

Automated Mechanical Engineering Design using Open Source CAE Software Packages

Fenics'18, Oxford, March 2018

Presented by Qingfeng Xia

Department of Engineering Science, University of Oxford

March 22, 2018

Outline

PDRA, Rolls-Royce UTC, Dept of Engineering Science, the University of Oxford.

1. Current and future CAE process
2. Introduction to design automation by Fenics and FreeCAD
3. Future plan on Automated CAE

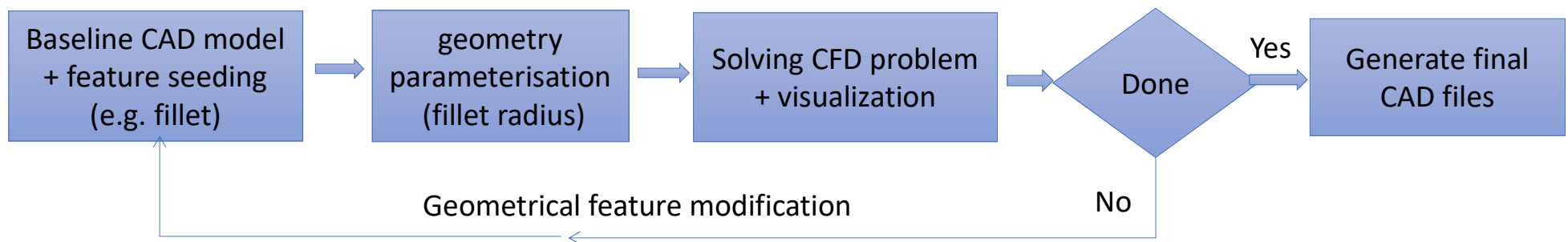
Acknowledgement:

This workshop registration is sponsored by Patrick

1. integrated design for Mechanical Engineering

ATI funded “*Geometry Enabled Modelling in Integrated Design Systems (GEMinIDS)*” research project (£18 million) aims to develop new gas turbine sub-system technologies with the goal of reducing engine emissions and improving fuel consumption “.

Rolls-Royce UTC of University of Southampton



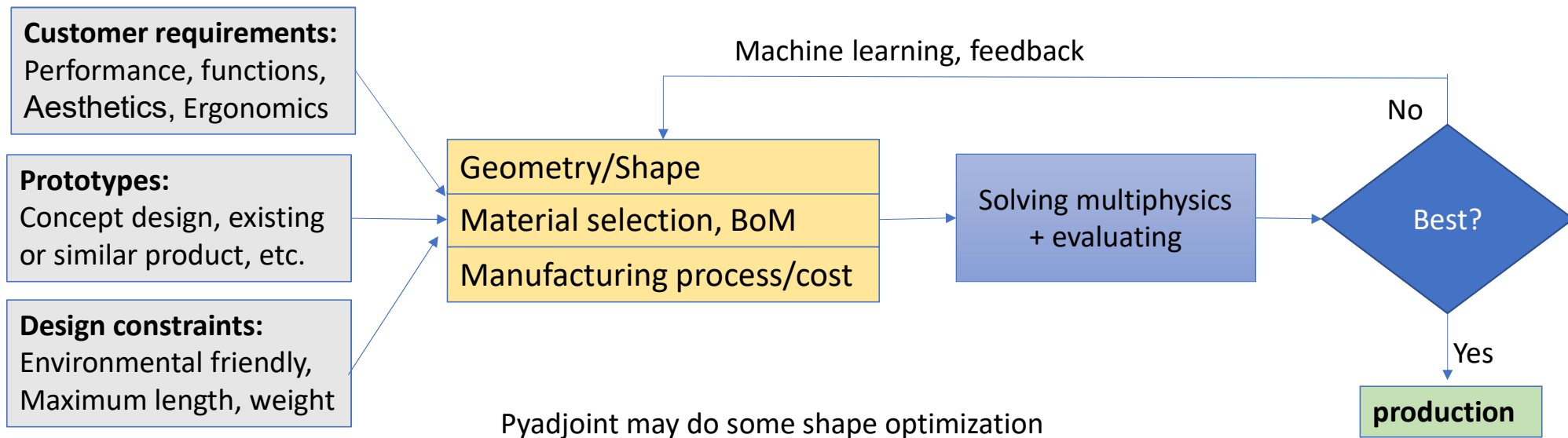
Features: geometry parameterisation and optimisation schemes to support Rolls-Royce gas turbine engine design processes (CFD).

