

# Integral Equation Methods for Vortex Dominated Flows, a High-order Conservative Eulerian Approach

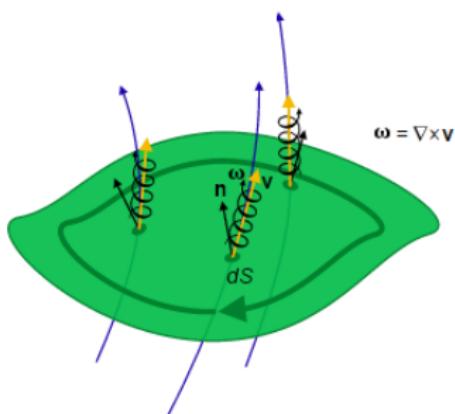
J. Bevan, UIUC

*ICERM/HKUST Fast Integral Equation Methods  
January 5, 2017*

# Vorticity and Circulation

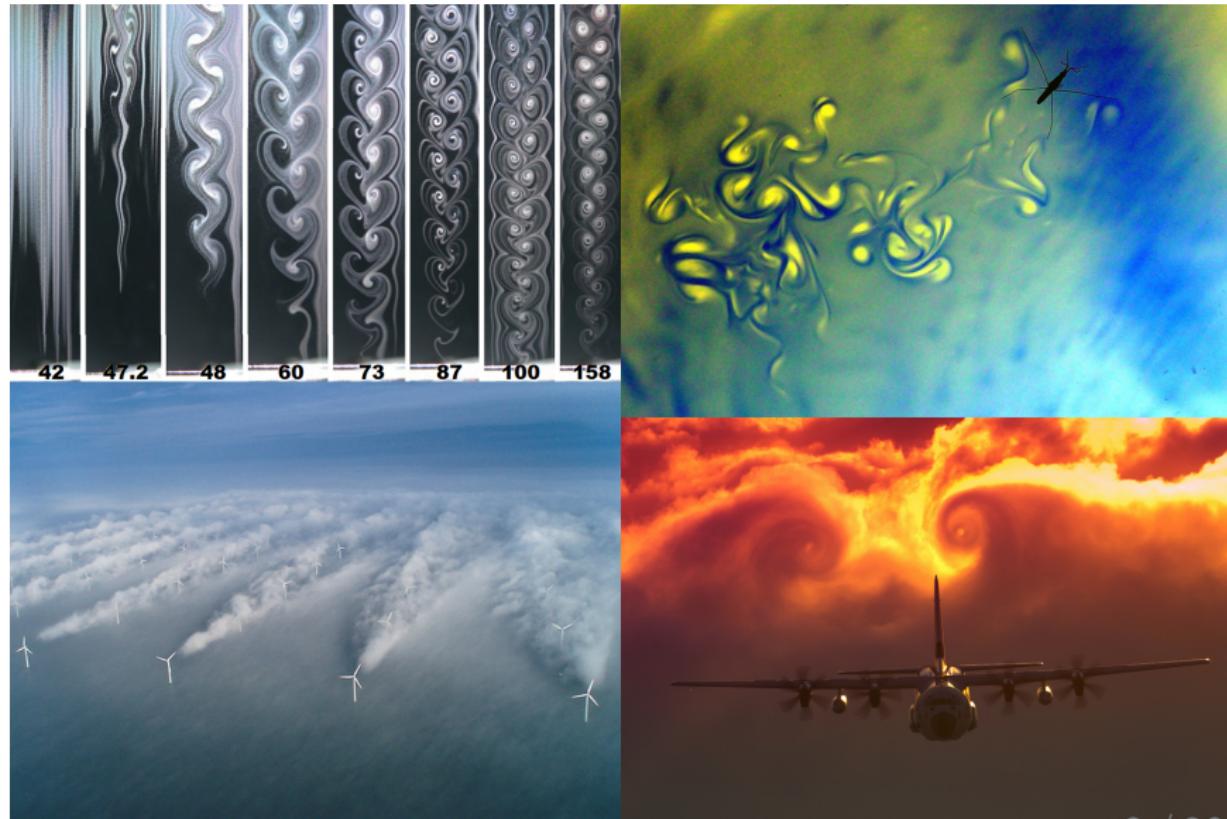
$$\boldsymbol{\omega} = \nabla \times \mathbf{u} \quad (1)$$

$$\Gamma = \oint_{\partial S} \mathbf{u} \cdot d\mathbf{l} = \iint_S \boldsymbol{\omega} \cdot d\mathbf{S} \quad (2)$$



<sup>0</sup><https://commons.wikimedia.org/wiki/File:Generalcirculation-vorticitydiagram.svg>

# Physical Examples and Motivating Problems



## Motivation

- ▶ Direct solution of Navier-Stokes impractical for many fluid problems
- ▶ Vorticity-velocity formulation well-suited for inviscid, incompressible, vortex dominated flows
- ▶ Lagrangian vortex methods are common approach<sup>1,2</sup>, but face several challenges<sup>3</sup>
- ▶ Initial vortex points become disorganized, re-meshing etc. required
- ▶ Eulerian approach avoids disorganization, extendable to high order
- ▶ Brown et al. successful with Eulerian approach for low order FVM<sup>4</sup> suggests high-order extension possible

---

<sup>1</sup>J. Strain. Fast adaptive 2D vortex methods. Journal of computational physics 132.1 (1997): 108-122.

<sup>2</sup>Moussa, C., Carley, M. J. (2008). A Lagrangian vortex method for unbounded flows. International journal for numerical methods in fluids, 58(2), 161-181.

<sup>3</sup>J. Strain. 2D vortex methods and singular quadrature rules. Journal of Computational Physics 124.1 (1996): 131-145.

<sup>4</sup>R.E. Brown. Rotor Wake Modeling for Flight Dynamic Simulation of Helicopters. AIAA Journal, 2000. Vol. 38(No. 1): p. 57-63.

# Goals of Technique

## Goals:

- ▶ Development of a high-order solver for inviscid incompressible vorticity-dominated flows in 2D
  - ▶ High-order advective solver capable of mixed order flux handling
  - ▶ High-order Biot-Savart evaluation routine

## Contributions:

- ▶ Complete high-order method for velocity-vorticity inviscid flow
- ▶ Validation of solver and underlying Eulerian vortex approach
- ▶ Evaluation of convergence, error, and performance of method and solver

## Proposed Method

- ▶ Eulerian representation of velocity and vorticity
- ▶ Solution of inviscid velocity-vorticity PDE
- ▶ Velocity evaluation via eval of integral equation (v.s. sol'n of PDE)<sup>5</sup>
- ▶ Vorticity advection via a line-DG<sup>6</sup> approach
- ▶ Method of lines, explicit time-stepping with Runge-Kutta<sup>7</sup>

---

<sup>5</sup>Winckelmans, G. S., and A. Leonard. Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows. *Journal of Computational Physics* 109.2 (1993): 247-273.

