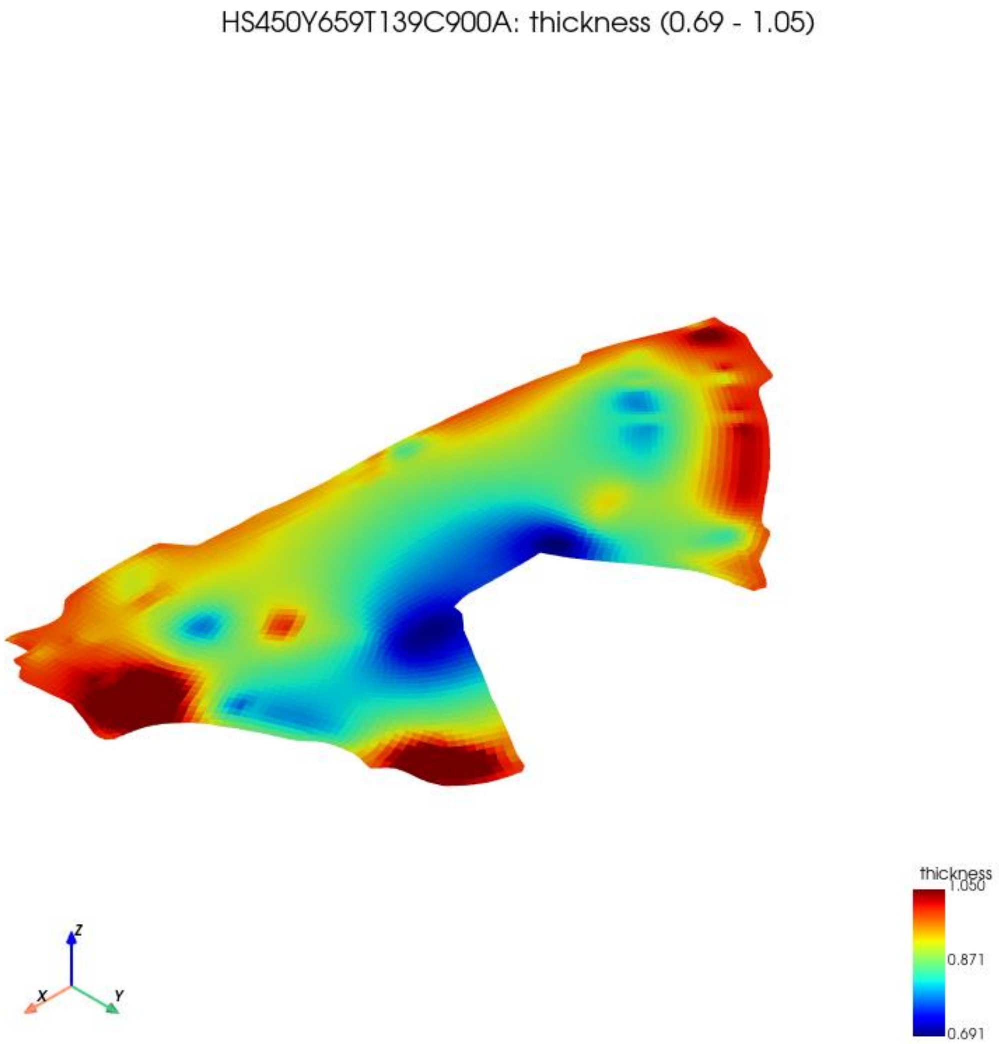
Validation – Yaris – Part 106 – rail inner support 2