Cons: no machine learning involved, not generalised for engineering design.

Future CAE in 4th industrial revolution

Reduce the time from concept design to product is also important for
Industry 4.0/4th industrial revolution.

The Key: automated and intelligent engineering design/simulation at system level.

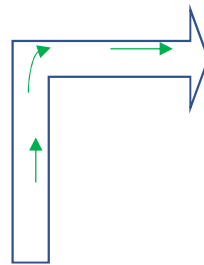
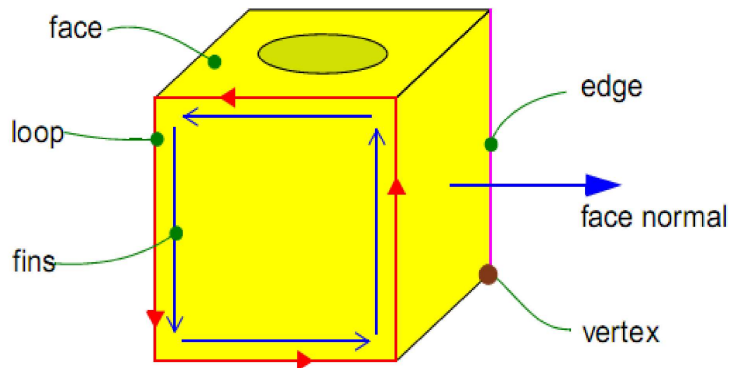


Challenges of automated CAE

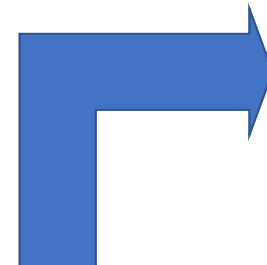
- CAD: changed topology of geometry after surface modification
- Automatic Meshing:
 - CFD is very sensitive to mesh quality, automatically meshing is not possible;
 - geometry defeature is essential but time-consuming for complicate geometry.
- Pre-processing:
 - lack of public material database (for case setup and material substitution);
- Computation:
 - to guarantee completion and correctness of simulation
- Post-processing/evaluation:
 - to shed light on geometry modification for better performance

Geometrical topology optimisation

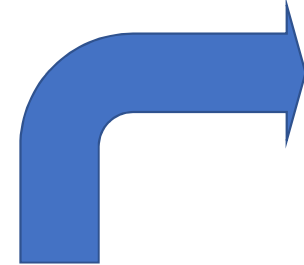
BREP: solid is constructed from an ordered-list of faces: each face has a unique integer ID



pipe flow

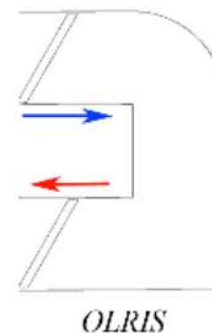
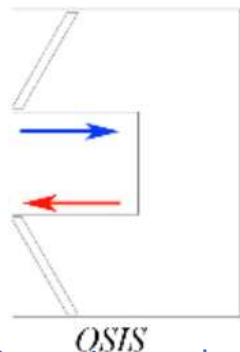
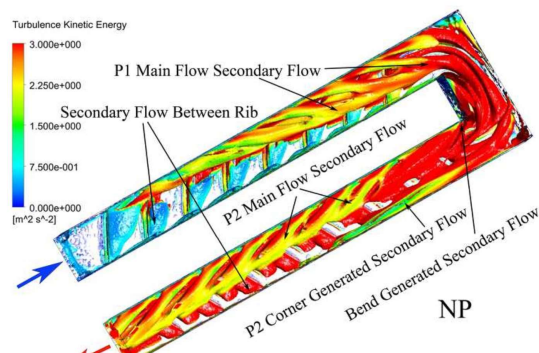


Parameter (diameter)



+ feature modification

Introduction of feature (hole, rib) onto a surface will change face ID -> invalidate meshing and boundary setup.

