

OpenFOAM iltamat MEMO

a.mikkonen@iki.fi

September 20, 2023

Contents

1	Intro	3
1.1	OpenFOAM versiot	3
1.2	Oheistyökaluja	4
2	Init ja eka ajo	6
3	Modaukset elbow caseen	7
4	cfmesh	9
5	Ohjelmointi	10
5.1	Hallitsevat yhtälöt	10
5.1.1	Energia	10
5.1.2	Sähkö	10

Chapter 1

Intro

1.1 OpenFOAM versiot

openfoam.org <https://doc.cfd.direct/openfoam/user-guide-v11/index>

- Alkuperäinen
- viimeisimmässä versiossa (v11) merkittäviä parannuksia ohjelmisto arkkitehtuuriin. Esim. useamman fysiikan yhdistely pitäisi olla jatkossa helpompaa

openfoam.com <https://www.openfoam.com/documentation/guides/latest/doc/openfoam-guide-physical-modelling.html>

- Forkattu openfoam.org versiosta jokunen vuosi sitten
- Näyttäisi olevan hieman kaupallisempi kuin .org

foam-extend <https://sourceforge.net/projects/foam-extend/>

- forkattu .org versiosta kymmenkunta vuotta sitten
- Yleisesti ottaen pahasti jäljessä kehitystä mutta muutamia hyviä ominaisuuksia
 - Kytketty matriisi ratkaisin, esim. solidin ja fluidin T kenttä yhdellä matriisin käännöllä
 - Pyöriviin koneisiin .org versiosta puuttuvia reunaehdoja yms.

Paljon laadukkaita kirjastoja esim. <https://www.blastfoam.org/examples/blastfoam-theory> joka keskittyy räjähteiden (TNT, C4,...) vaikutusten mallintamiseen. Sisältää myös edistyneempiä aika-askellus algoritmeja, kuorman tasauksen eri ytimille dynaamisessa verkon tihennyksessä, yms. räjähdde mallinnukseen tärkeää.

1.2 Oheistyökaluja

paraView (paraFoam) paraView OpenFOAMin standardi jälkikäsitelijä. paraFoam on pieni lisäosa joka hieman helpottaa käyttöä. Tarvitsee erillisen asennuksen

cfMesh OpenFOAMista erillinen verkottaja, joka muuttui maksulliseksi jokunen vuosi sitten. Vanha opensource versio on vielä saatavilla ja se on paketoituna .org versioon. Helppo käyttää. Ilmaisversiossa puutteita rajakerrosten luonnissa.

muut verkottaja Jos on tottunut tekemään verkon esim ANSYS työkaluin, verkon saa mitä todennäköisemmin konvertoitua openfoamille. fluentMeshToFoam (openfoam utility) pitäisi tehdä juuri sen mitä nimi sanoo mutta minulla ei ole kokemusta.

Algorithm 1.1 Asennus

```
Mukaillen ohjeita täältä
https://develop.openfoam.com/Development/openfoam/-/wikis/precompiled/debian
# Add the repository
curl https://dl.openfoam.com/add-debian-repo.sh | sudo bash

# Update the repository information
sudo apt update

# Install preferred package. Eg,
sudo apt install openfoam2306-default

# install paraview (post processing)
# (suositeltavaa olisi käyttää uusita paraview binääriä (https://www.paraview.org)
# mutta tämä on helpompaa)
sudo apt install paraviewopenfoam56

# add to .bash_aliases
alias of2306='source /usr/lib/openfoam/openfoam2306/etc/bashrc'
# and to .bashrc
of2306

#intialization
mkdir -p $FOAMRUN
run
cp -rf $FOAM_TUTORIALS .

# t
cd tutorials/incompressible/icoFoam/cavity
./Allrun
cd cavity
```

Chapter 2

Init ja eka ajo

Algorithm 2.1 Asennus

```
#initialization
mkdir -p $FOAM_RUN
run
cp -rf $FOAM_TUTORIALS .

# first run
cd tutorials/incompressible/icoFoam/elbow/
./Allrun
cd cavity
paraFoam
```

Chapter 3

Modaukset elbow caseen

1. siivous ja siirto
2. checkMesh
3. uniformFixedValue (function1)

Algorithm 3.1 Reunaehto vaihtoehtoja triviaalista edistyneeseen.

```
velocity-inlet-6
{
  // original
  //      type      fixedValue;
  //      value      uniform (0 3 0);

  type uniformFixedValue;

  // option 1
  //      uniformValue constant (0 3 0);

  // option 2
  //      uniformValue table
  //      (
  //          (0.0 (0 0 0))
  //          (10 (0 3 0))
  //          (20 (0 0 0))
  //          (30 (0 3 0))
  //          (40 (0 0 0))
  //          (60 (0 3 0))
  //      );

  // option 3
  uniformValue coded;
  name myExpression; // Name of generated PatchFunction1

  // Step function
  code
  #{
      const scalar start(10);
      const scalar t = time().value();

      const polyPatch& pp = this->patch();
      Pout<< "** Patch size:" << pp.size() << endl;

      if (t < start) {
          return tmp<vectorField>::New(pp.size(), vector(0, 0, 0));
      } else {
          return tmp<vectorField>::New(pp.size(), vector(0, 3, 0));
      }
  };
}
```

Chapter 4

cfmesh

Algorithm 4.1 cfmesh

```
surfaceFeatureEdges -angle 15 domain.stl domain.fms  
cartesianMesh
```

Chapter 5

Ohjelmointi

5.1 Hallitsevat yhtälöt

5.1.1 Energia

$$-\nabla \cdot k_s \nabla T = q_{tot} \quad (5.1)$$

$$q_{tot} = p - h \text{AoV} (T - T_{ref}) \quad (5.2)$$

$$\text{AoV} = \frac{A}{\text{Vol}} = \frac{1}{t} \quad (5.3)$$

5.1.2 Sähkö

The electric potential can be solved from the well-known Laplace's equation

$$\nabla \cdot \sigma \nabla V = 0 \quad (5.4)$$

where V is electric potential (V). The electric conductivity is σ . The electric current density $\mathbf{J}$ is solved from

$$\mathbf{J} = -\sigma \nabla V \quad (5.5)$$

Finally, the volumetric electric heat power p is

$$p = \frac{\partial P}{\partial V} = \mathbf{J} \cdot \mathbf{J} / \sigma = \frac{|\mathbf{J}|^2}{\sigma} \quad (5.6)$$