2 – Discontinuous Galerkin Methods for Flow Problems

Per-Olof Persson

Department of Mathematics, University of California, Berkeley
Mathematics Department, Lawrence Berkeley National Laboratory

Summer School on Discontinuous Galerkin Methods
International Center for Numerical Methods in Engineering
Barcelona, Spain

University of California
Berkeley

June 11-15, 2012

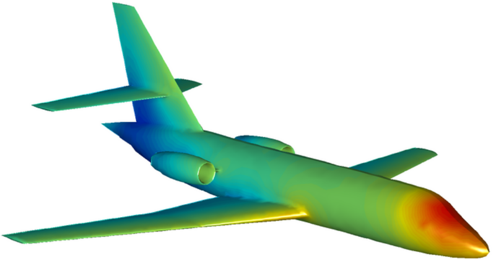
Berkeley Lab

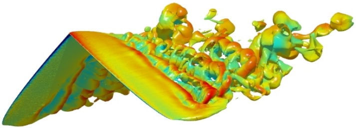
Outline

1. Introduction
2. The DG Method for Navier-Stokes
   - Application: Implicit Large Eddy Simulation
3. Curved Mesh Generation
4. Artificial Viscosity and Shock Capturing
   - Application: RAE2822
5. ALE for Deforming Domains
   - Application: Vertical Axis Wind Turbines
   - Application: Flapping Flight of Bat

Motivation

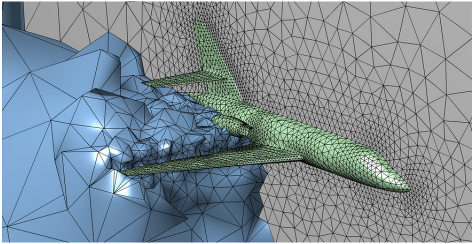
- Need for higher fidelity predictions in computational fluid dynamics
  - Turbulent flows, fluid/structure interaction, flapping flight
  - Wave propagation, multiscale phenomena, non-linear interactions
- Widely believed that high-order methods will become the standard in future generations of simulation software
- The DG method has emerged as one of the most promising schemes, since it provides high-order accuracy on fully unstructured meshes

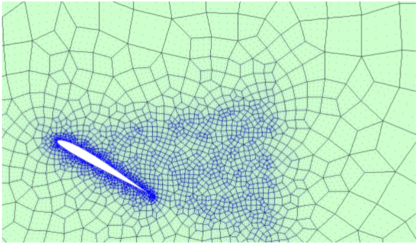




Why Unstructured Meshes?

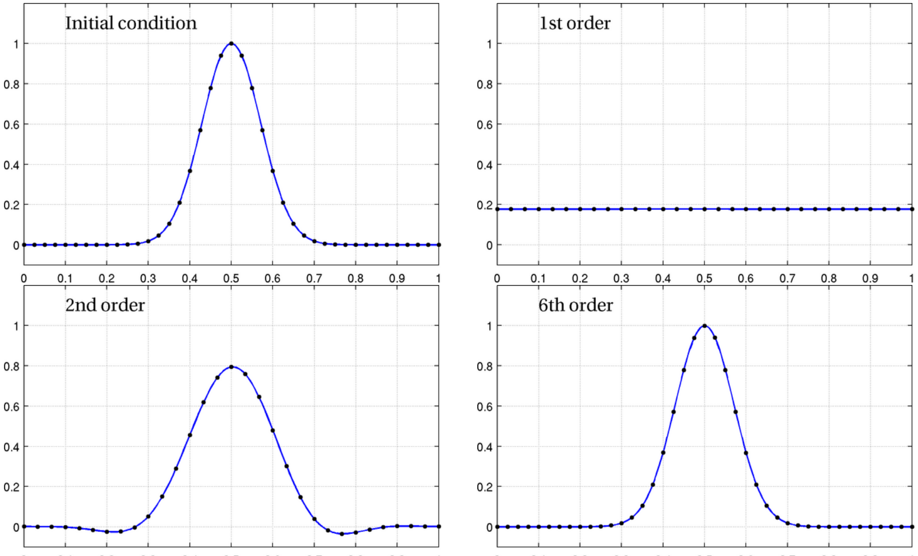
- Complex *geometries* need flexible element topologies
- Complex *solution fields* need spatially variable resolution
- Fully automated mesh generators for CAD geometries are based on unstructured simplex elements
- Real-world simulation software dominated by unstructured mesh discretization schemes





Why high-order accurate methods?

