Hochschule Konstanz
Fakultät Informatik

Kick off – Teamprojekt simFlow

23.03.2022

Hintergrund

Um Eigenschaften von Bauteilen zu berechnen (z.B. Stabilität, thermische Eigenschaften, aerodynamische Eigenschaften etc.), gibt es typischerweise folgenden Workflow:

- Berechnungsnetz erstellen (meist anhand von CAD-Daten)
- Wichtige Parameter setzen (Materialeigenschaften)
- Anfangs- und Randbedingungen festlegen (z.B. Geschwindigkeit am Eingang, Druck am Ausgang etc.)
- Rechnung starten

Typischerweise nutzt man sehr teure Lizenzsoftware (einerseits für CAD-Programme für die Konstruktion, andererseits Simulationssoftware wie Ansys, Comsol, Starccm, GTSuite etc.)

Idee: Openfoam ist ein sehr gut programmiertes C++-Programm für Strömungssimulation

Nachteil: über Konsole und Textdateien zu bedienen; Netzkonstruktion ist sehr mühselig

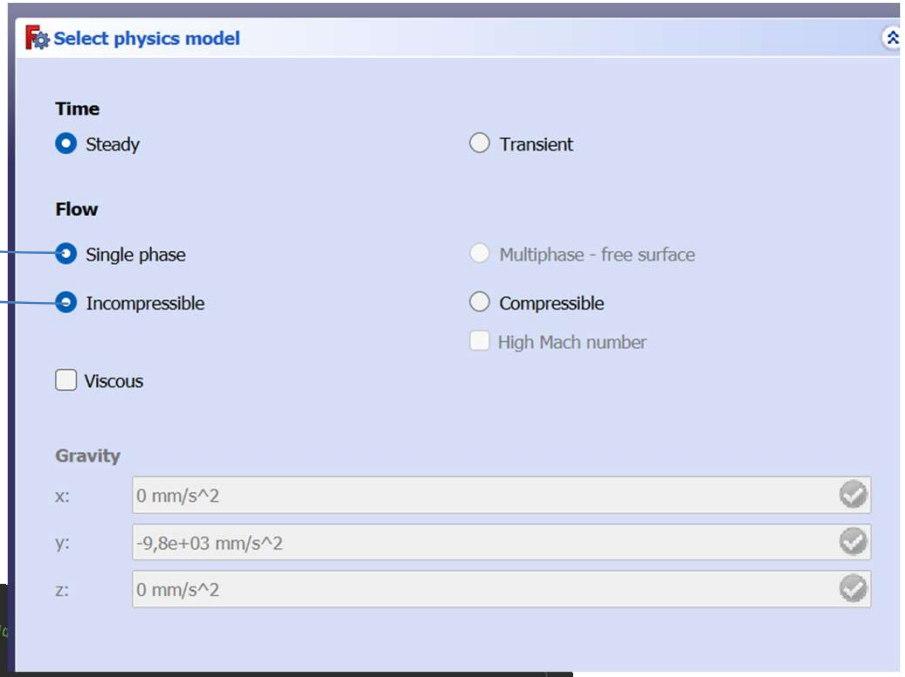
→ FreeCAD + Addon cfdOF liefert Opensource-Lösung, für die Erstellung von Berechnungsnetzen und die Berechnung! Leider sind bisher nur wenige Simulationsmöglichkeiten von Openfoam in cfdOF umgesetzt

Projektziel

1. Netzimport ermöglichen:
    1. in FreeCAD kann man – statt ein CAD-Netz zu konstruieren – direkt ein CFD-Netz in einem gängigen Format importieren.
    2. Dieses Netz wird dann von OpenFOAM in ein OpenFOAM-Netz konvertiert
2. Einen weiteren Solver implementieren und einen Testcase aufsetzen (z.B. transiente Strömungen oder stationäre Strömungen bei rotierenden Komponenten)

Beides sollte getestet und dokumentiert werden.

