Firedrake:

Decomposing Function Spaces

Koki Sagiyama[1]

work with

Lawrence Mitchell[2]
David A. Ham[1]

2nd PRISM Workshop
30/09/2020

[1]Department of Mathematics, Imperial College London
[2]Department of Computer Science, Durham University

Imperial College London

Firedrake

Firedrake:

Solve BVPs written in weak form using finite element methods.

[https://www.firedrakeproject.org/]

Boundary conditions:

DirichletBC class.

Current work:

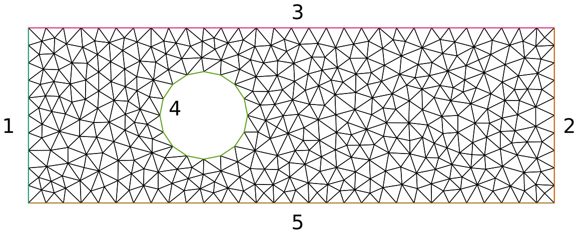
Decompose test function $\boldsymbol{v} \in V$:

$$\boldsymbol{v} = \boldsymbol{v}_0 + \boldsymbol{v}_1, \quad v_0 \in V_0, v_1 \in V_1.$$

$\boldsymbol{v}_0$: test domain equation (e.g., Stokes equation).

$\boldsymbol{v}_1$: test boundary condition (e.g., $\boldsymbol{u} \cdot \boldsymbol{n} = 0$).

Stokes Equation



Imperial College London

Stokes Equation

Solve for $(\boldsymbol{u}, p)$:

Domain:

$$\int (\nu \nabla \boldsymbol{u} : \nabla \boldsymbol{v} - p \nabla \cdot \boldsymbol{v} - \nabla \cdot \boldsymbol{u} q) \mathrm{d}\Omega = 0, \qquad \forall (\boldsymbol{v}, q) \in V.$$

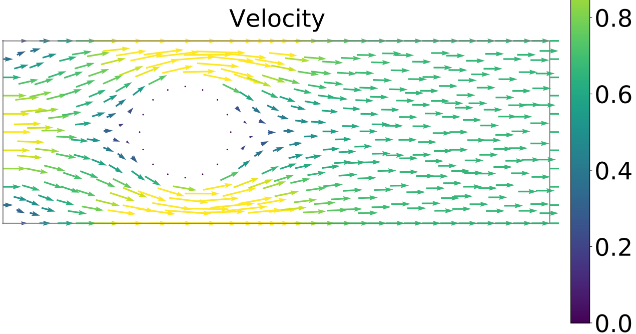
Boundary:

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{u}_{\text{inflow}}, && \text{on } \Gamma_1, \\ \nabla \boldsymbol{u} \cdot \boldsymbol{n} &= \mathbf{0}, && \text{on } \Gamma_2, \\ \boldsymbol{u} &= \mathbf{0}, && \text{on } \Gamma_4, \\ \boldsymbol{u} \cdot \boldsymbol{n} &= u_y = 0, && \text{on } \Gamma_{(3,5)}. \end{aligned}$$

Stokes Equation

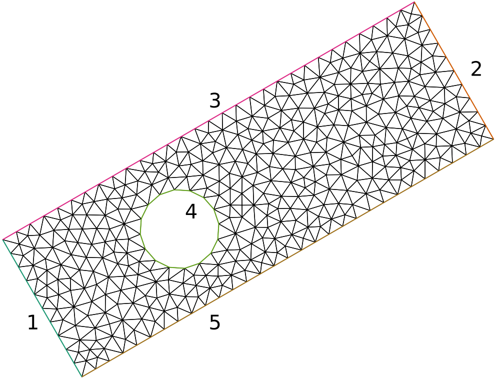
```
Vel = VectorFunctionSpace(mesh, "CG", 2)
Q = FunctionSpace(mesh, "CG", 1)
V = Vel * Q

vq = TestFunction(V)
up = Function(V)
v, q = split(vq)
u, p = split(up)

nu = 1
u_inflow = ...

bcs = [DirichletBC(V.sub(0), u_inflow, 1),
       DirichletBC(V.sub(0).sub(1), 0, (3, 5)),
       DirichletBC(V.sub(0), (0, 0), (4, ))]

F = nu * inner(grad(u), grad(v)) * dx \
  - inner(p, div(v)) * dx \
  - inner(div(u), q) * dx

solve(F == 0, up, bcs=bcs)
```