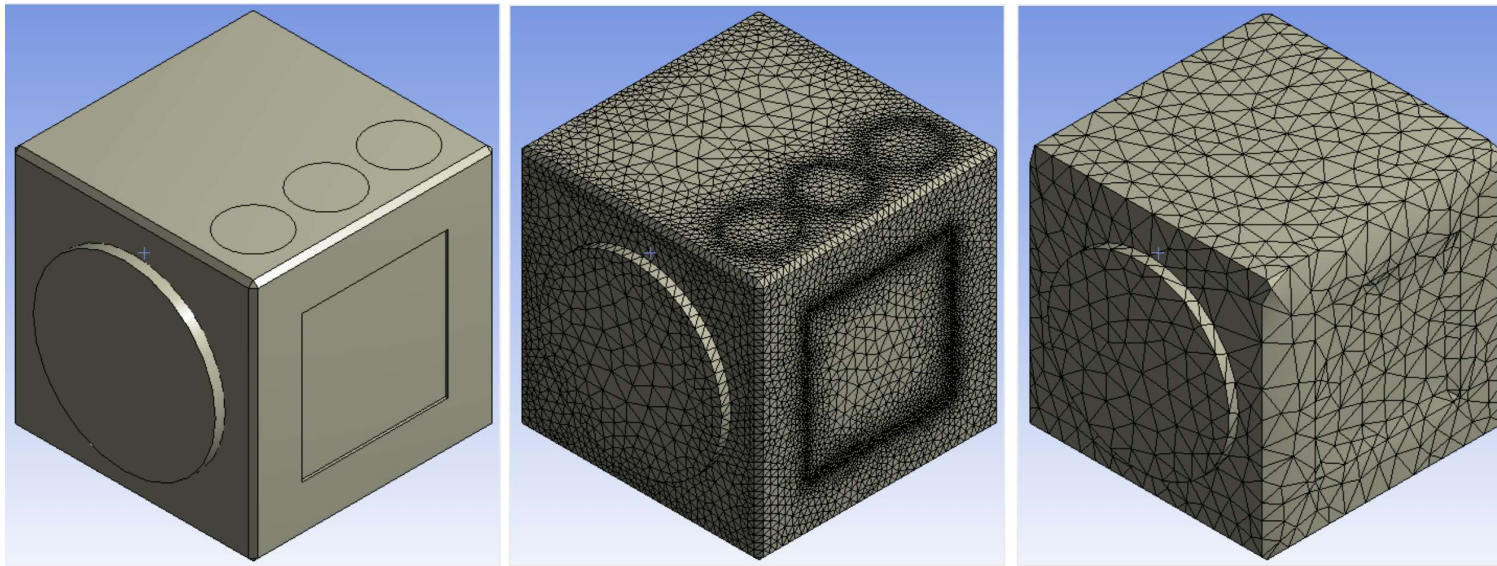


13% pressure drop reduced by bend shape optimisation for blade internal cooling,

Numerical study of the influence of rib orientation on heat transfer enhancement in two-pass ribbed rectangular channel, by Zhu Jiangnan, Tieyu Gao, Qingfeng Xia

Geometry defeature for meshing

Even with modern super-computer, whole turbine level simulation is still not feasible, therefore, mesh defeature is essential to accelerate computation.

