

# AI-Based Parameterization of Full Vehicle Models for Crash Simulations

## Project UmMatCraML

Dr. Ingolf Lepenies<sup>1\*</sup>, Dr. Peter Krause<sup>2</sup>,  
Dr. Sebastian Kriechenbauer,<sup>1</sup> Seraphin Lontsi<sup>2</sup>, Thorsten Pohl<sup>3</sup>

1 SCALE GmbH,

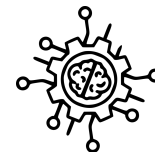
2 divis intelligent solutions GmbH

3 Stellantis OPEL AUTOMOBILE GmbH

16.10.2024

Research question: Can manufacturing- dependent component properties be predicted using machine learning techniques in crash simulations?

© SCALE GmbH – Disclosure to third parties only in consultation with SCALE.

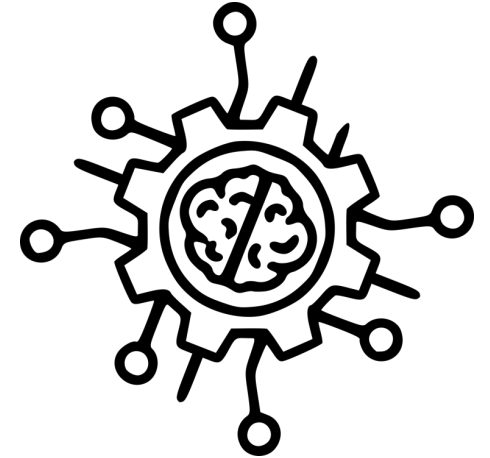


divis SCALE

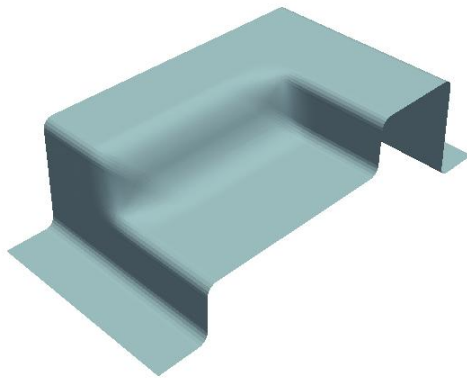
IT-Solutions for CAE

# Contents

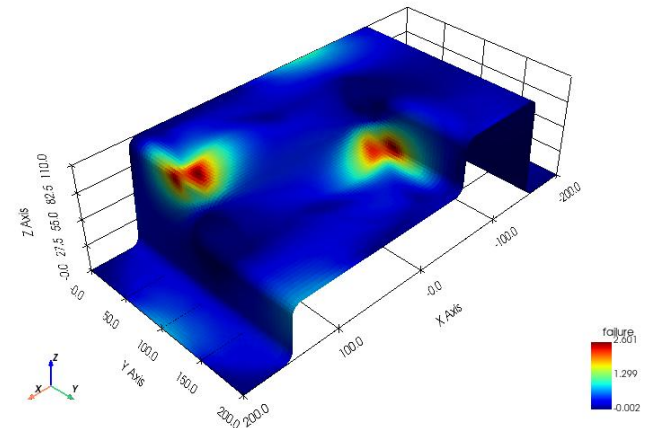
1. UmMatCra-ML Workflow
2. Training data generation and data management
3. ClearVu AI models
4. discussion of challenges and preliminary results
5. summary & outlook



rail0001mat00001

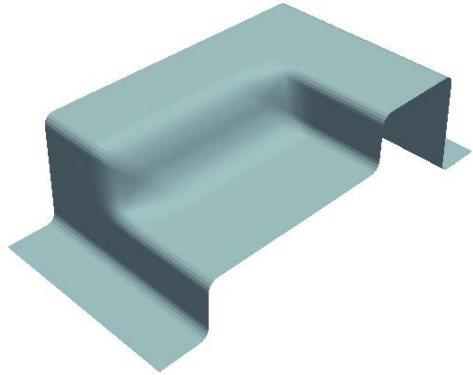


UmMatCrashML

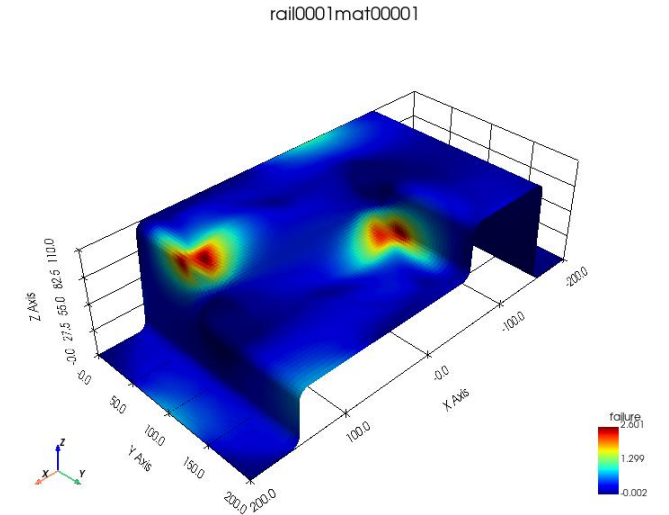


# #1 UmMatCrashML workflow

FEM model of a part



UmMatCrashML



## Input vector

Geometry: Ar\_uvmap

Material: sig\_0

Material: sig\_c

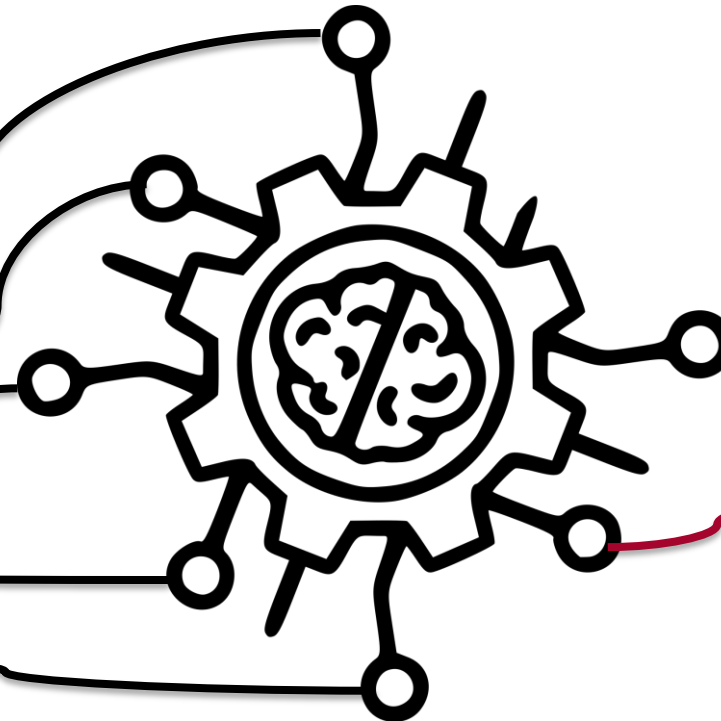
Material: eps\_c

Material: sig\_a

## Output vector

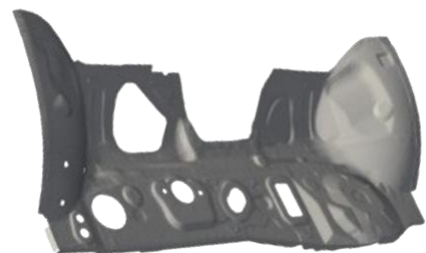
Ar-Isdyna (sheet thickness)

