Von Karman Viscous Pump

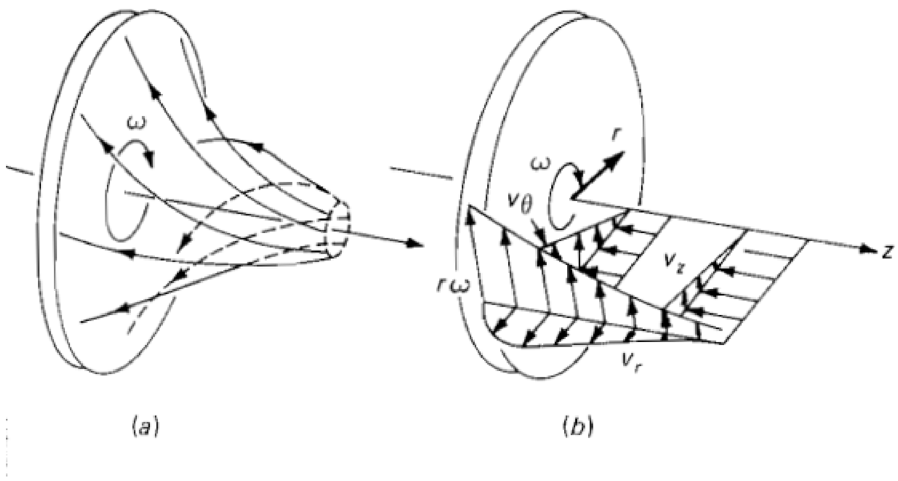
Andreas Slyngstad

26. februar 2016

Introduction of problem

Consider an infinite rotating disk placed at z=0 under a newtonian viscous fluid initially at rest, viscous forces will set up a rotating velocity field in the fluid. von Karman(1) showed that the steady state flow of this problem could be reduced to a set of ordinary differential equations. He solved them by approximate integration method, which will be used as reference for the CFD results in this test.

Spinning circle



Theory

Let the velocity vector components be defined as $\mathbf{v} = (v_r, v_\theta, v_z)$ By the assumption of radial symmetry, we expect our components to be independent of the angle θ. The continuity equation for a polar coordinate system in accordance with our assumption is defined by

$$\nabla \cdot \mathbf{v} = \frac{1}{r}\frac{\partial(rv_r)}{\partial r} + \frac{\partial v_z}{\partial z} = 0 \tag{1}$$

Moving on to the equation of momentum, the familiar Navier-Stokes equation yields.

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\frac{\nabla p}{\rho} + \frac{\mu}{\rho}\nabla^2 \mathbf{v} \tag{2}$$

Considering steady-flow we end up with the following component equations.

$$r: \qquad v_r\frac{\partial v_r}{\partial r} - \frac{v_\theta^2}{r} + v_z\frac{\partial v_r}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial r} + \frac{\mu}{\rho}(\frac{\partial^2 v_r}{\partial r^2} + \frac{1}{r}\frac{\partial v_r}{\partial r} + \frac{\partial^2 v_r}{\partial z^2} - \frac{v_r}{r^2}) \tag{3}$$

$$\theta: \qquad v_r\frac{\partial v_\theta}{\partial r} + v_z\frac{\partial v_\theta}{\partial z} + \frac{1}{r}v_\theta v_r = \frac{\mu}{\rho}(\frac{\partial^2 v_\theta}{\partial r^2} + \frac{1}{r}\frac{\partial v_\theta}{\partial r} + \frac{\partial^2 v_\theta}{\partial z^2} - \frac{v_\theta}{r^2}) \tag{4}$$

$$z: \qquad v_r\frac{\partial v_z}{\partial r} + \frac{\partial v_z}{\partial z} = \frac{1}{\rho}\frac{\partial p}{\partial z} + \frac{\mu}{\rho}(\frac{\partial^2 v_z}{\partial r^2} + \frac{1}{r}\frac{\partial v_z}{\partial r} + \frac{\partial^2 v_z}{\partial z^2}) \tag{5}$$

For our problem we define the following boundary conditions.

- On disk surface
$$v_r = 0, v_\theta = r\Omega, v_z = 0$$
- Disk circumference
$$\frac{\partial}{\partial r}(\frac{v_r}{r}) = 0, \frac{\partial}{\partial r}(\frac{v_\theta}{r}) = 0, \frac{\partial v_z}{\partial r} = 0$$
- End of integration domain
$$v_r = 0, v_\theta = 0, \frac{\partial v_z}{\partial z} = 0$$

Introducing von Karman's substitutions of the velocity components,

$$v_r = r\omega F(\zeta) \quad v_\theta = r\omega G(\zeta) \quad v_z = (\nu\omega)^{\frac{1}{2}} H(\zeta)$$
$$p = \rho\nu\omega, \quad \zeta = (\omega/\nu)^{\frac{1}{2}} z$$

we can rewrite (3-5) as a system of ODE's as follows

$$F^2 - G^2 + HF^{'} = F^{''} \tag{6}$$
$$2FG + HG^{'} = G^{''} \tag{7}$$
$$2F + H^{'} = 0 \tag{8}$$
$$P^{'} + 2F^{'} = -HH^{'} \tag{9}$$
$$\tag{10}$$

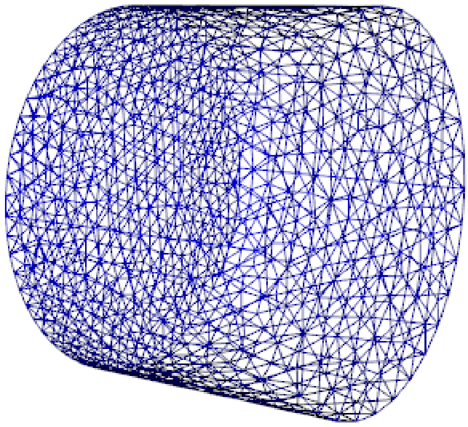
The corresponding boundary conditions yields

- $F = 0, \quad G = 1, \quad H = -a \quad$ at $\zeta = 0$
- $F = 0, \quad G = 0 \quad$ at $\zeta = \infty$

Solving the problem

For the presented problem, the Navier-Stokes solver Oasis based on the FEniCS software will be used to solve the PDE system using finite element method.
For the computation we have to construct a mesh to do our calculations upon. In this problem we will use Gmsh, a free 3D finite element grid generator. The problem will be solved on a pipe with r = 1 and height z = 3. The pipe will be constructed with denser elements at the spinning boundary, as seen in the presented figure.

Pipe mesh



Continuing on the reference solution, we derive Karman's method of approximate solution. Firstly we want to observe how the differential equations behave at the limit $\zeta \to \infty$. Assuming H approaches a limit $-c$ as while $F, G \to$ as $\zeta \to \infty$, we end up with the following relations for large ζ.

$$-cG^{'} = G^{''} \quad - cF^{'} = F'' \tag{11}$$

$$F = Ae^{-c\zeta}, \quad G = Be^{-c\zeta}, \quad H = -c + \frac{2A}{c}e^{-c\zeta} \tag{12}$$

As we can see, F and G approaches 0 exponentially, and we can assume they are 0 for some value of ζ which we will call ζ_0. This will be exploited in the following integration scheme.

Starting by integrating equation (6)-(7)from 0 to ∞, and using relation (8).

$$\int_0^\infty HF^{'}d\zeta = \left[HF\right]_0^\infty - \int_0^\infty H^{'}Fd\zeta = 2\int_0^\infty F^2 d\zeta$$
$$\int_0^\infty HG^{'}d\zeta = \left[HG\right]_0^\infty - \int_0^\infty H^{'}Gd\zeta = 2\int_0^\infty FG d\zeta$$

Combining these results with (12), we end up with the following result

$$-F^{'}(0) = \int_0^\infty (3F^2 - G^2)d\zeta \tag{13}$$

$$-G^{'}(0) = 4\int_0^\infty FGd\zeta \tag{14}$$

As Karman, we assume that due to the expoential growth that F and G are zero for values of ζ greater than ζ_0. As a result

$$F(\zeta_0) = 0, \quad F^{'}(\zeta_0) = 0, \quad G(\zeta_0) = 0, \quad G'(\zeta_0) = 0 \tag{15}$$

We can now also find $F''(0)$ and $G^{''}(0)$ by setting $\zeta = 0$ in the system of ODE's

$$F^{''}(0) = -1 \quad G^{''}(0) = 0 \tag{16}$$

Now if we let $F^{'}(0)$ is some constant a, the following functions fulfills equation (15)-(16) and the boundary conditions.

$$F = (1 - \frac{\zeta}{\zeta_0})^2 \ \left(a\zeta + (\frac{2a}{\zeta_0})\zeta^2\right) \tag{17}$$

$$G = (1 - \frac{\zeta}{\zeta_0})^2 \ (1 + \frac{\zeta}{2\zeta_0}) \tag{18}$$

This yields $G^{'}(0) = -\frac{3}{2\zeta_0}$. Inserting (17)-(18) in (13)-(14), we get a system of equations to solve a and ζ_0.

ζ	F	F'	G	G'	H	-P
0.0	0.0	0.51023	1.00	-0.61592	0.0	0.0

Computations

Comments

During programming had some trouble implementing the spinning boundary condition at the end of the pipe. I manage to overcome the trouble, by making a subclass of the FENicS Expression class.

```
class Rotating(Expression):

        def eval(self, value, x):
                r = sqrt(x[0]*x[0] + x[1]*x[1])
                theta = 0
                if x[0] > 0 and x[1] >= 0:
                        theta = atan(x[1]/x[0])
                elif x[0] > 0 and x[1] < 0:
                        theta = atan(x[1]/x[0]) + 2*pi
                elif x[0] < 0:
                        theta = atan(x[1]/x[0]) + pi
                elif x[0] == 0 and x[1] > 0:
                        theta = pi/2
                elif x[0] == 0 and x[1] < 0:
                        theta = 3*pi/2.
                elif x[0] == 0 and x[1] == 0: #To much ?
                        theta = 0

                #Velocity Component
                value[0] =  r * -sin(theta)
                value[1] = r * cos(theta)

        def value_shape(self):
                return(2,)
```