<sup>6</sup>P.O. Persson. A Sparse and High-Order Accurate Line-Based Discontinuous Galerkin Method for Unstructured Meshes. *J. Comp. Phys.*, Vol. 233, pp. 414-429, Jan 2013.

<sup>7</sup>Niegemann, Jens, Richard Diehl, and Kurt Busch. Efficient low-storage Runge–Kutta schemes with optimized stability regions. *Journal of Computational Physics* 231.2 (2012): 364-372.

## Theory: Navier-Stokes: Velocity-vorticity form

Navier-Stokes momentum equation

$$\rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \mu \nabla^2 \mathbf{u} + \frac{1}{3} \mu \nabla (\nabla \cdot \mathbf{u}) \quad (3)$$

where  $\mathbf{u}$  is the velocity field,  $p$  is the pressure field, and  $\rho$  is the density. Define *vorticity* as

$$\boldsymbol{\omega} = \nabla \times \mathbf{u} \quad (4)$$

Navier-Stokes can be recast as

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\omega} - \boldsymbol{\omega} \cdot \nabla \mathbf{u} = S(x, t) \quad (5)$$

viscous generation of vorticity,  $S$

## Velocity-vorticity form

Advantages:

- ▶ Explicit conservation of vorticity.
- ▶ Frequently distribution of the vorticity is sparse.
- ▶ No pressure term.

Simplified in 2D, vortex stretching zero. Express in terms of vortex flux  $f_i(\omega) = u_i \omega$

$$\frac{\partial \omega}{\partial t} + \frac{\partial f}{\partial x_i} = S(x, t) \quad (6)$$

## Velocity Evaluation

For incompressible flows velocity related to vorticity by

$$\nabla^2 u = -\nabla \times \omega \quad (7)$$

Invert to obtain Biot-Savart integral

$$u(x^*) = \int_{\Omega} K(x^*, x) \times \omega(x) dx \quad (8)$$

$x^*$  is velocity eval point,  $x$  is non-zero vorticity domain,  $K(x^*, x)$  singular Biot-Savart kernel<sup>8</sup>

$$K(x^*, x) = \frac{-1}{2\pi} \frac{x^* - x}{|x^* - x|^2} \quad (9)$$

---

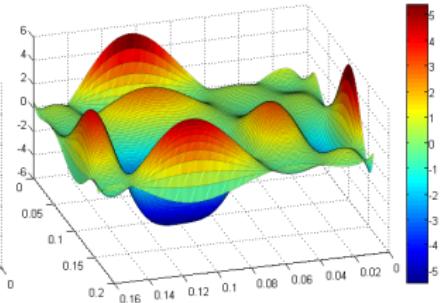
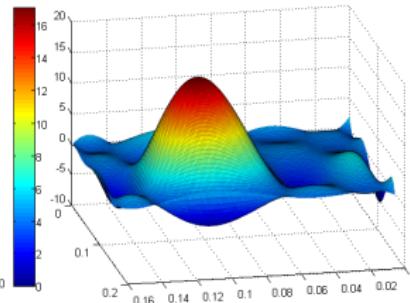
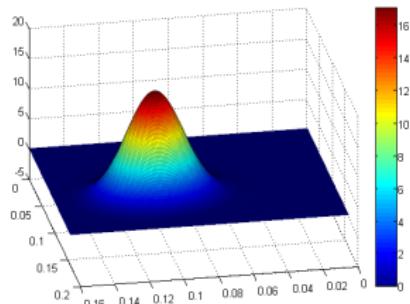
<sup>8</sup>Beale, J. Thomas, and Andrew Majda. High order accurate vortex methods with explicit velocity kernels. Journal of Computational Physics 58.2 (1985): 188-208.

## Kernel De-singularization: Two Viewpoints

- ▶ Lagrangian: Singularity in Biot-Savart kernel generates non-physical velocities near vortex points, de-singularize by approximation of Dirac delta function using finite cutoff radius  $\delta$ .
- ▶ Lagrangian: Heuristically, one deals with vortex “blobs”.
- ▶ Eulerian: Vorticity is not confined to points, but spatially varying; what purpose does de-singularization serve? Strictly practical.
- ▶ Eulerian: While Biot-Savart kernel converges analytically, no guarantees numerically. Quadrature assumes polynomial basis appropriate approximation.
- ▶ Eulerian: De-singularization means to an end, improve quadrature convergence qualities.

# Convergence of Biot-Savart Integral

- ▶ Nearly singular nature of Biot-Savart kernel difficult to integrate numerically
- ▶ Smaller the cutoff radius, larger quadrature errors
- ▶ Larger cutoff radius, larger velocity approximation errors
- ▶ Cutoff radius should be selected to balance both



## Discontinuous Galerkin (DG)

- Why DG vs others?

We seek to solve the PDE Eqn. (6). An approximate solution  $\tilde{\omega}$  has residual

$$\frac{\partial \tilde{\omega}}{\partial t} + \frac{\partial f}{\partial x_i} = R(x) \quad (10)$$

1-D case, vorticity sources omitted for simplicity.

DG approach<sup>9</sup>: minimize the  $L^2$  norm by orthogonal projection of residual onto approximating space. Complete basis made from test functions  $\phi_j$ , so:

$$\int_{\Omega} R(x) \phi_j \, dx = 0 \quad \text{for all } j \quad (11)$$

Substituting residual with conservation PDE yields:

$$\int_{\Omega} \frac{\partial \tilde{\omega}}{\partial t} \phi_j \, dx + \int_{\Omega} \frac{\partial f(\tilde{\omega})}{\partial x} \phi_j \, dx = 0 \quad \text{for all } j \quad (12)$$

---

<sup>9</sup>W.H Reed and T.R. Hill. Triangular mesh methods for the neutron transport equation. 1973.