effective plastic strain

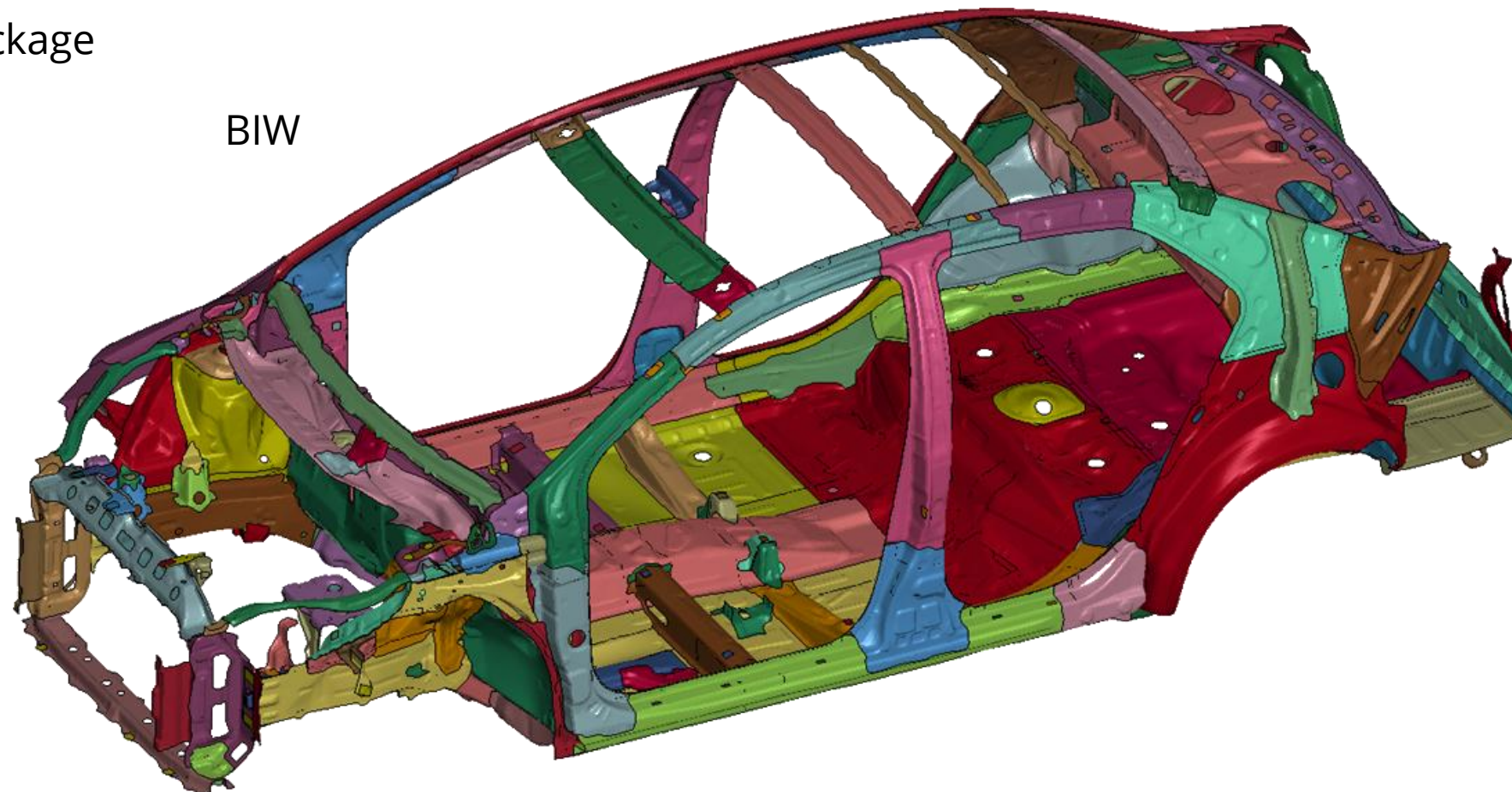


ClearVu KI model

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



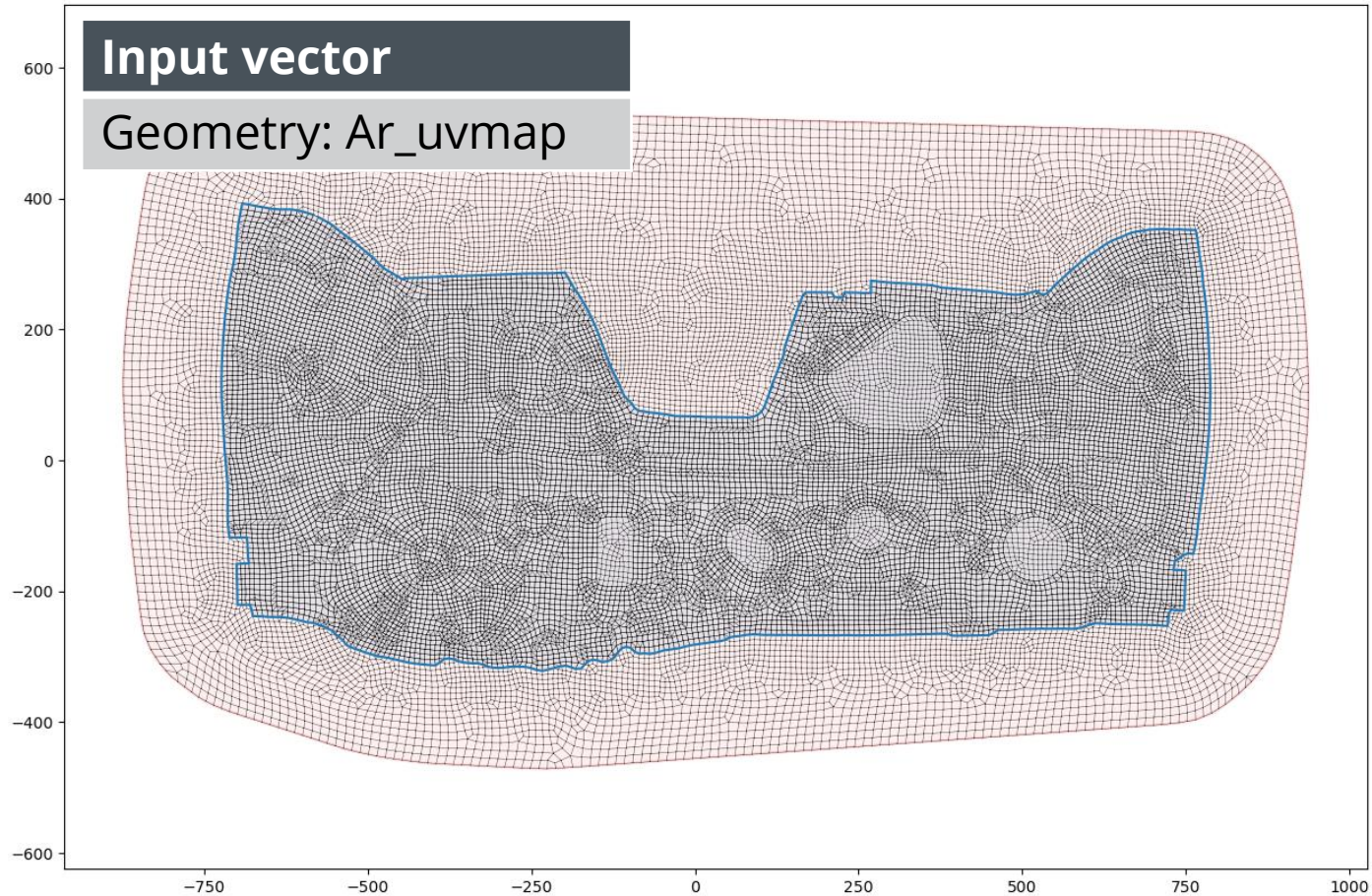
- automatic data upload SCALE.sdm



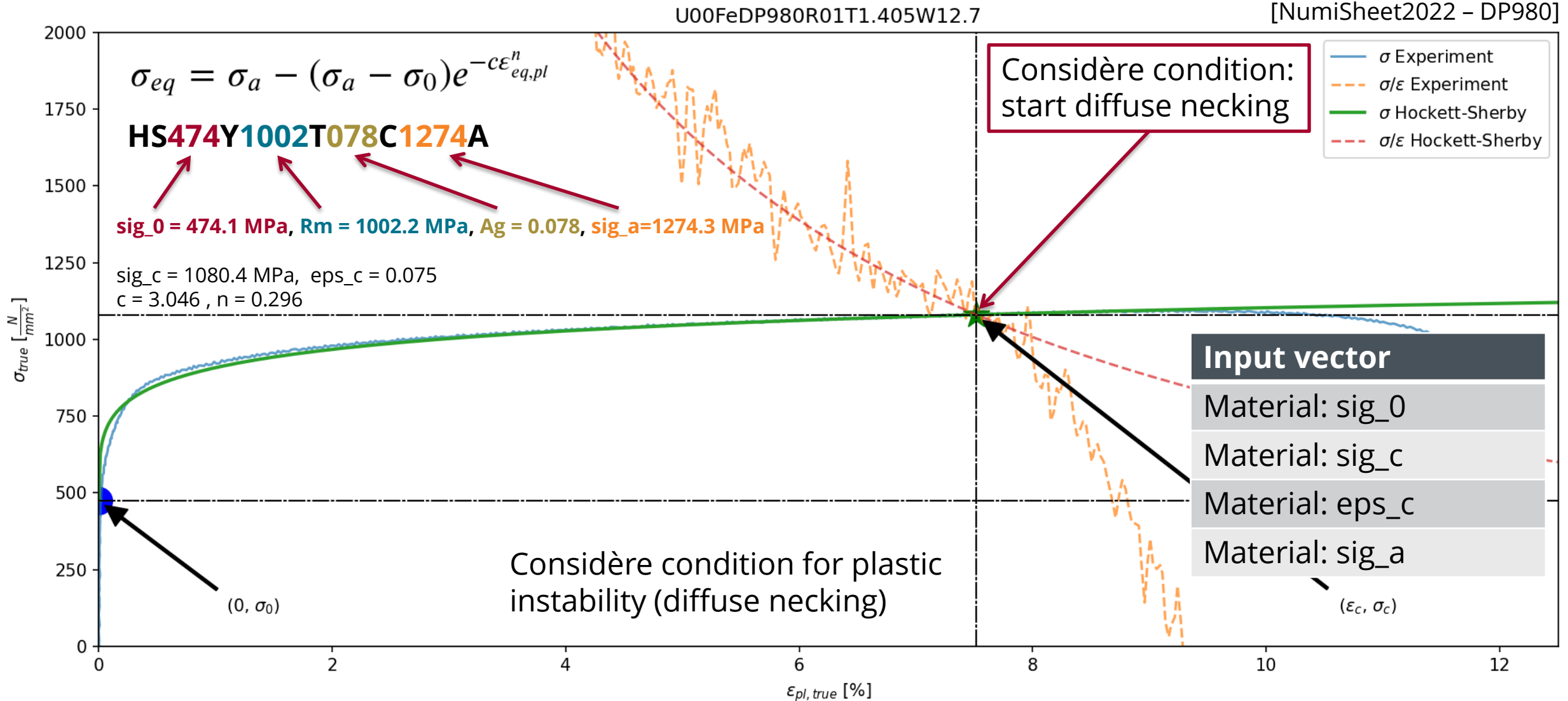
[<https://www.ccsa.gmu.edu/models/>]

# Automatic model creation - Onestepsolver

- Automatic preprocessing with scale.dynapy (Ar\_uvmap)
- Optional extrapolation of the coil blank



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



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



```
1 import pandas
2 from cva import Base
3 from umatcra import Base
4 manager = Base.Manager()
```

```
1 manager.get_input_variable_names()
```

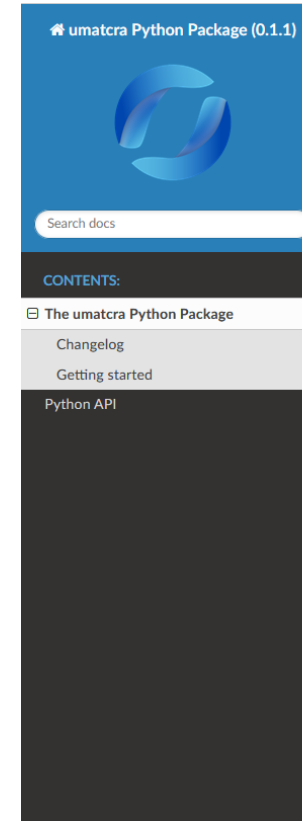
```
{'Ar_Isdyna': ['Ar_uvmap', 'mat_eps_c', 'mat_sig_0', 'mat_sig_a', 'mat_sig_c'],
'epseqpl': ['Ar_uvmap', 'mat_eps_c', 'mat_sig_0', 'mat_sig_a', 'mat_sig_c']}
```

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

	Ar_uvmap	mat_eps_c	mat_sig_0	mat_sig_a	mat_sig_c
0	0.2	0.1	200.0	300	250

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

	Ar_Isdyna	epseqpl
0	1.084302	0.138254



🏠 / 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.

```
1 from umatcra import Base
2
3 manager = Base.Manager()
```