- Scalar convection equation $u_t + u_x = 0$
- High-order gives *superior performance* for *equal resolution*



Numerical Schemes for Flow Problems

- Complex flow features $\rightarrow$ High-order accuracy required
- Complex geometries, adaptation $\rightarrow$ Unstructured grids required
- Discontinuous Galerkin (DG) methods have these properties:

	FVM	FDM	FEM	DG
1) High-order/Low dispersion	✗	✓	✓	✓
2) Unstructured meshes	✓	✗	✓	✓
3) Stability for conservation laws	✓	✓	✗	✓

- However, still several problems to resolve:
  - High CPU/memory requirements (compared to FVM or H-O FDM)
  - Low tolerance to under-resolved features
  - High-order geometry representation and mesh generation

Governing Equations

- The (compressible) *Navier-Stokes equations*:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = 0,$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_i}(\rho u_i u_j + p) = + \frac{\partial \tau_{ij}}{\partial x_j} \quad \text{for } i = 1, 2, 3,$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x_i}\left(u_j(\rho E + p)\right) = -\frac{\partial q_j}{\partial x_j} + \frac{\partial}{\partial x_j}(u_j \tau_{ij}),$$

with

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_j}\delta_{ij}\right), \qquad q_j = -\frac{\mu}{\mathrm{Pr}}\frac{\partial}{\partial x_j}\left(E + \frac{p}{\rho} - \frac{1}{2}u_k u_k\right)$$

$$p = (\gamma - 1)\rho\left(E - \frac{1}{2}u_k u_k\right)$$

- When $\mu = 0$, reduces to first-order system – *Euler's equations of gas dynamics*

Outline

The Discontinuous Galerkin Method

- (Reed/Hill 1973, Lesaint/Raviart 1974, Cockburn/Shu 1989-, etc)
- Write the first-order equations as a system of conservation laws:

$$\boldsymbol{u}_t + \nabla \cdot \boldsymbol{F}_i(\boldsymbol{u}) = 0$$

- Triangulate domain Ω into elements $\kappa \in \mathcal{T}_h$
- Seek approximate solution $\boldsymbol{u}_h$ in space of element-wise polynomials:

$$\boldsymbol{V}_h^p = \{\boldsymbol{v} \in L^2(\Omega) : \boldsymbol{v}|_\kappa \in P^p(\kappa)\ \forall \kappa \in T_h\}$$

- Multiply by test function $\boldsymbol{v}_h \in \boldsymbol{V}_h^p$ and integrate over element κ:

$$\int_\kappa [(\boldsymbol{u}_h)_t + \nabla \cdot \boldsymbol{F}_i(\boldsymbol{u}_h)]\, \boldsymbol{v}_h \, d\boldsymbol{x} = 0$$

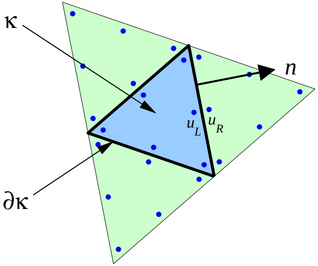
The Discontinuous Galerkin Method

- Integrate by parts:

$$\int_\kappa [(\boldsymbol{u}_h)_t]\, \boldsymbol{v}_h\, d\boldsymbol{x} - \int_\kappa \boldsymbol{F}_i(\boldsymbol{u}_h) \nabla \boldsymbol{v}_h\, d\boldsymbol{x} + \int_{\partial\kappa} \hat{\boldsymbol{F}}_i(\boldsymbol{u}_h^+, \boldsymbol{u}_h^-, \hat{\boldsymbol{n}}) \boldsymbol{v}_h^+\, ds = 0$$

  with numerical flux function $\hat{\boldsymbol{F}}_i(\boldsymbol{u}_L, \boldsymbol{u}_R, \hat{\boldsymbol{n}})$ for left/right states $\boldsymbol{u}_L$,$\boldsymbol{u}_R$ in direction $\hat{\boldsymbol{n}}$ (Godunov, Roe, Osher, Van Leer, Lax-Friedrichs, etc)

- Global problem: Find $\boldsymbol{u}_h \in \boldsymbol{V}_h^p$ such that this weighted residual is zero for all $\boldsymbol{v}_h \in \boldsymbol{V}_h^p$
- Error $= \mathcal{O}(h^{p+1})$ for smooth solutions



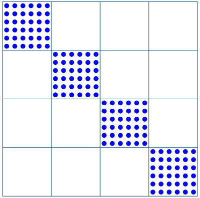
The DG Method – Observations

- Reduces to the finite volume method for $p = 0$:

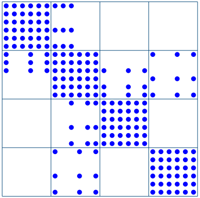
$$(\boldsymbol{u}_h)_t A_\kappa + \int_{\partial\kappa} \hat{\boldsymbol{F}}_i(\boldsymbol{u}_h^+, \boldsymbol{u}_h^-, \hat{\boldsymbol{n}})\, ds = 0$$

