

Vortex Dominated Flows: A High-Order, Conservative Eulerian Simulation Method

Josh Bevan

March 23, 2015

Abstract

A high-order, conservative Eulerian method is presented for the simulation of vortex dominated inviscid fluid flows. The primitive variable Navier-Stokes equations are recast in the velocity-vorticity form to explicitly enforce conservation of vorticity. The advection of the vorticity is then calculated via a two-step process each time-step: the velocity field is determined by evaluation of the Biot-Savart integral, and then a line-based discontinuous Galerkin (DG) Eulerian spatial discretization scheme is applied. The accuracy and convergence of this method is examined for test cases where an analytical solution exists, as well as more challenging test cases which lack an analytical solution. Of particular interest is the influence the discretization of the calculated velocity field has on the performance of the method.

Acknowledgments

Contents

List of Tables	v
List of Figures	vi
1 Introduction	1
1.1 Problem Formulation	1
1.2 Chosen Methods	1
1.3 Thesis Structure	2
2 Theory	3
2.1 Navier-Stokes: Velocity-vorticity form	3
2.2 Velocity Field Evaluation: The Biot-Savart Integral	4
2.3 Discontinuous Galerkin	5
3 Methodology	9
3.1 Overall Solver Structure	9
3.2 Algorithms of Note	9
3.3 Validation and Convergence of Proposed Methods	9
4 Results	10
4.1 Analytical Test Cases	10
4.2 Elliptical Blob	10
4.3 Arbitrary Patch	10
4.4 Dipole(misuse of term?)	10
4.5 Vortical System	10
5 Discussion	11
5.1 Analytical Validation of Model	11
5.2	11
6 Conclusion	12
6.1 Weak Formulation	12
7 Recommendations	13
8 Literature Cited	14

List of Tables

List of Figures

8.1	Dirty preliminary lit cited.	15
-----	--------------------------------------	----

1 Introduction

1.1 Problem Formulation

- DNS expensive
- For inviscid vortex dominated flows \rightarrow reformulate from primitive variables to velocity-vorticity
- Existing methods: Lagrangian vortex particle methods [Carley, Strain, Leonard] and VTM [Brown]
- Difficulty extending to high order: Particle methods (re-meshing), VTM (extended stencil)
- Available high order methods unsuitable: FD/FV (extended stencil and smearing), FE (non-conservative, ill-suited for hyperbolic), Spectral (Globally defined vs concentrated sparse vorticity)

1.2 Chosen Methods

- DG: conservative, local, flux funs handle hyperbolicity
- Line DG easy implementation for hexahedral meshes. Tensor product points allow possibility of easy biasing along principal flow directions. Allow easy translation of 1D methods for multidimensional domains
- Direct evaluation of BS integral: allows investigation of local order refinement effects on global convergence

1.3 Thesis Structure

- Theory: Develop method specific mathematics (DG, VTM, BS, etc)
- Methodology: Cover relevant notable implementation specifics (e.g. Solver structure, non-trivial algorithms), present convergence test structure (analytical cases, elliptical blob, 2-D "ring", several interacting patches) vs (convergence rate of: constant order velocity field, linear far field velocity [far= 1 element or X element separation], variable far field velocity order [heuristic?])
- Results: Results of matrix of tests (summary with aggregate plots, and selected results)
- Discussion: Analytical validation, comparison of velocity field fidelity test, comments on effects of global convergence

2 Theory

2.1 Navier-Stokes: Velocity-vorticity form

If the quantity *vorticity* is defined as

$$\omega = \nabla \times u \quad (2.1)$$

Then the traditional form of the Navier-Stokes equations can be recast, assuming an inviscid incompressible flow

$$\frac{\partial \omega}{\partial t} + u \cdot \nabla \omega - \omega \cdot \nabla u = S(x, t) \quad (2.2)$$

There are several benefits to the recast form: explicitly conserving vorticity, sparseness, pressure term need not be solved for... [elaborate]

Restricting to examining 2-D distributions of vorticity, several simplifications can be made. The originally vectorial vorticity becomes a scalar quantity, all vorticity is directed normal to the plane. As a result, the vortex stretching term in (2.2) becomes zero. The only non-zero component of ω is in the z-direction, however the gradient of the velocity field is zero in the z-direction, so the product is therefore zero. The result is

$$\frac{\partial \omega}{\partial t} + u \cdot \nabla \omega = S(x, t) \quad (2.3)$$

or if instead the second term is expressed in terms of the flux of the vorticity (where $f_i(\omega) = u_i \omega$):

$$\frac{\partial \omega}{\partial t} + \frac{\partial f}{\partial x_i} = S(x, t) \quad (2.4)$$