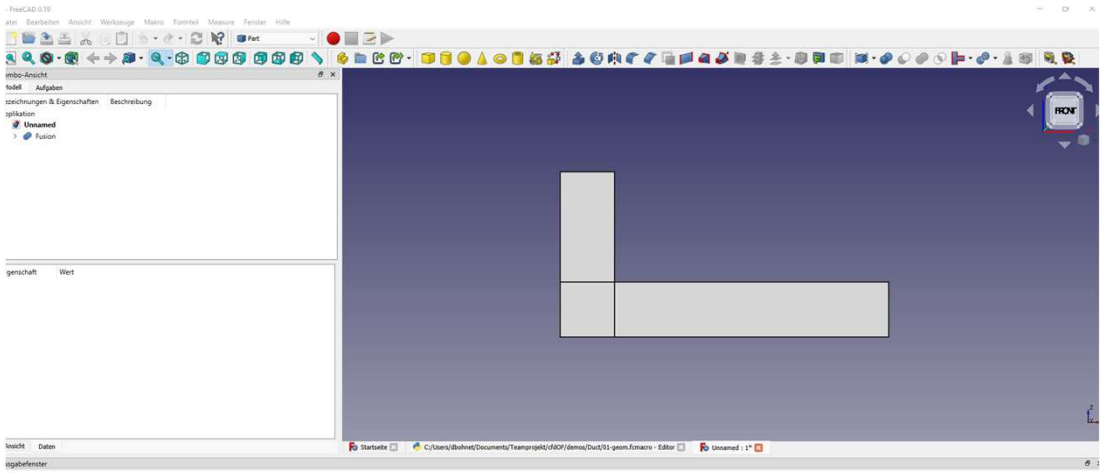
```python
def getSolverName(self):
    """ Solver name is selected based on selected physics. This should only be extended as additional physics are
    included. """
    solver = None
    if self.physics_model.Phase == 'Single':
        if len(self.material_objs) == 1:
            if self.physics_model.Flow == 'Incompressible':
                if self.physics_model.Thermal == 'None':
                    if self.physics_model.Time == 'Transient':
                        solver = 'pimpleFoam'
                    else:
                        if self.porousZone_objs or self.porousBafflesPresent():
                            solver = 'porousSimpleFoam'
                        else:
                            solver = 'simpleFoam'
                else:
                    raise RuntimeError("Only isothermal simulation currently supported for incompressible flow.")
            elif self.physics_model.Flow == 'Compressible':
                if self.physics_model.Time == 'Transient':
                    solver = 'buoyantPimpleFoam'
                else:
                    solver = 'buoyantSimpleFoam'
            elif self.physics_model.Flow == 'HighMachCompressible':
                solver = 'hisa'
```



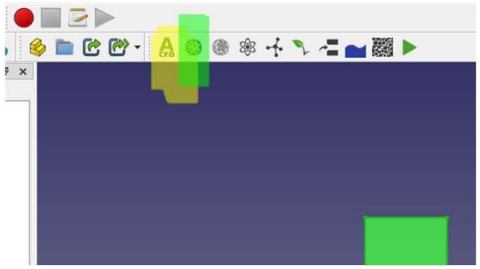
CfdCaseWriterFoam.py

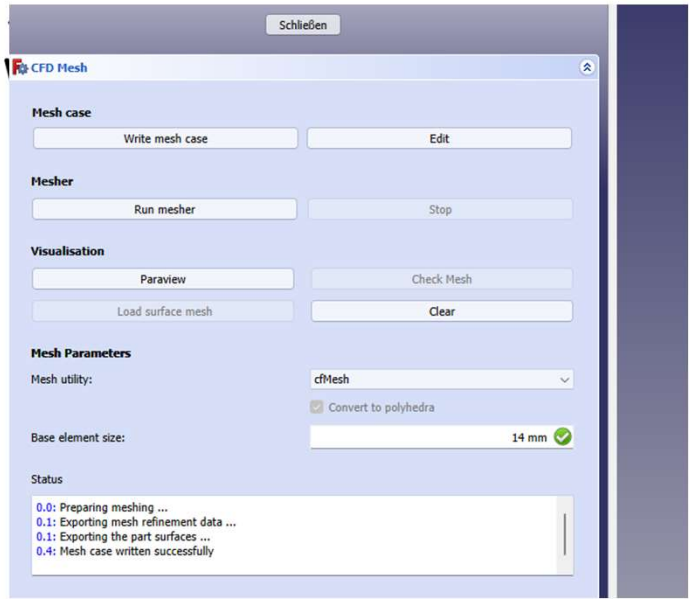
```python
def processInitialConditions(self):
    """ Do any required computations before case build. Boundc
    settings = self.settings
    initial_values = settings['initialValues']
    if settings['solver']['SolverName'] in ['simpleFoam', 'porousSimpleFoam', 'pimpleFoam']:
        mat_prop = settings['fluidProperties'][0]
        initial_values['KinematicPressure'] = initial_values['Pressure'] / mat_prop['Density']
    if settings['solver']['SolverName'] in ['interFoam', 'multiphaseInterFoam']:
        # Make sure the first n-1 alpha values exist, and write the n-th one
        # consistently for multiphaseInterFoam
        sum_alpha = 0.0
        alphas_new = {}
        for i, m in enumerate(settings['fluidProperties']):
            alpha_name = m['Name']
            if i == len(settings['fluidProperties'])-1:
                if settings['solver']['SolverName'] == 'multiphaseInterFoam':
                    alphas_new[alpha_name] = 1.0-sum_alpha
            else:
                alpha = Units.Quantity(initial_values.get('VolumeFractions', {}).get(alpha_name, '0')).Value
                alphas_new[alpha_name] = alpha
                sum_alpha += alpha
        initial_values['VolumeFractions'] = alphas_new
```

