Automating Parametric Geometry using FEniCS Tools

Presented at FEniCS'10

Peter Brune[1], Johan Hoffman[2], Johan Jansson[2]
[1]UChicago CS, [2]KTH CSC

5/11/10

Contents

- a novel use of FEM automation machinery
- problem of interest
- (VERY) preliminary results!

Automation Goals

- UFL allows for *Form Transformations*
- traverses the expression tree and applies some set of rules
- parametry may be phrased like this
- easy use of higher-order geometry with *standard FEM tools*

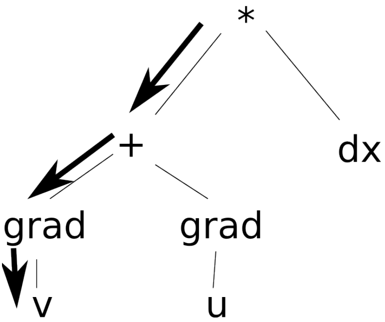


Figure: UFL Form Transformation

Parametry as a Transformation

- mapping from affine to parametric $J = \nabla \vec{x}_{iso}$
- where $\vec{x}_{iso}$ are the *Parametric Coordinates*
- J^{-1} maps back

- spatial derivatives transform as $\frac{\partial f}{\partial x_i} = (J^{-1} \nabla f)_i$
- normals transform as $\vec{n} = \frac{1}{|J^{-1}\vec{n}|} J^{-1} \vec{n}_{aff}$
- volume measures transform as $dx = \frac{1}{|J^{-1}|} dx_{aff}$
- facet measures transform as $ds = \frac{|J^{-1}\vec{n}|}{|J^{-1}|} ds_{aff}$

Example Transformation

$$\int_{\Omega} \nabla u \cdot \nabla v dx - \int_{\delta\Omega} \vec{n} \cdot \nabla u v ds = 0$$

Transforms to

$$\int_{\Omega_a} J^{-1} \nabla u \cdot J^{-1} \nabla v \frac{1}{|J^{-1}|} dx - \int_{\delta\Omega_a} \frac{1}{|J^{-1}\vec{n}|} J^{-1}\vec{n} \cdot J^{-1} \nabla u v \frac{|J^{-1}\vec{n}|}{|J^{-1}|} ds = 0$$

Doing this by hand for more complex systems can be painful, so...

UFL Extension for Parametry

- a preprocessor script that applies the transformation
- a number of definitions that can be shared
- allows for easy conversion of a complex system
- 32 forms per UNICORN solver
- easily appended to the form using `import`

Automating Parametry

- users can write a simple UFL form

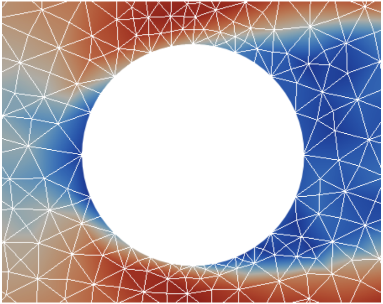
```
from isoparametry import *
iso_element = VectorElement(Lagrange, cell, 2)
iso_coords = Coefficient(iso_element)
J = grad(iso_coords)

...

a_a = inner(grad(u), grad(v))
a = a_a*dx(0) + apply_geometry(a_a, J)*dx(1)
L = f*v*dx(0) + apply_geometry(f*v, J)*dx(1)
```

Motivating Problems

- flow problems
- boundary-layer phenomena
- accurate calculation of physical quantities

Superparametric Geometry and Flow

- simple test problems have exactly described boundaries
- geometric error is quantifiable
- more accurate surface and volume quantities
- other methods (NURBS) have show this beneficial[Huerta]
- entire methods based on geometric error[Hughes]



Incompressible Navier-Stokes Equations

$$
\begin{aligned}
\dot{\vec{u}} - \nu \Delta \vec{u} + (\vec{u} \nabla \cdot \vec{u}) + \nabla p &= \vec{f} \text{ in } \Omega \\
\nabla \cdot \vec{u} &= 0 \text{ in } \Omega \\
\vec{n} \cdot \vec{u} &= 0 \text{ on } \delta\Omega
\end{aligned}
$$