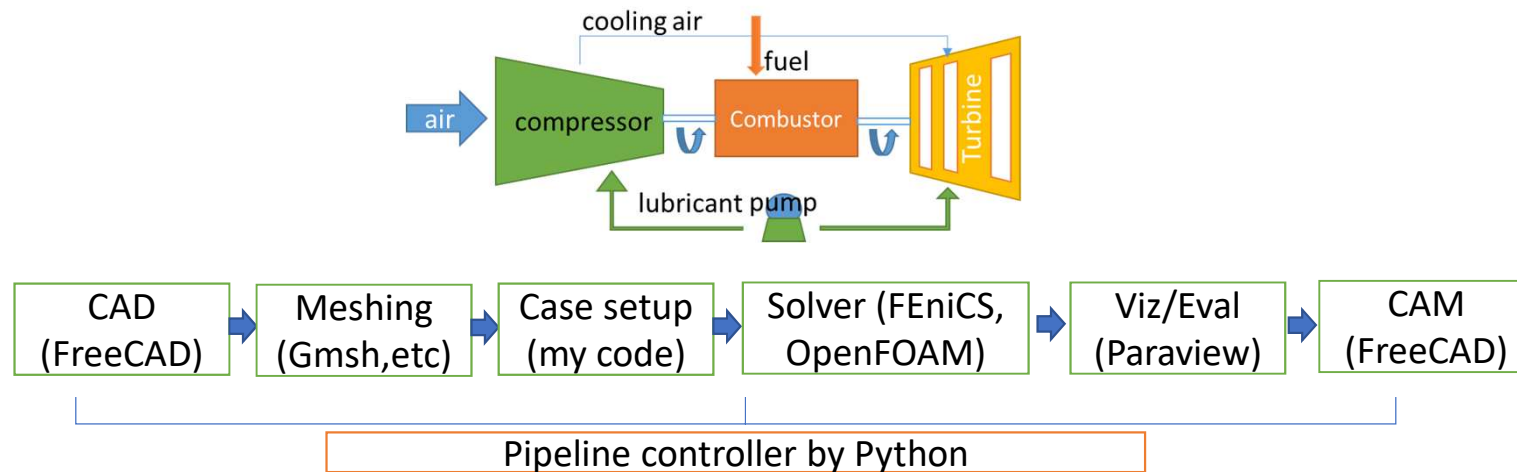


Source: Ansys Manual (geometry, automatic meshing, after defeaturing)

However, automatic mesh defeature is not available/usable from commercial solution, e.g. Ansys CFD, Rolls-Royce Hydro.

Pipelined CAE for a specific component

Currently, CAE is pipelined by Python supervising code for simulation automation. LGPL licensed Python open source packages are used as possible.



- Simulation process is automated for parametric geometry;
- Template Gmsh textual input is used to generate appropriate mesh;
- Mesh independence and performance evaluation is conducted automatically.

Introduction to FreeCAD

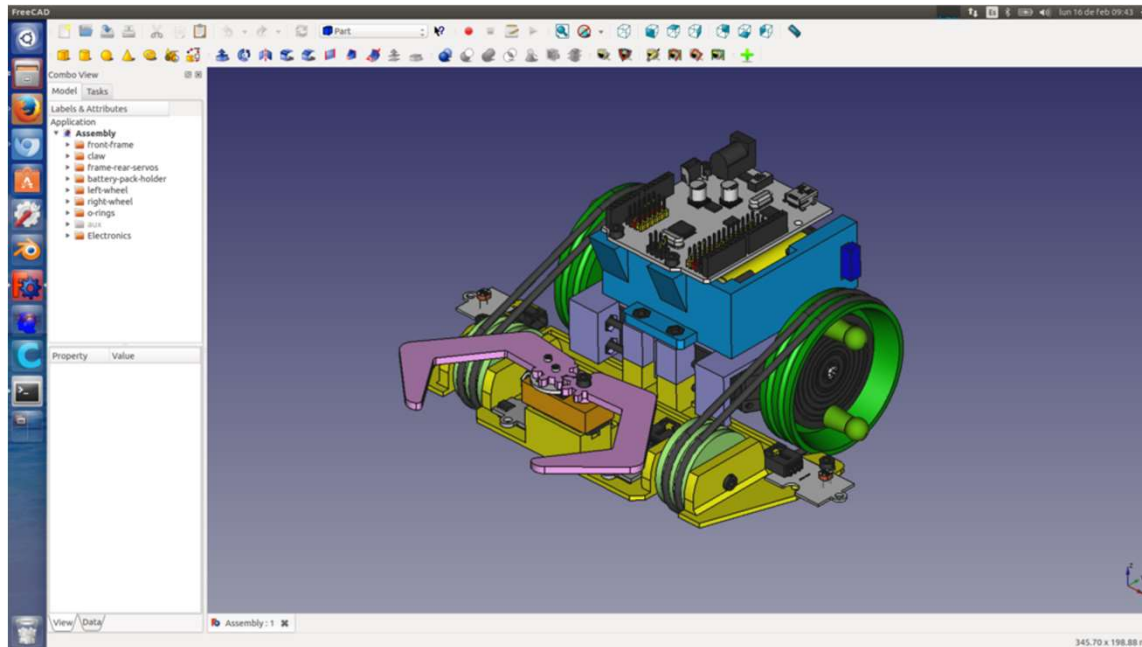


<https://www.freecadweb.org>

- Most feature complete open source CAD solution
- Open source LGPL licensed, cross-platform
- Standalone GUI or as Python modules
- Easy to extend (create new module)

I am the original author for
CFD module of FreeCAD

<https://github.com/qingfengxia/Cfd>



GUI operation can be recorded
as executable python script for
automated engineering design.

