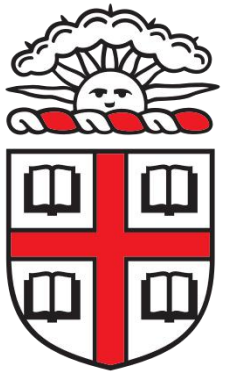


BROWN

Lecture 17
Streamlining Your Data-driven Process with `f3dasm`

M.P. van der Schelling

Delft University of Technology, the Netherlands

Martin van der Schelling (M.P.vanderSchelling@tudelft.nl)

Doctoral Researcher in Materials Science and Engineering

Data-Driven Design & Analysis of Structures and Materials

Fall 2024 ENGN2350

October 21st, 2024

Motivation

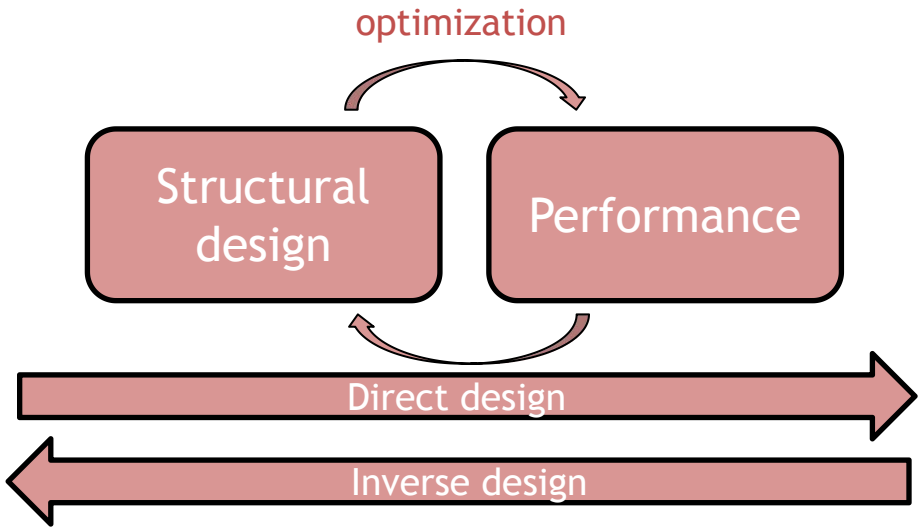
- Continuous search for the most efficient design
- High-dimensional solution space
- Impractical to conduct experimental investigations
- Challenge to accelerate the optimization process
- Inverse design approach