## Discontinuous Galerkin (DG)(cont.)

Use same space for both test functions and approximation, so Mth order vorticity approximation is

$$\omega(x, t) \approx \tilde{\omega}(x, t) = \sum_{i=0}^M a_i(t) \psi_i(x) \quad (13)$$

Substitute into Eqn. (12) and integrate by parts second term:

$$\sum_{i=0}^M \left[ \frac{da_i(t)}{dt} \int_{x_L}^{x_R} \psi_i(x) \phi_j(x) dx \right] + f \phi_j(x) \Big|_{x_L}^{x_R} - \int_{x_L}^{x_R} f(\tilde{\omega}) \frac{d\phi_j(x)}{dx} dx = 0 \quad (14)$$

Note: Local solution to PDE on an element.

## Discontinuous Galerkin (DG)(cont.)

All local solutions decoupled, also vorticity multiply defined at overlapping element boundaries. Recover global solution and treat element boundaries via an upwind flux function<sup>10</sup>(similar to finite volume method)

$$\hat{f}_{upwind}(x^+, x^-) = u\{\{\tilde{\omega}\}\} + \frac{|u|}{2}[[\tilde{\omega}]] \quad (15)$$

where  $\{\{\omega^+\}\} = \frac{\omega^+ + \omega^-}{2}$  and  $[[\omega]] = \omega^+ - \omega^-$

Applying change of variables to map to arbitrary computational element  $X \in [-1, 1]$  results in:

$$\frac{\Delta x}{2} \sum_{i=0}^M \left[ \frac{da_i}{dt} \int_{-1}^1 \psi_i \phi_j \, dX \right] + \hat{f} \phi_j \Big|_{x_L}^{x_R} - \int_{-1}^1 f(\tilde{\omega}) \frac{d\phi_j}{dX} \, dX = 0 \quad (16)$$

---

<sup>10</sup>Hesthaven, Jan S., and Tim Warburton. Nodal discontinuous Galerkin methods: algorithms, analysis, and applications. Vol. 54. Springer Science & Business Media, 2007.

# Solver Overview

```
Define problem parameters
Define solver parameters
Calculate derived solver parameters
Setup intitial conditions
Initialize solver
%Time stepping
for t=0 to end
    if datalog?=yes
        save system state to file and plot
    end
    %Loop through RK stages
    for s=1 to last_stage
        %For elements above threshold
        for each vorticity source
            calculate velocity contributions
        end
        %Calculate semi-discrete system terms
        interpolate boundary_vorticity
        calculate numerical_fluxes
        calculate total_surface_flux
        calculate internal_stiffness_flux

        vorticity_rate_of_change=...
            internal_stiffness_flux - total_surface_flux

        RK_stage= (RK_coeff_a*RK_stage) +...
            (time_step * vorticity_rate_of_change)
        vorticity= vorticity + RK_coeff_b * RK_stage
    end
end
```

## Methodology: Method Specific Choices

- ▶ Choose basis functions to be the interpolating Lagrange polynomials

$$\psi_i(x) = \ell_i(x) = \prod_{\substack{p=0 \\ p \neq i}}^M \frac{x - x_p}{x_i - x_p} \quad (17)$$

- ▶ Choose vorticity interpolation nodes to be the Gauss-Legendre points, collocate with quadrature points, results in simplification of mass matrix

$$\int \ell_i(x) \ell_j(x) dx = \delta_{ij} w_j \quad (18)$$

- ▶ Take line-DG<sup>6</sup> approach, form 2D basis as tensor product of 1D bases

$$f(x, y) \approx \left[ \sum_{j=0}^L z_j \ell_j(y) \right] \times \left[ \sum_{i=0}^M z_i \ell_i(x) \right] = \sum_{j=0}^M \sum_{i=0}^M z_{ij} \ell_j \ell_i = \sum_{j=0}^M z_{ij} \ell_j \sum_{i=0}^M \ell_i \quad (19)$$

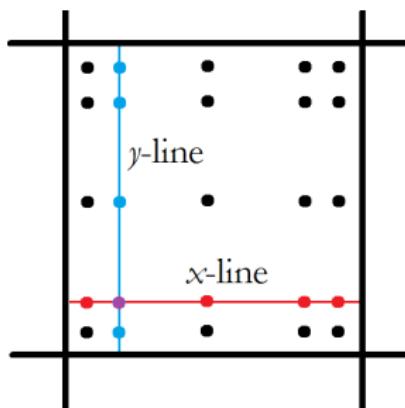
## Method Specific Choices(cont.)

- The PDE is now solved along each tensor direction

$$\frac{\Delta x}{2} \sum_{i=0}^M \left[ \frac{dz_{ij}}{dt} \int_{-1}^1 \ell_i \ell_j dX \right] + \hat{f} \ell_j \Big|_{x_L}^{x_R} - \int_{-1}^1 f(\tilde{\omega}) \ell'_j dX = 0 \quad (20)$$

- The rate of change at each node is the sum of the contribution along each tensor direction

$$\frac{\partial \omega_{ij}}{\partial t} = \left( \frac{\partial \omega_{ij}}{\partial t} \right)_{x-line} + \left( \frac{\partial \omega_{ij}}{\partial t} \right)_{y-line} \quad (21)$$



## Velocity Evaluation

- ▶ Calculate velocity at all velocity nodes by summation of contribution from all vorticity using de-singularized kernel and Gauss-Legendre quadrature

$$u(x^*) = \sum_{E=1}^{N_{mask}} [\omega_{pre}]^T K_\delta(x^* - [x_E]) \quad (22)$$

$$\omega_{pre} = [\omega(x_E)]. * [w_i \otimes w_j] \quad (23)$$

- ▶ Pre-multiplication of particular elemental vorticity source by outer product of Gauss-Legendre quadrature weights and computing as vector-vector product saves great deal of computational effort.
- ▶ Kernel values pre-calculated for “generalized” reference frame

## Explicit Time-Stepping

- ▶ Method of lines approach to semi-discrete system
- ▶ Low-storage explicit Runge-Kutta method used
- ▶ 14 stage-4th order “NRK14C” used to maximize stable time-step<sup>7</sup>
- ▶ Stability region almost 1.9 times larger per stage along negative real axis, chief consideration for dissipative upwind DG schemes

## Results: Validating Test Cases

- ▶ Perlman: Stationary vortex<sup>11</sup>

$$\omega(z) = (1 - |z|^2)^7, \quad |z| \leq 1 \quad \omega(z) = 0, \quad |z| > 1 \quad z^2 = x^2 + y^2 \quad (24)$$

- ▶ Has analytical solution to vorticity and velocity fields

$$u(z, t) = f(|z|) \begin{pmatrix} y \\ -x \end{pmatrix} \quad (25)$$

where

$$f(|z|) = \begin{cases} -\frac{1}{16|z|^2}(1 - (1 - |z|^2)^8) & |z| \leq 1 \\ -\frac{1}{16|z|^2} & |z| > 1 \end{cases}$$

---

<sup>11</sup>M. Perlman. On the accuracy of vortex methods, J. Comput. Phys. 59 (1985) 200–223.