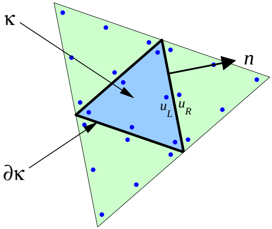
- Boundary conditions enforced naturally for any degree p
- Block-diagonal mass matrix (no overlap between basis functions)
- Block-wise compact stencil – neighboring elements connected

Mass Matrix



Jacobian





Viscous Discretization

- Write equations as system of first order equations:

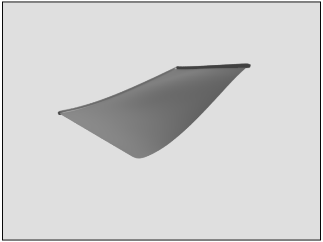
$$\boldsymbol{u}_t + \nabla \cdot \boldsymbol{F}_i(\boldsymbol{u}) - \nabla \cdot \boldsymbol{F}_v(\boldsymbol{u}, \boldsymbol{\sigma}) = 0$$

$$\boldsymbol{\sigma} - \nabla \boldsymbol{u} = 0$$

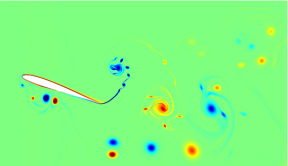
- Discretize using DG, choose appropriate numerical fluxes:
  - For inviscid component $\boldsymbol{F}_i$, use approximate Riemann solvers (Godunov, Roe, Osher, Van Leer, Lax-Friedrichs, etc)
  - For viscous component $\boldsymbol{F}_v$, use numerical fluxes $\hat{\boldsymbol{\sigma}}$, $\hat{u}$ according to methods such as IP/BR2/LDG/CDG

Implementation: The 3DG Software Package

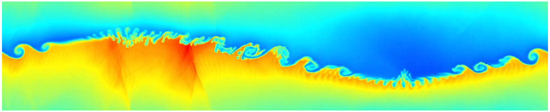
- High-order discretizations on unstructured meshes
- Optimized C++ code with MATLAB and Python interfaces
- Capable of simulating challenging problems:
  - complex real-world geometries
  - transitional flows, multiple scales
  - moving and deforming domains
  - fluid-structure interactions
- General multiphysics framework applicable to a wide range of challenging problems



Thin Structures



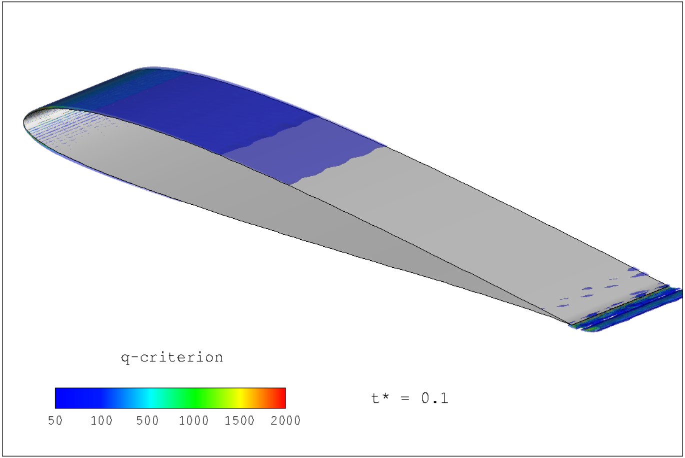
Unsteady Flows



Aeroacoustics

Example: ILES at *Re* = 60,000

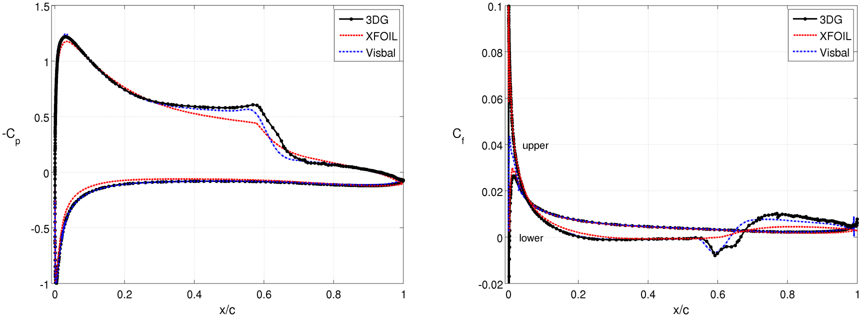
- Implicit Large Eddy Simulations for flow past airfoil
- Separation and transition well captured
- Vortical structures: iso-surfaces of q-criterion ($\nabla^2 p/2\rho$)



Example: ILES at *Re* = 60,000

- Good agreement with XFoil and previously published ILES

  [Uranga/Persson/Drela/Peraire ’11]

