forcesConventional - Documentation

Callum Bruce

November 16, 2021

Abstract

Documentation for forcesConventional, an OpenFOAM functionObject for calculating forces and moments acting on solid or porous bodies using a *conventional* approach. The forces and moments calculation for solid bodies is valid for the icoFoam solver. The force and moments calculation for porous bodies is valid for the porousIcoFoam solver.

Chapter 1

Governing equations

1.1 Navier Stokes

The unsteady, incompressible Navier Stokes continuity and momentum equations are given by:

$$\nabla \cdot \mathbf{u} = 0 \tag{1.1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p^* + \nu \nabla^2 \mathbf{u} + S_i \tag{1.2}$$

1.2 Darcy Brinkman Forchheimer

The unsteady, incompressible Darcy Brinkman Forchheimer (DBF) continuity and momentum equations are given by:

$$\nabla \cdot \mathbf{u} = 0 \tag{1.3}$$

$$\frac{1}{\phi}\frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\phi^2}(\mathbf{u} \cdot \nabla \mathbf{u}) = -\nabla p^* + \frac{\nu}{\phi}\nabla^2 \mathbf{u} - \frac{\nu}{K}\mathbf{u} - \frac{c_F}{\sqrt{K}}|\mathbf{u}|\mathbf{u} \tag{1.4}$$

see *porousIcoFoam_docs.pdf* for further details on the DBF governing equations.

Chapter 2

Forces and moments calculation

2.1 Force and moment components

2.1.1 Pressure component

The normal pressure force is calculated as a surface integral on a solid boundary or at the clear fluid - porous media interface and is written in discrete form as:

$$d\mathbf{F}_p|_i = \rho \mathbf{A}_i (p_i^* - p_{ref}^*) \tag{2.1}$$

$$\mathbf{F}_p = \sum_i d\mathbf{F}_p|_i \tag{2.2}$$

where $\mathbf{A}_i$ is the ith boundary face area vector. $\mathbf{A}_i$ is positive pointing into the body. The normal pressure moment is given, in discrete form, as:

$$d\mathbf{M}_p|_i = \mathbf{r}_i \times d\mathbf{F}_p|_i \tag{2.3}$$

$$\mathbf{M}_p = \sum_i d\mathbf{M}_p|_i \tag{2.4}$$

where $\mathbf{r}_i$ is the ith boundary face position vector.

2.1.2 Viscous component

The viscous force is calculated as a surface integral on a solid boundary or at the clear fluid - porous media interface and is written in discrete form as:

$$d\mathbf{F}_v|_i = \mathbf{A}_i \cdot \boldsymbol{\tau}_i \tag{2.5}$$

$$\mathbf{F}_v = \sum_i d\mathbf{F}_v|_i \tag{2.6}$$

where $\boldsymbol{\tau}_i$ is the viscous stress tensor on the boundary face. The viscous moment is given, in discrete form, as:

$$d\mathbf{M}_v|_i = \mathbf{r}_i \times d\mathbf{F}_v|_i \tag{2.7}$$

$$\mathbf{M}_v = \sum_i d\mathbf{M}_v|_i \tag{2.8}$$

2.1.3 Porous (Darcy) component

The Darcy force is calculated as a volume integral inside a porous body. Considering the third term on the right hand side (RHS) of equation 1.4 the Darcy force contribution is given, in discrete form, as:

$$d\mathbf{F}_d|_i = \rho \mathbf{V}_i(\frac{\nu}{K}\mathbf{u}) \tag{2.9}$$

$$\mathbf{F}_d = \sum_i d\mathbf{F}_d|_i \tag{2.10}$$

where $\mathbf{V}_i$ is the ith volume element (inside the porous media). The Darcy moment is calculated, in discrete form, as:

$$d\mathbf{M}_d|_i = \mathbf{r}_i \times d\mathbf{F}_d|_i \tag{2.11}$$

$$\mathbf{M}_d = \sum_i d\mathbf{M}_d|_i \tag{2.12}$$

2.1.4 Porous (Forchheimer) component

The Forchheimer force is calculated as a volume integral inside a porous body. Considering the fourth term on the RHS of equation 1.4 the Forchheimer force is given, in discrete form, as:

$$d\mathbf{F}_f|_i = \rho \mathbf{V}_i(\frac{c_F}{\sqrt{K}}|\mathbf{u}|\mathbf{u}) \tag{2.13}$$

$$\mathbf{F}_f = \sum_i d\mathbf{F}_f|_i \tag{2.14}$$

The Forchheimer moment is calculated, indiscrete form, as:

$$d\mathbf{M}_f|_i = \mathbf{r}_i \times d\mathbf{F}_f|_i \tag{2.15}$$

$$\mathbf{M}_f = \sum_i d\mathbf{M}_f|_i \tag{2.16}$$

2.2 Solid body forces and moments

Total force acting on a solid body is given as the sum of the pressure (2.2) and viscous (2.6) components:

$$\mathbf{F}_t = \mathbf{F}_p + \mathbf{F}_v \tag{2.17}$$

2.3 Porous body forces and moments

Consider the forces acting on the fluid outside the porous body which is given as the sum of the pressure (2.2) and viscous (2.6) acting at the clear fluid - porous media interface where $\mathbf{A}_i$ is this time positive pointing away from the porous body. It follows that the force acting on the porous media must be equal and opposite to the force experienced by the fluid due to the presence of the porous media. Total force acting on a porous body is given as the sum of the pressure (2.2) and viscous (2.6) components as given in equation 2.17.

The difference between the momentum flux in and out of the porous body is given as:

$$\dot{\mathbf{P}}_{\text{out}} - \dot{\mathbf{P}}_{\text{in}} = \mathbf{F}_t - (\mathbf{F}_d + \mathbf{F}_f) \tag{2.18}$$

2.4 forcesConventional functionObject

An OpenFOAM functionObject, forcesConventional, has been written to calculate the forces and moments acting on solid and porous bodies. This functionObject is intended for use with either the icoFoam or porousIcoFoam solver. Pressure, viscous and porous (Darcy and Forchheimer) force contributions are calculated using the definitions given in §2.1.1, 2.1.2, 2.1.3, 2.1.4.

To use the forcesConventional functionObject the following dictionary must be added to ./system/controlDict/functions:

```
forces_conventional
{
// Mandatory entries
type forcesConventional;
libs ("libmyForcesConventionalFunctionObject.so");
// Patches
patches (interface1);
// rho
rho 1.0;

// Optional entries
// Field names
p p;
U U;
K_ K_;
// pRef
pRef 0.0;
// Calculate for a porous body?
porosity true;
porousZone porousMedia;
}
```

Property	Type	Description
patches	List	List of patch names used to calculate pressure and viscous contributions - patch surface normals must point inside porous body
rho	Float	Density value to use in forces calculation
p	Name	Pressure field (volScalarField) name
U	Name	Velocity field (volVectorField) name
K_	Name	Permeability field (volScalarField) name
pRef	Float	reference pressure to use in forces calculation
porosity	Bool	Include porous force contributions true/false (default is false)
porousZone	Name	Name of the cellZone used to calculate porous contributions

Table 2.1: forcesConventional functionObject properties

Properties in the forcesConventional dictionary entry are documented in table 2.1. Source code for the forcesConventional functionObject can be downloaded from: https://github.com/c-bruce/forcesConventional.

2.4.1   forcesConventional verification

To verify the implementation of the forcesConventional functionObject a two-dimensional porous square cylinder case was set up and run using: a) pisoFoam solver using laminar turbulence model (see pisoFoam) with OpenFOAM's built-in Darcy-Forchheimer porous media model (see DarcyForchheimer) and built-in forces functionObject; and b) custom porousIcoFoam solver and custom forcesConventional functionObject. These approaches will herein be refered to as *built-in* and *custom* respectively. The aim of this verification case is to demonstrate that the forcesConventional function object has been correctly implemented. Note that in the built-in approach the porous media model only modifies the Navier Stokes equations by adding sink terms to the RHS of the governing equation (via fvOptions in OpenFOAM) and does not include the porosity terms that appear in equation 1.4 which modify the local acceleration, convective and viscous terms in the equation. Hence to match this set up in the custom approach porosity is set to unity everywhere. The porous square cylinder is centered on the origin and has edge length, $L = 1\ m$. A fixed velocity inlet condition ($u_\infty = 1\ ms^{-1}$) is applied $2L$ upstream of the origin, a fixed pressure outlet condition ($p^* = 0\ Pakg^{-1}m^3$) is applied $6L$ downstream of the origin and, slip boundary conditions are applied at $\pm 2L$ spanwise. Kinematic viscosity, ν, is set to $0.033333\ m^2s^{-1}$. This corresponds to a Reynolds number, $Re = 30$. Permeability, K, is set to $1 \times 10^{-4}\ m^2$. The Darcy number, Da is defined as $Da = K/L^2$ therefore, $Da = 1 \times 10^{-4}$. The form drag constant, c_F, is set to zero neglecting the Forchheimer term.

| C_D Contribution	Built-in	Custom	Difference (%)
Pressure	2.8021	2.7880	-0.50
Viscous	0.5495	0.5553	1.06
Porous (Darcy)	3.5176	3.5158	-0.05

Table 2.2: forcesConventional verification results.

Figure 2.1 shows the development, over time, of individual force contributions in non-dimensional form as the drag coefficient, C_D, for the *built-in* and *custom* approaches described above. The solution becomes steady after $\sim 10s$. Table 2.2 shows the steady state values for the *built-in* and *custom* approaches. Results in figure 2.1 and table 2.2 show that forces calculated using the *built-in* and *custom* approaches do not agree exactly. The largest percentage difference appears in the viscous contribution ($\sim 1\%$) while the pressure (0.5%) and porous (0.05%) contributions display smaller differences. The two approaches solve the governing equations in distinct ways: in the custom approach the porous terms are included directly in the porousIcoFoam `UEqu` while the built-in approach solves the Navier Stokes equations and includes the porous terms at run time using `fvOptions`. The size of the difference in results is small considering the differences in approaches outlined above and gives confidence that the forcesConventional functionObject has been correctly implemented.
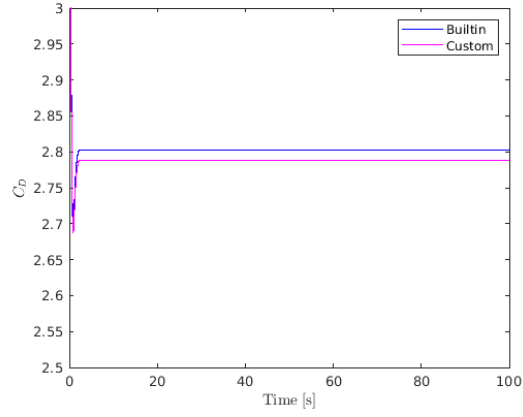
(a) Pressure C_D contribution
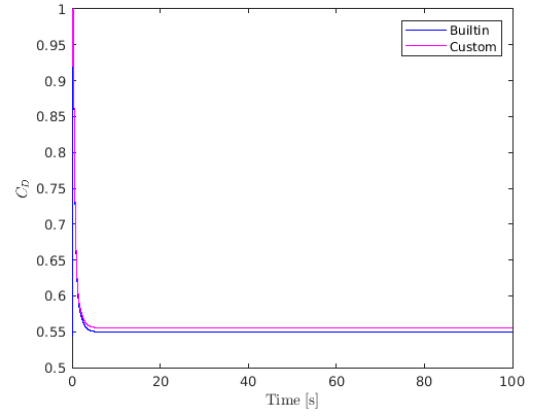
(b) Viscous C_D contribution
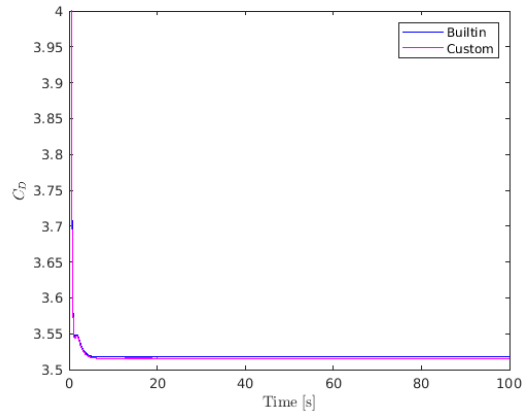
(c) Porous (Darcy) C_D contribution

Figure 2.1: Contributions to total C_D for the two-dimensional porous square cylinder verification case calculated using the built-in and custom approaches.

Bibliography