2. FenicsSolver (textual case input example)

```
solver = FenicsSolver.ScalerEquationSolver(setting)
result = solver.solve() # all you need after case setup
```

Python code

```
1 settings = {
2   'solver_name': 'ScalerEquationSolver',
3   'mesh': None, 'function_space': Q, 'periodic_boundary': None, 'fe_degree': 1,
4   'boundary_conditions': bcs,
5   'body_source': None,
6   'initial_values': {'temperature': T_ambient},
7   'material': {'density': 1000, 'specific_heat_capacity': 4200, 'thermal_conductivity':
8     0.1},
9   'solver_settings': {
10     'transient_settings': {'transient': False, 'starting_time': 0, 'time_step': 0.1,
11       'ending_time': 1},
12     'reference_values': {'temperature': T_ambient},
13     'solver_parameters': { # mapping to solver.parameters of Fenics
14       "relative_tolerance": 1e-9,
15       "maximum_iterations": 500,
16       "monitor_convergence": True, # print to console
17     },
18   },
19   # solver specific settings
20   'scaler_name': 'temperature',
21 }
```

<https://github.com/qingfengxia/FenicsSolver>

Object-oriented design, all solver is based on “SolverBase” class
This “ScalerEquationSolver” is a general scaler solver.

Why FenicsSolver?

1. Deal complex geometry in FreeCAD
2. Automate CAE/ no python coding
3. GUI for new user (as addon for FreeCAD)

Current status:

Tested:

- Linear elasticity
- Incompressible NS Equation
- Scaler diffusion-advection
(has problem for 3D advection)

Under-writing:

- Compressible flow
- Maxwell EM equation

Comment or contribution is warmly welcomed!

Fenics boundary condition for Multiphysics

```
from collections import OrderedDict
bcs_u = OrderedDict()

bcs_u["static"] = {'boundary': static_boundary, 'boundary_id': 1,
                   'values': [{ 'variable': "velocity", 'type': 'Dirichlet', 'value': zero_vel, 'unit': 'm/s'},
                              { 'variable': "temperature", 'type': 'Dirichlet', 'value': T_wall}   ]}

if using_3D:
    bcs_u["front_back"] = {'boundary': front_back_boundary, 'boundary_id': 5,
                           'values': [{ 'variable': "velocity", 'type': 'Dirichlet', 'value': zero_vel},
                                      { 'variable': "temperature", 'type': 'Dirichlet', 'value': T_wall}   ]}

bcs_p = OrderedDict()
bcs_p["outlet"] = {'boundary': outlet, 'boundary_id': 3,
                   'values': [{ 'variable': "pressure", 'type': 'Dirichlet', 'value': p_outlet},
                              { 'variable': "temperature", 'type': 'Dirichlet', 'value': T_ambient}   ]}
```

Example boundary setup for Navier-Stokes equation (fluid field) solver

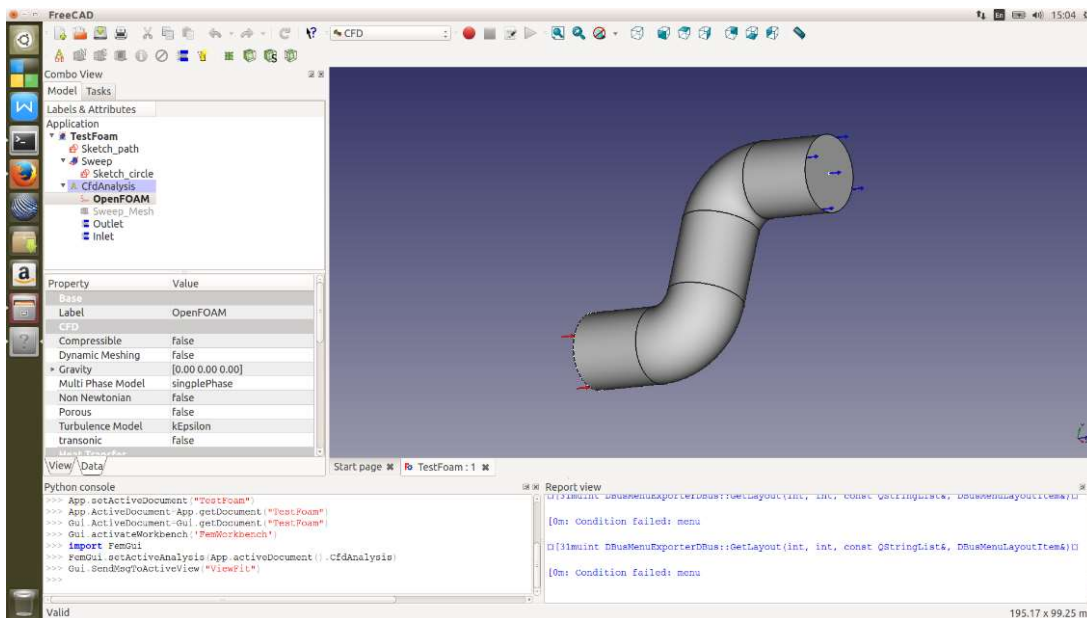
Each boundary setting contains info for 3 variables: Velocity, pressure and temperature.
It is designed for coupling Multiphysics solver.