- Öffnen in freeCAD von 01-geom.fcmacro aus dem Ordner…\cfdOF\demos\Duct
- Auf den grünen Pfeil drücken (im Start-Workbench)

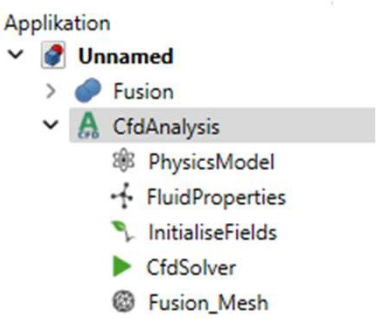


- Nun links im Modell „Fusion“ anwählen und auf die Workbench „cfdOF“ wechseln.
- Über das grüne „A“ (s. Screenshot) ein CfdAnalysis-Modell erzeugen
- Über das Netzsymbol (s. Screenshot) ein Netz erzeugen: hierfür „Write mesh case“ drücken. Anschließend „Run Mesher“. Über Paraview-Button können Sie sich das Netz anschauen.

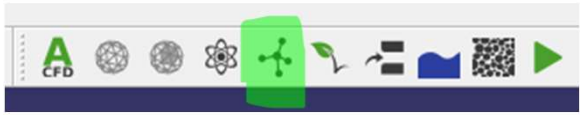




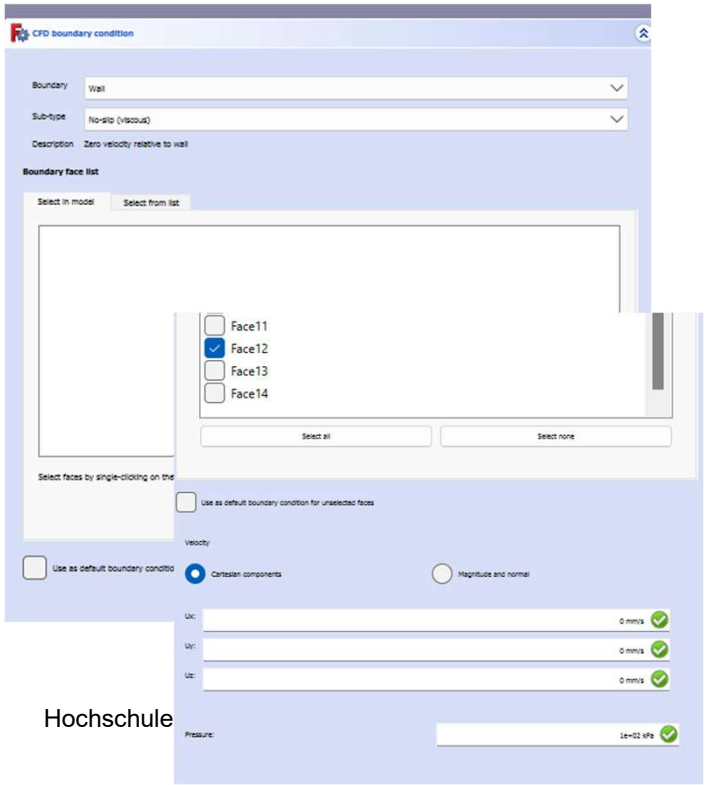
- Nun müssten im Modell folgende Unterfelder eingerichtet sein:
- Zunächst muss das „PhysicsModel“ gewählt werden, hierfür in der Taskleiste auf das entsprechende Symbol gehen. Einfach nichts ändern und OK drücken.



- Anschließend die FluidProperties auswählen. Nichts ändern. Auf OK klicken.



- Anschließend Wand/ Inlet/Outlet definieren



Insgesamt 3x auswählen,

Einmal die Faces (über Select from list) auswählen, die Wand sein sollen;

Dann die Faces, die Inlet sein sollen, und eine Geschwindigkeit vorgeben,

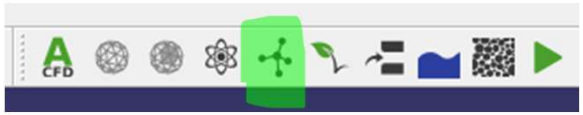
z.B. Uz =-10 mm/s (irgendwas Negatives);

Dann die Faces, die Outlet sein sollen, und einen Druck am Ausgang

vorgeben, z.B. 100000 Pa.