## Comparison of Cutoff Radius Effects(P)

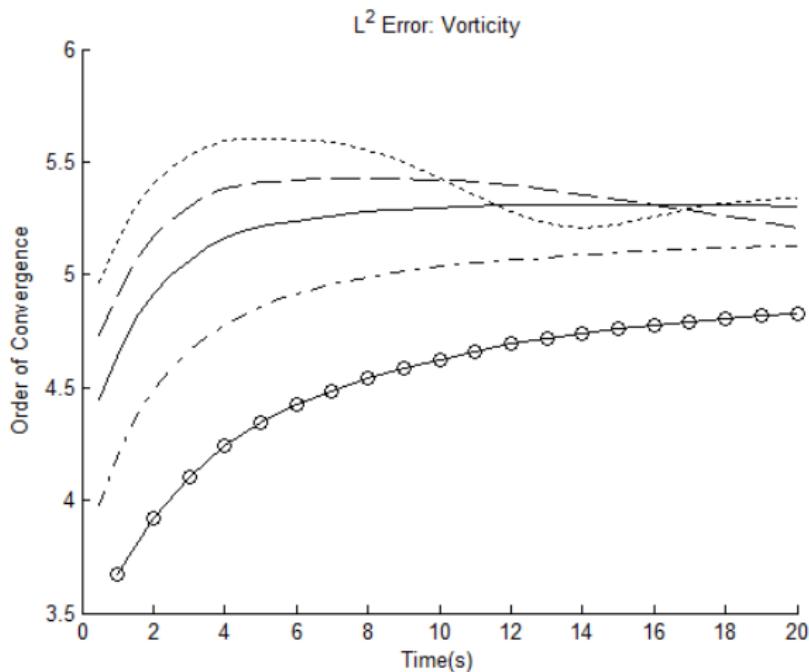


Figure: Comparison of convergence effects on vorticity by cutoff radius for  $\delta/\Delta x=0.1$ (..),  $0.2$ (- -),  $0.3$ (-),  $0.5$ (-.),  $0.9$ (-o).

## Comparison of Cutoff Radius Effects(P)(cont.)

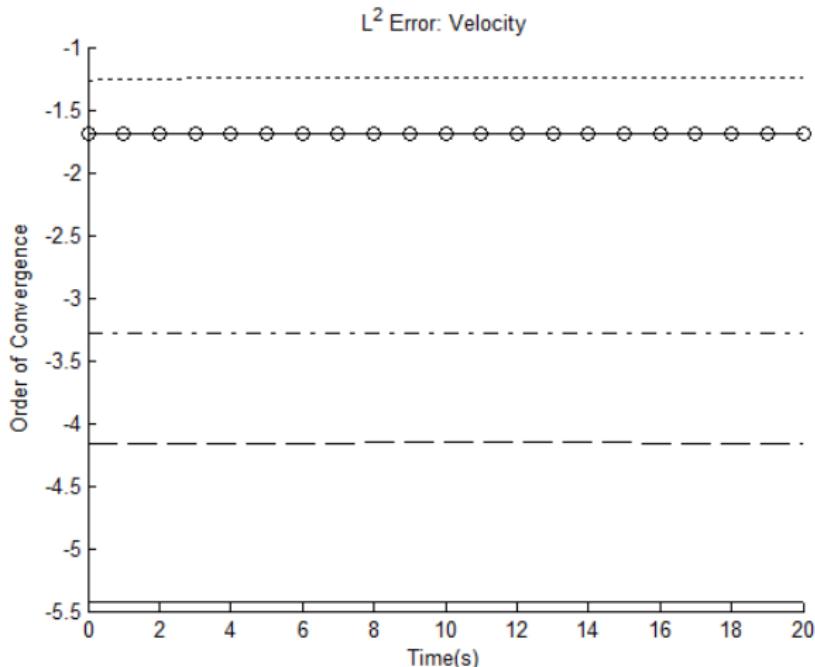


Figure: Comparison of convergence effects on velocity by cutoff radius for  $\delta/\Delta x = 0.1(..), 0.2(- -), 0.3(-), 0.5(- - -), 0.9(-o)$ .

## Observed Convergence Rate of Various Orders(P)

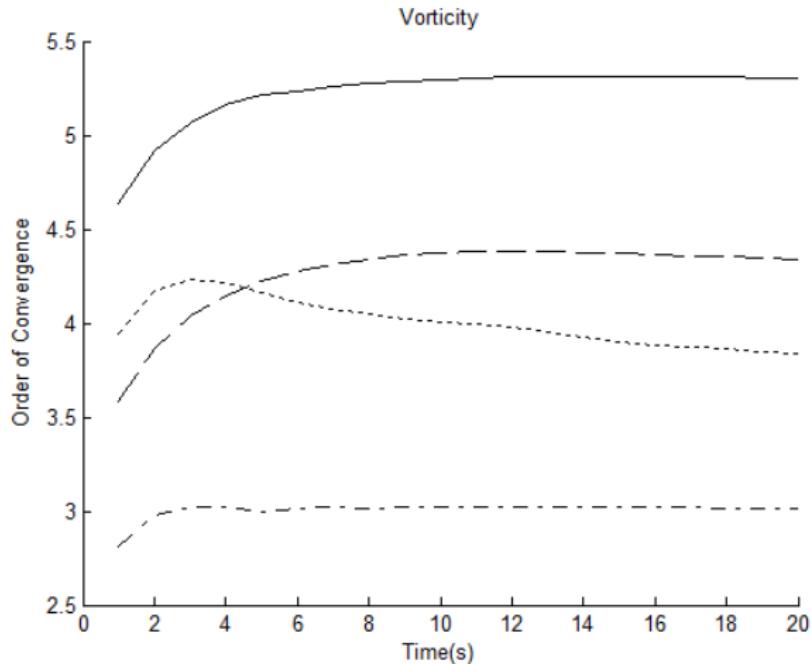


Figure: Comparison of convergence rate for various order methods:  
3rd(-.), 4th(..), 5th(- -), and 6th(-).