Data-driven methods leveraging machine learning

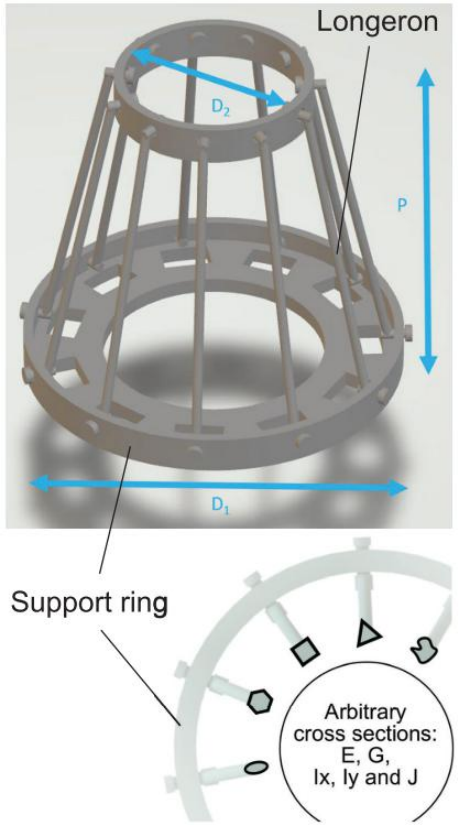
Fragile becomes supercompressible

Learning to Choose Optimizers

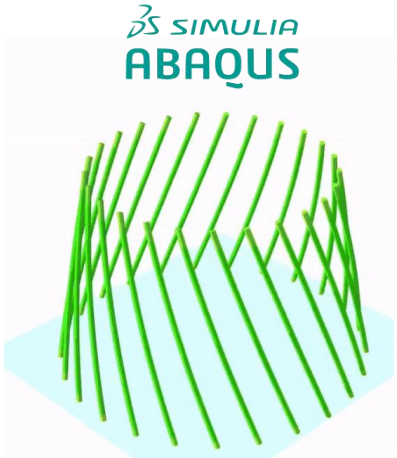
Body in White (BIW) of an Audi A2

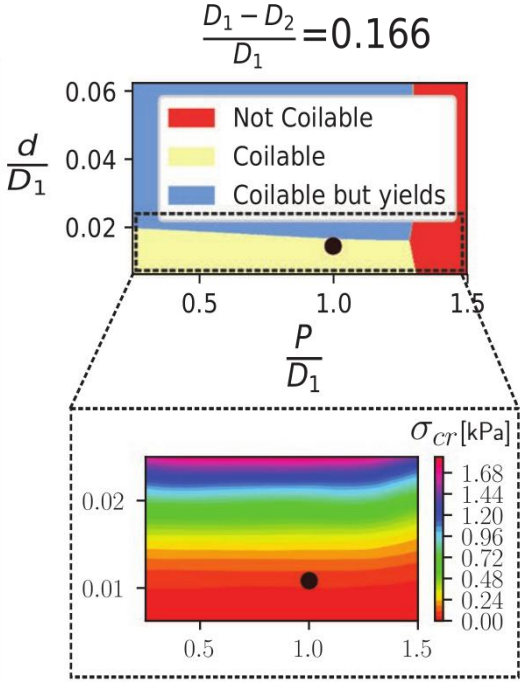


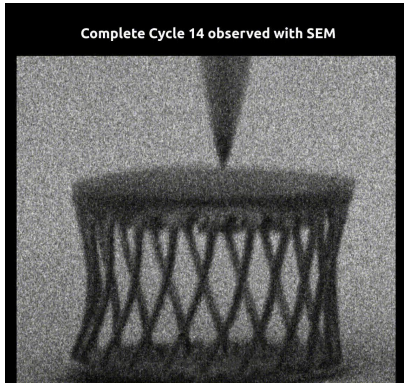
Fragile becomes supercompressible[1]