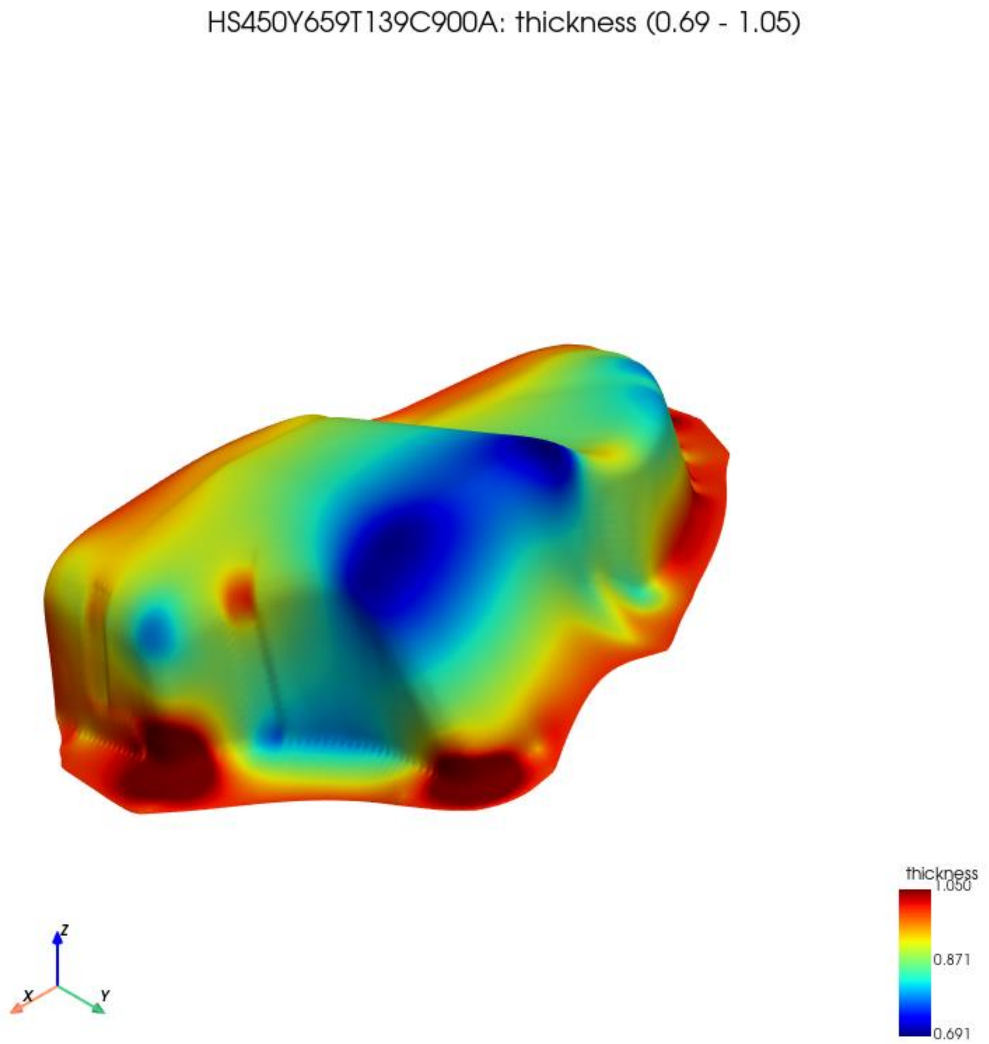


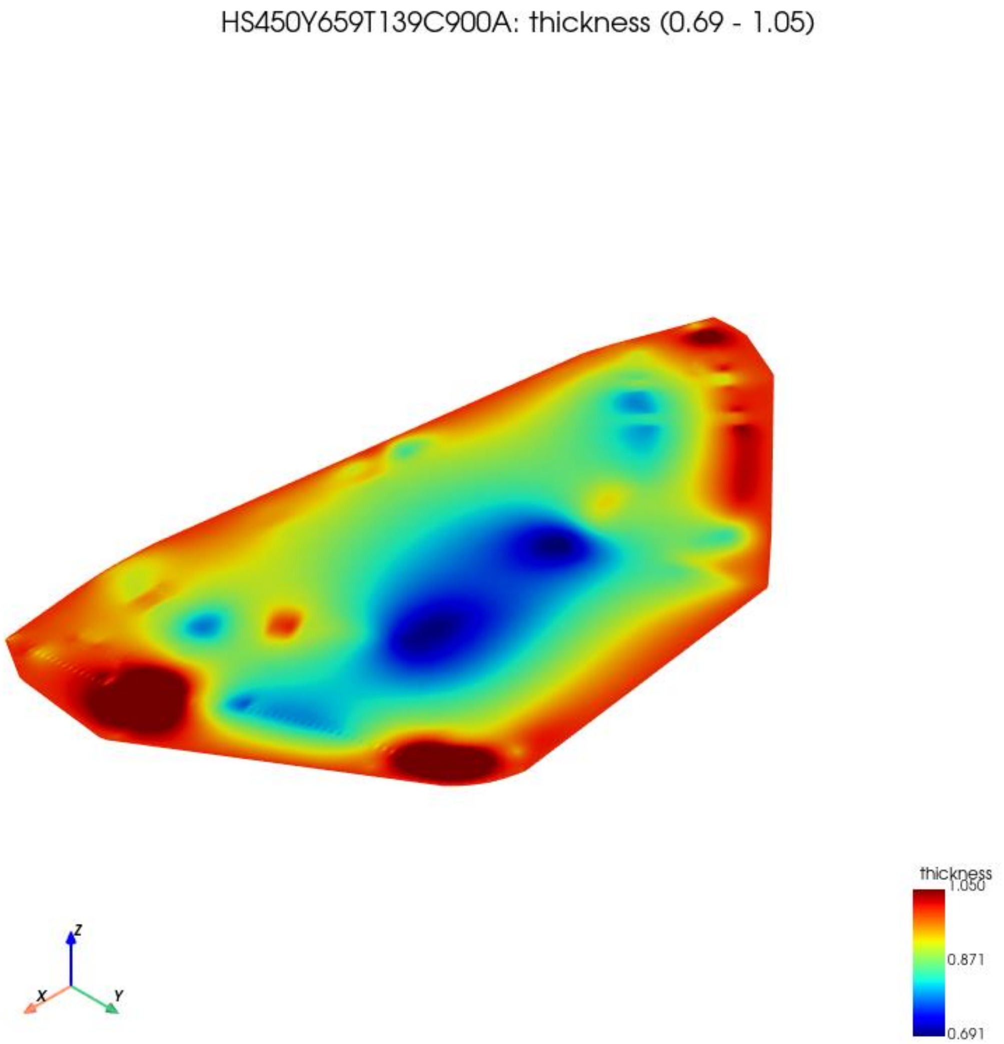
Validation – Yaris – Part 176 - wheelwell inner rear





