Immersed Boundary Hydrodynamics with Rigid Constraints

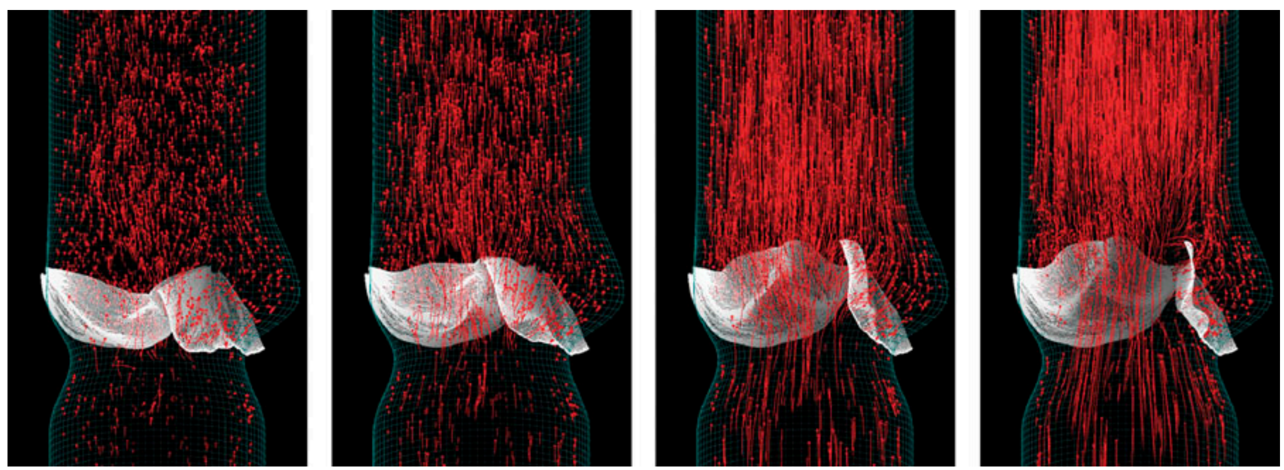
Kevin Silmore

12-10-18

18.337

Immersed Boundary (IB) method

- How to simulate fluid motion in the presence of an immersed flexible/rigid object?
- Avoid need to remesh at every step
- Peskin (1970s) to simulate flow in the heart



Peskin, *Acta Numerica* **11** (2002)
Griffith, *Int. J. Numer. Meth. Biomed. Eng.* **28** (2012).

Equations of motion

Navier-Stokes equation

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\boldsymbol{\nabla}\mathbf{u}\right) = \mu\nabla^2\mathbf{u} - \boldsymbol{\nabla}p + \mathbf{f}$$

$$\boldsymbol{\nabla}\cdot\mathbf{u} = 0$$

$$\mathbf{F}(\boldsymbol{\omega},t) = -\frac{\delta E}{\delta \mathbf{X}}$$

$$\mathbf{f}(\mathbf{x},t) = \int_\Omega \mathbf{F}(\boldsymbol{\omega},t)\delta(\mathbf{x}-\mathbf{X}(\boldsymbol{\omega}))\,\mathrm{d}\boldsymbol{\omega}$$

$$\frac{\partial \mathbf{X}(\boldsymbol{\omega},t)}{\partial t} = \int_{\mathcal{G}} \mathbf{u}(\mathbf{x},t)\delta(\mathbf{x}-\mathbf{X}(\boldsymbol{\omega}))\,\mathrm{d}\mathbf{x}$$

$$\rho(\mathbf{x},t) = \rho_0 + \int_\Omega \tilde{M}(\boldsymbol{\omega})\delta(\mathbf{x}-\mathbf{X}(\boldsymbol{\omega}))\,\mathrm{d}\boldsymbol{\omega}$$

IB forces and transformations

$\Omega =$ Lagrangian space

$\mathcal{G} =$ Eulerian space

Discretized equations of motion

$$\rho_0\left(\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}+\mathbf{N}(\mathbf{u})\right)=\mu L_h\mathbf{u}-\mathbf{D}_h p+\mathbf{f}$$

