OpenFOAM Stammtisch 2026 - Götingen

OptiFS: Optimization Framework with Simulation Software

PROBLEM & SOLUTION WE FOUND

We were looking for:

- Open source, parametric CAD optimisation framework
- Which can use advance multi-objective optimisation algorithms.

So we build one:

- **Framework**: Python
- **Optimisation algorithm**: Pymoo (Genetic algorithm)
- **CAD software:** Spaceclaim, or Excel based parametric CAD
- **Meshing**: Fluent, cfMesh
- **Solver**: OpenFOAM
- **Post processor**: Paraview

We call it OptiFS: Optimisation Framework with Simulation Software

AGENDA

- Introduction
- How OptiFS works
- How we use it
- How you can use it
- How we can improve it

INTRODUCTION

Jaydev Dave

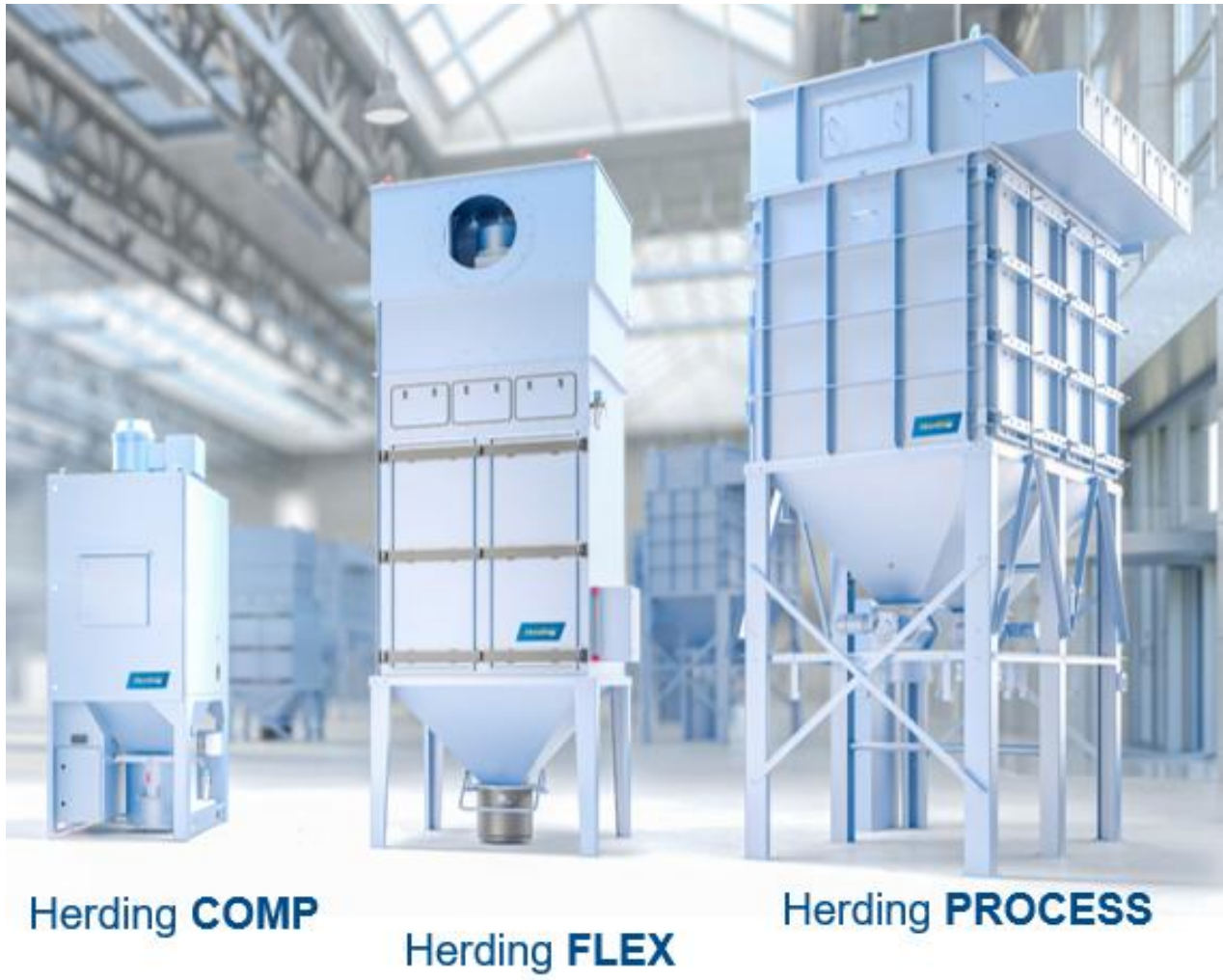
CFD Simulation Engineer

F&E

Herding GmbH Filtertechnik
Amberg, Byern

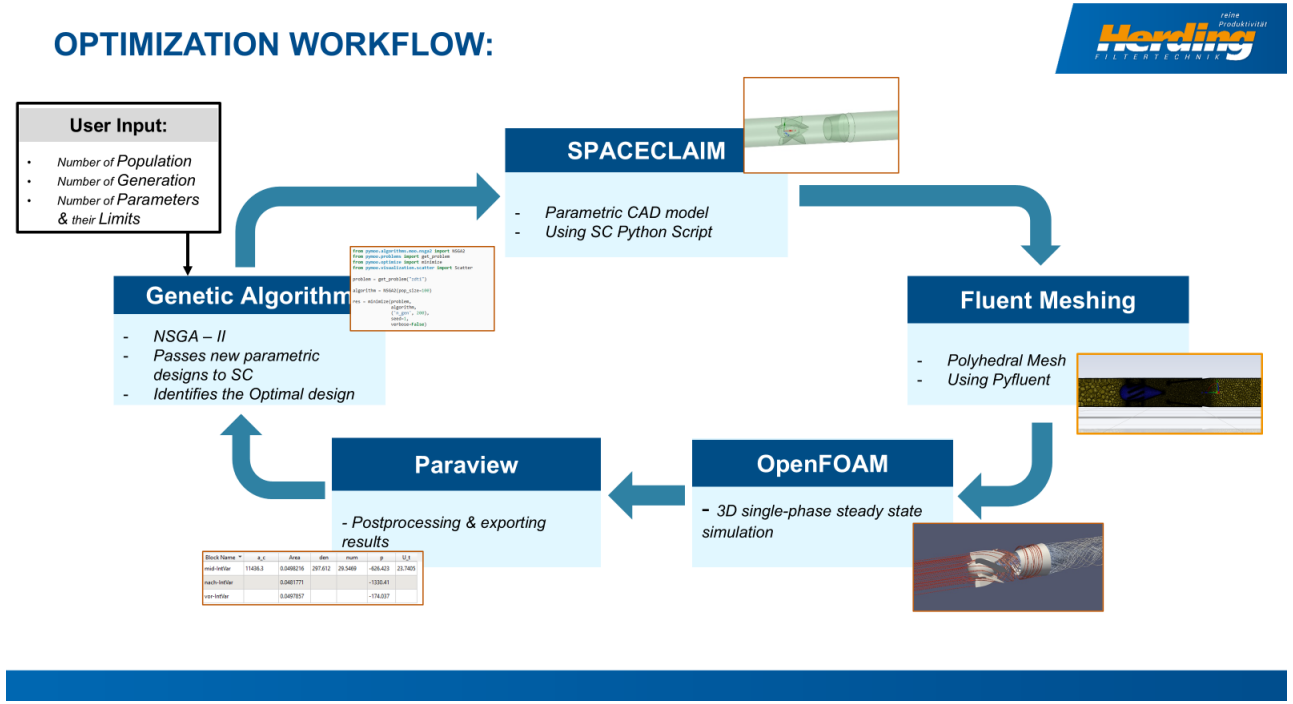
- Born and studied in India
- Bachelor Of Mechanical Engineering
- M.Sc. In Management Engineering In Computer Aided Mechanical Engineering At RWTH Aachen
- 2+ years of experience in CFD

Filter system series



OptiFS: How it works
Workflow

- Let first understand the workflow:

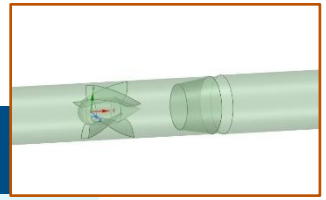


OPTIMIZATION WORKFLOW:

User Input:

- *Number of* Population
- *Number of* Generation
- *Number of* Parameters *& their* Limits

SPACECLAIM



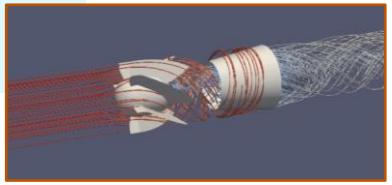
- *Parametric CAD model*
- *Using SC Python Script*

Fluent Meshing

- *Polyhedral Mesh*
- *Using Pyfluent*

OpenFOAM

- *3D single-phase steady state simulation*



Paraview

- *Postprocessing & exporting results*

Block Name	a_c	Area	den	num	p	U_t
mid-IntVar	11436.3	0.0498216	297.612	29.5469	-626.423	23.7405
nach-IntVar		0.0481771			-1330.41	
vor-IntVar		0.0497857			-174.037	

Genetic Algorithm

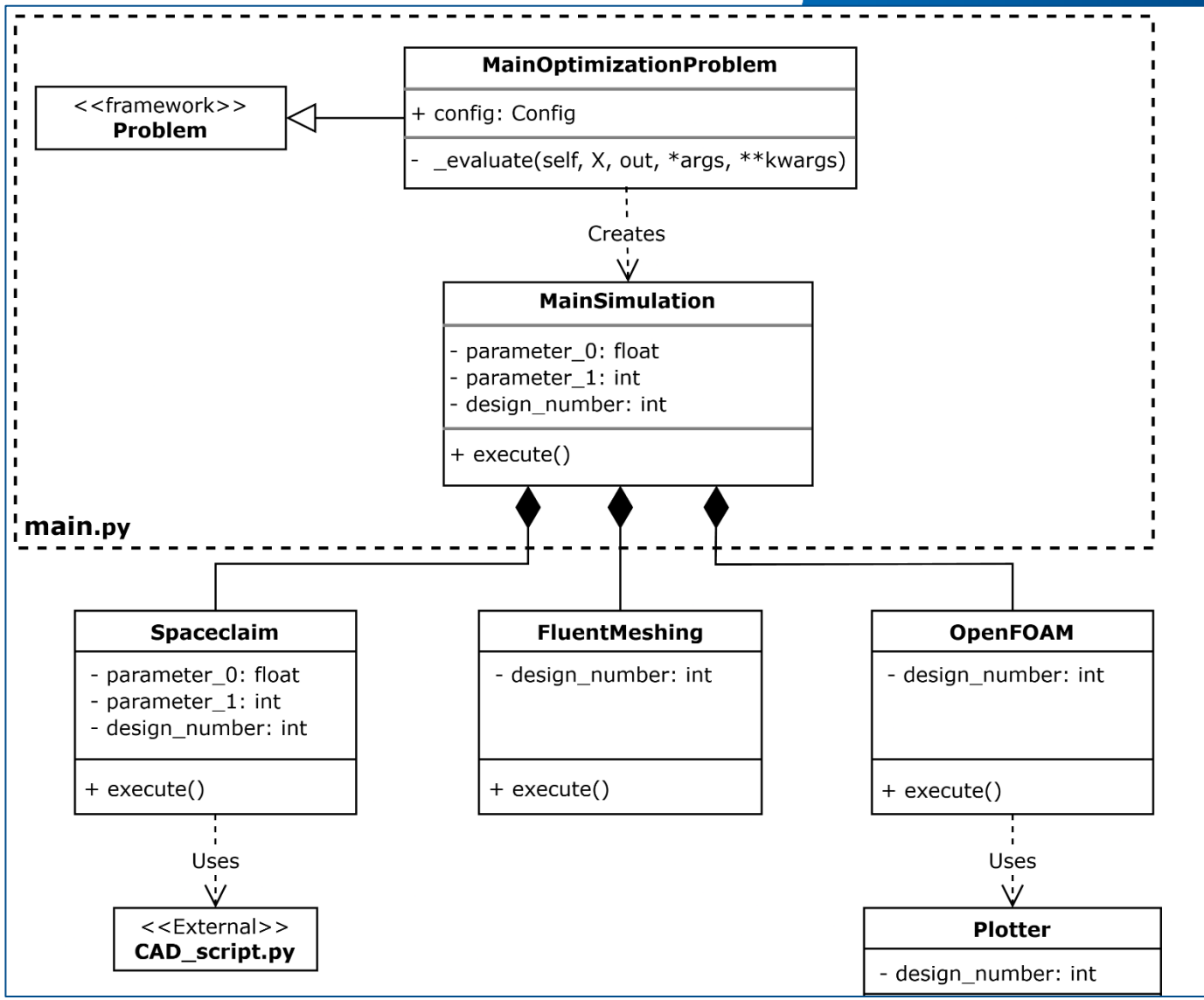
- *NSGA – II*
- *Passes new parametric designs to SC*
- *Identifies the Optimal design*

```
from pymoo.algorithms.moo.nsga2 import NSGA2
from pymoo.problems import get_problem
from pymoo.optimize import minimize
from pymoo.visualization.scatter import Scatter

problem = get_problem("zdt1")

algorithm = NSGA2(pop_size=100)

res = minimize(problem,
               algorithm,
               ('n_gen', 200),
               seed=1,
               verbose=False)
```