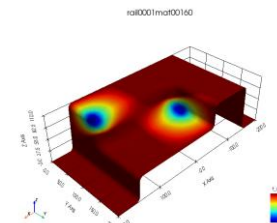
The manager object loads the models for Ar\_Isdyna and epseqpl. You can check the names for the input and output variables with

```
1 manager.get_input_variable_names()
2 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.

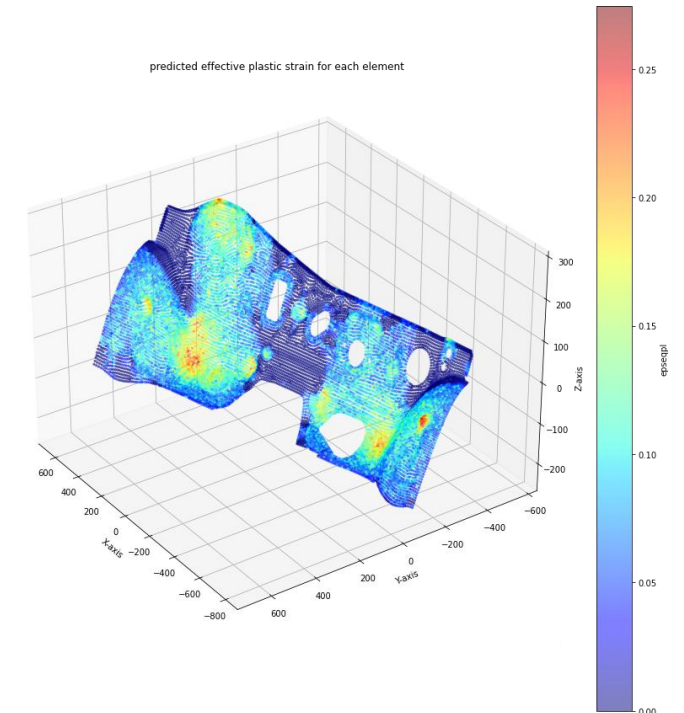
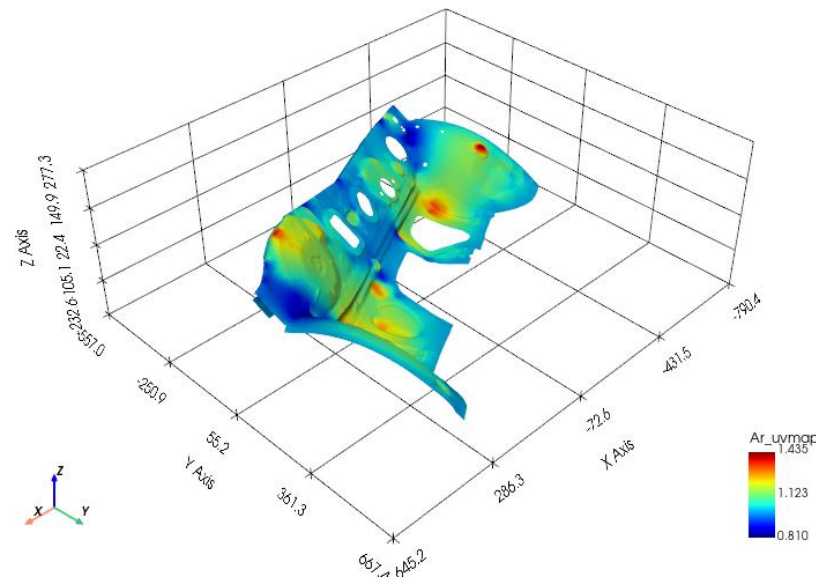
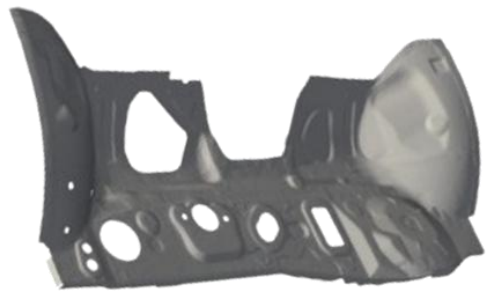
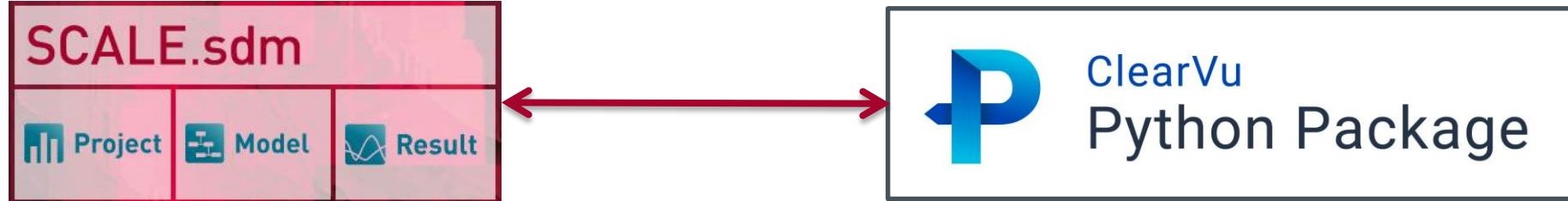
	Ar_uvmap	mat_eps_c	mat_sig_0	mat_sig_a	mat_sig_c
0	1.046333	0.1921093	168.35937	325.78125	228.51562
1	1.417264	0.1601818	648.98071	1324.4262	909.67407
...	...	...	...	...	...