Stokes Equation



Imperial College London

Stokes Equation



Stokes Equation

Solve for $(\boldsymbol{u}, p)$:

Domain:

$$\int (\nu \nabla \boldsymbol{u} : \nabla \boldsymbol{v} - p \nabla \cdot \boldsymbol{v} - \nabla \cdot \boldsymbol{u} q) \mathrm{d}\Omega = 0,$$

$$\forall (\boldsymbol{v}, q) \in V_0.$$

Boundary:

$$\boldsymbol{u} = \boldsymbol{u}_{\text{inflow}}, \qquad \text{on } \Gamma_1,$$

$$\nabla \boldsymbol{u} \cdot \boldsymbol{n} = \boldsymbol{0}, \qquad \text{on } \Gamma_2,$$

$$\boldsymbol{u} = \boldsymbol{0}, \qquad \text{on } \Gamma_4.$$

$$\int (\boldsymbol{u} \cdot \boldsymbol{n} - 0)(\boldsymbol{v} \cdot \boldsymbol{n}) \mathrm{d}\Gamma_{(3,5)} = 0, \qquad \forall (\boldsymbol{v}, q) \in V_1.$$

$$V_0 = \{ v = (\boldsymbol{u}, p) \in V : \boldsymbol{u} \cdot \boldsymbol{n} = 0 \text{ on } \Gamma_{(3,5)}. \}$$

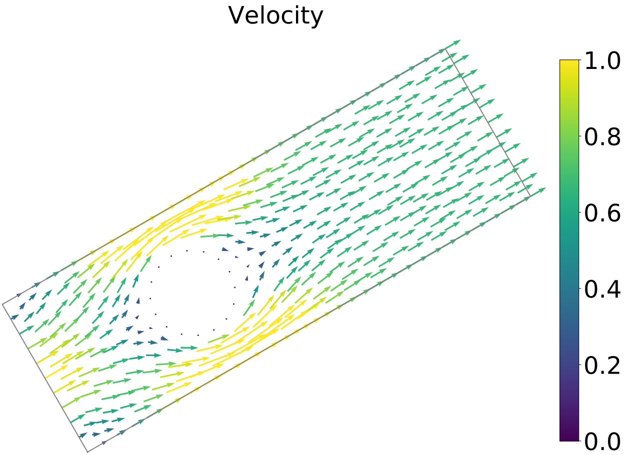
$$V_1 = \{ v = (\boldsymbol{u}, p) \in V : \exists v' \in V \text{ s.t. } v = v' - P_0 v'. \}$$

Imperial College London

Stokes Equation

```python
Vel = VectorFunctionSpace(mesh, "CG", 2)
Q = FunctionSpace(mesh, "CG", 1)
V = Vel * Q

vq = TestFunction(V)
up = Function(V)
v, q = split(vq)
u, p = split(up)

normal = FacetNormal(mesh)
V1 = BoundaryComponentSubspace(V.sub(0), (3, 5), normal)
vq1 = Masked(vq, V1)
v1, q1 = split(vq1)
v0 = v - v1
q0 = q - q1
...
bcs = [DirichletBC(V.sub(0), u_inflow, (1, )),
       DirichletBC(V.sub(0), (0, 0), (4, ))]
F = nu * inner(grad(u), grad(v0)) * dx \
  - inner(p, div(v0)) * dx \
  - inner(div(u), q0) * dx \
  + inner(dot(u, normal), dot(v1, normal)) * ds((3, 5))
solve(F == 0, up, bcs=bcs)
```