## Validating Test Cases

- ▶ Strain: Interacting vortex patches<sup>12</sup>

$$\omega(x, y, 0) = \sum_{j=1}^m \Omega_j \exp(-((x - x_j)^2 + (y - y_j)^2)/\rho_j^2) \quad (26)$$

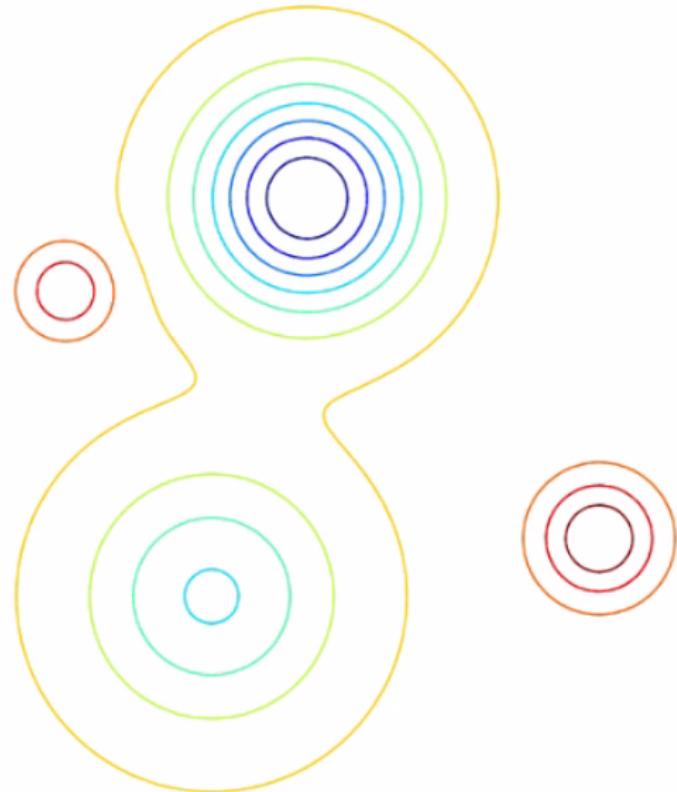
Table: Interacting Vortex Patch Parameters

j	$x_j$	$y_j$	$\rho_j$	$\Omega_j$
1	-0.6988	-1.7756	0.6768	-0.4515
2	1.4363	-1.4566	0.3294	0.4968
3	-0.1722	0.4175	0.5807	-0.9643
4	-1.5009	-0.0937	0.2504	0.3418

---

<sup>12</sup>J. Strain. 2D vortex methods and singular quadrature rules. Journal of Computational Physics 124.1 (1996): 131-145.

## Strain Test Case, t=0



## Approximated Convergence Rate(S)

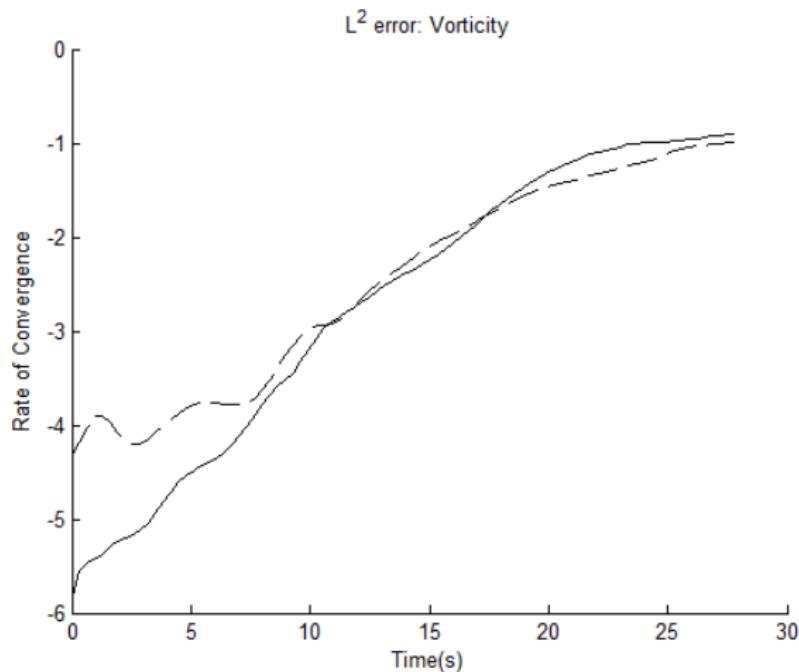


Figure: Dependency of rate of convergence on order of method, for a 4th order(--) and 6th order(-) method.

## Approximated Convergence Rate(S)(cont.)

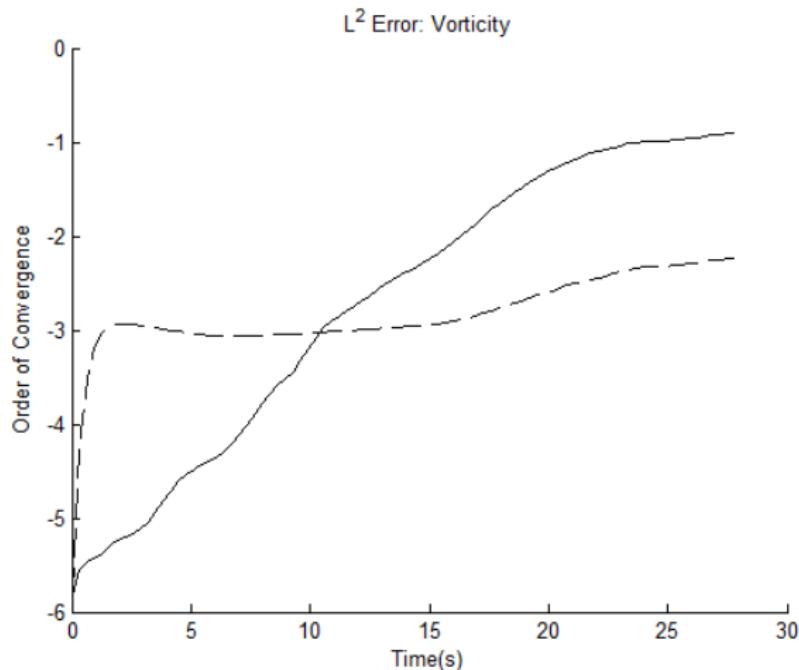


Figure: Dependency of rate of convergence on cutoff radius in sixth order method, for a  $\delta/\Delta x = 0.5$ (--) and  $0.25$ (-).