- **Goal**: Design supercompressible meta-material from brittle PLA





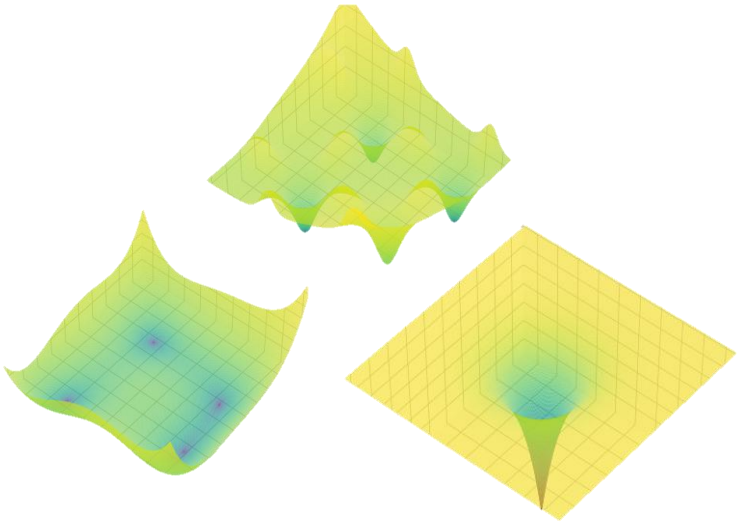


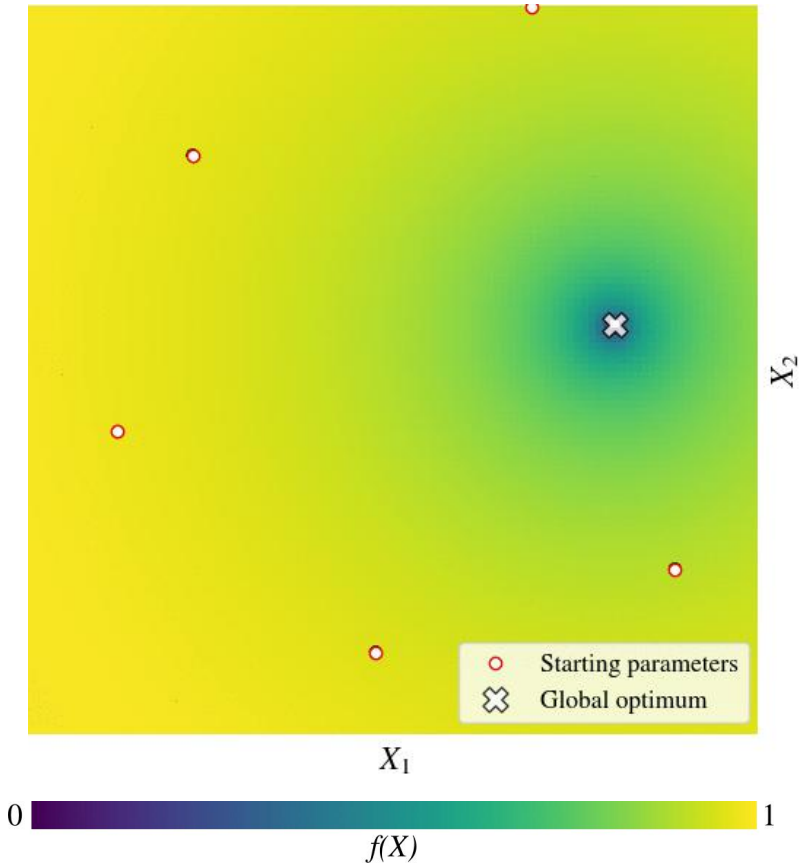
Determine initial design parameters → Sobol sequence sampling → Finite Element Analyses → Gaussian Process Classification & Regression → Optimization

[1] Bessa, M. A., Glowacki, P., & Houlder, M. (2019). Bayesian machine learning in metamaterial design: Fragile becomes supercompressible. *Advanced Materials*, *31*(48), 1904845.

Learning to Choose Optimizers

- **Goal**: Train a meta-optimizer to beat the ‘No Free Lunch’ theorem[2]





nD continuous input → Latin-Hypercube sampling → Evaluate a benchmark function → Optimization with various optimizers → Recommender system

[2] D. H. Wolpert and W. G. Macready, No free lunch theorems for optimization, IEEE Transactions on Evolutionary Computation 1, 67 (1997).

Data-driven process

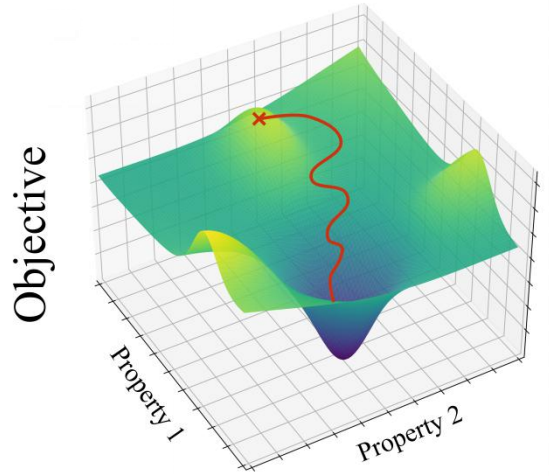
Looks like completely different studies ..

nD continuous input → Latin-Hypercube sampling → Evaluate a benchmark function → Optimization with various optimizers → Recommender system

Determine initial design parameters → Sobol sequence sampling → Finite Element Analyses → Gaussian Process Classification & Regression → Optimization

Data-driven process

Optimization



Design of Experiments

Machine Learning

Data generation

Data-driven process

Looks like completely different studies ..

nD continuous input → Latin-Hypercube sampling → Evaluate a benchmark function → Optimization with various optimizers → Recommender system

Determine initial design parameters → Sobol sequence sampling → Finite Element Analyses → Gaussian Process Classification & Regression → Optimization