OptiFS: How it works
Architecture

- Main.py contains:
  - Optimisation algorithm
    - Which passes the CAD parameter values to Spaceclaim.py
  - Spaceclaim.py generates the CAD
  - That CAD goes to Fluentmeshing.py
  - And so on…
- You can access the code at:

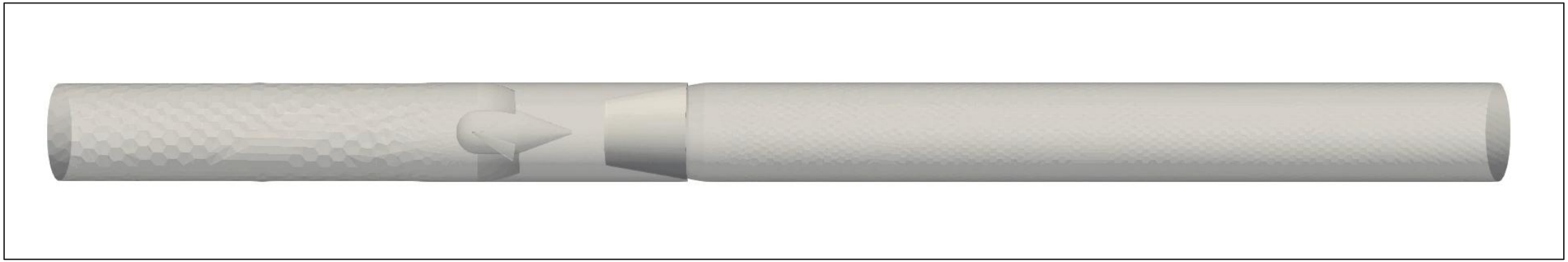
  https://github.com/jaydev-dave/OptiFS



UML Diagram

OptiFS: How we use it

- Optimisation filter element geometry.
- In past, my master thesis on hydrogen fuel cell optimisation used similar framework for STARCCM++
- Cyclone separator



OptiFS: How you can use it

- It is open source project.
- Clone the Github repo: https://github.com/jaydev-dave/OptiFS
- Follow the README, we created one tutorial

- All good, but I do not have access to spaceclaim and fluent meshing:
  - Instead of spaceclaim.py you can use any scriptable CAD software(FreeCAD) or excel based parametric CAD
  - Use cfmesh or any other command based masher

OptiFS: How we can improve it

- We are open for contribution.
- Repo restructuring
- Currently, switching a software need a bit of extra work.
  - It should be easy for user to use numbers of softwares.
- More software support could be added such as STARCCM++ and better mesher.
- Currently it only supports Pymoo based genetic algorithm.
  - More smatter algorithm picker can be added like how HEEDS SHERPA works

Please share your

Thoughts & Questions !

QR code for repo link

https://github.com/jaydev-dave/OptiFS

M.Sc
Jaydev Dave
CFD Simulation Engineer
Research & Development

Herding GmbH Filtertechnik
August-Borsig-Str. 3
GERMANY

Phone: +499621630-392
Dave.Jaydev@herding.de
www.herding.com

THANK YOU !