## Qualitative Comparison w/ Previous Work<sup>12</sup>

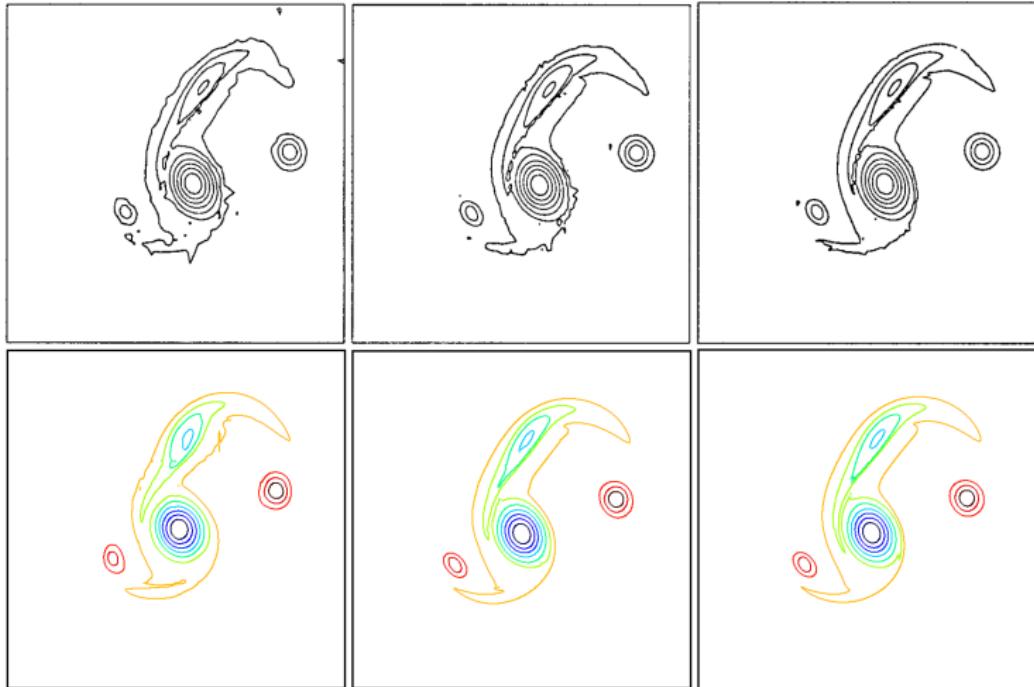


Figure: Comparison of Strain's results with present method  $t=28$ . Left to right, top to bottom, DOF = 6400, 12800, 25600; 3136, 7056, 63504. Reprinted with permission from Elsevier.

## Validating Test Cases

- ▶ Koumoutsakos: Elliptical vortex<sup>13</sup>

$$\omega^{II}(x, y, 0)_{mod} = 20(1 - ((x/a)^2 + (y/b)^2)^2 / 0.8^4) \quad a = 1, b = 2 \quad (27)$$

---

<sup>13</sup>P. Koumoutsakos. Inviscid axisymmetrization of an elliptical vortex, J. Comput. Phys. 138 (1997) 821–857.

## Qualitative Comparison of Vorticity(K)

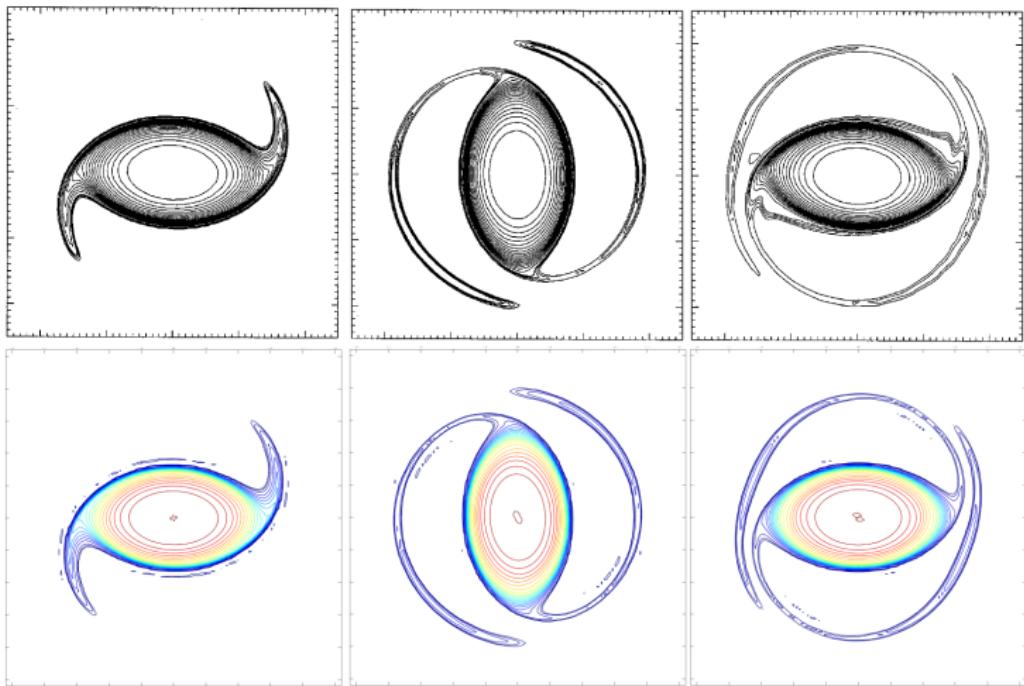


Figure: Comparison of vorticity, Koumoutsakos (top) and present method (bottom). From left to right, top to bottom  $t=1, 2, 4$ ;  $0.80, 1.93, 2.32$ . Reprinted with permission from Elsevier.