```
1 df = pandas.DataFrame(...)
2
3 manager.predict(df)
```



# UmMatCra-ML Workflow

- Software-Demonstrator: SCALE.sdm with ClearVu



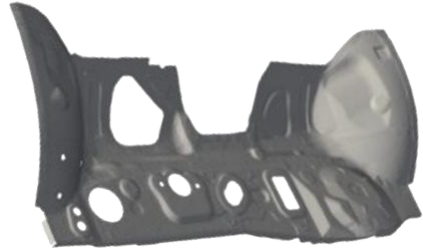


# AI training data

data generation, data management

# data management with SCALE.result

Data storage per part

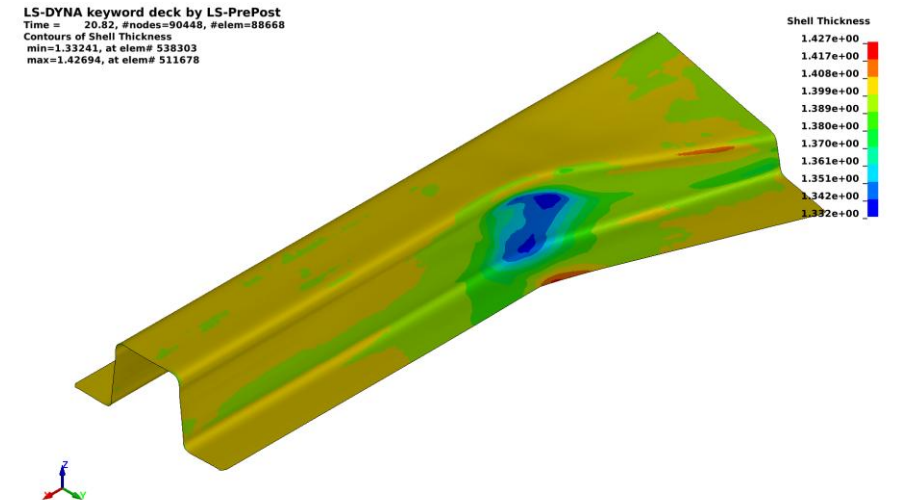
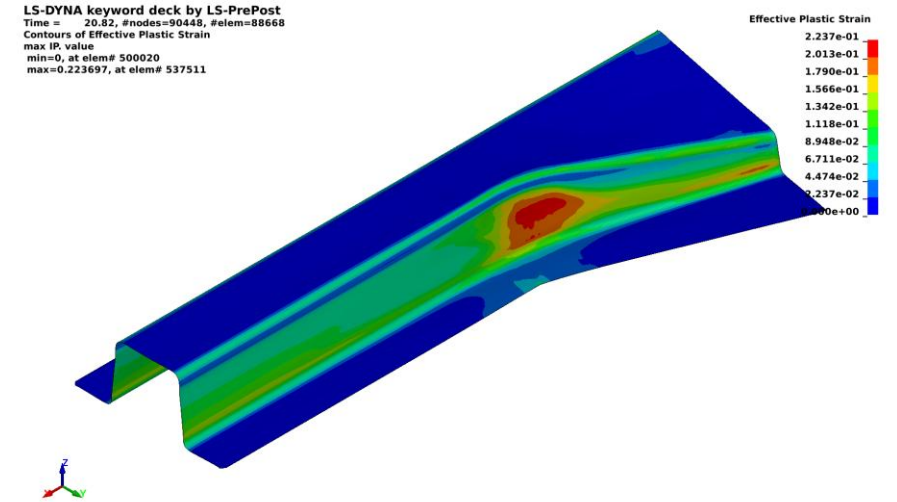
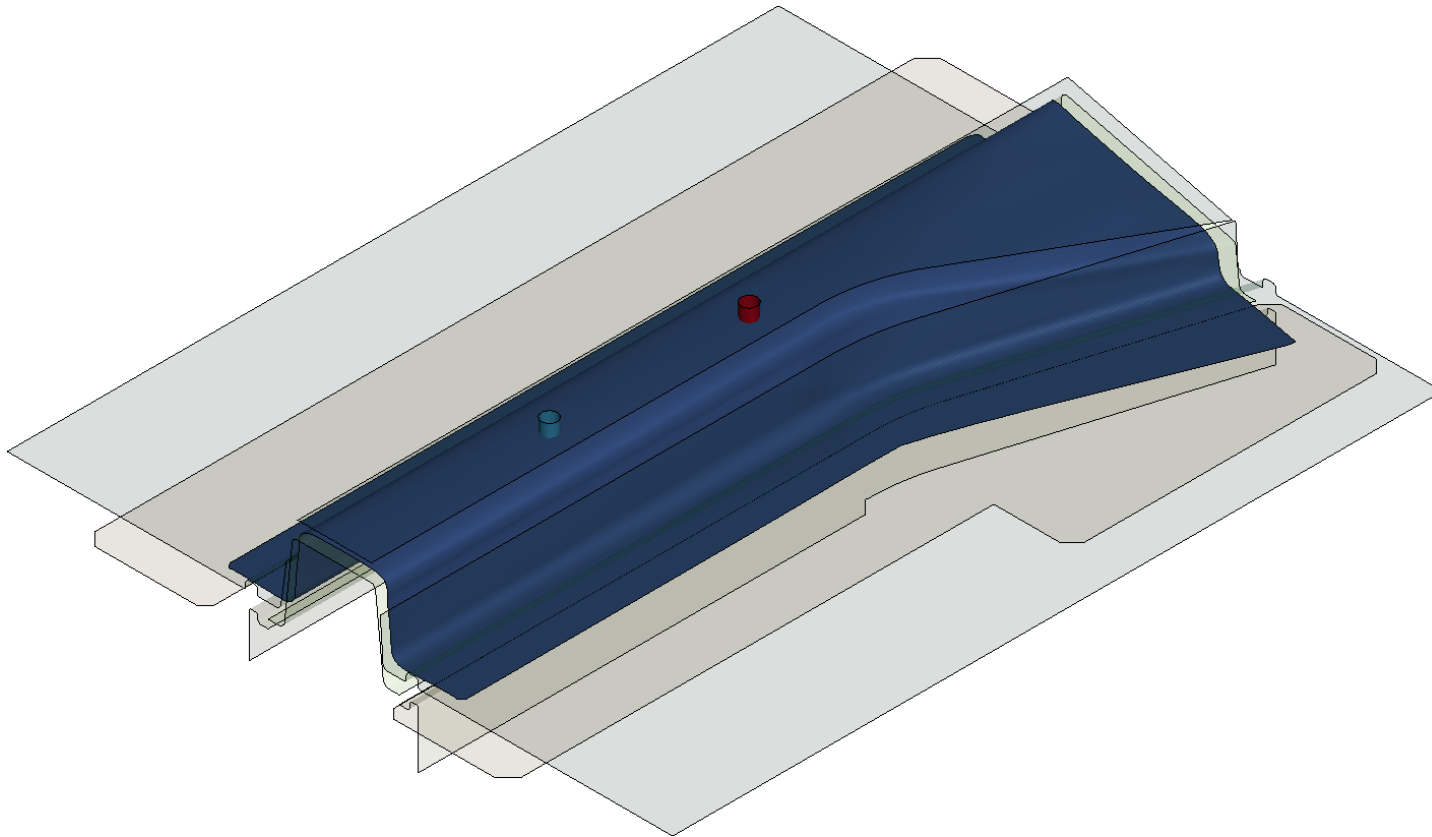


The screenshot displays the SCALE.result web interface. The top navigation bar includes the logo 'SCALE.sdm', a 'Result' dropdown, a 'Confidential information' warning, and user information 'KIProMF'. The main content area is divided into several sections:

- Projects and Collections:** A sidebar on the left shows 'Projects' with 'UmMatCraML' selected and 'Collections' with 'No rows to show'.
- Table:** A central table lists simulation runs with columns for 'Star', 'Run Name', 'Date', 'Name', 'SType', 't1', 'mat1', and 't2'. The table contains 20 rows of data, with the first row highlighted in blue.
- 3D Model:** A 3D model of a car chassis is displayed on the right side of the interface, showing various components in different colors.
- Scenarios:** A 'Scenarios' section at the bottom left shows 'Quickfilter...' and 'All' selected.
- Footer:** The bottom of the interface features a 'Filter' section with a small thumbnail of the car part, and the 'Result' logo on the right.

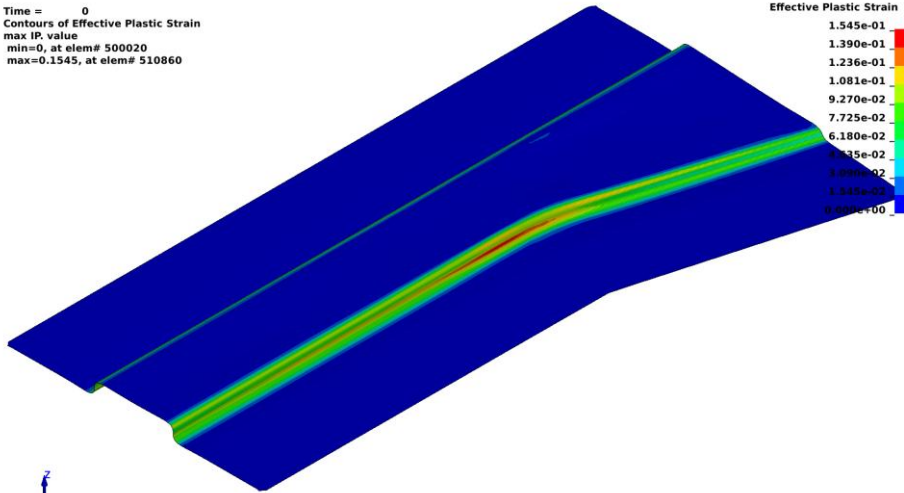
# HFS – parametrisiertes Tiefziehmodell

## Numisheet 2021 – Benchmark 1

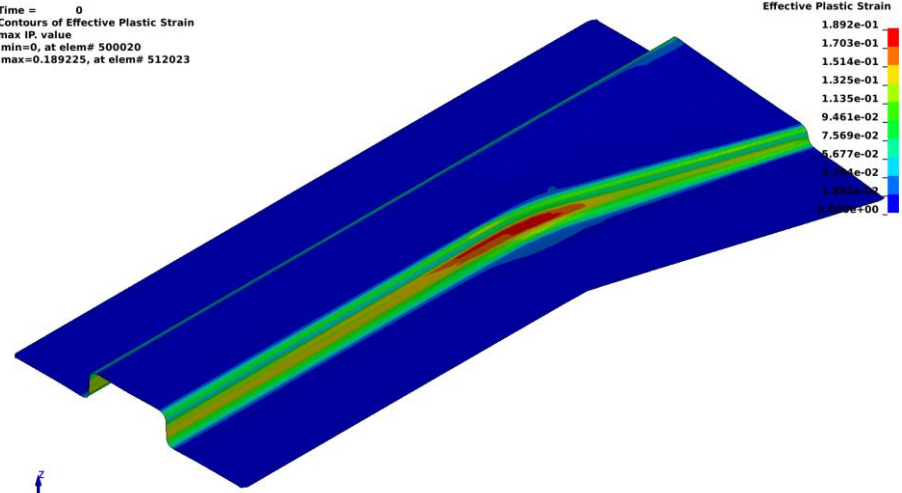


# HFS-Modell #1: Geometrie-Variationen

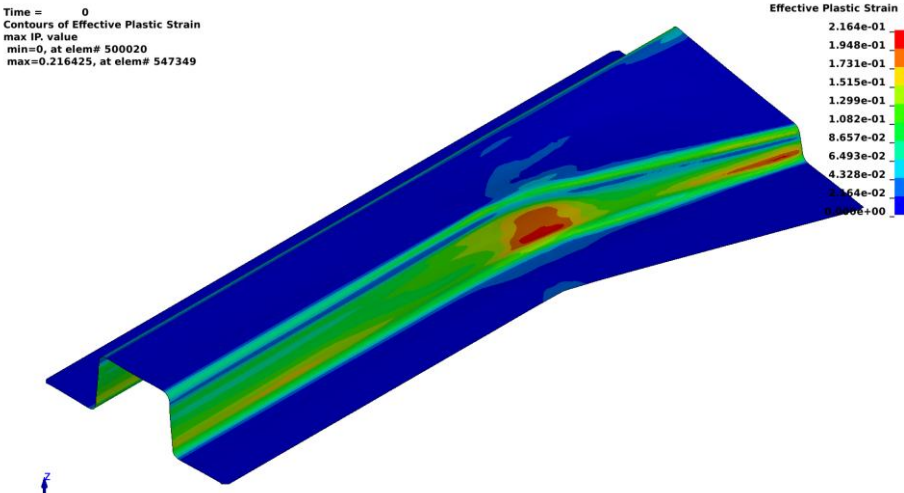
Time = 0  
Contours of Effective Plastic Strain  
max IP, value  
min=0, at elem# 500020  
max=0.1545, at elem# 510860



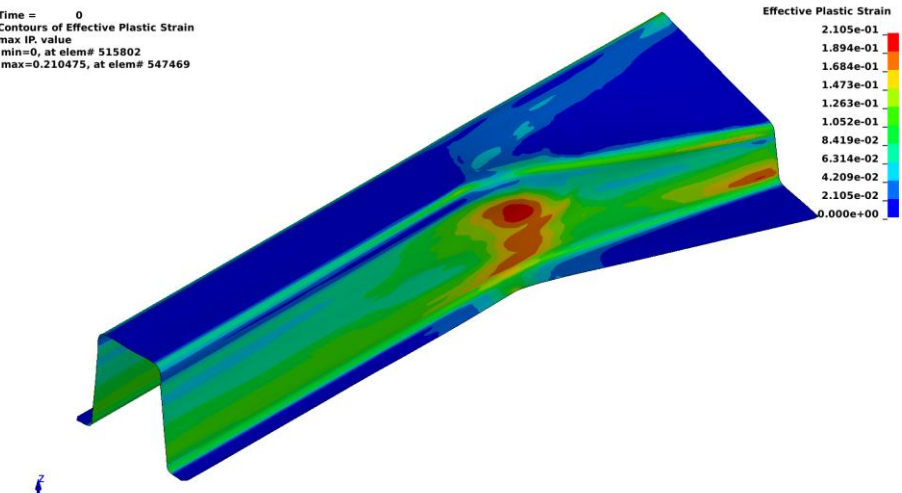
Time = 0  
Contours of Effective Plastic Strain  
max IP, value  
min=0, at elem# 500020  
max=0.189225, at elem# 512023



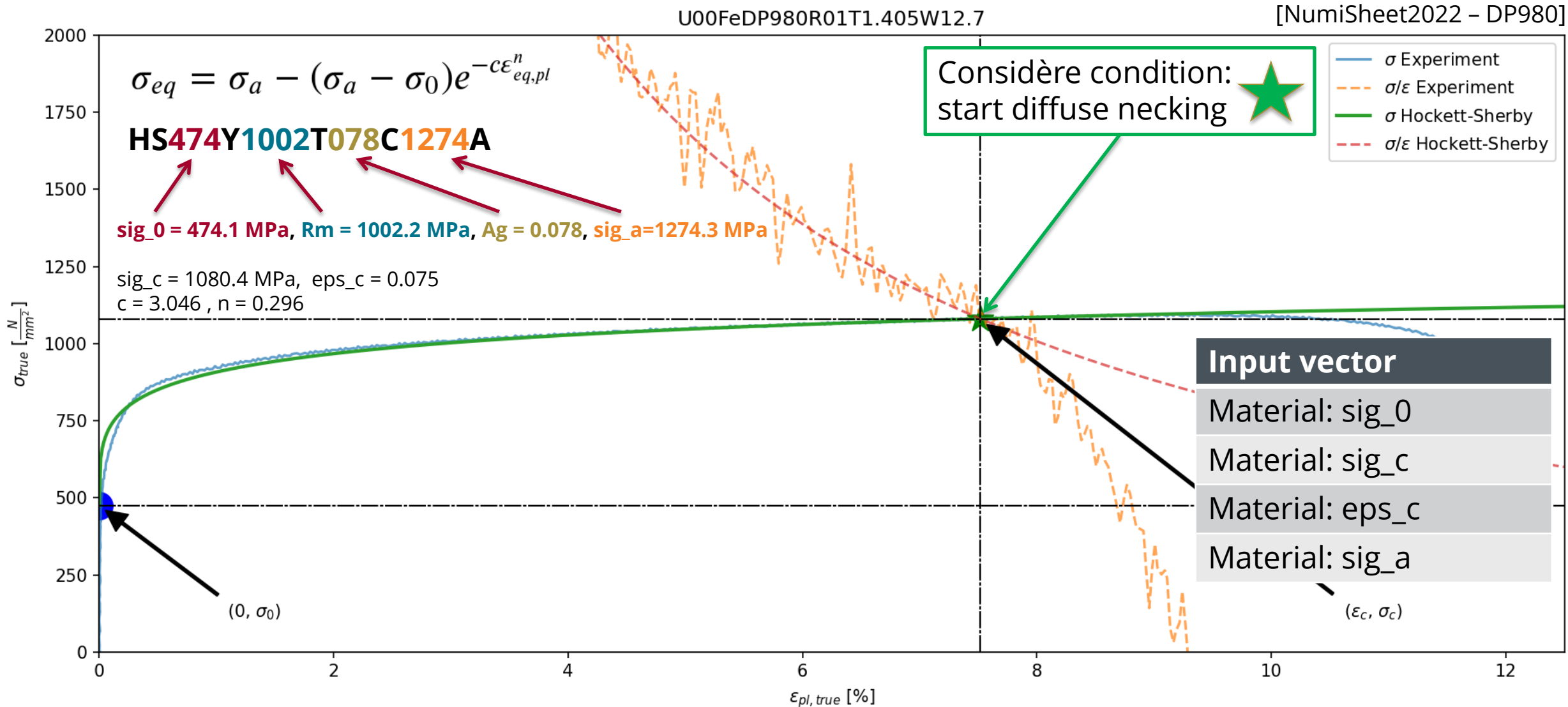
Time = 0  
Contours of Effective Plastic Strain  
max IP, value  
min=0, at elem# 500020  
max=0.216425, at elem# 547349



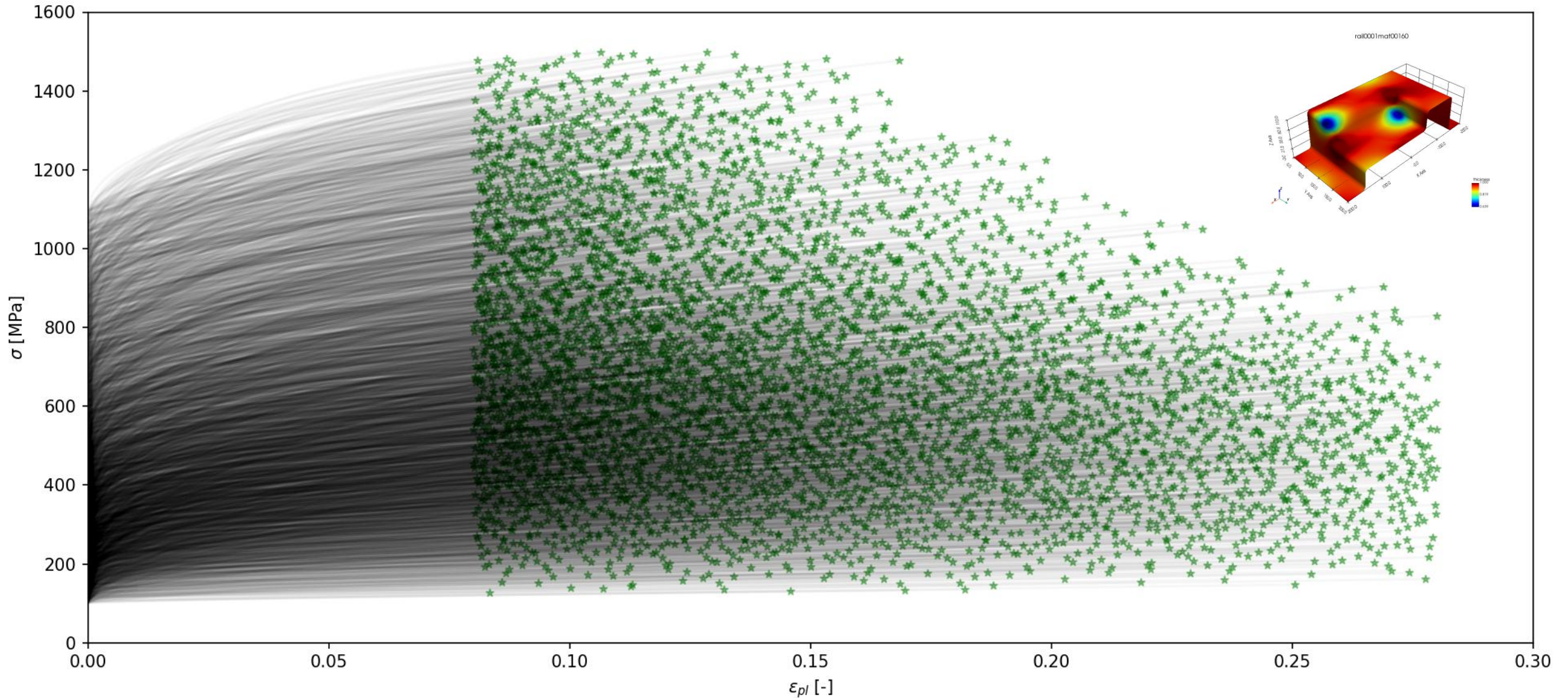
Time = 0  
Contours of Effective Plastic Strain  
max IP, value  
min=0, at elem# 515802  
max=0.210475, at elem# 547469



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

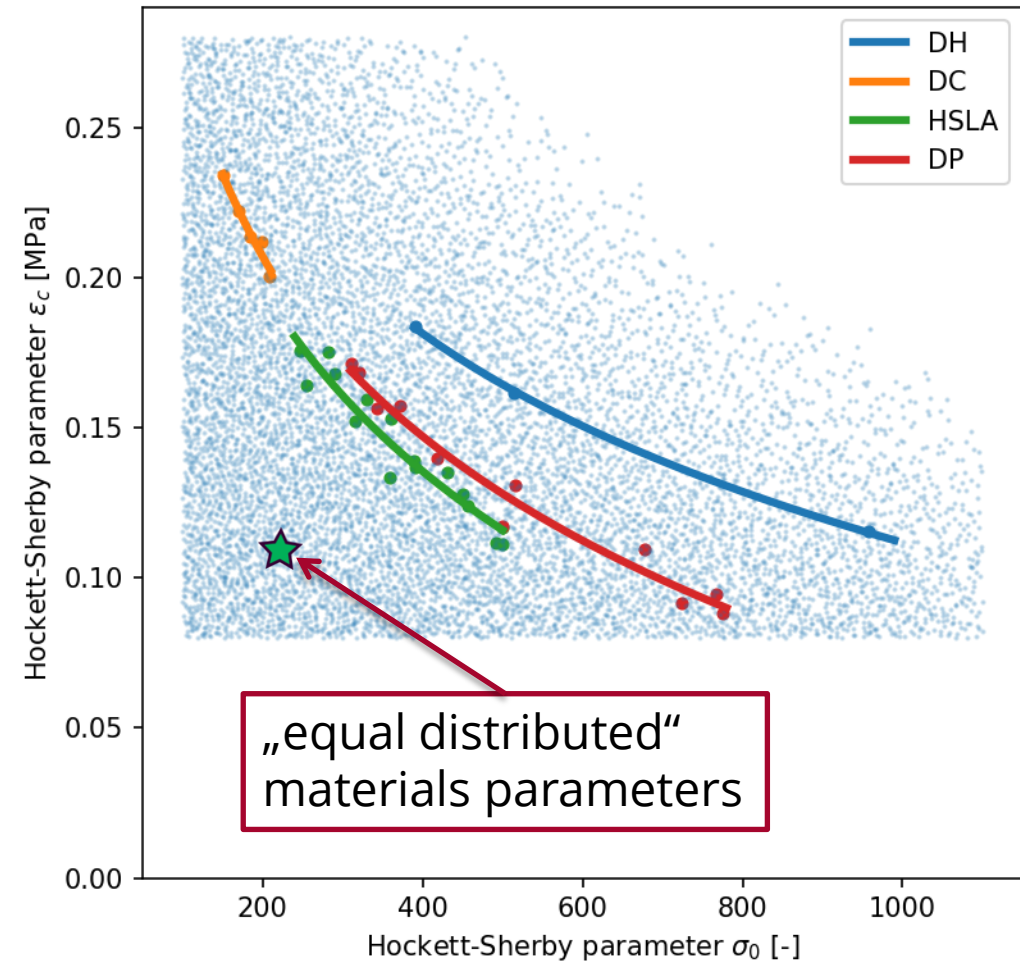
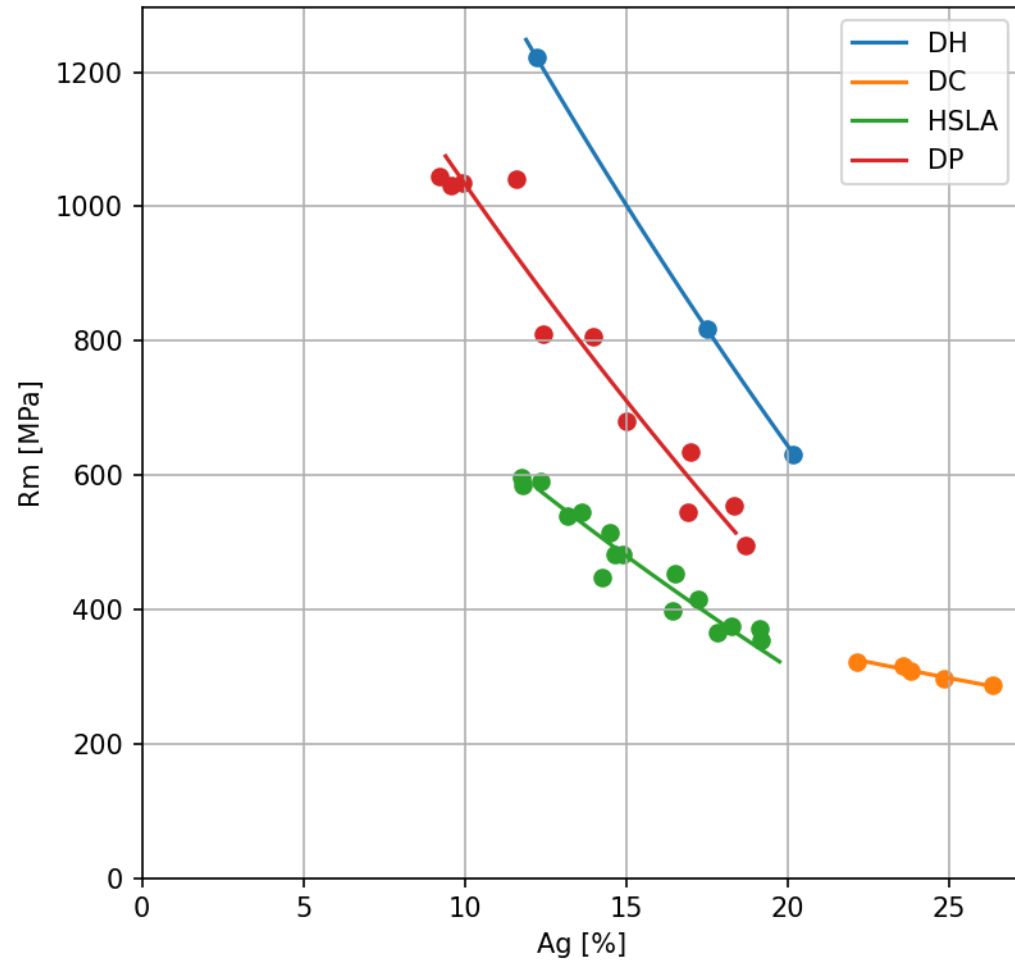


# 5000 generic hardening curves



# AP 1: continuous „real material DB“ vs. continuous „virtual material DB“

Fit of the Hockett-Sherby parameters from real materials tests vs. sampling of virtual material parameters



# ClearVu AI models

accuracy, performance



# Modellierungsaufgabe

## Eingangsseitig

- Bauteil spezifische Parameter
  - Werkstoff  $\varepsilon_c$ ,  $\sigma_0$ ,  $\sigma_a$  und  $\sigma_c$
- Für jedes Flächenelement die Flächendehnung
  - Ar\_uvmap

## Ausgangsseitig

- simulativ ermittelte Flächendehnung und effektive plastische Dehnung
  - Ar\_Isdyna
  - epseqpl

### Input vector

Geometry: Ar\_uvmap

Material: sig\_0

Material: sig\_c

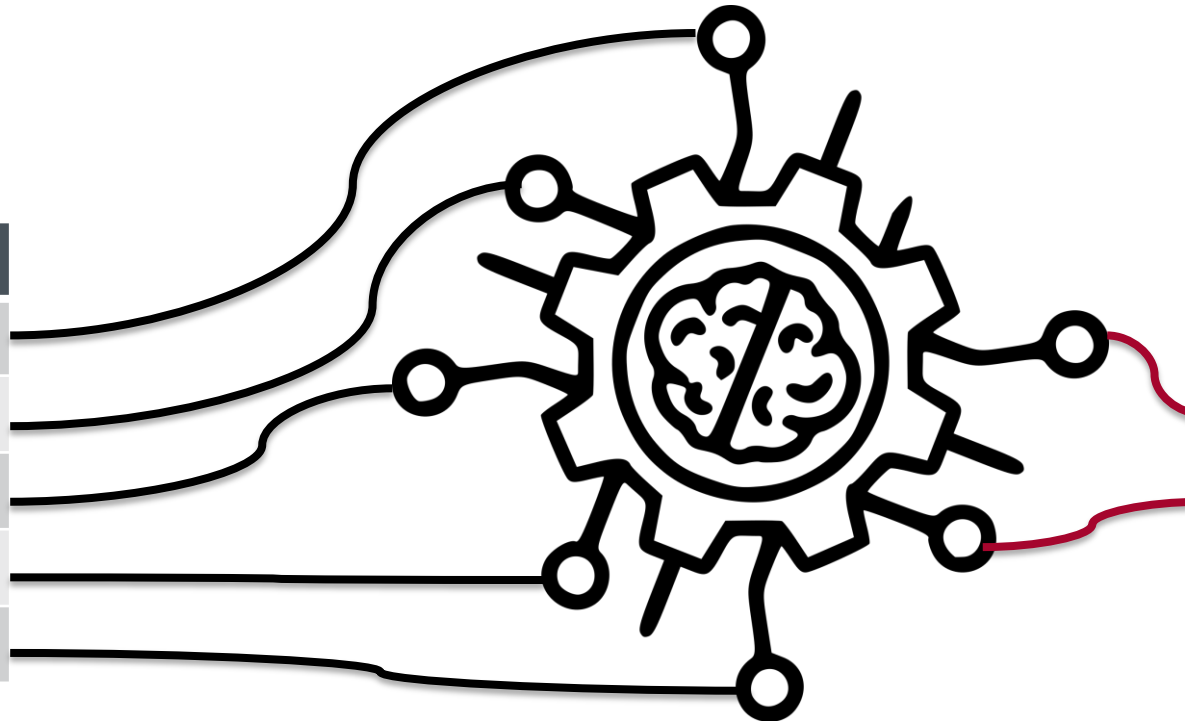
Material: eps\_c

Material: sig\_a

### Output vector

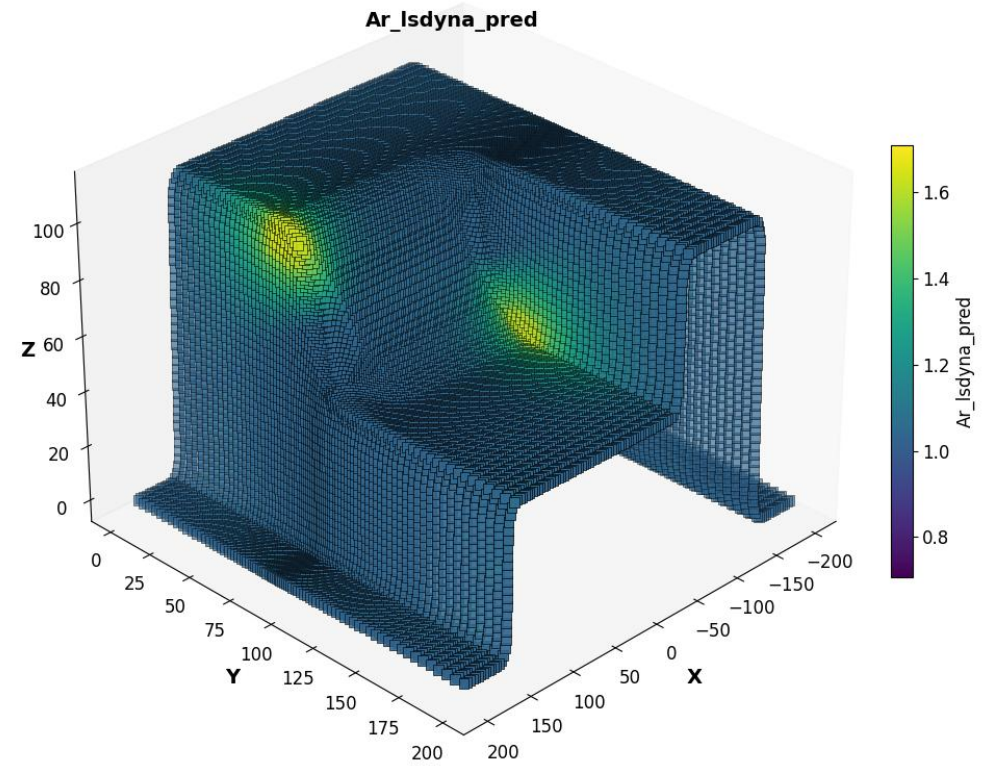
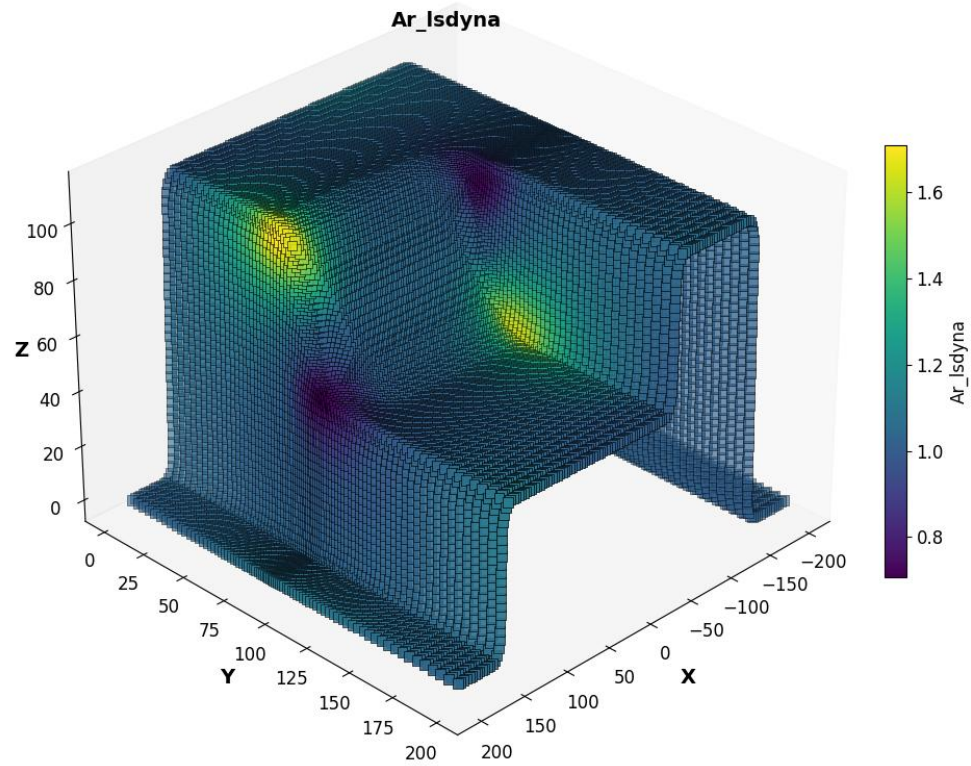
Ar-Isdyna (sheet thickness)

effective plastic strain



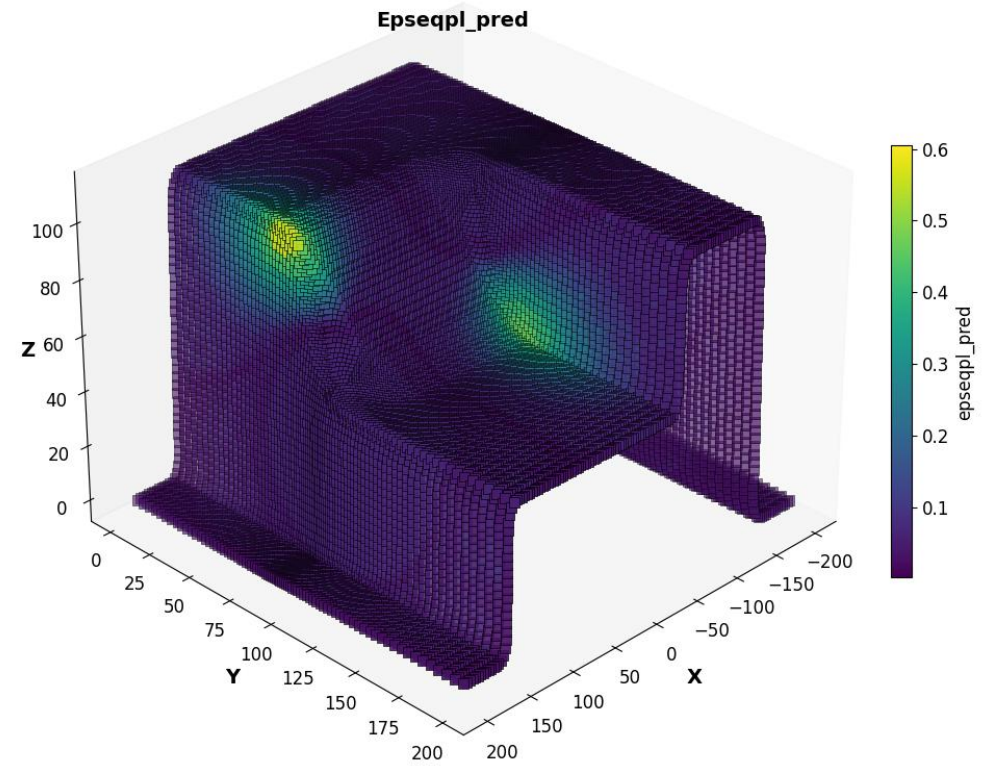
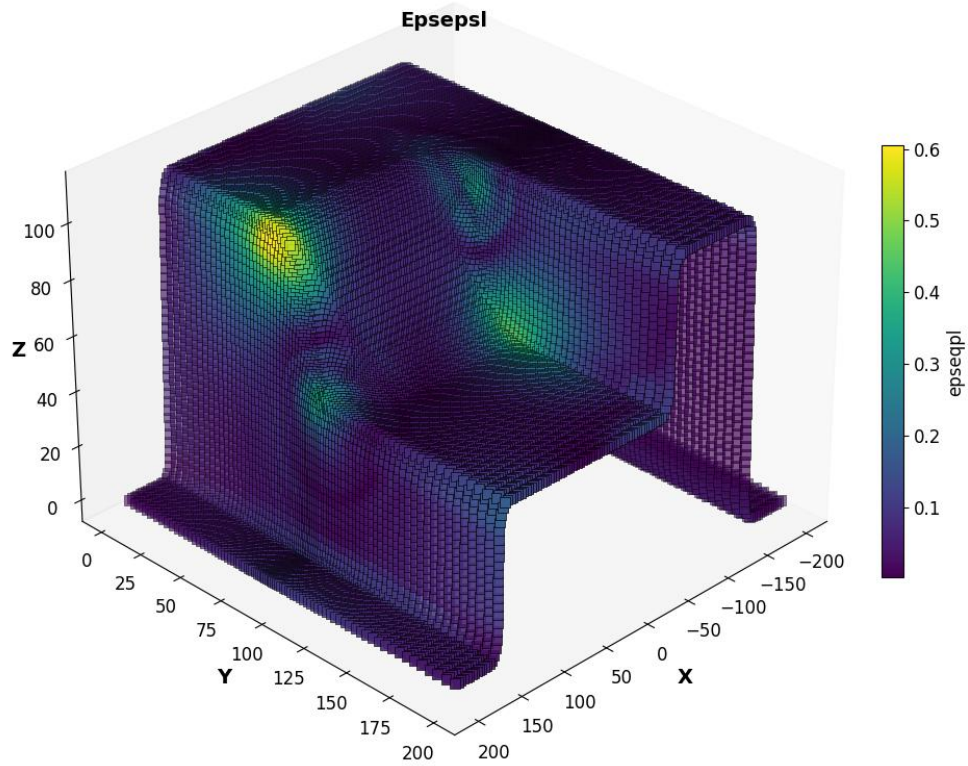
## rail0001mat00296

$$\sigma_0 = 115.38, \sigma_a = 629.2, \sigma_e = 0.13, \sigma_c = 168.7$$



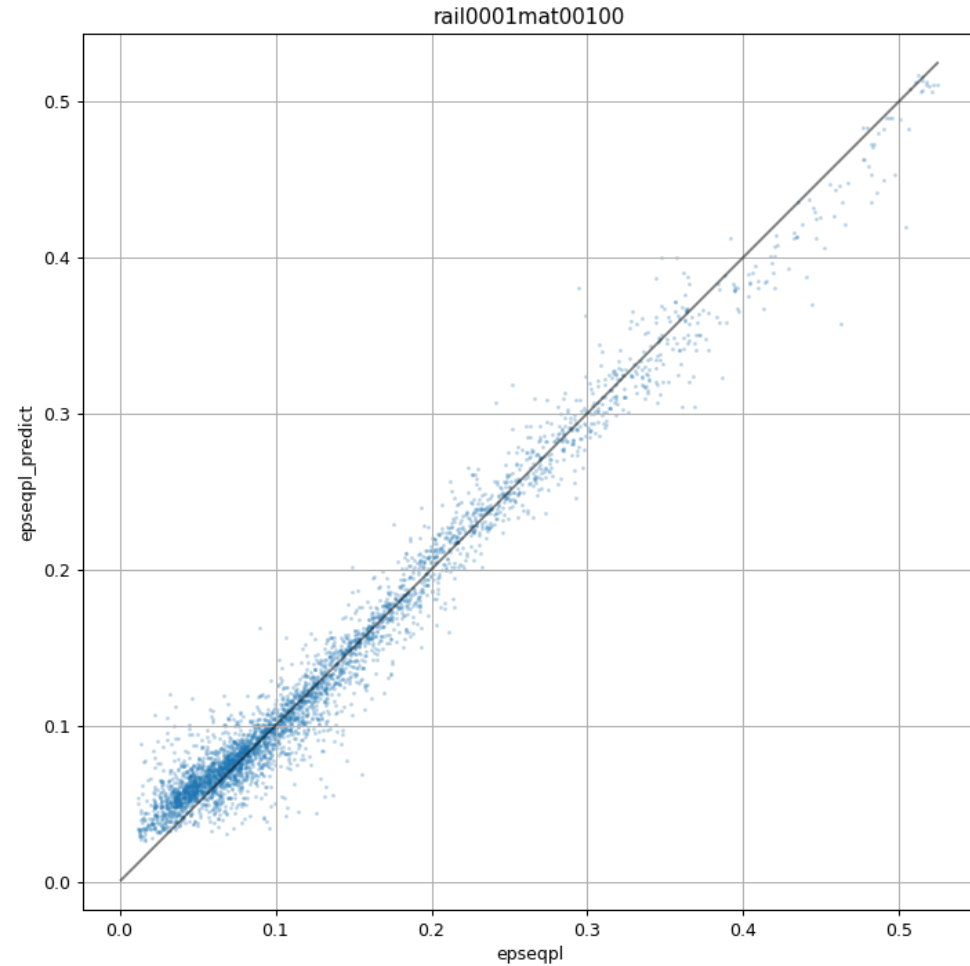
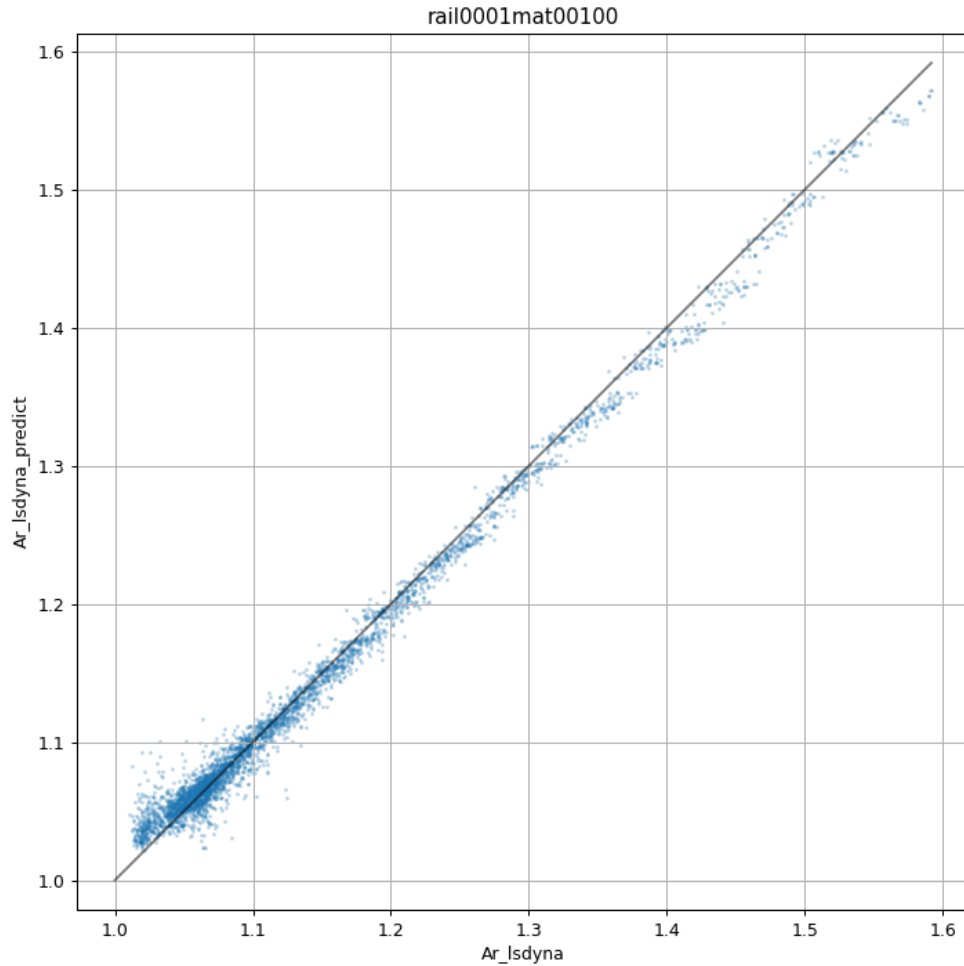
## rail0001mat00296

$$\sigma_0 = 115.38, \sigma_a = 629.2, \sigma_e = 0.13, \sigma_c = 168.7$$



# Accuracy OSS vs. AI & Performance

- prediction time for 10000 Elements  $0.767 \text{ s} \pm 43.3 \text{ ms}$  on Intel i7-8550U CPU @ 1.80GHz

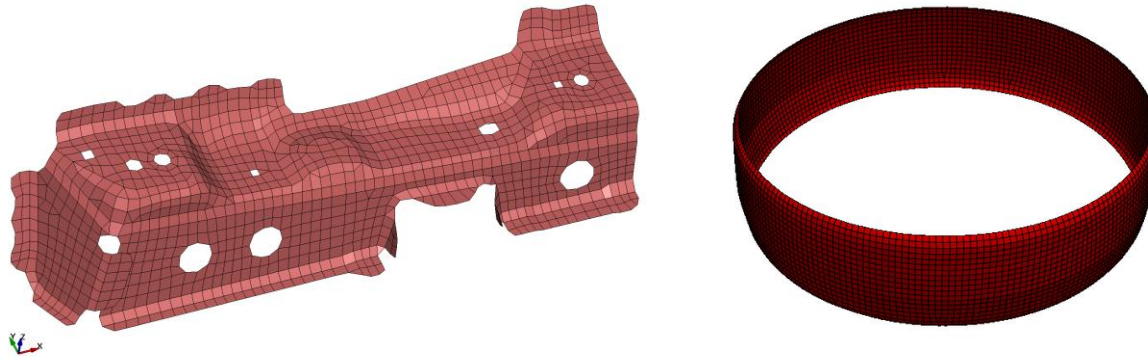


# Challenges Summary Outlook

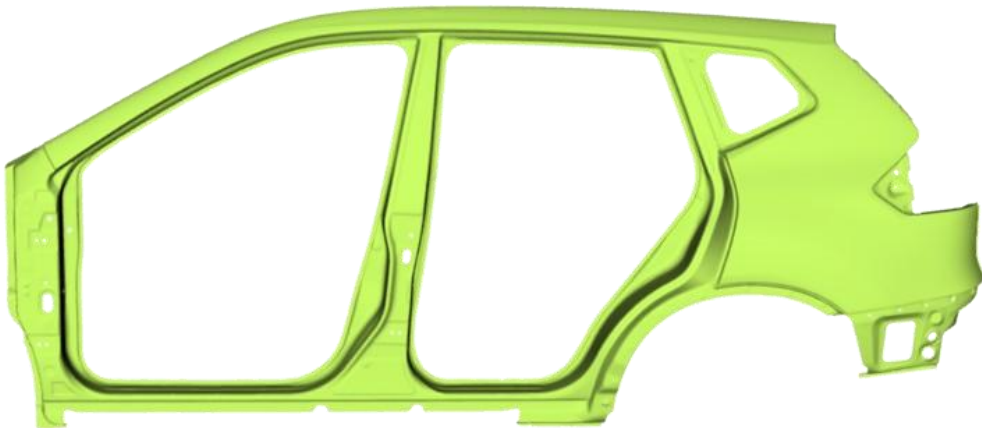
# Diskussion der Herausforderungen

- Mehrstufige Herstellungsprozesse (Biegen nach Zuschnitt) und Rohre

LS-DYNA keyword deck by LS-PrePost

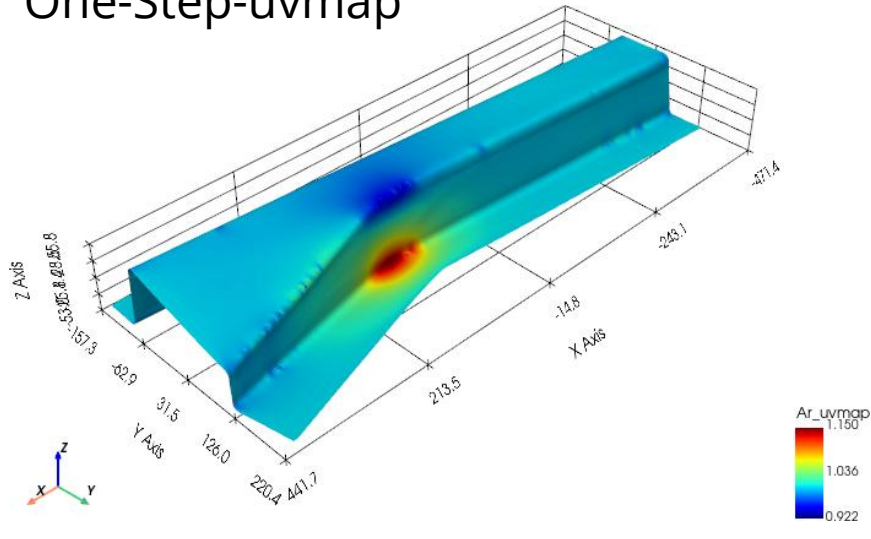


- Zuschnitt mit Loch bei Herstellung (Seitenrahmen)

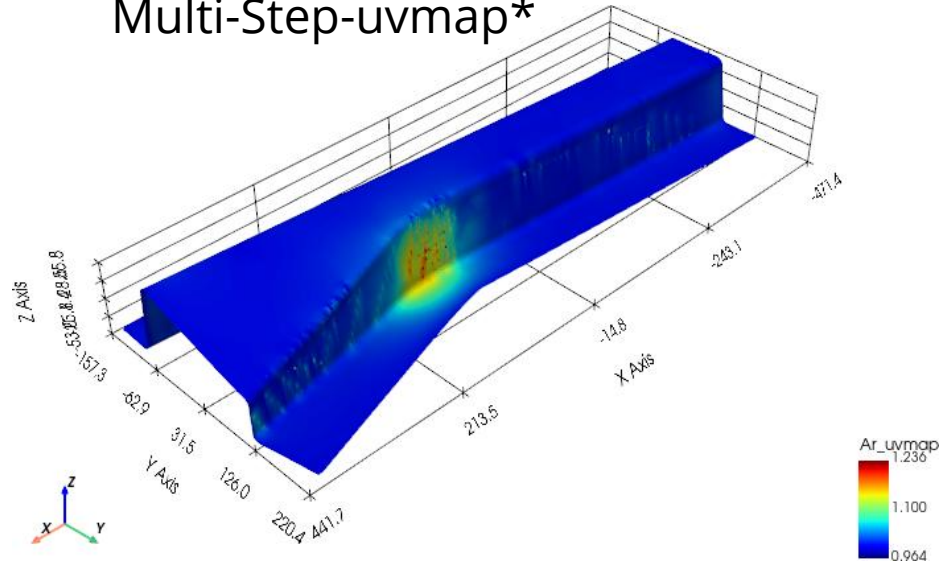


# Types of training data

One-Step-uvmap