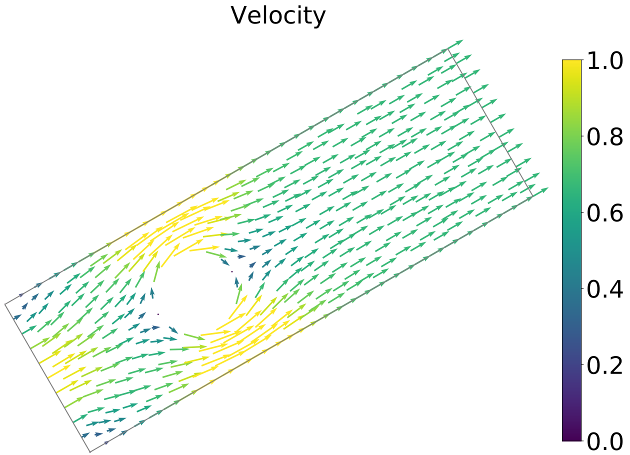
Stokes Equation



Stokes Equation

```
Vel = VectorFunctionSpace(mesh, "CG", 2)
Q = FunctionSpace(mesh, "CG", 1)
V = Vel * Q

vq = TestFunction(V)
up = Function(V)
v, q = split(vq)
u, p = split(up)

normal = FacetNormal(mesh)
V1 = BoundaryComponentSubspace(V.sub(0), (3, 4, 5), normal)
vq1 = Masked(vq, V1)
v1, q1 = split(vq1)
v0 = v - v1
q0 = q - q1
...
bcs = [DirichletBC(V.sub(0), u_inflow, (1, )), ]

F = nu * inner(grad(u), grad(v0)) * dx \
  - inner(p, div(v0)) * dx \
  - inner(div(u), q0) * dx \
  + inner(dot(u, normal), dot(v1, normal)) * ds((3, 4, 5))
solve(F == 0, up, bcs=bcs)
```

Stokes Equation



Hyper Elasticity

Solve for $\boldsymbol{u}$:

Domain:

$$\int (2\mu \boldsymbol{E} : \mathrm{sym}(\boldsymbol{F}^{\mathsf{T}} \cdot \nabla v) + \lambda \mathrm{tr} \boldsymbol{E} (\boldsymbol{F}^{\mathsf{T}} : \nabla v) + \rho \boldsymbol{g} \cdot \boldsymbol{v}) \mathrm{d}\Omega = 0,$$

where: $\forall \boldsymbol{v} \in V_0,$

$$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^{\mathsf{T}} \boldsymbol{F} - \boldsymbol{I}),$$

$$\boldsymbol{F} = \boldsymbol{I} + \nabla \boldsymbol{u}.$$

Boundary:

$$\int (\boldsymbol{u} \cdot \boldsymbol{n} - 0)(\boldsymbol{v} \cdot \boldsymbol{n}) \mathrm{d}\Gamma_{(1,3)} = 0,$$

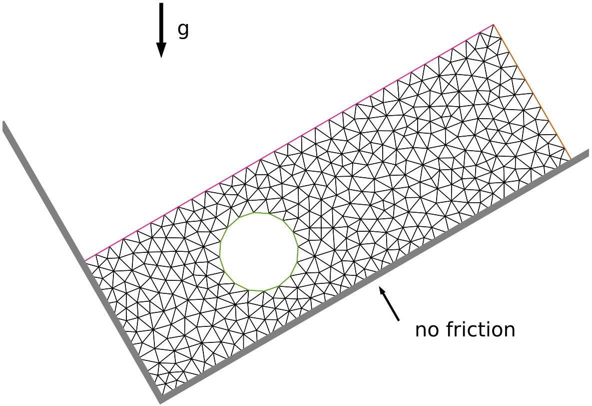
$\forall \boldsymbol{v} \in V_1.$

$$V_0 = \{\boldsymbol{v} \in V : \boldsymbol{v} \cdot \boldsymbol{n} = 0 \text{ on } \Gamma_{(1,3)}.\}$$

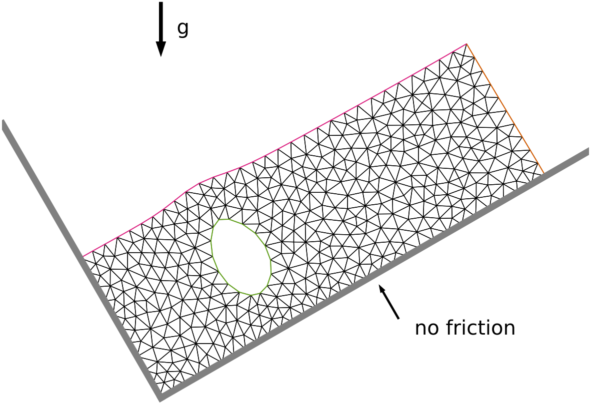
$$V_1 = \{\boldsymbol{v} \in V : \exists \boldsymbol{v}' \in V \text{ s.t. } \boldsymbol{v} = \boldsymbol{v}' - P_0 \boldsymbol{v}'.\}$$