## Qualitative Comparison of Vorticity(K)(cont.)

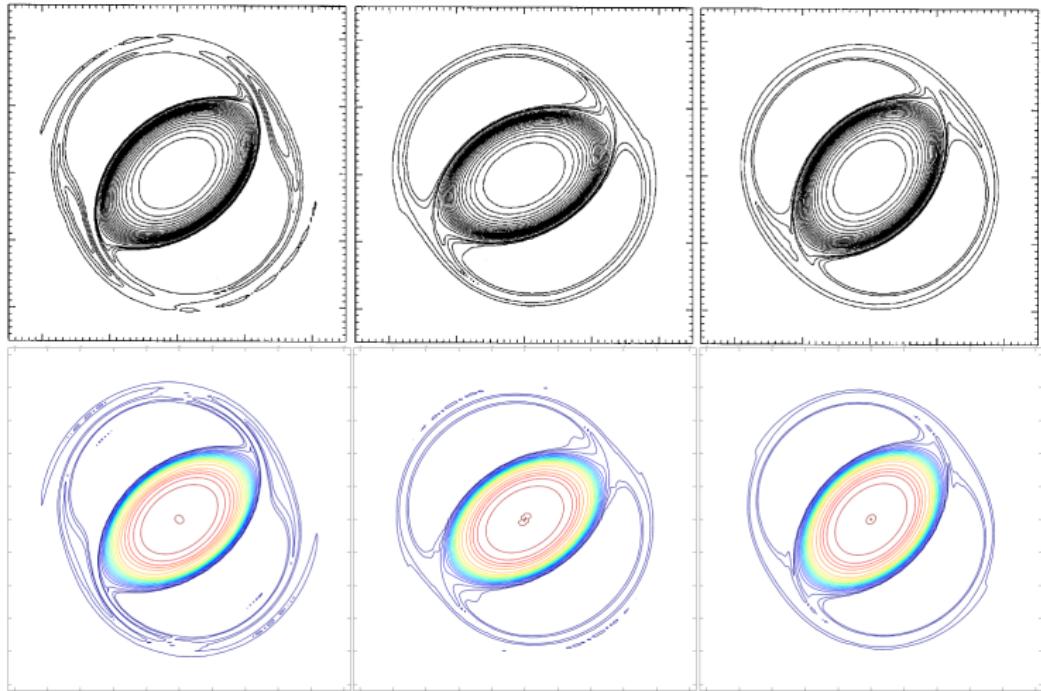
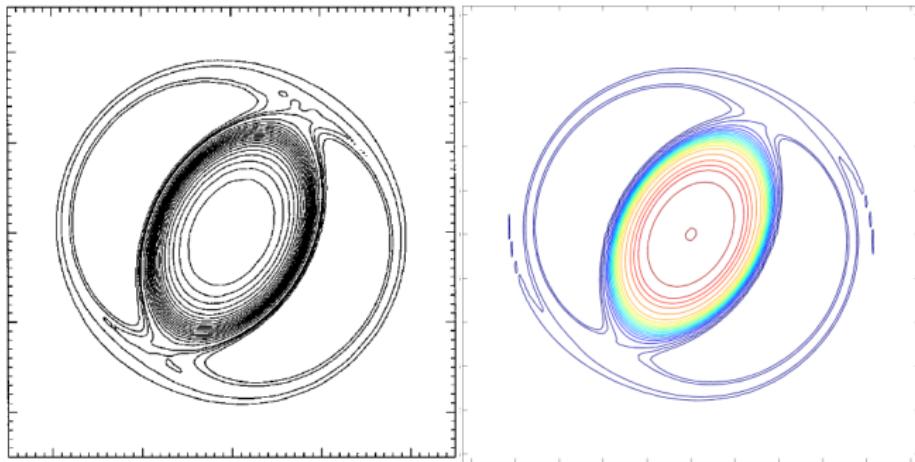


Figure: Comparison of vorticity, Koumoutsakos (top) and present method (bottom). From left to right top to bottom,: t=6, 12, 18; 5.94, 11.99, 17.94. Reprinted with permission from Elsevier.

## Qualitative Comparison of Vorticity(K)(cont.)



**Figure:** Comparison of vorticity, Koumoutsakos (left) and present method (right). From left to right:  $t=24$ ;  $23.98$ . Reprinted with permission from Elsevier.

# Comparison of Aspect Ratio(K)

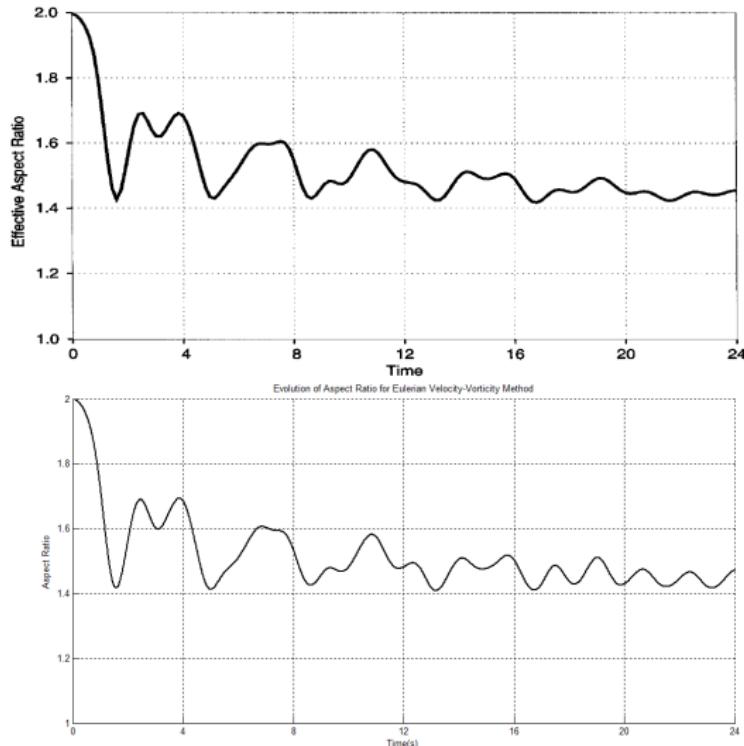


Figure: Comparison of effective aspect ratio, Koumoutsakos (top) and present method (bottom). Reprinted with permission from Elsevier.

## Discussion: Validation

- ▶ Analytical: With proper choice of kernel, cutoff radius, stage-wise velocity evaluation, and matching velocity order method able to obtain solution within discretization error of exact solution for velocity and vorticity.
- ▶ Qualitative: Excellent agreement in Strain test case even for test with far fewer DOFs
- ▶ Good agreement in Koumoutsakos test case, with some minor deviation towards end of period studied and minor artifacting at vortex body boundaries.
- ▶ DOFs required about equal to Koumoutsakos's results, despite being higher-order method.
- ▶ Arm filaments and vortex body boundary challenging for polynomial basis functions, discontinuous derivatives affect bound on interpolation error.

## Convergence Rate

- ▶ In Perlman test case, capable of near optimal convergence rates for stationary vortex test case.
- ▶ Half-order less convergence rate for higher order methods due to lack of as many vorticity derivatives.
- ▶ Non-optimal approximated convergence observed in Strain test case.
- ▶ Choice of two options, cutoff radius too small and gradual decay of convergence to first order from optimal. Cutoff radius too large, constant but non-optimal order of convergence.
- ▶ Tempting to blame smallest feature size or challenging evolution evolution for convergence, but testing with pairs of Perlman vortices shows same issues.

