Sim-a-a-S in the loop
Enhancing Rapid Prototyping of Complex Industrial Machines with Automatic Data Synthesis based on Numerical Simulation of Fluid Flow and Machine Learning
Markus Geveler, Stefan Turek, Tobias Herken – CSAI 2021
01. Introduction
Experiment vs Simulation of complex industrial machines

02. Simulation-as-a-Service
The engine that drives data synthesis

03. AI assistants
Machine Learning-driven assisting of Rapid Prototyping
Introduction

Experiment vs Simulation of complex industrial machines
Introduction
Rapid Prototyping based on simulation

The rapid prototyping loop involves:
1. Start tensor product of simulations
2. Add (batch of) process data (sets)
3. Analyse simulation results
4. Generate (batch of) geometric prototypes

This cycle is repeated to refine the prototypes.
Digital Twin

Frontend and data layer

- Analyse simulation results
- Add (batch of) geometric prototypes
- Start tensor product of simulations
- Rapid prototyping loop

(batch of) process data (sets)
Simulation batch

The simulation data tensor

Analyse simulation results

(batch of) geometric prototypes

rapid prototyping loop

Start tensor product of simulations

Add (batch of) process data (sets)
Results
The result data tensor
Simulation-as-a-Service

The engine that drives data synthesis: Sim-a-a-S := I-a-a-S + S-a-a-S

Simulation-as-a-Service
The engine that drives data synthesis

- Analyse simulation results
- (Batch of) geometric prototypes
- Start tensor product of simulations
- Add (batch of) process data (sets)

Sim-a-a-S loop

- Grid computing job and cluster management
- Automatic geometry processing jobs
- Numerical simulation jobs

Rapid prototyping loop

- Result data processing jobs

Datacenter/cluster i

Simulation queue

Simulation j

Cluster jobs for sim j
- Geometry processing
- Numerical simulation
- Result processing
- Cleanup
Digital Twin

Geometric layer: Automatic geometry processing 1

- Analyse simulation results
- Rapid prototyping loop
- Start tensor product of simulations
- (Batch of) geometric prototypes
- Add (Batch of) process data (sets)
- Grid computing job and cluster management
- Sim-a-a-S loop
- Automatic geometry processing jobs
- Result data processing jobs
- Numerical simulation jobs
- Grid computing job and cluster management
Digital Twin

Geometric layer: Automatic geometry processing II

- Analyse simulation results
- (Batch of) geometric prototypes
- Add (batch of) process data (sets)
- Start tensor product of simulations
- Rapid prototyping loop

Sim-a-a-S loop

- Grid computing job and cluster management
- Automatic geometry processing jobs
- Result data processing jobs
- Numerical simulation jobs

Rapid prototyping loop

Digital Twin

Geometric layer: Automatic geometry processing II
Sim-a-a-S

Numerical simulation of fluid flow: the behavioural layer of the Digital Twin

Solve NSEs efficiently

\[ u_t - \nu \Delta u + u \cdot \nabla u + \nabla p = f, \]
\[ -\nabla \cdot u = 0 \]

- recursive domain decomposition
- Newton-Krylov-\textit{multigrid schemes}
- local geometric multigrid highly hardware-optimised and accelerated with GPUs or similar
- smoother determines overall efficiency
Numerical simulation of fluid flow: the behavioural layer of the Digital Twin: FEAT family

**Weak form**
\[
\int_{\Omega} \partial_t u \, v \, dx + \int_{\Omega} (u \cdot \nabla u) \, v \, dx - \nu \int_{\Omega} \Delta u \, v \, dx + \int_{\Omega} \nabla p \, v \, dx = \int_{\Omega} f \, v \, dx
\]
\[
\int_{\Omega} (\nabla \cdot u) \, q \, dx = 0
\]

**FEM**
\[
(\partial_t u_h, v_h) + (u_h \cdot \nabla u_h, v_h) + \nu (\nabla u_h, \nabla v_h) - (p_h, \nabla \cdot v_h) = (f, v_h)
\]
\[-(q_h, \nabla \cdot u_h) = 0
\]

**Matrix form**
\[
\begin{bmatrix}
M & 0 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
\partial_t u \\
0
\end{bmatrix}
+ \begin{bmatrix}
K(u) + \nu L & B \\
-B^T & 0
\end{bmatrix}
\begin{bmatrix}
u \\
p
\end{bmatrix} = \begin{bmatrix} F \\
0
\end{bmatrix}
\]

\[
S(u) = \alpha M + \theta \Delta t (K(u) + \nu L)
\]