Data-driven process

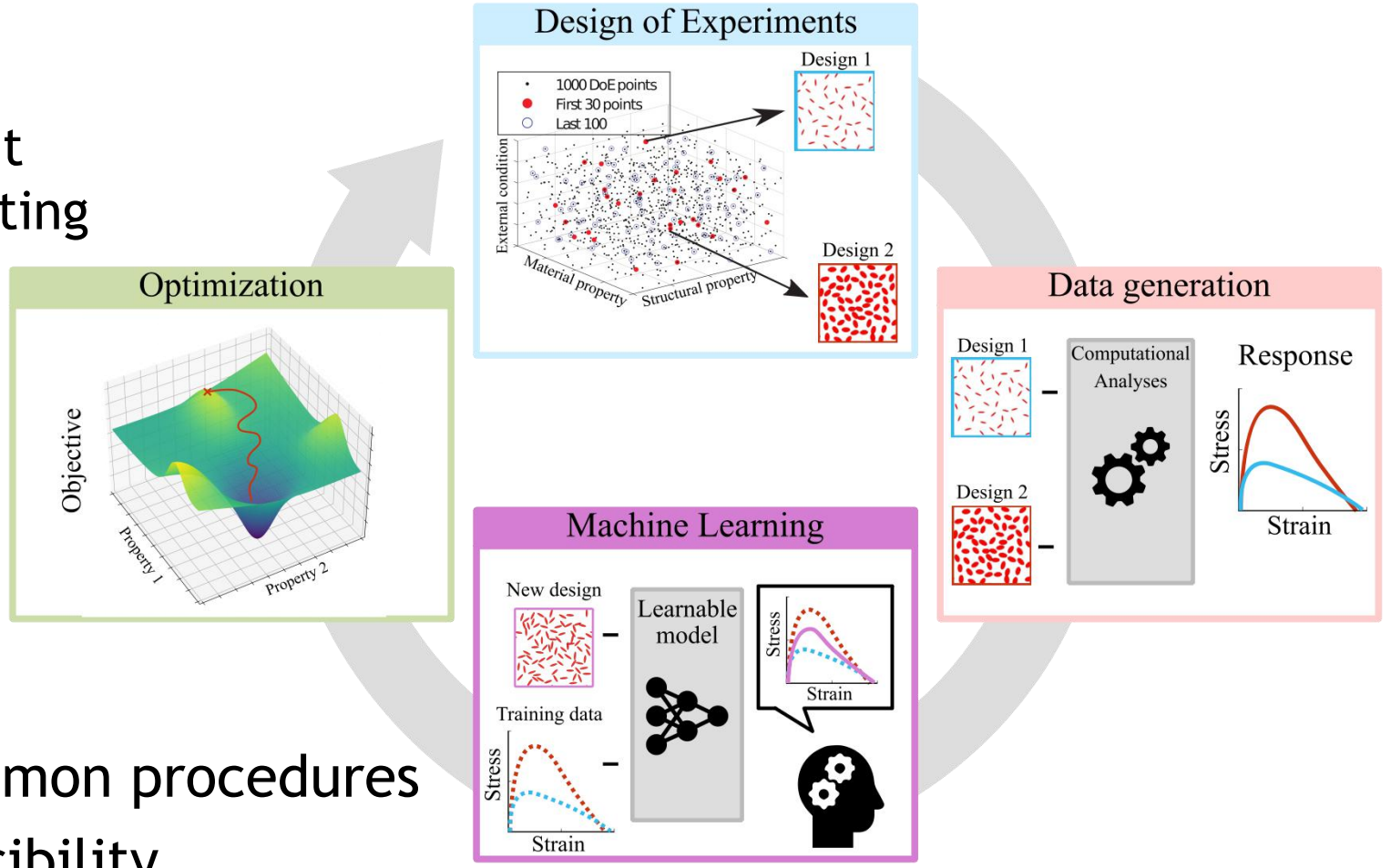
Abstraction discovers similarity in the workflow!

nD continuous input → Latin-Hypercube sampling → Evaluate a benchmark function → Optimization with various optimizers → Recommender system

Determine initial design parameters → Sobol sequence sampling → Finite Element Analyses → Gaussian Process Classification & Regression → Optimization

Challenges

- Different order of elements
- Practical challenges
  - Proper data management
  - Efficient parallel computing
  - Third-party software
- Open-source software



Opportunities

- Reinvent the wheel for common procedures
- Benchmarking and reproducibility

Framework for data-driven design and analysis of structures and materials

What's in the box ?