2.2 Velocity Field Evaluation: The Biot-Savart Integral

We shall decide that ω is the quantity of interest in the solution method. However, if this is the case then the question must be posed: how does one determine the velocity field? For an incompressible flow

$$\nabla^2 u = -\nabla \times \omega \quad (2.5)$$

If inverted, the Biot-Savart integral is obtained

$$u(x^*) = \int_{\Omega} K(x^*, x) \times \omega(x) dx \quad (2.6)$$

with the singular Biot-Savart kernel

$$K(x^*, x) = \frac{-1}{4\pi} \frac{x^* - x}{|x^* - x|^3} \quad (2.7)$$

There are several important points to note that are a consequence of this inversion. First, rather than solving the Poisson equation for the entirety of the domain we choose to evaluate the velocity at some subset. For the purposes of advecting the velocity we will only need velocities near the vorticity itself. However, if the number of required velocity evaluation points is roughly proportional to the N DOFs of our vorticity approximation, then it might be expected for the velocity calculations to scale as $\mathcal{O}(N^2)$.

There are numerous methods to reduce the computational complexity of similar N-body type problems to $\mathcal{O}(N \log N)$ or $\mathcal{O}(N)$. Cyclic reduction [Schumann, Sweet 1976], tree-codes [Lindsay/Krasny][Barnes-Hut], and FMM [Greengard] are all possibilities. However, because the velocity field calculation method is essentially decoupled from the discretization of the PDE, there is no a priori assurance that the velocity calculated is of sufficient fidelity to ensure convergence of the overall method (let alone convergence at the order one might expect based on solely the discretiza-

tion). For maximum flexibility in investigating this dependency of overall convergence on the calculated velocity field we shall eschew more efficient techniques so that we can directly control the fidelity of the velocity field.

The second point to consider regarding the Biot-Savart integral is the singular nature of the exact kernel. Lagrangian point vortex methods have the benefit that the singularity and it's associated non-physical velocities occur in a relatively small region; they assume that the advective effect of the point vortex on itself is negligible [hazy on this, clarify some]. Additionally they will frequently de-singularize the kernel by introducing a core function. The classical one used is the Rosenhead-Moore kernel [ref here]:

$$K(x^*, x) = \frac{-1}{4\pi} \frac{x^* - x}{(|x^* - x|^2 + \sigma^2)^{3/2}} \quad (2.8)$$

The high-order Eulerian approach taken here however means that the Biot-Savart integral diverges everywhere within any of the extended vorticity patches thanks to self-influence.

Expense of calculation (% of total program?), Self-terms: Subsplitting vs GL quadrature skip vs kernels

2.3 Discontinuous Galerkin

In order to solve (2.9) we adopt a method-of-lines approach. We will first spatially discretize the system to obtain the semi-discrete system, then we use an explicit time discretization method to march forward in time. Note that (2.9) has the form of a scalar conservation law, with ω being the conserved quantity. The velocity field that advects the conserved quantity has been calculated by evaluation of the Biot-Savart integral for the current timestep.

Our spatial discretization is at best an approximate solution to (2.9), that we shall denote as $\tilde{\omega}(x, t)$. Some residual will remain for the approximate solution of the PDE

at any time t

$$\frac{\partial \tilde{\omega}}{\partial t} + \frac{\partial f}{\partial x_i} = R(x) \quad (2.9)$$

where we have shown a 1-D case and omitted vorticity sources for simplicity.

The Discontinuous Galerkin (DG) approach attempts to approximately satisfy the PDE in the following way: Seek the best approximation in a finite vector test space \mathbb{W}_h in the L^2 norm sense. Minimize the L^2 norm by an orthogonal projection of the residual onto \mathbb{W}_h . Form a complete basis for \mathbb{W}_h with a set of test functions $\phi_j \in \mathbb{W}_h$, such that the orthogonal projection satisfies:

$$\int_{\Omega} R(x) \phi_j dx = 0 \quad \text{for all } j \quad (2.10)$$

Substituting the residual for our conservation PDE yields:

$$\int_{\Omega} \frac{\partial \tilde{\omega}}{\partial t} \phi_j dx + \int_{\Omega} \frac{\partial f(\tilde{\omega})}{\partial x} \phi_j dx = 0 \quad \text{for all } j \quad (2.11)$$

Choose the space of piecewise polynomials for the finite dimensional approximation space, and decompose the global domain into K elements where the local approximation space \mathbb{V}_h^k has basis functions $\psi_i^k(x)$ over the local domain $x \in [x_L, x_R]$ (henceforth we drop the element index k for brevity, except when describing two distinct elements that must be distinguished). Note that continuity of vorticity is not enforced across elements, and that the local approximation spaces for each element are independently defined. Finally, take the Bubnov-Galerkin choice of setting the test space to be the same as the approximation space so that $\phi_j = \psi_i \in \mathbb{V}_h$ if $i = j$. The local M th order approximation to vorticity takes the form

$$\omega(x, t) \approx \tilde{\omega}(x, t) = \sum_{i=0}^M a_i(t) \psi_i(x) \quad (2.12)$$

where a_i is some set of coefficients that are to be determined. The elemental approximating PDE is then

$$\sum_{i=0}^M \left[\frac{da_i(t)}{dt} \int_{x_L}^{x_R} \psi_i(x) \phi_j(x) dx \right] + \int_{x_L}^{x_R} \frac{\partial f(\tilde{\omega})}{\partial x} \phi_j dx = 0 \quad (2.13)$$

To reduce the smoothness requirements on the flux, integrate the second term by parts to yield

$$\sum_{i=0}^M \left[\frac{da_i(t)}{dt} \int_{x_L}^{x_R} \psi_i(x) \phi_j(x) dx \right] + f\phi_j(x) \Big|_{x_L}^{x_R} - \int_{x_L}^{x_R} f(\tilde{\omega}) \frac{d\phi_j(x)}{dx} dx = 0 \quad (2.14)$$

The local solution offers no way of recovering the global solution, until one realizes that the flux at the element boundaries is multiply defined between elements. The global solution is allowed to be piecewise discontinuous across boundaries, so there is no guarantee that the flux between neighboring elements would agree. To resolve this (and at the same time recover a global solution) define a numerical flux analogous to a Finite Volume Method (FVM) that takes as input the vorticity at the adjacent element boundaries. We will use an upwind flux, where defining the average as $\{\{\omega^+\}\} = \frac{\omega^+ + \omega^-}{2}$ and the jump as $[[\omega]] = \omega^+ - \omega^-$, yields

$$\hat{f}_{upwind}(x^+, x^-) = u\{\{\tilde{\omega}\}\} + \frac{|u|}{2} [[\tilde{\omega}]] \quad (2.15)$$

We would also like to map any element to a computational element by means of a mapping $x = g(X)$ and it's inverse $X = G(x)$. If we set the domain of our computational element to be $X \in [-1, 1]$ and the element size to be $\Delta x = x_R - x_L$, then the mapping is

$$x = g(X) = \frac{X+1}{2} \Delta x + x_L \quad , \quad X = G(x) = \frac{2(x - x_L)}{\Delta x} + 1 \quad (2.16)$$

Applying a change of variables using the mapping to (2.17), accounting for the numerical flux, and being sure to include the determinant of the Jacobian matrix that results from the mapping ($J = g'(X) = \Delta x/2$) in the first integral:

$$\frac{\Delta x}{2} \sum_{i=0}^M \left[\frac{da_i}{dt} \int_{-1}^1 \psi_i \phi_j dX \right] + \hat{f}\phi_j \Big|_{x_L}^{x_R} - \int_{-1}^1 f(\tilde{\omega}) \frac{d\phi_j}{dX} dX = 0 \quad (2.17)$$

(Note: no determinant of the Jacobian matrix appears in the second integral because one of the basis functions has been differentiated. When the chain rule is applied during the change of variables an extra $1/g'(X)$ appears that cancels out with the $g'(X)$ that occurred from the change of variables on the integral)

We have purposely left the flux in the second integral unevaluated in terms of the basis functions, as well as the choice of basis functions left as undecided. We defer these choices to a more in depth investigation in Methodology. We also have developed the equations past the initial PDE in a 1-D form. We will shortly show that the basis coefficients are chosen to be nodal values. The 2-D solution will be recovered following the Line-DG methodology [Line-DG Persson citation] element will be composed of a tensor product of two 1-D discretization

3 Methodology

3.1 Overall Solver Structure

IC and BC initialization, Lagrange pre-calculation, mass and stiffness matrices generation, 2 step spatial discretization, time discretization, post processing

3.2 Algorithms of Note

Vectorized Lagrange evaluation, Lagrange derivatives, etc

3.3 Validation and Convergence of Proposed Methods

(analytical cases, elliptical blob, 2-D "ring", several interacting patches) vs (convergence rate of: constant order velocity field, linear far field velocity [far= 1 element or X element separation], variable far field velocity order [heuristic?])