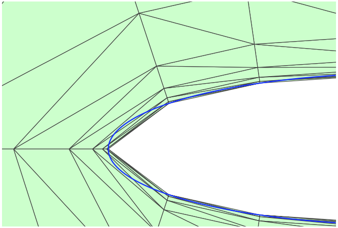


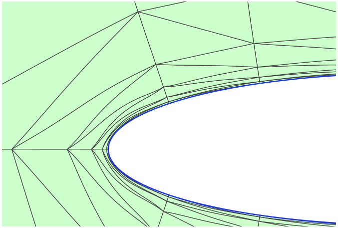
Average pressure and skin friction coefficients

Outline

Curved Mesh Generation

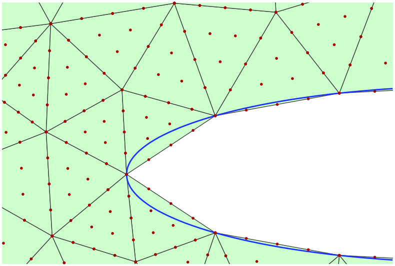
- Automatic generation of non-inverted curved elements largely an unresolved problem
- In general this is a global problem, affecting many elements except for simple isotropic 2-D meshes
- In [Persson/Peraire '09], we proposed a *non-linear solid mechanics* approach, where the mesh is considered an elastic deformable solid



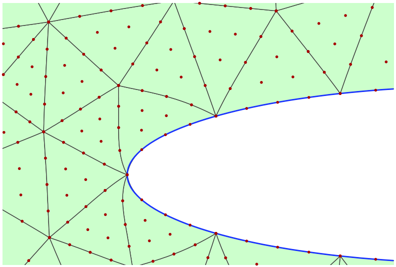


Curved Mesh Generation using Solid Mechanics

- The initial, straight-sided mesh corresponds to undeformed solid
- External forces come from the true boundary data
- Solving for a force equilibrium gives the deformed, curved, boundary conforming mesh
- Bottom-up approach can be used to obtain the boundary data



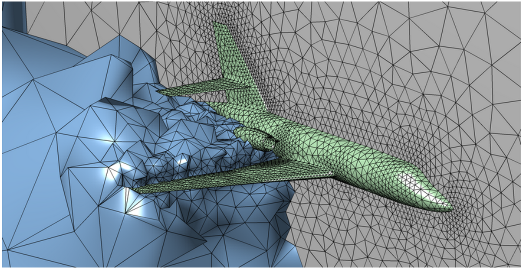
Reference domain, initial configuration



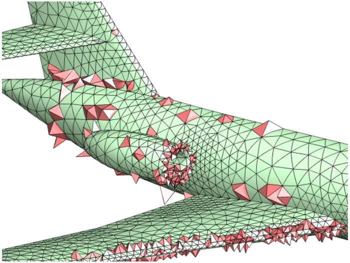
Equilibrium solution, final curved mesh

Tetrahedral Mesh of Falcon Aircraft

- Real-world mesh with coarse but realistic elements
- Unstructured Delaunay refinement mesh, with highly curved boundary segments
- Many elements would invert with a local element-wise approach



Tetrahedral mesh

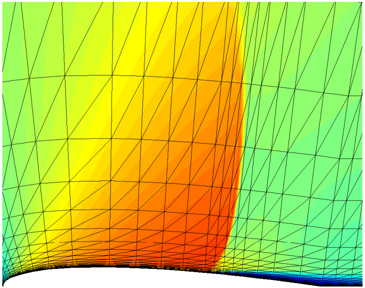


Elements with $I < 0.5$

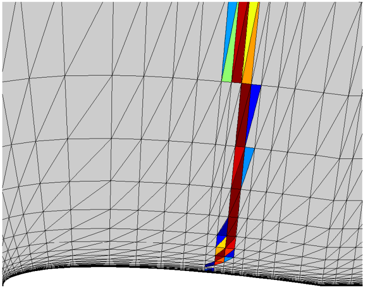
Outline

Artificial Viscosity for Underresolved Features

- Cannot resolve all solution features (shocks, RANS, singularities)
- Low dissipation makes DG sensitive to underresolution
- Detect by sensors and add viscosity [Persson/Peraire 06,07]
- Enables shock capturing with sub-cell resolution and robust solution of Spalart-Alamaras RANS model



Mach



Sensor

Shock Sensor

- Regularity of solution determined from the decay rate of expansion coefficients in orthogonal basis
- Example: Periodic Fourier case: $f(x) = \sum_{k=-\infty}^{\infty} g_k e^{ikx}$
  If $f(x)$ has m continuous derivatives $\rightarrow \ |g_k| \sim k^{-(m+1)}$
- For simplices: Expand solution in orthonormal Koornwinder basis:

$$u = \sum_{i=1}^{N(p)} u_i \psi_i, \quad \hat{u} = \sum_{i=1}^{N(p-1)} u_i \psi_i, \quad s_e = \log_{10} \left(\frac{(u - \hat{u}, u - \hat{u})_e}{(u, u)_e} \right)$$

- Determine elemental piecewise constant ε_e

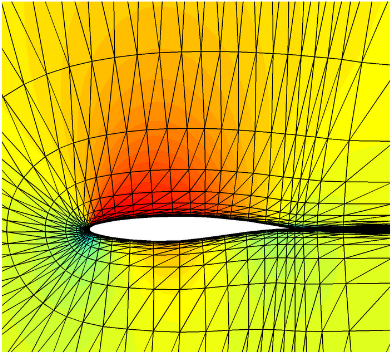
$$\varepsilon_e = \begin{cases} 0 & \text{if} \quad s_e < s_0 - \kappa \\ \frac{\varepsilon_0}{2} \left(1 + \sin \frac{\pi (s_e - s_0)}{2\kappa} \right) & \text{if} \quad s_0 - \kappa \le s_e \le s_0 + \kappa \\ \varepsilon_0 & \text{if} \quad s_e > s_0 + \kappa \end{cases} .$$

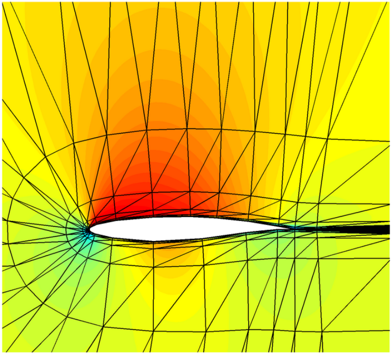
where $\varepsilon_0 \sim h/p$, $s_0 \sim 1/p^4$ and κ empirical

Example: RAE2822

- Turbulent RANS flow ($M = 0.675, \alpha = 2.31^o, \mathrm{Re} = 6.5 \cdot 10^6$)
- p-converged solution, fixed resolution h/p

$p = 2$ (constant h/p) $p = 4$



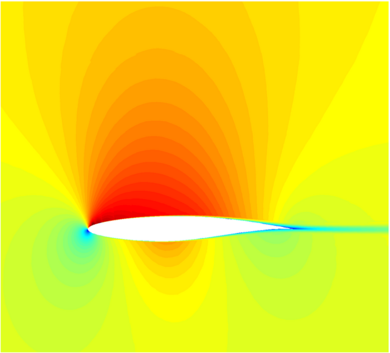


$C_L = 0.6144 \quad C_D = 0.0104$ $C_L = 0.6131 \quad C_D = 0.0103$

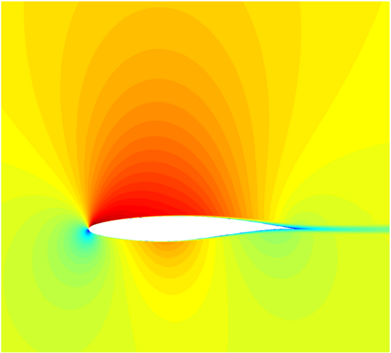
Example: RAE2822

- Turbulent RANS flow ($M = 0.675, \alpha = 2.31^o, \mathrm{Re} = 6.5 \cdot 10^6$)
- p-converged solution, fixed resolution h/p

$p = 2$ (constant h/p) $p = 4$



$C_L = 0.6144 \quad C_D = 0.0104$

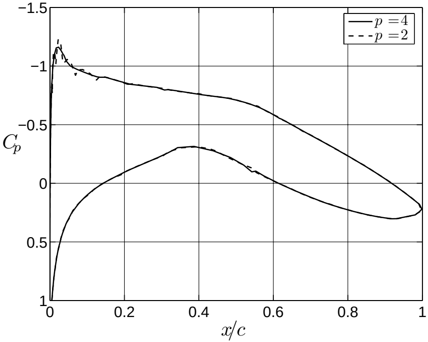


$C_L = 0.6131 \quad C_D = 0.0103$

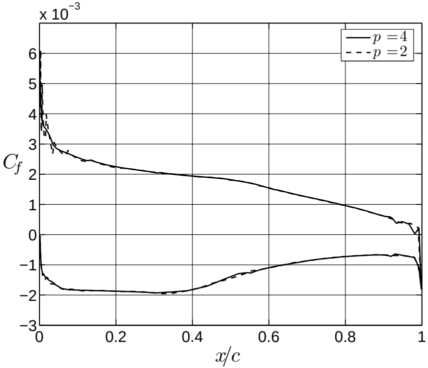
Example: RAE2822

- Highly accurate boundary forces even with coarse meshes

C_p



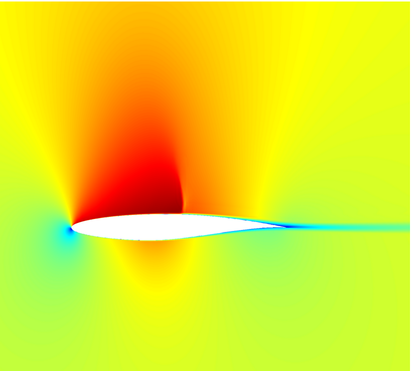
C_f

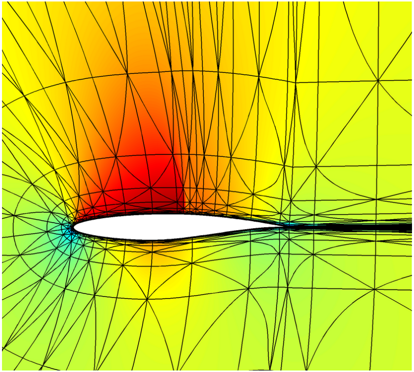


Example: RAE2822, Transonic

- Transonic flow ($M = 0.729, \mathrm{Re} = 6.5 \cdot 10^6$)
- Sub-cell resolution of shocks

$p = 4$

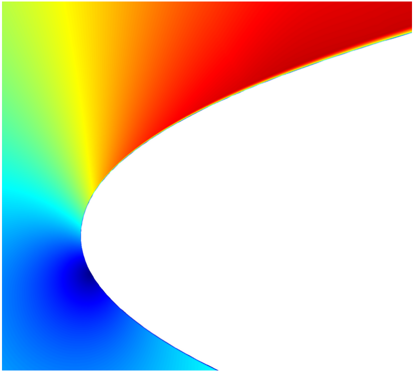


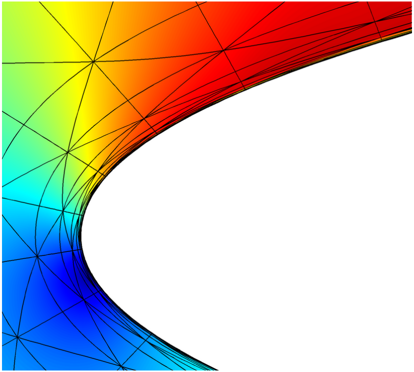


Example: RAE2822, Transonic

- Transonic flow ($M = 0.729, \mathrm{Re} = 6.5 \cdot 10^6$)
- Sub-cell resolution of shocks

$p = 4$





Outline

ALE Formulation for Deforming Domains

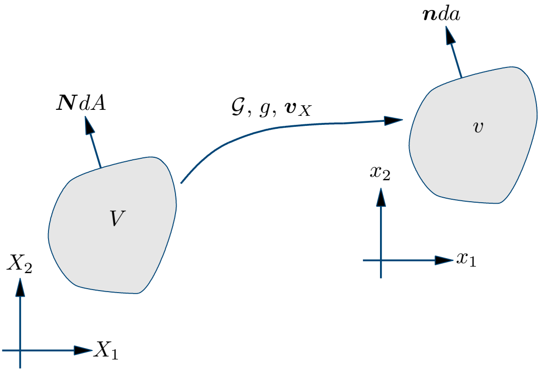
- Use mapping-based ALE formulation for moving domains [Visbal/Gaitonde '02], [Persson/Bonet/Peraire '09]
- Map from reference domain V to physical deformable domain $v(t)$
- Introduce the *mapping deformation gradient* $\boldsymbol{G}$ and the *mapping velocity* $\boldsymbol{v}_X$ as

$$\boldsymbol{G} = \nabla_X \boldsymbol{\mathcal{G}}$$

$$\boldsymbol{v}_X = \left. \frac{\partial \boldsymbol{\mathcal{G}}}{\partial t} \right|_X$$

and set $g = \det(\boldsymbol{G})$

- Transform equations to account for the motion



Transformed Equations

- The system of conservation laws in the physical domain $v(t)$

$$\left.\frac{\partial \boldsymbol{U}_x}{\partial t}\right|_x + \boldsymbol{\nabla}_x \cdot \boldsymbol{F}_x(\boldsymbol{U}_x, \boldsymbol{\nabla}_x \boldsymbol{U}_x) = 0$$

  can be written in the reference configuration V as

$$\left.\frac{\partial \boldsymbol{U}_X}{\partial t}\right|_X + \boldsymbol{\nabla}_X \cdot \boldsymbol{F}_X(\boldsymbol{U}_X, \boldsymbol{\nabla}_X \boldsymbol{U}_X) = 0$$

  where

$$\boldsymbol{U}_X = g\boldsymbol{U}_x \ , \qquad \boldsymbol{F}_X = g\boldsymbol{G}^{-1}\boldsymbol{F}_x - \boldsymbol{U}_X\boldsymbol{G}^{-1}\boldsymbol{v}_X$$