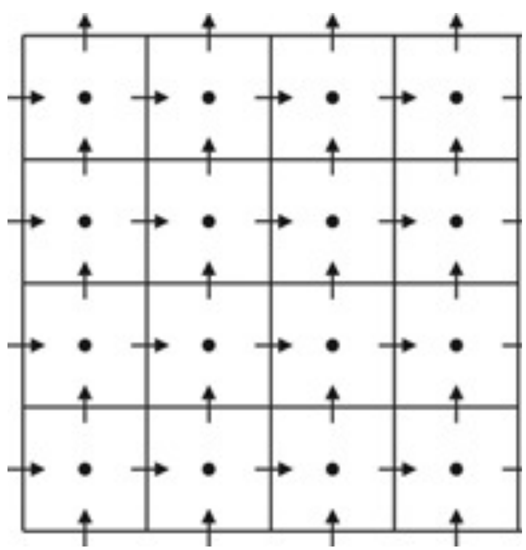
$$\mathbf{D}_h\cdot\mathbf{u}=0$$

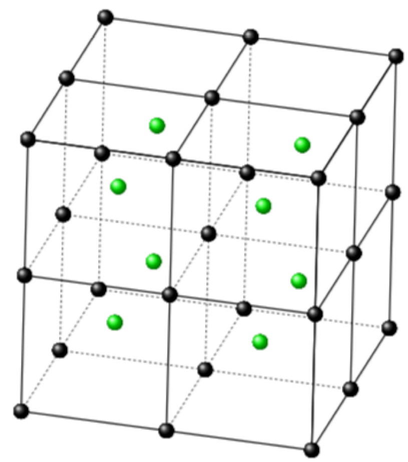
$$\mathbf{F}\Delta q\Delta r\Delta s=-\frac{\delta E}{\delta\mathbf{X}}$$

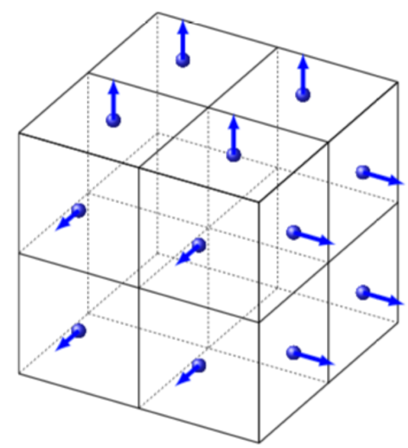
$$\mathbf{f}=\sum_{\boldsymbol{\omega}\in\Omega}\mathbf{F}(\boldsymbol{\omega},t)\delta_h(\mathbf{x}-\mathbf{X}(\boldsymbol{\omega}))\Delta q\Delta r\Delta s$$

$$\frac{\mathrm{d}\mathbf{X}}{\mathrm{d}t}=\sum_{\mathbf{x}\in\mathcal{G}}\mathbf{u}(\mathbf{x},t)\delta_h(\mathbf{x}-\mathbf{X}(\boldsymbol{\omega}))h^3$$