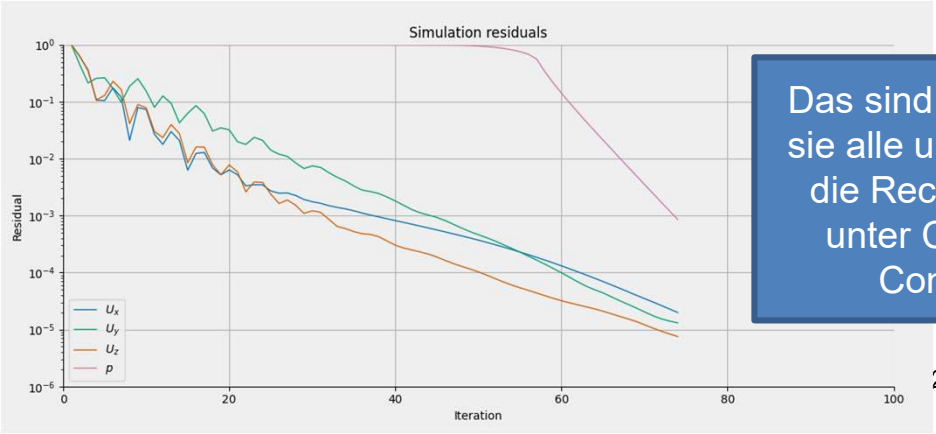
Jeweils mit Ok bestätigen.

- Anschließend die FluidProperties auswählen. Nichts ändern. Auf OK klicken.



- Anschließend die Anfangsbedingungen über „InitialiseFields“ auswählen, z.B. Use Values from Boundary (für Geschwindigkeit Inlet, für Druck Outlet wählen) oder Potential Flow für Geschwindigkeit, Boundary Value vom Outlet für Druck

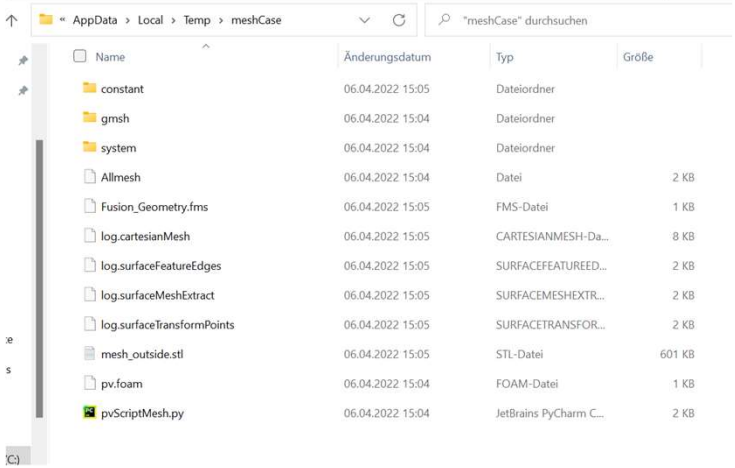
- Nun können Sie den Case rechnen, indem Sie auf den grünen Pfeil drücken. Erst „Write“, dann „Run“ drücken. In der Konsole sollten die Zwischenergebnisse der Rechnung sehen, das ist die Ausgabe von Openfoam.



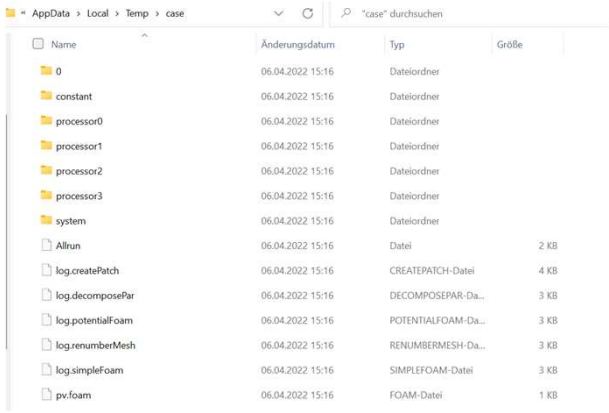
Das sind die Residuen. Wenn sie alle unter 10^{-3} sind, bricht die Rechnung ab. Das wird unter CfdSolver/ Iteration Control eingestellt.

Die Ergebnisse der einzelnen Teilschritte werden entsprechend der Ordner-Struktur von Openfoam in den von Ihnen angegebenen Outputordner geschrieben:

- Das Netz z.B. in den Ordner meshCase:



- Der Case, den man rechnet, in den Ordner case. Hier finden Sie alle Ihre Einstellungen aus der GUI wieder:



Über den „Paraview“-Button kann man sich die Ergebnisse anzeigen lassen: