Simulating blood flow in Zebrafish using Stokes' equations

Krister Stræte Karlsen

July 12, 2016

Generating mesh from original image

From the original image a smaller part is extracted(see the green box in Figure 1) and then a mesh is created using *The Vascular Modeling Toolkit*(VMTK).

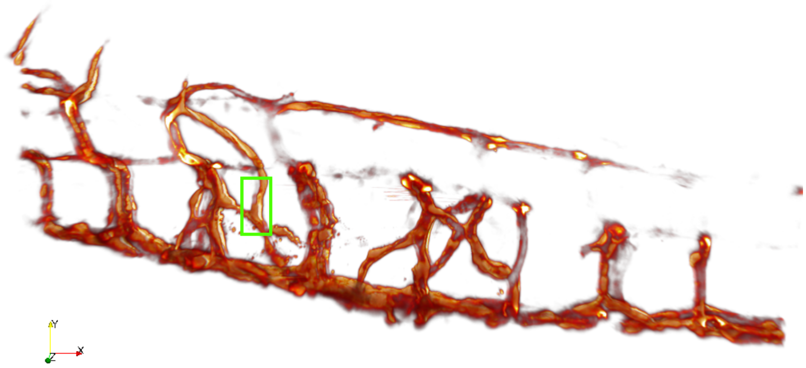


Figure 1: Original image of veins in Zebrafish.

To obtain a finite element mesh from the original image the following scripts/commands have been used:

1) Select a volume of interest(VOI)

```
vmtkimagevoiselector -ifile original.vti -ofile voi.vti
```

2) Segmentation

```
vmtklevelsetsegmentation -ifile voi.vti -ofile levelsets.vti
```

3) Create surface file

```
vmtkmarchingcubes -ifile levelsets.vti -ofile surf.vtp
```

4) Smoothing of surface

```
vmtksurfacesmoothing -ifile surf.vtp -passband 0.1 -iterations 30 -ofile
sm_surf.vtp
```

5) Clip surface

```
vmtksurfaceclipper -ifile sm_surf.vtp -ofile cl_surface.vtp
```

6) Generate mesh

```
vmtkmeshgenerator -ifile cl_surface.vtp -ofile zebramsh.vtu -edgelength
1.0
```

7) Convert to dolfin-format

```
vmtkmeshwriter -ifile zebramsh.vtu -entityidsarray CellEntityIds -ofile
zebra_mesh.xml
```

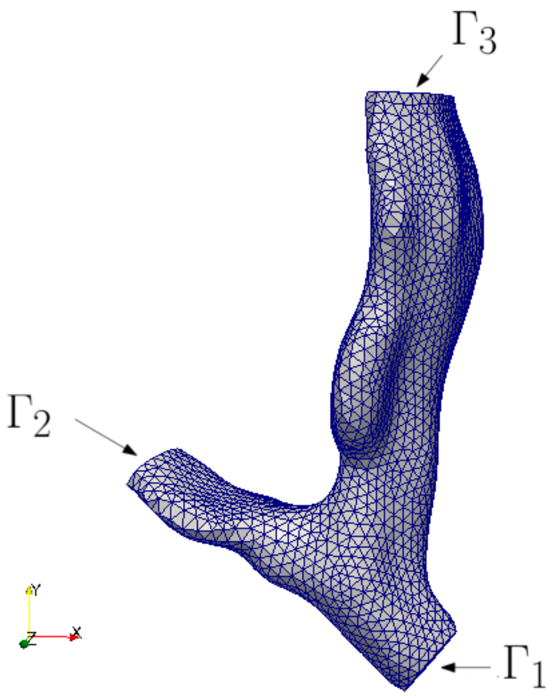


Figure 2: FEM mesh with marking of inlets/outlets.

Mathematical problem formulation

Zebrafish veins are tiny and the blood velocities very low, thus using Stokes flow as an approximation is reasonable.

The governing equations are

$$\left.\begin{aligned} -\Delta u + \nabla p &= f \\ \nabla \cdot u &= 0 \end{aligned}\right\} \quad in \quad \Omega$$

with boundary conditions

$$\begin{aligned} &p = p_1 \quad on \quad \Gamma_1 \\ &p = p_2 \quad on \quad \Gamma_2 \\ &p = p_3 \quad on \quad \Gamma_3 \\ &u = 0 \quad on \quad \partial\Omega \setminus \{\Gamma_1, \Gamma_2, \Gamma_3\}. \end{aligned}$$

To obtain the results featured in Figure 3

$$p_1 > p_2 = p_3 = 0.$$

Stabilized variational form of Stokes' equation

To get away with less degrees of freedom we use a stabilized $P_1 - P_1$ formulation of Stokes' equations, i.e:

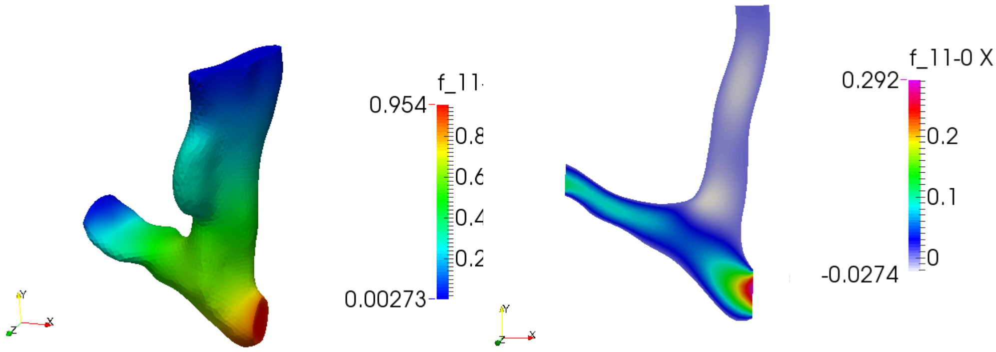
Find $u, p \in W$, $W = V \times Q$ such that

$$a((u,p),(v,q)) = L((v,q)) \quad \forall \quad v, q \in W$$

where

$$\begin{aligned} a((u,p),(v,q)) &= \int_\Omega \nabla u \cdot \nabla v - \nabla \cdot v \; p + \nabla \cdot u \; q + \epsilon \nabla q \cdot \nabla p \, \mathrm{d}x, \\ L((v,q)) &= \int_\Omega f \cdot v \, \mathrm{d}x + \epsilon \nabla q \cdot f \, \mathrm{d}s. \end{aligned}$$

where $\epsilon = \beta h^2$ and β is some number and h is the mesh cell size.



(a) Pressure.

(b) Velocity.

Figure 3: Results from solving the problem with FEniCS.

Results

Implementing and solving the problem above using *FEniCS* we get the results in Figure 3. For code see `p1p1_stokes.py` available at

https://github.com/krikarls/SummerProjectMEK4250