**Theta scheme**
\[
g = [M - (1 - \theta) \Delta t (K(u) + \nu L)] \cdot u^n + \theta \Delta t F^{n+1} + (1 - \theta) \Delta t F^n
\]

**Decouple**
\[
S(\tilde{u}) \cdot \tilde{u} = f_{FP}
\]
\[
P \cdot q = \frac{1}{\Delta t} B^T \cdot u
\]

- Pressure Poisson Problem Consumes most of the time
- Recursive Domain Decomposition Newton-Krylov-\textit{Multigrid schemes}
- Local geometric \textit{multigrid} highly hardware-optimised and accelerated with GPUs or similar
- \textit{Smoothers} determines overall efficiency
Sim-a-a-S
Data visualisation and -derivatives

**Sim-a-a-S loop**

- **Result data processing jobs**
  - Numerical simulation jobs
  - Automatic geometry processing jobs
  - Grid computing job and cluster management

- **Rapid prototyping loop**
  - Analyse simulation results
  - Start tensor product of simulations

- **Add (batch of) process data (sets)**
- **(Batch of) geometric prototypes**

- **Sim-a-a-S loop**
Sim-a-a-S
System Architecture

HTTPS
REST API

FORGE
GitHub

WebGL
ParaView

HLRS
Jülich

JURECA
AI-based assistance systems

Sim-a-a-S increases simulation throughput and allows for fast data synthesis + standardized data

Digital Twin

operation point (process- / material data)

CFD

simulation result

Sim-a-a-S
AI-based assistance systems

Connected to Sim-a-a-S

- Analyse simulation results
- Add (batch of) geometric prototypes
- Start tensor product of simulations
- Rapid prototyping loop
- (Batch of) process data (sets)
- Grid computing job and cluster management
- Automatic geometry processing jobs
- Numerical simulation jobs
- Result data processing jobs
- Sim-a-a-S loop
- Rapid prototyping loop

- Automatic data processing
- Asynchronous training of neural nets
- AI assistant loop
- Ready to use microservice for target function regression
- Automatic deployment of new models
AI-based assistance systems

Connected to Sim-a-a-S

readily trained neural net = instant response model for simple answer
**AI-based assistance systems**

Connected to Sim-a-a-S

- **Digital Twin (parametric)**
- **operation point**

Output:

- **max temperature gradient**
- **min washing time**
- **pressure estimate in axial direction**, „go!“, ...

Simple target function value(s) e.g. scalar quality-metrics, go/nogo, simple discrete representations

- **max temperature gradient**
- **min washing time**
- **pressure estimate in axial direction**

Test loss 0.145
Training loss 0.130
AI-based assistance systems

Connected to Sim-a-a-S

AI assistant loop

- automatic data processing
- asynchronous training of neural nets
- ready to use microservice for target function regression
- automatic deployment of new models

Cluster jobs for sim j
- geometry processing
- numerical simulation
- result processing
- send data to undeployed new model at Sim-a-a-S system
- cleanup
AI-based assistance systems
Connected to Sim-a-a-S

1) incremental training at the Sim-a-a-S gateway system
2) re-training starting at data saturation point or nightly
3) readily trained models generate pull request
4) new models go to new Sim-a-a-S system version at update time (2-weekly)
Results

First alpha assistants predict simple indicators at high accuracy.

Automatic Simulation ready for
- Extrusion Dies
- Single Screw Extruders
- Hotrunner systems
- Excenter screw pumps
- Twin Screw Extruders BETA
- Centrifugal pumps BETA
- Blow Mold

AI Assistants in ALPHA for
- Extrusion Dies
- Single Screw Extruders
- Blow Mold

AI Assistants in BETA for
- Extrusion Dies
- Single Screw Extruders
- Blow Mold
neural nets are used for real time forecast of probable target function values for assisting pre-selection

analysing simulation results
(batch of) geometric prototypes

rapid prototyping loop

start tensor product of simulations
add (batch of) process data (sets)

produces large batches of simulations based on variations of a digital twin

grid computing job and cluster management
automatic geometry processing jobs
Sim-a-a-S loop
result data processing jobs
numerical simulation jobs

produces large amounts of rigorous high end simulation results

automatic data processing
asynchronous training of neural nets

ready to use microservice for target function regression
automatic deployment of new models

produces models for the target functions based on neural nets that are trained with data derivatives from the synthetic results
Outlook

Construction parameter proposal
Thank you

https://ianus-simulation.de/en/stroemungsraum/