$\mu = 0.5,\ \lambda = 1.0,\ \boldsymbol{g} = (0, 1),\ \rho = 0.01.$

Hyper Elasticity



Hyper Elasticity



Cahn-Hilliard Equation [Racke and Zheng, 2003]

Solve for (c, μ):

Domain:

$$\int (\dot{c}q + \nabla\mu \cdot \nabla q)\mathrm{d}\Omega = 0,$$

$$\int (-\mu\nu + K\nabla c \cdot \nabla\nu + f(c)\nu)\mathrm{d}\Omega = 0,$$

where $f(c) = -c + c^3$. $\forall (q, \nu) \in V_0,$

Boundary:

$$\int (\frac{1}{d_s}\dot{c}q + \sigma_s \nabla_\tau c \cdot \nabla_\tau q + (\nabla c \cdot \boldsymbol{n})q + g_s cq - h_s q)\mathrm{d}\Gamma = 0,$$

$$\forall (q, \nu) \in V_1.$$

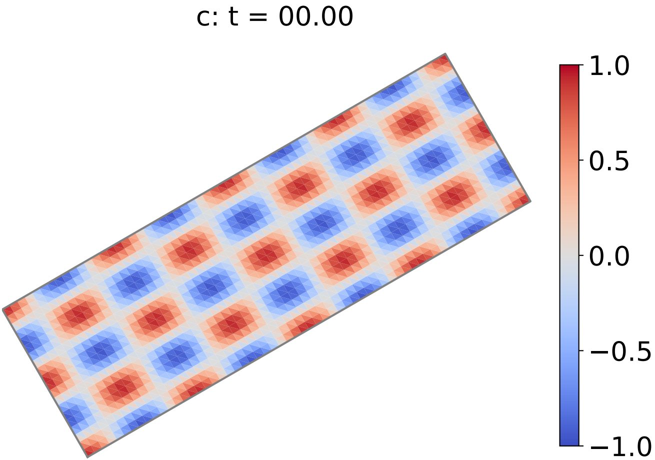
$$V_0 = \{v = (c, \mu) \in V : c = 0 \text{ on } \Gamma.\}$$

$$V_1 = \{v = (c, \mu) \in V : \exists v' \in V \text{ s.t. } v = v' - P_0 v'.\}$$

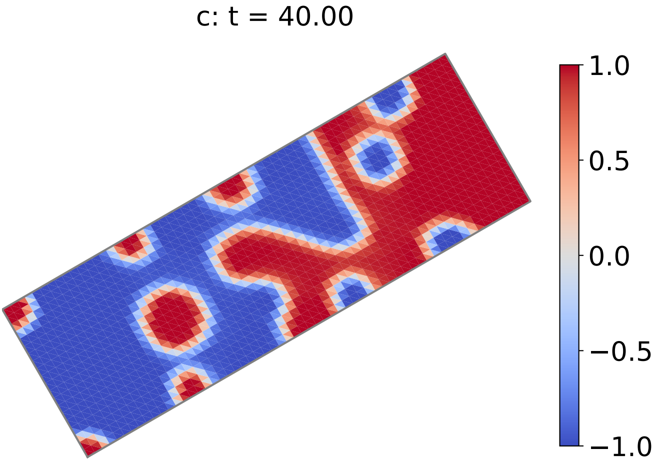
$K = 0.05,\ d_s = 1,\ \sigma_s = 0.001,\ g_s = 0.001,\ h_s = 0.005(x - 15).$

Backward Euler ($\Delta t = 0.1$).

Cahn-Hilliard Equation



Cahn-Hilliard Equation



Summary

Decompose function spaces.

Handle more general boundary conditions:

- Dirichlet boundary conditions:
  - Stokes equation.
  - Hyper elasticity.
- Equation boundary conditions:
  - Cahn-Hilliard equation with dynamic boundary conditions.

References

[1] R. Racke and S. Zheng
The Cahn-Hilliard Equation with Dynamic Boundary Conditions.
Advances in Differential Equations, 8(1):83–110, 2003.

Imperial College London