Choice of discretization grid

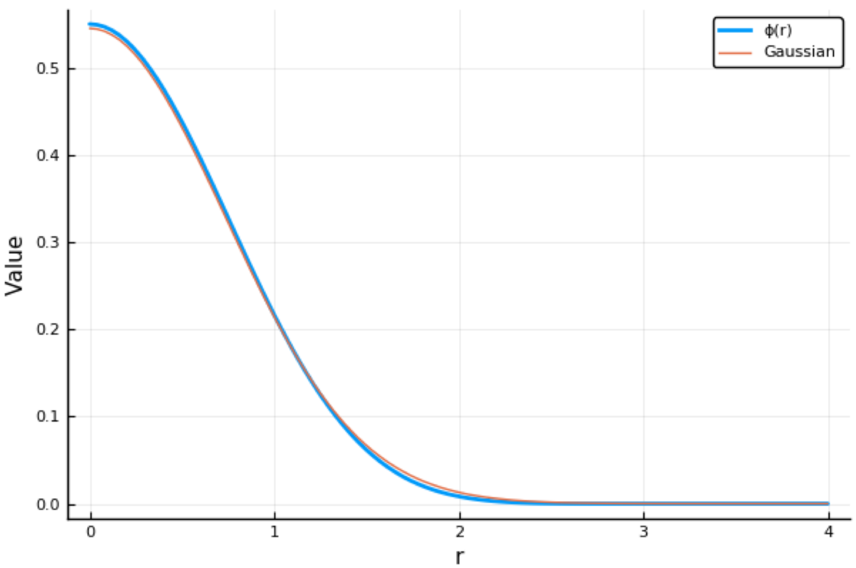


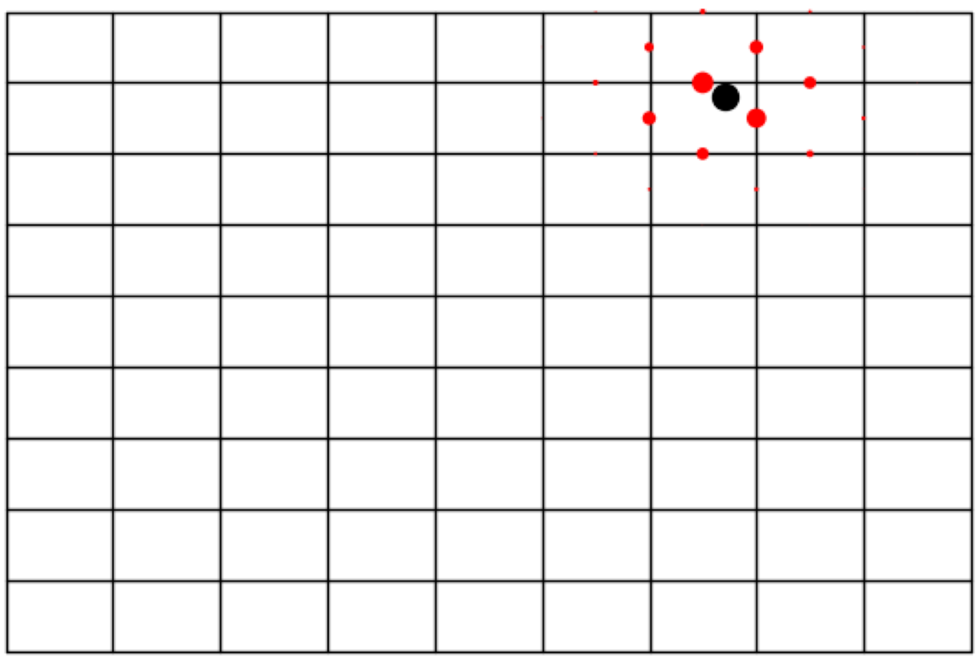




Bao et al., *J. Comp. Phys.* **347** (2017).
Griffith et al., *J. Comp. Phys.* **208** (2005).
Griffith, *Int. J. Numer. Meth. Biomed. Eng.* **28** (2012).

Kernel = regularized δ function





Linearization and Picard Iteration

Nonlinear!

$$\rho\left(\frac{\mathbf{u}^{(t+1)}-\mathbf{u}^{(t)}}{\Delta t}+\mathbf{u}^{(t+1)}\cdot\boldsymbol{\nabla}\mathbf{u}^{(t+1)}\right)=\mu\nabla^2\mathbf{u}^{(t+1)}-\boldsymbol{\nabla}p^{(t+1)}+\mathbf{f}^{(t)}$$

$$\boldsymbol{\nabla}\cdot\mathbf{u}^{(t+1)}=0$$

$$\Downarrow$$

$$\rho\left(\frac{\mathbf{u}^{(t+1)}-\mathbf{u}^{(t)}}{\Delta t}+\mathbf{v}\cdot\boldsymbol{\nabla}\mathbf{u}^{(t+1)}\right)=\mu\nabla^2\mathbf{u}^{(t+1)}-\boldsymbol{\nabla}p^{(t+1)}+\mathbf{f}^{(t)}$$

$$\boldsymbol{\nabla}\cdot\mathbf{u}^{(t+1)}=0$$

Benzi et al., *Acta Numerica* **14** (2005).