CFD module for FreeCAD

<https://github.com/qingfengxia/Cfd>

GUI for Fenics and OpenFOAM

OpenFOAM (written in C++ , finite volume method) has plenty solvers for complex flow and turbulent flow



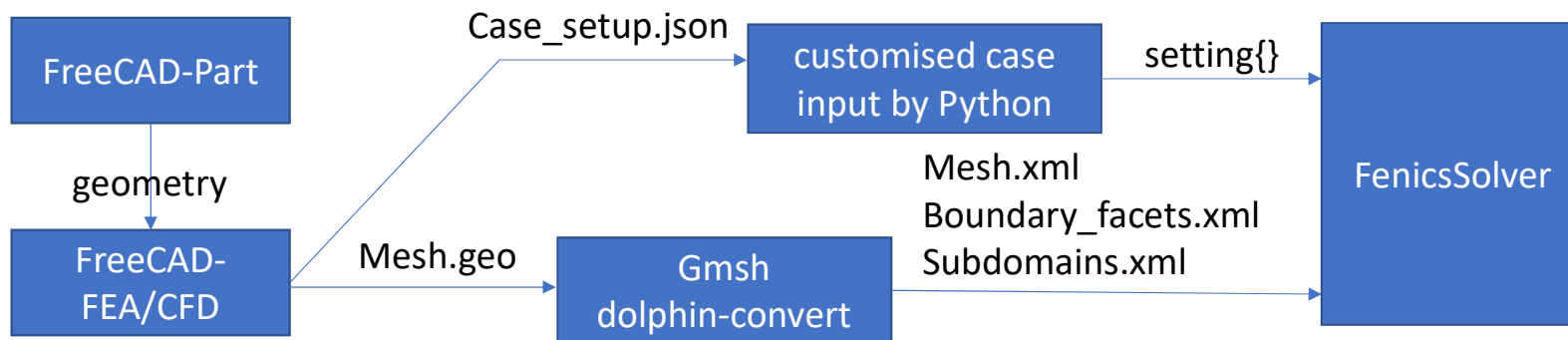
```
fvVectorMatrix UEqn
(
    fvm::ddt(U)
    + fvm::div(phi, U)
    - fvm::laplacian(nu, U)
);

solve(UEqn == -fvc::grad(p));
```

Navier-Stokes equation in OpenFOAM

- FoamCaseBuilder: python package I made to control OpenFOAM solver

Fenics as FreeCAD FEA solver



Note: *.geo file can be find in FreeCAD FEM/CFD module's working folder/tmp folder

1. FreeCAD's FemConstraint objects are created with a list of surface IDs, selected in FreeCAD GUI.
2. "Fenics CaseWriter" (in FreeCAD CFD module) allocates an ID ('boundary_id') for each FemConstraint.
3. FreeCAD calls Gmsh to export all *FemConstraint* objects into "boundary_facets.xml" , with 'boundary_id' matched.
4. In **FenicsSolver**, redefine 'ds' by "boundary_facets" and "ds('boundary_id')" to apply boundary condition

Topology independence: FreeCAD + CADQuery

CadQuery, inspired by jQuery (a popular javascript framework for web development), makes selection of topology elements incredibly easy

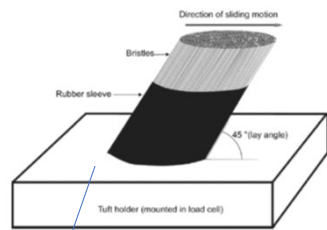
Code sample for CADQuery to select the face with max Z-axis coordinate by `faces(">Z")`:

```
# Create a box based on the dimensions above and add a 22mm center hole  
result = cq.Workplane("XY").box(length, height, thickness) \  
    .faces(">Z").workplane().hole(center_hole_dia)
```