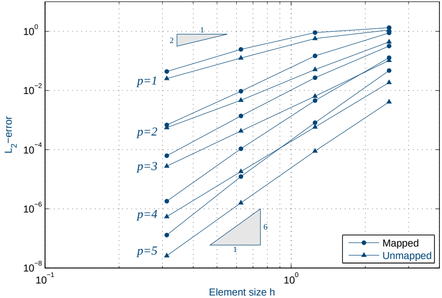
  and

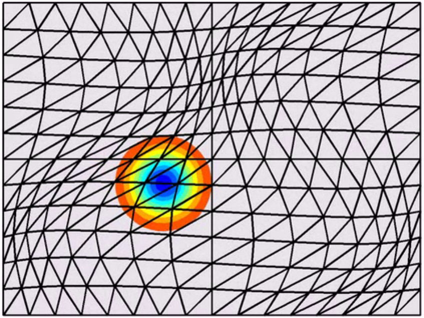
$$\boldsymbol{\nabla}_x \boldsymbol{U}_x = \boldsymbol{\nabla}_X(g^{-1}\boldsymbol{U}_X)\boldsymbol{G}^{-T} = (g^{-1}\boldsymbol{\nabla}_X \boldsymbol{U}_X - \boldsymbol{U}_X \boldsymbol{\nabla}_X(g^{-1}))\boldsymbol{G}^{-T}$$

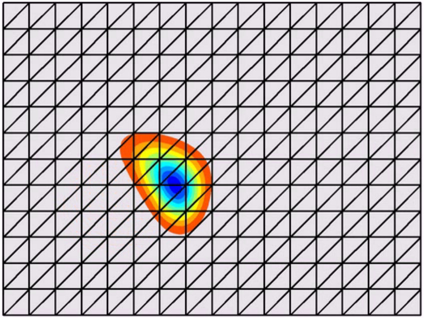
- Details in [Persson/Bonet/Peraire '09], including how to satisfy the so-called Geometric Conservation Law (GCL)

ALE Formulation for Deforming Domains

- Mapping-based formulation gives arbitrarily high-order accuracy in space and time







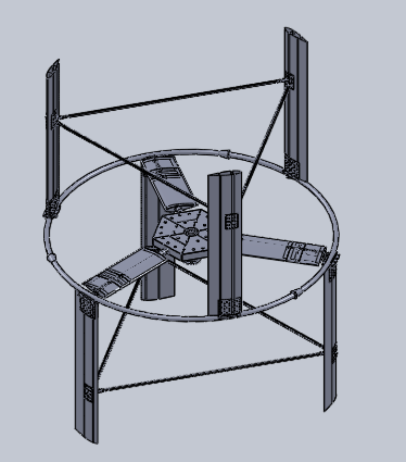
Vertical Axis Wind Turbines

- Recent interest in vertical axis wind turbines (VAWT):
  - 2D airfoils, easy to manufacture, supportable at both ends
  - Omnidirectional (good in gusty, low wind, e.g. close to ground)
  - Lower blade speeds – lower noise and impact
  - Can be packed close together in wind farms
- Numerical simulations can help overcome remaining challenges:
  - Lower theoretical (and practical) efficiency than HAWTs
  - Sensitive to design conditions
  - Structural problems, fatigue and catastrophic failure

Windterra ECO 1200 1Kw VAWT

Vertical Axis Wind Turbines

- Experimental design by G. Dahlbacka (LBNL) and collaborators
- 3kW unit, CAD design (left) and assembled unit (right)

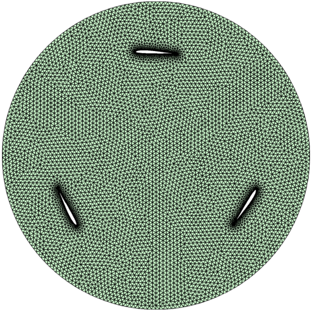


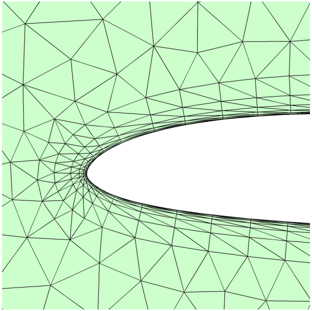
VAWT – Mathematical Model and Discretization

- Preliminary 2D simulation, using vertical symmetry
- Solve the Navier-Stokes equations in a rotating frame:

$$\mathcal{G}(X, Y, t) = \begin{bmatrix} \cos \omega t & -\sin \omega t \\ \sin \omega t & \cos \omega t \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix}$$