Rigid constraints lead to saddle point problem

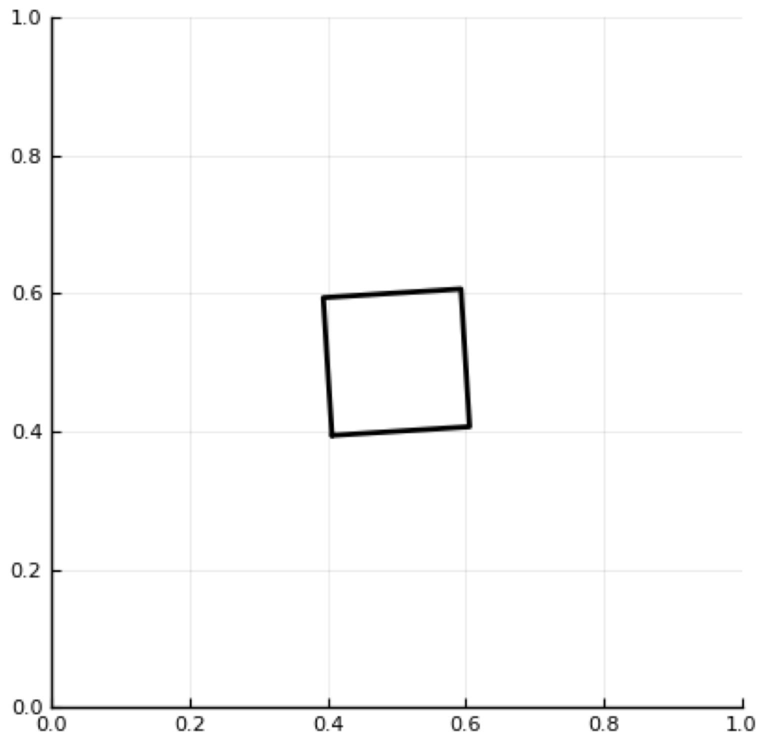
- Specify rigid body motion

$$\begin{bmatrix} \mathbf{A} & \mathbf{G} & -\mathbf{S}_{L\to E} \\ \mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}_{E\to L} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{(t+1)} \\ \mathbf{p} \\ \mathbf{F} \end{bmatrix} = \begin{bmatrix} \rho\mathbf{u}^{(t)}/\Delta t \\ \mathbf{0} \\ \mathrm{U} + \mathbf{\Omega} \times (\mathbf{X} - \bar{\mathbf{X}}) \end{bmatrix}$$

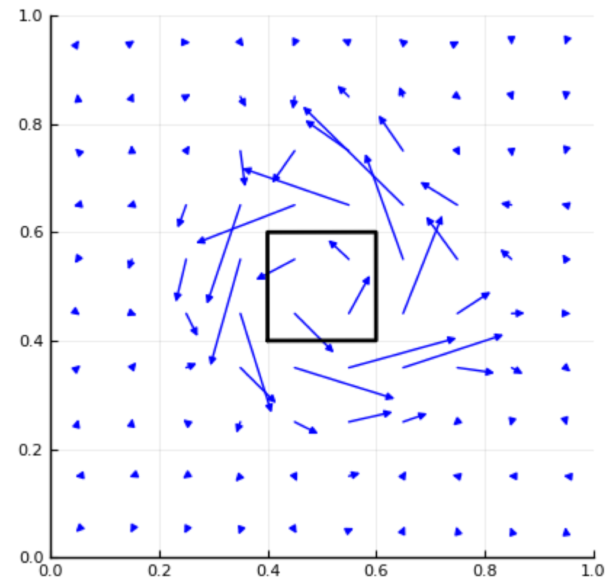
- Can solve iteratively with a Krylov method (IDR(s))

Kallemov et al., *Comm. in Appl. Math. and Comp. Science*, **11** (2016).

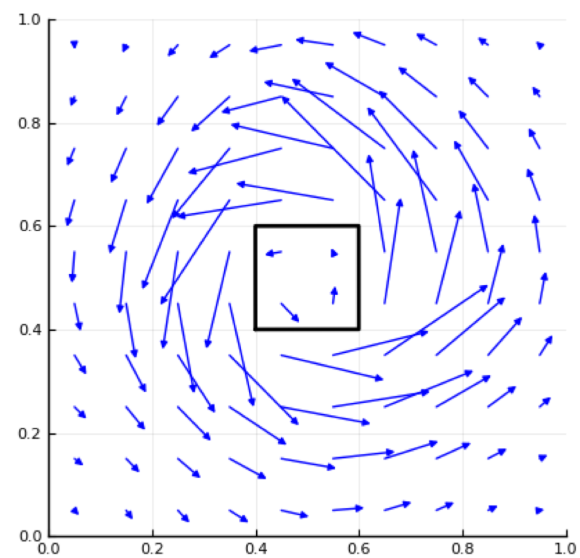
$$\mathbf{U} + \mathbf{\Omega} \times (\mathbf{X} - \bar{\mathbf{X}})$$



$\mu/\rho = 0.1$



$\mu/\rho = \infty$



Packages used:

- Plots
- LinearMaps
- IterativeSolvers