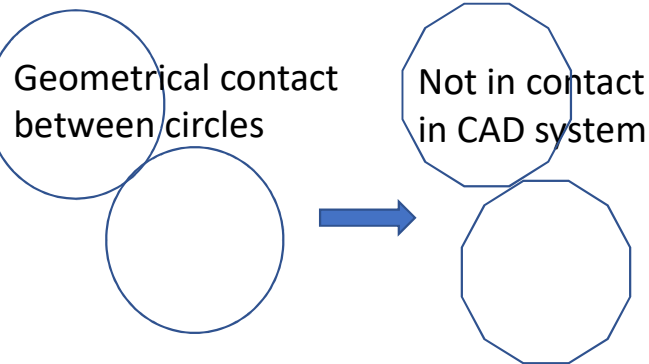
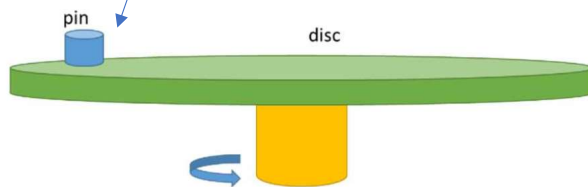
Corresponding to Fenics's : `near(x[2], zmax)` for boundary faces selection regardless of topology.

Multi-scale thermal modelling of brush seal

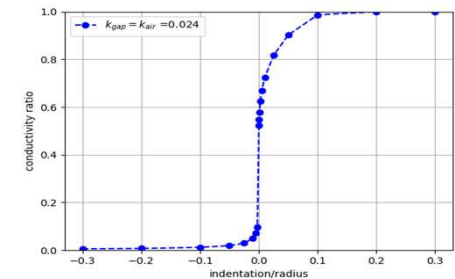
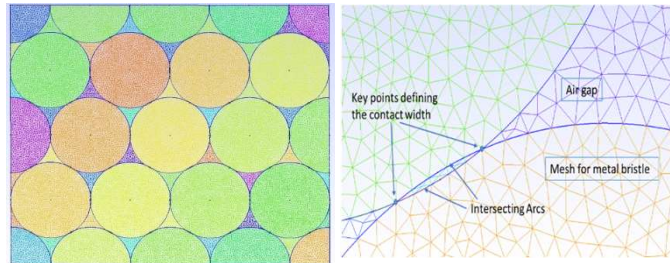
Example of pipelined CAE, applicable to other engineering design, compound material, MEMS



1. Microscale model bristle tuft as a solid pin with porous medium anisotropic heat conductivity (K)



2. Caution in geometry building and meshing multiscale contact problems



3. Conductivity K as a function of bristle packing parameters,

$$K = \begin{bmatrix} k_r & 0 & 0 \\ 0 & k_r & 0 \\ 0 & 0 & k_z \end{bmatrix}$$

4. Two-body heat transfer is model in macroscale as quasi-static way by my [FenicsSolver](#)

ASME TURBO-EXPO 2018: by [Qingfeng Xia](#), [David Gillespie](#), etc.

Quasi-static Thermal Modelling Of MULTI-SCALE Sliding Contact for Unlubricated Brush Seal Materials,
 “The most completed study of tuft/pin on disc”, recommended to publish in *Journal of Turbomachinery*

3. Future plan on FenicsSolver

Comment or contribution is warmly welcomed!