Multi-Step-uvmap\*



## Input vector

Geometry: **Ar\_uvmap**

Material: sig\_0

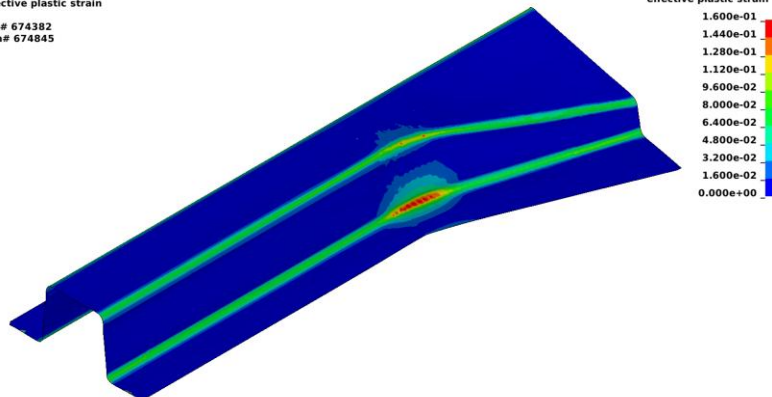
Material: sig\_c

Material: eps\_c

Material: sig\_a

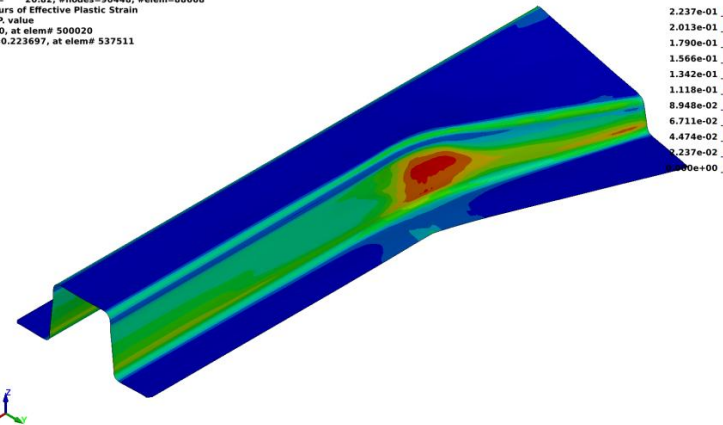
LFS – One-Step

LS-DYNA keyword deck by LS-PrePost  
Contours of effective plastic strain  
max IP value  
min=0, at elem# 674382  
max=1, at elem# 674845



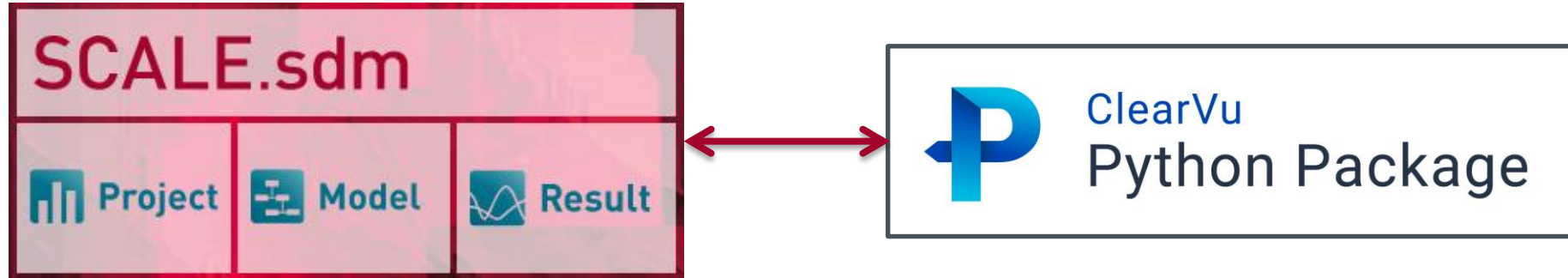
HFS – deep drawing with tools

LS-DYNA keyword deck by LS-PrePost  
Time = 20.82, #nodes=90448, #elem=88668  
Contours of Effective Plastic Strain  
max IP value  
min=0, at elem# 500020  
max=0.223697, at elem# 537511

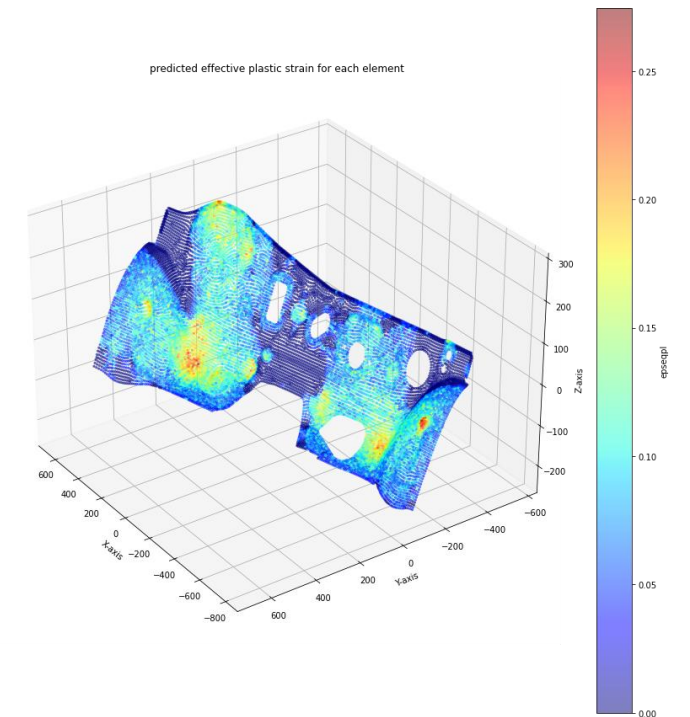
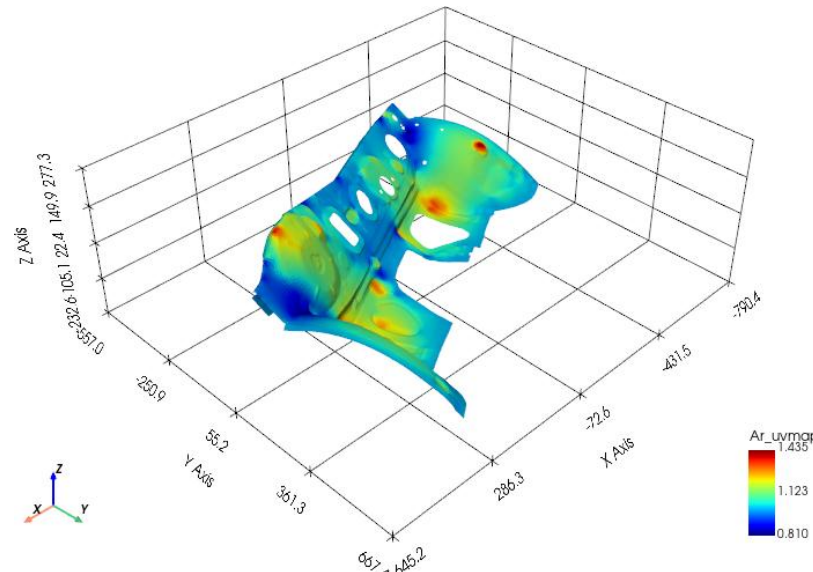
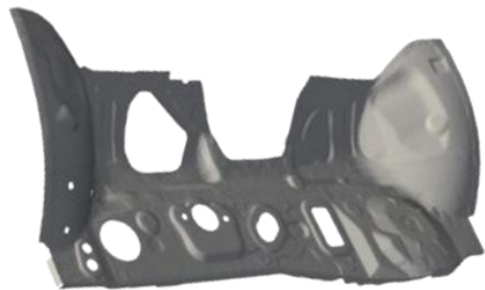


# Summary

- Workflow prototypically modelled with SCALE.sdm and ClearVu



- Proven functionality





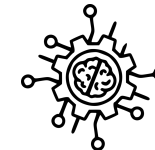
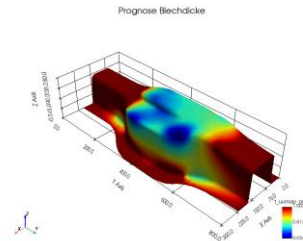
# Thank you for your attention

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



divis SCALE

IT-Solutions for CAE