## Exact Biot-Savart Integration

- ▶ Ideally: More accurately integrate B-S kernel, w/o extra error from de-singularization approximation.
- ▶ Example: Calculate velocity component at particular point:

$$u_y(T_x) = \int \int \frac{\mathbf{x} - T_x}{2\pi r^2} \omega(\mathbf{x}) d\mathbf{x} \quad (28)$$

- ▶  $\omega$  is actually Lagrange interpolation of vorticity (with interp. values  $z_{ij}$ ), so substitute:

$$u_y(T_x) = \frac{1}{2\pi} \sum_i \sum_j z_{ij} \int \int \frac{x - T_x}{r^2} \ell_i(x) \ell_j(y) dx dy \quad (29)$$

- ▶ Note: Variable part of interp. occurs outside integral. We can pre-calculate integrals for all combinations and store.
- ▶ Yields modified quadrature:

$$u(T_-) = \frac{1}{2\pi} \sum_i \sum_j z_{ij} W_{(-, T_x, T_y, i, j)} \quad (30)$$

## Exact B-S Integr.: Modified Kernel Values

- ▶ How do these weights vary spatially wrt target point? Divide special weights by tensor product of standard Gauss-Legendre:

$$u_x = \frac{1}{2\pi} \sum_i \sum_j z_{ij} \frac{W_{(i,j)}}{W_{GL}} W_{GL} \quad (31)$$

- ▶ Note: Can break into two parts;  $\omega$  interpolation ( $z_{ij}$ ) and B-S kernel values. Heuristically:

$$u_x = \frac{1}{2\pi} \sum_i \sum_j z_{ij} \tilde{k}_{ij} W_{GL} \quad (32)$$

- ▶ We have recovered something that looks like “modified” kernel values. How do they vary spatially?

## Exact B-S Integr.: Modified Kernel Values(cont.)

- Modifications to standard kernel values are local; neighboring/adjacent elements. Promising for FMM acceleration.

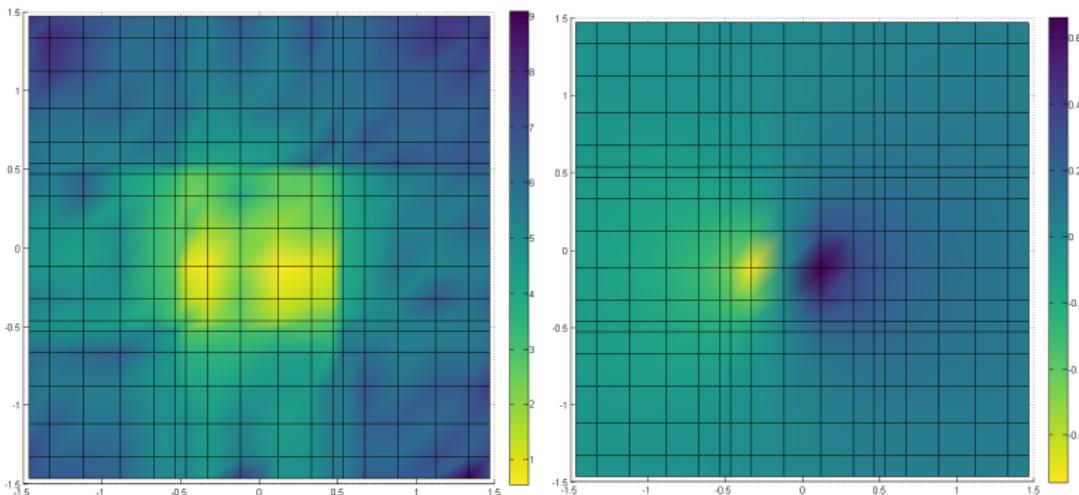


Figure: Number of digits in common between modified and original kernel(left), original kernel values(right); 5th order nodal set.

## Exact B-S Integr.: Modified Kernel Values(cont.)

- ▶ Apply modified scheme to same Strain test case previously studied, but now with exact kernel integration.

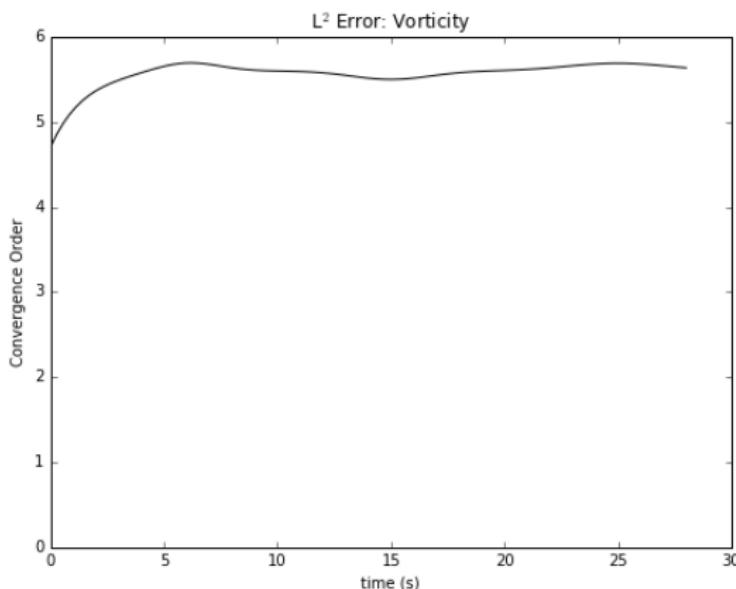


Figure: Dramatically improved convergence order with exact kernel integration scheme, same parameters as previous Stain test case plots.

## Modified Kernel Scheme Drawbacks, and Future Directions

- ▶ Drawback: Calculation/storage of modified kernel values for unstructured meshes non-trivial
- ▶ Drawback: Curse of dimensionality causes explosion of number of modified kernel values necessary for even moderate orders for 3-D problems.
- ▶ Alternate approach: Use QBX for evaluation of singular Biot-Savart integral.
- ▶ Requires: Volume QBX evaluation. What is the proper choice of kernel where expansion "center" can be both "close" to the target point, but sufficiently smoother to achieve high-order scheme.
- ▶ Requires: Fast algorithm version for unstructured meshes needs FMM capable of dealing with interplay between mesh elements (and associated discretizations) and the (potentially adaptive) FMM tree.