Framework for data-driven design and analysis of structures and materials

- ✅ Compatible with the **three major operating systems** and Python 3.8+
- ⭐ **User-friendly interfaces** for each of the stages in the data-driven process
- 🏃 **Built-in defaults** for getting started right away!
- 📈 Automatic **data-management** of your experiments
- 💾 Easy parallelization on **high-performance computing systems**

JOSS
The Journal of Open Source Software

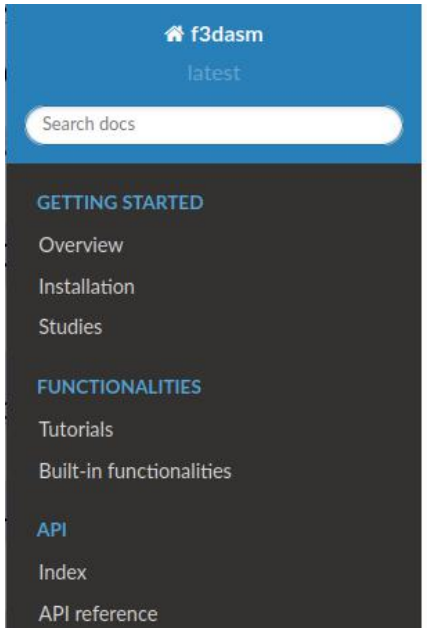
van der Schelling, M. P., B. P. Ferreira, and M. A. Bessa. "f3dasm: Framework for Data-Driven Design and Analysis of Structures and Materials." Journal of Open Source Software 9.100 (2024): 6912.

What's in the box ?

📖 Extensive online documentation

- Overview and statement of need
- Installation instructions for users and developers
- List of built-in defaults
- Tutorials and case studies
- Fully covered API documentation

https://www.f3dasm.readthedocs.io



Getting started

- Install as Python package hosted by PyPI or conda-forge:

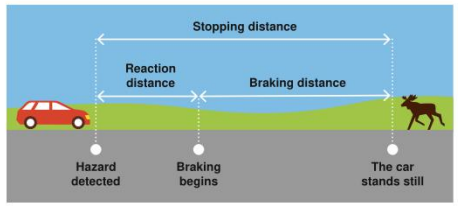
```shell
pip install -U f3dasm
conda install f3dasm
```

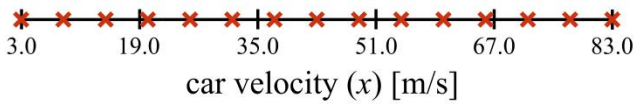
- .. or clone the source code from the GitHub repository:

```shell
git clone https://github.com/bessagroup/f3dasm.git
```

Practical session: car stopping distance problem

Design of Experiments

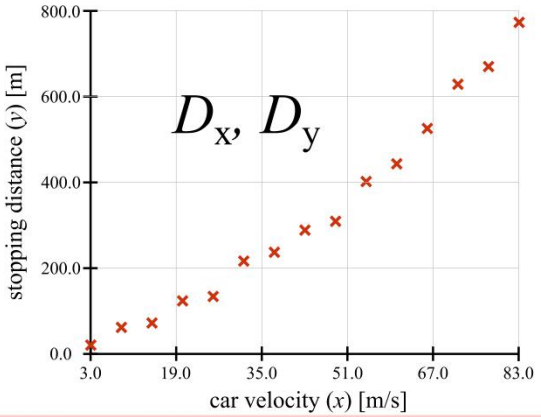




D_x

Data generation

$$y = zx + 0.1x^2$$



D_x, D_y

Machine Learning