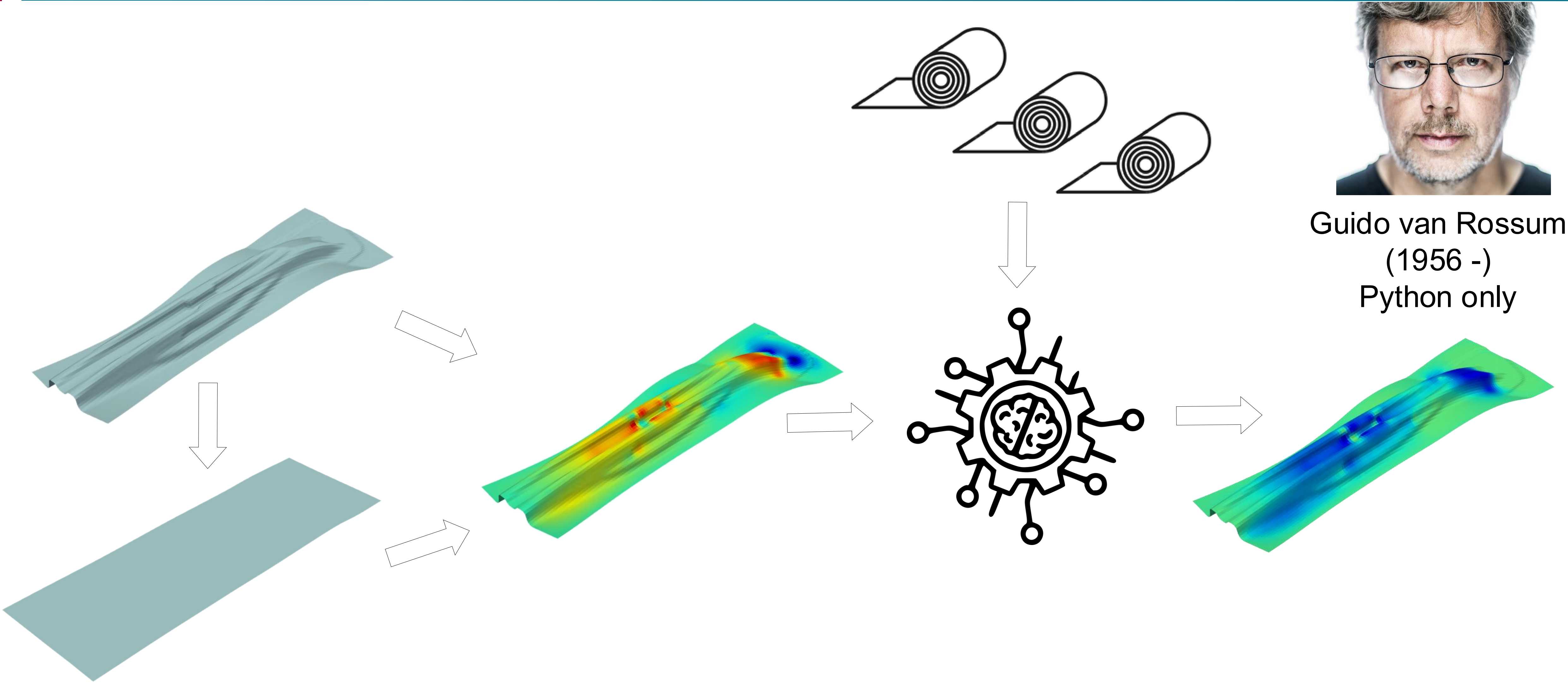
Validation – Yaris – Part 176 - wheelwell inner rear





Conclusion

Conclusion: No FEM, Python only, lightning fast, relatively acurate



Guido van Rossum
(1956 -)
Python only

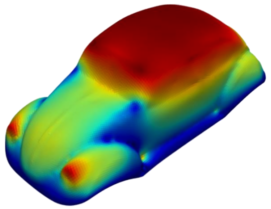
Thank you for your attention

SCALE GmbH - Dresden
Dr.-Ing. Ingolf Lepenies
Pohlandstraße 19
01309 Dresden

ingolf.lepenies@scale.eu
Tel: +49-351-312002-17

divis intelligent solutions GmbH
Dr.-Ing. Peter Krause
Joseph-von-Fraunhofer-Straße 20
44227 Dortmund

krause@divis-gmbh.de
Tel: +49-231-9700-347



IT-Solutions for CAE