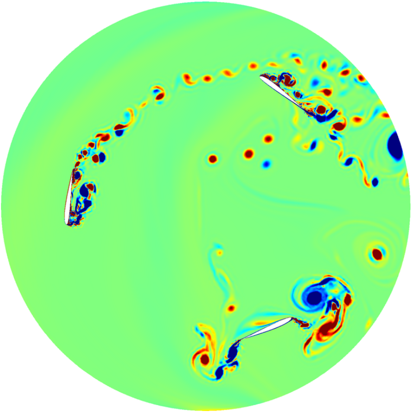
- Hybrid boundary layer/unstructured mesh, element degree $p = 3$

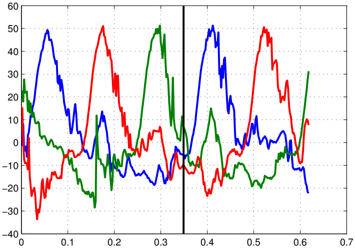




VAWT – Numerical Results

- Simulation with freestream wind speed 12 m/s (horizontally, from the left) and wing tip speed ratio 1.5
- Visualization by z-vorticity in rotating frame

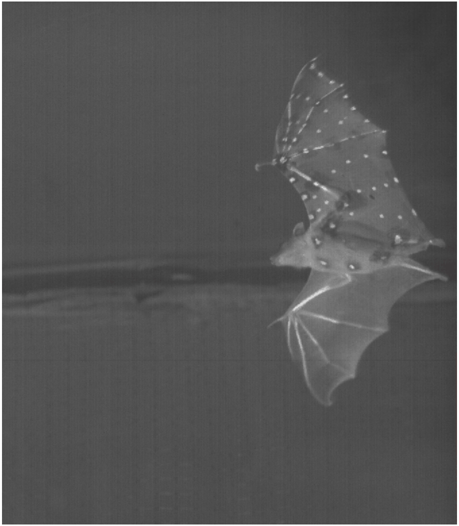




Moment on each blade vs. time

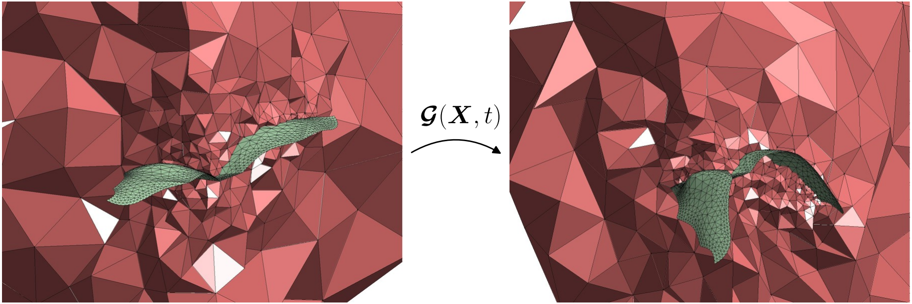
Bio-Inspiration for Flapping Wing MAVs

- Develop high-order accurate simulation capabilities that capture the complex physics in flapping flight
- Use the computational tools for increased understanding and to design optimized flapping kinematics



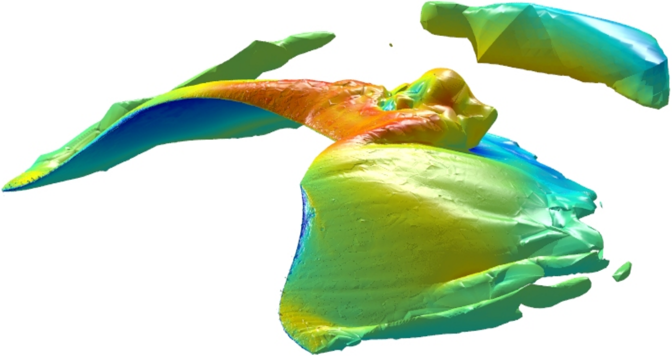
Domain Mapping

- Highly complex wing motion from measured data
- Construct mapping $\mathcal{G}(\boldsymbol{X}, t)$ *numerically* by nonlinear solid mechanics approach [Persson '09]
- A reference mesh (left) is deformed elastically to smoothly align with the prescribed wing motion (right)
- Grid velocity $\boldsymbol{v}_X = \frac{\partial \mathcal{G}}{\partial t}\Big|_X$ defined consistently with DIRK scheme



Flapping Bat Flight Simulation

- Visualization of Mach number on isosurface of entropy
- Unphysical separation around simplified animal “body”



Optimal Design of Flapping Wings

- Goal: Automatically generate optimized flapping wing kinematics [Persson/Willis '11]
- A multifidelity approach, with wake-only, panel, and high-order DG methods
- Example: Flapping wing pair, prescribed camber, solve for optimal wing twist distribution