4 Results

4.1 Analytical Test Cases

4.2 Elliptical Blob

4.3 Arbitrary Patch

4.4 Dipole(misuse of term?)

4.5 Vortical System

5 Discussion

5.1 Analytical Validation of Model

Comparison of existing analytical solutions vs model (Euler vortex, 5th order poly, other Saffmann test cases)

5.2

6 Conclusion

6.1 Weak Formulation

7 Recommendations

8 Literature Cited

- 1 Triangular Mesh Methods for the Neutron Transport Equation 1973 ReedHill.pdf
- 2 A Sparse and High-Order Accurate Line-Based DG Method for Unstruct Meshes 2011 Persson.pdf
- 10 Quad and WF Choices for Collocation DGSEM 2010 KoprivaGassner.pdf
- 11 HighOrd Unifying DG Formul for NS Eqns on 3D Mixed Grids 2011 HagaGaoWang.pdf
- 20 Rotor Wake Modeling for Flight Dynamic Simulation of Helicopters 2000 Brown.pdf
- 21 Efficient High-Resolution Wake Modeling Using the VTM 2005 Brown.pdf
- 22 Predicting the wake structure of the HART II rotor using the VTM 2008 Brown.pdf
- 23 Simulation of wind turbine wake interaction using the vorticity transport model 2010 Brown.pdf
- 200 Unified Analysis of DG for Elliptic 2002 ArnoldBrezziCockburnMarini.pdf
- 210 High-Order Accurate DG FEM for Num Sol of Compress NS 1997 BassiRebay.pdf
- 230 DG FEM for Diff Probs 1999 BrezziMMPR.pdf
- 231 DG approx for Elliptic Probs 2000 BrezziMMPR.pdf
- 310 Multidomain Spectral Sol'n of Compress Visc Flows 1994 Kopriva.pdf
- 311 Conservative Staggered-Grid Chebyshev Multidomain Method for Compress Flows 1995 Kopriva.pdf
- 312 Conservative Staggered-Grid Chebyshev Multidomain Method for Compress Flows 2 Semi-Struct 1996 Kopriva.pdf
- 313 A Staggered-Grid Multidomain Spectral Method for the Compressible Navier–Stokes Equations 1998 Kopriva.pdf
- 410 Lagrangian Vortex Method for Unbounded Flows 2008 CarleyMoussa.pdf
- 411 Eval of BiotSavart Integrals on Tetrahedral Meshes 2008 Carley.pdf
- 420 2D Vortex Methods and Singular Quadrature Rules 1995 Strain.pdf
- 421 Fast Adaptive 2D Vortex Methods 1997 Strain.pdf
- 430 The evaluation of the Biot–Savart integral 2000 Suh.pdf
- 440 Integral Eqn Formul'n of the Eqns of Motion of an Incomp Fluid 1992 Uhlman.pdf
- 450 Efficient FMM accelerated vortex methods in three dimensions via the Lamb–Helmholtz decomposition 2013 GumerovDuraiswami.pdf
- 460 Evaluation of the Biot–Savart Integral for Deformable Elliptical Gaussian Vortex Elements 2006 Rossi.pdf
- 470 High-resolution simulations of the flow around an impulsively started cylinder using vortex methods 1995 Leonard.pdf
- 510 Generalized Gaussian Quadrature Rules for Singular and Nearly Singular Integrals 2009 HuybrechsCools.pdf
- 520 Rates of Convergence of Gaussian Quadrature for Sinular Integrands 1984 LubinskyRabinowitz.pdf
- 530 Quadrature of Integrands with a Logarithmic Singularity 1993 Crow.pdf
- 540 Convergence Characteristics and Computational Cost of Two Algebraic Kernels in Vortex Methods with a Tree-Code Algorithm 2009 Weem
- 550 Particle Method and Adaptive Treecode for Vortex Sheet Motion in Three-Dimensional Flow 2001 LindsayKrasny.pdf
- 910 Is Guass Quad Better than Clenshaw-Curtis 2008 Trefethen.pdf
- 1010 Spectral method of decoupling the vorticity and stream function for the incompressible fluid flows 1999 LiSun.pdf
- 1020 Galerkin Spectral Method for the Vorticity and Stream Function Equations 1999 AuteriQuartapelle.pdf

Figure 8.1: Dirty preliminary lit cited.