- $\vec{u}$ is velocity
- ν is the viscosity
- p is the pressure
- $\vec{n}$ is the normal

Boundary Conditions

- slip condition imposed weakly

$$a_{mom}(\vec{u},\vec{v}) + \int_{\delta\Omega} \alpha(\vec{n}\cdot\vec{u})(\vec{n}\cdot\vec{v})ds = L_{mom}(\vec{v})$$

- α is a penalty coefficient
- allows for consideration of curvature effects
- want alpha to be **small**

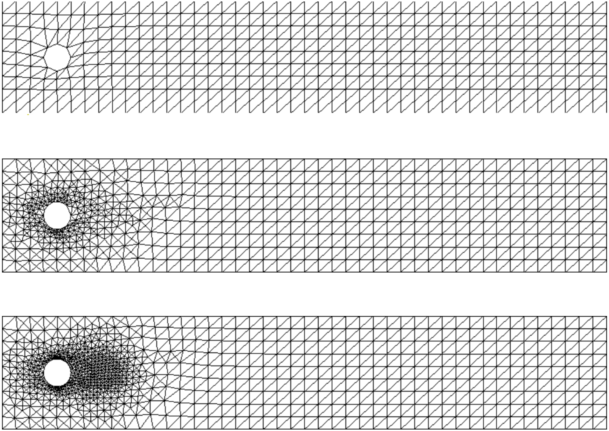
Error Estimation

- Use a standard error estimator with G2[Johnson, Hoffman]
- $M(\mathbf{u}) = \mathbf{u}, \phi = (\vec{u}, p), (\vec{\psi}, \theta)$
- **volume** and **surface area** extensions

$$\begin{aligned} M(\mathbf{u}) - M(\mathbf{u}_h)_h &= \\ (R(u) - R(u_h), \phi) &+ \\ (\vec{u}, \vec{\psi}) - (\vec{u}, \vec{\psi}_h)_h &- \\ (\vec{u} - \vec{u_h}, \nabla\vec{\psi} \cdot \vec{n} - \theta\vec{n}) & \end{aligned}$$

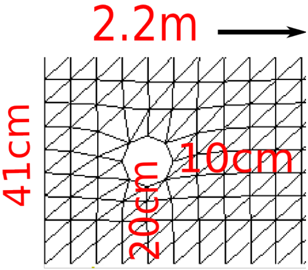
Refinement

- we can use this to refine adaptively using FEniCS tools
- get a progressively better-resolved answer



Initial Results

- cylinder test [Turek]
- both parametric and non-parametric weak BCs
- ten iterative refinements
- standard error-estimator



Drag

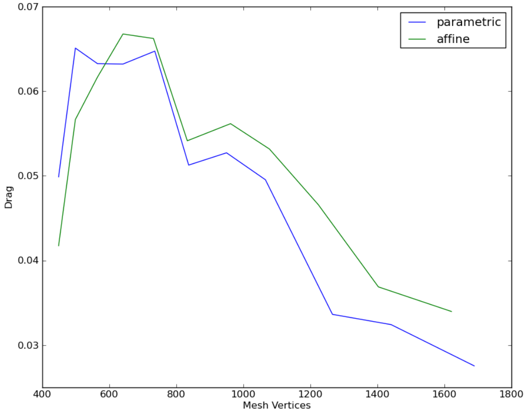
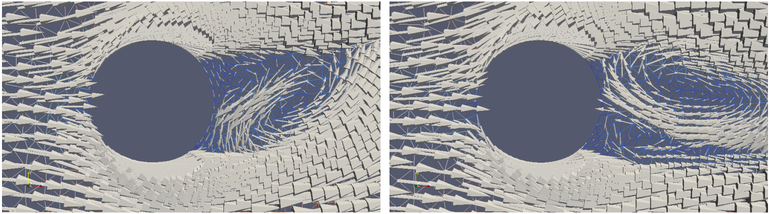


Figure: Time-Averaged Drag vs. Refinement

Separation



(a) parametric geometry (b) affine geometry

- parametric boundaries help lessen separation
- happens on coarse and fine meshes

Continuing Work

- see drag convergence
- incorporate the boundary error estimate
- work-per-digit of accuracy calculations
- airfoil calculations