<https://github.com/qingfengxia/FenicsSolver>

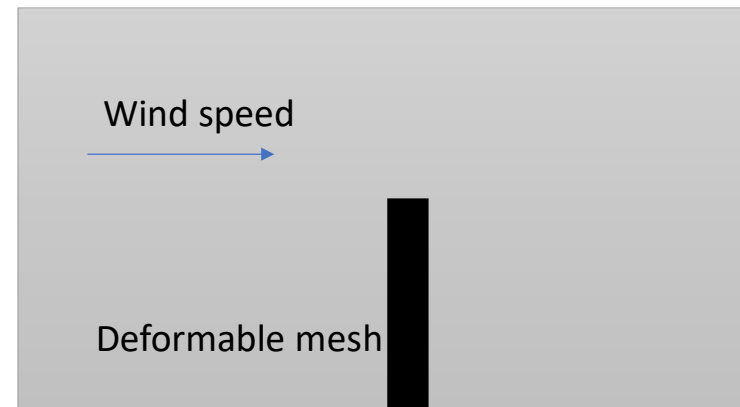
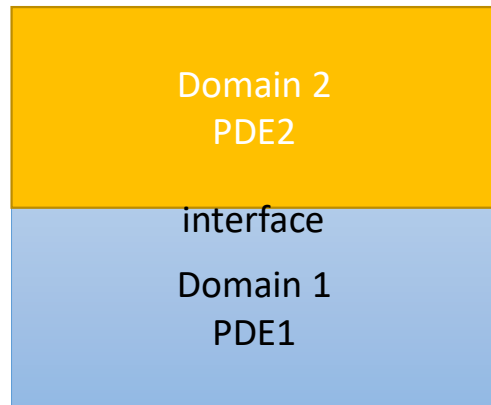
I hope to hear from community, which directions are worth doing.

- 1) coupling of Multiphysics and multi-body PDE
- 2) coupling with external solvers like OpenFOAM
- 3) coupling with System engineering modelling

3.1 Multiphysics and multi-body coupling

Target:

- reuse the developed solvers from FenicsSolver
- Enable multi-body and interface capturing and for imported geometry/mesh from FreeCAD
- Enable close coupling like: Flow-structural interaction

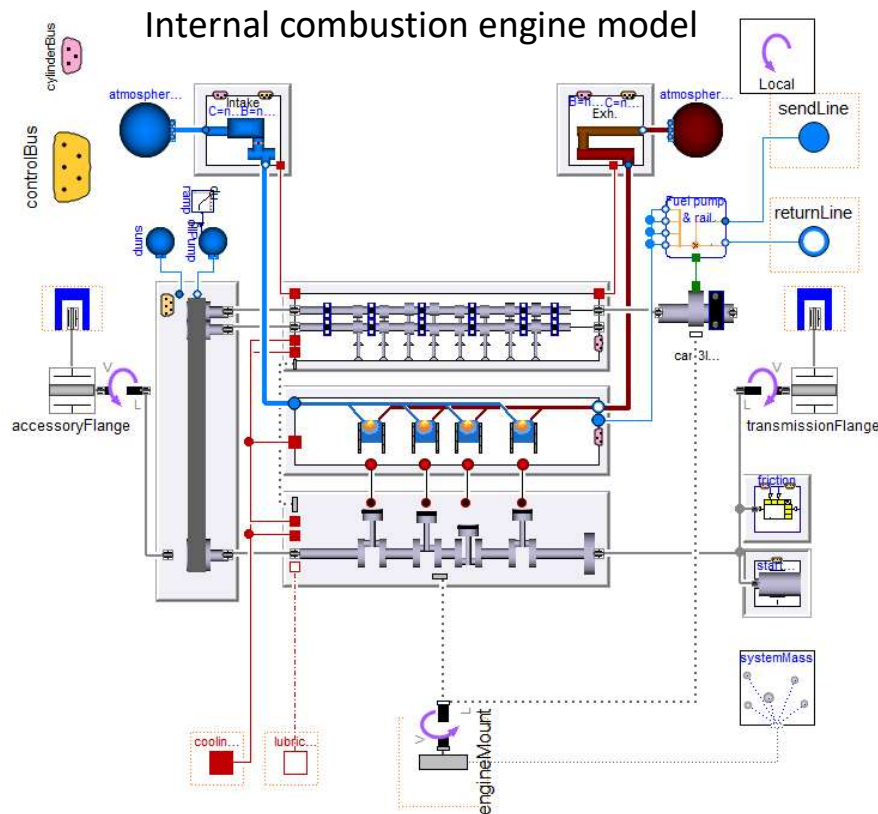


Flow-structural interaction

3.2 Coupling with OpenFOAM

- Advantages of OpenFOAM: multi-phase, turbulent flow
- VTK might be the data exchange format at interface.
- A coupling coordinator can be written in Python to control FenicsSolver and OpenFOAM.

3.3 System modelling + component CAE



System modelling (component network style modelling by [Simulink/Modelica](#)) focuses on dynamic response (ODE), while directly solving PDE for some key component is desired for model accuracy.

Q: what is the best way to coordinate different simulation process?

source: <http://www.claytex.com/blog/vesyma-engines-library/>